

Інформатика

Казанцева О. П., Стеценко І. В.

«ІНФОРМАТИКА»

ПІДРУЧНИК ДЛЯ 8 КЛАСУ
загальноосвітніх навчальних закладів



ТЕРНОПІЛЬ
НАВЧАЛЬНА КНИГА – БОГДАН
2016

УДК 004(075.3)
ББК 32.97я72
К 14

Казанцева О. П.

К 14 Інформатика : підручник для 8 кл. загальноосвіт. навч. закл. / О.П. Казанцева, І.В. Стеценко. — Тернопіль : Навчальна книга – Богдан, 2016. — 304 с. : іл.

ISBN 978-966-10-4480-6

УДК 004(075.3)
ББК 32.97я72

*Охороняється законом про авторське право.
Жодна частина цього видання не може бути відтворена
в будь-якому вигляді без дозволу автора чи видавництва.*

ISBN 978-966-10-4480-6

© Казанцева О.П., Стеценко І.В., 2016
© Навчальна книга – Богдан, 2016



УМОВНІ ПОЗНАЧЕННЯ



інтерактивна
електронна складова



відеоматеріали

Піктограмами  ,  у підручнику позначено ті його складові, які можна відкрити в електронній версії за посиланням:

<http://www.bohdan-digital.com/edu>.

Шановні восьмикласники!

Цей підручник допоможе вам продовжити вивчення інформатики. Матеріал поділений на дев'ять розділів. У підручнику описано хід виконання всіх передбачених програмою практичних робіт, створено файли-заготовки до вправ і практичних робіт (знаходяться під відповідним ім'ям на диску інтерактивної складової підручника в папці **Files**). У першому розділі ви продовжите докладніше вивчати інформаційні процеси, зокрема обробку інформації та кодування даних. У другому розділі ви дізнаєтеся про характеристики програмного забезпечення, призначення складових комп'ютера та параметри їх підбору під потребу користувача. Вивчаючи матеріал третього розділу, ви будете працювати в текстовому редакторі MS Word. З четвертого розділу ви дізнаєтеся про мультимедійні продукти і технології. Із п'ятого розділу ви розпочнете і в шостому продовжите вивчати основи програмування в навчальному середовищі Visual Basic. У сьомому розділі ви продовжите роботу в табличному процесорі MS Excel, яку розпочали в сьомому класі. У восьмому розділі будете розв'язувати компетентнісні задачі, а на завершення підготуєте проект на вибрану тему.

ПУТІВНИК ПІДРУЧНИКОМ

У рубриці підсумовано матеріал параграфу і виокремлено основні поняття і терміни

Пропоновані питання вимагають творчого пошуку і поглиблених знань:
* — достатній рівень,
** — високий рівень навчальних досягнень

Практичні завдання дають можливість закріпити вивчений матеріал, працюючи за комп'ютером

§29. Використання перемикачів і прапорців

Висновки

Виведення зображення з файла здійснюється завдяки елементу керування **PictureBox**.

Основна властивість **PictureBox** — **Image**, що визначає саме зображення **SizeMode**, яка встановлює спосіб розміщення рисунка.

Вибір властивостей елемента керування **PictureBox** здійснюють за допомогою меню.

Контрольні запитання та завдання

1. Що таке розділ? За яких умов створюється новий розділ?
2. Що таке колонтитул? Які є види колонтитулів?

Питання для роздумів

- 1*. Що сприяло поширенню персональних комп'ютерів?
- 2*. Які ви бачите перспективи розвитку обчислювальної техніки?

Завдання для досліджень

- 1". Знайдіть інформацію про алгоритми стиснення Лемпеля — Зів — Велча (англ. Lempel — Ziv — Welch, LZW) і дізнайтесь, як він працює.

Вміщені запитання і завдання дають змогу перевірити засвоєні знання

Ці завдання розраховані на учнів, які захоплюються інформатикою

ПРАКТИЧНА РОБОТА №1

Розв'язування задач на визначення довжини двійкового коду даних різних типів


Завдання. Розв'язати задачі на визначення довжини двійкового коду.

ВПРАВА 21



Завдання. Створити анімацію з низки фотографій, скориставшись онлановим сервісом для створення GIF-анімації.

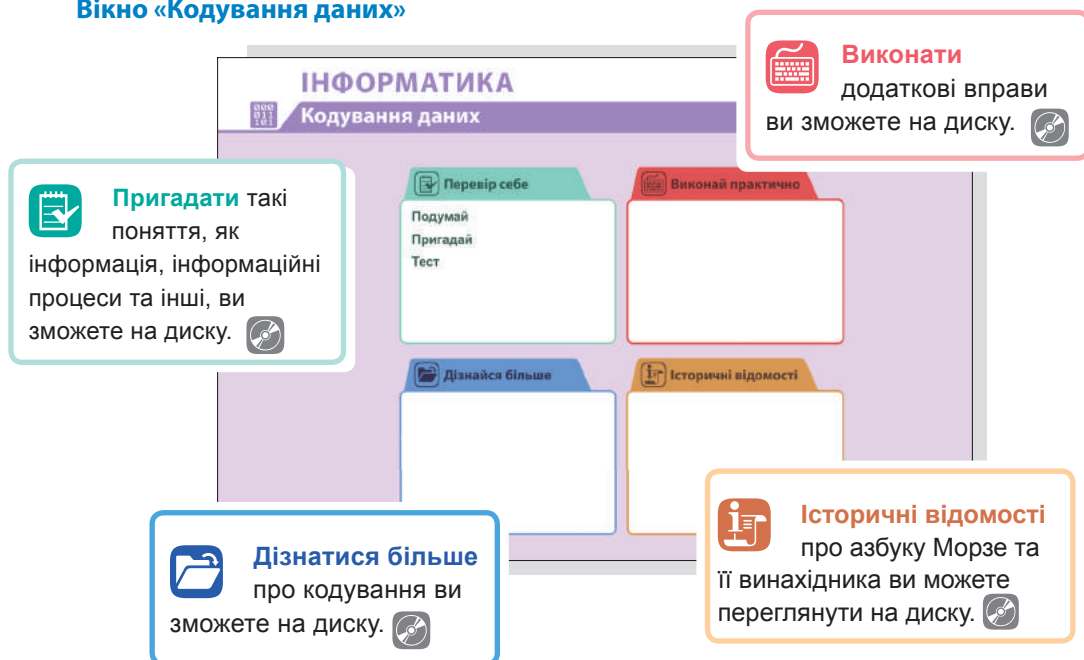
ПУТІВНИК ІНТЕРАКТИВНОЮ ЕЛЕКТРОННОЮ СКЛАДОВОЮ ПІДРУЧНИКА

Зручним супроводженням усього матеріалу підручника інформатики є його інтерактивна електронна складова, посилання на використання якої позначено піктограмою .

Головне вікно програми

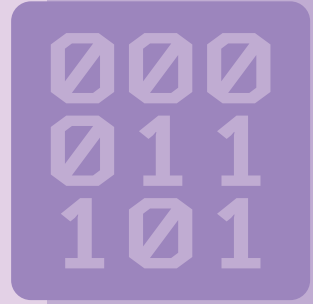


Вікно «Кодування даних»



Розділ 1

Кодування даних



§1

Роль і місце кодування в інформаційних процесах

1.1.

ЗБЕРІГАННЯ, ПЕРЕДАВАННЯ ТА ОБРОБКА ІНФОРМАЦІЇ

У 5 класі ви вже ознайомилися, що таке інформація та інформаційні процеси, а саме: пошук, зберігання, передавання, обробка, використання та захист інформації.

Для передавання і зберігання інформації використовують різні за способом сприйняття знакові системи:

- зорові (літери, музичні ноти, дорожні знаки);
- слухові (усне мовлення, звукові сигнали тощо);
- дотикові (шрифт Брайля для сліпих);
- нюхальні;
- смакові.

Знакові системи, що їх використовують люди, побудовані на основі певного алфавіту і правил виконання операцій над знаками. Основними такими системами є *природні мови*, наприклад, українська, англійська та ін. Кожна природна мова має свій алфавіт. Із символів алфавіту, записаних у певній послідовності та з дотриманням правил *граматики*, утворюють основні об'єкти мови — *слова*. Зі слів відповідно до правил *синтаксису* будують речення.

Окрім природних мов, люди використовують *формальні мови* (системи числення, мову алгебри, мови програмування). Основна відмінність формальних мов від природних полягає в тому, що вони мають строгі правила граматики і синтаксису. У формальних мовах як знаки застосовують не лише літери і цифри, а й інші символи, наприклад, хімічні формули, ноти, дорожні знаки, точки і тире (азбука Морзе).



Пригадати такі поняття, як інформація, інформаційні процеси та інші, ви зможете на диску.



Для подання інформації люди широко застосовують і графічну форму — рисунки, схеми, креслення, карти, графіки та діаграми. Звичайно, форми подання інформації часто поєднують — текст з графікою, знакові сигнали зі звуками тощо.

Сукупність знаків, що відображають (містять) певну інформацію, називають **повідомленням**.

Повідомлення передають на відстань з допомогою *матеріальних носіїв* (папір, флеш-пам'ять, магнітний диск) або *сигналів* (звукових, електричних, оптичних, радіосигналів).

Важливим інформаційним процесом є обробка інформації, тобто перетворення її з одного виду на інший (чи той самий), що виконується за строгими формальними правилами. Обробка може бути пов'язана з отриманням нової інформації. Проте нас цікавитиме обробка, внаслідок якої змінюється форма, а не зміст інформації. До цього виду обробки належить структурування та кодування інформації. Ці процеси необхідні для забезпечення інших інформаційних процесів, оскільки інформація часто існує у формі, не прийнятній для її передавання та зберігання. Тому для перетворення інформації в прийнятну форму застосовують *кодування*, про яке йтиметься далі.

Зазначимо, що варто розрізняти обробку й опрацювання інформації. Опрацювання — це докладне ознайомлення з чимось, створення чогось на підставі ретельного вивчення, добирання та систематизації матеріалів. Таким чином, опрацьовувати інформацію може лише людина, а в контексті комп'ютерних і телекомунікаційних технологій йдеться лише про обробку інформації.

1.2.

КОДУВАННЯ ІНФОРМАЦІЇ

Розглянемо докладніше, що таке кодування і як його застосовують у комп'ютерних технологіях.

Кодування інформації — це процес перетворення інформації з форми, зручної для безпосереднього використання, у форму, зручну для передавання, зберігання або автоматичної обробки.



Прикладом кодування є запис усних повідомлень у текстовій формі, музики — у вигляді нот. Текстове кодування надає можливість нескінченну кількість текстових повідомлень подати з допомогою алфавіту, що складається зі скінченної кількості знаків (наприклад, український алфавіт має 33 літери). Аналогічно, все різноманіття мелодій записують, використовуючи лише сім нот.

У комп'ютері застосовується двійкове кодування, коли інформація будь-якого виду (текстова, числова, графічна, звукова, відео) записується у вигляді коду, алфавіт якого складається лише з двох символів, що умовно позначаються, як «0» та «1». Це пов'язано з тим, що апаратні складові сучасних комп'ютерів здатні обробляти лише сигнали в двох станах. Фізично змоделювати ці стани можна по-різному: намагнічено/ненамагнічено (жорсткий диск), розімкнуто/замкнуто, сигнал з амплітудами 1/0 (рис. 1.1).

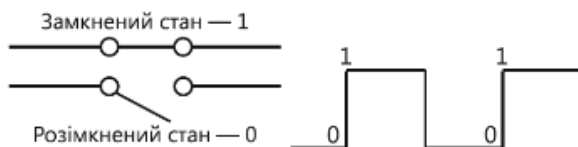




Рис. 1.1. Фізична реалізація станів «0» та «1»

Для того, щоб із закодованого повідомлення видобути інформацію, потрібно виконати операцію *декодування*.

Не слід плутати кодування інформації з її *шифруванням* — перетворенням, що виконують з метою ускладнити або унеможливити розуміння чи зміну інформації в разі її перехоплення.

Основна відмінність між кодуванням й шифруванням полягає в тому, що для відновлення закодованого повідомлення достатньо знати правило заміни, а для розшифрування зашифрованого повідомлення на додаток до правила потрібно мати ще й ключ до шифру.

 **Дізнатися більше**
про кодування ви
зможете на диску. 

Розглянемо це на прикладі. Одним з найвідоміших методів шифрування є шифр Цезаря, яким користувався римський імператор Гай Юлій Цезар у листуванні з генералами для захисту військових повідомлень. Принцип методу полягає в тому, щоб циклічно зсунути алфавіт, а ключ — це кількість літер, на які робиться зсув. Цезар для свого зв'язку використовував цифру 3. Він заміняв першу літеру алфавіту на четверту (А на D), другу — на п'яту (В на Е) і т. д.

Навіть якщо ви знатимете, що шифрування виконано заміною літер, ви не зможете прочитати повідомлення, не знаючи ключа (в нашому випадку — це цифра 3).

Український алфавіт																																	
А	Б	В	Г	Д	Е	Є	Ж	З	И	І	Ї	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ь	Ю	Я	'	КИЇВ
Г	Д	Е	Є	Ж	З	И	І	Ї	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ь	Ю	Я	'	А	Б	В	НІЛД
Український алфавіт за кодом Цезаря зі зсувом на 3 позиції																																	

1.3.

ДВІЙКОВЕ КОДУВАННЯ

Сучасні комп'ютери — це пристрої, здатні обробляти сигнали, які мають два стани (їх позначено цифрами 0 та 1). Отже, щоб комп'ютер міг зберегти та обробити числові значення, текст, зображення, звук чи відео, їх потрібно піддати двійковому кодуванню. Спосіб кодування інформації в комп'ютері насамперед залежить від виду інформації, а саме від того, що потрібно кодувати: числа, текст, графічні зображення чи звук.

Двійкові коди цілих чисел будують, записуючи їх у двійковій системі числення.

Система числення — це сукупність правил запису чисел за допомогою певного набору символів.

У звичній системі числення для запису чисел ми використовуємо десять різних знаків (цифри 0, 1, 2, ..., 9). Тому її називають *десятьковою системою числення*. Натомість у комп'ютері інформація зберігається та обробляється в двійковій системі, що містить лише дві цифри — 0 та 1. Послідовність двійкових цифр називають *двійковим кодом*. Кожна цифра (розряд) у двійковому коді називається бітом. На практиці частіше використовують більшу одиницю — байт, який дорівнює 8 біт.

1.4.

ПЕРЕВЕДЕННЯ ЧИСЛА В ДВІЙКОВУ СИСТЕМУ ЧИСЛЕННЯ

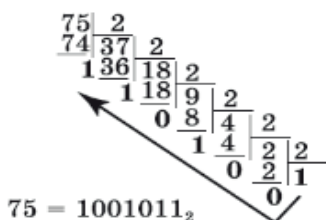


Рис. 1.2. Переведення числа 75 з десятикової системи в двійкову

Як записати десятикове число в двійковій формі, щоб його зрозумів комп'ютер?

Для переведення числа з десятикової системи числення в двійкову його потрібно послідовно ділити на 2 до тих пір, поки не залишиться 1. Відповідне число в двійковій системі числення утворюється як послідовність остач від ділення, записаних у зворотному порядку, починаючи з останньої.

Приклад переведення числа 75 з десятикової системи числення в двійкову наведено на рис. 1.2.

1.5.

ЦІЛІ ЧИСЛА В КОМП'ЮТЕРІ

1	0	0	1	1
1	1	0	0	0
0	0	1	1	1
1	1	1	1	0

Пам'ять комп'ютера можна умовно зобразити, як аркуш паперу в клітинку. В кожній такій клітинці (комірці пам'яті) міститься 0 чи 1.

Такий вигляд після двійкового кодування має в пам'яті інформація будь-якого виду: текстова, числова, графічна. Як її декодувати? Адже ми не знаємо навіть, де починається і де закінчується

елемент даних (число, літера, піксел тощо). Один із способів вирішення цієї проблеми полягає в тому, щоб заздалегідь домовитися, за яким правилом дані записуються в клітинки.

Для більшої наочності перейдемо на десяткову систему. Домовмося, що для запису числа відводиться чотири клітинки. Знаючи це, легко визначити, що в наведеному нижче повідомленні записано три числа: 5671, 1370023 (рис. 1.3).



Рис. 1.3. Повідомлення, що містить три числа

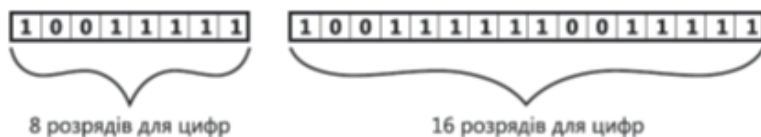
А якщо нам потрібно буде записати в повідомленні п'ятицифрове число (наприклад, 10500), то це зробити не вдасться, оскільки найбільше число, яке можна записати в такому повідомленні, дорівнює 9999.

Для зберігання чисел у пам'яті комп'ютера також відводиться певна кількість розрядів, і записуються числа в двійковій системі числення. Але водночас, як показано на попередньому прикладі, завжди існує обмеження на максимальне число.


З курсу математики ви знаєте, що є числа натуральні, цілі, раціональні. У комп'ютері для чисел різного типу застосовуються різні подання. Для цілих чисел їх два: *беззнакове* і *знакове*.

Беззнакове подання цілого числа

У беззнаковому поданні (рис. 1.4) всі розряди відведено для цифр числа. Воно використовується, коли відомо, що значення, які записуватимуться в комірки пам'яті, завжди додатні (наприклад, порядкові номери, дати).



Дізнатися більше

про переведення з однієї системи числення в іншу ви зможете на диску. 



Історичні відомості —


з історії чисел — ви зможете дізнатися на диску. 

Рис. 1.4. Беззнакове подання цілого числа

Якщо для числа відведено k розрядів, то в найбільшого числа всі вони дорівнюватимуть одиниці, тобто це буде число, що складається з k одиниць. Його десяткове значення дорівнюватиме $2^k - 1$. Найменше число матиме нулі в усіх розрядах і завжди дорівнюватиме нулю. Максимальні десяткові числа для беззнакового подання при різних значеннях k наведено в табл. 1.1.

Таблиця 1.1. Максимальні десяткові числа для різного беззнакового подання

Кількість розрядів	Максимальне число
8	255 ($2^8 - 1$)
16	65535 ($2^{16} - 1$)
32	4294967295 ($2^{32} - 1$)
64	18446744073709551615 ($2^{64} - 1$)

Знакове подання цілого числа

У знаковому поданні старший (лівий) розряд відведено для знака числа (0 — додатне, 1 — від'ємне), а інші — для цифр (рис. 1.5). Воно використовується для запису числових величин, які можуть набувати додатних і від'ємних значень (температура, результати арифметичних операцій тощо).

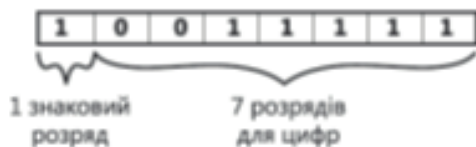


Рис. 1.5. Знакове подання цілого числа

Значення границь діапазонів для знакового подання при різних значеннях k наведено в табл. 1.2.

Таблиця 1.2. Максимальні і мінімальні десяткові числа для різного знакового подання

Кількість розрядів	Мінімальне число	Максимальне число
8	-128	127
16	-32768	32767
32	-2147483648	2147483647
64	-9223372036854775808	9223372036854775807

Ви бачите, що в комп'ютері як у знаковому, так і в беззнаковому поданні множина цілих чисел обмежена і залежить від кількості розрядів, що відводяться для подання числа. Натомість у математиці множина цілих чисел є нескінченною.

Зверніть також увагу на такі дві особливості знакового подання:

- абсолютна величина максимального числа на 1 менша за абсолютну величину мінімального;
- найбільше додатне число, яке можна записати в знаковому поданні, майже в два рази менше за найбільше число в беззнаковому поданні.

ВПРАВА 1



Завдання. Ознайомитись із особливостями подання чисел.

1. Відкрийте в програмі MS Excel файл **Перетворення.xls**, який містить формули для переведення чисел з десяткової системи в двійкову. (Файл знаходиться в папці **Files** на диску.)
2. Уведіть у відповідну клітинку десяткове число 255. Яке двійкове число йому відповідає? Запишіть це число в робочий зошит. Потім уведіть десяткове число 256 і також запишіть його в двійковій формі. Дайте відповідь на запитання:
 - а) Чому подання числа 256 має такий вигляд?



Висновки

Основними знаковими системами, що їх використовують люди, є *природні та формальні мови*. Основна відмінність формальних мов від природних полягає в тому, що вони мають строгі правила граматики і синтаксису.

Обробка інформації — це процес перетворення інформації з одного виду в інший (чи той самий), що виконується за строгими формальними правилами.

Кодування інформації — це перетворення інформації з форми, зручної для безпосереднього використання, у форму, зручну для передавання, зберігання або автоматичної обробки.

Для того, щоб із закодованого повідомлення видобути інформацію, потрібно виконати операцію *декодування*.

Система числення — це сукупність правил запису чисел за допомогою певного набору символів.

Послідовність двійкових цифр називають *двійковим кодом*. Кожна цифра в двійковому коді називається *бітом*.



Контрольні запитання та завдання

1. Як люди зберігають і передають інформацію?
2. Якою може бути обробка інформації?
3. Що таке кодування і декодування?
4. Що називають двійковим кодом?
5. Що таке біт?
- 6*. Назвіть способи подання цілих чисел і поясніть їх суть.



Питання для роздумів

- 1*. Для чого використовують кодування?
- 2**. Чи може число записане в десятковій системі числення дорівнювати числу записаному в двійковій системі числення?



Завдання для досліджень

- 1*. Визначте, як перевести числа з двійкової системи числення в десяткову.
- 2*. Підготуйте повідомлення про історію походження чисел.
- 3*. Поясніть, що таке штрих-код, яке його походження. Наведіть приклади використання штрих-кодів.
- 4*. Де використовують QR-коди?

§2

Кодування символів

2.1.

АЗБУКА МОРЗЕ

Одним із перших цифрових способів передачі інформації став код Морзе. Код Морзе — це нерівномірний телеграфний код, де спосіб кодування літер алфавіту, цифр та інших символів подано певною комбінацією «крапок» і «тире». За одиницю часу приймається тривалість однієї крапки, тривалість тире дорівнює трьом крапкам, пауза між елементами одного знака — одна крапка, між знаками в слові — 3 крапки, між словами — 7 крапок. Азбука, з якої формується зазначений код, була названа на честь американського винахідника Семюела Морзе, який придумав її в 1838 році. Телеграф і радіотелеграф спочатку використовували азбуку Морзе, пізніше почали застосовувати код Бодо та ASCII, які є більш зручними для автоматизації.

**Історичні відомості**
про азбуку Морзе та

її винахідника ви можете
переглянути на диску.

2.2.

КОДУВАННЯ СИМВОЛІВ У КОМП'ЮТЕРНІЙ ТЕХНІЦІ

Символи тексту людина розрізняє за накресленням. Наприклад, кожен з нас швидко визначить у тексті літеру «а», яким би шрифтом її не було набрано. Якщо слідувати аналогії зі сприйняттям тексту людиною, то в комп'ютері потрібно зберігати зображення літер, потім з них складати слова і зберігати зображення рядків.

Зрозуміло, що цей спосіб неефективний, оскільки доведеться зберігати безліч зображень, хоча кількість літер, з яких їх складено, обмежена (це літери алфавіту тої мови, якою подано текст).

Тому для комп'ютерного подання текстової інформації застосовується інший спосіб: символи алфавіту кодуються двійковими числами, а текст подається у вигляді набору чисел — кодів символів, що його складають.

Щоб такий набір чисел можна було декодувати, слід знати, де закінчується і де починається код кожного символу, а для цього, як було показано на прикладі чисел, необхідно перед кодуванням визначити потрібну для кодування кількість розрядів. Тож яку довжину коду символу взяти, щоб закодувати всі символи, які можуть трапитися в тексті?

Якщо це двійковий код з довжиною 2 розряди, то з його допомогою можна буде закодувати алфавіт, що містить 4 символи, оскільки за цієї довжини коду існує чотири різні його комбінації. Двійковий код з довжиною 3 розряди дозволяє закодувати алфавіт, що містить 8 символів (рис. 1.6).

Код з довжиною 2 розряди		
Код	Літера	Десяткове значення коду
00	А	0
01	Б	1
10	В	2
11	Г	3

Код з довжиною 3 розряди		
Код	Літера	Десяткове значення коду
000	А	0
001	Б	1
010	В	2
011	Г	3
100	Ґ	4
101	Д	5
110	Е	6
111	Є	7

Рис. 1.6. Кодування символів двійковими кодами різної довжини

Прослідковується така закономірність:

2 розряди — $2 \cdot 2 = 4 = 2^2$ символів;

3 розряди — $2 \cdot 2 \cdot 2 = 8 = 2^3$ символів;

4 розряди — $2 \cdot 2 \cdot 2 \cdot 2 = 32 = 2^4$ символів;

N розрядів — $2 \cdot 2 \cdot 2 \cdot \dots \cdot 2 \cdot 2 = 2^N$ символів.

З допомогою двійкового коду завдовжки N розрядів можна закодувати алфавіт, що містить 2^N символів.

Таким чином, щоб закодувати текст, слід обрати довжину коду, достатню для кодування потрібної кількості символів, і скласти таблицю, де буде вказано, який код якому символу відповідає. Такі таблиці вже складено, і називають їх наборами символів.

Таблиця 1.3. Символи з розширеного набору символів ASCII та їхні коди

Символ	Десяткове значення коду	Код	Символ	Десяткове значення коду	Код
пробіл	32	00100000	0	48	00110000
!	33	00100001	1	49	00110001
#	35	00100011	2	50	00110010
\$	36	00100100	3	51	00110011
*	42	00101010	4	52	00110100
+	43	00101011	5	53	00110101
,	44	00101100	6	54	00110110
-	45	00101101	7	55	00110111
.	46	00101110	8	56	00111000
/	47	00101111	9	57	00111001
A	65	01000001	N	78	01001110
B	66	01000010	O	79	01001111
C	67	01000011	P	80	01010000
D	68	01000100	Q	81	01010001
E	69	01000101	R	82	01010010
F	70	01000110	S	83	01010011
G	71	01000111	T	84	01010100
H	72	01001000	U	85	01010101
I	73	01001001	V	86	01010110
J	74	01001010	W	87	01010111
K	75	01001011	X	88	01011000
L	76	01001100	Y	89	01011001
M	77	01001101	Z	90	01011010

Сьогодні найпоширенішими є набори символів ASCII, Unicode («юнікод») і сумісні з ними. Фрагмент розширеного набору символів ASCII

наведено в табл. 1.3. У цьому наборі, як ви можете пересвідчитися, для кодування застосовують 8 розрядів. Розмір алфавіту, який можна закодувати з допомогою такого набору, вміщує 256 символів (пронумеровані десятковими цифрами від 0 до 255). Цього достатньо, щоб закодувати цифри, знаки пунктуації, латинські літери (великі й малі) та літери кирилиці (великі й малі).

Недолік такої таблиці кодування — неможливість закодувати текст, що містить фрагменти багатьма мовами, наприклад, англійською (латиниця), українською (кирилиця) і німецькою (латиниця та умляути).

Щоб усунути це обмеження, в 1991 році було запропоновано стандарт кодування Unicode, який дозволяє використовувати в текстах будь-які символи будь-яких мов світу. Загалом в Unicode для кодування символів відведено 31 розряд двійкового коду.

Використовуючи Unicode, закодовано всі алфавіти відомих мов, зокрема і «мертвих» (єгипетські ієрогліфи, писемність майя, етрусський алфавіт). Для мов, що мають кілька алфавітів або варіантів написання (наприклад, японська, індійська), закодовано всі варіанти. В Unicode внесено всі математичні, хімічні, музичні та інші наукові символні позначення. Потенційна ємність Unicode така велика, що наразі використовується лише незначний відсоток доступних кодів символів.

Перші 128 символів у стандарті Unicode збігаються з таблицею ASCII. Далі розміщено основні алфавіти сучасних мов. Вони містяться в першій частині таблиці, значення їхніх кодів не перевищують $2^{16} = 65536$. Тому в сучасних комп'ютерах застосовується скорочена 16-розрядна версія Unicode, яка називається *базовою багатомовною площиною* (Base Multilingual Plane, BMP).

2.3.

ОСОБЛИВОСТІ КОДУВАННЯ КИРИЛИЦІ

У наборі ASCII перші 32 коди (з 0 по 31) відведені для операцій (перенесення рядка, скасування попередньої операції, подавання звукового сигналу). Ці коди мають сприйматися пристроєм виведення як команди.

Наступні коди, з 32 по 127, є інтернаціональними і відповідають символам латинського алфавіту, цифрам, знакам арифметичних операцій та знакам пунктуації. Коди з 128 по 255 — національні, тобто в кодуваннях для різних мов тому самому коду відповідають різні символи.

Для українських і російських літер є кілька *кодових таблиць* (code pages), що різняться розташуванням символів. Найчастіше вам траплятимуться такі: KOI8-U, KOI8-R, Windows-1251, Code Page 866 (CP 866), ISO 8859. Тексти, створені з використанням однієї таблиці, неправильно

відображаються у разі використання іншої. Саме це є причиною того, що інколи в браузері замість тексту ви бачите беззмстовний набір знаків (рис. 1.7).



Рис. 1.7. Символи відображаються неправильно

ВПРАВА 2



Завдання. Ознайомитися з особливостями кодування тексту.

1. Запустіть програму Word, виберіть команду для вставлення символів у текст на вкладці **Вставлення** ▶ **Символ** ▶ **Інші символи**. Відкриється вікно, показане на рис. 1.8.

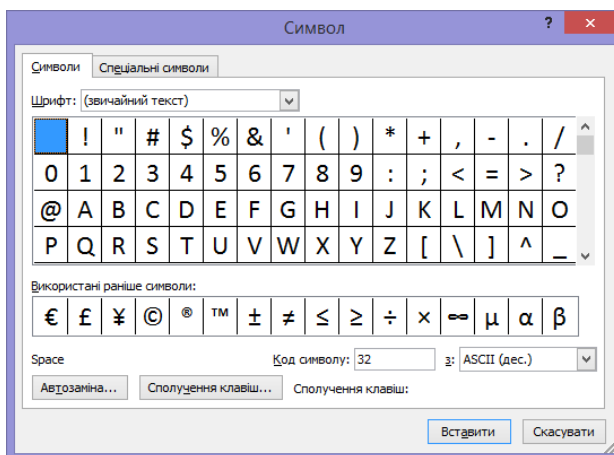


Рис. 1.8. Діалогове вікно для вставлення символів

2. Виберіть у списку **Шрифт** елемент **звичайний текст**, а у списку **з**: елемент **ASCII (дес.)**. Збільште вікно, як показано на рисунку.
3. Зверніть увагу, що у вікні обрано символ пробілу. Його код (32) відображається в нижньому правому куті вікна. Праворуч від поля з кодом розташовано список, у якому обирають таблицю символів і формат відображення значень їхніх кодів (десятковий чи шістнадцятковий).

4. Знайдіть у таблиці та клацніть символ 126 — після нього, окрім, знаків йдуть літери мов, у яких застосовується латиниця (німецька, голландська, фінська). Виберіть у списку таблиць елемент **кирилиця (дес.)**. Набір символів після 126 символу змінився — у таблиці з'явилися літери кирилиці (рис. 1.9).

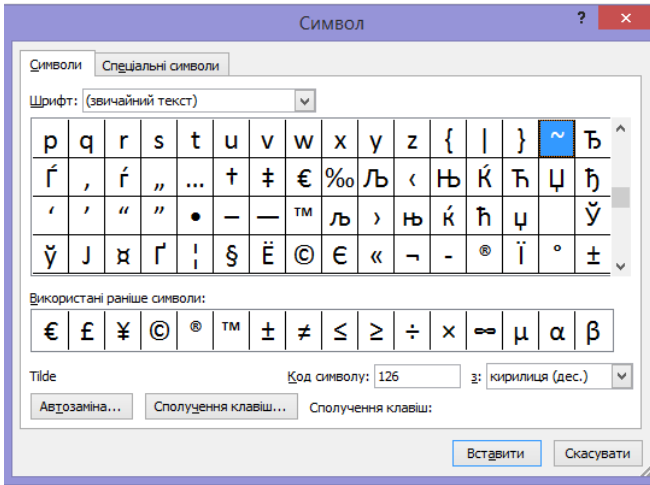


Рис. 1.9. Символи кирилиці в таблиці символів

5. Виберіть у списку **з:** таблицю **Юнікод (шіст.)**. У верхній частині вікна праворуч з'явиться список **Набір**. Розгорніть його, ви побачите, які набори символів і літер закодовано в цій таблиці (рис. 1.10).

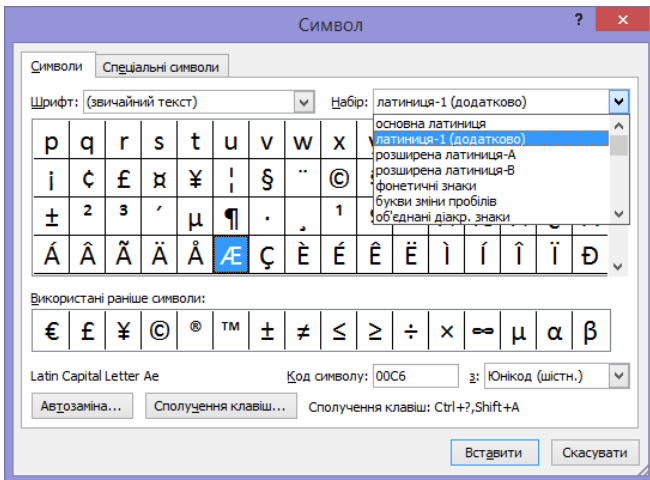


Рис. 1.10. Перелік наборів символів, закодованих у таблиці Юнікод

6. Відкрийте в браузері сторінку uk.wikipedia.org, що містить текст українською мовою. У контекстному меню сторінки виберіть команду **Кодировка** (Кодування) та інше кодування сторінки (наприклад, **Западноевропейская (ISO)**). Як зміниться її текст? Поверніть початкове кодування сторінки, обравши елемент **Юникод (UTF-8)**.

**Висновки**

Для комп'ютерного подання текстової інформації символи алфавіту кодуються двійковими числами, а текст подається у вигляді набору чисел — кодів символів, що його складають.

З допомогою двійкового коду з довжиною N розрядів можна закодувати алфавіт, що містить 2^N символів.

Для кодування тексту створено таблиці, в яких вказано, який код якому символу відповідає, їх називають *наборами символів*.

Найпоширенішими є набори символів ASCII, Unicode та сумісні з ними.

**Контрольні запитання та завдання**

1. Який підхід до кодування запропонував свого часу Семюел Морзе?
2. Який підхід застосовано до кодування текстової інформації в комп'ютерних технологіях?
3. Яку назву має стандартний набір, що складається з 256 символів?
4. Яку назву має стандарт кодування, що дозволяє здійснити кодування всіх алфавітів відомих мов?
5. Чому на веб-сторінці може неправильно відображатися текст?

**Питання для роздумів**

- 1*. Яку мінімальну довжину коду матиме повідомлення, подане за допомогою алфавіту з 65 символів?
- 2**. Чому азбука Морзе є нерівномірним кодом?

**Завдання для досліджень**

- 1*. Підготуйте повідомлення про використання на практиці різних способів кодування текстової інформації.
- 2*. Дізнайтеся, що таке стенографія і яке її призначення.

§3

Вимірювання інформації

3.1.

ЯК ВИМІРЯТИ ІНФОРМАЦІЮ

Якщо спробувати записати на CD-диск кінофільм з високою якістю зображення, зробити це не вдасться, оскільки фільм не поміститься на CD. А якщо, користуючись «повільним» інтернет-підключенням, переглядати онлайн-відео, воно буде «гальмувати». Щоб уникнути таких ситуацій, слід враховувати, який обсяг інформації бере участь в інформаційному процесі.

Інформація, за визначенням, — це щось корисне і зрозуміле користувачеві. Виходить, що текстовий файл, який містить 100 літер «А» або пробілів і займає на диску певний простір, не несе інформації. Отже, до вимірювання інформації мають існувати принаймні два підходи — такий, що враховує зміст повідомлення, і такий, що дає змогу оцінити обсяг його даних. Окрім цього, потрібно ввести еталонну одиницю, подібно до того, як для вимірювання маси введено еталонну одиницю кілограм, для вимірювання довжини — метр.

3.2.

ЗМІСТОВИЙ ПІДХІД

Основою змістового підходу до вимірювання інформації є зміст інформаційного повідомлення. Згідно з цим підходом, кількість інформації розглядається як ступінь зменшення невизначеності знань у результаті отримання повідомлення.

Розглянемо процес підкидання монети. Вона може впасти на один з двох боків («орел» чи «решка»), отже, перед її підкиданням невизначеність дорівнює 2. Невизначеність перед підкиданням грального кубика дорівнює 6 (у кубика шість граней), невизначеність того, скільки балів можна отримати за відповідь на уроці, становить 12 (за дванадцятибальної системи оцінювання), а невизначеність того, який колір вибере користувач із шістнадцятиколірної палітри програми Paint, — 16 (рис. 1.11).

Невизначеність знання про результат певної події дорівнює кількості можливих результатів цієї події.

Якщо ми не знаємо значення жодного розряду коду кольору палітри, невизначеність кольору дорівнює 16. Коли нам стане відомий старший (четвертий) розряд, невизначеність зменшиться в два рази, оскільки кіль-

кість варіантів буде дорівнювати 8, адже ми точно знатимемо, з якої частини таблиці обрано колір — лівої чи правої. Якщо на додаток до цього знати третій розряд, стане відомо, в якій половині стовпця міститиметься обраний колір — верхній чи нижній.

Код	Повідомлення
0	Орел
1	Решка

Код	Колір	Код	Колір
0000	1	1000	9
0001	2	1001	10
0010	3	1010	11
0011	4	1011	12
0100	5	1100	13
0101	6	1101	14
0110	7	1110	15
0111	8	1111	16

Рис. 1.11

Таким чином, щоб дізнатися, який колір обрав користувач, достатньо задати чотири запитання (кожне про значення відповідного розряду коду) та отримати у відповідь чотири повідомлення, що містять відповідь «так» чи «ні» (0 або 1). Згідно зі змістовим підходом, кожне таке повідомлення містить один біт інформації.

Один біт інформації — це кількість інформації, що зменшує невизначеність у два рази.

Отже, двійковий код кольору шістнадцятиколірної палітри містить 4 біти інформації, а повідомлення про те, яким боком впала монетка, — 1 біт.

Як ви вже знаєте, біт є мінімальною одиницею вимірювання інформації.

Наступна за величиною одиниця — *байт*, за ним йдуть *кілобайт*, *мегабайт*, *гігабайт* і *терабайт*.

$$1 \text{ байт} = 8 \text{ біт} = 2^3 \text{ біт};$$

$$1 \text{ кілобайт (Кбайт)} = 2^{10} = 1024 \text{ байт};$$

$$1 \text{ мегабайт (Мбайт)} = 2^{10} \text{ Кбайт} = 1024 \text{ Кбайт};$$

$$1 \text{ гігабайт (Гбайт)} = 2^{10} \text{ Мбайт} = 1024 \text{ Мбайт};$$

$$1 \text{ терабайт (Тбайт)} = 2^{10} \text{ Гбайт} = 1024 \text{ Гбайт}.$$

1 біт

0 0 1 1 0 1 1 0

1 байт = 8 біт

Зверніть увагу на те, що в інформатиці система утворення кратних одиниць відрізняється від прийнятої в інших науках. У традиційних метричних системах як множники кратних одиниць застосовуються кратні 10 коефіцієнти 1000, 100 000, 100 000 000 і т. д. та відповідні префікси «кіло», «мега», «гіга» і т. д.

Оскільки в комп'ютері застосовується двійкове кодування, в кратних одиницях вимірювання використовуються коефіцієнти, кратні 2. Щоб відрізнити десяткові та двійкові коефіцієнти, двійкові домовилися писати з великої літери.

Приклад 1. Під час відгадування цілого числа, значення якого знаходиться в діапазоні від 1 до N , було отримано 5 біт інформації. Чому дорівнює число N ?

Розв'язання

Визначимо невизначеність повідомлення з довжиною 5 біт: $2^5 = 32$. За допомогою такого повідомлення можна закодувати будь-яке з 32 різних чисел. Якщо це числа з діапазону від 1 до 32, то $N = 32$.

Відповідь: $N = 32$.

Приклад 2. Світлове табло складається з лампочок, кожна з яких може перебувати в одному з двох станів: «увімкнена» або «вимкнена». Яку найменшу кількість лампочок має містити табло, щоб з його допомогою можна було передати 140 різних сигналів?

Розв'язання

Визначимо, скільки двійкових розрядів потрібно для кодування 140 різних сигналів:

$$128 < 140 < 256 \Rightarrow 2^7 < 140 < 2^8.$$

Отже, щоб табло могло відтворити 140 різних сигналів, воно має містити 8 лампочок.

Відповідь: 8 лампочок.

3.3. АЛФАВІТНИЙ ПІДХІД

З допомогою алфавітного підходу можна виміряти кількість інформації в тексті, поданому будь-якою мовою. Водночас не враховуються такі її властивості, як корисність, актуальність або повнота. Кількість інформації не пов'язана зі змістом тексту, а залежить від *інформаційної ваги символу в алфавіті*.

Алфавіт — це весь набір букв, символів пунктуації та інших символів і знаків (включаючи пробіли), що застосовуються в тексті. Загальна кількість символів алфавіту визначає його *потужність*.

Інформаційна вага символу та потужність алфавіту пов'язані між собою такою формулою: $N = 2^b$, де N — потужність алфавіту, b — інформаційна вага символу.

Найменшу потужність (рівну 1) має алфавіт з двома символами ($2 = 2^1$). Саме такий алфавіт — двійковий — застосовується в комп'ютері.

Інформаційна вага символу двійкового алфавіту прийнята за одиницю інформації, і називається ця одиниця також *бітом*. Як і в разі змістового підходу, від неї утворюються більші одиниці (байт, кілобайт і т. д.).

Якщо, наприклад, алфавіт містить 256 символів, то інформаційна вага одного його символу становитиме 8 біт ($2^8 = 256$). Згадайте принципи кодування тексту: саме за такої довжини двійкового коду можна закодувати всі символи зазначеного алфавіту.


У цьому випадку складене з символів повідомлення можна буде декодувати однозначно, тобто видобути з нього точну інформацію. Якщо зменшити кількість бітів, то деякі символи закодувати не вдасться, у повідомленні може з'явитися, скажімо, слово «кр*к», і не буде зрозуміло, яке це слово: «крик» чи «крок».

За алфавітного підходу кількість інформації в тексті визначається так, як це робиться в разі обчислення загальної ваги певної кількості однакових предметів.

Кількість інформації в текстовому повідомленні дорівнює кількості символів у ньому, помноженій на інформаційну вагу символу.

Приклад 3. Визначити кількість інформації в текстовому повідомленні, що містить 96 символів. Для запису повідомлення використано алфавіт, що складається з 128 символів.



Перевірити свої знання, пройти тест і розгадати кросворд ви зможете на диску. 

Розв'язання

Визначаємо потужність алфавіту: $128 = 2^7$.

Отже інформаційна вага символу становить 7 біт.

Обчислюємо кількість інформації: $96 \cdot 7 = 672$ біт = 84 байти.

Відповідь: 672 біти, або 84 байти.

Приклад 4. На клавіатурі набрано текст: «Наше життя — це наші думки». Визначити кількість інформації в ньому в бітах і байтах.

Розв'язання

Прийmemo, що інформаційна вага символів, використаних для набору тексту, дорівнює 8, оскільки, швидше за все, йдеться про набір символів ASCII.

Полічимо кількість символів у повідомленні: 27 (з урахуванням пробілів і знаків пунктуації).

Визначаємо кількість інформації у повідомленні: $27 \cdot 8 = 216$ біт = 27 байт.

Відповідь: 216 біт, або 27 байт.

**Висновки**

Вимірюючи інформацію, використовують змістовий та алфавітний підходи. Згідно зі змістовим підходом, кількість інформації розглядається як ступінь зменшення невизначеності знань у результаті отримання повідомлення.

Невизначеність знання про результат певної події дорівнює кількості можливих результатів цієї події.

Один біт інформації — це кількість інформації, що зменшує невизначеність у два рази.

В алфавітному підході кількість інформації залежить від *інформаційної ваги символу в алфавіті*.

Алфавіт — це весь набір букв, символів пунктуації та інших символів і знаків, що застосовуються у тексті. Загальна кількість символів визначає його *потужність*.

Інформаційна вага символу та потужність алфавіту пов'язані між собою такою формулою: $N = 2^b$, де N — потужність алфавіту, а b — інформаційна вага символу.

Кількість інформації в текстовому повідомленні дорівнює кількості символів у ньому, помноженій на інформаційну вагу символу.

**Контрольні запитання та завдання**

1. Які підходи використовують для вимірювання інформації?
2. Які одиниці вимірювання інформації ви знаєте?
- 3*. Поясніть принцип змістового підходу вимірювання інформації.
4. Що таке інформаційна вага символу в алфавіті?
5. Що таке алфавіт?
6. Як вимірюється кількість інформації текстового повідомлення?

**Питання для роздумів**

- 1*. У басейні 4 доріжки для плавання. Інструктор повідомив школярів, що вони будуть плавати на другій доріжці. Скільки інформації отримали школярі?
- 2**. Марійка задумала число від 1 до 100. Яку найменшу кількість спроб має зробити Андрій, щоб відгадати число, яке задумала Марійка?
- 3**. Повідомлення про те, що Івасик живе на 12 поверсі, несе 4 біти інформації. Скільки поверхів у будинку?



Завдання для досліджень

- 1*. Дізнайтеся про інші способи вимірювання інформації. Поясніть їх принцип.

ПРАКТИЧНА РОБОТА №1

Розв'язування задач на визначення довжини двійкового коду різних типів

Завдання. Розв'язати задачі на визначення довжини двійкового коду.

З метою вдосконалення практичних умінь і навичок роботи з електронним процесором рекомендуємо розв'язувати пропонувані задачі, використовуючи електронні таблиці. Для цього слід завантажити файл **ПР1.xls** з папки **Files** на диску, що містить шаблони задач (у файлі шаблону до клітинок зафарбованих жовтим кольором введіть дані задачі, до клітинок зафарбованих зеленим кольором — формули для виконання обчислень).

Задача 1. У телевізійному шоу «Таланти України» телеглядачам потрібно обрати одного з чотирьох найкращих учасників. Організаційний комітет шоу отримав інформацію від 151552 телеглядачів. Який обсяг інформації в кілобайтах необхідно опрацювати оргкомітету?

Задача 2. Швидкість передавання даних каналом зв'язку становить 256 Кбіт/с. Файл було передано через канал за 2 хв. Визначте обсяг файлу в мегабайтах.

Задача 3. Текст, що має довжину 100 символів, записаний з використанням алфавіту потужністю 64 символи. Який інформаційний обсяг тексту в байтах?

Задача 4. Інформаційний обсяг набраного на комп'ютері тексту становить 3,5 Кбайт. Скільки символів містить цей текст? Врахуйте, що інформаційна вага символу становить 8 біт.

Задача 5. Середня швидкість читання учня 8-го класу становить 500 символів за хвилину. Визначте, скільки байтів інформації він встигне опрацювати за 30 хв безперервного читання. Врахуйте, що інформаційна вага символу становить 8 біт.

Щоб краще зрозуміти спосіб виконання завдання, в MS Excel книга **ПР1.xls** містить також розв'язання прикладів 3 і 4, представлених у даному параграфі вище (рис. 1.12).

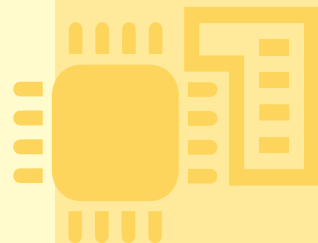
	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	Довжина тексту	96	символів									
3	Потужність алфавіту	128	символів									
4	Інформаційна вага символу	7	біт	До якого								
5	Обсяг тексту	672	біт	ступеня								
6	Обсяг тексту	84	байт	необхідно								
7				піднести								
				число 2, щоб								

Приклад 3. Визначити кількість інформації в текстовому повідомленні, що містить 96 символів (для запису повідомлення застосовано алфавіт, що складається з 128 символів).

Рис. 1.12. Зразок виконання Прикладу 2 в середовищі MS Excel

Після виконання завдання збережіть файл під назвою **ПР1_Прізвище.xls**.

Розділ 2



Апаратно-програмне забезпечення комп'ютера



§4

Апаратне забезпечення

4.1.

БУДОВА КОМП'ЮТЕРА

Комп'ютер — це програмований електронний пристрій, який приймає дані, обробляє їх, відображає результати у вигляді інформаційних повідомлень і за потреби зберігає дані для подальшого їх використання.

У цьому визначенні згадуються чотири інформаційних процеси — введення, обробка, виведення і зберігання. Усі сучасні комп'ютери містять компоненти, що забезпечують їх виконання. Це — центральний процесор, внутрішня і зовнішня пам'ять та пристрої введення і виведення. Зв'язок між ними підтримується за допомогою спеціальної інформаційної магістралі — шини.

Розглянемо ці компоненти і процеси докладніше.

Процеси обробки. Обробляє дані та виконує програми головний компонент комп'ютера — *центральний процесор*. Він також керує всіма блоками комп'ютера.

Процеси зберігання. Комп'ютер зберігає програми і дані в пам'яті. Традиційно її поділяють на внутрішню і зовнішню. *Внутрішня пам'ять* — це швидкодіюча електронна пам'ять, розташована на системній (материнській) платі. *Зовнішня пам'ять* виготовляється у вигляді різноманітних *пристроїв зберігання*.

Процеси введення і виведення. Щоб користувачі отримували інформацію про результати роботи комп'ютера, його оснащують *пристроями виведення* (монітором, звуковими колонками, веб-камерою), а для того, щоб вони могли вводити до комп'ютера нову інформацію та керувати ним, застосовуються *пристрої введення* (клавіатура, миша, тачпад).



Щоб комп'ютери мали можливість обмінюватися даними через мережу, їх оснащують *комунікаційними пристроями* (модемом, мережевою платою). Залежно від того, що робить комп'ютер — передає дані в мережу чи приймає їх, ці пристрої виконують процеси введення або виведення.

Класифікацію пристроїв комп'ютера за їх призначенням наведено на рис. 2.1.

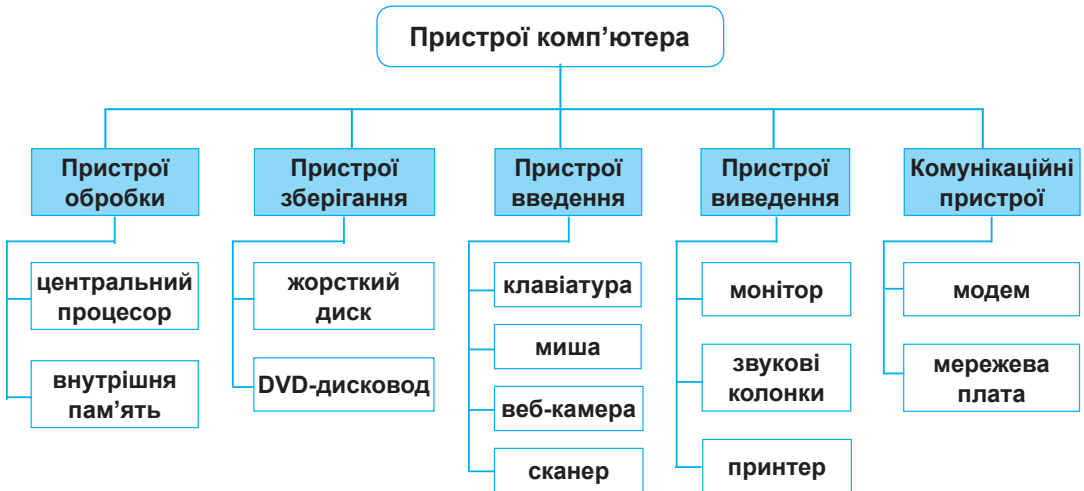


Рис. 2.1. Класифікація пристроїв комп'ютера

4.2.

ЦЕНТРАЛЬНИЙ ПРОЦЕСОР



Рис. 2.2. Мікросхеми центрального процесора

Центральний процесор (ЦП) — основний компонент комп'ютера, призначений для керування всіма його пристроями та виконання арифметичних і логічних операцій над даними. Це компактний (розміром близько 5×5×0,3 см) модуль, що вставляється в гніздо на материнській платі.

Сьогодні серед виробників процесорів лідирують дві компанії — Intel Corporation та Advanced Micro Devices (AMD) (рис. 2.2).

Можливості комп'ютера великою мірою залежать від характеристик процесора, насамперед від його *продуктивності*, яку зазвичай оцінюють за тактовою частотою і за кількістю ядер — головних обчислювальних модулів.

Тактова частота визначає, скільки елементарних операцій (тактів) процесор виконує за секунду, і вимірюється в мегагерцах (1 МГц = 10⁶ Гц) і гігагерцах (1 ГГц = 10⁹ Гц).



Дізнатися більше
про характеристики
центрального процесора
ви зможете на диску.





Багатоядерний процесор може мати два чи більше обчислювальних ядра на одному процесорному кристалі або в одному корпусі. Сучасні багатоядерні процесори мають 2, 4 і навіть 8 ядер з тактовою частотою кожного від 1,66 ГГц до 4,7 ГГц.

4.3.

МАТЕРИНСЬКА ПЛАТА

Материнська плата — пристрій, на якому встановлено внутрішні компоненти комп'ютера. Використовується для передавання даних між усіма його пристроями.

Це складна багатошарова друкована плата, до якої підключено практично всі пристрої комп'ютера. Друкована плата є пластиною, що вкрита сіткою провідників-доріжок, по яких електричні сигнали надходять до змонтованих на платі мікросхем і рознімів, куди вставляють інші пристрої комп'ютера (рис. 2.3).



Дізнатися більше

про материнську плату ви можете на диску.

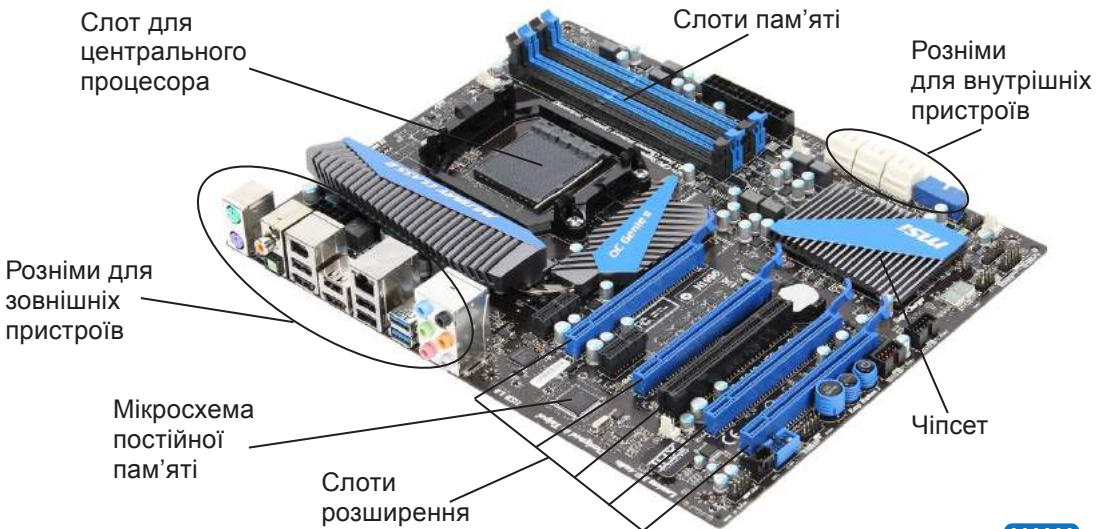


Рис. 2.3. Материнська плата



4.4.

ВНУТРІШНЯ ПАМ'ЯТЬ

Як зазначалося, пам'ять комп'ютера поділяють на зовнішню і внутрішню. Остання, в свою чергу, складається з оперативної та постійної пам'яті. Основна характеристика внутрішньої пам'яті — обсяг, який вимірюють у мегабайтах (Мбайт) і гігабайтах (Гбайт).

Оперативна пам'ять, або ОЗП (оперативний запам'ятовуючий пристрій), є основною частиною внутрішньої пам'яті (рис. 2.4), де зберігають-



ся дані та програми для виконуваних у поточний момент завдань. У разі вимкнення електроживлення вміст оперативної пам'яті втрачається.

Оперативна пам'ять — швидкодіюча пам'ять, призначена для записування, зберігання і зчитування інформації у процесі її обробки.



Рис. 2.4. Модулі оперативної пам'яті



Рис. 2.5. Мікросхема постійної пам'яті

Якби комп'ютер мав лише оперативну пам'ять, вміст якої після вимкнення живлення втрачається, то щоразу, ввімкнувши комп'ютер, доводилося б вводити програми і дані заново. Те саме можна сказати і про налаштування телевізора і мобільного телефону.

Цієї проблеми вдається уникнути завдяки наявності в комп'ютері *постійної пам'яті, або ПЗП (постійного запам'ятовуючого пристрою)*. У неї записують інформацію, що не змінюється під час роботи комп'ютера. Постійна пам'ять енергонезалежна, тобто здатна зберігати інформацію і після вимкнення живлення. Виготовляють її у вигляді мікросхем (рис. 2.5).

Постійна пам'ять — швидкодіюча енергонезалежна пам'ять, призначена для зберігання інформації, що не змінюється під час виконання програм. Ця пам'ять забезпечує лише можливість зчитування інформації.

Є такі мікросхеми постійної пам'яті, які користувач може багаторазово програмувати, а інформація з них стирається в електричний спосіб. До пам'яті такого типу належить, наприклад, *флеш-пам'ять*.

4.5.

ЗОВНІШНЯ ПАМ'ЯТЬ

Зовнішня пам'ять — це пам'ять, призначена для тривалого зберігання даних, цілісність яких не залежить від того, ввімкнено чи вимкнено комп'ютер. Пам'ять цього типу виготовляють у вигляді різноманітних пристроїв для *зберігання цифрових даних*.

Пристрій зберігання даних складається з носія, на якому записано дані, та допоміжного обладнання, що забезпечує можливість їх записувати, зчитувати і передавати.

Носій даних може бути знімним або стаціонарним. Після вимкнення живлення інформація, що міститься на пристрої зберігання даних, не втрачається.



Нині найпоширенішими пристроями зберігання та носіями даних є дискові накопичувачі, флеш-пам'ять та оптичні диски. Розглянемо їх докладніше.

Дискові накопичувачі (жорсткі диски) — основні пристрої зберігання даних в інформаційних системах (рис. 2.6). Зазвичай вони розміщені в системному блоці (корпусі) комп'ютера, проте можуть бути виконані і як зовнішні пристрої. У такому разі диск підключають до комп'ютера через порт USB.

Основні характеристики дискового накопичувача: *ємність, тип інтерфейсу* (спосіб підключення диска), *швидкість передавання даних та середній час доступу до даних*.

Для довгострокового зберігання інформації сьогоденні використовують оптичні носії — диски DVD, CD та BD (Blu-ray Disk). Записування даних на ці носії та зчитування з них здійснюють, застосовуючи лазер. Для цього використовують відповідні дисководи та плеєри.


Вище вже згадувалося про флеш-пам'ять. Цю технологію використовують для виготовлення зовнішніх пристроїв зберігання (флешок), які підключають до комп'ютера через порт USB. Ємність флешок варіюється в діапазоні від 2 до 256 Гбайт, і це ще не межа.



Рис. 2.6. Дискові накопичувачі




Дізнатися більше

про магнітні та твердотілі накопичувачі ви зможете на диску. 



Дізнатися більше

про оптичні носії та оптичні дисководи ви зможете на диску. 

4.6.

ВВЕДЕННЯ І ВИВЕДЕННЯ ІНФОРМАЦІЇ

З допомогою пристроїв введення користувачі вводять до комп'ютера нову інформацію та подають йому команди. Найпоширеніші з цих пристроїв — клавіатура, миша, сканер, веб-камера і мікрофон.


Щоб користувач міг дізнатися про результати роботи комп'ютера, їх потрібно подати в прийнятній для людини формі, тобто у вигляді візуальної чи звукової інформації. Для цього призначені пристрої виведення, серед яких найбільшого поширення набули монітор, звукові колонки та принтер.

Клавіатура та миша

Клавіатура має символні клавіші, призначені для введення букв, цифр і знаків пунктуації, а також спеціальні — керуючі і функціональні. Клавіші на клавіатурі згруповано в блоки. Алфавітно-цифровий блок клавіш відповідає клавіатурі друкарської машинки. Цифровий блок нагадує клавіатуру калькулятора. Його клавіші працюють у двох режимах — введення цифр і математичних знаків та керування курсором (режим обирається клавішею




Дізнатися більше

про призначення різних клавіш клавіатури ви зможете на диску. 



Історичні відомості

Історія виникнення миші пов'язана з ім'ям відомого американського вченого Дугласа Енгельбарта та науково-дослідною лабораторією ХероxPARC. 



Дізнатися більше


про принцип дії та характеристики сканерів ви зможете на диску. 



Рис. 2.7. Сканери



Рис. 2.8. Веб-камера



Рис. 2.9. Гарнітура

Num Lock). Окремий блок утворюють клавіші керування курсором; про їх призначення неважко здогадатися з назви. Спеціальні клавіші розташовано в різних місцях клавіатури.

Маніпулятор «миша» (або просто миша) — це вказівний пристрій введення. Переміщуючи його, користувач керує вказівником — спеціальним курсором, що відображається на моніторі. Миша передає відомості про своє переміщення програмі, яка у відповідь переміщує зображення вказівника. Взагалі миша має кілька кнопок (дві, три або більше), коліщатко та інші додаткові елементи керування. З допомогою миші запускають програми, відкривають файли, переміщують фрагменти зображень тощо.

Сканер

З допомогою сканера (рис. 2.7) створюють цифрові копії зображення з паперових або інших носіїв. Процес отримання такої копії називають скануванням. Відскановане зображення, на якому є текст, можна перетворити на текстовий документ, використавши спеціальну програму для оптичного розпізнавання текстів. Сканери бувають планшетними і ручними. Останні зазвичай використовують для зчитування штрих-кодів.

Веб-камера

Так називають цифрову фотокамеру, що в реальному часі фіксує зображення, призначені для подальшого передавання через Інтернет. Зазвичай веб-камери (рис. 2.8) транслиують зображення міських вулиць, приміщень, де проходять конференції, тощо. Веб-камера потрібна також для відеозв'язку через Інтернет з допомогою програм на зразок Skype (Скайпу).

Мікрофон

Щоб записати звук або обмінятися голосовими повідомленнями через Інтернет, потрібен комп'ютер, оснащений *мікрофоном*. Це пристрій, що перетворює звукові коливання на коливання електричного струму. Комп'ютери деяких видів (наприклад, ноутбуки) мають вбудований мікрофон і веб-камеру, натомість для настільних ПК ці пристрої зазвичай доводиться купувати окремо. Часто мікрофон поєднується з навушниками в одному пристрої, який називається гарнітурою (рис. 2.9).



Інші пристрої і технології введення

Окрім названих вище, існують й інші пристрої введення. Це, наприклад, ігрові пристрої введення (джойстики, рулі, штурвали, педалі, танцювальні майданчики тощо) та графічні планшети (дігітайзери), призначені для малювання від руки в графічних програмах (рис. 2.10). Пристроєм введення графічної інформації можна вважати цифрову камеру.



Рис. 2.10. Дігітайзер


Поширення набувають також різноманітні сенсорні технології, що надають можливість керувати пристроями з допомогою пальців. У ноутбуках це — *тачпад* (невелика сенсорна панель, що дублює функції миші), а в планшетних ПК та смартфонах — *тачскрін* (сенсорний екран).

Монітор

На моніторі у вигляді текстової та графічної інформації відтворюються результати роботи комп'ютера. Крім того, на моніторі відображаються елементи керування, які користувач може вибрати з допомогою миші та активізувати в такий спосіб певну дію (ініціювати виконання команди, видалити фрагмент тексту тощо). Зображення на моніторі формується з пікселів. Донедавна поширеними були монітори на основі електронно-променевої трубки (ЕПТ-монітори), але тепер їх замінили РК-монітори, тобто рідинно-кристалічні. Основна характеристика монітора — *розмір діагоналі його екрана*.



Дізнатися більше

про відтворення зображення на екрані монітора ви зможете на диску. 

Звукові колонки


Більшість сучасних персональних комп'ютерів обладнано звуковими колонками, призначеними для відтворення звуку (рис. 2.11). Таких колонок може бути від двох до восьми. Колонки поділяються на пасивні (не мають власного підсилювача) та активні (мають підсилювач).



Рис. 2.11. Колонки



Дізнатися більше

про види принтерів і принцип їх роботи ви зможете на диску. 

Принтер

Призначення принтера — виведення результатів роботи комп'ютера на папір чи прозору плівку. Хоча існує кілька різновидів цих пристроїв, сьогодні користувачі віддають перевагу двом з них, а саме: струменевим та лазерним.



4.7.

МУЛЬТИМЕДІЙНЕ ОБЛАДНАННЯ

Під терміном мультимедіа розуміють поєднання різних форм подання інформації (текстової, графічної, аудіо- і відеоформ). Для відтворення мультимедіа застосовують комп'ютери та спеціальне обладнання.



Це мультимедійні проектори, які можуть проектувати на екран відео- і комп'ютерні зображення, а також інтерактивні електронні дошки. Хоча така дошка має вигляд звичайної білої дошки, все, що записується на її поверхні, миттєво з'являється на екрані комп'ютера. Електронну дошку можна використовувати як інтерактивний монітор — у такому разі вона перетворюється на великий сенсорний екран.

4.8.

ВІД АБАКА ДО СУПЕРКОМП'ЮТЕРА

Перш ніж досягти сучасного рівня, обчислювальна техніка пройшла тривалий шлях розвитку. Загалом усю її історію можна поділити на три етапи — домеханічний, механічний та електронно-обчислювальний.



Історичні відомості

про те, як розвивалася обчислювальна техніка, ви можете дізнатися на диску.

Сучасні комп'ютери залежно від розміру, призначення і вартості поділяють на шість типів.

Персональні комп'ютери. Комп'ютери, якими в певний момент часу може користуватися одна особа. До цієї категорії належать настільні комп'ютери, ноутбуки, планшетні комп'ютери.

Вбудовані комп'ютери. Це невеликі комп'ютери, якими оснащують побутові прилади на зразок пральних машин, а також автомобілі, виробниче устаткування тощо. Приміром, в автомобілі комп'ютер керує клімат-контролем, роботою двигуна та інших систем, спрощує паркування, надає різноманітну інформацію водієві.

Мобільні пристрої. До комп'ютерів цього типу належать смартфони й інші комунікаційні пристрої, що мають обчислювальні можливості та здатні підключатися до Інтернету. До цієї категорії можна віднести портативні програвачі й ігрові пристрої, які підключаються до Інтернету.

Сервери середнього класу. Це комп'ютери, призначені для інформаційних систем невеликих і середніх підприємств. Такі сервери потужніші за персональні комп'ютери, можуть одночасно обслуговувати багато користувачів, які підключаються до них з допомогою своїх персональних комп'ютерів.



Сервери вищого класу. Потужні комп'ютери, що їх встановлюють великі підприємства, банки, мобільні оператори. Сервери цього класу розраховані на безперервну цілодобову роботу й одночасно обслуговують запити тисяч користувачів.

Суперкомп'ютери. Так називають комп'ютери з величезною обчислювальною потужністю. Їх використовують для виконання завдань, які вимагають інтенсивних обчислень, зокрема для прогнозування погодних умов, керування косміч-



ними польотами. Традиційний суперкомп'ютер коштує кілька мільйонів доларів. Щоб зменшити його вартість, почали використовувати суперкомп'ютерні кластери, що складаються з кількох серверів середнього класу, які діють як один. Прикладом такого суперкомп'ютера є встановлений у Київському політехнічному університеті суперкомп'ютер, який сьогодні найпотужніший в Україні. Він увійшов до складу 500 найкращих суперкомп'ютерів світу. Його потужність еквівалентна тисячі комп'ютерів нового покоління.



ВПРАВА 3



Завдання. Виконати завдання на слайдах презентації **Vprava_3.ppt**, яка знаходиться в папці **Files** на диску.

Описи завдань вміщено в примітках до слайдів.



Висновки

Комп'ютер — це програмований електронний пристрій, який приймає дані, обробляє їх, відображає результати у вигляді інформаційних повідомлень і за потреби зберігає дані для подальшого їх використання.

Центральний процесор (ЦП) — основний компонент комп'ютера, призначений для керування всіма його пристроями та виконання арифметичних і логічних операцій над даними.

Материнська плата — пристрій, на якому встановлено внутрішні компоненти комп'ютера. Використовується для передавання даних між усіма його пристроями.

Оперативна пам'ять — швидкодіюча пам'ять, призначена для записування, зберігання і читання інформації у процесі її обробки.

Постійна пам'ять — швидкодіюча енергонезалежна пам'ять, призначена для зберігання інформації, що не змінюється під час виконання програм. Ця пам'ять забезпечує лише можливість зчитування інформації.

Зовнішня пам'ять — це пам'ять, призначена для тривалого зберігання даних, цілісність яких не залежить від того, ввімкнено чи вимкнено комп'ютер.

Найпоширеніші пристрої введення інформації — клавіатура, миша, сканер, веб-камера, мікрофон; пристрої виведення інформації — монітор, звукові колонки та принтер.



Контрольні запитання та завдання

1. Для чого призначено центральний процесор?
2. Які функції виконує оперативна пам'ять?
3. Яка частина внутрішньої пам'яті є енергонезалежною?
4. Які пристрої зберігання даних ви знаєте?
5. Назвіть найпоширеніші пристрої введення та виведення інформації.
6. Які існують типи комп'ютерів?



Питання для роздумів

- 1*. Що сприяло поширенню персональних комп'ютерів?
- 2*. Які ви бачите перспективи розвитку обчислювальної техніки?



Завдання для досліджень

- 1*. Відшукайте інформацію про 3D-принтери та перспективи їх застосування.
- 2*. Визначте ємність жорсткого диска, встановленого на вашому комп'ютері в класі та вдома.
- 3*. Відшукайте інформацію про еволюцію носіїв інформації.
- 4**. З'ясуйте, що таке дата-центр і які послуги він надає.

§5

Вибір складових комп'ютера під потребу

5.1.

ПОЗНАЧЕННЯ КОМПЛЕКТУЮЧИХ СИСТЕМНОГО БЛОКА КОМП'ЮТЕРА

Настільні персональні комп'ютери та ноутбуки користувачі можуть використовувати для роботи з офісними документами, відтворення мультимедіа, для ігор тощо. Конфігурація комп'ютера і мінімальні параметри його складових, що дадуть змогу виконувати ці завдання, різні. Тому потрібно навчитися підбирати оптимальну комплектацію комп'ютера, щоб не витратити зайві кошти й отримати комп'ютер, який відповідатиме всім вимогам користувача під час роботи і відпочинку.



Почнемо з дослідження інформації про комплектуючі, які ви можете побачити в магазині комп'ютерної техніки або на сайті інтернет-магазину. Наводимо приклади описів системних блоків, що пропонують в одному з інтернет-магазинів:

1. AMD Athlon II X4 740 (3.2 ГГц) / RAM 4 Гб / HDD 1 Тб / nVidia GeForce GT 730, 2 Гб / DVD±RW / LAN;
2. Intel Celeron Dual Core G540T (2.1 ГГц) / RAM 2 Гб / HDD 250 Гб / Intel HD Graphics 2500 / без ОД / LAN;
3. Intel Celeron Dual Core G1820 (2.7 ГГц) / RAM 4 Гб / HDD 500 Гб / Intel HD Graphics / без ОД / LAN.

Проаналізувавши ці дані, розподілимо їх за компонентами, що дасть змогу порівняти їхні характеристики (табл. 2.1).

Таблиця 2.1. Порівняння характеристик системних блоків

Компонент	ПК1	ПК2	ПК3
Процесор	AMD Athlon II X4 740 (3.2 ГГц) (4 ядра по 3,2 ГГц)	Intel Celeron Dual Core G540T (2.1 ГГц) (2 ядра по 2,1 ГГц)	Intel Celeron Dual Core G1820 (2.7 ГГц) (2 ядра по 2,7 ГГц)
Оперативна пам'ять	RAM 4 Гб	RAM 2 Гб	RAM 4 Гб
Жорсткий диск	HDD 1 Тб	HDD 250 Гб	HDD 500 Гб
Відеокарта	nVidia GeForce GT 730, 2 Гб (дискретна відеокарта з пам'яттю 2 Гб)	Intel HD Graphics 2500 (інтегрована відеокарта)	Intel HD Graphics (інтегрована відеокарта)
Оптичний диск-вод	DVD±RW	без ОД	без ОД

5.2.

ОГЛЯД КОМП'ЮТЕРНИХ КОМПЛЕКТУЮЧИХ

Розглянемо характеристики комп'ютерних компонентів, на які слід звертати увагу, вибираючи комп'ютер залежно від потреб.

Центральний процесор і оперативна пам'ять

Як зазначалося, сьогодні на ринку центральних процесорів переважають два виробники — Intel Corporation та Advanced Micro Devices Inc. (AMD). Кожний з них пропонує повну лінійку сучасних процесорів з інтегрованою графікою, що дозволяє в деяких випадках обійтися без зовнішньої відеокарти. Ці лінійки умовно можна розділити за призначенням (табл. 2.2).



**Таблиця 2.2.** Модель процесора та обсяг ОЗП для комп'ютерів різного призначення

Компонент	Потужні ігри	Ігри, обробка мультимедійних даних і графіки	Невимогливі ігри, любительська обробка відео і фото	Офісні завдання, навчання
Процесор Intel®	Intel® Core™ i7 (i7-4770, i7-4820K, i7-4960X)	Intel® Core™ i5 (i5-4440, i5-4570, i5-4670K)	Intel® Core™ i3 (i3-4130, i3-4340)	Intel® Pentium® (G3220, G3430)
Процесор AMD	AMD A10 APU (A10-5700, A10-6700T, A10-6800K)	AMD A8 APU (A8-5500, A8-6500T, A8-6600K)	AMD A6 APU (A6-5400K, A6-6400K)	AMD A4 APU (A4-4000, A4-5300, A4-6300)
ОЗП	16–32 Гбайт	8–16 Гбайт	4–8 Гбайт	2–4 Гбайт

На відміну від процесорів, уся сучасна оперативна пам'ять представлена виробниками належить до одного типу — DDR3 і відрізняється лише робочою частотою (швидкодією) та можливістю працювати на підвищеній частоті. Вибір оперативної пам'яті для комп'ютера визначається насамперед фінансовими можливостями покупця та довірою до певного виробника.

Частоти найпоширеніших модулів пам'яті знаходяться у межах 1066–1866 МГц. Найпопулярніші на сьогодні виробники — Kingston, Hynix, Crucial (для ігрових систем — Geil та G.Skill).

Краще використовувати два однакових модулі пам'яті меншої ємності, ніж один більшої, оскільки в цьому випадку вмикається більш швидкий двоканальний доступ до пам'яті. Не варто гнатися за великою ємністю пам'яті та встановлювати 16 Гбайт, оскільки практичної користі від цього не буде. Windows 7/8/10 з усіма запущеними програмами та антивірусною програмою вимагає 0,5–1 Гб, а потужні сучасні ігри зрідка використовують більше 2 Гб, тому для офісних комп'ютерів достатньо 2 Гб, для універсальних — 4 Гб, для ігрових — 8 Гб.

Материнські плати

Чи не першим параметром, на який потрібно звертати увагу при виборі комплекту процесор-материнська плата, є тип з'єднання (socket), за допомогою якого процесор кріпиться до плати. Для сучасних процесорів Intel® — Socket LGA1150, для AMD — Socket FM2(+).



Відеокарти

На ринку відеоадаптерів представлено продукцію двох виробників графічних процесорів: nVidia® та Ati™, які на сьогодні пропонують відеокарти на будь-який смак. Відеокарти бувають інтегровані, зовнішні та дискретні.

Інтегровані відеокарти вбудовуються в материнську плату і заміні не підлягають. Подібний варіант ідеально підходить для вирішення невимогливих офісних завдань.

Зовнішні відеокарти найчастіше підключають до комп'ютера нестандартним способом і призначені для розширення можливостей малопотужних мобільних комп'ютерів (ноутбуки, нетбуки тощо).

Дискретна відеокарта фактично є окремою платою, яка може бути встановлена у відповідний слот материнської плати. Таку карту можна замінити, а система на її основі може бути модернізована без особливих зусиль шляхом встановлення нової дискретної відеокарти.

Вибираючи відеокарту, особливу увагу треба звертати на такі основні характеристики:

- обсяг відеопам'яті;
- тип пам'яті;
- розрядність шини пам'яті;
- частота графічного процесора та пам'яті;
- спосіб охолодження.

Дискові накопичувачі

Найбільш потужними виробниками жорстких дисків сьогодні вважаються компанії WD, Seagate, Hitachi та Samsung. У домашніх комп'ютерах в основному використовують моделі дисків ємністю від 250 Гбайт до 6 Тбайт.

Для офісного робочого комп'ютера достатньо 500 Гбайт, а, скажімо, для відеомонтажу може і 15 Тбайт не вистачати. Тому обсяг диска потрібно вибирати відповідно до потреб користувача. Ставити дорогий місткий диск у робочий ПК, на якому будуть встановлені тільки операційна система й офісні програми, не доцільно.

Чим вища швидкість обертання магнітних пластин, тим швидший (продуктивніший) жорсткий диск. Однак із зростанням швидкості збільшується його енергоспоживання і шум. Найпоширеніші моделі мають швидкість обертання 5 400, 5 900, 7 200 або 10 000 об/хв.

Диски зі швидкістю обертання 5400 – 5900 об/хв — оптимальний варіант для робочого комп'ютера або ноутбука. Вони тихо працюють, не дуже нагріваються і досить економічні. Швидкість передавання даних у таких моделях досягає 150 Мбіт/с. Як правило, чим менша швидкість обертання, тим довше служить накопичувач.



Моделі зі швидкістю обертання 7200 об/хв підійдуть для стаціонарних комп'ютерів середнього і вищого класу. Реальна швидкість читання / запису таких накопичувачів досягає 180 Мбіт/с.

Вибір приводу оптичних дисків

У зв'язку з розвитком швидкісних каналів Інтернету привід оптичних дисків практично не потрібний. Якщо користувачу все ж таки цей привід потрібний, то це має бути привід DVD-RW. За ступенем надійності користувачі зупиняють свій вибір на приводах таких відомих виробників, як LG, Samsung, Asus.



Контрольні запитання та завдання

1. Для яких цілей зазвичай використовують персональні комп'ютери?
2. Як це впливає на вибір їхньої конфігурації?
3. Чи завжди наявність більшого обсягу ОЗП впливає на швидкодію комп'ютера?
4. Що таке дискретна відеокарта? Які переваги вона має порівняно із вбудованою?



Завдання для досліджень

1. Порівняйте основні характеристики (центральний процесор, оперативна пам'ять) настільних комп'ютерів і смартфонів. Зробіть висновок.

ПРАКТИЧНА РОБОТА №2

Конфігурація комп'ютера під потребу


Завдання. Використовуючи дані якого-небудь інтернет-магазину комп'ютерної техніки, підібрати три системні блоки під потребу одного з користувачів (згідно вашого варіанта) з розрахунку: найвища якість — найнижча ціна. Скласти порівняльну таблицю. Вказати, який з обраних комп'ютерів найбільше відповідає вимогам. Обґрунтувати свій вибір.

Найменування комплектуючого	ПК1	ПК2	ПК3
Процесор			
...			



- Варіант 1. Для учня 8-го класу.
- Варіант 2. Для бухгалтера підприємства.
- Варіант 3. Для веб-дизайнера.
- Варіант 4. Для тестувальника сучасних ігор.
- Варіант 5. Для учителя математики.
- Варіант 6. Для оператора відеомонтажу.



Перевірити свої знання і пройти тест ви зможете на диску. 

§6

Програмне забезпечення

6.1.

РОЛЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ, ЙОГО КЛАСИФІКАЦІЯ

Одна з причин, чому комп'ютерні технології так швидко проникли в усі сфери нашого життя, полягає в тому, що комп'ютер є *універсальним* пристроєм, тобто він може використовуватися для виконання широкого кола завдань, пов'язаних із обробкою, зберіганням та передаванням інформації. Забезпечується ця універсальність завдяки такій складовій комп'ютерних систем, як програмне забезпечення (ПЗ).

Під **програмним забезпеченням** розуміють сукупність комп'ютерних програм і даних, які містять інструкції для комп'ютера стосовно того, що та як він має робити.

Комп'ютер є програмованим пристроєм. Це означає, що він усе робить у відповідності до програми. Відразу після виготовлення комп'ютер не містить жодних програм і фактично не може працювати. На такий комп'ютер встановлюють (інсталюють) певне програмне забезпечення, і воно, а також програми, з якими комп'ютер може працювати через мережу, визначають, які завдання можна буде виконувати з його допомогою.

Встановлені програми зберігаються на зовнішніх пристроях комп'ютера та в його постійній пам'яті. Щоб комп'ютер почав працювати згідно з тією чи іншою програмою, її потрібно запусити на виконання. Це може бути зроблено в автоматичному режимі (наприклад, після увімкнення комп'ютера) або вручну користувачем, який клацає на ярлик програми. Після цього програма і потрібні їй дані зчитуються з диска та завантажуються в оперативну пам'ять. Центральний процесор послідовно зчитує програму з пам'яті та обробляє інструкції, що в ній містяться.



Зазначимо, що користувачеві може бути доступне програмне забезпечення, розміщене не лише на його комп'ютері, а й в мережі (зокрема, в Інтернеті) — звичайно, за умови, що його комп'ютер підключено до цієї мережі.

За час існування комп'ютерної техніки було розроблено дуже багато різноманітного програмного забезпечення. Залежно від призначення його поділяють на *прикладне, системне та інструментальне* (рис. 2.12).



Рис. 2.12. Класифікація програмного забезпечення

6.2.

ПРИКЛАДНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ



Прикладне програмне забезпечення призначене для виконання певних завдань користувача та зазвичай розраховане на безпосередню взаємодію з ним.

Сьогодні існує велике розмаїття прикладних програм. Серед них є окремі спеціалізовані програми (медіапрограваачі, браузері, відеоредактори тощо), набори (пакети) програм (Microsoft Office — для офісних працівників, Adobe Suite — для дизайнерів), великі системи для керування цілими підприємствами і виробничими процесами (SAP R3).



6.3.

СИСТЕМНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

Зазвичай прикладне програмне забезпечення не здатне працювати без системного програмного забезпечення, до якого належать *операційна система, драйвери та утиліти*.

Операційна система (ОС) — це набір програм, який забезпечує зв'язок між пристроями і прикладними програмами та керує пристроями, обчислювальними процесами і ресурсами.

Операційна система виконує такі завдання, як завантаження програм в оперативну пам'ять, їх виконання, забезпечення доступу до пристроїв введення-виведення, керування оперативною пам'яттю і доступом до даних на жорстких дисках. Операційна система також надає інтерфейс користувача.

Вона є ключовим компонентом комп'ютерної системи, який утворює прошарок між прикладними програмами та апаратним забезпеченням.

Проте не всім обчислювальним пристроям потрібна операційна система. Наприклад, без неї працюють мікрокомп'ютери, вбудовані в побутові прилади, автомобілі, мобільні телефони. Ці пристрої постійно виконують лише одну програму, яка запускається на виконання, коли їх вмикають. Багато ігрових приставок також не потребують операційної системи, оскільки при ввімкненні вони запускають програму з картриджа або компакт-диска.

Необхідність в операційній системі виникає, коли обчислювальна система має виконувати багато завдань та коли користувачу потрібно надати можливість керувати виконанням окремих програм.

Сьогодні найпоширенішими операційними системами для настільних комп'ютерів і серверів є Windows та UNIX/Linux різних версій. Для мобільних пристроїв і планшетних ПК також розроблено спеціальні операційні системи, представниками яких є Android та Windows Mobile.

Найчастіше операційні системи класифікують за такими ознаками:

за цільовим пристроєм — існують операційні системи для потужних комп'ютерів-серверів, для персональних комп'ютерів, мобільних пристроїв;

за кількістю програм, що виконуються одночасно, — розрізняють однозадачні та багатозадачні операційні системи;





Дізнатися більше
про процес
завантаження
комп'ютера ви зможете
на диску.

за типом інтерфейсу — з текстовим інтерфейсом і графічним;

за кількістю розрядів даних, що обробляються одночасно — ця класифікація ґрунтується на розрядності процесорів. Скористатися перевагами більшої розрядності можна за умови, що ОС та прикладні програми розроблені спеціально для процесора з такою розрядністю.

6.4.

СКЛАДОВІ ОПЕРАЦІЙНОЇ СИСТЕМИ

Операційна система має такі складові:

- ядро, що містить планувальник, який керує розподілом ресурсів;
- драйвери пристроїв, які допомагають керувати обладнанням;
- мережева підсистема;
- файлова система;
- системні бібліотеки;
- оболонка з утилітами (інтерфейс користувача).

Одні з цих складових постійно перебувають в оперативній пам'яті комп'ютера, поки він працює, інші завантажуються і запускаються на виконання лише за потреби.

! Зауважимо, що операційна система — дуже складний програмний продукт, який потребує обережного ставлення. Тому не видаляйте файлів, призначення яких вам невідоме, не порушуйте процес завантаження системи і не змінюйте налаштування, якщо не впевнені в результаті.



Дізнатися більше
про драйвери ви
зможете на диску.

Важливою складовою операційної системи є *драйвери* — спеціалізовані програми, з допомогою яких забезпечується доступ програм до апаратного забезпечення.

До системного програмного забезпечення відносять також *утиліти* — програми, призначені для виконання таких завдань, як захист від вірусів, запис даних на компакт-диски, догляд за жорстким диском, моніторинг роботи комп'ютера, точне настроювання його параметрів тощо.

Проте такий поділ програмного забезпечення за призначенням не є однозначним. Наприклад, браузер Internet Explorer, який, без сумніву, належить до прикладних програм, є невід'ємною складовою операційної системи Windows, а деякі програми запису компакт-дисків, які загалом вважаються утилітами, мають стільки додаткових функцій, що їх можна віднести до прикладних програм.



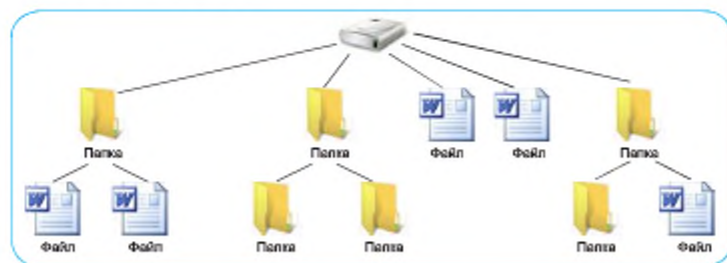
Звичайні користувачі безпосередньо не взаємодіють з такими компонентами операційної системи, як ядро або системні бібліотеки, натомість із файловою системою та інтерфейсом користувача стикаються повсякчас.

6.5. ФАЙЛОВА СИСТЕМА

Ви, звичайно, знаєте, що *файлом* називають іменовану сукупність даних, яка зберігається на зовнішньому запам'ятовуючому пристрої і пересилається та обробляється як єдине ціле. Файл може містити програму, числові дані, текст, закодоване зображення тощо.

Порядок, що визначає спосіб організації, зберігання та іменування файлів на носіях даних, називають **файловою системою**.

Забезпечують такий порядок відповідні програми, які входять до складу операційної системи. Зазначимо, що поняття файлової системи застосовується не лише до комп'ютерів, а й до інших оснащених пристроями зберігання пристроїв (цифрових камер, мобільних телефонів, електронних книжок тощо).



Файлова система

- Організовує кластери у файли і каталоги.
- Відслідковує, які кластери зайняті, вільні або пошкоджені.
- Відслідковує призначення імен файлам.
- Надає можливість впорядковувати файли за каталогами.
- Надає засоби для керування файлами (видалення, копіювання тощо).
- Забезпечує шифрування файлів.
- Надає можливості керувати доступом до файлів.
- Надає програмам засоби для запису даних у файли.

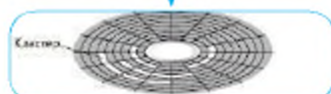


Рис. 2.13. Місце і роль файлової системи



Одне із завдань файлової системи — забезпечити ефективне використання фізичного простору носія. Це досягається завдяки тому, що фізично файли розташовуються на ньому не суцільними великими масивами, а невеликими блоками — *кластерами*. Такий підхід дозволяє уникнути ситуації, коли внаслідок видалення файлу на диску утворюється порожній простір і його не можна заповнити, оскільки неможливо підібрати файл, що має такий самий обсяг. Під час зберігання файлу на диску саме драйвери файлової системи забезпечують такий його запис.

Файлова система (рис. 2.13) також визначає довжину *імені файлу*, допустимі в іменах символи, максимально можливий обсяг файлу, набір його атрибутів (властивостей).

Для впорядкування файлів на пристроях файлова система надає можливість створювати структури *каталогів*. У системі Windows їх називають *папками* (за аналогією з канцелярськими папками). *Каталог* — це іменованний об'єкт файлової системи, який може містити файли та інші каталоги.

Загалом існує понад п'ятдесят файлових систем, які різняться підходами до іменування файлів, методами організації їх у каталоги та додатковими можливостями, такими як шифрування файлів чи встановлення обмежень на доступ до них.

6.6.

ФОРМАТУВАННЯ НОСІЇВ ДАНИХ

Файлова система перед першим використанням носія має відповідним чином його підготувати, зокрема визначити, в якій ділянці розміщуватиметься перелік файлів і каталогів та якими будуть кластери. З цією метою нові носії з магнітним способом запису завжди піддають форматуванню.

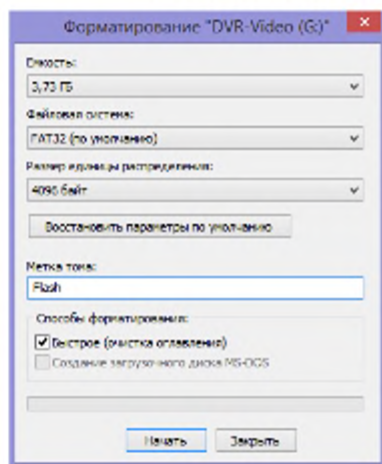


Рис. 2.14. Діалогове вікно форматування пристрою

Форматування — це розмітка пристроїв зберігання файловою системою.

Розмітка носіїв з лазерним способом запису виконується під час запису даних на них. Форматують також пристрої, на яких вже є дані. При цьому можливі такі варіанти:

- повне форматування — передбачає повторну розмітку диска, а тому всі дані, що містяться на ньому знищуються;
- швидке форматування — очищуються кореневий каталог і таблиці розміщення файлів, дані в кластерах залишаються, а отже, існує можливість їх відновлення.



Практично кожному користувачеві доводиться формувати свою флешку, а тому розглянемо послідовність дій.

1. Вставити флешку в роз'єм порту USB.
2. Відкрити вікно **Мій комп'ютер**.
3. У контекстному меню флеш-накопичувача вибрати команду **Формувати...**
4. У вікні **Формувати** задати параметри форматування (рис. 2.14).
5. Натиснути кнопку **Почати**. Потім натиснути кнопку **Закрити**.



Висновки

Залежно від призначення програмне забезпечення поділяють на прикладне, системне та інструментальне.

Прикладне програмне забезпечення не здатне працювати без системного програмного забезпечення, до якого належать *операційна система, драйвери й утиліти*.

Операційна система — це набір програм, який забезпечує зв'язок між пристроями і прикладними програмами та керує пристроями, обчислювальними процесами і ресурсами.

Драйвери — спеціалізовані програми, з допомогою яких забезпечується доступ програм до апаратного забезпечення.

Утиліти — програми, призначені для виконання таких завдань, як захист від вірусів, запис даних на компакт-диски, догляд за жорстким диском, моніторинг роботи комп'ютера, точне налаштування його параметрів тощо.

Порядок, що визначає спосіб організації, зберігання та іменування файлів на носіях даних, називають *файловою системою*.

Форматування — це розмітка пристроїв зберігання файловою системою.



Контрольні запитання та завдання

1. Що таке програмне забезпечення комп'ютера? Яку роль воно відіграє?
2. Перелічіть види прикладного програмного забезпечення. Наведіть приклади.
3. Яке програмне забезпечення вважається системним?
4. Які складові операційної системи? Яка роль кожної з них?
5. Які завдання виконує файлова система?
6. Яке призначення має операція форматування носія даних?



Питання для роздумів

1. Чому програму MS Excel можна віднести до прикладного програмного забезпечення?
2. Чому, на вашу думку, спостерігається поступове стирання відмінностей між настільними та мобільними ОС?
3. Чому планшетні ПК постачаються із встановленою ОС?



Завдання для досліджень

1. З'ясуйте, чи встановлено на вашому мобільному пристрої ОС. Якщо так, то яку.
2. Знайдіть інформацію про операційну систему Linux. Які переваги та недоліки має ОС Linux порівняно з Windows?
3. З'ясуйте, що таке хмарні обчислення. Де використовують цю технологію?

§7

Розробка і розповсюдження програмного забезпечення

7.1.

ІНСТРУМЕНТАЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ



Програмне забезпечення, яке використовується на етапах проектування, розробки і тестування іншого програмного забезпечення, називають *інструментальним*. Для кожного з перелічених етапів воно своє. Зокрема, розробники (програмісти) зазвичай працюють в інтегрованому середовищі розробки, що має зручний графічний інтерфейс та засоби для виконання всіх завдань. Аналогом такого середовища призначеного для дітей є знайома вам програма Scratch.

7.2.

РОЗПОВСЮДЖЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Дізнатися більше
про створення і підтримку програмного забезпечення ви зможете на диску.



Програмне забезпечення є результатом праці багатьох людей, але, на відміну від одягу, меблів чи смартфонів, воно є нематеріальним продуктом. Постачатися програмне забезпечення може у встановленому вигляді разом із пристроєм (комп'ютером, планшетом тощо) або окремо — на носіях або через завантаження його файлів з Інтернету.



Майже все програмне забезпечення захищається законами про авторське право. Правила використання такого програмного забезпечення визначаються ліцензією.

Ліцензія на програмне забезпечення — правовий документ, що визначає правила використання та поширення програмного забезпечення.

Загалом усе ліцензоване програмне забезпечення можна розділити на дві великі групи.

Власницьке, або пропрієтарне (англ. proprietary software), програмне забезпечення є платним. Видавці (розробники) можуть продавати його самі або через посередників (дистриб'юторів, реселерів). Ліцензія посередника надає йому право лише продавати програмне забезпечення. Існують також спеціальні ліцензії для постачальників і виробників обладнання, що надають їм право встановлювати ПЗ на свої вироби.

Приватні особи і підприємства, що купують ПЗ окремо або разом з обладнанням, вважаються кінцевими користувачами. Їхня ліцензія називається ліцензією кінцевого користувача (End User License Agreement — EULA). Вона регламентує, скільки копій програмного забезпечення можна встановити або скільки користувачів можуть мати доступ до встановленого ПЗ. Кінцевий користувач є лише власником своєї копії ПЗ, він не має авторських прав на ПЗ, а отже, і прав на його розповсюдження (передавання), навіть безкоштовне.

Текст ліцензійного договору кінцевого користувача, як правило, вбудований у програмний продукт і відображається під час його встановлення. Користувач повинен прочитати і прийняти його, і лише після цього він зможе користуватися програмним продуктом.

Вільне та відкрите програмне забезпечення (англ. FLOSS — free/libre/open source software) є переважно безкоштовним (у деяких випадках користувачам пропонують зробити пожертву для розробників програми. Але це не означає, що для цього ПЗ зовсім не існує обмежень.



Дізнатися більше

про вільне і відкрите програмне забезпечення та ознайомитися з термінами з ліцензування ви зможете на диску.



7.3.

ІНСТАЛЯЦІЯ ТА ДЕЙНСТАЛЯЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Як зазначалося, відразу після виготовлення комп'ютер не містить жодних програм і фактично не може працювати. Спочатку на нього встановлюють (інсталюють) відповідну операційну систему (якщо це передбачено), а потім — певне програмне забезпечення. Більшість настільних комп'ютерів тепер продають з уже встановленою ОС, планшети



та смартфони без ОС взагалі не постачають, тому користувачеві немає потреби вміти інсталювати ОС. Але як встановлювати прикладні програми, повинен знати кожний користувач.

Зазвичай програми постачаються на дисках або завантажуються з Інтернету у вигляді дистрибутиву — набору, що містить спеціально призначені для інсталяції файли, а також програму-інсталятор. Достатньо запустити його, і він сам виконає всю роботу з інсталяції — перевірить, чи відповідає система вимогам програми, розпакує і перепише у відповідні папки на жорсткому диску файли програми, внесе відповідні зміни в конфігураційні параметри комп'ютера.

Під час своєї роботи інсталятор виводить діалогові вікна, в яких користувачеві пропонується обрати функціональні компоненти програми, папку для її розміщення тощо. Після завершення роботи інсталятора програму можна запускати на виконання.

Якщо умовами ліцензування передбачена процедура активації ПЗ, після першого запуску програми відобразиться вікно для введення ключа продукту. Правильність ключа перевіряється через Інтернет, тому для успішної активації потрібно, щоб комп'ютер було підключено до Всесвітньої мережі.

Зазначимо, що процедури інсталяції потребують не всі програми. Є програми, для яких процес встановлення обмежується копіюванням файлів програми на жорсткий диск. Його виконує користувач самостійно. Такі програми називають *портативними* або переносними (portable application).

Для вилучення їх з комп'ютера достатньо видалити файли програми з диска. А от програми, що пройшли процедуру інсталяції, у таких спосіб у жодному разі вилучати не можна. Натомість потрібно за допомогою системних засобів або деінсталятора самої програми виконати її деінсталяцію. Під час цієї процедури не лише видаляються файли програми, а й відновлюються змінені нею параметри в операційній системі.

7.4.

СУМІСНІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Ведучи мову про сумісність певних об'єктів, ми зазвичай маємо на увазі їхню здатність взаємодіяти між собою. Щодо сумісності комп'ютерів у цілому, то тут можна виділити сумісність апаратну (на рівні вузлів), інформаційну (на рівні даних) та програмну (здатність виконувати однакові програми з тим самим результатом).

Щодо програмного забезпечення, то для успішного його використання насамперед слід враховувати сумісність з операційною системою, наприклад, прикладні програми, призначені для використання в системі Windows, не працюватимуть на комп'ютері з іншою операційною системою.



Проблема сумісності також часто виникає у випадках, коли сучасні програми намагаються встановити на комп'ютерах із застарілими операційними системами або, навпаки, давно розроблені програми — на комп'ютерах під керуванням останніх версій ОС.

Необхідно враховувати ці фактори під час встановлення програм на свої комп'ютери.

ВПРАВА 4



Завдання. Виконати завдання з презентації **Vprava_4.ppt**, яка знаходиться в папці **Files** на диску. Описи завдань вміщено в примітках до слайдів.



Висновки

Програмне забезпечення, яке використовується на етапах проектування, розробки і тестування іншого програмного забезпечення називають *інструментальним*.

Ліцензія на програмне забезпечення — правовий документ, що визначає правила використання та поширення програмного забезпечення.



Контрольні запитання та завдання

1. Що таке інструментальне програмне забезпечення? Наведіть приклади.
2. Опишіть основні етапи розробки програмного забезпечення.
3. Що таке ліцензія на програмне забезпечення?
4. На які дві групи можна розділити ліцензоване програмне забезпечення?
5. Що таке ліцензія кінцевого користувача? Які права вона надає?
6. Що таке інсталяція і деінсталяція програмного забезпечення?



Питання для роздумів

- 1*. У яких випадках зручно використовувати портативне (переносне) програмне забезпечення? Опишіть загальну схему такого використання.
- 2*. Чому програми, призначені для однієї ОС, не працюють під керуванням іншої?



Завдання для досліджень

- 1**. Знайдіть інформацію про різновиди інструментального програмного середовища. Опишіть особливості використання принаймні одного з них.
- 2**. Знайдіть інформацію про придбання, встановлення і видалення прикладних програм для смартфонів під керуванням ОС Android.

§8

Стиснення та архівування даних

8.1.

СТИСНЕННЯ ДАНИХ

Хоча ємність пристроїв зберігання постійно збільшується, питання ефективного використання дискового простору й досі залишається актуальним. Це пов'язано не лише із загальним збільшенням обсягів інформації у світі, а й зі зміною її структури — зростанням частки графічної, відео- та аудіоінформації, що зберігається у файлах великих обсягів. Обсяг файлів зменшують, стискаючи дані.

Стиснення даних — це процедура перекодування даних з метою зменшення їх обсягу.

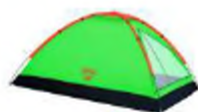


Рис. 2.15.
Приклади
стиснення
в побуті

Для цього використовують різні методи, зокрема заміну часто повторюваних послідовностей коротшим значенням (кодом). Такими послідовностями можуть бути масиви пікселів однакового кольору в растровому зображенні, слова в тексті тощо. Наприклад, будь-яка тисяча символів тексту українською мовою містить літер «о» чи «і» на порядок більше, ніж літер «ф». Припустимо, що кожний символ тексту кодується двійковою послідовністю певної довжини. Зрозуміло, що використовуючи коротші коди для поширених символів, загальна довжина закодованого тексту буде меншою.

Буває стиснення *без втрат*, коли вихідні дані можна відновити без викривлень, і з *втратами*, коли дані відновлюються лише з викривленнями, які проте можуть бути непомітними для людини. Стиснення без втрат застосовують до текстової інформації, а стиснення із втратами — до звукової, графічної та відеоінформації, і саме воно дає змогу значно зменшити обсяги даних. В останньому разі на основі спеціальних досліджень визначають, якою інформацією можна пожертвувати. Відомо, наприклад, що



зір людини дуже чутливий до зміни яскравості й значно менше — до кольорового відтінку, тому під час стиснення фотографій інформацію про колір частини пікселів можна видалити. Їхній колір під час розпакування зображення визначатиметься за кольором сусідніх пікселів.

ВИДИ СТИСНЕННЯ

без втрат

стиснений файл можна відновити в початковому вигляді, знаючи алгоритм стиснення

2+2+2+2+2+2+2+2 → 2 · 8
 КОКОС КО=* → **С
 ВАРВАР ВАР=* → **

із втратами

при стисненні частина інформації безповоротно втрачається



8.2.

ПРОГРАМИ-АРХІВАТОРИ

Для стиснення даних можна використовувати спеціальні програми — *архіватори*. Файли, створені цими програмами, називають *архівами*.

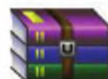
Архіватор записує вміст файлів у архів і долучає до нього інформацію про імена файлів та обсяги їхніх оригіналів. Архіватор може створити один або кілька архівів з одного чи кількох файлів.

Архіватор — це програма, що стискає та об'єднує файли.

Процедура стиснення потребує певного часу. Що більший ступінь стиснення, то менша швидкість цього процесу, і навпаки.

Програми-архіватори можуть працювати з архівними файлами різних типів. Найпопулярніші типи архівів — ARJ, ZIP, RAR, 7-Z. Найчастіше застосовують ZIP і RAR. Цю практику запроваджено завдяки програмам WinZip та WinRAR, які мають зручний інтерфейс і багато додаткових функцій, а також забезпечують досить високий ступінь стиснення.

Можливість стиснення даних не єдина перевага архіваторів. Щоб не втратити інформацію, її періодично копіюють на інші носії. Ця процедура, яку називають *резервним копіюванням*, є важливою складовою захисту даних. Для створення резервних копій зручно використовувати програми-архіватори, адже вони дають змогу впорядковувати файли. Наприклад, файли можна об'єднувати в архів за датою створення, за належністю до певного завдання чи проекту тощо.





8.3.

АРХІВУВАННЯ ФАЙЛІВ І ПАПОК У WINDOWS

В операційну систему Windows вбудовано засоби для створення ZIP-архівів та керування ними. У програмі **Провідник** архіви подаються у вигляді папок із зображенням змійки. Відкривши таку папку, ви побачите в рядку адреси ім'я файла з розширенням **.zip**.

Щоб створити архів, потрібно клацнути правою кнопкою миші на значок папки (або файла), яку необхідно стиснути, і в контекстному меню вибрати команду **Надіслати** ▶ **Стиснута папка** (рис. 2.16). У поточній папці з'явиться стиснута папка з таким самим ім'ям, як у архівованого об'єкта.

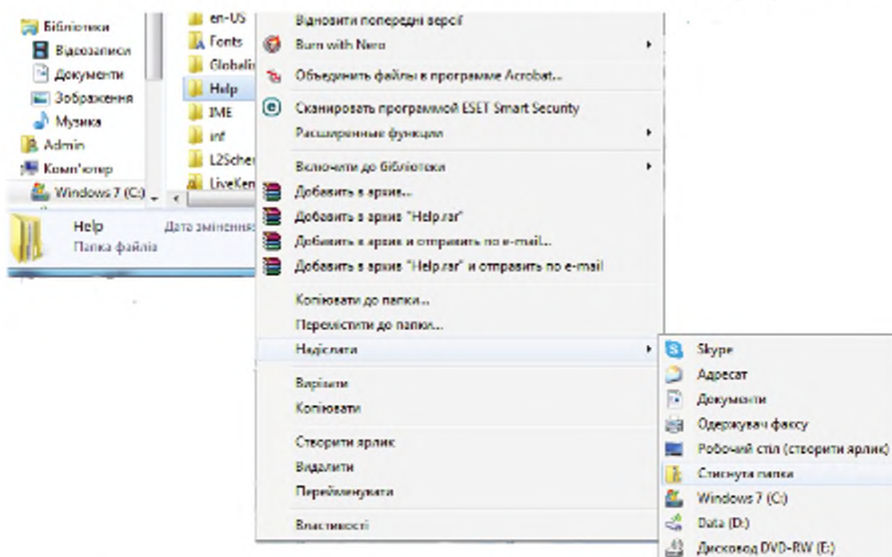


Рис. 2.16

Щоб відкрити стиснуту папку, достатньо двічі клацнути на її значок. Після цього на екрані в окремому вікні буде відображено її вміст. Для відкриття файла, збереженого в стиснутій папці, потрібно перейти до цієї папки та двічі клацнути значок файла.

Можливості засобів архівування в операційній системі Windows обмежені, тому для створення архівів та керування ними зазвичай застосовують архіватор.

8.4.

ПРОГРАМА-АРХІВАТОР WINRAR

Архіватор WinRAR має простий і зручний інтерфейс та забезпечує високий ступінь стиснення. Це умовно-безкоштовна програма, тобто її можна безкоштовно використовувати впродовж сорока днів.

Якщо програма WinRAR встановлена на комп'ютері, її можна запустити, клацнувши кнопку **Пуск** і вибравши команду **Усі програми** ▶ **WinRAR** ▶ **WinRAR**. На екрані відкриється вікно програми (рис. 2.17).

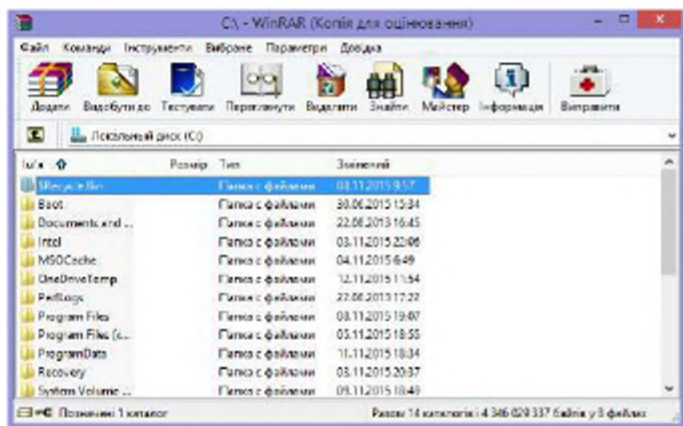


Рис. 2.17. Вікно програми WinRAR

У рядку меню програми є елементи **Файл**, **Команди**, **Інструменти**, **Вибране**, **Параметри** та **Довідка**. Меню **Файл** містить команди для керування папками, файлами та буфером обміну, а також для встановлення пароля. Усі команди створення, перегляду, тестування та відновлення архівів розміщено в меню **Команди**. З допомогою команд меню **Інструменти** архів можна перевірити на наявність вірусів, відновити його, створити звіт. Меню **Вибране** містить список вибраних папок і архівів.

8.5. АРХІВУВАННЯ ФАЙЛІВ

Після запуску програми WinRAR у її вікні буде відображено вміст тієї папки, з RAR-файлами якої працювали востаннє. Щоб створити новий архів, треба перейти до папки, де містяться файли, які потрібно заархівувати. Якщо є потреба змінити поточний диск, то можна натиснути клавіші **Ctrl+D**, або скористатися списком дисків, розташованих під панеллю інструментів, або клацнути значок із зображенням диска в нижньому лівому куті вікна.

Перейшовши до потрібної папки, з допомогою миші необхідно виділити файли і папки, які слід заархівувати, та клацнути кнопку **Додати** чи вибрати команду **Команди** ▶ **Додати файли до архіву**.

У діалоговому вікні **Ім'я архіву та параметри** слід ввести ім'я архіву чи підтвердити ім'я, запропоноване програмою за умовчанням (рис. 2.18). Тут можна також задати формат нового архіву, метод стиснення та інші параметри архівування.

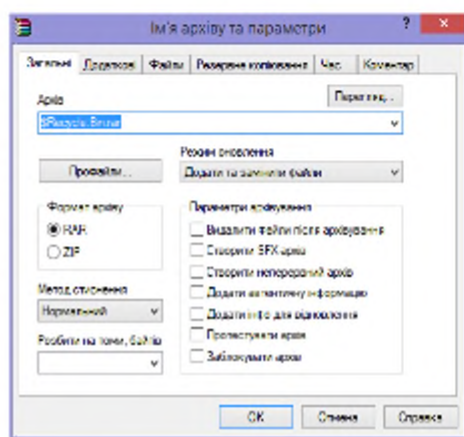


Рис. 2.18. Діалогове вікно програми WinRAR



Програма WinRAR дає змогу створювати архіви в двох форматах — RAR і ZIP. Частіше застосовують другий, хоча формат RAR здебільшого забезпечує краще стиснення.

Після вибору формату архіву потрібно задати один із шести методів стиснення: *без стиснення*, *швидкісний*, *швидкий*, *нормальний*, *хороший* чи *максимальний*. У разі застосування першого методу файли поміщаються до архіву в початковому вигляді. Найменший обсяг архів матиме в разі вибору максимального стиснення. Цей метод доцільно вибирати під час створення архівів, призначених для довгострокового зберігання або для передавання через Інтернет.

Установивши параметри архіву, у вікні **Ім'я архіву та параметри** клацніть кнопку **ОК** і запусить процес архівування.



Перевірити свої знання, пройти тест і розгадати кросворд ви зможете на диску.

Якщо на комп'ютері встановлено програму WinRAR, файли можна пакувати в RAR-архів безпосередньо в програмі **Провідник**. Для цього слід виділити файл (або папку), клацнувши його правою кнопкою миші, та в контекстному меню вибрати команду **Додати до архіву**. В результаті відкриється вікно **Ім'я архіву та параметри**.

8.6.

БАГАТОТОМНИЙ АРХІВ

Часто виникає потреба поділити архів на частини (томи), наприклад, для передавання його через Інтернет або запису на оптичні диски. Такий архів називають *багатотомним*.

Багатотомний архів — це архів RAR, що зберігається в кількох файлах, які називають томами.



abc.rar



abc.r00

Томи підтримує лише формат RAR. За умовчанням кожен том (частина багатотомного архіву) отримує ім'я **АРХІВ partNNN**, де **NNN** — номер тому. Всі томи мають бути збережені в одній папці; розпаковувати їх слід, починаючи з першого.



abc.r01



abc.r02



abc.r03

Щоб створити багатотомний архів, потрібно відкрити вікно **Ім'я архіву та параметри**, в полі **Розбити на томи, байтів** зазначити обсяг тому і клацнути кнопку **ОК**.

8.7.

САМОРОЗПАКОВУВАНИЙ АРХІВ

Якщо користувач, для якого призначено архів, не має програми для його розпакування, він не зможе видобути файли з цього архіву. У такому разі доцільно створювати саморозпаковуваний архів. Для цього у вікні **Ім'я архіву та параметри** потрібно встановити прапорець **Створити SFX-архів** (англ. self extracting archive — саморозпаковуваний архів).



Саморозпаковуваний архів — це архів із приєднаним виконуваним модулем, який дає змогу видобути файли без запуску відповідного архіватора. Ім'я такого архіву, як і будь-якого виконуваного файла, має розширення `.exe`.



Висновки

Стиснення даних — це процедура перекодування даних з метою зменшення їх обсягу.

Архіватор — це програма, що стискає та об'єднує файли. Найпопулярніші типи архівів — ARJ, ZIP, RAR, 7-Z.

Багатотомний архів — це архів RAR, що зберігається в кількох файлах, які називають томами.

Саморозпаковуваний архів — це архів із приєднаним виконуваним модулем, який дає змогу видобути файли без запуску відповідного архіватора. Ім'я такого архіву, як і будь-якого виконуваного файла, має розширення `.exe`.



Контрольні запитання та завдання

1. Що таке стиснення даних?
2. Яким буває стиснення даних?
3. Для чого застосовують програми-архіватори?
4. Що таке багатотомний архів? У яких випадках його доцільно створювати?
5. Що таке саморозпаковуваний архів? Коли його слід створювати?
6. Складіть і опишіть порядок резервного копіювання файлів з використанням програми архіватора.



Питання для роздумів

1. Одним із методів стиснення зображень є такий: у файлі окремо зберігається палітра кольорів, а при записі даних самого зображення замість кольору пікселя записується його номер у палітрі. Чи буде ефективним цей метод для стиснення чорно-білих зображень? Поясніть свою відповідь.



Завдання для досліджень

- 1**. Знайдіть інформацію про алгоритм стиснення Лемпеля — Зіва — Велча (англ. Lempel — Ziv — Welch, LZW) і дізнайтесь, як він працює.



ПРАКТИЧНА РОБОТА №3

Архівування та розархівування даних

Завдання. Створити архіви різних видів з допомогою програми-архіватора та видобути файли з архівів.

- Створіть звичайний архів, до якого помістіть кілька файлів. Для цього в програмі **Провідник** перейдіть до папки з файлами, які потрібно заархівувати (її ім'я має вказати вчитель).
 - Виберіть ці файли, клацніть їх правою кнопкою миші та в контекстному меню виберіть команду **WinRAR** ▶ **Додати до архіву**. Відобразиться вікно **Ім'я архіву та параметри**. (Якщо вибрати команду **WinRAR** ▶ **Додати до <ім'я_файла>**, то архів буде створено в тій самій папці, де містяться вихідні файли.)
 - У вікні задайте ім'я архіву, виберіть формат **ZIP**, залиште нормальний метод стиснення. Клацніть кнопку **ОК**.
 - Створіть саморозпаковуваний архів. Для цього виконайте такі дії.
 - Мишкою виберіть ті самі файли, клацніть їх правою кнопкою миші та в контекстному меню виберіть команду **WinRAR** ▶ **Додати до архіву**, щоб відкрити вікно **Ім'я архіву та параметри**. (Якщо вибрати команду **WinRAR** ▶ **Додати до <ім'я_файла>**, то архів буде створено в тій самій папці, де містяться вихідні файли.)
 - У вікні задайте ім'я архіву, виберіть формат **RAR**, залиште нормальний метод стиснення та встановіть прапорець **Створити SFX-архів**. Клацніть кнопку **ОК**. Архів буде створено.
 - Створіть на робочому столі папку та перемістіть до неї саморозпаковуваний архів. Відкрийте цю папку і двічі клацніть файл архіву. Його вміст буде розпаковано в поточну папку.
 - Перейдіть до створеної папки і створіть у ній ще одну папку. Перемістіть до неї звичайний архів (він має розширення **.zip**). Двічі клацніть ім'я файла цього архіву. Відкриється вікно програми-архіватора.
 - З допомогою миші виберіть усі файли архіву та клацніть кнопку **Видобути до**, щоб розпакувати архів.
 - Перемістіть у створену папку файл фотографії з розширенням **.jpg**, аудіофайл з розширенням **.mp3** і файл з текстом, вказані вчителем. Створіть **RAR**-архіви для кожного з них. Порівняйте обсяги початкових файлів та архівів. Визначте ступінь стиснення, натиснувши кнопку **Інформація** у вікні архіватора. Результати запишіть у таблицю (табл. 2.3.) та поясніть їх. Завершивши роботу, видаліть усі папки, створені для розпакування архівів.

Таблиця 2.3. Порівняння ступеня стиснення файлів різних форматів

Файл	Обсяг вихідного файла	Обсяг архіву	Ступінь стиснення
jpg			
mp3			
txt			

Розділ 3

Опрацювання текстових даних



§9

Списки в текстових документах

9.1.

ПОВТОРЕННЯ

У 6-му класі ви почали вивчати текстовий редактор і навчилися створювати, редагувати, форматовувати і друкувати текстові документи. Цього року ви навчитеся створювати документи, що містять списки, таблиці та ілюстрації, а також дізнаєтесь, як працювати зі структурою документа і зробити до нього зміст. Для початку коротко повторимо основні поняття.

У документі, створеному в текстовому процесорі Microsoft Word, можна виділити такі текстові об'єкти: символ, слово, речення, абзац, рядок. Окрім того документ може містити автофігури, малюнки, таблиці, формули, діаграми, об'єкти WordArt, SmartArt та ін.

Робота над текстом насамперед передбачає його редагування — вставлення додаткових символів і видалення зайвих, виправлення помилок тощо. Для цього можна використовувати такі операції:

- видалення — виділити текст і натиснути клавіші **Delete** та **Backspace**;
- копіювання та переміщення — виділити текст і скористатися клавішами **Ctrl+C**, **Ctrl+X**, **Ctrl+V**;
- перевірка правопису — вибрати команди автоматичного пошуку та виправлення помилок;
- пошук і заміна тексту — скористатися командами **Пошук** і **Замінити**.

Щоб змінити вигляд тексту, застосовують:

- форматування символів — зміна шрифту, розміру, написання (курсивне, напівжирне), способу підкреслення, кольору, міжсимвольного інтервалу;



Пригадати

основні поняття, що пов'язані з текстовим редактором, ви зможете на диску.

- форматування абзаців — встановлення вирівнювання (за лівим краєм, за правим краєм, за шириною, по центру), міжрядкового інтервалу, інтервалів до і після абзаців, абзацного відступу.

9.2.

ФОРМАТИ ФАЙЛІВ ТЕКСТОВИХ ДОКУМЕНТІВ

Нагадаємо, що для комп'ютерного подання текстової інформації символи алфавіту кодуються двійковими числами, а текст подається у вигляді набору чисел — кодів символів, що його складають.

Файли, що створюють з допомогою текстових редакторів, окрім кодів символів, можуть містити дані про форматування тексту, а також додаткові елементи (наприклад, вставлені в текст рисунки). Уся ця інформація звичайно закодована в двійковій формі.

Щоб програма могла правильно відобразити текстовий документ, вона заздалегідь має «знати», в якому місці файлу міститься власне текст, а в якому — інші елементи. Тому, коли ви зберігаєте текстовий документ у файлі, програма має розмістити всю необхідну інформацію за певними правилами, тобто зберегти файл у певному *форматі*.

Формат файлу — опис того, які дані в якому місці файлу розташовано.



Дізнатися більше
про текстові
формати файлів ви
зможете на диску.

Щоб визначити формат файлу, достатньо подивитися на розширення його імені, тобто на частину імені, що йде після крапки. Наприклад, файли, які містять форматований текст і можуть редагуватися у текстовому редакторі зазвичай мають розширення **.doc**, **.docx**, **.rtf**, а файли, що містять лише текст без даних про форматування мають розширення **.txt**.

9.3.

ВИДИ СПИСКІВ

Марковані	Нумеровані	Багаторічкові
<ul style="list-style-type: none"> ✓ понеділок ✓ вівторок ✓ середа ✓ четвер ✓ п'ятниця ✓ субота ✓ неділя 	<ol style="list-style-type: none"> 1. понеділок 2. вівторок 3. середа 4. четвер 5. п'ятниця 6. субота 7. неділя 	<p>I. Робочі дні</p> <ul style="list-style-type: none"> • понеділок • вівторок • середа • четвер • п'ятниця <p>II. Вихідні дні</p> <ul style="list-style-type: none"> • субота • неділя

Рис. 3.1. Види списків

У тексті часто трапляються різноманітні переліки, наприклад, перелік днів тижня, складових комп'ютера, інгредієнтів страви тощо. Одним із прийомів, який дозволяє покращити сприйняття такого тексту, є оформлення його у вигляді маркованого списку.

Якщо ж йдеться про подання послідовності дій або, наприклад, пунктів договору, то в такому разі доцільно створювати нумеровані списки. Це дозволяє, у свою чергу, робити посилання на блоки тексту, вказуючи їхні номери.

Список також може бути багаторівневим, типовий зразок — зміст книжки, що містить кілька рівнів заголовків (рис. 3.1).

9.4. СТВОРЕННЯ СПИСКІВ

У програмі Word створювати списки можна під час введення тексту. Окрім цього, на список можна перетворити весь набраний текст. Розглянемо методи створення маркованих і нумерованих списків.

Створення списку під час введення тексту

1. Встановити курсор у рядок, з якого має починатися список, і натиснути кнопку для створення маркованого списку або кнопку для створення нумерованого списку (кнопки містяться на вкладці **Основне** в групі **Абзац**).
2. Щоб створити новий елемент списку, потрібно натиснути клавішу **Enter**. Кожний абзац вважатиметься окремим елементом списку, і йому надаватиметься порядковий номер або маркер.
3. Щоб завершити список, треба натиснути клавішу **Enter** два рази або один раз натиснути **Enter** і кнопку чи .

Перетворення готового тексту на список

Щоб текст перетворити на список, потрібно ввести всі елементи списку (кожний в окремому абзаці), виділити їх і натиснути кнопку чи .

Перетворення списку на текст

Фрагмент тексту, оформлений у вигляді списку, можна перетворити на звичайний текст. Для цього треба виділити список і натиснути ту з кнопок чи , яка має вигляд натиснутої.

9.5. ЗМІНА ЗОВНІШНЬОГО ВИГЛЯДУ СПИСКІВ

За умовчанням у маркованих списках застосовується маркер у вигляді кружечка, а в нумерованих — арабські цифри. За потреби ці настройки можна змінити.

Щоб змінити вигляд маркованого списку, потрібно встановити курсор у будь-яке місце списку і натиснути стрілочку на кнопці **Маркери**. Відобразиться панель, на якій можна: 1) обрати інший маркер; 2) змінити рівень списку; 3) визначити новий маркер (рис. 3.2).

Вигляд нумерованого списку можна змінити за допомогою панелі, що відкривається після натискання стрілки на кнопці **Нумерація**. У цьому випадку можна: 1) обрати вид нумерованого списку; 2) змінити рівень списку; 3) визначити новий формат номера; 4) змінити значення початкового номера (рис. 3.3).

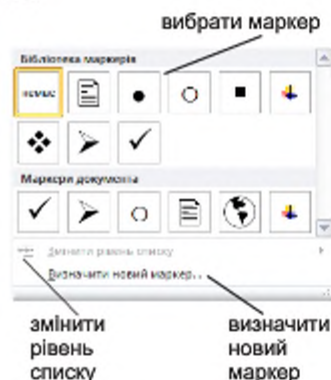
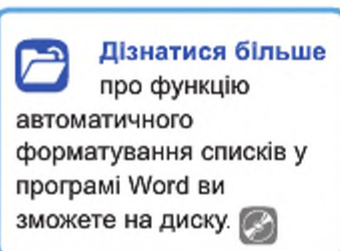


Рис. 3.2. Панель для налаштування маркованого списку

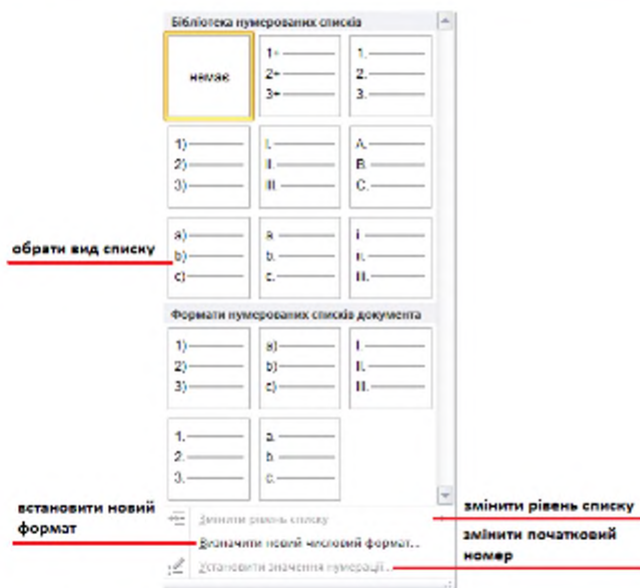


Рис. 3.3. Панель для налаштування нумерованого списку

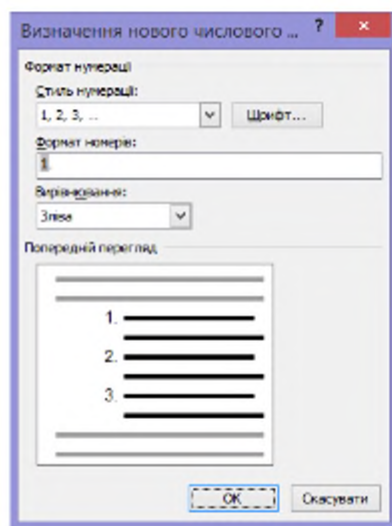


Рис. 3.4. Вікно визначення нового числового формату

Потреба визначити новий числовий формат може виникнути, коли, наприклад, програма для нумерації використовує латинські літери, а потрібно — українські. Для цього треба вибрати на панелі кнопки **Нумерація** команду **Визначити новий числовий формат...** і в діалоговому вікні, що відкриється, обрати в списку **Стиль нумерації**: літери з кирилиці. Зверніть увагу на поле **Формат номерів**: Символ, який генерується програмою, виділено в ньому сірим (рис. 3.4). Ви можете ввести в це поле

додатковий символ, і він з'явиться в нумерації. Зокрема, в такий спосіб можна додати до нумерації знак параграфа.

Щоб розбити один список на два, потрібно задати для другого списку нумерацію з початку. Для цього встановити курсор у перший елемент цього списку та на панелі кнопки **Нумерація** вибрати команду **Установити значення нумерації**. В діалоговому вікні, що відкриється, встановити перемикач **Розпочати новий список** (рис. 3.5).

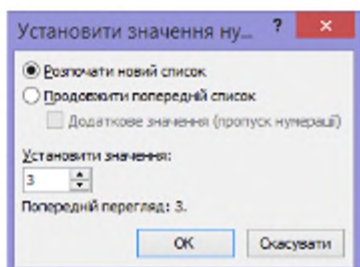



Рис. 3.5. Вікно встановлення значень нумерації

9.6.

БАГАТОРІВНЕВІ СПИСКИ

Багаторівневі списки можна створювати або під час їх набору, або з готових блоків тексту. У першому випадку перед набором тексту потрібно вибрати стиль списку в меню кнопки , а потім під час уведення



тексту встановлювати рівень елементів за допомогою кнопок (Збільшити відступ) і (Зменшити відступ) або клавіатури (для пониження рівня натискати клавішу **Tab**, а для підвищення — клавіші **Shift+Tab**).

Щоб перетворити готовий блок тексту на багаторівневий список, потрібно виконати такі дії:

1. Виділити текст і вибрати стиль списку в меню кнопки . Текст буде оформлено як однорівневий список.
2. Якщо абзац потрібно оформити як елемент другого рівня, треба встановити на ньому курсор і натиснути клавішу **Tab** або кнопку (Збільшити відступ). Щоб створити елемент третього рівня, клацнути цю кнопку ще раз.



Дізнатися більше
про багаторівневі
списки в програмі Word
ви зможете на диску.

Щоб змінити вигляд багаторівневого списку, необхідно відкрити панель кнопки **Багаторівневий список**, де можна: 1) обрати вид багаторівневого списку з бібліотеки; 2) змінити рівень списку; 3) визначити новий багаторівневий список; 4) визначити новий стиль списку.

ВПРАВА 5



Завдання. Створити марковані та нумеровані списки.


1. Введіть перелік базових кольорів веселки (червоний, оранжевий, жовтий, зелений, блакитний, синій, фіолетовий), натискаючи клавішу **Enter** після кожного слова.
2. Виділіть введені слова і натисніть кнопку . Ви отримаєте маркований список.
3. Змініть маркер списку на ✓. Для цього натисніть стрілочку на кнопці і виберіть відповідний маркер на панелі.
4. Створіть копію списку, розташувавши її нижче.
5. Виділіть нижній список і натисніть кнопку . Він стане нумерованим.
6. Встановіть для списку нумерацію римськими цифрами. Для цього натисніть стрілочку на кнопці та оберіть відповідну нумерацію на панелі.
7. Створіть нижче копію щойно створеного списку.
8. Перетворіть елементи списку на звичайний текст, щоб вийшло таке речення:

Веселка має сім кольорів: червоний, оранжевий, жовтий, зелений, блакитний, синій, фіолетовий.

Виконайте такі дії:



Виконати
вправи, щоб
навчитися створювати
різноманітні списки, ви
зможете на диску.

- а) виділіть список кольорів і натисніть кнопку . Список має перетворитися на звичайний текст, але записаний у стовпчик;
 - б) введіть перед текстом фразу: «Веселка має сім кольорів:»;
 - в) після двокрапки введіть пробіл та натисніть клавішу **Delete** (це потрібно для того, щоб текст з нижнього рядка «підняти» до щойно введеної фрази, тобто для того, щоб видалити символ кінця абзацу);
 - г) у такий же спосіб «підніміть» інші елементи списку.
9. Збережіть завдання у власній папці під назвою **Вправа_5_Прізвище**.



Висновки

Формат файла — це опис того, які дані, в якому місці файла розташовано. Поширені текстові формати: **.txt**, **.doc**, **.docx**, **.rtf**.

Форматування можна здійснювати на рівні символів і на рівні абзаців.

Для символів задають такі параметри, як шрифт, розмір, написання, спосіб підкреслення і колір.

До основних параметрів форматування абзаців належать вирівнювання, міжрядковий інтервал, інтервали до і після абзацу, відступ першого рядка.

Програма Word дає змогу створювати марковані, нумеровані та багаторівневі списки. Їх можна створювати під час набору тексту або на основі готового тексту.



Контрольні запитання та завдання

1. Перелічіть основні об'єкти, з якими працює програма Word.
2. Які способи редагування тексту ви знаєте?
3. Які параметри форматування задають на рівні символів, а які — на рівні абзаців?
4. Що таке формат файла? Які поширені формати текстових файлів ви знаєте?
5. Які є види списків? Коли кожен зокрема доцільно використовувати?
6. Опишіть порядок створення списків у програмі Word.



Питання для роздумів

1. Чому після копіювання тексту, відформатованого в програмі Word, у програму Блокнот зникає форматування символів, але маркери і нумерація списків залишаються?



Завдання для досліджень

1. Дослідіть, які параметри списків можна змінювати за допомогою контекстного меню.
2. З'ясуйте, для чого призначений формат PDF. Чи можна редагувати файли в цьому форматі в текстовому редакторі Word?
3. З'ясуйте, які формати застосовують для електронних книг і мобільних пристроїв.

§10

Таблиці в текстових документах

10.1.

ОСНОВНІ ЕЛЕМЕНТИ ТАБЛИЦІ

Подання інформації у вигляді таблиці дає змогу представити її наочно. У програмі MS Word передбачено зручні засоби для створення таблиць.

Таблиця — це об'єкт текстового документа, що складається з клітинок, упорядкованих за стовпцями та рядками. Клітинки таблиці можуть містити текст, цифри, малюнки та інші таблиці.

Таблиця також має маркери переміщення та зміни розмірів (рис. 3.6), з допомогою яких її можна переміщувати або змінювати розміри. Клітинки таблиці відділяються межами. За умовчанням це суцільні лінії чорного кольору, але загалом їхній вигляд може бути різним. Зокрема, межа може бути невидимою; тоді на екрані вона відображується як тонка сіра лінія і на друк не виводиться.



Рис. 3.6. Складові таблиці

10.2.

СТВОРЕННЯ ТАБЛИЦЬ

У документі Word таблицю можна створити кількома способами: вставити порожню таблицю, попередньо задавши для неї кількість рядків і стовпців; намалювати таблицю подібно до того, як це роблять олівцем на папері; перетворити на таблицю частину тексту.

Вставлення таблиці

Цьому способу слід віддавати перевагу у випадках, коли заздалегідь відомо, яку кількість стовпців і рядків має містити таблиця.

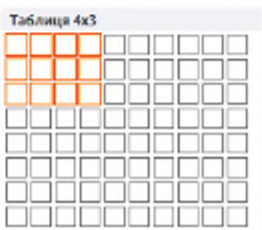



Рис. 3.7. Панель швидкого додавання таблиці



1. Встановити курсор у рядок, де має бути таблиця, перейти на вкладку **Вставлення** і натиснути кнопку  (**Таблиця**).

2. Виділити на макеті таблиці (рис. 3.7) потрібну кількість рядків і стовпців та клацнути лівою кнопкою миші.

Великі таблиці краще створювати, вибравши на панелі команду **Вставити таблицю**, після чого відкривається діалогове вікно, в якому можна задати для таблиці кількість рядків і стовпців.

Креслення таблиці

Якщо необхідно створити таблицю складної структури, доцільно скористатися способом креслення.

1. Встановити курсор у рядок, де має бути таблиця, натиснути кнопку  (**Таблиця**) і обрати на її панелі команду **Накреслити таблицю**. Вказівник миші набуде вигляду олівця .


2. Окреслити ділянку, яку займатиме таблиця. Для цього розташувати вказівник миші в одному її куті, натиснути ліву клавішу миші та перемістити вказівник у протилежний по діагоналі кут ділянки (рис. 3.8).




Рис. 3.8. Побудова таблиці методом креслення


№	Найменування	Ціна, грн
1.	Приппер	1200
2.	Сканер	800

Рис. 3.9. Текст з розділювачами

3. Намалювати межі всередині таблиці. Щоб видалити межу, потрібно вибрати інструмент  (**Гумка**) на вкладці **Конструктор** і клацнути на цій межі.

Перетворення блока тексту на таблицю

У вигляді таблиці можна оформити набраний в документі фрагмент тексту. Щоб програма могла це зробити автоматично, текст має містити символи, які використовуються лише як розділювачі. Зазвичай це символи табуляції та кінця рядка (рис. 3.9), які можна побачити, ввімкнувши режим недрукованих символів, натиснувши інструмент  (вигляд кнопки недрук. символів) на вкладці **Основне** групи **Абзац**. Перші з них позначатимуть місця поділу тексту на стовпці, а другі — місця поділу на рядки.

1. Підготувати блок тексту, як описано вище, і виділити його. В меню кнопки  (**Таблиця**) вибрати команду **Перетворити на таблицю**, щоб відкривалося однойменне діалогове вікно (рис. 3.10).

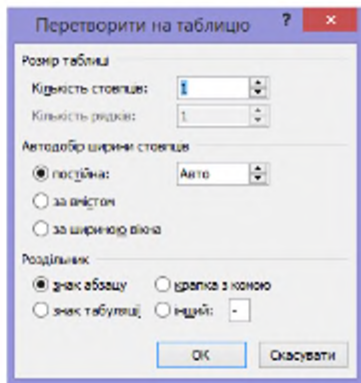


Рис. 3.10. Вікно Перетворити на таблицю



- Вказати, який символ використовується в тексті як розділювач стовпців, і клацнути **ОК**. Таблицю буде створено.

За потреби таблицю можна перетворити на блок тексту. Для цього потрібно встановити курсор у таблицю і на вкладці **Макет** натиснути кнопку **Перетворити на текст**. У діалоговому вікні, що відкриється, вибрати символ-розділювач між текстом клітинок і натиснути **ОК**.

10.3.

ВИДІЛЕННЯ ЕЛЕМЕНТІВ ТАБЛИЦІ

Вам уже відомо, що перш ніж виконувати над об'єктами певні дії, їх потрібно виділити. Це стосується і багатьох операцій з таблицями.

Виділення стовця або групи стовпців. Підвести вказівник миші до верхньої межі таблиці, щоб він набув форми маленької спрямованої вниз стрілки (↓), клацнути мишею, і програма виділить стовець, у якому розташовано вказівник. Якщо протягнути вказівник ліворуч або праворуч, утримуючи ліву клавішу миші, буде виділено декілька стовпців.

Виділення рядка або групи рядків. Для виділення рядка розмістити ліворуч від нього вказівник миші, щоб той набув вигляду контурної стрілки, нахиленої праворуч (↗), і клацнути мишею. Якщо протягнути вказівник униз чи вгору, утримуючи ліву клавішу миші, буде виділено кілька рядків.

Виділення клітинки або групи клітинок. Підвести вказівник до лівої межі клітинки, щоб він набув вигляду маленької нахиленої праворуч стрілки (↖), натиснути ліву кнопку миші. Для виділення групи клітинок, не відпускаючи ліву кнопку миші, протягнути вказівник ліворуч, праворуч, вгору або вниз. Утримуючи клавішу **Ctrl**, можна виділяти несуміжні клітинки, клацаючи по них курсором миші.

Виділення всієї таблиці. Клацнути на маркері переміщення таблиці (+).

10.4.

ВВЕДЕННЯ ТА РЕДАГУВАННЯ ТЕКСТУ

Щоб увести текст у клітинку, потрібно клацнути в ній, а потім набрати текст на клавіатурі або вставити з буфера обміну. Для переходу до наступної клітинки (праворуч) треба натиснути клавішу **Tab** чи →, або просто перемістити курсор за допомогою миші, клацнувши у відповідній клітинці. Для переміщення на одну клітинку ліворуч необхідно натиснути клавіші **Shift+Tab** чи ←.

Якщо розташований в клітинці текст потрібно перемістити, скопіювати або видалити, необхідно виділити його за допомогою миші або клавіш керування курсором і натиснути клавішу **Delete** або **Backspace**, і текст буде видалено. Щоб скопіювати або перемістити текст, можна скористатися командами роботи з буфером обміну або перетягнути його, використовуючи мишку.

За потреби ввести текст, який має передувати таблиці, розташованій на початку документа, курсор слід встановити на початок першого рядка таблиці та натиснути клавішу **Enter**.

10.5.

ФОРМАТУВАННЯ ТАБЛИЦЬ

Дані в таблиці та й саму таблицю можна формувати — змінювати параметри тексту, оформляти межі, замальовувати внутрішні ділянки клітинок та ін. Майже всі ці операції виконують на вкладках **Макет** і **Конструктор**, які з'являються в переліку вкладок, якщо встановити курсор у будь-яке місце таблиці.

Форматування тексту

Форматування тексту в клітинках виконується так само, як і форматування звичайного тексту. Для таблиці доступні також усі можливості форматування абзаців, а в групі **Вирівнювання** на вкладці **Макет** містяться інструменти, призначені для вирівнювання тексту відносно меж клітинки, зміни напрямку тексту і відступів тексту від меж клітинки (рис. 3.11).

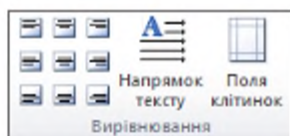


Рис. 3.11

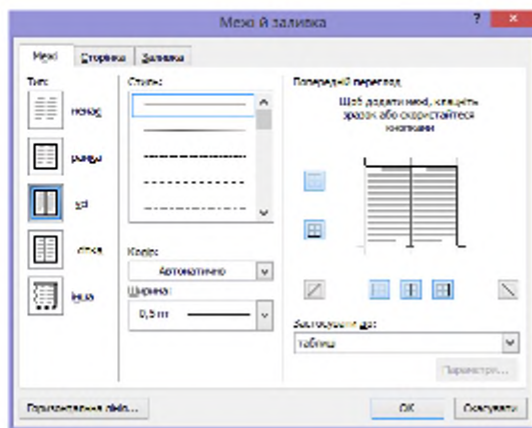


Рис. 3.12. Вікно Межі й заливка

Зміна параметрів меж і заливки клітинок

На вкладці **Конструктор** розташовано інструменти, призначені для вибору стилю, товщини і кольору меж таблиці. Коли натиснути стрілочку біля кнопки **Межі** (Межі), на екрані відкривається панель для вибору відображення меж. На цій панелі є команда **Межі й заливка**, яка відображає вікно для налаштування всіх параметрів форматування клітинок таблиці (рис. 3.12). Кнопка **Межі** також продубльована на вкладці **Основне** в групі **Абзац**.

Колір клітинки можна змінити також у вкладці **Конструктор**, обравши інструмент **Затінення** (Затінення).

У групі **Стилі таблиць** на вкладці **Конструктор** містяться вбудовані зразки форматування таблиць, які можуть задовольнити потреби у форматуванні таблиць навіть найвибагливіших користувачів. Пересуваючись курсором по представлених зразках, таблиця автоматично змінює свій вигляд. Щоб застосувати один із запропонованих стилів, достатньо клацнути на одному із зразків.

10.6.


ВСТАВЛЕННЯ ТА ВИДАЛЕННЯ РЯДКІВ І СТОВПЦІВ


Встановити курсор у клітинку таблиці, біля якої потрібно додати рядок (стовпець), перейти на вкладку **Макет** і натиснути кнопку **Вставити**




(Вставити




зверху) чи  (Вставити знизу) — добавиться рядок; натиснути кнопку  (Вставити зліва) чи  (Вставити справа) — добавиться стовпець.



Можна добавляти рядки і стовпці по-іншому. Щоб додати рядок, потрібно встановити курсор перед маркером кінця рядка, після якого має бути розташований новий рядок, і натиснути клавішу **Enter**. Якщо маркери таблиці не відображаються, треба клацнути на кнопку  (**Недруковані знаки**). Для того, щоб додати рядок у кінець таблиці, необхідно перейти в останню клітинку і натиснути клавішу **Tab**.

Щоб видалити рядок (стовпець) з таблиці, треба встановити в ньому курсор і натиснути кнопку  (**Видалити**), обравши елемент, який необхідно видалити.

10.7.

ЗМІНЕННЯ ШИРИНИ СТОВПЦІВ І ВИСОТИ РЯДКІВ

Щоб змінити межі клітинки, потрібно встановити вказівник миші на межу і, коли він набуде вигляду двонапрямленої стрілки , перетягнути межу в потрібне місце.

Якщо необхідно, щоб усі стовпці (рядки) мали однакову ширину чи висоту, треба виділити їх і натиснути кнопку  **Вирівняти ширину стовпців** (**Вирівняти ширину стовпців**) або  **Вирівняти висоту рядків** (**Вирівняти висоту рядків**).

Зручним способом вирівнювання є також використання функції  (**Автодобр**), яка надає можливість встановити розміри стовпців залежно від обсягу тексту (команда **Автодобр за вмістом**), розтягнути таблицю по ширині вікна (команда **Автодобр за шириною вікна**) або встановити фіксовані ширини стовпців (команда **Фіксована ширина стовпця**).

10.8.

ОБ'ЄДНАННЯ ТА ПОДІЛ КЛІТИНОК

Під час побудови таблиць нерідко виникає потреба об'єднати групу клітинок. Приміром, коли необхідно створити заголовок, який описуватиме всю таблицю чи кілька стовпців, або, навпаки, поділити одну клітинку на кілька (рис. 3.13).

Предмети	Бали				
	За семестр	Річні	Навчальних екскурсій	Державної підсумкової атестації	Підсумкові
Українська мова	I	II			
Математика					
...					

Рис. 3.13. Приклад таблиці з об'єднаними клітинками

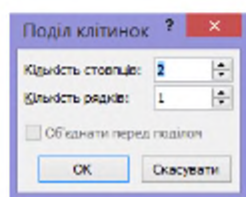


Рис. 3.14. Вікно Поділ клітинок

Щоб об'єднати клітинки, потрібно виділити їх і на вкладці **Макет** натиснути кнопку **Об'єднати клітинки** (**Об'єднати клітинки**). Інший спосіб: на вкладці **Конструктор** обрати інструмент (**Гумка**) і клацнути на межі, яку слід забрати.

Для того, щоб поділити клітинку, треба встановити в ній курсор і на вкладці **Макет** натиснути кнопку **Розділити клітинки** (**Розділити клітинки**). У діалоговому вікні, що відкриється (рис. 3.14), задати, на скільки стовпців та (або) рядків потрібно розділити клітинку.

ВПРАВА 6

	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	10	12	14		
3	3	6	9	12					
4		8							
5		10			25				
6									
7		14							
8		16							
9		18							

Рис. 3.15



Завдання. Побудувати таблицю за зразком на рис. 3.15 та заповнити її.

На вкладці **Вставлення** натисніть кнопку **Таблиця** і в її меню оберіть команду **Вставити таблицю**. У вікні, що відкриється, встановіть кількість стовпців — 10, кількість рядків — 11, автопідбір ширини стовпців — за вмістом. Натисніть **ОК**. Заповніть таблицю даними. Задайте заливку клітинок за зразком.

Збережіть файл під іменем **Вправа_6_Прізвище**.

ВПРАВА 7



Завдання. Побудувати таблицю за зразком і заповнити її.

№	Найменування	Характеристики		
		Розмір	Колір	Ціна, грн
1	Кросівки	36	Чорний	850
2	Чоботи	39	Червоний	1200
3	Черевики	38	Сірий	640

Створіть таблицю 5×5. Об'єднайте першу і другу клітинки першого стовпця. Для цього виділіть їх та оберіть інструмент **Об'єднати клітинки** (**Об'єднати клітинки**) на вкладці **Макет**. Аналогічно об'єднайте першу і другу клітинки другого стовпця та останні три клітинки першого рядка. Заповніть таблицю даними. Задайте розміщення тексту в клітинках за зразком, використовуючи інструменти на вкладці **Макет**.

Збережіть файл під іменем **Вправа_7_Прізвище**.



Виконати

вправи, щоб навчитися створювати таблиці, ви зможете на диску.

**Висновки**

Створити таблицю в документі Word можна трьома способами: вставити порожню таблицю; намалювати її; перетворити на таблицю фрагмент тексту.

Для форматування тексту в клітинках слід використовувати команди **Шрифт** та **Абзац** на вкладці **Основне** і **Вирівнювання** на вкладці **Макет**.

Щоб швидко відформатувати таблицю за зразком, можна скористатися вбудованими стилями на вкладці **Конструктор**.

Під час редагування таблиці можна додавати та видаляти її рядки і стовпці, змінювати їхню ширину і висоту, об'єднувати та ділити клітинки. Всі ці дії можна виконати за допомогою інструментів, розташованих на вкладці **Макет**.

**Контрольні запитання та завдання**

1. Назвіть методи створення таблиць у документі Word.
2. Які є можливості для форматування тексту таблиці?
3. Які методи виділення елементів таблиці ви знаєте?
4. Як розфарбувати таблицю?
5. Як додавати або видаляти рядки і стовпці таблиці? Опишіть способи з використанням клавіатури.
6. Як виконати об'єднання і поділ клітинок таблиці?

**Питання для роздумів**

1. Наведіть приклади використання функції розділення клітинки.

**Завдання для досліджень**

- 1*. Дослідіть, як у таблиці Word можна виконати обчислення суми числових значень у рядку або стовпці.
- 2*. З'ясуйте, які види таблиць можна додати командою **Експрес-таблиці** в меню інструмента **Таблиці**.
- 3*. Знайдіть спосіб поділу таблиці на дві.

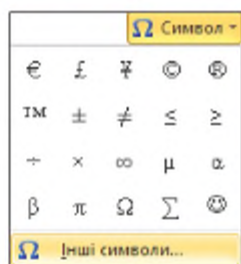
§11
Символи, формули, графічні об'єкти в текстовому документі
11.1.
ВСТАВЛЕННЯ СИМВОЛІВ


Рис. 3.16. Вікно найбільш уживаних символів

Крім тексту, набраного з клавіатури, в документ Word можна вставити різноманітні символи, знаки і навіть формули.

Щоб вставити в текст символ, відсутній на клавіатурі, потрібно перейти на вкладку **Вставлення** і натиснути кнопку (**Символ**). Відкриється панель (рис. 3.16), на якій можна обрати символ за допомогою миші.

Якщо потрібного символу на панелі немає, треба вибрати команду **Інші символи**. У вікні **Символ** (рис. 3.17) обравши шрифт і набір у відповідних полях, можна знайти необхідний символ, клацнути на ньому і натиснути кнопки **Вставити** та **Закрити**.



Дізнатися більше,
як вставляти символи
з допомогою призначених
комбінацій клавіш,
ви зможете на диску.

Розглянемо детальніше вкладку **Спеціальні символи** (рис. 3.18) вікна **Символ**, на якій вміщено перелік символів і комбінацій клавіш для їх введення, що використовуються в тексті. Серед них є символ абзацу та кілька видів пробілів, які вважаються недрукованими. Для їх відображення на екрані достатньо натиснути кнопку на вкладці

Основне. Щоб забрати недруковані символи з екрана, треба ще раз натиснути цю кнопку.

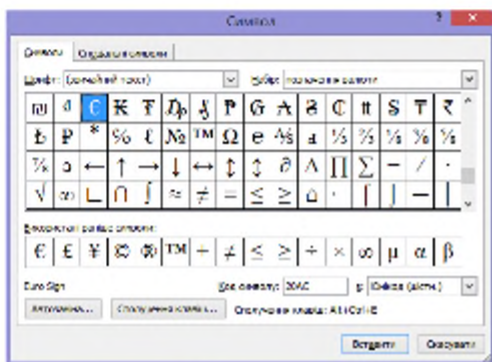


Рис. 3.17. Вікно **Символ**

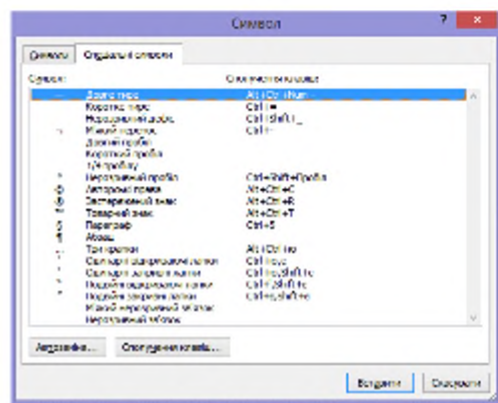


Рис. 3.18. Вкладка **Спеціальні символи** вікна **Символ**

ВПРАВА 8
САМОСТІЙНО


Завдання. Створити список знаків зодіаку, скориставшись шрифтом **Wingdings**: ♈ — Овен, ♉ — Телець, ♊ — Близнюки, ♋ — Рак, ♌ — Лев, ♍ — Діва, ♎ — Терези, ♏ — Скорпіон, ♐ — Стрілець, ♑ — Козеріг, ♒ — Водолій, ♓ — Риби.



11.2.

ВСТАВЛЕННЯ ФОРМУЛ

Нескладні математичні, фізичні та хімічні формули можна вводити, вставляючи спеціальні символи описаним вище способом. Щоб увести складні формули, потрібно натиснути кнопку **Π Рівняння** (Рівняння), що міститься на вкладці **Вставлення**.

Відкривши панель цієї кнопки, ви побачите шаблони поширених формул, які можна вставляти в документ одним клацанням миші. Проте в більшості випадків потрібно вводити формулу «з нуля». У такому разі треба клацнути зазначену кнопку, після чого на екрані відобразиться вкладка **Знаряддя для формул** (рис. 3.19), а в місці розташування текстового курсора з'явиться поле для введення формули: **Введіть тут формулу!,-**. Щоб робити далі, ви дізнаєтесь, виконавши наступну вправу.



Рис. 3.19. Вкладка **Знаряддя для формул**

ВПРАВА 9



Завдання. Увести такі формули:

а) $\rho = \frac{m}{v}$; б) $\left(\frac{a}{b}\right)^n = \frac{a^n}{b^n}$; в) $t = \frac{S_1+S_2}{v_1-v_2}$;

г) $(ab)^n = \underbrace{ab \cdot ab \cdot \dots \cdot ab}_{n \text{ разів}} = \underbrace{aa \dots a}_{n \text{ разів}} \cdot \underbrace{bb \dots b}_{n \text{ разів}} = a^n \cdot b^n$.

1. Перейдіть на вкладку **Вставлення** і натисніть кнопку **Π Рівняння** (Рівняння).
2. У групі **Символи** натисніть кнопку **Додатково** й у списку вгорі панелі, що відкривається, виберіть **Грецькі літери** (рис. 3.20).



Рис. 3.20. Вибір групи символів для формул

3. Клацніть символ ρ та введіть після нього знак рівності з клавіатури.

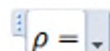



Рис. 3.21. Вибір шаблону дроби

Рис. 3.22. Вибір шаблону індексу

4. Натисніть кнопку **Дріб** і виберіть перший шаблон (рис. 3.21).
5. У чисельнику введіть літеру m , у знаменнику – v . Для завершення введення формули клацніть поза її межами.
6. Для введення другої формули виконайте п. 1.
7. Натисніть кнопку **Індекс** і виберіть перший шаблон (рис. 3.22).
8. Виділіть у шаблоні більший прямокутник, натисніть кнопку **Дужка** та оберіть перший шаблон. У результаті ви отримаєте такий вигляд майбутньої формули: $(a^b)^c$.
9. Виділіть у цьому шаблоні прямокутник всередині дужок, натисніть кнопку **Дріб** і оберіть перший шаблон. Ви одержите такий вигляд майбутньої формули: $(\frac{a}{b})^c$.
10. У чисельнику введіть літеру a , у знаменнику — b , а в показнику степе-
ня — n .
11. Встановіть курсор у крайнє положення праворуч та продовжте введення
формули самостійно.
12. Формули $v)$ і $г)^*$ введіть самостійно.
13. Збережіть файл під іменем **Вправа_9_Прізвище**.

11.3.

ВСТАВЛЕННЯ ГРАФІЧНИХ ОБ'ЄКТІВ

У документ можна додавати ілюстративний матеріал, зокрема малюнки чи світлини з файлів, автофігури, об'єкти WordArt, SmartArt тощо. Способи додавання та редагування цих об'єктів ви розглядали в 5 і 6 класах, вивчаючи редактор презентацій і текстовий редактор. Нагадаємо, що для вставлення графічних об'єктів використовують групу **Зображення** вкладки **Вставлення**, а для їх форматування — відповідну вкладку, що з'являється в меню після вибору такого об'єкта.

ВПРАВА 10



Завдання. Створити малюнок, як на рис. 3.23.

1. Спочатку намалюйте будинок.
 - а) Побудуйте прямокутник, скориставшись кнопкою  (**Фігури**) на вкладці **Вставлення**.



Рис. 3.23. Малюнок, створений засобами програми Word

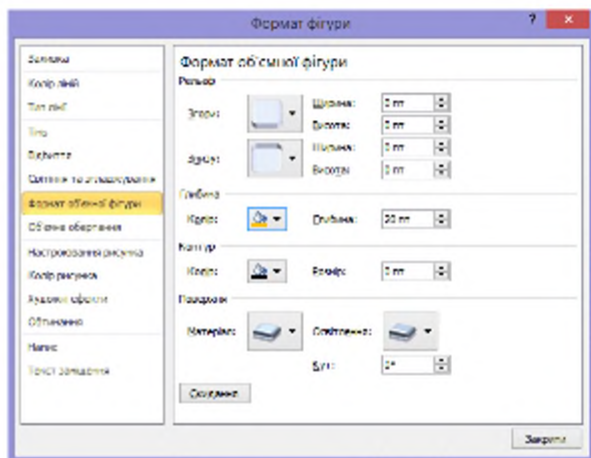


Рис. 3.24

б) Зафарбуйте його темно-жовтим кольором. Для цього виділіть прямокутник і на вкладці **Формат** натисніть кнопку **Заливка фігури** й оберіть колір.

в) Застосуйте до прямокутника ефект об'єму:

- відкрийте діалогове вікно групи **Стилі фігур** на вкладці **Формат**;
- на панелі ліворуч виберіть пункт **Формат об'ємної фігури** (рис. 3.24);
- у розділі **Глибина** встановіть потрібний колір у полі **Колір** і значення 20 пт у полі **Глибина**;
- на панелі ліворуч виберіть пункт **Об'ємне обертання**;
- у списку **Заготовки** виберіть **Нахил праворуч угору**.

г) Аналогічно побудуйте зображення даху (для точного розташування даху над будинком пересувайте його за допомогою клавіш керування курсором, утримуючи натиснутою клавішу **Ctrl**).

г) Створіть зображення вікна, використовуючи інструменти **Прямокутник** і **Лінія**.

д) Створіть зображення балкона, використовуючи інструмент **Прямокутник** та ефект об'єму.

е) Згрупуйте об'єкти вікна і балкона. Для цього виділіть їх, утримуючи клавішу **Ctrl**, і на вкладці **Формат** виберіть команду **Групувати** ► **Групувати**.

є) Створіть потрібну кількість копій вікна. Для цього перетягуйте його в потрібне місце, утримуючи клавішу **Ctrl**.

ж) Згрупуйте всі елементи будинку і створіть його копію.

2. Намалюйте паркан.

а) Побудуйте прямокутник і трикутник, зафарбуйте їх.

- для заливки дощечки паркана на вкладці **Формат** натисніть кнопку **Заливка фігури** ► **Гرادієнт** ► **Інші градієнти**;
- у вікні **Формат фігури** виберіть перемикач **Градiєнтна заливка**;

- встановіть параметри напрямку, кута та кольору точок градієнта, як на рис. 3.25, і закрийте вікно.

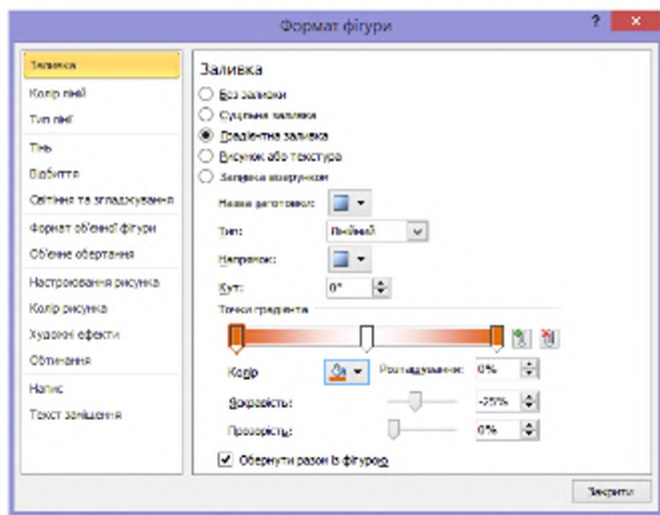


Рис. 3.25. Встановлення параметрів заливки у вікні **Формат фігури**

- Розташуйте трикутник над прямокутником і згрупуйте їх.
 - Скопіюйте фігури потрібну кількість разів.
 - Розташуйте паркан між будинками.
- Створіть зображення дерев самостійно, використовуючи об'єкти прямокутник та овал.
 - Намалюйте автомобіль.
 - Виконайте зображення передньої сторони машини, використовуючи інструмент (**Полілінія**) з групи **Фігури**.
 - Зафарбуйте зображення та застосуйте до нього ефект об'єму.
 - Вікна машини створіть з допомогою інструмента **Полілінія**.
 - Колесо намалюйте, використовуючи інструмент **Овал** і застосувавши до нього ефект об'єму.



Рис. 3.26

г) Розташуйте об'ємне зображення колеса за зображенням машини, вибравши в контекстному меню об'єкта колеса команду **На задній план**.

д) Поверх машини помістіть звичайне зображення овалу того ж кольору та розміру, що й об'ємне колесо (рис. 3.26).

- Дорогу та світлофор намалюйте самостійно.
- Збережіть файл під іменем **Вправа_10_Прізвище**.



Висновки

Кнопки та команди для вставлення в текст символів, формул і графічних об'єктів містяться на вкладці **Вставлення**.

Щоб увести в текст відсутні на клавіатурі символи, використовують кнопку **Символ** (Символ).

Щоб увести складні формули, потрібно натиснути кнопку **Рівняння** (Рівняння).

Якщо потрібно намалювати в документі фігуру, стрілку чи інший подібний об'єкт, його створюють з допомогою інструментів і команд з групи **Зображення** на вкладці **Вставлення**.



Контрольні запитання та завдання

1. Як вставити в текст символ, якого немає на клавіатурі?
2. Які символи належать до недрукованих? Як їх відобразити на екрані?
3. Опишіть загальний порядок введення формул у документ.
4. Як додати графічний об'єкт до документа?
5. Як виконати форматування графічного об'єкта?
6. Які параметри форматування можна задавати для графічних об'єктів?



Питання для роздумів

- 1*. Під час виконання яких операцій необхідно відображати недруковані символи?
- 2**. Якщо скопіювати формулу $\left(\frac{a}{b}\right)^n = \frac{a^n}{b^n}$ у документі в програмі Word і вставити в документ програми Блокнот, вона набуде такого вигляду: $(a/b)^n = a^n/b^n$. Чому так відбувається?



Завдання для досліджень

- 1*. Який інструмент потрібно використати, щоб записати таку систему рівнянь:

$$\begin{cases} 2x + 4y = 5 \\ 3x - 4y = 8 \end{cases} ?$$

2*. Як зафарбувати фігуру зображенням з файла?



3**. Дослідіть можливості команди **Знімок екрана**, розташованої на вкладці **Вставлення**.

ПРАКТИЧНА РОБОТА №4

Створення текстового документа, що містить об'єкти різних типів

Завдання 1. Записати текст і побудувати рисунок до геометричної задачі. Відформатувати його за зразком (рис. 3.27).

Задача. Доведіть, що в трикутнику ABC

$$\angle AOB = 90^\circ + \frac{\angle C}{2}, \text{ де } O \text{ — центр.}$$

Доведення.

$$\angle AOB = 180^\circ - \left(\frac{\angle A}{2} + \frac{\angle B}{2} \right) = 180^\circ - \frac{180^\circ - \angle C}{2} = 90^\circ + \frac{\angle C}{2}.$$

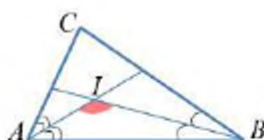


Рис. 3.27

Завдання 2. Намалювати ілюстрацію, що демонструє фізичний дослід (рис. 3.28).

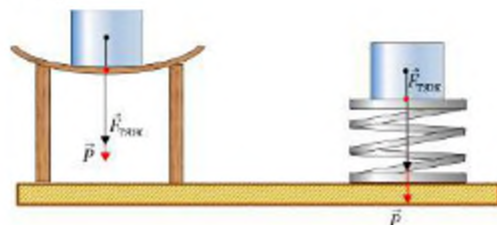


Рис. 3.28

§12

Використання стилів і шаблонів документів

12.1.

СТИЛІ ФОРМАТУВАННЯ

Вивчаючи засоби і методи форматування тексту, ви користувалися панеллю **Форматування** (Microsoft Word 2003) або групами **Шрифт** і **Абзац** на вкладці **Основне** (Microsoft Word 2007, 2010). Параметри форматування при цьому призначалися окремо для кожного абзацу та фра-



мента тексту. Цей метод називається *безпосереднім форматуванням*. Основний його недолік очевидний — для зміни форматування потрібно кожного разу вручну задавати параметри об'єкта. Уникнути цього можна в разі форматування з використанням стилів.

Стиль — це іменовані набір параметрів форматування, які можна застосувати до об'єкта певного типу.

Якщо певному об'єктові документа призначати стиль, то його зовнішній вигляд стане таким, як визначено в цьому стилі. У разі зміни значень заданих у стилі параметрів зовнішній вигляд об'єкта автоматично зміниться. Те саме станеться, якщо йому призначати інший стиль. Усі об'єкти, відформатовані з використанням одного стилю, матимуть однаковий вигляд.

Завдяки таким перевагам, як швидкість застосування, можливість забезпечити однотипне оформлення та в разі потреби швидко змінити, форматування об'єктів з використанням стилів широко застосовується в програмах роботи з текстом, графічних редакторах і багатьох інших прикладних програмах.

12.2.

ТИПИ СТИЛІВ

Оскільки кожний вид об'єктів має свій набір параметрів, для їх зберігання потрібні стилі різних типів. Зокрема, в програмі Word їх чотири:

- стиль знака — визначає лише параметри символів (шрифт, розмір, написання тощо);
- стиль абзацу — визначає параметри абзацу і параметри шрифту, що застосовуються до всього абзацу;
- стиль таблиці — зберігає інформацію про оформлення меж таблиці, кольори та різні параметри тексту, що в ній зберігається, про заливку клітинок, вирівнювання тексту в них;
- стиль списку — містить відомості про форматування маркованого або нумерованого списку.

Стилі зберігаються у документах і шаблонах (файли, на основі яких створюються документи).

12.3.

ПЕРЕГЛЯД ІНФОРМАЦІЇ ПРО СТИЛІ

Переглядати та застосовувати стилі, а також керувати ними можна за допомогою панелі стилів, яка відображається справа після клацання значка відкриття діалогового вікна в розділі **Стилі** на вкладці **Основне** (рис. 3.29).

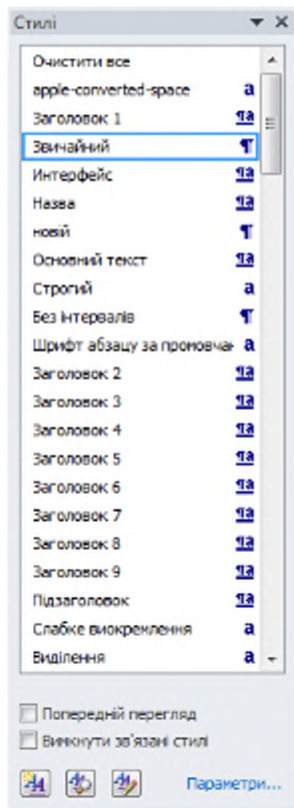


Рис. 3.29

Якщо в нижній частині списку встановити прапорець **Попередній перегляд**, назви стилів у переліку відобразатимуться у відформатованому вигляді. Стиль поточного абзацу або фрагмента виділено на панелі рамкою. Якщо навести на назву стилю вказівник миші, біля нього з'явиться стрілочка, щоб перейти в меню, яке містить команди для роботи зі стилем (зміна, видалення, виділення в тексті об'єктів з даним стилем тощо).

12.4.

ЗАСТОСУВАННЯ СТИЛІВ

Форматувати текст за допомогою стилів дуже просто: виділіть фрагмент тексту або встановіть курсор у потрібний абзац чи слово, відкрийте вікно **Стилі** та виберіть у ньому стиль символу чи абзацу. Вигляд тексту зміниться — він відповідатиме заданим у стилі параметрам.

Використовуючи стильове оформлення, слід враховувати таке: якщо до фрагмента застосовано і стиль знака, і стиль абзацу, то перевагу має перший з них.

Щодо параметрів, які визначають написання, то в цьому випадку діє таке правило: якщо в обох стилях параметр включено, то в тексті його буде виключено.

Поверх будь-якого стилю можна виконувати і безпосереднє форматування.


Якщо до тексту застосовано безпосереднє форматування, то перед використанням стилів бажано його позбутися, оскільки воно має перевагу. Наприклад, ви не зможете отримати однакові на вигляд заголовки після їх форматування за допомогою стилів, якщо заголовки вже виділені шляхом безпосереднього форматування. Щоб позбутися зайвого форматування, потрібно виділити всі його входження та натиснути **Ctrl+пробіл**.

12.5.

СТВОРЕННЯ І ЗМІНА СТИЛІВ



Дізнатися більше про правила стильового оформлення документів різних типів ви зможете на диску.

У разі потреби для оформлення документів ви можете створювати власні стилі. Щоб створити стиль абзацу, треба встановити в ньому курсор і на панелі **Стилі** натиснути кнопку  (**Створити стиль**). На екрані відобразиться вікно, як на рис. 3.30.

У ньому потрібно задати ім'я і тип стилю, вибрати параметри форматування шрифту й абзацу в розділі **Форматування** та натиснути кнопку **ОК**. Новий стиль з'явиться в переліку стилів, і його можна буде застосовувати.

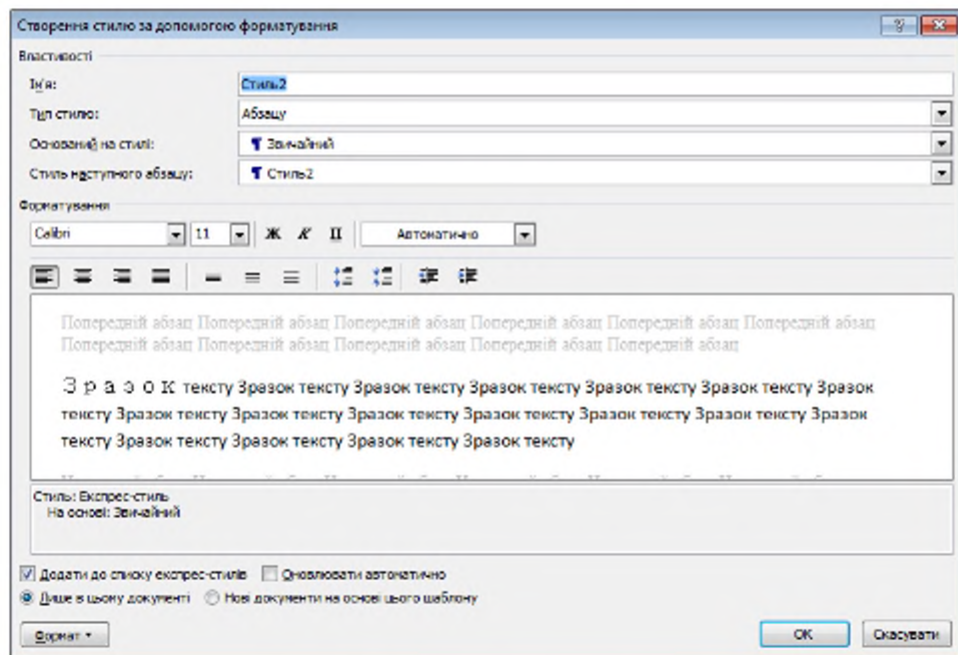


Рис. 3.30. Вікно Створення стилю за допомогою форматування

Змінюють параметри стилю аналогічно. На панелі **Стилі** треба відкрити меню стилю, вибрати команду **Змінити**, і з'явиться таке саме вікно, як для створення стилю.

Щоб отримати доступ до всіх параметрів, натиснути кнопку **Формат** і вибрати потрібну команду в меню цієї кнопки (рис. 3.31).

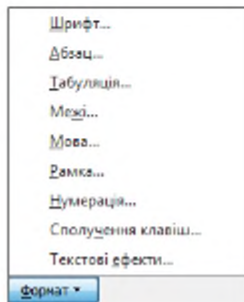


Рис. 3.31

ВПРАВА 11



Завдання. Створити власні стилі та застосувати їх до текстових фрагментів.

1. Відкрийте файл **Грози в цифрах.doc**, що знаходиться в папці **Files** на диску.
2. Створіть окремі стилі для заголовка та числових даних. Для заголовка: ім'я стилю — **Основний заголовок**, тип — **Абзац**, шрифт — **Times New Roman**, розмір — **14**, напівжирний. Для числових даних: ім'я стилю — **Число**, тип — **Символ**, шрифт — **Consolas**, розмір — **11**, зелений, напівжирний, курсив. Застосуйте ці стилі.
3. Змініть стиль підкресленого тексту, задавши для нього такі параметри: шрифт — **Arial**, колір — синій, напівжирний, курсив. Для цього клацніть на будь-якому підкресленому слові — на панелі стилів активним стане стиль

виділений текст. Клацніть на стрілочці праворуч біля стилю, виберіть команду **Змінити** та змініть параметри стилю у вікні, що відкриється.

- Аналогічно змініть стиль тексту, виділеного червоним кольором (цей стиль у даному документі має назву **Цікаві факти**). Задайте для нього такі параметри: шрифт — **Times New Roman**, колір — **коричневий**, розмір шрифту — **12**, напівжирний, курсив.
- Збережіть документ під назвою **Вправа_11_Прізвище**.

12.6.

ШАБЛони ДОКУМЕНТІВ

Як зберегти стилі, щоб використовувати їх для подальшої роботи з іншими документами? Як повторно використати набір стилів, призначений для оформлення документів певного виду?

Для вирішення цих та інших завдань у програмі Word створено механізм шаблонів, про який йтиметься далі.

Шаблон документа — це файл з розширенням **.dot** або **.dotx**, у якому зберігаються набір стилів, настройки інтерфейсу користувача програми та призначення комбінацій клавіш.

Шаблон може містити текст і графіку, які потрібно вставити в усі створені на його основі документи.

Будь-який документ Word базується на певному шаблоні. Під час створення документа в нього копіюються стилі абзаців, знаків, таблиць і списків з шаблону. Параметри цих стилів визначають початковий вигляд тексту документа.




Дізнатися більше
про копіювання стилів ви зможете на диску.



За умовчанням програма використовує шаблон **Normal.dotm**. Щоб застосувати до документа інший шаблон, його потрібно вибрати на етапі створення документа або приєднати згодом.

12.7.

СТВОРЕННЯ ДОКУМЕНТА НА ОСНОВІ ШАБЛОНУ

На вкладці **Файл** вибрати спочатку команду **Створити**, а потім — відповідний шаблон. Якщо це буде шаблон  (**Новий документ**), створюваний документ базуватиметься на шаблоні **Normal**.



Дізнатися більше
про шаблони документів ви зможете на диску.



Колекцію вбудованих шаблонів можна поповнити, звернувшись на сайт **Office.com**. Для цього в меню **Шаблони** треба обрати відповідні групи шаблонів і завантажити потрібні шаблони з сайту.



12.8.

СТВОРЕННЯ ШАБЛОНУ

На додаток до шаблонів, які пропонує програма Word, можна створювати власні шаблони. Це можуть бути, наприклад, шаблони для оформлення лабораторних робіт, рефератів, інформаційних листків. Це прискорює підготовку документів та усуває потребу кожного разу налаштовувати параметри форматування стилів.

Загальний підхід до створення і використання власних шаблонів такий: створити документ, виконати в ньому всі необхідні налаштування, зберегти документ як шаблон у форматі **.dotx** і закрити. Якщо відкрити файл шаблону, то програма Word створить документ на основі цього шаблону.

ВПРАВА 12



Завдання. Створити шаблон бейджика, в який можна вставити фотографію учня і ввести дані про нього (прізвище, ім'я, клас, назву школи).

1. Створіть новий документ і вставте в нього таблицю 1×1.
2. Встановіть розміри бейджика. Для цього в контекстному меню таблиці виберіть команду **Властивості таблиці**. На вкладці **Таблиця** вікна, що відкривається, встановіть її ширину **9 см**, а на вкладці **Рядок** задайте ширину рядка — **5,5 см**.
3. Розбийте таблицю на клітинки, як показано на рис. 3.32. Для цього перейдіть на вкладку **Макет** і скористайтесь інструментом **Накреслити таблицю**.
4. Додайте до бейджика об'єкти для введення відомостей про учня.
 - а) Перейдіть на вкладку **Розробник** і встановіть режим конструктора, натиснувши відповідну кнопку.

Рис. 3.32

Примітка. За умовчанням вкладка **Розробник** у вікні програми Word не відображається. Щоб додати її на стрічку, на вкладці **Файл** виберіть команду **Параметри** і у вікні, що відкривається, виберіть команду **Настроїти стрічку**. У списку праворуч поставте галочку поряд з написом **Розробник** і натисніть **ОК**.

б) Помістіть курсор у першу клітинку, де має бути фотографія, і в групі **Елементи керування** цієї ж вкладки натисніть кнопку . Клацніть кнопку **Змінити рисунок** у верхній частині щойно доданого елемента і додайте будь-яке зображення з файла на диску.

в) Помістіть курсор у верхню клітинку другого стовпця, натисніть у групі **Елементи керування** кнопку , а потім введіть у створеному елементі керування текст «Прізвище учня».



Рис. 3.33

г) Аналогічно додайте елементи керування до інших клітинок бейджика, відформатуйте їх за зразком (рис. 3.33). Зверніть увагу на те, що слова «Учень» і «класу» введено за межами елемента керування. Відмініть режим конструктора.

5. Зробіть невидимими внутрішні межі таблиці. Для цього перейдіть на вкладку **Макет** і скористайтеся кнопкою **Межі**.

- Збережіть створений документ як шаблон. На вкладці **Файл** виберіть команду **Зберегти як...**, введіть у вікні ім'я файла **Вправа_12_Прізвище**, а в списку **Тип файла** виберіть **Шаблон Word (*.dotx)**.
- Відкрийте щойно створений шаблон і заповніть у ньому поля даними про себе.
- Збережіть документ під іменем **Вправа_12_Прізвище.docx**. Зверніть увагу, що після виконання цієї вправи у вас має бути збережено два документи, які мають те саме ім'я — **Вправа_12_Прізвище**, але різні розширення — **.dotx** та **.docx**.



Висновки

Стиль — це іменованій набір параметрів форматування, які можна застосувати до об'єкта певного типу. Всі об'єкти, відформатовані з використанням одного стилю, матимуть однаковий вигляд.

Якщо до фрагмента застосовано і стиль знака, і стиль абзацу, перевагу має перший з них.

Шаблон документа — це файл з розширенням **.dot** або **.dotx**, у якому зберігаються набір стилів, настройки інтерфейсу користувача програми та призначення комбінацій клавіш.

За умовчанням програма використовує для документа шаблон **Normal.dotm**, але можна обрати для нього інший шаблон або створити власний.



Контрольні запитання та завдання

- Що таке стиль? Які переваги надає застосування стилів?
- Як переглянути інформацію про стилі?
- До яких об'єктів документа Word можна застосувати стилі? Як це зробити?
- Опишіть послідовність створення нового стилю.
- Що таке шаблон документа? Які параметри документа він задає?
- Опишіть порядок створення шаблонів документа.

**Питання для роздумів**

- 1*. Наведіть приклади використання стилів.
- 2*. Як відрізнити документ Word від шаблону Word?

**Завдання для досліджень**


1. Визначте, чи можна при зміні параметрів стилю змінити його тип. Наприклад, стиль було застосовано до абзацу. Чи можна його застосувати до знака?
2. Прочитайте в довідці Word, як створити стиль списку і створіть такий стиль.
- 3*. Визначте, що відбувається в разі видалення стилю з документа. Інформацію з цього питання можна отримати в довідковій системі Word. Ознайомтеся з нею і спробуйте видалити стиль з документа.

§13**Верстка сторінок документа.
Створення змісту і покажчика****13.1.****НАСТРОЮВАННЯ ПАРАМЕТРІВ СТОРІНКИ**

Будь-який документ Word базується на певному шаблоні, в якому серед інших налаштувань задано формат сторінки — за умовчанням це А4 (297×210 мм).

Якщо у вікні програми включено відображення лінійок, ви побачите вертикальну і горизонтальну лінійки, що складаються з кількох шкал. Темні шкали визначають розміри полів сторінки, а білі — розміри ділянки тексту.

**Дізнатися більше**

про режими перегляду документа ви зможете на диску. 

Поля — це незайнята площа сторінки документа між текстом і краями аркуша. За умовчанням для полів встановлено такі розміри: верхнє, нижнє і праве — 1,5 см, лівє — 2,5 см.

Щоб змінити розмір поля, потрібно встановити вказівник миші на межу між темною і білою частинами лінійки і, коли він набуде вигляду двонапрямленої стрілки, перетягнути межу в потрібному напрямку.

Зазначимо, що налаштувати параметри сторінки методом перетягування меж можна в режимі розмітки сторінки.

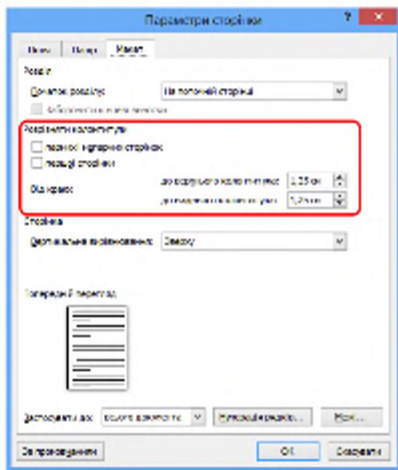


Рис. 3.34. Вкладка Поля вікна Параметри сторінки

Для подання деяких видів інформації (новин, оглядів, блоків, оголошень) текст краще розташовувати в газетних колонках — у такому разі він переходитиме з кінця однієї колонки на початок наступної. Цей спосіб зазвичай застосовують для оформлення газетно-журнальної продукції. Розташування в колонках дає змогу зменшити довжину рядка для тексту, набраного дрібним шрифтом, і покращити завдяки цьому зручність його читання. Уявіть газетну сторінку, набрану шрифтом 9 кегля, на якій рядки йтимуть від краю до краю паперу. Чи зручно їй буде читати? Програма Word надає можливість розташовувати текст у кількох колонках, для цього потрібно виконати певні дії.

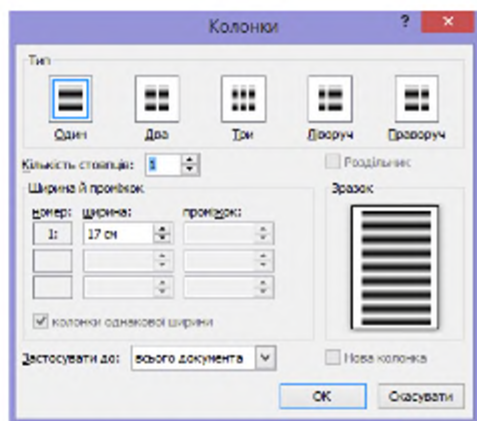



Рис. 3.35. Вікно для встановлення параметрів колонок

Загалом для встановлення параметрів сторінки у програмі Word передбачена окрема вкладка та діалогове вікно (рис. 3.34). З їх допомогою, окрім полів, можна встановлювати розмір та орієнтацію паперу, кількість сторінок на аркуші паперу та інші параметри.

13.2. СТВОРЕННЯ КОЛОНОК

Коли формат і поля сторінки документа визначено, можна переходити до розміщення тексту та розбивки документа на сторінки.

За умовчанням текст розташовується в одну колонку. Такий спосіб використовують у більшості документів, зокрема в рефератах, лабораторних роботах, а також у листах, угодах та інших ділових документах.

1. Переконалися, що відкритий документ у режимі розмітки сторінки, і виділити текст, який потрібно розмістити в кількох колонках.
2. Перейти на вкладку **Розмітка сторінки** та в групі **Параметри сторінки** клацнути кнопки  (**Стовпці**). Обрати кількість колонок. Щоб виконати інші налаштування колонок, потрібно вибрати команду **Додаткові стовпці**, після чого буде відкрито діалогове вікно **Колонки** (рис. 3.35).

Розташування тексту в кількох колонках можна скасувати. Для цього потрібно виділити текст і застосувати до нього режим подання в одну колонку.

ВПРАВА 13

САМОСТІЙНО



Завдання. Створити документ, що містить довільні три вірша, кожен принаймні по 12 рядків. Розмістити кожен вірш у дві колонки. На рис. 3.36 представлено зразок розташування вірша в режимі відображення недрукованих символів.

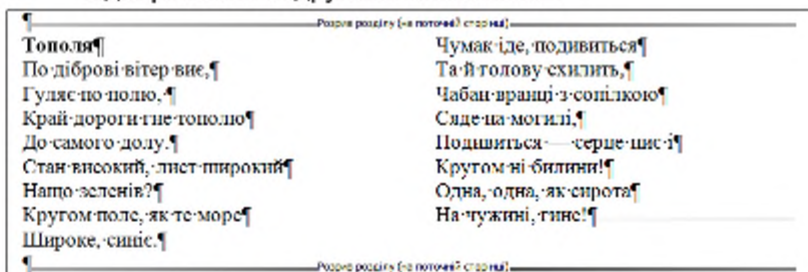


Рис. 3.36

Зберегти файл під іменем **Вправа_13_Прізвище**.

13.3.

ВЕРСТКА СТОРІНОК

Якщо текст не поміщається на одній сторінці, програма автоматично розбиває документ на сторінки. Проте в деяких випадках, наприклад, коли заголовок у кінці сторінки потрібно перенести на наступну сторінку або коли розділ документа має розпочинатися з нової сторінки, виникає потреба в примусовому переміщенні тексту. Щоб розірвати сторінку вручну, потрібно встановити курсор у місце розриву сторінки і натиснути клавіші **Ctrl+Enter** або перейти на вкладку **Розмітка сторінки**, натиснути кнопку **Розриви** й обрати вид розриву. Програма вставить у це місце відповідний недрукований символ.


Розрив сторінки — місце, де закінчується одна сторінка і починається інша.

Програма Word дає змогу по-різному оформлювати окремі блоки тексту, наприклад, розташувати один фрагмент тексту в кількох колонках. Це стає можливим завдяки тому, що документ можна поділити на кілька розділів (не варто плутати їх з розділами і підрозділами, на які текст поділяється за допомогою заголовків).

Розділ — це частина документа, що має певні параметри форматування сторінки.

Новий розділ створюється у разі зміни таких параметрів, як кількість стовпців, розміри полів та орієнтація сторінки. Це робиться автоматично, коли користувач задає розташування блока тексту в кілька колонок чи вибирає у списку **Застосувати до** діалогового вікна **Параметри сторінки** елемент **кінець документа**, або вручну — за допомогою команди **Розрив**.

За умовчанням у документі є лише один розділ. Для створення нового розділу потрібно встановити курсор у місце, де має закінчуватися наявний, натиснути кнопку **Розрив** на вкладці **Розмітка сторінки** і вибрати відповідний варіант у групі **Розриви розділів** панелі, що відкриється. Розділ буде створено, і можна задати для нього параметри сторінки і розташування тексту. Щоб задати інші значення цих параметрів для наступного блока тексту, потрібно буде спочатку створити ще один розділ.

Розриви розділів і розриви сторінок можна побачити в режимі відображення недрукованих символів. Нагадаємо, що його встановлюють за допомогою кнопки  (рис. 3.37).

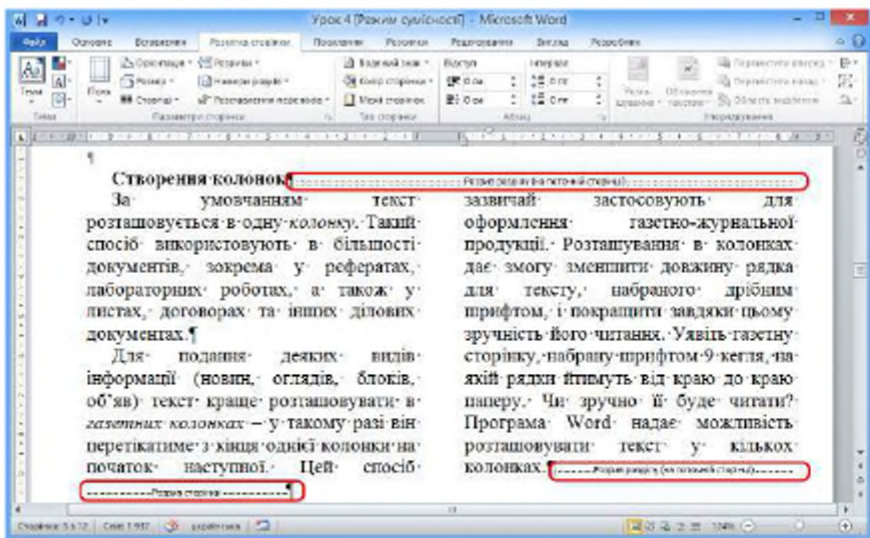


Рис. 3.37. Розрив розділу та розрив сторінки в режимі відображення недрукованих символів

Створення розриву сторінки і розриву розділу можна скасувати. Для цього потрібно спочатку активізувати режим відображення недрукованих символів, а потім встановити курсор на початок лінії розриву і натиснути клавішу **Delete**. Після цього розділ матиме такі самі параметри сторінки і розташування тексту, що й попередній розділ.

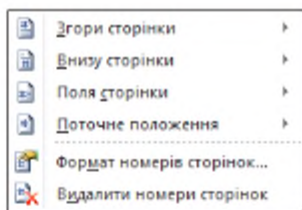



Рис. 3.38

13.4.

НУМЕРАЦІЯ СТОРІНОК

Сторінки великих документів бажано нумерувати. Зробити це можна на вкладці **Вставлення**, вибравши кнопку  (**Номер сторінки**) в групі **Колонтитули**. Після цього відкриється панель, на якій можна обрати розташування номера, встановити його формат у діалоговому вікні, а також видалити номери сторінок (рис. 3.38).

Коли нумерація для документа задана, в ньому автоматично створюється колонтитул і на екрані відкривається вкладка **Конструктор Знаряддя для колонтитулів**.

Колонтитул — це розташований вверху чи внизу сторінки рядок, у якому зазвичай міститься назва документа або розділу та номер сторінки. Колонтитул також може містити малюнок.

Ця вкладка призначена для налаштування параметрів колонтитулів. Наприклад, ви можете, встановивши прапорець **Інакше для першої сторінки**, задати, чи буде номер на першій сторінці документа.

Змінити формат номера сторінки можна у вікні **Формат номера сторінки** (рис. 3.39). Воно відкривається за допомогою однойменної команди, яка міститься на панелі кнопки **Номер сторінки**. Ви знайдете цю кнопку на панелі **Вставлення** або на панелі **Конструктор Знаряддя для колонтитулів** (рис. 3.41).

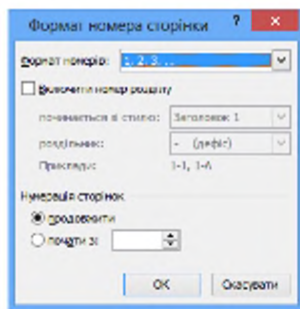


Рис. 3.39

Зазначене вікно дає змогу не лише змінити формат номера (1, 2, 3 або A, B, C), а й додати до номера сторінки номер розділу. Якщо потрібно розпочати нумерацію з номера, відмінного від 1, його треба ввести в поле **почати з:**.

Якщо документ поділено на кілька розділів, для кожного з них можна задати свою нумерацію. Спочатку потрібно задати нумерацію для першого розділу. Потім встановити курсор на початок другого розділу і відкрити вікно **Формат номерів сторінки**. Задати форматування для номерів сторінок цього розділу. Щоб створити наскрізну нумерацію сторінок, треба вибрати перемикач **продовжити**, щоб нумерація була власною, потрібно вказати в полі **почати з:**, із якого номера її розпочати.

13.5.

ОФОРМЛЕННЯ КОЛОНТИТУЛІВ

У документі Word можна створювати як нижні, так і верхні колонтитули. Водночас можна задати однакові колонтитули для всього документа або особливі для певного розділу, першої сторінки розділу, парних і непарних сторінок.

Колонтитули відображаються лише в режимах розмітки сторінки і попереднього перегляду, а також у надрукованих документах.

Щоб створити колонтитул, на вкладці **Вставлення** в групі **Колонтитули** потрібно натиснути кнопку **Верхній колонтитул** або **Нижній колонтитул**. Після чого відкриється панель, на якій можна обрати вигляд колонтитула, змінити або видалити його (рис. 3.40).


Рис. 3.40

У режимі роботи з колонтитулами можна, клацнувши один з колонтитулів, увести в нього текст і, користуючись вкладкою **Знаряддя для колонтитулів Конструктор**, вставити номер сторінки та налаштувати його параметри, додати дату та час, певні експрес-блоки, рисунки, здійснити перехід між колонтитулами, виконати налаштування колонтитулу першої сторінки.

За умовчанням зміни, що вносяться в один колонтитул, відображаються в усіх колонтитулах. Щоб задати для розділу інші колонтитули, на вкладці конструктора треба ввімкнути режим **Як у попередньому** (Як у попередньому). Зв'язок між колонтитулами поточного і попереднього розділів буде розірвано, і можна буде редагувати їх окремо. Для відновлення зв'язку потрібно активізувати зазначений режим.


Для переміщення між верхнім і нижнім колонтитулами використовують кнопку **Перейти до верхнього колонтитула** (Перейти до верхнього колонтитула) або **Перейти до нижнього колонтитула** (Перейти до нижнього колонтитула), а для переходу до колонтитулів інших розділів — кнопки **Попередній** (Попередній) та **Наступний** (Наступний).

Щоб отримати можливість задавати окремі колонтитули для першої сторінки або для парних і непарних сторінок, слід встановити відповідні прапорці в розділі **Параметри** на вкладці **Конструктор** (рис. 3.41).


Рис. 3.41. Панель Конструктор Знаряддя для колонтитулів

Також можна виконати певні настройки у вікні **Параметри сторінки** (рис. 3.42), перейшовши на вкладку **Макет** і встановивши відповідні прапорці на ділянці **Розрізняти колонтитули**.

Зверніть увагу, що на цій вкладці можна задати відстань від краю сторінки до колонтитула. Змінювати розмір і розташування області колонтитула можна й за допомогою лінійок у вікні документа.

Для виходу з режиму роботи з колонтитулами потрібно клацнути кнопку  (Закрити колонтитули) на вкладці **Конструктор** або клацнути два рази мишею в основному тексті.

Вміст колонтитула, а також номери сторінок можна видалити вручну в режимі роботи з колонтитулами — у такий самий спосіб, як видається основний текст документа.

ВПРАВА 14

САМОСТІЙНО



Завдання. Відкрити документ **Вправа_13_Прізвище**. Створивши розриви сторінок, розмістити вірші на різних сторінках. Додати верхній та нижній колонтитули. У верхньому колонтитулі розташувати напис із датою, власним прізвищем та іменем, а в нижньому — назву вірша. Задати нумерацію сторінок (номери мають бути внизу по центру). Зберегти файл під іменем **Вправа_14_Прізвище**.

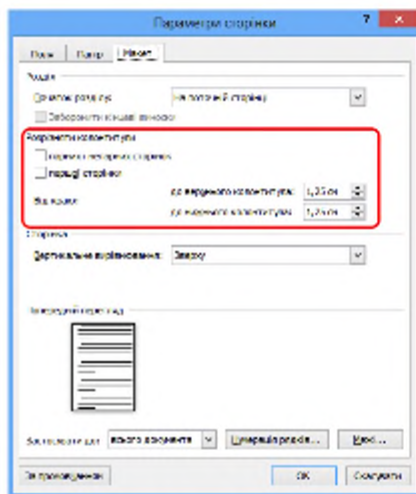


Рис. 3.42

13.6.

СТВОРЕННЯ ПОСИЛАНЬ

Для створення позатекстових посилань, предметного покажчика, змісту і списку літератури в програмі Word призначена вкладка **Посилання**. Розглянемо групи елементів цієї вкладки.

Група **Виноски** дозволяє додавати звичайні та кінцеві виноски (рис. 3.43). Вміст звичайної виноски розташовується безпосередньо в нижньому колонтитулі сторінки, на якій було встановлено виноску, а вміст кінцевої виноски — на останній сторінці документа (рис. 3.44).

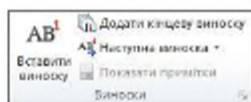


Рис. 3.43

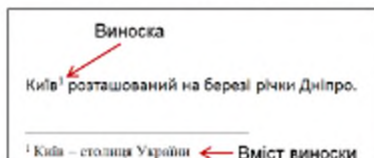


Рис. 3.44

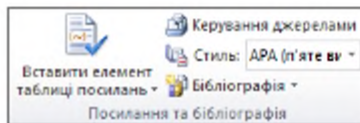


Рис. 3.45

Група **Посилання та бібліографія** (рис. 3.45) пропонує потужні засоби для роботи з джерелами інформації та дозволяє керувати списком літератури, додавати посилання на те чи інше джерело.

Група **Підписи** (рис. 3.46) дозволяє додавати до ілюстрації або таблиці назви, на які можна посилатися в тексті.

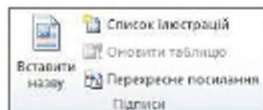


Рис. 3.46

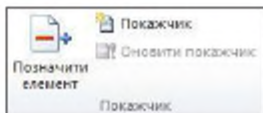


Рис. 3.47

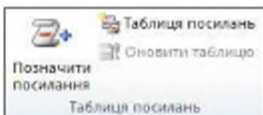


Рис. 3.48

Програма автоматично здійснює послідовну нумерацію нових зображень. Завдяки цьому в тексті можна робити посилання на певне зображення.

Група **Покажчик** (рис. 3.47) дозволяє створити предметний покажчик, який зазвичай розміщують у кінці книжки.

Група **Таблиця посилань** (рис. 3.48) надає можливість створювати в документах таблиці посилань на перелік справ, законодавчих актів та інших авторитетних джерел.

13.7.

СТВОРЕННЯ СТРУКТУРИ ДОКУМЕНТА

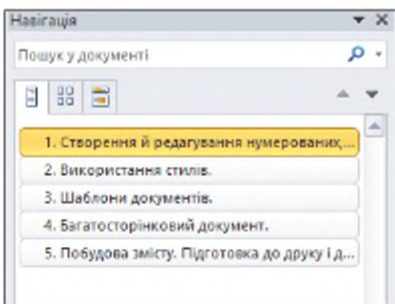


Рис. 3.49

Працюючи з великим документом, доцільно форматувати його з використанням стилів. Насамперед йдеться про призначення заголовкам і підзаголовкам стилів різного рівня. Заголовку верхнього рівня варто призначити стиль **Заголовок 1**, наступного — **Заголовок 2** і т. д. до стилю **Заголовок 6**.

Це дозволяє не лише змінювати форматування всіх заголовків вибраного стилю, а й користуватися перевагами області переходів і швидко виводити на екран потрібний фрагмент документа.

Відобразити цю область можна, встановивши на вкладці **Вигляд** прапорець **Область переходів**. Після цього ліворуч відобразиться панель зі структурою заголовків (рис. 3.49).

Робота на цій панелі подібна до роботи з деревом папок: вибираючи заголовки на панелі, можна переходити до потрібних підрозділів документа.

ВПРАВА 15



Завдання. Оформити заголовки в пропонуваному тексті.

1. Завантажте документ **Історія Інтернету.doc**, який знаходиться в папці **Files** на диску.
2. Для основного тексту змініть стиль **Звичайний** так, щоб він був вирівняний по ширині, мав відступ першого рядка абзацу **1 см**, розмір шрифту — **14**.
3. Для назви тексту застосуйте заголовок першого рівня, обравши стиль **Заголовок 1**.
4. Для назв розділів (**Вступ**, **І. Сучасний розвиток мережі Інтернет...**) застосуйте заголовок другого рівня, обравши стиль **Заголовок 2**.
5. Для назв підрозділів застосуйте заголовок третього рівня.
6. Змініть параметри стилю **Заголовок 2**, встановивши шрифт — **Consolas**, розмір — **14**, колір — темно-синій.

7. Відобразить на екрані область переходів і переміститься з одного розділу до іншого.
8. Збережіть документ під іменем **Вправа_15_Прізвище**.

13.8.

АВТОМАТИЧНЕ СТВОРЕННЯ ЗМІСТУ

Якщо документ великий, для нього доцільно створити зміст. Програма Word формує його автоматично, але це можливо лише в разі, якщо для заголовків застосовано вбудовані стилі заголовків або якщо заголовкам документа вручну призначено рівні в режимі перегляду документа **Структура**.

Щоб створити зміст, потрібно встановити курсор у те місце документа, де він має розміщуватися (зазвичай це початок або кінець документа) і на вкладці **Посилання** в розділі **Зміст** натиснути кнопку



Зміст. На панелі можна обрати

один із вбудованих зразків змісту або вибрати команду **Вставити зміст**. У другому випадку відкриється вікно з активною вкладкою **Зміст** (рис. 3.50).

У цьому вікні в полі **Заповнювач:** можна змінити символ-заповнювач, у полі **Формати:** задати оформлення елементів змісту, в полі **Рівні:** вказати, скільки рівнів заголовків потрібно включити в зміст. Налаштувавши вигляд змісту, треба клацнути кнопку **ОК**, і програма згенерує його. Щоб відредагувати або переформатувати зміст, потрібно клацнути на ньому та внести необхідні зміни. Для оновлення змісту можна скористатися командою **Оновити поле** в контекстному меню змісту (рис. 3.51).

У діалоговому вікні, що відкриється, можна встановити такі параметри:

- **оновити лише номери сторінок** (використовується в разі можливої зміни розташування тексту відносно сторінок, додавання або видалення основного тексту);
- **оновити цілком** (використовується в разі зміни заголовків, включених до змісту).

Для швидкого переходу від змісту до відповідного розділу тексту слід клацнути на його назві в змісті, утримуючи клавішу **Ctrl**.

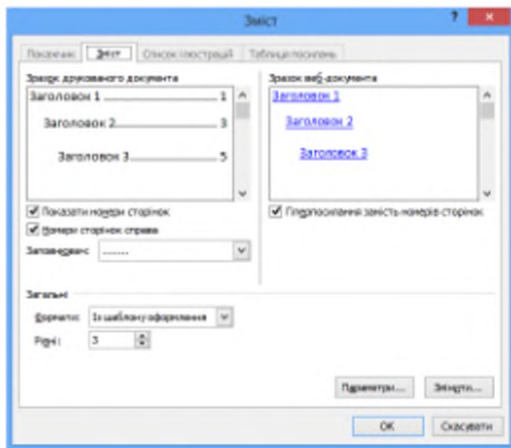


Рис. 3.50. Вікно **Зміст**

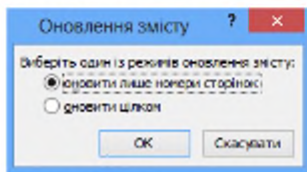


Рис. 3.51. Вікно **Оновлення змісту**

13.9.

СТВОРЕННЯ ПОКАЖЧИКІВ

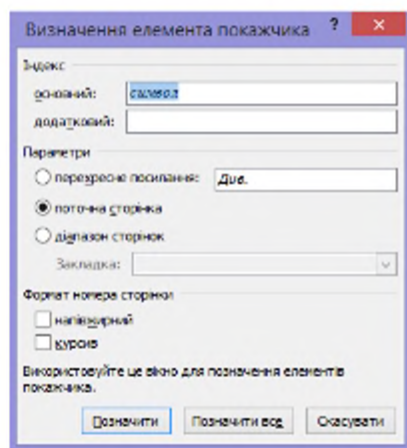


Рис. 3.52. Вікно Визначення елемента покажчика

У наукових книгах, методичних посібниках, підручниках часто створюють предметний покажчик ключових понять і термінів.

Щоб створити предметний покажчик, потрібно спочатку вручну позначити в тексті слова, які мають бути в нього включені, а вже потім автоматично згенерувати предметний покажчик.

Щоб відмітити слово, яке має бути занесено до предметного покажчика, треба виділити його, перейти на вкладку **Посилання**, в групі **Покажчики** обрати кнопку  (**Позначити елемент**). Після цього відкриється діалогове вікно (рис. 3.52), в якому можна встановити параметри відображення сторінок, виконати позначення виділеного слова або всіх слів, що збігаються із заданим у документі.

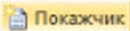
Після встановлення параметрів і закриття вікна елемент покажчика можна побачити в режимі відображення недрукованих символів (у фігурних дужках), як на рис. 3.53.

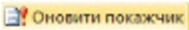
тексту є символ "ХЕ" символ "Я" як

Рис. 3.53

абзаці, 2	рядок, 4
документ, 5	символ, 21, 23
речення, 8	слова, 11

Рис. 3.54. Приклад предметного покажчика

Коли всі необхідні слова буде відмічено, потрібно натиснути кнопку  (**Покажчик**). Після цього відкриється діалогове вікно, в якому треба задати певні настройки. Після закриття вікна в кінці документа буде створено предметний покажчик, який міститиме відмічені раніше слова та номери сторінок, на яких вони розташовані.

Якщо потім буде додано чи вилучено слова, відмічені для покажчика, то його склад необхідно сформувавати знову, скориставшись кнопкою  (**Оновити покажчик**).

На рис. 3.54 подано фрагмент предметного покажчика, що містить деякі ключові слова з цього розділу.

ВПРАВА 16



Завдання. Сформувавати зміст і предметний покажчик.

Відкрийте файл **Вправа_15_Прізвище**. Створіть у ньому титульну сторінку, що має містити назву «Вірші» та відповідну ілюстрацію. Номер з пер-

шої сторінки заберіть. На другій сторінці додайте зміст, який має містити назви віршів з номерами сторінок. У кожному вірші позначте 2 – 3 слова для подальшого включення їх до предметного покажчика. Створіть на останній сторінці предметний покажчик. Збережіть файл під іменем **Вправа_16_Прізвище**.



Висновки

Щоб забезпечити найкраще розташування тексту та інших об'єктів документа, варто спочатку настроїти параметри сторінки — вибрати формат паперу, орієнтацію сторінки і розміри її полів.

За потреби текст на сторінці можна розмістити в кількох колонках. Програма дає змогу задавати кількість колонок та їх ширину.

Розділ — це частина документа, що має певні параметри форматування сторінки.

У програмі Word сторінки документа можна пронумерувати та створити на них колонтитули як верхні, так і нижні.

Колонтитул може містити не лише номер сторінки, а й іншу інформацію. Можна задати різні колонтитули для парних і непарних сторінок, а також для титульної сторінки.

Програма Word дозволяє створювати виноски, списки літератури та покажчики.

Програма Word здатна автоматично створити зміст документа, якщо заголовки відформатовані належним чином.



Контрольні запитання та завдання

1. Що таке розділ? За яких умов створюється новий розділ?
2. Що таке колонтитул? Які є види колонтитулів?
3. Опишіть послідовність дій для додавання номерів сторінки. Як вилучити номер на першій сторінці документа?
4. Як вилучити створений колонтитул або номер сторінки?
5. Які можливості надають інструменти на вкладці **Посилання**?
6. За яких умов можна автоматично створити зміст у документі?



Питання для роздумів

- 1*. Наведіть приклади використання колонтитулів.
- 2*. Як змінити початок нумерації сторінок у документі?



Завдання для досліджень

- 1*. Ознайомтеся з призначенням інструментів групи **Виноски** на вкладці **Посилання**. З'ясуйте, як додавати виноски.
- 2*. Додайте до текстового документа кілька зображень та пронумеруйте їх з допомогою команди **Вставити назву** з групи **Підписи** на вкладці **Посилання**. Поясніть принцип нумерації зображень.

§14

Створення складних документів

14.1.

РОБОТА З КІЛЬКОМА ДОКУМЕНТАМИ

Програма Word надає користувачеві можливість працювати одночасно з кількома документами. Наприклад, створюючи документ, ви переглядаєте інший документ або порівнюєте дві версії того самого тексту. Послідовне відкриття документів будь-яким з відомих вам способів приводить до відкриття кількох документів одночасно із відображенням стосу значків програми Word на панелі завдань. Якщо клацнути на цей стосик, відобразяться міні-вікна відкритих документів, з яких можна обрати документ для виведення на екран (рис. 3.55).



Рис. 3.55

Працюючи з кількома документами одночасно, можна копіювати і переміщувати фрагменти тексту з одного документа до іншого за допомогою буфера обміну, а також вставляти вміст інших файлів у активний документ, той, з яким користувач працює в даний момент.

14.2.

АЛГОРИТМ РОБОТИ НАД СКЛАДНИМ ДОКУМЕНТОМ


При створенні, редагуванні та форматуванні документа великих розмірів із складним форматуванням бажано дотримуватися певної послідовності дій. Ознайомтеся з наведеним нижче орієнтовним алгоритмом роботи над складним документом та виконайте, керуючись ним, наступну практичну роботу. (Звичайно, залежно від завдання та початкового тексту документа деякі пункти алгоритму можуть бути пропущені, переміщені або доповнені.)



1. Обрати формат паперу та задати поля сторінки.
2. Очистити текст від надлишкового форматування.
3. Усунути зайві пробіли, розриви рядків, абзаців тощо.
4. Встановити мову для перевірки правопису.
5. Перевірити правопис.
6. Встановити стиль для основного тексту та застосувати його до всього документа.
7. Виконати форматування окремих фрагментів тексту документа.
8. Створити списки.
9. Визначити структуру заголовків документа, надавши їм відповідні стилі.
10. Додати таблиці, графічні об'єкти, формули тощо.
11. Виконати форматування графічних об'єктів.
12. Оформити титульні сторінки.
13. Додати колонтитули.
14. Задати нумерацію сторінок.
15. Створити зміст.
16. Сформувати покажчик.



Перевірити свої знання, пройти

тест і розгадати кросворд ви зможете на диску. 

ПРАКТИЧНА РОБОТА №5

Структура документа. Автоматизоване створення змісту та покажчиків

1. Скопіюйте до своєї робочої папки файл **7_чудес_Києва.doc**, який знаходиться в папці **Files** на диску, або інший вказаний учителем файл.
2. Задайте параметри сторінок документа.
 - а) Відкрийте діалогове вікно **Параметри сторінки** в групі **Параметри сторінки** на вкладці **Розмітка сторінки**.
 - б) Встановіть для полів зверху, знизу, справа і зліва розмір 1 см.
 - в) Встановіть альбомну орієнтацію.
 - г) У полі **кілька сторінок** встановіть значення — **брошура**.
3. Виділіть увесь текст документа та очистіть його вміст від надлишкового форматування, обравши на панелі **Стил** команду **Очистити все**.
4. Позбавтеся від зайвих пробілів і розривів рядків у документі.
 - а) Для вилучення розриву рядків на вкладці **Основне** в розділі **Редагування** натисніть кнопку **Замінити**. У полі **Знайти** введіть **^l** («л» латинське означає недрукований символ розриву рядка), у полі **Замінити на** — введіть пробіл та натисніть кнопку **Замінити все**. У діалоговому вікні, що з'явиться, натисніть **ОК**.
 - б) Для вилучення зайвих пробілів у цьому ж вікні в полі **Знайти** введіть **^w** (означає всі пробіли між будь-якими двома друкованими символами), а в

полі **Замінити на** — один пробіл і натисніть кнопку **Замінити все**. У діалоговому вікні, що з'явиться, натисніть **ОК**.

5. Задайте для документа режим перевірки українського правопису. Для цього виділіть увесь документ, на вкладці **Рецензування** натисніть кнопку **Мова**, виберіть команду **Вибрати мову перевірки**. У діалоговому вікні, що відкриється, оберіть українську мову та закрийте вікно, натиснувши **ОК**.
6. Перевірте документ на наявність помилок, обравши на вкладці **Рецензування** команду **Правопис і граматика**.
7. Створіть новий стиль для основного тексту та застосуйте його до всього документа. Надайте стилю назву **основний**. Стиль основного тексту повинен мати такі параметри: шрифт — **Times New Roman**, розмір — **14**, міжрядковий інтервал — **одинарний**, вирівнювання — **по ширині**, відступ першого рядка — **1 см**, відступи та інтервали до і після абзаців — **0**.
8. Вивчіть структуру заголовків документа та призначте їм відповідні стилі. Розробіть структуру таким чином, щоб вона мала принаймні два рівні заголовків. За потреби певні заголовки додайте самостійно.
 - а) Для заголовків першого рівня встановіть такі параметри форматування: шрифт — **Arial**, розмір — **16**, напівжирний, міжрядковий інтервал — **одинарний**, вирівнювання — **за лівим краєм**, відступ першого рядка — **1 см**, інтервал після абзацу — **6 пт**, колір — **темно-зелений**.
 - б) для заголовків другого рівня встановіть такі параметри форматування: шрифт — **Arial**, розмір — **15**, напівжирний, курсив, міжрядковий інтервал — **одинарний**, вирівнювання — **за лівим краєм**, відступ першого рядка — **1 см**, інтервал після абзацу — **6 пт**, колір — **синій**.
9. Виконайте редагування заголовків таким чином, щоб перед заголовком був пропущений один рядок, а після заголовка пропущених рядків не було.
10. Перед основним текстом додайте дві чисті сторінки: перша — титульна сторінка, друга — для змісту. Для цього помістіть курсор на початок документа або після назви книжки та два рази натисніть клавіші **Ctrl+Enter**.
11. Створіть розриви сторінок у місцях розташування заголовків першого рівня таким чином, щоб кожний заголовок першого рівня розпочинався з нової сторінки.
12. Виконайте оформлення титульної сторінки: сторінка обов'язково має мати назву та ілюстрацію.
13. Додайте верхній колонтитул, розташували в ньому назву документа.
14. Додайте нумерацію сторінок, розташовуючи її внизу по центру.
15. Створіть зміст. Виділіть у змісті заголовки першого рівня напівжирним шрифтом і встановіть міжрядковий інтервал 1,5 рядка.
16. Додайте покажчик, виділивши перед цим ключові слова за власним бажанням (не менше 20 слів).
17. Збережіть документ під іменем **ПР5_Прізвище**.

Розділ 4

Опрацювання об'єктів мультимедіа



§15

Мультимедіа як ознака сучасного інформаційного світу

15.1.

ПОНЯТТЯ МУЛЬТИМЕДІА

Важко знайти поняття, яке б мало більше трактувань, ніж *мультимедіа*. Під цим терміном розуміють одночасне використання різних форм подання інформації, дані, що містять інформацію в різних формах (зазвичай у звуковій і візуальній), системи, що підтримують інтерактивне використання тексту, аудіо, відео й графіки, особливий вид комп'ютерної технології, сукупність комп'ютерних технологій і т. д.

То що ж таке мультимедіа? Скористаємось означенням Європейської комісії з проблем впровадження і використання нових технологій.

Мультимедіа — це програмний продукт, що містить колекції зображень, текстів і даних, які супроводжуються звуком, відео, анімацією та іншими візуальними ефектами, оснащений інтерактивним інтерфейсом з елементами керування.

Отже, мультимедійний продукт містить такі основні складові: текст, аудіо (звук), зображення, анімацію, відео (рухомі зображення, фільми), інтерактивні елементи.

Слово «мультимедіа» вперше було використано в 1965 році для опису шоу, яке поєднало в собі живу рок-музику, кіно, експериментальні світлові ефекти і нетрадиційне мистецтво. Далі впродовж 40 років цей термін означав різне: скажімо, наприкінці 1970-х років під ним розуміли презентації, одержані із зображень від кількох проекторів, синхронізованих зі звуковою доріжкою. І лише на межі XX і XXI століть це слово набуло сучасного значення.



15.2.

ЗАСТОСУВАННЯ МУЛЬТИМЕДІА



Сьогодні можна говорити про те, що ми живемо в мультимедійному інформаційному середовищі. Мультимедійні продукти і технології так чи інакше використовуються практично в усіх галузях людської діяльності. Ось перелік лише деяких з них.

Креативна економіка (економіка знань) об'єднує галузі, що базуються на інтелектуальній діяльності (реклама, архітектура, кінематографія, дизайн, моделювання одягу, музейна діяльність тощо). Фахівці цих галузей створюють мультимедійний продукт (кіно, реклама, інтерактивні музеї та екскурсії) і застосовують засоби мультимедіа у своїй діяльності під час проектування та моделювання.

Комерційна діяльність. Підприємства та підприємці застосовують мультимедійні презентації, рекламу, сайти з мультимедіа для просування і продажу своєї продукції.

Індустрія розваг. Мультимедіа застосовується для створення спецефектів у кіно та анімації, для створення вражаючих шоу (наприклад, лазерних), інтерактивних ігор.

Освіта. Продуктами мультимедіа є комп'ютерні навчальні курси, електронні навчальні матеріали, підручники, енциклопедії тощо.

Наука та інженерія. Без застосування комп'ютерного моделювання і симуляції практично неможливі сьогодні наукові дослідження, проектування і розробка промислової продукції.

Медицина. Віртуальні операції, моделі людського тіла та процесів, що в ньому відбуваються, широко застосовують під час навчання лікарів.

15.3.

СКЛАДОВІ МУЛЬТИМЕДІА

Як зазначалося, мультимедійний продукт містить такі основні складові: текст, анімація, ілюстрації, аудіо (звук), відео (рухомі зображення, фільми), інтерактивні елементи. Розглянемо їх докладніше.

Текст — важлива складова таких мультимедійних продуктів, як електронні презентації, сайти, електронні підручники. Великий вибір шрифтів, різноманітні статичні й динамічні ефекти для оформлення написів дозволяють розробникам мультимедіа підвищити сприйняття тексту та досягти високих естетичних результатів під час його оформлення.



Ілюстрації. Знімки, малюнки, комп'ютерна графіка — усе це є сьогодні в розпорядженні тих, хто прагне створити ефективний мультимедійний продукт. Розвиток цифрової фотографії, графічних редакторів і мобільних пристроїв сприяли тому, що сьогодні майже кожен може створювати якісний ілюстративний матеріал.

Анімація. Під анімацією розуміють серію зображень, які відтворюються послідовно та імітують рух. Для її створення можна застосовувати програмні технології на зразок Adobe Flash, а також анімовані GIF-файли.

Аудіо (музика, голос, звукові ефекти) — невід'ємна частина багатьох мультимедійних продуктів. Наприклад, голос диктора в навчальних курсах, музичний супровід в іграх, підкасти (цифрові аудіофайли, що поширюються через Інтернет і призначені для відтворення на ПК та медіапрогравачах). Аудіо записують за допомогою мікрофона або MIDI-інструментів, наприклад MIDI-клавіатури, що імітує клавіатуру фортепіано.

Відео. На відміну від анімації, що складається з окремих зображень, які відтворюються послідовно, відео — це неперервний потік візуальної інформації, який розділяється на окремі кадри під час запису. Коли вони відтворюються з певною частотою (зазвичай 30 кадрів/с), око сприймає їх як безперервний потік візуальної інформації. Відео отримують за допомогою відеокамери, а розповсюджують за допомогою телебачення, кінофільмів (записаних на носіях) та через Інтернет.

Інтерактивні елементи створюються в програмний спосіб і забезпечують взаємодію з користувачем. Це — різноманітні елементи інтерфейсу користувача, гіперпосилання, керовані користувачем анімаційні об'єкти.

15.4.

СТВОРЕННЯ GIF-АНИМАЦІЇ

Одним із простих способів створення анімації є використання можливостей формату GIF (*Graphic Interchange Format* — формат обміну зображеннями). Насамперед йдеться про те, що у файлі GIF можна зберегти кілька зображень і задавати для них час відображення на екрані. Коли такий файл відкрити в програмі, що підтримує цю властивість, наприклад, у браузері, зображення починають відтворюватися послідовно.

Формат GIF підтримує лише 256 кольорів зображення, але серед них може бути колір, визначений як прозорий. Цю властивість формату підтримують браузери, а тому крізь піксели зображення, які є прозорими, буде видно фон, на якому це зображення розташоване. Ці дві властивості формату GIF дозволяють створювати анімовані зображення, зокрема для банерів. Загальний порядок створення GIF-анімації такий:



Дізнатися більше
про створення
GIF-анімацій ви зможете
на диску.

1. У графічному редакторі підготувати кадри анімації.
2. Завантажити кадри в редактор для створення GIF-анімації.
3. Встановити для кожного з них тривалість відтворення і зберегти файл.

GIF-анімацію можна створити із спеціально створених зображень, набору фотографій, а також з відео. В останньому випадку можна скористатися одним з онлайн-сервісів (детальніше цю тему ми розглянемо на одному з наступних уроків).

ВПРАВА 17



Завдання. Створити GIF-анімацію, що демонструє рух м'яча, який відскакує від поверхні (рис. 4.1). Скористайтесь програмою Microsoft GIF Animator, яку легко можна знайти в Інтернеті. Інсталювати програму не треба — просто завантажте її на комп'ютер, запустіть на виконання і можете працювати.



Рис. 4.1. Кадри для GIF-анімації

1. Відкрийте графічний редактор Paint та створіть у ньому перший кадр анімації: Дійте за таким планом.
 - а) Встановіть для малюнка розміри, наприклад, 100×200 пікселів, відобразіть сітку, збільште зображення на екрані, щоб вам зручно було працювати;
 - б) Обравши відповідні кольори, намалюйте горизонтальну поверхню, застосуйте для цього інструмент **Прямокутник**, задайте для нього суцільну заливку без контуру.
 - в) Встановіть інші кольори і за допомогою інструмента **Круг** намалюйте м'ячик, розташований на поверхні.
 - г) Збережіть малюнок у файлі **Кадр_1.gif**.
2. Створіть другий, третій, четвертий і п'ятий кадри за алгоритмом:
 - а) Виділіть м'ячик та перемістіть його приблизно на 10 пікселів, у другому, третьому і четвертому кадрах — уверх, а в п'ятому — униз (якщо під час виділення ви захопите кольорові пікселі, при переміщенні утвориться шлейф; зафарбуйте його білим кольором або зробіть інше виділення).
 - б) Збережіть малюнок у файлі **Кадр_N.gif**, де N = 2, 3, 4, 5.
3. Створіть шостий кадр так само, як і попередні, але м'ячик перемістіть униз до самої поверхні та розтягніть його трохи в ширину, щоб створити ефект стискання м'яча від удару об поверхню. Збережіть малюнок у файлі **Кадр_6.gif**. Кадри анімації готові.



4. Відкрийте програму Microsoft GIF Animator, клацнувши файл **gifanimator.exe**. Відкривши **Провідник**, перетягніть кадри анімації у вікно GIF Animator.
Інший спосіб: відкрийте один з кадрів у програмі, а потім, натискаючи кнопку у вигляді порожнього аркуша з хрестиком, завантажте інші кадри (слідкуйте за порядком кадрів).
5. Перейдіть на вкладку **Animation** (Анімація). Послідовно вибираючи кадри анімації, встановлюйте для них у полі **Duration** (Тривалість) значення 10 с.
6. Збережіть анімацію у файлі **Вправа_17_Прізвище.gif**. Відкрийте файл анімації.

**Висновки**

Мультимедіа — це програмний продукт, що містить колекції зображень, текстів і даних, які супроводжуються звуком, відео, анімацією та іншими візуальними ефектами, і оснащений інтерактивним інтерфейсом з елементами керування.

Мультимедіа застосовують для створення комп'ютерних навчальних матеріалів, реклами, комп'ютерних ігор, а також для моделювання та симуляції в медицині, математиці, наукових дослідженнях.

Один із способів створення анімації — використання можливостей формату GIF.

**Контрольні запитання та завдання**

1. Що таке мультимедіа?
2. У яких галузях людської діяльності застосовують мультимедіа?
3. Опишіть основні складові мультимедіа, наведіть приклади, що демонструють роль кожної складової.
4. У який спосіб і з яких джерел отримують анімацію, аудіо та відео?
5. Через які властивості формат GIF використовують для створення анімації?
6. Опишіть порядок створення GIF-анімації.

**Питання для роздумів**

- 1*. Чому підтримка прозорості у форматі GIF є такою важливою особливістю для створення анімації?



Завдання для досліджень

- 1*. Вставте створений у вправі 17 анімований GIF у документ Word. Який вигляд він матиме? Спробуйте пояснити, чому це відбувається.
- 2*. З'ясуйте, які є онлайнні ресурси для створення анімованих GIF-файлів. Як вони працюють?

§16

Зберігання і відтворення мультимедійних даних

16.1.

АУДИОДАНИ ТА АУДИОФОРМАТИ

Щоб зафіксувати звуки довкола нас (голоси, музику, звуки природи тощо), застосовують техніку звукозапису. Загалом для цього потрібно мати:

- 1) мікрофон — пристрій, що перетворює акустичні коливання (рис. 4.2) на електричний сигнал, або звуковий синтезатор;
- 2) пристрій, що перетворює електричні коливання на послідовність цифр (ми розглядатимемо цифровий запис);
- 3) пристрій для запису даних на цифровий носій (жорсткий диск, компакт-диск тощо).

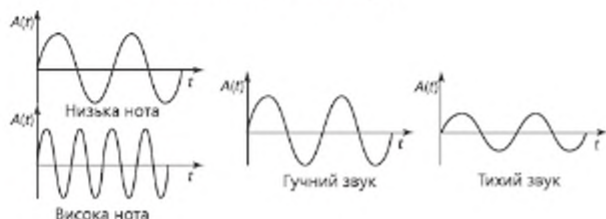


Рис. 4.2. Акустичне коливання (звукова хвиля) характеризується частотою і амплітудою коливань

Звукові сигнали, щоб зберегти в цифровому вигляді, дискретизують, тобто через певні проміжки часу вимірюють амплітуду сигналу і записують її величину в двійковій формі.

Під час відтворення звуку ці дані потоком передаються на відповідний пристрій. Наприклад, якщо йдеться про звук у форматі CD-Audio, то за 1 с передається 1411,2 Кбіт. Кількість інформації, переданої за одиницю часу під час відтворення файла, називають *бітровою частотою* (*бітрейтом*, англ. *bit rate*) і вимірюють у бітах за секунду (біт/с). Чим більший бітрейт, тим ближчий звук до оригіналу. Обсяг звукового файла в байтах дорівнює добутку бітрейту на тривалість звучання в секундах, поділеному на 8.



Для зменшення обсягу звукових даних застосовують стиснення. Стиснення звуку з втратами зазвичай базується на тому, що люди не розрізняють або погано розрізняють негучні сигнали на фоні гучних.

Алгоритми стиснення видаляють негучні сигнали, завдяки чому вдається значно зменшити значення бітрейту. Сьогодні найчастіше використовують бітрейти величиною 256, 192 і 128 кбіт/с.

Для аматорських завдань із звукового оформлення кліпів можна записати голос за допомогою диктофона, вбудованого в смартфон. Також можна скористатися аудіозаписами з Інтернету.

Формати файлів для зберігання звуку

Оскільки для зберігання звукової інформації застосовують різні методи кодування та стиснення, існує багато звукових форматів файлів. Ознайомтеся з деякими з них.

MP3 (Motion Picture Experts Group Layer 3). Забезпечує високий ступінь стиснення даних (в 11 разів вищий порівняно з форматом CD-Audio) і дає змогу створювати файли невеликого обсягу, які мають розширення **mp3**. Використовує стиснення з втратами, підтримує стерео і потокове передавання. Сьогодні вважається найпоширенішим форматом зберігання аудіозаписів в Інтернеті.

Monkey's Audio. Формат кодування аудіо без втрат. Файли мають розширення **.ape** (аудіодані) та **.apl** (метадані). Забезпечує дуже високий ступінь стискання аудіо, проте распаковування вимагає великих обчислювальних ресурсів.

WMA (Windows Media Audio). Ліцензований формат, розроблений корпорацією Microsoft як альтернатива MP3. Номінально він забезпечує більш ефективне стиснення даних порівняно з конкурентом, однак загалом його переваги не є очевидними. Як послідовний прихильник дотримання авторських прав корпорація Microsoft забезпечила підтримку формату цифрової системи керування авторськими правами (DRM), що унеможлиблює прослуховування захищеного запису на будь-якому комп'ютері, крім того, з якого він був завантажений.

Cue Sheet (файл розмітки). Текстовий файл з метаданими, що описують послідовність і тривалість треків на CD (також є файли розмітки для DVD і Blu-ray). Підтримуються багатьма програмами для запису оптичних дисків та медіапрогравачами.

16.2.

ВІДЕОДАНИ ТА ВІДЕОФОРМАТИ

Щоб зафіксувати візуальну інформацію, можна застосовувати *відеозапис* — електронну технологію запису відеосигналу (сигналу телебачення) або цифрового потоку відеоданих на фізичний носій.



Звичайно, це можна робити і за допомогою комп'ютера. У такому разі йдеться про захоплення відео — процес перетворення відеосигналу із зовнішнього джерела на цифровий потік і запис його у відеофайл.

Зовнішнім джерелом можуть бути відеокамери, магнітофони, потокове відео з мережі, TV-тюнери, ресивери цифрового телебачення та інші пристрої.

Визначимо, що відеозображення телевізійної якості в «початковому» вигляді мають дуже великий обсяг даних. Проведемо прості розрахунки, виходячи з того, що відео — це потік нерухомих зображень-кадрів. Якщо розмір статичного зображення становить 800×600 пікселів (точок), а колір кожного пікселя кодується двома байтами, то зображення займатиме в пам'яті $800 \times 600 \times 2 = 960\,000$ байт, тобто трохи менше 1 Мбайт. Це начебто й небагато: на компакт-диску поміститься понад 700 таких зображень. Але якщо йдеться про фільм, стандартна швидкість відтворення якого становить 24 кадри за секунду, то в не стисненому вигляді на компакт-диск можна буде записати лише фрагмент тривалістю $700 : 24 = 29$ с. Ось чому відеозображення неодмінно потребують стиснення (компресії).

Для стиснення відеоданих застосовують алгоритми стиснення з втратами, коли початкові дані можна відновити лише з викривленнями, які можуть бути непомітними для людини.

Алгоритми стиснення відео ґрунтуються на відповідних алгоритмах для статичних зображень, використовуючи їх у різних поєднаннях і послідовностях. Але є і додаткові методи компресії. Наприклад, кожний наступний кадр можна зберігати не повністю, а лише його відмінності від попереднього. Інший алгоритм — Motion Prediction (передбачення руху) — базується на тому, що об'єкти зазвичай швидко рухаються, але повільно змінюються. З огляду на це зображення поділяють на блоки, порівнюють їх з відповідними блоками попереднього кадру і найменш відмінні зберігають всього один раз на кілька кадрів.

Формати файлів для зберігання відео

Як і аудіо, відео зберігається у файлах різноманітних форматів. Розглянемо найпоширеніші з них.

WMV (Windows Media Video). Формат розроблений корпорацією Microsoft для програвача Windows Media. Відеофайли, збережені в цьому форматі, мають розширення **.wmv** або **.asf**.

MPEG. Створений групою MPEG (Moving Pictures Experts Group) ще на початку 1990-х років. Файли цього формату мають розширення **.dat**, **.mpg**, **.mpe**, **.mpeg**, **.mpl**, **.mp2** або **.mp4**. Група MPEG розробила три алгоритми стиснення відеоданих. MPEG-1, який був створений першим,



забезпечував середню якість зображення і не дуже високий ступінь стиснення. MPEG-2 давав відчутно кращу якість відеозображень. MPEG-4 забезпечує ще вищу якість і більший ступінь стиснення порівняно зі своїми попередниками, підтримує додаткові можливості (захист від несанкціонованого копіювання, використання інтерактивних елементів) і потокове передавання відео.

16.3.

КОДЕКИ

Алгоритми стиснення та кодування мультимедійних даних реалізують у спеціальних програмах, що називаються *кодеками*. Інколи кодеками називають також пристрої, здатні перетворювати аналоговий сигнал на цифровий і навпаки. Зокрема, кодеком є звукова карта.

Кодек (англ. *codec*, від *coder/decoder* — кодування/декодування) — програма, що перетворює потік даних або сигнали для їх подальшого використання. Кодеки можуть як кодувати, так і розкодувати мультимедійні дані.

Звукові та візуальні дані потребують різних методів стиснення, а тому для них розроблені окремі кодеки. Кожен кодек використовує свої алгоритми стиснення і зазвичай не працює з даними від іншого кодека. Кодеки взаємодіють з певними прикладними програмами, зокрема з медіапрогравами, і допомагають їм відтворювати медіадані. Кодеки вбудовано в медіапрограми, їх можна встановити як окремі програми (наприклад, FLAC, WavPack, LAME, Div, Xvi) або як пакети, найбільшим і найпопулярнішим з яких сьогодні вважається K-Lite.

16.4.

МЕДІАКОНТЕЙНЕРИ

Для того, щоб створити мультимедійний об'єкт, часто недостатньо зберегти дані в одному з форматів, що забезпечують стиснення і кодування даних, оскільки вони не надають можливість синхронізувати (узгодити в часі) відео- та аудіопотоки і додати до них супровідний текст або меню. Такі можливості є у *форматів-медіаконтейнерів*.

Медіаконтейнер — формат, що дає змогу розміщувати в одному файлі мультимедійні дані різних типів і синхронізувати звук, відеозображення і текстову інформацію.

Розглянемо кілька найпоширеніших форматів-контейнерів.

WAV (Waveform Audio Format). Контейнер для зберігання нестисненого звуку. Створений корпорацією Microsoft і прийнятий як стандарт для звукового супроводу роботи системи Windows і комп'ютерних ігор.



Звук у форматі WAV зберігається без втрати якості, але відсутність стиснення призводить до того, що обсяги wav-файлів дуже великі. Через цей та деякі інші недоліки формат WAV не набув популярності серед користувачів, проте його і зараз широко використовують під час редагування звуку та на переносних аудіопристроях, зокрема на цифрових диктофонах.

Matroska Multimedia Container. Формат мультимедійного контейнера та набір інструментів для роботи з ним. Розширення файлів у цьому форматі: **.mkv** — для відео (з субтитрами та звуком), **.mka** — для аудіофайлів, **.mks** — для субтитрів, **.mk3d** — для 3D-відео.

AVI (Audio and Video Interleaved). Надає можливість об'єднувати нестиснені або закодовані різноманітними кодеками аудіо- та відеодані. Донедавна AVI був лідером серед форматів для платформи Windows, однак через певні обмеження, зокрема на обсяг файла (не більше 4 Гбайт), його популярність зменшилася.

MOV (QuickTime Movie). Контейнер, розроблений корпорацією Apple Computer спеціально для мультимедійного програвача QuickTime, який можуть відтворювати й інші програвачі. Як і формат AVI, дозволяє поєднувати аудіо- та відеопотоки, закодовані в різний спосіб.

16.5.

КОНВЕРТУВАННЯ ВІДЕО ТА АУДІО

Якщо вихідний формат відео- або аудіофайла не підходить для обробки і відтворення, можна конвертувати файл в інший формат. Для цього є досить багато різноманітних безкоштовних програм, зокрема й онлайн-ових.



Дізнатися більше
про роботу
відеоковертера ви
зможете на диску.

Конвертування може здійснюватися як у певний формат, так і у формат певного пристрою (ігрова консоль, смартфон тощо). Програми-конвертори також часто мають функції з монтажу, об'єднання відео, створення ефектів і субтитрів. Огляди та переліки таких програм легко знайти в Інтернеті.

16.6.

МУЛЬТИМЕДІЙНІ ПРОГРАВАЧІ

Для відтворення (прослуховування чи перегляду) аудіо- та відеофайлів використовують спеціальні програми — *програвачі* (їх ще називають *плеєрами*), переважно більшість яких можна безкоштовно завантажити з Інтернету. Перелічимо найпопулярніші програвачі, які підтримують більшість поширених форматів мультимедійних файлів.

VLC Media Player. Потужний програвач, що відворює більшість аудіо- та відеоформатів, зокрема MPEG-1, MPEG-2, MPEG-4, DivX, DVD, VCD і т. д.



Media Player Classic Home Cinema. Безкоштовний програвач з простим і зручним інтерфейсом. Підтримує дуже широкий спектр аудіо- та відеоформатів. Якщо файл не вдається відтворити за допомогою інших програм, варто спробувати зробити це в Media Player Classic.

GOM Player. Безкоштовний медіаплеєр для Windows. Може програвати більшість медіафайлів без сторонніх кодеків, а також пошкоджені та недозавантажені файли. Підтримує знімок кадру, запис звуку, субтитри.

Програвач Windows Media (Windows Media Player). Входить до складу операційної системи Windows (рис. 4.3), відтворює аудіо- та відеофайли більшості популярних форматів.

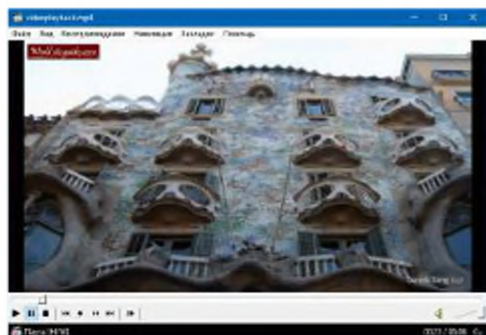


Рис. 4.3. Інтерфейс мультимедійного програвача

ВПРАВА 18



Завдання. У практичній роботі №6 потрібно створити відеокліп, присвячений творінням видатного іспанського архітектора Антоніо Гауді (1852 — 1926). Для цього необхідно завантажити з YouTube два відео, на яких показані будинок Бальо (будинок Батло, Casa Batlló) і парк Гуель (Park Güell). Проте формат цих відео не підтримується програмою-редактором, у якій треба зробити відеокліп. Тому потрібно конвертувати ці відео в інший формат.

1. Відкрийте файл `casa_batllo.mp4` у вікні медіапрогравача.
2. Запустіть процес його відтворення.
3. Виберіть команду **Файл** ▶ **Свойства** (Файл ▶ Властивості) і передивіться інформацію про файл у вікні, що відкриється. Запишіть дані про обсяг файла в мегабайтах у таблицю.
4. Відкрийте файл `park_guel.mp4`, передивіться його властивості та запишіть дані про обсяг файла в таблицю.
5. Конвертуйте обидва файли у формат AVI, скориставшись програмою, яку вам вкаже вчитель.
6. Послідовно відкрийте файли `casa_batllo.avi` та `park_guel.avi`, передивіться їх властивості та запишіть дані про обсяг файла в таблицю.



7. Порівняйте отримані обсяги. Зробіть висновки.

Назва файла	Обсяг файла MP4	Обсяг файла AVI
casa_batllo.mp4		
park_guel.mp4		



Висновки

Щоб зафіксувати звуки довкола нас (голоси, музику, звуки природи тощо), застосовують техніку звукозапису.

Захоплення відео — процес перетворення відеосигналу із зовнішнього джерела на цифровий потік і запис його у відеофайл з метою обробки, зберігання та відтворення. Зовнішнім джерелом можуть бути відеокамери, магнітофони, потокове відео з мережі, TV-тюнери, ресивери цифрового телебачення та інші пристрої.

Для стиснення аудіо- та відеоданих застосовують алгоритми стиснення з втратами, коли початкові дані можна відновити лише з викривленнями, які можуть бути непомітними для людини.

Кодек — програма, що перетворює потік даних або сигнали для їх подальшого використання. Кодеки можуть як кодувати, так і розкодувати мультимедійні дані.

Медіаконтейнер — формат, що дає змогу розміщувати в одному файлі мультимедійні дані різних типів і синхронізувати звук, відеозображення і текстову інформацію.

Для *відтворення* (прослуховування чи перегляду) аудіо- та відеофайлів використовують спеціальні програми — *програвачі*.



Контрольні запитання та завдання

1. Де можна взяти аудіо для аматорського проекту?
2. Що таке захоплення відео? З яких пристроїв можна одержати відеопотік?
3. Назвіть найпоширеніші формати для зберігання аудіо- та відеофайлів.
4. Що таке кодек? Яке його призначення?
5. Що ще крім аудіо- та відеоданих можуть містити мультимедійні контейнери?
6. Які програвачі мультимедіа вважаються найпопулярнішими?

**Питання для роздумів**

- 1*. Чи можна, на вашу думку, створити один кодек для звуку та відео?
- 2*. Чому під час редагування звуку використовують файли формату WAV, а не MP3?
- 3*. Яку проблему, що виникає в разі окремого зберігання аудіо- та відеоряду, дозволяють вирішити мультимедійні контейнери?

**Завдання для досліджень**

- 1*. Знайдіть в Інтернеті інформацію про формати-медіаконтейнери ASF, MP4, RealMedia, 3gp. Які вони мають характерні особливості?
- 2*. З'ясуйте, що таке відео високої чіткості. Що означають написи 720p, 1080p, 1080i, 4K?
- 3*. Які є можливості щодо конвертування аудіо та відео он-лайн?

§17**Основи обробки мультимедійних даних****17.1.****ОГЛЯД ПРОГРАМИ КІНОСТУДІЯ WINDOWS**

«Життєвий цикл» будь-якого електронного документа складається з кількох фаз: спочатку його створюють, потім редагують, поширюють та переглядають. Створюють і редагують документи, як правило, за допомогою програм-редакторів, а переглядають або в тих самих редакторах, або в спеціальних переглядачах чи програвачах. Не є винятком і мультимедійні документи — аудіо- та відеофайли. Призначені для них програвачі ми розглядали в попередньому параграфі, а тепер почнемо працювати з простим мультимедійним редактором — програмою Windows Movie Maker (Кіностудія Windows) версії 2.6. З допомогою цієї програми можна змонтувати власний фільм просто на домашньому комп'ютері, записати його на компакт-диск, надіслати друзям електронною поштою або розмістити в Інтернеті.

Windows Movie Maker раніше входила до складу операційних систем Microsoft Windows. Потім версію програми під назвою Кіностудія Windows було включено до набору Основи Windows, який завантажуєть-



ся та встановлюється окремо. Можливості цієї версії щодо редагування відео менші, тому для того, щоб оволодіти основними прийомами обробки відео, краще скористатися програмою Windows Movie Maker версії 2.6. Основні її функції перелічені на рис. 4.4.

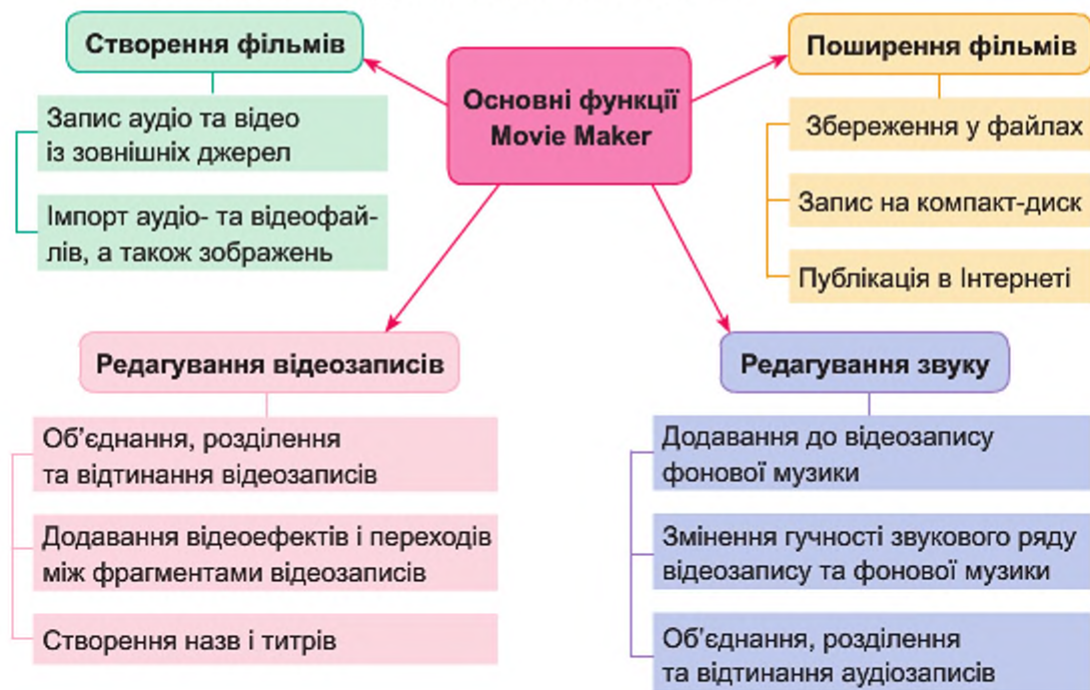


Рис. 4.4. Основні функції програми Windows Movie Maker

17.2.

ДОКУМЕНТИ Й ОБ'ЄКТИ КІНОСТУДІЯ WINDOWS

Документи Windows Movie Maker називають *проектами* і зберігають у файлах з розширенням `.mswmm`. Їх можна відкривати і редагувати лише за допомогою цієї програми. Однак метою роботи користувача в Movie Maker є створення не проекту, а відеофайла, який можна переглядати й поширювати. Програма дає можливість записувати готові фільми у форматі WMV (Windows Media Video).

У проектах Movie Maker можна працювати з об'єктами таких типів:

- відеозаписи (фільми);
- аудіозаписи (звуки і музика);
- зображення;
- відеоефекти (ефекти, застосовані до фрагментів відеозаписів);
- переходи (ефекти, що відображуються під час зміни фрагментів);
- назви і титри (текстові написи на відеозображенні).



Об'єкти перших трьох типів зберігаються в окремих файлах і можуть бути імпортовані в проект, а об'єкти трьох останніх є внутрішніми об'єктами програми і зберігаються в одному файлі з проектом або фільмом. Відеозаписи автоматично чи вручну поділяються на фрагменти, які називаються *кліпами*. Це основні об'єкти, з якими працюють під час монтажу фільму. На фрагменти можна ділити й аудіозаписи.

17.3.

ІНТЕРФЕЙС ПРОГРАМИ WINDOWS MOVIE MAKER

У верхній частині вікна програми Movie Maker розташовано рядок меню та панель інструментів, а внизу — рядок стану. Між ними міститься робоча область, яка складається з кількох частин (рис. 4.5).

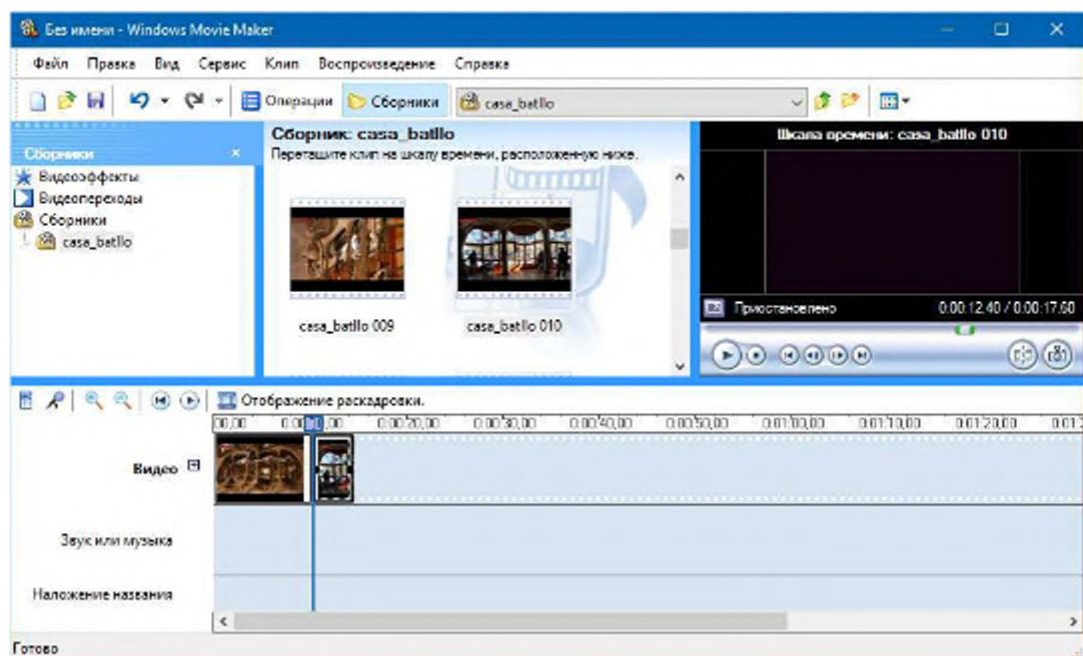


Рис. 4.5. Вікно програми Windows Movie Maker

Нижню частину призначено для налаштування часових параметрів фільму. В ній відображається або *часова шкала* (як на рис. 4.5), або *розкладка кадрів*. На розкладці можна визначати лише порядок кліпів у фільмі, а на часовій шкалі налаштувати їх відтворення з точністю до сотих часток секунди.

Верхня частина вікна поділена на три частини.

- Ліворуч за умовчанням відображається *панель завдань*, призначена для спрощення доступу до найуживаніших команд.



- У центрі верхньої частини робочої області розташована *область вмісту*, в якій відображається вміст поточного документа: збірки кліпів, відеоефекти або переходи між фрагментами фільму. Для перемикання між цими групами елементів потрібно вибрати відповідне значення зі списку **Папка**, розташованого на панелі інструментів.
- Праворуч вгорі розміщено *вікно монітора* з програвачем, який дає змогу відтворити мультимедійний об'єкт, вибраний в області вмісту.

17.4.**ПОЧАТОК РОБОТИ НАД ФІЛЬМОМ**

Розглянемо основні етапи роботи над фільмом у середовищі Movie Maker. Передусім зазначимо, що перед початком роботи в цій програмі доцільно розробити загальний план фільму, тобто описати його структуру, послідовність і тривалість відтворення всіх складових. Це дасть змогу в майбутньому зосередитись на технічних аспектах роботи над відеофільмом й уникнути його переробки чи реструктуризації. Створення плану фільму можна вважати «нульовим» етапом. Наступні етапи такі:

1. Імпорт аудіо- чи відеозаписів з файла, відеокамери, TV-тюнера або іншого джерела.
2. Монтаж фільму, тобто розрізання, «склеювання» і налаштування параметрів його фрагментів.
3. Запис фільму у файл, на компакт-диск або публікація його в Інтернеті.

Імпорт мультимедійних об'єктів

Щоб імпортувати в проект Movie Maker відеозапис із файла, потрібно клацнути відповідне посилання в розділі **Запис відео** (Запис відео) на панелі завдань (рис. 4.5), вибрати потрібний файл у діалоговому вікні **Імпорт файла** (Імпорт файла) і клацнути кнопку **Імпорт** (Імпорт). З'явиться вікно, в якому відобразатиметься хід процесу імпорту. За потреби його можна зупинити кнопкою **Отмена** (Скасувати).

Якщо у вікні **Імпорт файла** (Імпорт файла) встановлено прапорець **Создание клипов для файлов видео** (Створення кліпів для файлів відео), то під час імпорту довгі відеозаписи автоматично поділятимуться на кліпи (невеликі відеофрагменти). Новий кліп починається тоді, коли повністю змінюється зображення на відеокадрі, тобто коли починається новий епізод, відбувається зміна плану тощо. Кожний кліп в області вмісту зображується значком, під яким вказана назва цього кліпу. У кінці назви міститься його порядковий номер. Імпорт файлів аудіозаписів і зображень здійснюється так само, але поділ на кліпи в цьому разі не робиться; кожному такому файлові відповідає один кліп.




Аудіо- та відеоряд можна імпортувати не лише з файлів, а й із зовнішніх пристроїв, зокрема з камери, веб-камери, звукової плати, TV-тюнера. Для цього призначено посилання **Запис з відеоустройства** (Запис з відеопристрою) на панелі завдань.

! Важливо розуміти, що під час імпорту об'єкти не переносяться і не копіюються у файл проекту — їх файли залишаються там, де і були раніше, а в середовищі Movie Maker ви працюєте лише з посиланнями на об'єкти. Тому файли, що використовуються в проектах, краще не видаляти і не перейменовувати. Якщо є достатньо вільного дискового простору, перед створенням проекту всі необхідні ресурси доцільно скопіювати до окремої папки, з якої вони згодом імпортуватимуться.

Збірки кліпів

Під час монтажу користувач зазвичай працює з кількома вихідними відеофайлами, кожен з яких поділяється на сотні, а інколи й на тисячі кліпів. З метою впорядкування кліпи розміщуються в папках, що називаються *збірками кліпів*. Панель збірок відображається в тій самій області, що й панель завдань. Щоб побачити збірки, треба клацнути кнопку **Сборники** (Збірки) на панелі інструментів, а щоб повернутися до панелі завдань — кнопку **Операции** (Операції).

Для кожного імпортованого відеофайла відводиться окрема збірка. Можна створити і власну збірку, клацнувши кнопку  **Создать папку сборников** (Створити папку збірок). У власні збірки з області вмісту можна перетягувати кліпи потрібні для монтажу відеосюжетів.

17.5.

ОСНОВИ МОНТАЖУ ФІЛЬМУ

Коли план фільму створено й у середовищі мультимедійного редактора імпортовано всі необхідні об'єкти, можна починати монтаж фільму, тобто розміщувати його фрагменти в часі, налаштовувати параметри звуку тощо.

Часова шкала

У програмі Movie Maker фільми монтують за допомогою часової шкали (рис. 4.6), розміщуючи на ній відео- та аудіокліпи, переходи між кліпами, назви і титри. Часову шкалу поділено на п'ять доріжок, кожна з яких призначено для відображення мультимедійних даних певного типу:

- на доріжці **Видео** (Відео) розміщують відеоряд;
- доріжку **Переход** (Перехід) призначено для розміщення відеоефектів;
- на доріжці **Звук** міститиметься звуковий супровід відеоряду;



- на доріжці **Звук или музика** (Звук чи музика) можна розмістити другий аудіоряд (наприклад, фонову музику), який накладатиметься на основний аудіоряд фільму;
- доріжку **Наложение названия** (Накладання назви) призначено для розміщення назв і титрів, що відобразатимуться поверх відеоряду.

Перші три доріжки стосуються відеоряду, а тому їх можна відображати як одну доріжку, якщо клацнути кнопку поряд із назвою доріжки **Видео** (Відео).

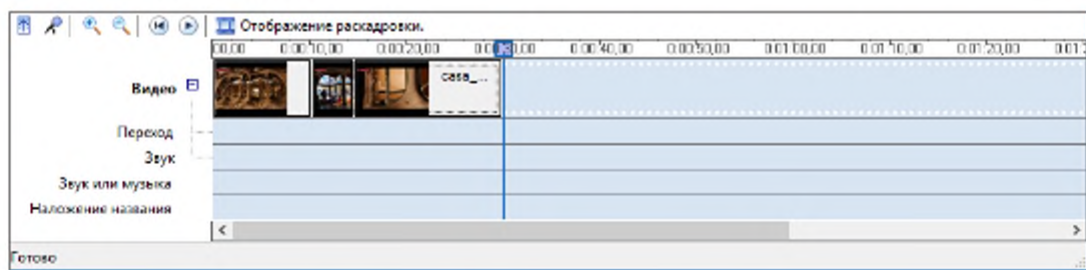


Рис. 4.6. Часова шкала

Зверху на часовій шкалі розташовано лінійку позначок часу у форматі **години : хвилини : секунди, соті частки секунди** (наприклад, 00:02:50,00). Блакитним прямокутником на ній позначено *головку відтворення*; пов'язана з нею вертикальна синя лінія вказує, який момент відеокліпу в цей час відтворюється на моніторі. Головку можна перетягувати за допомогою миші, переглядаючи в такий спосіб окремі кадри відеоролика.

Будь-який об'єкт — кліп, відеоперехід або накладена назва — зображується на доріжці часової шкали як прямокутник, ліва та права межі якого відповідають часу початку та часу завершення його відтворення. Об'єкти, розташовані на різних лінійках, але на одній вертикалі, відтворюватимуться одночасно. Відрізки часу, в які не відтворюється жодного вмісту з певної доріжки, позначаються на ній синім кольором.

Операції з кліпами

Розглянемо, які операції можна виконувати над кліпами за допомогою часової шкали.

Додавання кліпів до часової шкали. Щоб розмістити кліп на часовій шкалі, його потрібно перетягнути з області вмісту на відповідну доріжку. Кліпи додаткового аудіоряду та назви можна розміщувати в довільних точках часової шкали, а відеокліпи — лише один за одним (відеоряд не може містити пустот).

Виділення кліпів. Якщо клацнути прямокутник об'єкта на часовій шкалі, його буде виділено рамкою та білим кольором.



Переміщення кліпів. Якщо встановити курсор всередині виділеного кліпу, він набуде форми руки, тоді кліп можна переміщувати доріжкою часової шкали вправо чи ліво.

Видалення кліпів. Щоб видалити виділений кліп з часової шкали, достатньо натиснути клавішу **Delete**.

Відтинання початку або кінця кліпу. Зверніть увагу також на праву та ліву межі виділеного прямокутника: на них у вигляді чорних трикутників відображаються *маркери монтажу* (рис. 4.7). У разі наведення на маркер монтажу курсор набуває вигляду подвійної червоної стрілки — це означає, що маркер можна переміщувати. Якщо його перетягнути всередину прямокутника, кліп буде відігнуто, якщо назовні — розширено, але він не перевищуватиме початкової величини. Ці операції виконують також за допомогою команд меню **Кліп** (**Кліп**), про що йтиметься в підпункті «Точний монтаж кліпів».

Поділ кліпу на частини. Щоб поділити кліп на дві частини, слід встановити головку відтворення на точку поділу та виконати команду **Кліп** ▶ **Разделить** (**Кліп** ▶ **Поділити**).

Об'єднання кліпів. Щоб об'єднати два послідовно розміщених кліпи, їх слід виділити, утримуючи клавішу **Shift**, і виконати команду **Кліп** ▶ **Объединить** (**Кліп** ▶ **Об'єднати**).

Ці операції є базовими. Застосовуючи їх, можна виконувати складніші завдання з обробки відеоряду. Наприклад, щоб вставити всередину одного кліпу інший, слід перший кліп розділити, а другий перемістити в точку поділу.

Точний монтаж кліпів

Щоб фільм виглядав як єдине ціле, його фрагменти потрібно припасовувати з точністю до сотих часток секунди, або одного кадру. Головне завдання в цьому випадку — встановити головку відтворення саме на той кадр, який потрібно. З цією метою застосовують низку операцій.

Збільшення та зменшення масштабу часової шкали (кнопки зліва від часової шкали). Масштаб часової шкали збільшують, щоб «розтягнути» на всю часову шкалу невеликий (10 – 15 с) часовий відрізок, і зменшують, коли на шкалі потрібно вмістити більший проміжок часу.

Автоматичне відтворення фільму (кнопка зліва від часової шкали або така сама кнопка на програвачі). Виконується з того моменту, на який вказує головка відтворення.

Пауза (кнопка на програвачі). Зупиняє зображення на поточному кадрі та припиняє відтворення аудіоряду.

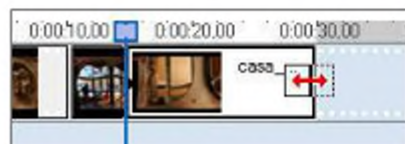


Рис. 4.7



Зупинення (кнопка зліва від часової шкали або кнопка на програвачі). Встановлює головку відтворення на початок запису.

Покадрове прокручування (кнопки на програвачі).

Прокручування кліпів (кнопки на програвачі). Застосовують для переходу до кліпу, який є попереднім до виділеного кліпу та наступним після нього.

Коли головку відтворення встановлено на потрібний кадр, ви можете:

- поділити на частини кліп у поточному кадрі (команда **Клип** ▶ **Розділити** (Кліп ▶ Поділити));
- відітнути частину кліпу від початку до поточного кадру (команда **Клип** ▶ **Задати початкову точку монтажу** (Кліп ▶ Задати початкову точку монтажу) або клавіші **Ctrl+Shift+I**);
- відітнути частину кліпу від поточного кадру до кінця (команда **Клип** ▶ **Задати кінцеву точку монтажу** (Кліп ▶ Задати кінцеву точку монтажу) або клавіші **Ctrl+Shift+O**).

ВПРАВА 19



Завдання. Підготувати відеофрагменти для відеокліпу, який ви створюватимете в практичній роботі №6. Для цього з відео у форматі AVI, на яких показані витвори архітектора А. Гауді, створити два відео тривалістю 1 хвилину кожне.

1. Запустіть програму Windows Movie Maker та імпортуйте в неї файл **casa_batllo.avi**. Переконайтеся, що ліворуч у вікні відображається панель завдань і скористайтеся посиланням **Імпорт відео** (Імпорт відео). Дочекайтеся, поки відео завантажиться і буде поділено на кліпи.
2. Створіть збірку **Будинок Батло**. Натисніть кнопку **Сборники** (Збірки), щоб відобразити дерево збірок. Створіть нову збірку з вказаним іменем.
3. Переглядайте кліпи та перетягуйте ті, що вам сподобаються, у збірку. (Для перегляду кліп слід виділити і натиснути кнопку відтворення в програвачі.)
4. У такий самий спосіб створіть збірку **Парк Гуель**, завантаживши файл **park_guel.avi**.
5. Створіть відеоролик, на якому показаний будинок Батло. Для цього переконайтеся, що в нижній частині вікна програми відображається часова шкала, відкрийте папку відповідної збірки, виділіть усі кліпи та перетягніть їх на лінійку **Відео** (Відео).
6. За потреби обітніть деякі кліпи. Користуйтеся для цього маркерами монтажу.
7. Збережіть відеоролик у файлі, обравши для нього формат WMV.
8. Видаліть фільм з часової шкали та створіть відео для парку Гуель.

**Висновки**

Редагують мультимедійні документи (аудіо- та відеофайли) за допомогою спеціальних програм, одна з них — редактор Windows Movie Maker.

Документ редактора Windows Movie Maker називається *проектом* і відкривається лише в цій програмі.

У проекті можна працювати з відеозаписами, аудіозаписами, зображеннями, а також з відеоефектами, переходами, назвами і титрами.

Загальний порядок роботи над фільмом такий: скласти план фільму, імпортувати в середовище програми всі необхідні вихідні матеріали, виконати монтаж і налаштування параметрів фільму, записати його у файл, на компакт-диск або опублікувати в Інтернеті.

**Контрольні запитання та завдання**

1. Які є способи додавання мультимедійних даних до проектів Movie Maker?
2. Чим проект Movie Maker відрізняється від готового фільму, створеного в середовищі цієї програми?
3. Як саме відеофайл поділяється на кліпи під час його імпорту в Movie Maker?
4. Що станеться в разі переміщення аудіофайла, який використовувався у проекті Movie Maker, в інше місце?
5. Які доріжки містить часова шкала в Movie Maker?
6. Які дії можна виконувати з кліпом, виділеним на часовій шкалі?

**Питання для роздумів**

1. Чому доріжка для додаткового аудіозапису є, а доріжки для додаткового відеозапису немає?
2. Коли недоцільно застосовувати автоматичний поділ на кліпи файла, що імпортується?
3. Чи можна встановити початок відтворення вмісту мультимедійного об'єкта точно на 7,00 с? Якщо ні, то чому? Якщо так, то як це зробити?



Завдання для досліджень

1. З'ясуйте, за допомогою яких клавіш фільм перемотують на один кадр уперед та на один кадр назад.
2. Імпортуйте в проект Movie Maker аудіо- чи відеоряд із зовнішнього джерела, наприклад із смартфона.
3. На панелі завдань є команда **Монтаж автофільма** (Монтаж автофільму). Дізнайтеся, для чого вона призначена та чим різняться типи автофільмів.
4. З'ясуйте, як встановити для фільму такі параметри:
 - авторство;
 - відношення сторін кадру (4:3 чи 16:9);
 - необхідність автоматичного завантаження кодеків з Інтернету в разі їх відсутності.

§18

Редагування мультимедійних даних

18.1.

ОФОРМЛЕННЯ ВІДЕОРЯДУ

Зверніть увагу на те, як змонтовано телепередачі: один сюжет замінюється іншим не раптово, а з якимось ефектом (наповзання, розчинення тощо), під час інтерв'ю в субтитрах вказують інформацію про людину, а починається передача із заставки. Застосування цих та інших засобів можна назвати «оздобленням», адже вони надають телепередачі привабливого вигляду, справляють на глядача приємне враження. Мінімальний набір засобів оздоблення фільмів є і в Movie Maker.

Мультимедійний редактор Movie Maker надає в наше розпорядження засоби, які допоможуть зробити відеозображення привабливішим. Це відео ефекти, що їх застосовують до кліпів, переходи, які дозволяють спеціальним чином накладати кліпи, та назви і титри — текстова інформація, що може відображатися на відеоряді.

Відео ефекти

У програмі Windows Movie Maker передбачено близько трьох десятків відео ефектів, які можна застосовувати до окремих кліпів: затемнення, висвітлення, змінення контрастності, розмиття контурів зображення тощо. Щоб відобразити їх значки в області вмісту, в розкритому списку на панелі інструментів слід вибрати пункт **Видеоэффекты** (Відео ефекти). Після подвійного клацання значка ефекту він відтворюється у вікні монітора.



Щоб додати до кліпу ефект, його значок потрібно перетягнути на прямокутник кліпу на часовій шкалі (цей прямокутник буде позначено зірочкою). Інший спосіб — клацнути правою кнопкою миші на прямокутнику кліпу і вибрати в контекстному меню команду **Відеоэффекты** (Відео ефекти). В результаті буде виведено діалогове вікно **Добавление или удаление видеоэффектов** (Додання чи видалення відео ефектів), зображене на рис. 4.8. З його допомогою можна не лише додавати, а й видаляти ефекти, що застосовуватимуться до кліпу.

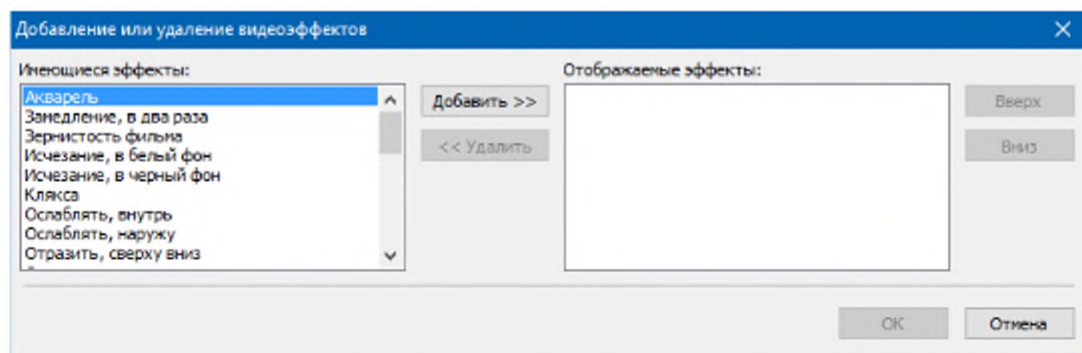


Рис. 4.8. Вікно **Добавление или удаление видеоэффектов**

Мабуть, найпопулярнішими є ефекти **Появление, из белого** (Поява, з білого) та **Появление, из черного** (Поява, з чорного). Перший з них відтворюється на початку показу фрагмента і полягає в поступовому проявленні зображення на білому або чорному тлі. Ефекти **Исчезание, в белый фон** (Висвітлення до білого тла) та **Исчезание, в черный фон** (Затемнення до чорного тла) поступово зменшують насиченість зображення до досягнення білого або чорного кольору тла і відтворюються наприкінці показу фрагмента.

Переходи

У Movie Maker можна обрати також спосіб заміни одного кліпу іншим, який називають *переходом*. Для цього потрібно вибрати в розкритому списку на панелі інструментів пункт **Видеопереходы** (Відеопереходи), після чого в області вмісту відобразяться значки переходів. Щоб переглянути перехід, його значок слід двічі клацнути, а щоб застосувати до фрагментів фільму — перетягнути на лінійку **Переход** (Перехід) в позицію, де завершується один відеокліп і починається інший. У результаті на цій лінійці відобразиться прямокутник, а межі кліпів перетинатимуться. Тобто під час переходу відтворюватимуться два кліпи одночасно й один поступово замінюватиме інший.

Тривалість переходу зазвичай становить 1,25 с, але час його відтворення можна змінювати, розтягуючи або стискаючи прямокутник.



Назви та титри

Будь-який гарно оформлений фільм містить певну текстову інформацію — назву фільму, ім'я автора або режисера, фільми іноземною мовою супроводжуються субтитрами. У Movie Maker текст, що виводиться наприкінці фільму, називають *титрами*, а будь-який інший текст у фільмі — *назвами*.

Щоб додати до кліпу назви або титри, потрібно виділити його на часовій шкалі та в розділі **Монтаж фільма** (Монтаж фільму) на панелі завдань клацнути посилання **Создание названий и титров** (Створення назв і титрів). Це посилання запускає майстер назв і титрів. У першому вікні майстра вибирають місце розташування текстової інформації, а в другому — вводять сам текст і задають параметри його шрифту та анімації. Увівши текст і встановивши всі параметри, слід клацнути посилання **Готово, добавить название в фильм** (Готово, додати назву до фільму).

Назва може займати не більше двох рядків, а титри — багато рядків, адже в кінці фільму, як правило, прокручується велика кількість текстової інформації: назва кіностудії, імена режисера, сценариста, оператора, всіх акторів тощо. Скільки буде рядків у назві та як саме відобразитиметься текст назви або титрів, залежить від обраного ефекту анімації.

Найчастіше використовуються такі різновиди анімації назв, як **Субтитр** (відображення назви в нижній частині екрана, по центру, невеликим шрифтом), **Заголовок новостей** (Заголовок новин), **Серпантин** (відображення рухомого рядка новин). Також є ефектні нестандартні анімації, такі як **Движущиеся надписи, слоями** (Написи, що рухаються, шарами).

Якщо додавати назву на початок фільму або титри в його кінець, вони розмістяться на доріжці відео до або після наявного відеоряду і відобразатимуться на чорному тлі. Щоб відобразити назву поверх наявного зображення, її прямокутник слід перетягнути на доріжку **Наложение названия** (Накладання назви) і розмістити в потрібному місці часової шкали. Якщо ж додати назву до виділеного кліпу, вона розміститься на доріжці **Наложение названия** (Накладання назви) автоматично.

18.2.

КЕРУВАННЯ АУДІОРЯДОМ


Відеозапис, що імпортується в середовище Movie Maker або записується із зовнішнього пристрою, як правило, має власний звуковий ряд. Він відображається на третій зверху доріжці часової шкали, що називається **Звук**. Ви можете додати до фільму ще один звуковий ряд, наприклад, фонову музику або голосовий коментар. Об'єкти цього звукового ряду розміщуються на четвертій зверху доріжці **Звук или музыка** (Звук чи музика), і ними можна керувати так само, як відеокліпами: об'єднувати, поділяти на частини, обтинати тощо. Більше того, звукові кліпи можна



розміщувати в довільних точках часової шкали, в той час як відеокліпи — лише один за одним.

Є два способи створення додаткового аудіоряду.


Імпорт аудіофайла. Звуковий файл імпортують за допомогою послання **Імпорт звука или музики** (Імпорт звуку або музики) на панелі завдань, а потім, як і будь-який кліп, перетягують з області вмісту на часову шкалу.

Запис голосового коментаря. Додатковий аудіоряд записують з мікрофона. Ця функція активізується кнопкою  **Шкала времени коментаря**, розміщеною зліва від часової шкали.

Важливо розуміти головну відмінність між звуковим супроводом відеоряду та додатковим аудіорядом: другий, на відміну від першого, незалежний від вмісту шкали відео. Ви можете як завгодно поділяти на кліпи вміст шкали **Звук или музыка** (Звук чи музика), але кліпи на шкалі **Звук** можуть бути лише ті самі, що й на шкалі **Відео** (Відео); їх навіть неможливо окремо виділити. Інакше кажучи, аудіоряд, імпортований в складі відеозапису, *синхронізовано* з відеорядом, а додатковий аудіоряд — ні; його синхронізація відбувається лише під час запису проекту в мультимедійний файл.

Проте гучністю як звукового супроводу відео, так і додаткового аудіоряду керують однаково — з допомогою команд контекстного меню аудіокліпу (рис. 4.9):

- щоб вимкнути звук кліпу, треба вибрати команду **Выключить** (Вимкнути);
- для стишення звуку кліпу або збільшення його гучності — команда **Громкость** (Гучність);
- щоб гучність звуку поступово наростала протягом кількох перших секунд відтворення кліпу, потрібно встановити прапорець **Появление** (Поява);
- щоб звук поступово стишувався наприкінці кліпу, треба встановити прапорець **Исчезание** (Зникнення).

Коригуючи гучність звуку окремих кліпів, можна досягати цікавих ефектів, наприклад, наближення голосів (прапорець **Появление**) або припинення фонові музики на той час, поки в основному аудіоряді озвучується важливий текст. Можна також визначати співвідношення гучності основного і додаткового аудіоряду в цілому. Для цього призначено кнопку  **Установить уровень звука** (Встановити рівень звуку), розташовану зліва від часової шкали.

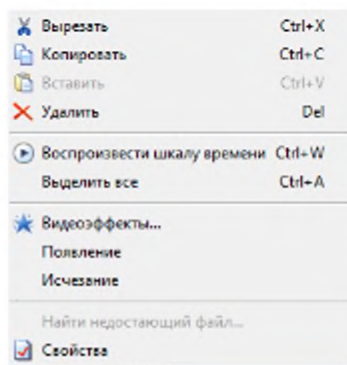



Рис. 4.9. Контекстне меню аудіокліпу



Виконати додаткове

завдання, створивши відеофільм «Світ моїх захоплень», ви зможете на диску. 

**Висновки**

Для оформлення відеоряду в Movie Maker можна застосовувати відеоєфекти, переходи між кліпами, а також додавати до фільму назви і титри.

У Movie Maker текст, що відображується наприкінці фільму, називають *титрами*, а будь-який інший текст у фільмі — *назвами*. Титри і назви можна розміщувати на окремій часовій доріжці й накладати на відеоряд.

Відеозапис має власний звуковий ряд, який відображається на доріжці часової шкали **Звук**. Користувач може додати до фільму ще один звуковий ряд, який розміщуватиметься на доріжці **Звук или музика**.

Додатковий аудіоряд не залежить від відеоряду, і його можна поділяти на кліпи в довільний спосіб. Натомість звуковий супровід відеоряду містить ті самі кліпи, що й сам відеоряд.

**Контрольні запитання та завдання**

1. Чим відеоєфекти відрізняються від переходів?
2. У чому полягає відмінність між назвами і титрами?
3. Від чого залежить, як буде виводитися назва — у два рядки чи в один?
4. Яка з доріжок — **Звук** чи **Звук или музика** — є незалежною від доріжки **Відео**?
5. Які існують можливості щодо керування гучністю аудіокліпу?
6. Як зменшити гучність звукового супроводу відеоряду в усьому фільмі, одночасно підвищивши гучність додаткового аудіоряду?

**Питання для роздумів**

1. Чи може доріжка **Звук** бути порожньою в той момент часу, коли на доріжці **Відео** відтворюється кліп? Якщо так, поясніть, у якому випадку; якщо ні, поясніть, чому.
2. Чи може бути навпаки — доріжка **Звук** зайнята кліпом, а доріжка **Відео** в цей момент порожня?
3. Чому до кліпів додаткового аудіоряду не можна застосовувати переходи?
4. Як до відеокліпу приєднати кілька додаткових звукових рядів?



Завдання для досліджень

1. У який спосіб під час відеопереходу звуковий ряд одного кліпу замінюється звуковим рядом іншого кліпу? Чи залежить цей спосіб від типу переходу?
2. Замініть фонову музику у фільмі, створеному в практичній роботі, власним коментарем, записаним з мікрофона.
- 3*. Створіть відеоролик про свою сім'ю або друзів, знявши матеріал на камеру чи мобільний телефон, а потім змонтувавши його в Movie Maker.

ПРАКТИЧНА РОБОТА №6

Створення відеокліпу. Додавання відеоефектів, налаштування часових параметрів аудіо- та відеоряду

Завдання. Створити відео тривалістю приблизно 2 хвилини, яке демонструє роботи іспанського архітектора А. Гауді — будинок Батло та парк Гуель. Назва відео «Неповторні споруди Антоніо Гауді» має розташовуватися на тлі титульного зображення, після назви глядач має бачити портрет архітектора. Далі відео про будинок Батло, на його початку — назва «Будинок Батло», після нього — відео про парк Гуель також з назвою на початку. Між відео мають бути переходи. Під час демонстрації повинні звучати дві музичні мелодії. Завершити відео слід титрами з відомостями про автора.

Заготовки для виконання цієї практичної роботи можна завантажити з папки **Files** на диску.

1. Запустіть програму Windows Movie Maker та створіть новий проект.
2. Імпортуйте в проект два відеокліпи, титульне зображення, портрет А. Гауді та аудіофайли супроводу.
3. Перетягніть на відеодоріжку часової шкали фонове зображення, портрет і два відеокліпи.
4. На музичну доріжку перетягніть перший аудіозапис. Його початок має збігатися з початком кліпу. Обітніть його, щоб він закінчувався в кінці першого відео. Перетягніть другий аудіозапис, розмістіть його після першого.
5. Створіть назви в такий спосіб. Виділіть відповідний відеокліп та клацніть посилання **Создание названий и титров** (Створення назв і титрів). Виберіть команду додавання назви перед кліпом, введіть текст і встановіть параметри шрифту. Коли назва відобразиться на доріжці часової шкали, відкоригуйте тривалість її відтворення (2 – 3 с).
6. Створіть переходи між кліпами.
7. Додайте до початку назву відео, а в кінець — титри.
8. Перегляньте відео, скорегуйте його в разі потреби і збережіть відео у WMV-файлі.



§19

Мультимедіа у Вебі

19.1.

РОЛЬ І МІСЦЕ МУЛЬТИМЕДІА У ВЕБІ



Історичні відомості
про створення
мультфільмів ви зможете
дізнатися на диску.

Досі ми розглядали технології мультимедіа, які передбачають, що мультимедійні дані зберігаються локально, редагуються і відтворюються за допомогою програм, встановлених на комп'ютері.

Проте мультимедіа давно є невід'ємною частиною Веб-простору. Сьогодні на сторінках сайтів ми бачимо не лише графіку, а й відео та анімацію. Мережа Інтернет пропонує нам онлайнві ігри, віртуальні музеї, потужні сервіси для розміщення і перегляду відео, аудіо і фотографій.

Усе це стало можливим насамперед завдяки появі високошвидкісних інтернет-підключень, зокрема й мобільних (3G, 4G). А з появою смартфонів і веб-програм створювати й розповсюджувати мультимедійний контент може кожен — для цього потрібен пристрій з функціями диктофона і камери та підключення до Інтернету.

Важливими складовими мультимедіа є відео, аудіо та анімація. Розглянемо, які технології застосовуються для їх відтворення в браузері.

19.2.

ПОТОКОВЕ АУДІО ТА ВІДЕО

Мультимедійні дані не набули б такого поширення, якби не поява технологій *потокowego відео* та *аудіо*, що дають змогу передавати відео-дані в режимі реального часу через Інтернет. Саме тому можна переглядати відео з веб-камер, відеоролики на веб-сторінках і користуватися таким ресурсом, як YouTube.

Основна перевага потокowego відео та аудіо полягає в тому, що користувач не повинен чекати повного завантаження файлу на свій комп'ютер — дані передаються незалежними блоками безперервним потоком і відтворюються в міру надходження на комп'ютер.

19.3.

РОЗМІЩЕННЯ АНІМАЦІЇ, ВІДЕО ТА АУДІО НА ВЕБ-СТОРІНКАХ

До недавнього часу для створення веб-анімації та підтримки мультимедіа сайтами переважно застосовувалась технологія Adobe Flash. Вона надає можливість поєднувати векторну і растрову графіку, створювати



анімацію, інтерактивні елементи, підтримує потокове відтворення аудіо та відео. З допомогою цієї технології можна створювати відеозображення різного рівня складності, проте найбільше вона придатна для анімаційних роликів на основі векторної графіки.

Flash-кліпи зберігаються у файлах формату SWF, і для їх перегляду потрібен спеціальний програвач Adobe Flash Player. Він встановлюється на комп'ютер як окрема програма або як плагін браузера.

Відтворення відео на веб-сторінках у цьому випадку забезпечується в такий спосіб. Відео подається у форматі Flash Video (FLV), а на веб-сторінці робиться посилання на нього. Браузер для відтворення відео запускає плагін чи допоміжну програму. Звичайно, якщо їх на комп'ютері немає, відео не відтворюється.

Технологія Adobe Flash має низку недоліків, серед яких «дірки» у безпеці, велике навантаження на ЦП, недостатній контроль помилок, що призводить до порушення роботи всього браузера. Окрім цього, Flash не підтримують мобільні пристрої (смартфони).

Тому сьогодні у Всесвітній павутині технологію Flash поступово витісняє мова розмітки веб-сторінок HTML 5, яка забезпечує покращену підтримку мультимедіа-технологій та дозволяє створювати складні веб-програми. У разі її використання браузер відтворює відео, не звертаючись до Flash-плеєра.

ВПРАВА 20



Завдання. Дослідити відмінність у відтворенні відео в разі розміщення посилань на нього в документі Word та на веб-сторінці.

1. Запустіть програму Word, створіть текстовий документ, що містить посилання на відео **casa_batllo.mp4** (або інше відео, яке вам вкаже вчитель). Для створення посилання перейдіть на вкладку **Вставка** і скористайтеся командою **Гіперпосилання**. Збережіть файл.
2. Клацніть посилання при натиснутій клавіші **Ctrl**. Запуститься встановлений на комп'ютері медіапрогравач, і відео почне відтворюватися в ньому.
3. Збережіть документ як веб-сторінку і відкрийте цю сторінку в браузері. Клацніть посилання на відео. Воно почне відтворюватися в браузері.
4. Поміркуйте, чому спостерігається така відмінність у роботі текстового процесора та браузера.



19.4.

ОНЛАЙНОВІ КОНВЕРТОРИ ВІДЕО ТА АУДІО, GIF-РЕДАКТОРИ

Сьогодні он-лайн пропонують велику кількість допоміжних сервісів для тих, хто хоче працювати з аудіо та відео. Якщо відео або аудіо, яке ви створили або завантажили, має формат, який не підходить для обробки чи не відтворюється, ви можете легко конвертувати його он-лайн.

Онлайнві конвертори працюють за однаковою схемою. Потрібно обрати цільовий формат, вказати файл на своєму комп'ютері та за потреби додаткові параметри. Конвертор завантажує файл на свій сайт, конвертує і пропонує користувачеві вікно для завантаження файлу в цільовому форматі на комп'ютер. Звичайно, робити цю операцію слід за наявності швидкісного підключення. Також треба врахувати, що в конвертора можуть бути встановлені обмеження на обсяг файлів, які обробляються безкоштовно.

Без проблем можна знайти також онлайнві редактори GIF-файлів, що дозволяють створювати GIF-файли не лише із зображень, а й з відео.

ВПРАВА 21

САМОСТІЙНО



Завдання. Створити анімацію з низки фотографій, скориставшись онлановим сервісом для створення GIF-анімації.

19.5.

РОБОТА З ІНТЕРНЕТ-ДЖЕРЕЛАМИ МУЛЬТИМЕДІЙНОГО КОНТЕНТУ

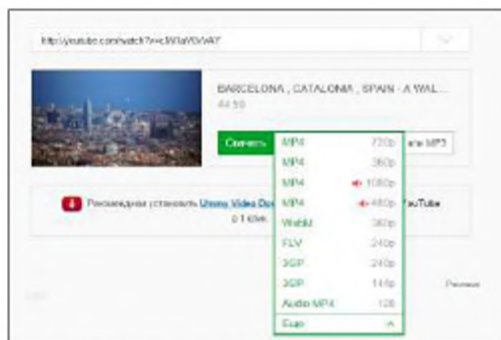


Рис. 4.10. Інтерфейс для завантаження аудіо та відео з YouTube

Щоб створювати аматорське відео, можна не лише знімати фрагменти самостійно, а й брати відео, аудіо і графіку з відкритих джерел, таких як колекції зображень, музичні колекції та відеохостинги. Звичайно, водночас слід враховувати можливі проблеми з порушенням авторських прав. Розглянемо, як завантажити відео та аудіо з YouTube.

1 спосіб. Відкрийте на YouTube сторінку з відео, яке вам потрібно завантажити. Додайте символи ss перед словом youtube в адресі ресурсу:

<https://www.youtube.com/watch?v=ewe1ioXIuXI>

<https://www.ssyoutube.com/watch?v=ewe1ioXIuXI>



Натисніть клавішу **Enter** і ви перейдете на сайт **savefrom.net**. Тут ви зможете завантажити об'єкт на свій комп'ютер, обравши його формат (рис. 4.10).

2 спосіб. Відкрийте на YouTube сторінку з відео, яке вам потрібно завантажити. Скопіюйте адресу об'єкта, відкрийте сайт **https://savedeo.com**, вставте в поле на сторінці цього сайту адресу і натисніть кнопку **Download**. Вам буде запропоновано різноманітні формати для зберігання об'єкта на вашому комп'ютері.



Дізнатися більше
про аніме ви
зможете на диску

ВПРАВА 22**САМОСТІЙНО**

Завдання. Створити в програмі Movie Maker слайд-шоу до улюбленої музичної композиції. Аудіозапис можна завантажити з YouTube, а фотографії знайти в Інтернеті.

**Висновки**

Завдяки широкій підтримці мультимедіа у Вебі можна, користуючись лише браузером, переглядати анімацію та відео, відтворювати звук, грати в онлайн-ігри.

З появою смартфонів і веб-програм створювати й розповсюджувати мультимедійний контент може кожен — для цього потрібен пристрій з функціями диктофона і камери та підключення до Інтернету.

Донедавна для створення веб-анімації і підтримки мультимедіа сайтами переважно застосовувалась технологія Adobe Flash, сьогодні її витісняє мова HTML 5, яка забезпечує покращену підтримку мультимедіа-технологій та дозволяє створювати складні веб-програми.

**Контрольні запитання та завдання**

1. Наведіть приклади використання мультимедіа у Вебі.
2. Яка характерна її особливість?
3. Що таке потокове відео та аудіо?
4. У який спосіб на веб-сторінках розміщують відео та аудіо?
5. Опишіть, як працюють онлайн-конвертори.
6. Опишіть способи завантаження відео та аудіо з YouTube.

**Питання для роздумів**

1. Чи завжди технологія, яка витісняє попередню, є кращою?



Завдання для досліджень

- 1*. Знайдіть інформацію про GIFV. Скористайтеся можливостями цієї технології.
- 2*. З'ясуйте, що таке Adobe Shockwave.
- 3*. Дослідіть, які можливості надає сервіс YouTube **Об'єктив** (Об'єктив).

§20

Розміщення аудіо- та відеофайлів в Інтернеті

20.1.

ПЛАТФОРМИ ДЛЯ РОЗМІЩЕННЯ ФОТОГРАФІЙ ТА ВІДЕО

Сьогодні в еру бурхливого розвитку платформ хостингу контенту, соціальних мереж і мобільних пристроїв розміщення фотографій та відео в Інтернеті стало повсякденною справою багатьох користувачів. На нашу думку, немає сенсу наводити в підручнику перелік популярних сьогодні платформ, оскільки з часом цей перелік застаріє.

Проте загальні напрямки розвитку можна прослідкувати. Візьмемо, наприклад, фотографію. У доцифрову еру фотографії друкували на папері та впорядковували в альбоми. Потім, з появою перших цифрових камер, кількість знімків значно збільшилася, їх перестали роздруковувати і переважно зберігали на комп'ютері або записували на компакт-диски. Упорядкування знімків стало вимагати багато часу, і більшість людей займалося цим лише час до часу.

Невдовзі з'явилися веб-хостинги для фотографій, і їх почали зберігати в он-лайн. Але проблема впорядкування фотоколекцій на локальних комп'ютерах залишилася. Наступний етап розвитку технологій — поява смартфонів і служб, що забезпечують синхронізацію даних на різних пристроях (настільному ПК, смартфоні, ноутбучі). Тепер можна, користуючись власною адресою електронної пошти, приєднати до такої служби всі свої пристрої та синхронізувати дані на них.

Одна з таких служб — <https://photos.google.com>, яка може автоматично копіювати знімки з вашого смартфона на інші ваші пристрої. Також вона дозволяє в різноманітний спосіб упорядковувати колекції знімків. Звичайно, обсяг наданого дискового простору в он-лайн не безмежний, і періодично доводиться переміщувати впорядковані колекції знімків до основної колекції.



20.2.

ВІДЕОХОСТИНГ YOUTUBE

Відеохостинг YouTube був створений трьома колишніми працівниками компанії PayPal 2005 року. Приблизно через рік його продали компанії Google, і зараз він діє як їхній дочірній підрозділ. Сайт надає користувачам можливість завантажувати, переглядати, оцінювати, коментувати та ділитися відеозаписами. Сьогодні він є одним з перших за кількістю відвідувачів.

YouTube має досить розвинену довідкову систему, яка допомагає ефективно користуватися цією платформою. Посилання для виклику довідки міститься в кінці сторінки.

Звертаємо вашу увагу на політику YouTube щодо захисту авторських прав. Заборонено завантажувати на сайт відео, зміст якого порушує закон США про авторське право. Працівники YouTube часто вимикають звук у таких відео, а потім взагалі видаляють їх із сайту.



Дізнатися більше

про секрети створення анімації від Disney ви зможете на диску



Контрольні запитання та завдання

1. Які служби для розміщення відеоконтенту ви знаєте?
2. Чи користувалися ви ними? Якщо так, то опишіть роботу з цими службами.
3. Як ви впорядковуєте свою колекцію знімків?
4. Як працює служба, що дозволяє синхронізувати ваші дані на різних пристроях?
5. Опишіть, як ви користуєтеся YouTube.



Питання для роздумів

1. У якому напрямку, на вашу думку, розвиватимуться платформи відеохостингу?



Завдання для досліджень

1. Знайдіть інформацію про відеохостинг VIMEO.



ПРАКТИЧНА РОБОТА №7

Розміщення аудіо- та відеоматеріалів в Інтернеті

Завдання. Завантажити на YouTube відео, створене під час виконання практичної роботи.



Перевірити

свої знання,
пройти тест і розгадати
кросворд ви зможете
на диску.

1. Відкрийте сторінку YouTube в браузері на комп'ютері. Увійдіть у свій акаунт.
2. Натисніть кнопку **Додати відео** ліворуч у верхній частині сторінки.
3. Задайте тип доступу: відкритий доступ, доступ за посиланням, обмежений доступ.
4. Виберіть файл для завантаження. Під час завантаження введіть основну інформацію про відео і налаштуйте додаткові параметри. Наприклад, тут можна вказати, чи отримають повідомлення про нове відео ваші передплатники.
5. Щоб додати відео на YouTube, натисніть **Опублікувати**. У разі вибору відкритого доступу, воно відразу публікується на YouTube. В інших випадках натисніть або **Готово**, щоб просто завантажити ролик на сайт, або **Поділитися**, щоб відкрити до нього доступ.
6. Поки ви не натиснете кнопку **Опублікувати**, відео буде доступне тільки для вас. Ви зможете в будь-який момент поділитися доданим відео, змінивши його вигляд у **Менеджері відео**.
7. Коли завантаження буде завершено, ви отримаєте посилання на відео електронною поштою. Ви можете відразу переслати цього листа друзям, щоб вони подивилися нове відео.
8. Оберіть збережене як відео слайд-шоу з улюбленою музичною композицією та завантажте його на YouTube.

Розділ 5

Основи подійно-та об'єктно-орієнтованого програмування



§21

Мова програмування

21.1.

ПОНЯТТЯ МОВИ ПРОГРАМУВАННЯ

У цьому розділі ми розпочинаємо вивчення основ програмування. Вивчаючи мову програмування, ви зможете в майбутньому навчитися складати програми, що прискорюють обробку інформації в будь-яких галузях або навіть оволодіти професією програміста.

Для початку пригадаймо, які розділи виділяють в інформатиці і як вони взаємопов'язані між собою.

Вивчення науки інформатики можна поділити на такі основні складові:

- *технічна* (ознайомлення з технічними засобами обробки інформації);
- *програмна* (ознайомлення з програмними засобами, що дозволяють обробляти інформацію);
- *алгоритмічна* (визначення послідовності дій, які слід вказати програмному засобу для обробки інформації).

Із схеми видно, що комп'ютер — це формальний виконавець, тобто він не думає про зміст команд. Комп'ютер лише виконує програму. Програмні засоби, з якими працює комп'ютер, призначені для розв'язання різноманітних завдань.



Історичні відомості
про мови програмування ви зможете дізнатися на диску.




Обробляючи інформацію за допомогою програмного засобу, користувач може самостійно виконати послідовність дій, а може записати цю послідовність дій на мові зрозумілій комп'ютеру та надати вказівку щодо її виконання. Таку мову ще називають мовою програмування.

Мова програмування — формалізована мова для запису алгоритмів, призначених для виконання за допомогою комп'ютера.



Дізнатися більше

про мови програмування та їх походження ви зможете на диску. 

Мова програмування (як і будь-яка інша мова) має три складові: алфавіт, синтаксис і семантику.

Алфавіт — фіксований для даної мови набір символів (літери, цифри, спеціальні символи).

Синтаксис — набір правил, що встановлюють, які комбінації символів є осмисленими реченнями на цій мові.

Семантика — визначає смислове значення речень у мові.

Комп'ютерна програма — запис алгоритму мовою програмування.

Незнання цих правил тягне за собою появу помилок у програмах, виправлення яких може викликати великі труднощі.

21.2.

СИСТЕМА ПРОГРАМУВАННЯ

Для того, щоб створити програму зрозумілу для виконання комп'ютером, потрібний комплекс програмних засобів, який називають системою програмування.

Система програмування — система для створення та редагування програм певною мовою програмування.

Перелічимо основні програми, що мають входити в систему програмування:

- текстовий редактор (для написання програми);
- транслятор (для перетворення програми в машинний код);
- засоби автоматизації зборки (вбудовані команди, бібліотеки тощо);
- налагоджувач (для перевірки та виправлення помилок програми).

З поняттям текстового редактора ви вже ознайомлені. Розглянемо, що таке транслятор та які види трансляторів можуть бути використані в системі програмування.

Оскільки комп'ютер має можливість обробляти інформацію подану лише в двійковому коді, то має існувати спеціальна програма, що перекладає зрозумілі для людини символи в символи зрозумілі машинам.

Транслятор — це програмний засіб, що здійснює перетворення програми з однієї мови програмування в програму на іншій мові програмування.

Транслятори бувають двох видів: інтерпретатори та компілятори.

Інтерпретатор перетворює та виконує програму по рядках, а компілятор читає програму цілком і перекладає її в машинний код, який потім і виконується.

Однією з проблем при введенні програми є допуск помилок. Комп'ютер не зможе виконати програму з помилками. Зазвичай всі мови програмування високого рівня містять систему пошуку помилок у програмі та виведення довідкової інформації для усунення цих помилок. Ці функції виконує *налагоджувач* системи програмування.



У цьому параграфі ми розпочали ознайомлення із найцікавішим, але і з найважчим розділом інформатики — програмуванням. На наступних уроках ви будете вивчати мову програмування Visual Basic.NET, що відноситься до групи програм об'єктно-орієнтованого програмування.



Висновки

Мова програмування — формалізована мова для запису алгоритмів, призначених для виконання за допомогою комп'ютера.

Комп'ютерна програма — запис алгоритму мовою програмування.

Мова програмування має три складові: алфавіт, синтаксис і семантику.

Система програмування — система для створення та редагування програм певною мовою програмування.

Транслятор — це програмний засіб, що здійснює перетворення програми з однієї мови програмування в програму на іншій мові програмування.

Інтерпретатор — перетворює та виконує програму по рядках.

Компілятор читає програму цілком та перекладає її в машинний код, який потім і виконується.

**Контрольні запитання та завдання**

1. Що таке мова програмування?
2. Що таке машинна мова?
3. Що таке мова асемблера?
4. Що таке мови високого рівня?
5. Як класифікують мови програмування?
6. Що таке система програмування? Із яких програм вона складається?
7. Що таке транслятор? Які види трансляторів ви знаєте?

**Питання для роздумів**

1. Що робить програміст? тестувальник?
2. Чому існує така велика кількість мов програмування?

**Завдання для досліджень**

1. Підготуйте повідомлення про будь-яку мову програмування (за-сновник мови, з якою метою створювалася мова, яке має призна-чення тощо).

§22**Основні поняття об'єктно-орієнтованого програмування****22.1.****ОБ'ЄКТИ. ВЛАСТИВОСТІ І МЕТОДИ ОБ'ЄКТІВ**

Об'єктно-орієнтоване програмування (ООП) широко використовується для створення різних прикладних програм, що мають графічний інтерфейс. Зокрема, програми MS Office були створені саме за допомогою об'єктно-орієнтованих мов, які дозволяють, як у конструкторі, з деякого набору елементів (програмних об'єктів) збирати робоче вікно будь-якої програми.

Основною одиницею в ООП є *програмний об'єкт*, який поєднує вла-стивості та дії, які з ним можна виконувати.

Для прикладу, розглянемо робоче вікно додатка MS Paint (рис. 5.1).

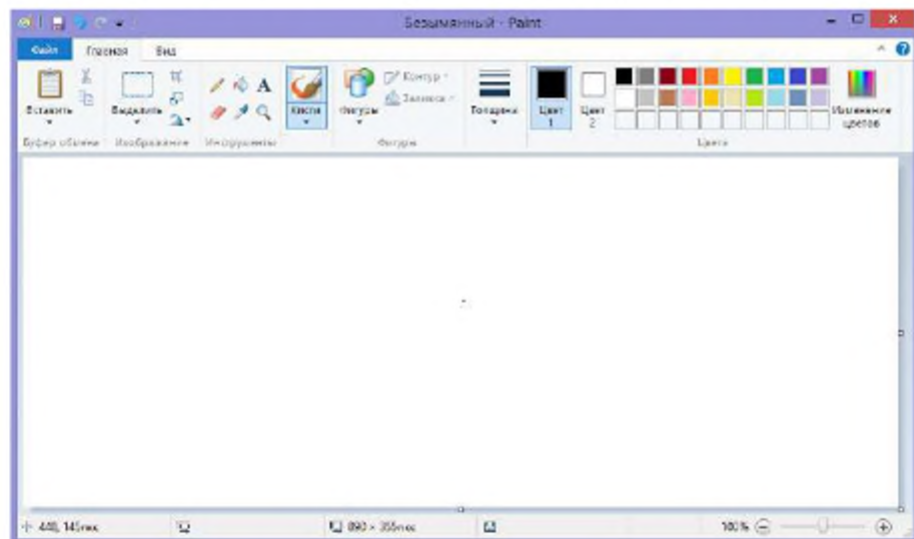


Рис. 5.1. Вікно програми MS Paint

У цьому додатку об'єктами є вікно, кнопки, вкладки, палітра, робоче поле тощо.

Програмні об'єкти мають певні *властивості*. Так, кнопки мають такі властивості: розмір, місце розташування на панелі інструментів, активність кнопки тощо. Крім властивостей, об'єкт також може виконувати певні дії, що називаються *методами*. Скажімо, одна кнопка дозволяє намалювати лінію, інша виконати виділення певного фрагмента. Отже, методи визначають дії, які може виконувати той чи інший об'єкт.

22.2.

ГРАФІЧНИЙ ІНТЕРФЕЙС КОРИСТУВАЧА

Працюючи за комп'ютером, ми використовуємо екранні вікна, текстові поля, кнопки, меню, вкладки тощо. Сукупність таких візуальних елементів, з якими взаємодіє користувач, називають *графічним інтерфейсом користувача*.

Графічний інтерфейс користувача — це візуалізований засіб взаємодії програми з користувачем.

Система програмування VB.NET дозволяє формувати таку візуальну складову без написання програмного коду за допомогою спеціальних засобів графічного редагування, що мають назву *елементи керування*. Саме тому таке програмування називають *візуальним*.

Елементи керування є об'єктами, що характеризуються властивостями і методами. До елементів керування, які використовуються для підтримки діалогу з користувачем при виконанні програми, відносяться меню, вкладки, текстові поля, командні кнопки, перемикачі, поля вибору, списки, смуги прокручування тощо. Більшість елементів інтерфейсу Windows призначені для організації введення даних користувачем та виведення результатів роботи програми на екран, папір або у файл.

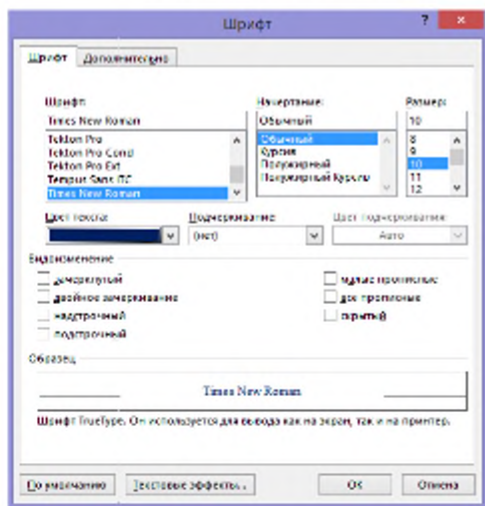


Рис. 5.2. Вікно Шрифт

Так на рис. 5.2 на прикладі діалогового вікна **Шрифт** показано використання деяких основних елементів керування, таких як текстові поля, написи, прапорці, списки, кнопки. Вікно **Шрифт**, як і більшість інших вікон, створене за допомогою об'єкта форма.

Форма — це об'єкт, який використовують для створення вікон програми. На формі розміщують усі інші елементи керування, що утворюють вікно програми. При створенні екранна форма є чистим робочим полем, на якому користувач може розміщувати різні елементи керування, а саме: кнопки, перемикачі, малюнки тощо.

Вміст вікна екранної форми охоплює:

- сукупність властивостей цього вікна з їхніми значеннями;
- сукупність об'єктів, які розміщує користувач у цьому вікні;
- сукупність властивостей цих об'єктів з їхніми значеннями.

Більшість програм на VB.NET складаються, як мінімум, з однієї екранної форми, що зовні нагадує звичайне вікно, з яким працює ОС Windows. Але екранна форма — це не тільки гарна картинка для виведення даних на екран. Вона також надає користувачеві можливість взаємодіяти з програмою за допомогою введення даних або активізації різних її функцій.

22.3.

ПОДІЇ, ОБРОБНИКИ ПОДІЙ

Для кожного з елементів керування користувач може передбачити деяку *подію*. Подією програми може бути дія мишки (клацання або подвійне клацання кнопкою, перетягування об'єкта, наведення курсору на об'єкт), натиснення певної клавіші на клавіатурі тощо. Для кожної події

користувач може передбачити різні способи реагування на них, що вимагає вже написання програмного коду за допомогою мови програмування.

Подія — це характеристика об'єкта, яка вказує, на яку дію користувача може реагувати об'єкт.

З будь-якою подією можна пов'язати певні операції оброблення даних. Для цього пишуть *процедуру* мовою програмування. Для кожної події, яка обробляється, складається окрема процедура. Опис процесів оброблення даних у процедурах називають *кодом*. З кожним об'єктом може відбуватися безліч подій, але обробляються тільки необхідні, тільки ті, до яких написано процедури оброблення.

Обробник події — це процедура, яка виконується, як реакція на подію.

Отже, у VB.NET кожна програма складається з інтерфейсу користувача та програмного коду. Інтерфейс програми забезпечує введення і виведення програмного коду, що, в свою чергу, забезпечує опрацювання інформації.



Висновки

Основною одиницею в об'єктно-орієнтованому програмуванні є *програмний об'єкт*, який поєднує властивості та дії, які з ним можна виконувати.

Метод описує дії, які може виконувати той чи інший об'єкт.

Графічний інтерфейс користувача — це візуалізований засіб взаємодії програми з користувачем.

Елементи керування — це об'єкти, за допомогою яких створюється графічний інтерфейс.

Форма — це об'єкт, призначений для створення вікон, в яких відображаються процеси виконання програми.

Подія — це характеристика об'єкта, яка вказує, на яку дію користувача може реагувати об'єкт. З будь-якою подією можна пов'язати операції оброблення даних, для яких записують *процедуру* мовою програмування. Опис процесів оброблення даних у процедурах називають *кодом*.

Обробник події — це процедура, яка виконується, як реакція на подію.



Контрольні запитання та завдання

1. Що таке програмний об'єкт? Наведіть приклади об'єктів у Windows-додатках.
2. Наведіть приклади властивостей об'єкта «кнопка».
3. Що таке графічний інтерфейс користувача?
4. Що таке елементи керування? Наведіть приклади елементів керування.
5. Що таке екранна форма?
6. Що таке подія? Наведіть приклади подій.
7. З яких частин складається програма в об'єктно-орієнтованому програмуванні?



Питання для роздумів

- 1*. Розгляньте об'єкт «мобільний телефон». Вкажіть, які він має властивості та значення цих властивостей.
- 2*. Опишіть у вигляді набору властивостей об'єкт «учень».



Завдання для досліджень

- 1*. У текстовому редакторі MS Word визначте елементи керування діалогового вікна **Абзац**.

§23

Ознайомлення із середовищем програмування

23.1.

ПОЧАТОК РОБОТИ В СЕРЕДОВИЩІ

У цьому параграфі ми розпочинаємо роботу з одним із потужних середовищ програмування Visual Basic.Net*.

* Для ознайомлення із середовищем візуального програмування обрано Visual Basic.Net, що входить до пакета Visual Studio (версія Visual Basic 2010). Дане середовище має російський інтерфейс, до всіх кодів програми є спливаючі підказки на російській мові.

Якщо на комп'ютері встановлено операційну систему Windows 7 і нижче доцільно використовувати 2010 версію даного пакета (зазначимо, що в цьому випадку не обов'язково встановлювати весь пакет, достатньо встановити лише Visual Basic 2010 Express, який займає близько 150 Мб). Якщо на комп'ютері встановлено систему Window 8 і вище, слід встановлювати Visual Studio 2012 і вище.

Зазначимо, що всі версії є вільно розповсюдженими і не потребують ліцензії.

Мова Visual Basic (VB) є результатом значного вдосконалення алгоритмічної мови програмування Basic. Завдяки VB можна створювати як прості, так і досить складні прикладні системи. Її можна використовувати для розв'язування обчислювальних задач, для роботи з базами даних, мультимедійними засобами, для розв'язування задач в Інтернеті. Навіть, якщо в майбутньому у вас не виникне потреби розробляти програмні засоби, ознайомлення з VB дозволить вам зрозуміти, як працюють різні додатки операційної системи Windows. Зазначимо також, що MS Office має вбудований VBA (аналог VB), який дозволяє програмувати певні задачі безпосередньо в таких програмах, як MS Word, Excel, PowerPoint тощо.

Перейдемо безпосередньо до роботи із самим середовищем.

Запуск програми VB 2010 можна виконати через меню кнопки **Пуск**, обравши **Усі програми** ▶ **Microsoft Visual Studio 2010** ▶ **Microsoft Visual Studio 2010**.

При першому запуску програми в діалоговому вікні, що з'явиться, слід обрати **Visual Basic**.

Якщо ж було інстальовано пакет Visual Basic 2010 Express, то для завантаження через кнопку **Пуск** слід обрати **Усі програми** ▶ **Microsoft Visual Studio 2010 Express** ▶ **Microsoft Visual Basic 2010 Express**.

Після запуску програми з'являється вікно, фрагмент якого представлено на рис. 5.3.

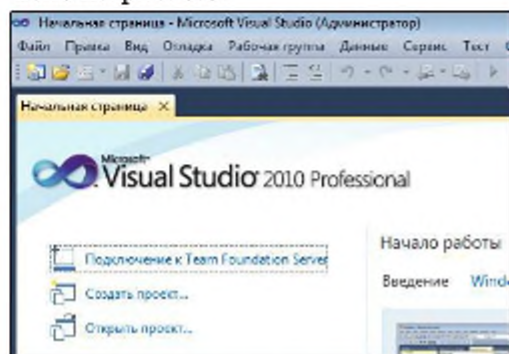


Рис. 5.3. Фрагмент вікна програми Visual Studio 2010

Програми створені в середовищі VB складаються з різних об'єктів. Ці програми прийнято називати *проектами*.

Щоб створити новий проект, треба обрати команду **Создать проект...** (Створити проект) і в наступному вікні, що відкриється, обрати середовище програмування **Visual Basic** та **Приложение Windows Forms** (Додаток Windows Forms) (рис. 5.4).

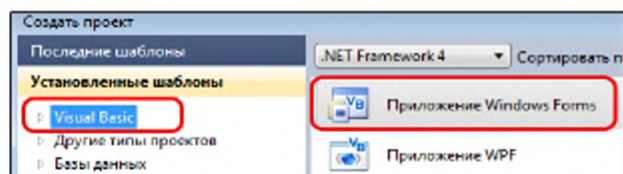


Рис. 5.4. Вікно **Создать проект** (Створити проект)

У результаті перед вами з'явиться вікно для подальшого проектування (рис. 5.5).

23.2.

СТРУКТУРА ГОЛОВНОГО ВІКНА

Головне вікно VB (рис. 5.5) за зовнішнім виглядом нагадує стандартне вікно Windows. Воно складається із заголовка, меню, панелі інструментів, робочої ділянки тощо. Робоча ділянка містить ряд вікон, кількість, місце розташування і розмір яких встановлює користувач.

Меню головного вікна VB містить команди, що згруповані за функціональною ознакою.

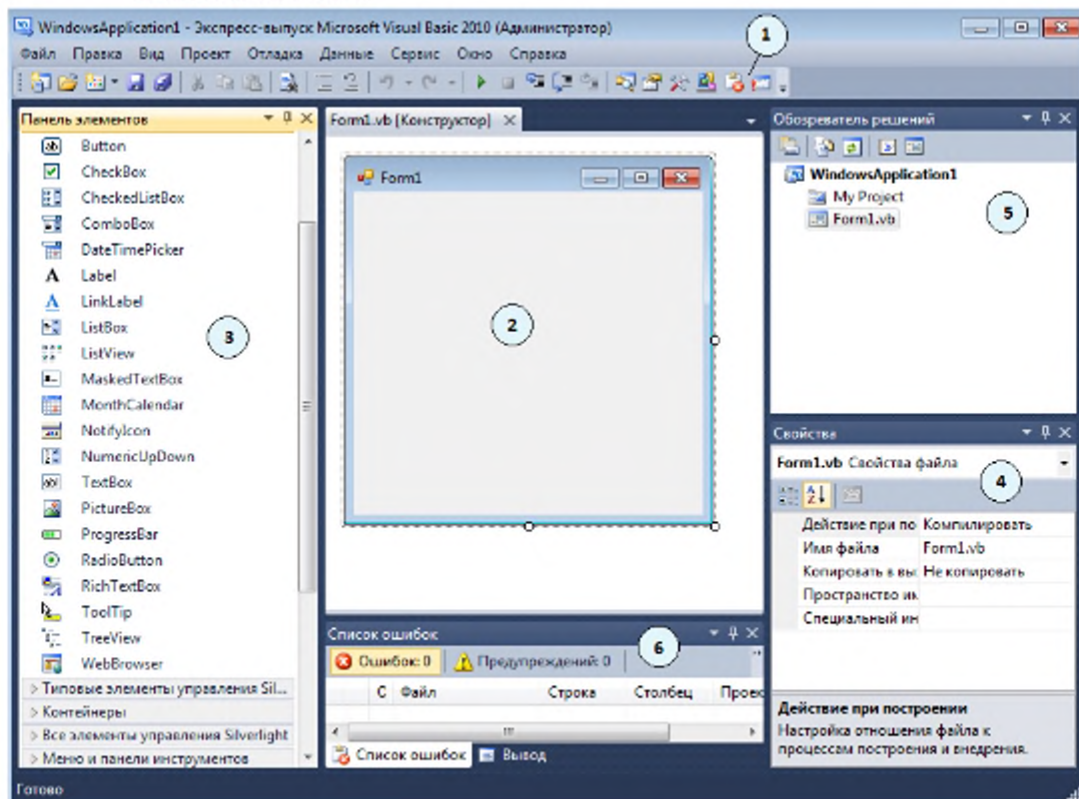


Рис. 5.5. Структура головного вікна: 1) панель інструментів; 2) вікно форми **Form1.vb**; 3) панель елементів керування **Панель элементов**; 4) вікно властивостей об'єкта **Свойства**; 5) вікно складових проекту **Обозреватель решений**; 6) вікно списку помилок **Список ошибок**.

На панелі інструментів розміщено інструменти, як і в більшості Windows-додатків, які призначені для керування роботою середовища. Ось призначення тільки деяких кнопок, що стануть у нагоді для відображення тих чи інших вікон середовища.



вікно складових проекту
Обозреватель решений



вікно властивостей об'єкта
Свойства



вікно **Панель элементов**



вікно **Список ошибок**

23.3. ВІКНО ФОРМИ

При завантаженні VB 2010 у центрі головного вікна середовища розташовано форму (рис. 5.6), яка вміщена у вкладці, назва якої складається з назви форми (**Form1.vb**) і напису **[Конструктор]**. Це свідчить про відображення режиму візуального конструювання форми проекту. У вкладці вікна кодування подій, що відноситься до цієї ж форми, напису **[Конструктор]** не буде. Вікно форми спочатку порожнє. Для візуального оформлення проекту використовують елементи керування, що розташовані на вікні **Панель елементів**. Отже, форма — це вікно, що містить заголовок, робочу ділянку та межу.

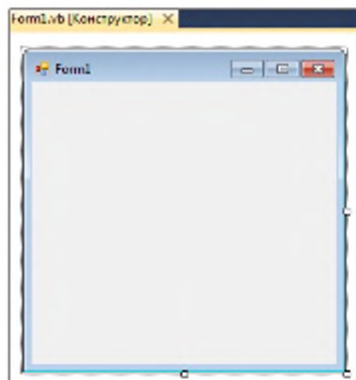





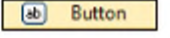
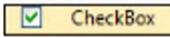
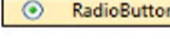
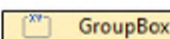
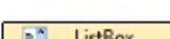
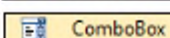
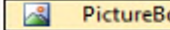
Рис. 5.6. Вікно форми

23.4. ПАНЕЛЬ ЕЛЕМЕНТІВ КЕРУВАННЯ

Панель елементів керування розміщується ліворуч від форми і в разі потреби відкривається інструментом  **Панель елементів** (Панель елементів), що знаходиться на панелі інструментів. Панель можна зафіксувати у вікні за допомогою спеціального значка у вигляді кнопки, що розташований у верхньому правому куті панелі.

Щоб на формі розташувати елемент керування, достатньо його обрати на панелі елементів керування, а потім клацнути лівою клавішею миші на формі або провести умовний прямокутник для встановлення бажаного розміру.

Найбільш уживані елементи керування

- | | |
|---|--|
|  Label | — елемент відображення тексту, який користувач не може змінити з клавіатури |
|  TextBox | — текстове поле для введення текстових даних |
|  Button | — кнопка, при клацанні на яку відбувається визначена подія |
|  CheckBox | — прапорець для вибору якого-небудь параметра |
|  RadioButton | — перемикач для вибору одного з двох станів: увімкнено та вимкнено |
|  GroupBox | — рамка для об'єднання в групу кількох елементів керування (прапорців або перемикачів) |
|  ListBox | — список для вибору елементів списку |
|  ComboBox | — поле зі списком для вибору елементів списку |
|  PictureBox | — відображення рисунка |

23.5.

ВІКНО ВЛАСТИВОСТЕЙ

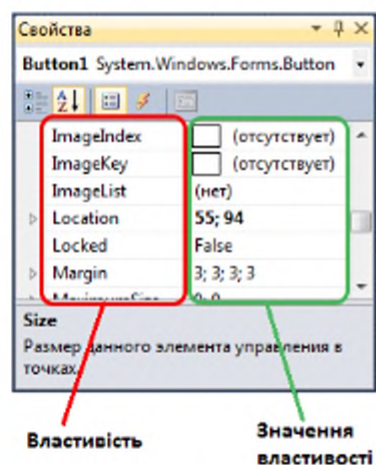




Рис. 5.7. Вікно **Свойства** (Властивості)

Кожний елемент керування розміщений на формі та сама форма мають певні властивості. Переглянути або змінити ці властивості можна у вікні **Свойства** (Властивості). У лівому стовпці вікна розташовано назву властивості, у правому — її значення (рис. 5.7). Назви властивостей може бути відображено як за абеткою, так і за призначенням. Щоб вибрати спосіб відображення властивостей, обирають інструмент  **По категоріям** для виведення властивостей за категоріями або  **В алфавитном порядку** для виведення властивостей за абеткою.

Якщо клацнути елемент керування на формі, то у вікні **Свойства** (Властивості) з'являться його властивості. Так, досить поширеними є такі властивості: **Name** — ім'я об'єкта, **Text** — текст.

ВПРАВА 23



Завдання. У новій формі змінити ім'я, заголовок і розмір. Додати елементи керування **Label** і **Button**. Змінити властивість **Text** у цих елементах.



1. У вікні властивостей оберіть **Name** та введіть ім'я форми **Знайомство**.
2. Оберіть властивість **Text** і введіть текст — **Знайомство**.
3. Оберіть властивість **Size** та введіть значення — **400; 400**.
4. Додайте до форми елемент керування **Label**. Для цього оберіть його в панелі елементів та клацніть на формі.
5. Оберіть властивість **Text** і введіть своє прізвище та ім'я.
6. Додайте до форми елемент керування **Button**. Для цього оберіть його в панелі елементів та клацніть на формі.
7. Оберіть властивість **Text** і введіть значення — **OK**.

Примітка. Вилучити зайві елементи керування можна за допомогою клавіші **Delete**.

23.6.

ВІКНО СКЛАДОВИХ ПРОЕКТУ

Проект може складатися з багатьох компонентів (форм, модулів, класів тощо). Побачити ієрархію цих компонентів і перейти безпосередньо до роботи з потрібними можна за допомогою вікна складових проекту **Обозреватель решений** (Браузер рішень).

Під смугою заголовка вікна (рис. 5.8) розміщені кнопки для відображення різних режимів проекту. В майбутньому проекті найчастіше ми будемо користуватися кнопками  **Перейти к коду** (Перейти до коду) та  **Открыть в конструкторе** (Відкрити в конструкторі).

У вікні проекту можна перейти до потрібної форми, клацнувши на ній два рази лівою клавішею миші.

Щоб створити нову форму, треба в меню **Проект** обрати **Добавить форму Windows...** (Додати форму Windows).

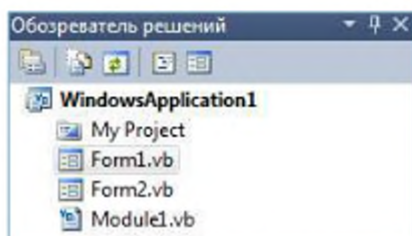


Рис. 5.8. Вікно **Обозреватель решений** (Браузер рішень)



Висновки

Головне вікно середовища VB містить такі основні складові: меню; панель інструментів; **Панель елементів**; вікно форми; вікно складових проекту **Обозреватель решений**; вікно властивостей об'єкта **Свойства**; вікно **Список ошибок**.

Форма — вікно, що містить заголовок, робочу ділянку та межу.

Візуальне оформлення проекту здійснюється з використанням елементів керування, що розташовані на панелі елементів керування.

Кожний елемент керування розміщений на формі та сама форма мають певні властивості, що розташовані у вікні **Свойства**.



Контрольні запитання та завдання

1. Опишіть основні складові головного вікна VB.NET.
2. Що таке форма?
3. Які елементи керування ви знаєте? Вкажіть їх призначення.
4. Назвіть властивості елементів керування.
5. Як змінити властивість елемента керування?
6. Яке призначення вікна **Обозреватель решений** (Браузер рішень)?
7. Як додати нову форму?



Питання для роздумів

- 1*. Як можна використати для створення проектів властивість **Visible Enabled**?
- 2*. Які елементи керування, розглянуті в цьому параграфі, траплялися вам при роботі з Windows-додатками? Розкажіть, у яких випадках вони використовувалися.



Завдання для досліджень

- 1*. Дослідіть способи зміни розташування елементів головного вікна середовища програмування? Поміняйте панелі **Свойства** та **Панель елементів** місцями.

§24

Створення найпростішого проекту

Розміщення елементів керування на формі не призводить до створення готового програмного продукту, а тільки вибудовує зовнішній вигляд його представлення — форму, яка ще не буде «реагувати» на дії користувача. Для цього треба написати *програмний код*, за допомогою якого буде опрацьовуватись необхідна подія.

Набір команд, який використовують для опису реакції на конкретну подію та записують за правилами мови програмування, називають **процедурою**.


Наприклад, до подій можна віднести клацання кнопкою миші, подвійне клацання миші, наведення курсору на об'єкт, зміну розміру об'єкта, переміщення об'єкта по екрану тощо. Зверніть увагу, що події відбуваються тільки під час виконання програми, а не під час її проектування.

Виконання процедур дозволяє змінювати властивості елементів керування, виконувати обчислення, виводити вікна повідомлень тощо.

24.1.

ВІКНО ПРОГРАМНОГО КОДУ

При запуску VB вікно коду не відображається, воно відображається тільки тоді, коли потрібно редагувати код.

Для доступу до вікна коду можна двічі клацнути лівою клав'яшею миші на формі чи деякому елементі керування, або вибрати кнопку  **Перейти к коду** у вікні **Обозреватель решений**.

Вікно програмного коду об'єкта має назву, що відповідає імені форми.

Розглянемо зовнішній вигляд вікна програмного коду та призначення його основних елементів.

При подвійному клацанні на формі у вікні коду за умовчанням утворюється *чотири* рядки коду.

- !** Верхній та нижній рядки (`Public Class Form1` та `End Class`) оголошують клас процедур для елементів керування форми (їх видаляти не можна!). На рис. 5.9 `Public Class Form1` — заголовок класу, `End Class` — закриття класу процедур.

Внутрішні рядки (`Private Sub...` та `End Sub`) оголошують процедуру для певного елемента керування. Тут `Private Sub...` — заголовок процедури, `End Sub` — закриття процедури.

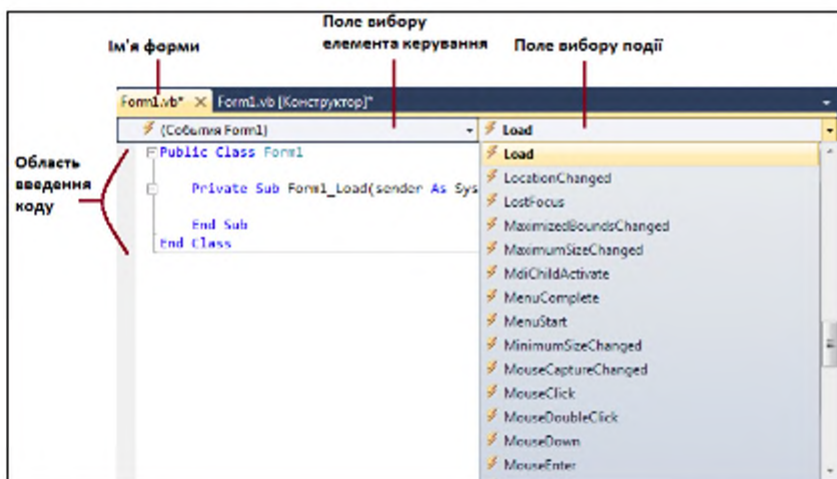


Рис. 5.9. Вікно коду

У заголовку процедури вказується ім'я, яке складається з імені відповідного елемента керування, символу підкреслення та події, яка найчастіше зустрічається при роботі з даним елементом.

Якщо потрібно обрати іншу подію, її назву вибирають зі списку на полі справа, яке розташоване під смугою заголовка вікна коду.

Події, які часто використовують при створенні проектів: **Click** (кляцання на об'єкті), **DoubleClick** (подвійне кляцання на об'єкті), **Load** (завантаження форми).

Якщо потрібно написати процедуру для іншого елемента керування, то його ім'я вибирається зі списку ліворуч.

Сам код програми треба записувати між службовими словами `Private Sub...` та `End Sub`.

Крім тексту поточної процедури, в робочій ділянці вікна коду відображаються тексти інших процедур, пов'язаних з усіма елементами даної форми.

24.2. ЗМІНА ВЛАСТИВОСТЕЙ ОБ'ЄКТА

Створюючи коди проекту, необхідно вміти правильно записувати зміни властивостей елементів керування. Синтаксис зміни властивості елемента керування має такий загальний вигляд:

```
<Ім'я об'єкта>.<Властивість>=<Значення властивості>
```

Наприклад,

```
Label1.Text = "Привіт!!!"
```

```
Button1.Visible = True
```

```
TextBox1.Enabled = False
```

У процесі створення програмного коду при введенні імені описаного об'єкта з'являється список можливих імен усіх об'єктів даної форми та властивостей. Для уникнення помилок при введенні імен елементів керування та їх властивостей бажано обрати значення зі списку (рис. 5.10).

```
Private Sub Button1_Click(sender As System.Object, e As EventArgs)
    But
```

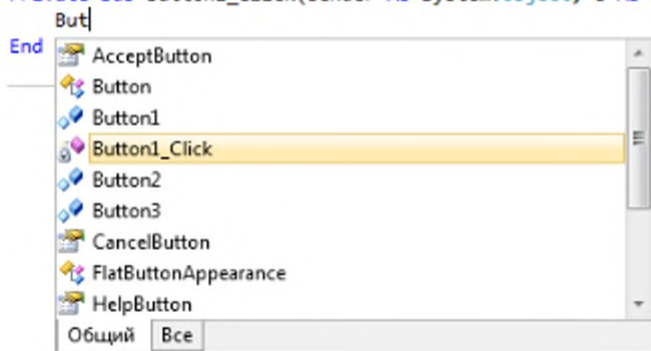



Рис. 5.10. Вікно для вибору імені елемента керування

Якщо ви ввели ім'я елемента керування, а в списку воно не відображено, то це означає, що ви припустилися помилки.

24.3.

ВИКОНАННЯ ПРОГРАМИ

Після введення коду програми її можна запустити на виконання та зупинити виконання. Усе це можна виконати за допомогою команд **Начать отладку** (Розпочати налагодження) або **Остановить отладку** (Зупинити налагодження) з меню **Отладка** (Налагодження). Досить зручно для виконання цих дій використовувати відповідні інструменти або комбінації клавіш.

 — **Начать отладку (F5)** (Розпочати налагодження)

 — **Остановить отладку (Ctrl+Alt+Break)** (Зупинити налагодження)

24.4.

РЕДАГУВАННЯ КОДУ ОБРОБНИКА ПОДІЇ



Щоб створити програму, потрібно навчитись редагувати програмний код. Кожну команду в VB.NET записують з нового рядка. Якщо потрібно записати дві команди в одному рядку, то їх відокремлюють двокрапкою «:».

Якщо в кодї програми необхідно записати певні пояснення, які не є програмним кодом і не мають виконуватися програмою, їх записують після знака апострофа. Інакше це називають *коментарем*. Коментар відляється у кодї програми зеленим кольором:

```
'Це коментар
```

У коментар зручно перетворювати незавершені фрагменти програм, які можуть знадобитися пізніше.

Доволі часто в програмі виникає потреба введення подібних за змістом програмних кодів, які можна копіювати відомими вам способами копіювання фрагментів тексту: за допомогою панелі інструментів; з використанням контекстного меню; з використанням комбінацій клавіш **Ctrl+C** (копіювати) і **Ctrl+V** (вставити).

Як і в більшості текстових редакторів, у VB.NET також можна відмінити останні дії введення та редагування тексту, а також відмінити дії за допомогою відповідних кнопок  (Відмінити) та  (Повернути назад) або з використанням комбінації клавіш **Ctrl+Z** (Відміна дії) та **Ctrl+Y** (Відновлення відміненої дії). Також ці дії можна виконати за допомогою команд меню **Правка**.

Якщо код програми введено неправильно, то він буде підкреслений кольоровою хвилястою лінією, а у вікні **Список помилок** з'явиться повідомлення про помилку, де буде вказано місце розташування помилки та причини виникнення (рис. 5.11). Щоб перейти в місце розташування помилки в коді, достатньо клацнути два рази на повідомлення про помилку у вікні **Список помилок**.



Дізнатися більше
про основні події
ви зможете на диску.

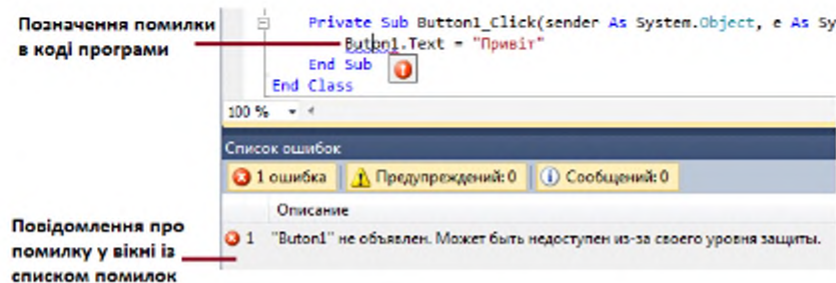


Рис. 5.11. Помилка в програмі та повідомлення у вікні **Список помилок**

! Зауважимо, що у вікні програмного коду рядок коду може мати довжину понад 500 символів. І якщо на комп'ютері для перегляду коду можна застосувати смугу прокручування, то на папері такого не зробіш. Отож, для перенесення рядків програмного коду в підручнику або в зошиті прийнято використовувати знак нижнього підкреслення « », який при введенні коду на комп'ютері не використовується.

24.5. ЗБЕРЕЖЕННЯ ПРОЕКТУ

Щоб зберегти проект, треба обрати команду **Файл** ▶ **Сохранить все** (Файл ▶ Зберегти все). Далі в діалоговому вікні ввести назву проекту, наприклад **NewProject**, (за умовчанням проект має назву **WindowsApplication1**) та обрати папку, де він буде зберігатися.



Дізнатися більше,
як зберігати проекти
не за умовчанням,
ви зможете на диску.

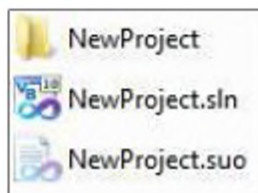


Рис. 5.12. Вміст папки збереженого проекту

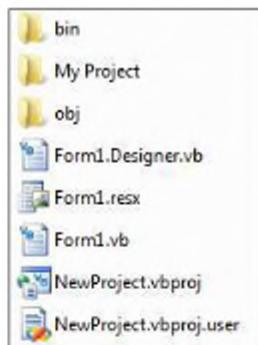


Рис. 5.13. Вміст папки **NewProject**

Після збереження проекту генерується багато файлів, які мають різне призначення. Якщо відкрити папку збереженого проекту (рис. 5.12), то в ній розташовано папку з назвою проекту **NewProject** і два файли — з розширенням **.sln** (головний файл) і з розширенням **.suo** (службовий файл, що відображається тільки в режимі показу прихованих файлів).

У папці **NewProject** (рис. 5.13) автоматично буде створено три папки: **bin**, **NewProject**, **obj**; файли, що стосуються форми (для кожної форми по три файли) та ще два файли проекту.

Якщо проект потрібно перенести на інший комп'ютер, то слід копіювати папку найвищого рівня і для відкриття проекту обрати файл **NewProject.sln** або **NewProject.vbproj**.

Середовище VB.NET містить перекладач програм користувача в коді машини, який називають транслятором-компілятором. Він дає змогу не лише перекладати і виконувати програми, а й створювати виконуваний **exe**-файли. Збереження проекту призводить також до створення файла з розширенням **.exe**, який можна запустити на виконання без встановлення на комп'ютері середовища програмування. Цей файл автоматично генерується до папки **bin\Debug** після запуску та збереження проекту.

ВПРАВА 24



Завдання. Створити проект з двома кнопками. Якщо натиснути першу кнопку, на ній має з'явитися ваше прізвище, натиснути другу кнопку — ваше ім'я.

1. Створіть новий проект із формою **Windows**.
2. Розташуйте на формі елементи керування **Button1** та **Button2**.
3. Для об'єкта **Button1** створіть процедуру виведення прізвища.
 - а) Клацніть два рази на кнопці **Button1**. У результаті чого буде утворено рядки заголовка та кінця процедури для події **Button1.Click** (клацання миші по кнопці **Button1**).

```
Private Sub Button1_Click (sender As Object, e As EventArgs) _
Handles Button1.Click
End Sub
```

Пояснення. Перший та останній рядки в кожному програмному коді записуються автоматично, тобто програмісту їх не треба записувати. Тут **Button1_Click** — це ім'я процедури, яке утворюється з імені елемента керування та події, що до нього буде застосована. Проте сама дія і елемент керування, до якого буде дія застосована, записується в кінці заголовка процедури:

«**Handles Button1.Click**».

б) Між утвореними рядками заголовка та кінця процедури введіть програмний код, у якому замість слова «Прізвище» введіть власне прізвище:

```
Button1.Text = «Прізвище»
```

4. Клацніть два рази на кнопці **Button2** та введіть програмний код, у якому замість слова «Ім'я» введіть власне ім'я:


```
Button2.Text = «Ім'я»
```

- Запустіть проект на виконання. Натисніть по черзі на кожну кнопку. Переконайтеся, чи правильно виконується проект.
- Збережіть проект, обравши команду **Файл** ▶ **Сохранить все**.
- У діалоговому вікні в полі **Имя** введіть назву проекту **Вправа_24_Прізвище** та вкажіть папку для збереження проекту.
- Закрийте середовище програмування.
- Зайдіть до папки, яку ви вказали для створення проекту, оберіть папку **Вправа_24_Прізвище** і запустіть звідти файл **Вправа_24_Прізвище.sln**.
- Переконайтеся, що відкрито збережений вами проект. Закрийте вікно середовища.
- Зайдіть до папки **Вправа_24\Вправа_24\bin\Debug** та запустіть файл **Вправа_24_Прізвище.exe**. Переконайтеся, що запущено збережений вами проект. Перевірте проект, натиснувши по черзі на кнопки.



Висновки

Набір команд, який використовують для опису реакції на конкретну подію та записують за правилами мови програмування, називають *процедурою*.

Щоб викликати вікно програмного коду, треба двічі клацнути лівою клавішею миші на формі чи на елементі керування, або натиснути кнопку  **Перейти к коду** у вікні **Обозреватель решений**.

Синтаксис зміни властивості елемента керування має такий загальний вигляд: `<Ім'я об'єкта>.<Властивість>=<Значення властивості>`

Щоб зберегти весь проект, треба обрати команду **Файл** ▶ **Сохранить все**.



Контрольні запитання та завдання

- Що таке процедура?
- Назвіть, які ви знаєте події. До яких елементів керування їх можна застосувати?
- Як у вікні коду обрати потрібний елемент керування?

4. Як у вікні коду змінити подію, яку треба застосувати до елемента керування?
5. Опишіть способи запуску та зупинки проекту.
6. Що таке коментар? Як записують коментар у програмному коді?
7. Як виконати копіювання програмного коду?
8. Де можна побачити повідомлення про помилку в програмному коді?
9. Як зберегти проект?
10. Які файли генеруються при збереженні проекту?



Питання для роздумів

- 1*. Що треба зробити перед тим, як запустити проект на виконання?
- 2*. У яких випадках у проекті можна використати подію **MouseMove**?
- 3*. Як виправити помилку, що з'явилася в коді програми?



Завдання для досліджень

1. Дослідіть призначення подій **Resize** та **KeyPress** при застосуванні їх до форми.
- 2*. Визначте, як можна додати та забрати коментар за допомогою контекстного меню виділеного рядка програмного коду.
- 3*. Визначте, яким чином можна змінити колір шрифту текстового напису в програмному коді.

§25

Властивості та методи елементів керування

Кожний елемент керування має свої властивості. Про деякі властивості вам уже відомо. В залежності від об'єктів набір властивостей дещо змінюється. Однак деякі об'єкти мають спільні властивості.

Змінювати властивості елементів керування можна по-різному:

- у режимі *конструювання* за допомогою вікна **Свойства**;
- під час виконання *програмного коду*, який прописаний заздалегідь у вікні коду.

Розглянемо деякі особливості зміни властивостей в режимі конструювання у вікні **Свойства**.

Властивість, біля назви якої розташовано невеличкий трикутник, має декілька значень. Так, наприклад, властивість **Size** має значення ширини та висоти елемента керування.

Набір властивостей залежить від типу елемента керування. Властивості будь-якого елемента керування можна проглянути, якщо виділити елемент на формі або вибрати його ім'я на полі зі списком, яке розміщене під смугою заголовка у вікні **Свойства**.

Властивість можна змінити такими способами: обрати одне із запропонованих значень у списку; ввести значення самостійно; встановити значення в діалоговому вікні.

25.1. НАЙБІЛЬШ УЖИВАНІ ВЛАСТИВОСТІ ЕЛЕМЕНТІВ КЕРУВАННЯ

Однією з основних властивостей, яку мають усі елементи керування, є властивість **Name**, завдяки якій встановлюють ім'я елемента керування. Зазвичай один і той самий клас елементів керування має однакове ім'я, після якого встановлено число, що визначає послідовність розташування цього елемента на формі. Так для написів — це імена **Label1**, **Label2**, **Label3**, для кнопок — **Button1**, **Button2**, **Button3** і т. д. Властивість **Name** можна встановити лише в режимі конструювання. Під час виконання програми цю властивість змінити не можна.

Існують деякі обмеження на встановлення імен. Ім'я елемента керування:

- може містити літери, цифри та символи підкреслення;
- має починатися з літери або символу підкреслення, але не з цифри;
- не може містити пробілів та інших символів, крім знака підкреслення.

Нижче наведено таблицю, що містить перелік найбільш уживаних властивостей, а також таблицю властивостей, які притаманні лише для форм.

Таблиця 5.1. Найбільш уживані властивості елементів керування

Властивість	Пояснення
AutoSize	Встановлення можливості зміни розмірів елемента керування
BackColor	Колір фону
Enabled	Активність
ForeColor	Колір тексту
Font	Встановлення параметрів шрифту (назва, розмір, накреслення тощо)
Location	Місце розташування від лівого та верхнього краю
Size	Розмір (ширина і висота)
Text	Текстовий напис
Visible	Видимість



Дізнатися більше
про властивості
елементів керування ви
зможете на диску.



Дізнатися більше
про властивості
форм ви зможете на
диску.

Таблиця 5.2. Основні властивості форм (Form)

Властивість	Пояснення
BackgroundImage	Фоновий малюнок
FormBorderStyle	Тип границь форми
Icon	Іконка форми (знаходиться у верхньому лівому куті заголовка форми та при згортанні на панелі задач)
Opacity	Прозорість
StartPosition	Визначення позиції початкового завантаження вікна

Зауважимо, якщо користувач не знає призначення тієї чи іншої властивості елемента керування, достатньо клацнути та цю властивість у вікні **Свойства** (Властивості) та прочитати пояснення в нижній частині цього вікна.

ВПРАВА 25



Завдання. Створити форму за зразком (рис. 5.14). Задати імена елементам керування та встановити необхідні властивості.

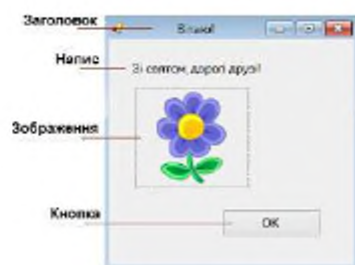


Рис. 5.14





Виконати вправи «Покупка», «Прайс-лист» і «Анкета» ви зможете на диску.

- Визначте необхідні властивості для **форми**.
 - До властивості **Name** введіть ім'я елемента керування **Вітання**.
 - До властивості **Text** введіть текст «Вітання!».
- Додайте об'єкт **Напис** та встановіть його властивості.
 - Оберіть на панелі елемент керування **Label** та розташуйте його на формі.
 - До властивості **Name** введіть ім'я **Свято**.
 - До властивості **Text** введіть текст «Зі святом, друзі!»
- Додайте об'єкт **Кнопка** та встановіть його властивості.
 - Оберіть на панелі елемент керування **Button** та встановіть його на формі.
 - До властивості **Name** введіть ім'я **ОК**.
 - До властивості **Text** введіть текст **ОК**.
- Додайте об'єкт **Зображення** та встановіть його властивості.
 - Оберіть елемент керування **PictureBox** та встановіть прямокутник потрібного розміру на формі.
 - До властивості **Name** введіть ім'я **Свято**.
 - У властивості **SizeMode** оберіть значення **Zoom**.
 - У властивості **Image** оберіть **Локальний ресурс** ▶ **Імпорт** (Локальний ресурс ▶ Імпорт), виберіть якийсь малюнок, що міститься на комп'ютері та натисніть **ОК**.
- Збережіть проект під назвою **Вправа_25_Прізвище**.

25.2.

ПОНЯТТЯ МЕТОДУ

Встановлюючи властивості об'єктів у програмному кодї, ви напевно помітили, що в списку, який з'являється для вибору його властивостей, значення позначено різними значками. Так, властивості позначено піктограмою у вигляді руки з аркушем . Звернемо увагу на слова позначені фіолетовим ромбом  (рис. 5.15). Їх також можна застосувати до позначеного об'єкта. Проте це не властивості, а методи. Властивості — це характеристики об'єктів. *Методи*, на відміну від властивостей, відповідають за виконання певних дій даним об'єктом.

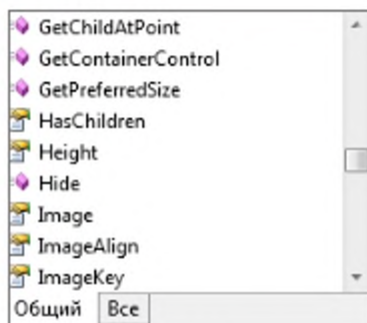


Рис. 5.15

Метод визначає дії, які може виконати об'єкт.

На відміну від властивостей методи не можна викликати через вікно **Свойства**, їх можна задати лише у вікні коду проекту.

Для кожного об'єкта в системі Visual Basic.NET визначено кілька стандартних методів, якими може оперувати програміст.

Подібно до того, як властивості об'єкта визначають його вигляд і поведінку, методи описують дії, що пов'язані з цим об'єктом. Фактично, метод — це фрагмент програмного коду, вбудований в об'єкт. Використовуючи вбудований метод, можна розв'язувати поставлену задачу стосовно об'єкта, жодних додаткових команд при цьому від програміста не вимагається.

Як і у випадку з властивостями, з кожним типом об'єктів співвідносяться різні методи та різна їх кількість. З об'єктом класу «форма» пов'язані більше десятка методів. Деякі методи можуть використовуватися практично з усіма типами об'єктів, інші можуть бути більш специфічними.

Щоб викликати метод, потрібно вказати ім'я об'єкта і через крапку ім'я методу.

<Ім'я об'єкта>.<Ім'я методу>

Наприклад, `Form2.Close ()` — закрити форму.

У таблиці нижче наведено приклади деяких методів, що використовують для форми.

Таблиця 5.3. Приклади методів, що використовуються для форм

Метод	Пояснення
Activate	Зробити активною
CenterToScreen	Розташувати по центру екрана (лише для поточної форми)
Close	Закрити
Hide	Приховати
Show	Відобразити

Зауважимо, якщо метод застосовують до поточної форми, то до неї звертаються не за іменем, а використовуючи слово **Me**. Наприклад, `Me.Hide` — приховати форму.

Для деяких методів є потреба, щоб для їх виклику ввели певну інформацію, яка буде впливати і на те, як виконується метод, і на результати його виконання. Дані, що передаються методам, називають *аргументами* і записують після назви методу в дужках через кому.

ВПРАВА 26



Завдання. Створити презентацію «Пристрої друку». На першій формі розташувати назву презентації та кнопки переходу до відповідних видів пристроїв друку: **Лазерний принтер**, **Струменевий принтер**, **3D-принтер**, **Плотер**. На другій та наступних формах розташувати назву принтера, його зображення та кнопки переходу до попередньої, наступної та першої форми.

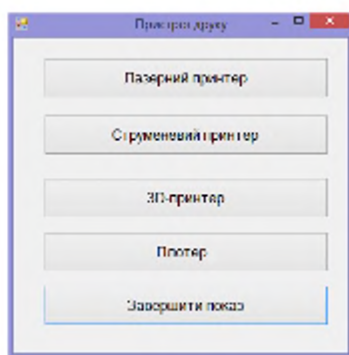


Рис. 5.16

Зображення для проекту можна скопіювати з папки **Files > Вправа_26** на диску або завантажити з мережі Інтернет.

1. Створіть форму за зразком (рис. 5.16), розташувавши на ній п'ять кнопок. Властивості елементів керування задайте самостійно. Надайте формі ім'я **Zm1st**.
2. Додайте до проекту ще чотири форми, обравши для цього в меню **Проект** команду **Додати форму Windows...** (Додати форму Windows...).
3. Надайте формам імена **Laz_Print**, **Strum_Print**, **3D_Print** та **Plotter**.
4. Наступні форми створіть згідно зразка (рис. 5.17), розташувавши на них відповідні написи та зображення.

Для відображення стрілок ліворуч і праворуч встановіть назву шрифту **Webdings**, увівши цифру 3 для стрілки ліворуч і 4 — для стрілки праворуч.

5. Задайте коди для елементів керування форми **Zm1st**.

а) Для завантаження форми задайте метод, що встановлює форму по центру екрана.

```
Me.CenterToScreen()
```

б) Щоб клацати на кнопці **Лазерний принтер**, задайте методи для приховування форми **Zm1st** та відображення форми **Laz_Print**.

```
'Приховати поточну форму
```

```
Me.Hide()
```

```
'Відобразити форму Laz_Print
```

```
Laz_Print.Show()
```

в) Аналогічно створіть коди для кнопок, що мають здійснювати перехід до інших форм.

г) Для клацання на кнопці **Завершити показ** введіть команду

End

Завдяки цій команді буде завершено демонстрацію проекту.

6. Для інших форм задайте коди самостійно. В результаті форми мають завантажуватися по центру та здійснювати перехід до потрібних форм.

7. Самостійно додайте ще дві форми з іншими видами пристроїв друку.

8. Запустіть проект та переконайтеся, чи правильно він працює.

9. Збережіть проект під назвою **Вправа_26_Прізвище**.

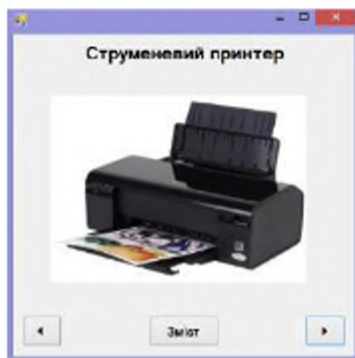


Рис. 5.17



Виконати вправу «Київ» ви зможете на диску.



Висновки

Властивості елементів керування можна змінювати в режимі конструювання за допомогою вікна властивостей та під час виконання програмного коду, який має бути записаний заздалегідь у вікні коду.

Методи відповідають за виконання певних дій даним об'єктом.

Щоб викликати метод, потрібно вказати ім'я об'єкта і через крапку ім'я методу: <Ім'я об'єкта>. <Ім'я методу>

Якщо метод застосовується до поточної форми, то до неї звертаються не за іменем, а використовуючи слово **Me**.



Контрольні запитання та завдання

1. Які існують способи зміни властивостей елементів керування?
2. Яких правил слід дотримуватися при зміні імені елемента керування?
3. Які параметри можна встановити за допомогою властивості **Font**?
4. Яке призначення властивості **AutoSize**?
5. Яке призначення властивостей **Visible** та **Enabled**? Яких значень вони можуть набувати?
6. Що таке метод?
7. Як викликати метод у програмному коді?
8. Які основні методи форм ви знаєте?



Питання для роздумів

- 1*. Чому в програмному коді не можна змінити властивість **Name**?
- 2*. Чим метод відрізняється від властивості об'єкта?



Завдання для досліджень

- 1*. Визначте, за допомогою якої властивості можна встановити максимальні та мінімальні розміри елемента керування?
- 2**. Визначте способи зміни параметрів вирівнювання тексту в елементах керування **Label** і **TextBox**.
- 3**. Визначте методи, які притаманні об'єктові «кнопка».

§26

Використання вікон повідомлень

26.1.

СИНТАКСИС ПРОЦЕДУРИ

Зручним засобом представлення даних є вікна повідомлень, виведення яких здійснюється при виконанні програмного коду.

Щоб вивести вікно повідомлень, використовують процедуру **MsgBox**. Крім текстового напису, вікно повідомлення може містити визначені програмістом інформаційні значки та кнопки.

Процедура **MsgBox** має такий синтаксис:

`MsgBox(<Повідомлення>, <Кнопки> + <Значок>, <Заголовок>)`

<Повідомлення> — це будь-який текст, який містить певну інформацію.

<Кнопки> + <Значок> — значення, що відповідають розміщенню у вікні повідомлення певних кнопок і значків. Цей параметр є необов'язковим, і якщо в програмному коді він не вказаний, то у вікні повідомлення буде розташована лише кнопка **ОК**, а значки взагалі будуть відсутні.

<Заголовок> — напис у рядку заголовка необов'язковий, і в разі його відсутності в заголовку буде відображено назву проекту.

При зверненні до процедури **MsgBox** відкривається вікно, що містить деяке повідомлення і, як правило, хоча б одну командну кнопку (за умовчанням це кнопка **ОК**), яка використовується для підтвердження повідомлення.

Наприклад, після виконання наступної команди буде виведено вікно, яке зображене на рис. 5.18.

`MsgBox(«Петренко Іван», 1+64, «Прізвище та ім'я»)`

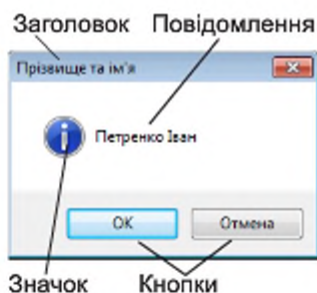


Рис. 5.18

26.2.

КОНСТАНТИ ТА ЗНАЧЕННЯ ПРОЦЕДУРИ MSGBOX

Константи та значення процедури **MsgBox** записують після префікса **MsgBoxStyle** (наприклад, **MsgBoxStyle.OkOnly**) або відповідним числовим значенням (наприклад, 0). Після знака «+» записують константу або значення з категорії «Значки». Їх константи також записують після префікса **MsgBoxStyle**.

Таблиця 5.4. Константи та значення процедури **MsgBox**

Константи	Значення	Опис
Категорія кнопки		
OkOnly	0	тільки кнопка «ОК»
OkCancel	1	кнопки «ОК» і «Отмена»
vbAbortRetryIgnore	2	кнопки «Прервать», «Повтор» і «Пропустить»
vbYesNoCancel	3	кнопки «Да», «Нет» і «Отмена»
vbYesNo	4	кнопки «Да» і «Нет»
vbRetryCancel	5	кнопки «Повтор» і «Отмена»
Категорія значки		
Critical	16	 — значок «Стоп»
Question	32	 — значок «Знак запитання»
Exclamation	48	 — значок «Знак оклику»
Information	64	 — значок «Інформація»


ВПРАВА 27



Завдання. Створити проект, що буде виводити вікно повідомлення з привітанням.

- Створіть новий проект та додайте на формі кнопку.
- Задайте програмний код для події клацання на кнопці.
`MsgBox(«Привіт, мене звали Ольга!», 0 + 64, «Привітання»)`
- Запустіть проект на виконання.
- Змініть програмний код таким чином, щоб користувач міг самостійно вводити своє ім'я до текстового поля.
 - Розташуйте на формі текстове поле та надайте йому ім'я **Im**.
 - Змініть програмний код проекту на такий:
`MsgBox(«Привіт, мене звали » & Im.Text, 0 + 64, «Привітання»)`



Виконати вправи «Опитування», «Обмін повідомленнями» і «Анкета_2» ви зможете на диску. 

Тут знак «&» ставиться для з'єднання постійних текстових значень з текстом, який буде введено до текстового поля **Im**.

- Запустіть проект на виконання. Після запуску проекту спочатку необхідно до текстового поля **Im** ввести своє ім'я, а потім клацнути по кнопці.
- Збережіть проект під назвою **Вправа_27_Прізвище**.



Висновки

Щоб вивести повідомлення в окремому вікні, використовують процедуру **MsgBox**.

Загальний вигляд процедури **MsgBox** має вигляд:

`MsgBox(<Повідомлення>, <Кнопки> + <Значок>, <Заголовок>)`



Контрольні запитання та завдання

- За допомогою якої процедури можна вивести повідомлення в окремому вікні?
- Які елементи має вікно повідомлення?



Питання для роздумів

- У яких випадках використовують вікна повідомлень у програмах?
- Для чого у вікні повідомлень використовують набір кнопок?

ПРАКТИЧНА РОБОТА №8

Створення об'єктно-орієнтованої програми, що відображає вікно повідомлення

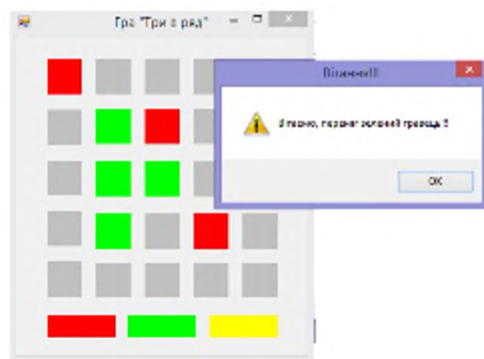


Рис. 5.19

Завдання. Створити проект реалізації гри «Три в ряд» (рис. 5.19). Гравці по черзі, встановлюючи власний колір квадрата, мають виставити в ряд (по горизонталі, по вертикалі або по діагоналі) три квадратики одного кольору. Перший гравець, клацаючи на квадратику один раз, встановлює червоний колір квадрата, другий має клацати на квадратику два рази, встановлюючи зелений колір. Після завершення гри треба клацнути на прямокутнику того кольору, хто переміг, або на жовтому в разі нічії. Після чого має з'явитися текстове повідомлення про перемогу червоного, зеленого гравця або оголошення нічії.

1. Встановіть для форми заголовок «Гра "Три в ряд"».
2. Розташуйте на формі елементи керування **Label** та встановіть потрібні властивості.

а) Додайте спочатку один елемент керування **Label** та встановіть для нього властивості, як у наведеній таблиці.

Властивість	Значення
Name	L11
AutoSize	False
Text	Відсутній
Size	40;40
BackColor	Сірий

б) Після встановлення властивостей створіть чотири копії даного об'єкта, перетягуючи його праворуч з утриманням лівої клавіші миші **+Ctrl**.

в) Встановіть для копій значення властивості **Name** — **L12**, **L13**, **L14** та **L15** відповідно.

г) Встановіть однакові відстані між об'єктами, обравши в меню **Формат Інтервал по горизонталі** ▶ **Сделать равным** (Інтервал по горизонталі ▶ Зробити рівним).

ґ) Виділіть створені об'єкти та створіть їх копії, перетягуючи вниз та утримуючи ліву клавішу миші **+Ctrl**.

д) Надайте копіям другого ряду імена **L21**, ..., **L25**, копіям третього ряду — **L31**, ..., **L35**. Для четвертого та п'ятого ряду задайте імена аналогічно.

е) У нижній частині форми додайте ще три об'єкти **Label**, встановлюючи їхні властивості самостійно.

3. Задайте коди для об'єкта **L11**, при натисненні на який його колір змінюється на червоний, а при подвійному натисненні — на зелений.

а) Введіть код для події клацання на кнопці:

```
L11.BackColor = Color.Red
```

б) У вікні коду в списку праворуч оберіть подію **DoubleClick** — подвійне клацання на об'єкті. В результаті чого мають бути сформовані два рядки програмного коду: заголовок і кінець процедури.

```
Private Sub L11_DoubleClick(sender As Object, e As EventArgs)
    Handles L11.DoubleClick
```

```
End Sub
```

в) Між даними рядками введіть такий код:

```
L11.BackColor = Color.Green
```


4. Аналогічно створіть коди для інших об'єктів ігрового поля.
5. Самостійно задайте програмні коди для виведення текстових повідомлень.
6. Пограйте у створену вами гру з однокласником.
7. Збережіть проект під назвою **ПР8_Прізвище**.

§27

Елементи керування «напис» і «кнопка»

У цьому параграфі основну увагу буде приділено таким елементам керування, як **Label** (напис) та **Button** (кнопка). **Label** — елемент керування призначений для створення написів на формі. **Button** — елемент для відображення кнопок. Деякі основні властивості цих елементів керування вам відомі.



Дізнатися більше
про властивості
елементів керування
Label і **Button** ви зможете
на диску. 

Нагадаємо, щоб вивести текст на ці об'єкти, треба змінити властивість **Text**, а за допомогою властивості **Font** можна змінити параметри шрифту. Властивість **BackColor** змінює колір фону об'єктів, а **ForeColor** — колір розташованого на них тексту. Часто виникає потреба змінити розмір елемента керування **Label**. Щоб це зробити, потрібно значення властивості **AutoSize** встановити **False**.

У таблицях 5.5 і 5.6 наведено деякі властивості, які притаманні переважно цим елементам керування.

Таблиця 5.5. Основні властивості елемента керування **Label**

Властивості	Пояснення
BorderStyle	Тип межі
Image	Зображення, що буде відображено на мітці
TextAlign	Вирівнювання тексту відносно рамки мітки (використовується коли AutoSize = False)

Таблиця 5.6. Основні властивості елемента керування **Button**

Властивості	Пояснення
FlatStyle	Визначає загальний вигляд кнопки, вигляд при наведенні та клацанні
FlatAppearance	Комбінована властивість для встановлення параметрів кнопки в разі встановлення FlatStyle = Flat
BackgroundImage	Встановлення зображення на кнопці в разі встановлення FlatStyle = Flat
BackgroundImageLayout	Спосіб розташування зображення на кнопці

ВПРАВА 28



Завдання. Створити проект, який містить стовпчики з українськими та англійськими словами. При запуску програми англійські слова мають бути невидимими. При натисненні на українське слово відповідне йому

англійське слово стає видимим, а при подвійному натисненні на українське слово — англійське слово стає знову невидимим.

1. Створіть форму за зразком (рис. 5.20), розташувавши на формі шість елементів керування **Label**.
2. Змініть для цих елементів керування значення властивості **Name** на ті, що вказані у виносках зображення форми (L1, ..., L6), розмір шрифту встановіть — 18, інші властивості змініть згідно значень наведених у таблиці нижче.

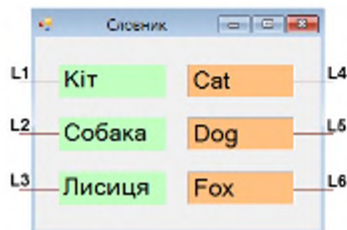


Рис. 5.20

Властивість Ім'я об'єкта	Text	AutoSize	BorderStyle	Visible	BackColor
L1	Кіт	False			Зелений
L2	Собака	False			Зелений
L3	Лисиця	False			Зелений
L4	Cat	False	Fixed3d	False	Оранжевий
L5	Dog	False	Fixed3d	False	Оранжевий
L6	Fox	False	Fixed3d	False	Оранжевий

3. Для об'єкта L1 створіть процедуру для відображення об'єкта L4 у разі натиснення на L1.
`L4.Visible = True`
4. Створіть процедуру, щоб при натисненні на L1 мітка L4 ставала б невидимою. Для цього у вікні коду для події **DoubleClick** введіть програмний код:
`L4.Visible = False`
5. Створіть аналогічно коди для інших міток.
6. Збережіть даний проект під назвою **Вправа_28_Прізвище**.

ВПРАВА 29



Завдання. Створити проект, у якому на формі розташувати чотири кнопки, що відповідають порам року та одну мітку (**Label**) за зразком (рис. 5.21). При клацанні на кнопку в мітці мають відобразитися назви місяців, що відповідають цій порі року, розташовані у стовпчик. Колір літер повинен змінюватися відповідно до кольору кнопки, на яку було натиснуто. При початковому завантаженні форми мітка має бути порожньою.

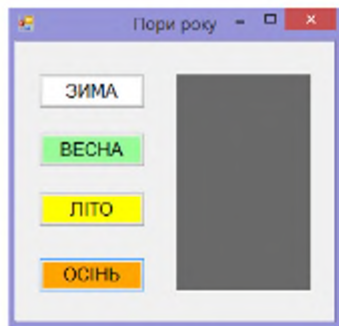


Рис. 5.21

1. Створіть форму за зразком, розташували на ній чотири кнопки та одну мітку.
2. Задайте властивості елементам керування, узявши значення з таблиці наведеної нижче.

Елемент керування	Name	Text	BackColor	Font Size	AutoSize	BorderStyle
Button1	Z	Зима	White	12		
Button2	V	Весна	PaleGreen	12		
Button3	L	Літо	Yellow	12		
Button4	O	Осінь	Orange	12		
Label1	Month		DimGray	12	False	Fixed3D

3. Клацніть два рази на кнопку **Z** та введіть код, що відповідає події клацання на кнопці.
 'Надання напису значення переліку місяців,
 'де кожний місяць записаний з нового рядка
 Month.Text = «Грудень» + vbNewLine + «Січень» + vbNewLine + «Лютий»
 'Встановлення білого кольору літер
 Month.ForeColor = Color.White
 Тут vbNewLine — команда, що здійснює перехід на новий рядок.
4. Для інших кнопок коди задайте аналогічно.
5. Запустіть проект та переконайтеся, що він працює правильно.
6. Збережіть проект під назвою **Вправа_29_Прізвище**.



Висновки

Щоб створити текстовий напис на формі, використовують елемент керування **Label**.

Щоб розташувати на формі кнопки, застосовують елемент керування **Button**.

Щоб встановити текст на ці об'єкти, треба змінити властивість **Text**.

Font — властивість, що дозволяє змінювати параметри шрифту.



Контрольні запитання та завдання

1. Яке призначення елементів **Label**? **Button**?
2. Які основні властивості притаманні елементам керування **Label**?
3. Які основні властивості притаманні елементам керування **Button**?



Питання для роздумів

- 1*. Як розташувати у **Button** текст у декілька рядків?
- 2*. У мітці **Label1** записано число 5, а в мітці **Label2** записано число 3. Що буде записано в мітці **Label3** після виконання програмного коду: `Label3.Text = Label1.Text + Label2.Text`.



Завдання для досліджень

- 1*. Перегляньте у вікні **Свойства** послідовно всі властивості елементів керування **Label** і **Button**. Встановіть їх призначення.

ПРАКТИЧНА РОБОТА №9

Створення програми з кнопками та написами

Завдання. Створити проект, що має відтворювати дії телефону (рис. 5.22). При введенні телефонного номеру він послідовно набирається у вікні об'єкта **Label**. При натисненні на зелену кнопку (Кнопку виклику) всі кнопки та об'єкт **Label** стають неактивними. При натисненні на червону кнопку (Кнопку відміни виклику) всі кнопки стають активними і набраний номер витирається.

1. Створіть форму із заголовком «Телефон». Встановіть для форми властивості:

FormBorderStyle = FixedToolWindow,

StartPosition = WindowScreen.

2. Розташуйте на формі кнопки та мітку за зразком.
3. Встановіть початкові властивості для кнопок з цифрами та символами, узявши значення з таблиці нижче.




Рис. 5.22

Властивість	Значення
Name	B1, B2, ..., B9, B0, BZ (для зірочки), BR (для решітки)
Text	1, 2, ..., 9, 0, *, #
BackColor	SkyBlue
Font	ComicSansMS, 14
Size	45; 45
FlatStyle	Flat
FlatAppearance — BorderColor	White
FlatAppearance — MouseDownBackColor	LightBlue
FlatAppearance — MouseOverBackColor	DeepSkyBlue

- Для зеленої та червоної кнопок встановіть імена **BGreen** і **BRed** відповідно. Заберіть текст і змініть колір фону.
- Для мітки встановіть властивості згідно наведеної нижче таблиці.




Виконати вправи «Дата народження», «Світлофор» і «Улюблена тварина» ви зможете на диску. 

Властивість	Значення
Name	L
Text	Забрати
AutoSize	False
BorderStyle	Fixed3D
Font	ComicSansMS, 12

- Для кнопки **B1** введіть програмний код для події клацання на ній.
 $L.Text = L.Text + «1»$
 У цьому коді спочатку буде виконано дію $L.Text + «1»$, що означає приєднання до текстового напису мітки **L** цифри «1». Після чого текстовий напис мітки **L** замінюється новим значенням.
 Тобто якщо в **L** містився напис «55» після виконання команди $L.Text + «1»$ значення буде «551».
 Замість розглянутого вище коду можна також записати:
 $L.Text = L.Text + B1.Text$
 Тут **B1.Text** має те значення, що розташовано на кнопці, а саме «1».
- Коди для кнопок **B2**, **B3**, ..., **B9**, **B0**, **BZ** та **BR** введіть самостійно.
- Задайте коди для кнопки **BGreen**.



Перевірити свої знання, розгадати кросворд і пройти тест ви зможете на диску. 

```
B0.Enabled = False
B1.Enabled = False
...
BR.Enabled = False
BZ.Enabled = False
```

- Задайте коди для кнопки **BRed** самостійно.
- Запустіть проект і перевірте правильність його виконання.
- Збережіть проект під назвою **ПР9_Прізвище**.

Розділ 6

Алгоритми роботи з об'єктами та величинами

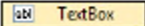


§28

Елементи для введення даних. Текстове поле

28.1.

ВЛАСТИВОСТІ ТЕКСТОВОГО ПОЛЯ



Для обробки певної інформації в середовищі VB.NET необхідно навчитися вводити дані до пам'яті комп'ютера. Нерідко для цього використовують ряд елементів керування, про можливості яких мова піде далі. Найчастіше вживаним елементом введення даних є текстове поле, що утворюється на формі за допомогою елемента керування  **TextBox**.

Деякі властивості його вам уже відомі, оскільки вони характерні й для багатьох інших елементів керування — **Text**, **BackColor**, **ForeColor**, **Font**, **Visible**, **Enabled**, **AutoSize**. Властивості, які притаманні переважно текстовому полю наведено в табл. 6.1.

Таблиця 6.1. Основні властивості елемента керування **TextBox**

Властивість текстового поля	Пояснення
MaxLength	максимальна довжина
MultiLine	можливість розташування тексту в кілька рядків
PasswordChar	символ пароля
ReadOnly	можливість змінювати текстову інформацію
ScrollBars	встановлення лінійок прокрутки
TextAlign	вирівнювання тексту
WordWrap	визначає, чи автоматично переносити текст на новий рядок

TextBox у режимі конструктора має своє меню (рис. 6.1), завдяки якому можна встановити одну з важливих його властивостей — **MultiLine** (розміщення тексту в декілька рядків).

 **Дізнатися більше**
про основні властивості елемента керування **TextBox** ви зможете на диску. 

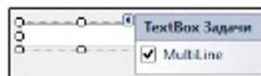


Рис. 6.1

28.2.

ВСТАНОВЛЕННЯ ВЛАСТИВОСТЕЙ В ПРОГРАМНОМУ КОДІ

Змінюють властивості текстового поля в програмному кодї подібно, як і для інших елементів керування. Одним з основних призначень текстового поля є відображення в ньому текстової інформації. Розглянемо деякі особливості програмних кодів для редагування текстових повідомлень.

Щоб ввести текст у текстовому полі, програмний код має такий вигляд:
`TextBox1.Text = «Текстове повідомлення»`

Якщо в текстове поле необхідно дописати деякий текст, то для цього застосовують код наступного змісту:

`TextBox1.Text = TextBox1.Text + «Текстове повідомлення»`

Якщо є потреба розташувати текст у декілька рядків, у режимі конструювання встановлюють властивість `MultiLine = True`. А в програмному кодї, щоб дописати текст з наступного рядка, додають команду `vbNewLine`.

‘Перейти на наступний рядок та дописати текст «Текстове повідомлення»

`TextBox1.Text = TextBox1.Text + vbNewLine + «Текстове повідомлення»`

‘Дописати текст «Текстове повідомлення» та перейти на наступний рядок

`TextBox1.Text = TextBox1.Text + «Текстове повідомлення» + vbNewLine`

Докладніше з можливостями використання текстових полів ви ознайомитеся, виконуючи практичні вправи.

Часто при виконанні програмного коду виникає потреба очистити текст від напису. Це можна зробити двома способами:

`TextBox1.Text = «»` або `TextBox1.Text = Nothing`

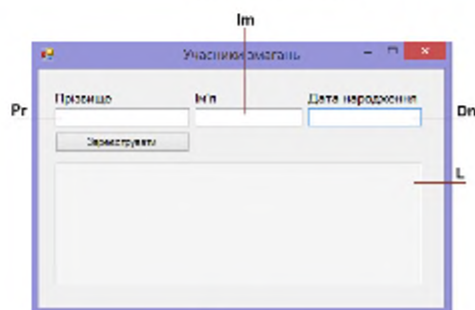


Рис. 6.2

ВПРАВА 30



Завдання. Створити проект, що містить форму реєстрації учасників змагаль за зразком (рис. 6.2). Після введення даних у текстові поля та натиснення кнопки **Зареєструвати** дані в рядок записуються до текстового напису розташованого під текстовими полями, а текстові поля очищуються від написів. При наступному введенні даних і натисненні кнопки **Зареєструвати** дані розташовуються під попереднім записом.

- Створіть форму за зразком (рис. 6.2), встановіть початкові властивості елементів керування самостійно. Задайте елементам керування імена, вказані у виносках.
- Введіть код для кнопки **Зареєструвати** для події `Click`.
 ‘Дописати текст до попереднього та перейти на наступний рядок
`L.Text = L.Text + Pr.Text + « » + Im.Text + « » + Dn.Text + vbNewLine`
 ‘Очищення текстових полів від написів



Pr.Text = Nothing

Im.Text = Nothing

Dn.Text = Nothing

- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_30_Прізвище**.

ВПРАВА 31



Завдання. Створити форму із заголовком «Текстові вікна та мітки» та виконати вказані нижче дії.

- Розмістіть у вікні форми два текстові поля і дві мітки.
- Надайте текстовим полям імена **T1** і **T2**, а міткам — **L1** і **L2**.
- Очистіть від тексту всі написи на створених об'єктах. Для міток встановіть властивість **AutoSize** = False, **BorderStyle** = Fixed3d.
- Здайте вирівнювання в текстових полях по центру, а в мітках — до правого краю.
- Встановіть у текстових полях розмір шрифту — 14 пт, колір — червоний; в мітках розмір шрифту — 12 пт, колір — синій.
- Зробіть так, щоб при введенні тексту до текстового поля **T1** він відображався в полі мітки **L1**, аналогічно текст поля **T2** — у **L2**.
Значення властивості **Text** об'єкта **L1** повинно дорівнювати значенню властивості **Text** об'єкта **T1** ($L1.Text = T1.Text$).
- Зробіть так, щоб при натисненні лівої клавіші миші на **L1** текст з поля **T1** вилучався ($T1.Text = Nothing$).
- Зробіть так, щоб при подвійному натисканні на **L2** текст у полі **T2** вирівнювався по лівому краю ($T2.TextAlign = HorizontalAlignment.Right$).
- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_31_Прізвище**.

ВПРАВА 32

САМОСТІЙНО



Завдання. Створити проект, що містить реєстраційну форму працівника за зразком (рис. 6.3). Після введення даних до відповідного текстового поля попереднє поле стає неактивним. Після введення даних до останнього текстового поля і одноразового клацання лівою клавішею миші на відповідній мітці текстове поле стає неактивним. Для зміни напису у відповідних текстових полях слід клацнути два рази на відповідні їм мітки. (При переході від одного текстового поля до іншого слід застосувати подію зміни фокусу **LostFocus**). Зберегти проект під назвою **Вправа_32_Прізвище**.

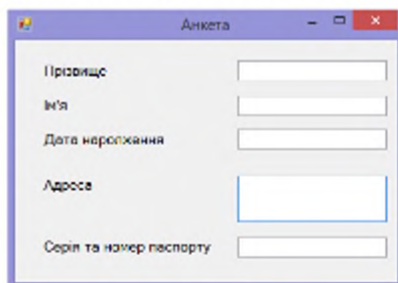


Рис. 6.3

**Висновки**

Елемент керування **TextBox** призначений для введення та виведення даних.

Щоб розташувати текст у два рядки, використовують властивість **MultiLine**.

Щоб ввести текст у текстовому полі, після знака «=» записують потрібний текст виділений лапками.

Щоб дописати текст, застосовують код такого змісту:

```
TextBox1.Text = TextBox1.Text + «Текстове повідомлення»
```

Щоб перемістити текст на наступний рядок, додають команду **vbNewLine**.

Очищують текстове поле від напису, застосовуючи такий програмний код:

```
TextBox1.Text = «» або TextBox1.Text = Nothing
```

**Контрольні запитання та завдання**

1. За допомогою якого елемента керування можна створити текстове поле?
2. За допомогою якої властивості можна змінити текстовий напис?
3. Назвіть способи зміни розташування тексту в декілька рядків.
4. Якою командою можна переводити текст на наступний рядок?
5. Як очистити текстове поле від написів?

**Питання для роздумів**

- 1*. Для чого використовують властивість **MaxLength**? У яких випадках використовують цю властивість?
- 2*. Текстове поле можна використовувати не тільки для введення даних, а й для виведення. Що потрібно зробити, щоб дані, виведені як результат, не можна було змінити?

**Завдання для досліджень**

- 1**. У текстових полях **T1** і **T2** містяться деякі написи. При натисненні на кнопку ці написи мають помінятися місцями. Який програмний код для цього необхідно записати?



§29



Використання перемикачів і прапорців

Уводити дані можна, не лише набираючи текст або числа до текстового поля, а й обираючи один із запропонованих програмою варіантів. Це можна робити за допомогою так званих *перемикачів* і *прапорців*, які створюють за допомогою елементів керування **RadioButton** та **CheckBox** відповідно. Як правило, ці елементи керування використовують, щоб створити форми для тестування, опитування, голосування тощо.

29.1.

ВИКОРИСТАННЯ ПЕРЕМИКАЧІВ

Елементи керування класу «перемикачі» **RadioButton** дають змогу вибирати на формі лише одне значення з деякого набору запропонованих значень (рис. 6.4).

Даний елемент керування може бути в одному з двох станів: перемикач увімкнено — ; перемикач вимкнено — .

Вибираючи один з виглядів перемикача, змінюється його властивість **Checked**. Якщо перемикач увімкнено, то **Checked = True**, якщо вимкнено — **Checked = False**.

Якщо потрібно на формі розташувати декілька груп з перемикачами, для цього використовують елемент керування **GroupBox**. Прикладом такого групування може бути форма, що містить декілька запитань на одній формі (рис. 6.5).

Групування елементів необхідно передбачити заздалегідь, до моменту вставлення їх на форму. Спочатку на формі слід розташувати елементи керування **GroupBox**, а потім на них розташувати елементи керування **RadioButton**. Основною властивістю елемента **GroupBox** є його підпис, який встановлює користувач, використовуючи властивість **Text**.

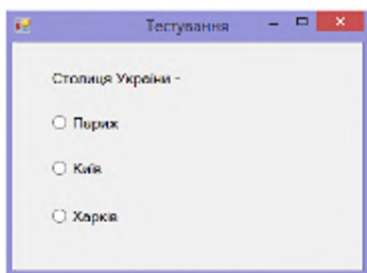


Рис. 6.4

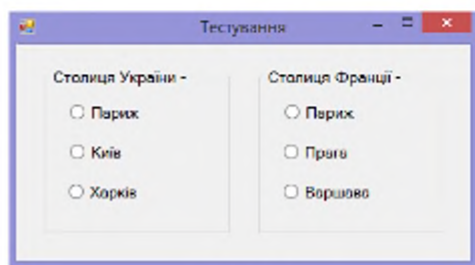


Рис. 6.5

ВПРАВА 33



Завдання. Створити проект «Найбільші міста України», в якому можна буде вибрати інформацію про одне з найбільших міст України. Після вибору міста в текстовому полі буде з'являтися інформація про рік його заснування, кількість населення та площу. Інформацію про міста можна знайти в мережі Інтернет або завантажити з файла **Вправа_33** у папці **Files** на диску.

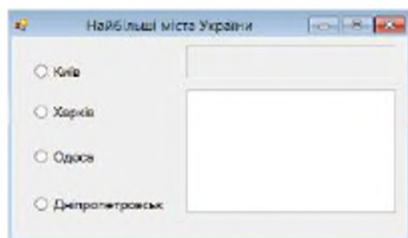


Рис. 6.6

1. Створіть форму за зразком (рис. 6.6). Задайте перемикачам імена **R1**, **R2**, **R3** і **R4** відповідно. Мітці розташованій над текстовим полем надайте ім'я **L**, для текстового поля — **T**.
2. Задайте коди, щоб відображалася інформація про місто, вибране за допомогою перемикача.
 - а) Клацніть два рази на перемикачі **R1**. У результаті має з'явитися вікно коду, де буде

відображено заголовок процедури **CheckedChanged** (Зміна значення властивості **Checked**), що є подією за умовчанням для елемента керування **RadioButton**.

б) Введіть код для даної процедури.

```
'Надання об'єкту Label тексту, що має R1, а саме слова «Київ»
L.Text = R1.Text
```

```
'Формування та виведення тексту в текстовому полі
```

```
T.Text = «Засноване у V столітті» + vbNewLine +
«Кількість населення - 2904000» + vbNewLine + «Площа - 847,67 км кв.»
```

в) Для інших перемикачів задайте коди самостійно.

3. Запустіть проект та перевірте правильність його виконання.

Примітка. Часто після запуску проекту один з елементів керування **RadioButton** знаходиться вже в увімкненому стані, хоча за умовчанням у всіх перемикачів значення властивості **Checked** = **False**. Щоб такого не траплялося і всі перемикачі при запуску проекту були вимкнені, потрібно під час конструювання проекту для кожного **RadioButton** змінити значення властивості **Checked** спочатку на **True**, а потім знову на **False**.

4. Збережіть проект під назвою **Вправа_33_Прізвище**.

29.2.

ВИКОРИСТАННЯ ПРАПОРЦІВ

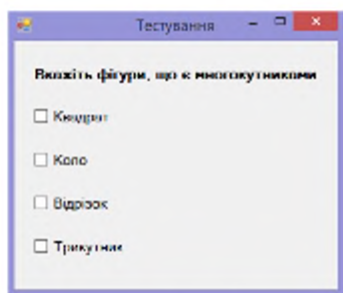


Рис. 6.7

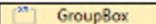
Елементи керування класу «прапорці» 

CheckBox використовують для того, щоб користувач мав змогу обрати одне або декілька значень з деякого набору пропонувананих значень (рис. 6.7).

Даний елемент керування може бути в одному з трьох станів, які можна встановлювати під час конструювання, змінюючи властивість **CheckState**:

- перемикач увімкнено (**CheckState** = **Checked**);
- перемикач вимкнено (**CheckState** = **Unchecked**);
- перемикач не має визначеного стану (**CheckState** = **Indeterminate**).

Після запуску проекту відображення прапорців можна змінити на стан увімкнено або вимкнено . У результаті чого змінюється властивість **Checked**, що набуває значення **True** або **False** відповідно.

Розташовуючи групи прапорців, як і у випадку з перемикачами, застосовують елемент керування  **GroupBox**.



ВПРАВА 34



Завдання. Створити проект «Дитяча кав'ярня «Золотий ключик»» (рис. 6.8), у якому користувач має обрати бажані страви, натискаючи на відповідні прапорці, що знаходяться біля їхніх назв. Встановлюючи прапорець, відповідний елемент керування стає неактивним. При натисненні на кнопку **Редагувати замовлення** прапорці стають порожніми та активними. Після вибору прапорців і натиснення на кнопку **Сформувати замовлення** з'являється вікно з текстовим повідомленням «Замовлення» (рис. 6.8), що містить перелік обраних страв.

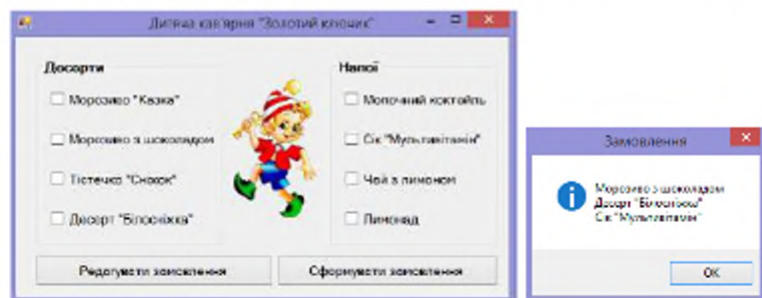


Рис. 6.8. Приклад форми проекту і зразок вікна повідомлення проекту

1. Розташуйте на формі елементи керування за зразком (рис. 6.8) самостійно. Спочатку додайте два елементи керування **GroupBox**, у яких потім розташуйте прапорці.
2. Задайте прапорцям імена **C1**, ..., **C8** відповідно.
3. У довільному місці форми додайте елемент керування **Label**, для якого надайте ім'я **Zam**, властивість **Text** залишіть у ньому порожньою, для властивості **Visible** встановіть значення **False**.

Даний елемент керування буде невидимим, але в ньому буде формуватися текст, який після натиснення кнопки **Сформувати замовлення** буде розташовано у вікні повідомлення.

4. Задайте коди, щоб при виборі страви за допомогою прапорця значення його тексту додавалося до об'єкта **Zam**, а сам прапорець ставав неактивним.

а) Клацніть два рази на прапорці **C1**. У результаті має відобразитися вікно коду, де буде записано заголовок процедури **CheckedChanged**, що є подією за умовчанням для елемента керування **CheckBox**.

б) Введіть код для даної процедури.

‘Додавання до текстового напису значення тексту обраного прапорця

```
L.Text = L.Text + C1.Text + vbCrLf
```

‘Прапорець стає неактивним

```
C1.Enabled = False
```

в) Для інших прапорців задайте коди самостійно.



5. Задайте код для кнопки **Редагувати замовлення**.
 - а) Введіть код, який буде робити прапорець порожнім та активним.
`'Прапорець стає порожнім`
`C1.Checked = False`
`'Прапорець стає активним`
`C1.Enabled = True`
 - б) Для інших прапорців допишіть код аналогічно.
 - в) Запишіть код для очищення об'єкта **Zam** від попередніх записів.
`'Очищення текстового напису`
`Zam.Text = Nothing`
6. Створіть код для кнопки **Сформувати замовлення**, у результаті якого має з'явитися текстове повідомлення з обраними стравами.
`MsgBox(Zam.Text, 0 + 64, «Замовлення»)`
У даному коді `Zam.Text` — текст, сформований в об'єкті **Zam**, `0+64` — кнопка **Ok** і значок «Інформація», `«Замовлення»` — заголовок текстового повідомлення.
7. Запустіть проект та переконайтеся у правильності його виконання.
8. Збережіть проект під назвою **Вправа_34_Прізвище**.



Висновки

Щоб вибрати на формі лише один варіант з деякого набору запропонованих значень, використовують *перемикачі*, які встановлюють за допомогою елементів керування **RadioButton**.

Якщо в програмі виникає потреба вибрати один або більше варіантів з деякого набору запропонованих значень, використовують прапорці, які встановлюють на формі за допомогою елементів керування **CheckBox**.

Основною властивістю перемикачів і прапорців є властивість **Checked**. Якщо перемикач увімкнено, то **Checked = True**, якщо вимкнено — **Checked = False**.

Якщо на формі потрібно розташувати декілька груп, застосовуючи перемикачі або прапорці, використовують елемент керування **GroupBox**.



Контрольні запитання та завдання

1. Для чого використовують перемикачі та прапорці?
2. За допомогою яких елементів керування можна встановити перемикачі та прапорці?
3. Яка властивість визначає стан перемикачів?
4. У яких станах може знаходитися елемент керування **CheckBox**? Завдяки якій властивості ці стани встановлюють?
5. Як розташувати на формі декілька груп перемикачів або прапорців?



Питання для роздумів

- 1*. Вкажіть відмінності у використанні перемикачів і прапорців?
- 2*. Наведіть приклади використання перемикачів (прапорців).



Завдання для досліджень

- 1*. Наведіть приклади використання перемикачів (прапорців) у текстовому редакторі MS Word.
- 2*. Встановіть залежність між властивостями **CheckState** та **Checked** для прапорців. Які зміни відбуваються з властивістю **CheckState** при зміні властивості **Checked**?

§30

Списки

Ще одним засобом для введення та виведення даних є списки (рис. 6.9). У VB.NET використовують два елементи керування для відтворення списків: **ListBox**, що відтворює звичайні списки; **ComboBox**, який створює випадаючі списки.

У списки можна сформувати назви річок, прізвища, розміри шрифтів, оцінки учнів тощо.

Кожен елемент списку має свій номер. Нумерація в списках починається з нуля.

Формувати список можна як у режимі конструктора, так і під час виконання програмного коду. До списку можна додавати елементи, вилучати елементи, повністю очищати список.

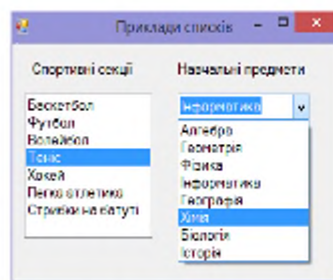


Рис. 6.9. Приклади списків

30.1.

ЕЛЕМЕНТ КЕРУВАННЯ LISTBOX

Елемент керування **ListBox** дозволяє створити звичайний список, елементи якого завжди відображені на формі. В деяких випадках, щоб переглянути елементи списку, потрібно застосувати смугу прокручування.

Щоб додати значення до списку, є декілька способів.

У режимі конструктора відкрити діалогове вікно у властивості **Items** та ввести елементи списку (кожний елемент списку з нового рядка).

Це ж діалогове вікно можна відкрити в меню елемента керування **ListBox**, обравши команду **Изменить элементы...** (Змінити елементи) (рис. 6.10).

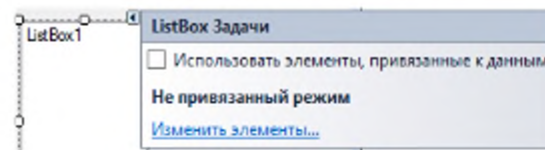


Рис. 6.10



Для опрацювання списків за допомогою програмних кодів використовують методи, можливості яких представлено в табл. 6.2.

Таблиця 6.2. Методи для опрацювання списків

Метод	Опис	Приклад
Add	Додавання значення в кінець списку	ListBox.Items.Add(«січень»)
AddRange	Додавання значень усього списку	Listbox1.Items.AddRange(ListBox2.Items)
Insert	Додавання значення під номером n із розсуванням списку	Listbox.Items.Insert(0, «лютий»)
Clear	Очищення списку	Listbox.Items.Clear()
RemoveAt	Вилучення елемента за вказаним номером	Listbox.Items.RemoveAt(0)
Remove	Вилучення елемента за вказаним значенням	Listbox.Items.Remove(«лютий»)

Щоб визначити номер обраного мишкою елемента списку, використовують властивість **SelectedIndex**, а для виведення самого значення елемента списку — властивість **SelectedItem**.

Щоб визначити значення елемента списку за його номером, використовують команду `Listbox.Items.Item(n)`, де n — номер елемента списку.

Можливості створення та використання списків розглянемо, виконуючи вправи.

ВПРАВА 35

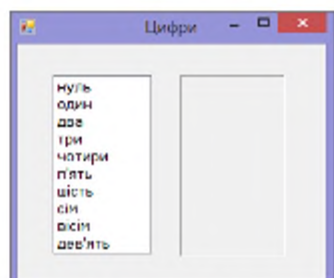


Рис. 6.11



Завдання. Створити проект «Цифри», в якому об'єкт «список» містить перелік цифр записаний у вигляді тексту («нуль», «один», «два» і т. д.). При виборі елемента списку в текстовому написі розташованому поряд із списком записується відповідне значення у вигляді цифри (рис. 6.11).

1. Розташуйте на формі елементи керування **ListBox** і **Label**. Надайте їм імена **LB** і **Cifra** відповідно. Задайте необхідні властивості елементам керування.
2. У режимі конструктора створіть список із значеннями «нуль», «один», «два» і т. д.
3. Введіть код, який здійснює реалізацію проекту.
 - а) Клацніть два рази на об'єкті список. У результаті має відобразитися вікно коду, де буде записано заголовки



Виконати вправу «Формування покупки» ви зможете на диску.



процедури `SelectedIndexChanged`, що є подією за умовчанням для елемента керування `ListBox`.

б) Введіть код для даної процедури.

‘Надання напису значення номеру елемента списку


```
Cifra.Text = Lb.SelectedIndex
```

4. Запустіть проект та переконайтеся у правильності його виконання.

5. Збережіть проект під назвою **Вправа_35_Прізвище**.

30.2.

ЕЛЕМЕНТ КЕРУВАННЯ COMBOBOX

Ще один вид списків реалізовано за допомогою елемента керування  `ComboBox`, що має вигляд випадального списку. Список розгортається після натиснення кнопки зі стрілкою. Після вибору елемента зі списку він згортається, а вибраний елемент розташовується на текстовому полі.

Для створення та обробки елементів списку використовують ті самі властивості і методи, що і для елемента керування `ListBox`. На відміну від об'єкта `ListBox`, у `ComboBox` є властивість `Text`, що визначає значення тексту, який відображено в ньому, коли список згорнутий.

ВПРАВА 36



Завдання. Створити проект «Оцінки» (рис. 6.12), в якому в звичайному списку міститься перелік учнів, у випадальних списках — перелік навчальних предметів і кількість балів. Після вибору прізвища інформація розташовується в об'єкті «напис» і список стає неактивним. Після вибору предмета та оцінки інформація потрапляє до текстового поля і здійснюється перехід на наступний рядок, значення обрані для наступного предмета та оцінки додаються в текстовому полі нижче. Кнопка **Редагувати** робить список з учнями активним, очищує напис, текстове поле та обрані значення у випадальних списках. Кнопка **Надіслати результат** виводить текстове повідомлення (рис. 6.12), заголовком якого є прізвище та ім'я учня, а повідомлення містить інформацію про його успішність.

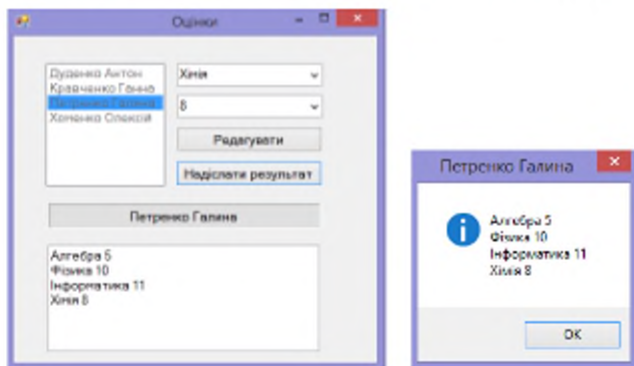


Рис. 6.12. Зразок форми проекту і зразок текстового повідомлення проекту



1. Створіть форму за зразком (рис. 6.12, а), розташувавши на ній один список, два випадючі списки, дві кнопки, напис і текстове поле.
2. Список назвіть ім'ям **LB** і введіть декілька прізвищ та імен учнів.
3. Для випадючих списків задайте імена **CB1** і **CB2** відповідно. До списку **CB1** введіть перелік навчальних предметів, до **CB2** — оцінки від 1 до 12.
4. Напису надайте ім'я **PR**, текстовому полю — **Ocin**, для текстового поля властивість **ReadOnly** (Тільки для читання) встановіть **True**. Інші властивості елементів керування встановіть самостійно.
5. Введіть код для події вибору значення зі списку учнів. У результаті чого прізвище та ім'я учня мають відобразитися в об'єкті «напис», а список повинен стати неактивним.
 - а) Клацніть два рази на об'єкт **ListBox**.
 - б) Введіть код для даної процедури.


```
'Запис обраного прізвища та імені до об'єкта напис
Pr.Text = LB.SelectedItem
'Список стає неактивним
LB.Enabled = False
```
6. Введіть код для події вибору значень із випадючих списків, після чого вони записуються до текстового поля.
 - а) Клацніть два рази на об'єкт **CB2**. У результаті має відобразитися вікно коду, де буде записано заголовок процедури **SelectedIndexChanged**, що є подією за умовчанням для елемента керування **ComboBox**.
 - б) Введіть код для даної процедури.

```
Ocin.Text = Ocin.Text + CB1.Text + « » + CB2.Text + vbNewLine
```

Подія має відбуватися після покрокового вибору значень спочатку в списку **CB1**, а потім — у **CB2**. Отже, з даного коду видно, що до попереднього тексту буде додано текстове значення поля **CB1**, далі через пробіл додано значення текстового поля **CB2** і здійснено перехід на наступний рядок завдяки команді **vbNewLine**.
7. Введіть код для кнопки **Редагувати** самостійно. В результаті список має стати активним, напис, текстове поле та значення, обрані у випадючих списках, — порожніми.
8. Введіть код для кнопки **Надіслати результат**, що виводить текстове повідомлення, в якому заголовком є прізвище та ім'я учня, а текст повідомлення містить інформацію про його успішність.

```
MsgBox(Ocin.Text, 0 + 64, Pr.Text)
```
9. Запустіть проект та переконайтеся у правильності його виконання.
10. Збережіть проект під назвою **Вправа_36_Прізвище**.



Виконати вправу
«Закупівля» і «Розклад»
ви зможете на диску .



Висновки

Для створення списків використовують елементи керування **ListBox** і **ComboBox**.

Елемент керування **ListBox** дозволяє створити звичайний список, елементи якого завжди відображені на формі. Елемент керування **ComboBox** дозволяє сформувати випадаючий список.

Щоб сформувати список у режимі конструктора, використовують властивість **Items**.

Для опрацювання списків за допомогою програмних кодів використовують такі методи: **Add** (додавання елемента), **AddRange** (додавання списку), **Insert** (додавання елемента на вказане місце), **Clear** (очищення списку), **RemoveAt** (вилучення вказаного за номером елемента), **Remove** (вилучення вказаного елемента).

Щоб визначити номер обраного мишею елемента списку, використовують властивість **SelectedIndex**, а щоб вивести саме значення елемента списку — властивість **SelectedItem**.

Щоб визначити значення елемента списку за його номером, використовують команду `ListBox.Items.Item(n)`, де n — номер елемента списку.



Контрольні запитання та завдання

1. Для чого використовують списки?
2. За допомогою яких елементів керування можна створити списки?
3. Як ввести елементи списку в режимі конструктора?
4. Як додати елементи списку, використовуючи програмний код?
5. Вкажіть методи, які дозволяють вилучити елементи зі списку.
6. За допомогою якої властивості можна визначити значення обраного в списку елемента? номер обраного в списку елемента?



Питання для роздумів

- 1*. Вкажіть відмінності у використанні елементів керування **ListBox** і **ComboBox**.



Завдання для досліджень

- 1*. Наведіть приклади використання списків у MS Excel.
- 2*. Визначте призначення властивості **Sorted**, яку використовують для елементів керування **ListBox** і **ComboBox**. У яких випадках зміна її значення є доцільною?



§31

Типи величин

31.1.

ОСНОВНІ ПОНЯТТЯ МОВИ ПРОГРАМУВАННЯ

Вам уже відомо, що однією із складових мови програмування є алфавіт. Алфавіт мови програмування VB.NET складається з таких символів:

- латинських літер та літер кирилиці;
- цифр від 0 до 9;
- символів арифметичних операцій (+, -, *, /, ^);
- символів відношення (=, <, > і т. ін.);
- розділових символів (крапка, кома, двокрапка і т. ін.);
- спеціальних символів (!, #, \$ і т. ін.)

Створюють програмний код у мові програмування за допомогою так званих службових слів, що мають фіксоване стандартне призначення.

Наприклад, **End** — кінець, **Sub** — процедура.

31.2.

ПОНЯТТЯ ВЕЛИЧИН

Об'єкти, над якими виконуються певні операції в середовищі Visual Basic, прийнято називати *величинами*. Величини характеризуються іменем, типом і значенням.

Величинами в мові програмування називають об'єкти, над якими виконуються певні операції.

Правила запису імені величин:

- першим символом в імені має бути літера;
- ім'я може складатися з букв, цифр та символу підкреслення «_»;
- ім'я має бути унікальним і не збігатися з ключовими словами (**Sub**, **End**, **Now** тощо);
- ім'я не може містити більше, ніж 255 символів.

Величини можуть бути різних типів, деякі основні типи величин представлені в табл. 6.3. Залежно від того, який тип має величина, для неї виділяється певний обсяг у байтах на ділянці пам'яті.

Таблиця 6.3. Основні типи величин

Тип	Опис	Діапазон значень	Приклади
Integer	Цілі числа	$\approx \pm 2$ млрд	-100, 0, 30000
Long	Довгі цілі числа	$\approx \pm 10^{17}$	-2000000000
Single	Десяткові числа до 6 знаків після крапки	$\approx \pm 10^{38}$	-3,58; 78,00001
String	Текстові (рядкові)	до 65000 символів	"Hello", "2455"
Boolean	Логічні величини	True або False	True, False



Величини, значення яких є сталими під час виконання програми, прийнято називати *константами*.

Для опису констант у програмному кодї записують команду:

```
Const <назва 1> = <значення 1>, <назва 2> = <значення 2>
```

Наприклад,

```
Const pi = 3.14, t = «Інформатика»
```

Величини, які під час виконання програми можуть набувати різних значень, називають *змінними*.

31.3.

ОГОЛОШЕННЯ ЗМІННИХ У ПРОГРАМІ

Перед використанням змінної її необхідно оголосити. Оголошують змінну, як правило, використовуючи команду **Dim**. Після переліку змінних вказується службове слово **As**, а за ним — тип змінних. Загальний вигляд оголошення змінних такий:

```
Dim <перелік змінних через кому> As <тип>
```

Якщо програма містить декілька типів змінних, тоді загальний вигляд їх буде такий:

```
Dim <перелік змінних 1> As <тип 1>, <перелік змінних 2> As <тип 2>  
або
```

```
Dim <перелік змінних 1> As <тип 1>
```

```
Dim <перелік змінних 2> As <тип 2>
```

Наприклад,

```
Dim N, S As Integer
```

```
Dim a1, a2 As Boolean, x2, zxc As Single
```

```
Dim ab2, Текст As String
```



Дізнатися більше

про запис змінних

ви зможете на диску.



Висновки

Об'єкти, над якими виконуються певні операції в середовищі програмування, називають *величинами*. Величини характеризуються іменем, типом і значенням.

Основні типи величин: **Integer** (цілі числа), **Single** (дійсні числа), **String** (текст), **Boolean** (логічний тип).

Величини, значення яких є сталими під час виконання програми, називають *константами*. Для опису констант у програмному кодї використовують команду **Const**.

Величини, які під час виконання програми можуть набувати різних значень, називають *змінними*. Оголошують змінну, використовуючи команду **Dim**.



Контрольні запитання та завдання

1. Що таке величина? Чим вона характеризується?
2. Які правила використовують для запису імені величин?
3. Назвіть основні типи величин.
4. Що таке константа? Як її описати?
5. Що таке змінна? Як описують змінні?



Питання для роздумів

- 1*. Навіщо використовують різні типи даних цілих чисел?
- 2*. Чи обмежена кількість змінних у програмі?



Завдання для досліджень

- 1*. Визначте, які ще типи змінних використовують у середовищі програмування. Вкажіть їх можливі значення та призначення.

§32

Присвоєння значень величинам

32.1.

ОПЕРАТОР ПРИСВОЄННЯ

Щоб надати змінній деякого значення, для цього використовують *оператор присвоєння*. Власне його можливості вам уже відомі з попереднього розділу. Нагадаємо його загальний вигляд та принцип його виконання.

Оператор присвоєння має вигляд:

$\langle \text{Змінна} \rangle = \langle \text{вираз} \rangle$

Виконуючи команду такого виду, програма спочатку виконує вираз, після чого присвоює його значення змінній, тобто записує значення виразу в ділянку пам'яті, що відведена для змінної.

Наприклад:

1. $x_1 = 5$ (змінна x_1 одержує значення 5);
2. $d = -d$ (значення змінної d буде замінено на протилежне за знаком значення);
3. $x = x + 1$ (поточне значення змінної x буде збільшено на 1).

Отже, з наведених прикладів видно, що кожне наступне присвоєння значення змінній стирає її попереднє значення.

Використовуючи оператори присвоєння, ви повинні звертати особливу увагу на те, щоб ліва і права частина оператора присвоєння мали однаковий тип даних.



Так, неправильною буде команда `t1.Visible = 5`, оскільки властивість **Visible** може набувати тільки значення `True` або `False`.

Зазначимо також, якщо змінна *a* була оголошена типу **Integer**, то в результаті виконання дії присвоєння `a = 2.3`, змінна набуває значення 2 (відбудеться округлення дробового числа до цілого).

Коли змінну оголошено, її можна використовувати в програмному коді. Перше присвоєння значення змінній називають *ініціалізацією*. Зручним способом є поєднання оголошення та ініціалізації змінної.

Програмні коди наведені нижче є ідентичними.

Оголошення змінної з наступною ініціалізацією	Поєднання оголошення та ініціалізації змінної
<code>Dim a As Integer a = 100</code>	<code>Dim a As Integer = 100</code>

Іноді, використання змінної, якій не було присвоєно початкове значення, може призвести до збою програми.

Для присвоєння змінним певних даних використовують елементи керування. Одним з поширених способів є використання текстового поля. Для присвоєння даних, що знаходяться в текстовому полі на формі, слід записати команду присвоєння, яка має такий загальний вигляд:

```
<Ім'я змінної> = <Ім'я текстового поля>.Text
```

Наприклад,

```
a = TextBox1.Text
```

Важливим також є можливість виведення значень змінних, що здійснюється переважно на елемент керування **Label** або у вигляді виведення текстового повідомлення.

Так, для присвоєння значення змінної елементові керування **Label**, треба записати команду, що має такий загальний вигляд:

```
<Ім'я мітки>.Text = <Ім'я змінної>
```

Наприклад,

```
Label1.Text = S
```

Для виведення значення змінної в текстове повідомлення, код записують таким чином:

```
MsgBox(S)
```

32.2.

ВВЕДЕННЯ ДАНИХ ЗАСОБАМИ ДІАЛОГОВОГО ВІКНА

Присвоїти змінним значення можна, ввівши дані за допомогою діалогового вікна, яке викликають, використовуючи вбудовану функцію **InputBox**.

Функція **InputBox** має такий синтаксис:

```
<змінна> = InputBox(<запрошення>, <заголовок>, <початкове значення>, <_<x-коорд. вікна>, <y-коорд. вікна>)
```

<змінна> — набуває значення виконання функції. Якщо введення значення закінчується натисненням **Ок**, то результатом функції є введене значення, а якщо **Отмена**, то результатом є порожній рядок.

<запрошення> — це будь-який текст, який є підказкою для користувача про те, яку інформацію він може ввести до спеціального поля введення, що знаходиться в цьому вікні.

<заголовок> — напис у рядку заголовка.

<початкове значення> — це значення, яке буде введено автоматично (за замовчуванням), якщо користувач погоджується з ним.

<x-коорд. вікна>, <y-коорд. вікна> — координати розташування вікна відносно екрана монітора.

При зверненні до функції **InputDialog** відкривається вікно (рис. 6.13), на якому виводиться **Запрошення** і **Заголовок**, а в текстовому полі відображається **Початкове значення**.

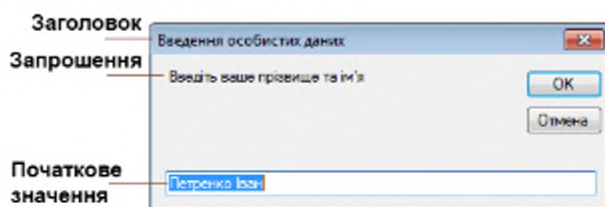


Рис. 6.13

Наприклад, виклик даного вікна може бути описано наступним чином:
`a = InputBox («Введіть ваше прізвище та ім'я», «Введення особистих даних»),`
`«Петренко Іван»)`

У функції **InputDialog** усі параметри, крім першого, можна не застосовувати. Для використання цієї функції достатньо запису скороченої форми:
`<змінна> = InputBox(<запрошення>)`

ВПРАВА 37

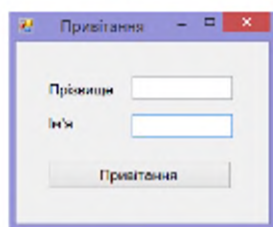


Рис. 6.14



Завдання. Створити проект, у якому деяким змінним мають бути присвоєні введені на формі прізвище та ім'я (рис. 6.14). Після натиснення на кнопку **Привітання** має з'явитися повідомлення у форматі «Привіт, <Ім'я> <Прізвище>».

1. Створіть форму за зразком (рис. 6.14). Надайте текстовим полям імена **T1** і **T2** відповідно. Інші властивості елементів керування встановіть самостійно.
2. Введіть код для кнопки **Привітання**.

```

'Оголошення змінних
Dim Pr, Im As String
'Присвоєння змінній Pr значення текстового поля T1
Pr = T1.Text
'Присвоєння змінній Im значення текстового поля T2
Im = T2.Text
    
```

**Виведення текстового повідомлення**

```
MsgBox(«Привіт, » & Im & « » & Pr, 0 + 32, «Привітання»)
```

- Запустіть проект та переконайтеся у правильності його виконання.
- Збережіть проект під назвою **Вправа_37_Прізвище**.

ВПРАВА 38

Завдання. Створити проект, у якому деяким змінним мають бути присвоєні введені за допомогою діалогових вікон прізвище та ім'я. Після натиснення на кнопку **Привітання** має з'явитися повідомлення у форматі «Привіт, <Ім'я> <Прізвище>».

- Розташуйте на формі лише одну кнопку **Привітання**.
- Введіть код для кнопки **Привітання**.

Оголошення змінних

```
Dim Pr, Im As String
```

Присвоєння значення змінній Pr

```
Pr = InputBox(«Введіть своє прізвище», «Введення прізвища»)
```

Присвоєння значення змінній Im

```
Im = InputBox(«Введіть своє ім'я», «Введення імені»)
```

Виведення текстового повідомлення

```
MsgBox(«Привіт, » & Im & « » & Pr, 0 + 32, «Привітання»)
```

- Запустіть проект та переконайтеся у правильності його виконання.
- Збережіть проект під назвою **Вправа_38_Прізвище**.

**Перевірити**

свої знання

і пройти тест ви зможете на диску.

**Висновки**

Щоб надати змінній деякого значення, використовують *оператор* присвоєння. Загальний вигляд оператора такий: <змінна> = <вираз>

Щоб вводити дані, можна використовувати діалогове вікно, що викликається за допомогою вбудованої функції **InputBox**, яка має вигляд: <змінна> = InputBox(<запрошення>, <заголовок>, <початкове значення>, <x-коорд. вікна>, <y-коорд. вікна>)

**Контрольні запитання та завдання**

- Опишіть принцип виконання команди присвоєння.
- Що таке ініціалізація?
- За допомогою якої команди можна ввести дані через діалогове вікно? Опишіть її синтаксис.

**Питання для роздумів**

- Вкажіть команду присвоєння для збільшення змінної *a* в два рази.
- Чим відрізняється ініціалізація від присвоєння?



Завдання для досліджень

- 1*. Що буде записано в діалоговому вікні для введення даних, якщо у функції `InputBox` не вказати параметр заголовка?
- 2*. Який вигляд буде мати програмний код, у якому пропущено один з параметрів функції `InputBox`?

§33

Співвідношення типів даних та елементів для введення даних

У попередніх параграфах ми розглянули використання різних елементів керування, що дозволяють вводити та виводити різні типи даних при запуску проекту на виконання. Призначення розглянутих елементів керування наведено в табл. 6.4.

Таблиця 6.4. Призначення основних елементів керування

Елемент керування	Призначення
Label (текстовий напис)	Виведення текстового повідомлення
TextBox (текстове поле)	Введення та виведення тексту
Button (кнопка)	Запуск програмного коду
RadioButton (перемикач)	Вибір одного із запропонованих значень
CheckBox (прапорець)	Вибір одного або більше із запропонованих значень
ListBox (список)	Представлення списку елементів
ComboBox (випадаючий список)	Представлення випадаючого списку

Також ми розглянули можливість виведення даних за допомогою вікна текстового повідомлення, що здійснюється за допомогою команди `MsgBox`, та введення даних за допомогою команди `InputBox`.

Після того, як ми дізналися, що бувають величини різних типів, визначимо, які типи даних можуть виводити та вводити різні елементи керування.

33.1.

ВВЕДЕННЯ ТА ВИВЕДЕННЯ ЧИСЛОВИХ І ТЕКСТОВИХ ВЕЛИЧИН

Числові величини в мові програмування мають такі основні типи: `Integer`, `Long`, `Single`; а текстові величини мають тип `String`. Усі елементи керування, крім `ListBox`, мають властивість `Text`, завдяки якій можна виводити числові та текстові величини у вигляді напису на даному елементі керування.



Змінним, які оголошені як числові або текстові величини, можна присвоїти значення властивості **Text**. Якщо величина була оголошена числовою, а властивість **Text** замість чисел містить літери, то при виконанні програмного коду буде видано помилку про невідповідність типів, але не навпаки, оскільки будь-яке число може бути визначено як набір символів, представлених у вигляді цифр.

Так, розглянемо деякі приклади кодів для введення та виведення числових величин до різних елементів керування.

Форма **Form1** містить значення елементів керування **Label** і **TextBox**, як на рис. 6.15.

У вікні коду описано змінні таким чином:

```
Dim a, b As Integer, x, y, z As String
```

Розглянемо приклади кодів присвоєння змінним деяких значень:

`a = Label.Text` — змінна *a* набуде значення «100»;

`x = Label.Text` — змінна *x* набуде значення «100»;

`b = TextBox.Text` — присвоєння помилкове, оскільки значення властивості **Text** є текстом, а змінна *b* має тип **Integer** (ціле число);

`y = TextBox.Text` — змінна *y* отримає значення «100 грн»;

`z = Form1.Text` — змінна *z* отримає значення «Приклад».

Для виведення даних у програмному коді здійснюють присвоєння властивості **Text** різних елементів керування значень змінних. Наприклад,

`Label.Text = a` — на мітку **Label** буде виведено числове значення змінної *a*.

`TextBox.Text = x` — у текстовому полі **TextBox** буде виведено текстове значення змінної *x*.

У більшості випадків значення властивості **Text** для введення та виведення даних використовують переважно для елементів керування **Label**, **TextBox**, **ComboBox**.

Зазначимо також, що є випадок, коли властивості **Text** елемента керування **TextBox** не можна присвоїти значення числової величини. Це тоді, коли його властивість **MultiLine** приймає значення **True**. В цьому випадку **TextBox** може отримувати тільки значення текстових змінних типу **String**.

Що ж робити, коли все ж таки виникає потреба вивести до такого текстового поля числові значення? У такому разі треба числову величину перевести в рядковий тип з допомогою функції `Cstr(<ім'я величини>)`.

Зручними для введення та виведення значень величин є елементи керування **ListBox** і **ComboBox**, які призначені для створення та відображення списків.

Для присвоєння змінній елемента списку використовують команди:

`<змінна> = ListBox.Items.Item(n)` — присвоєння змінній значення елемента списку під номером *n*;

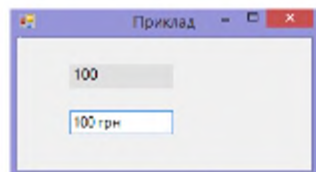


Рис. 6.15



`<змінна> = ListBox.SelectedItem` — присвоєння змінній значення виділеного елемента.

Аналогічно коди записують і для випадваючого списку **ComboBox**.

Щоб до списку занести значення змінної, використовують команди:

`ListBox1.Items.Add(<змінна>)` — додати значення змінної до списку.

`ListBox1.Items.Insert(n, <змінна>)` — додати значення змінної до списку під номером n (у цьому випадку попередні позиції списку не повинні бути порожніми).

33.2.

ВВЕДЕННЯ ТА ВИВЕДЕННЯ ЛОГІЧНИХ ВЕЛИЧИН

Величини логічного типу одержують значення переважно від перемикачів і прапорців.

Інформація, яку отримують від користувача елементи керування **RadioButton** (перемикач) і **CheckBox** (прапорець), — це встановлення одного зі станів: увімкнено (`True`) або вимкнено (`False`). Значення такого типу можуть отримувати змінні логічного типу, що при оголошенні змінних записують як **Boolean**.

Щоб присвоїти змінним значення, що відображають стан перемикачів і прапорців, можна записати команди в такому вигляді:

```
Dim a, b As Boolean
a = RadioButton1.Checked
b = CheckBox1.Checked
```

Як бачимо, саме значення властивості **Checked** визначає стан перемикачів і прапорців.

Значення `True` або `False` змінні логічного типу можуть отримати і з інших елементів керування з властивості **Text**. Проте ця властивість може набувати лише значення `True` або `1` у разі істинності та `False` або `0`, якщо потрібно повідомити про хибне значення. Інакше програма сповістить про помилковий код. Також при виведенні значення логічних змінних до властивості **Text** буде виведено тільки одне із значень `True` або `False`.

Значення логічних змінних можна надавати таким властивостям, як **Visible**, **Enabled** тощо.

Наприклад, `Button.Visible = a` — оскільки змінна a має тип **Boolean**, то набуває значення `True` або `False`.

Далі розглянемо виконання проекту, що містить більшість зазначених вище елементів керування.

ВПРАВА 39



Завдання. Створити проект, у якому реалізувати введення двох значень за допомогою елементів керування **TextBox**. У програмному коді реалізувати присвоєння цих значень змінним різних типів та виведення результату їх додавання до елементів керування **Label**.



1. Створіть проект за зразком (рис. 6.16), надавши елементам керування імена, що вказані у виносках зображення форми.

2. Задайте коди для процедури натиснення на кнопку.

а) Натисніть два рази на кнопку та у вікні програмного коду введіть опис змінних: `Dim a1, a2 As Integer, b1, b2 As Single, c1, c2 As String`

б) Введіть код для присвоєння змінним значень відповідних текстових полів:

`a1 = T1.Text`

`a2 = T2.Text`

`b1 = T1.Text`

`b2 = T2.Text`

`c1 = T1.Text`

`c2 = T2.Text`

в) Введіть код для виведення результатів додавання змінних одного типу до елементів керування **Label**: `L1.Text = a1 + a2`

`L2.Text = b1 + b2`

`L3.Text = c1 + c2`

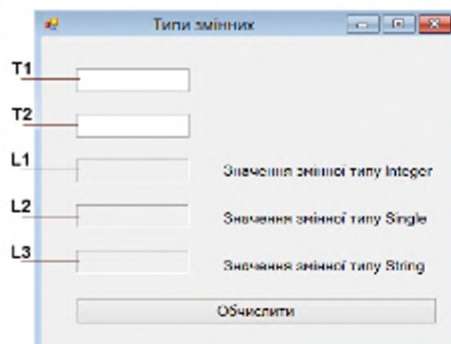


Рис. 6.16

3. Запустіть проект на виконання та введіть до текстових полів два цілих числа. Наприклад, 5 і 6. Натисніть кнопку **Обчислити**. Проаналізуйте виведені результати: результати додавання цілих і дійсних чисел є однаковими (11), при додаванні рядкових величин цифри мають бути написані поруч «56».

4. Змініть значення текстових полів на дробові числа. Наприклад, 5,5 і 4,2. Натисніть кнопку **Обчислити**. Проаналізуйте виведені результати: при додаванні значень змінних цілого типу результат буде округлено до цілого значення (10), при додаванні чисел дійсного типу буде результат 9,7, а при додаванні рядкових величин знову символи будуть дописані один за одним (5,54,2).

5. Змініть значення текстових полів на деякий текст. Натисніть кнопку **Обчислити**. В результаті програма має повідомити про помилку в її коді. Адже відбудеться присвоєння змінним числового типу текстових значень, що неприпустимо.

6. Поекспериментуйте із введенням різних значень до текстових полів.

7. Збережіть проект під назвою **Вправа_39_Прізвище**.



Висновки

Введення та виведення числових і текстових величин у більшості випадків здійснюють за допомогою значень властивості **Text** елементів керування **Label**, **TextBox**, **ComboBox**.

Якщо властивість **MultiLine** елемента керування **TextBox** набуває значення **True**, властивості **Text** можна присвоїти тільки значення змінних рядкового типу.



Щоб перевести числову величину в рядковий тип, використовують функцію **Cstr()**.

Введення та виведення значень змінних можливе до властивості **Item** елементів керування **ListBox** і **ComboBox**, що призначені для створення та відображення списків.

Величини логічного типу одержують значення переважно від перемикачів і прапорців.



Контрольні запитання та завдання

1. Які елементи керування можуть відображати значення числових і текстових величин?
2. За допомогою яких елементів керування можна вводити числові та текстові значення?
3. З яких елементів керування одержують значення логічного типу?



Питання для роздумів

- 1*. Що спільного в елементах керування **Label** і **Button**?
- 2**. Чи може список містити елементи різних типів? Поясніть, чому.



Завдання для досліджень

- 1*. Визначте, яке значення набуває за умовчанням змінна логічного типу.
- 2*. Дослідіть, яким буде результат обчислень, якщо у вправі 39 знак додавання замінити на віднімання. Поясніть одержані результати.

§34

Числові величини

34.1.

АРИФМЕТИЧНІ ОПЕРАЦІЇ

Часто при обробці даних використовують числові величини цілого та дійсного типу. До числових величин відносять змінні типу **Integer**, **Long**, **Single** тощо. Щоб описати дії з числовими даними, використовують арифметичні вирази. Вони дають змогу використовувати різноманітні математичні формули. Користувач самостійно створює арифметичні вирази, комбінуючи константи, змінні, функції та поєднуючи їх із символами арифметичних операцій (табл. 6.5).



Таблиця 6.5. Арифметичні операції

Операція або функція	Пояснення	Приклад	Пріоритет операції
+	додавання	$2 + 6 = 8$	5
-	віднімання	$2 - 6 = -4$	5
*	множення	$2 * 6 = 12$	2
/	ділення	$7 / 2 = 3,5$	2
\	ділення націло	$7 \setminus 2 = 3$	3
Mod	остача від ділення	$7 \text{ Mod } 3 = 1$	4
^	піднесення до степеня	$3 ^ 2 = 9$	1

ВПРАВА 40



Завдання. Створити проект визначення площі прямокутника за його сторонами, в якому при натисненні на кнопку **Пуск** обчислюється значення площі.

Створюючи проекти, які потребують обчислень, бажано дотримуватися певної послідовності дій, що буде представлена при виконанні даної вправи. В залежності від складності розв'язання задачі деякі дії можна пропускати.

1. Визначте вхідні та вихідні дані.

Визначте імена змінних, яким вони будуть присвоєні.

Вхідні: a, b — сторони прямокутника.

Вихідні: S — площа прямокутника.

2. Визначте математичну модель розв'язання задачі. (В даному випадку математична модель — це формула, за якою дана задача має бути розв'язана). Згідно визначеної формули (формул) встановіть, які можливо ще проміжні змінні будуть використані при розв'язанні задачі.

$$S = a \cdot b.$$

Як видно з обраної формули, для розв'язання даної задачі проміжні змінні не потрібні.

3. Створіть форму для реалізації проекту за зразком (рис. 6.17).
4. Надайте імена текстовим полям T1 і T2, а мітці, що має містити результат, — L. Встановіть для мітки L властивості **AutoSize** = False, **BorderStyle** = Fixed3D.
5. Запишіть програмний код кнопки **Обчислити**.

'оголошення змінних

`Dim a, b, S As Integer`

'присвоєння змінній a вмісту першого текстового поля

`a = T1.Text`

'присвоєння змінній b вмісту другого текстового поля

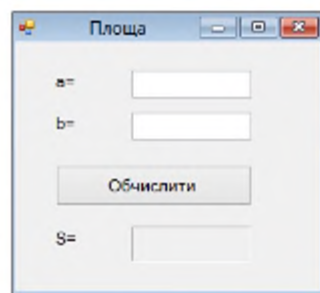


Рис. 6.17



```
b = T2.Text  
'формула обчислення площі  
S = a * b  
'присвоєння мітці значення площі S  
L1.Text = S
```

Зауваження. За умови введення даних за допомогою функції **InputBox** і виведення результату з використанням процедури **MsgBox** для реалізації даного проекту використанням елементів керування на формі можна знехтувати. А програмний код записати для події клацання на форму в такому вигляді:

```
'оголошення змінних  
Dim a, b, S As Integer  
'присвоєння значення змінній a  
a = InputBox("a=")  
'присвоєння значення змінній b  
b = InputBox("b=")  
'формула обчислення площі  
S = a * b  
'виведення значення площі S  
MsgBox("S=" & S)
```

6. Запустіть проект та переконайтеся у правильності його виконання.

Після запуску проекту необхідно ввести два числа та натиснути кнопку **Обчислити**. Переконайтеся, що результат виконання програми є добутком цих чисел. Наприклад, для чисел 5 і 6 результат має бути 30.

Процес перевірки правильності виконання програми за різних умов і значень називають *тестуванням програми*.

Оголошення змінних	— Dim a, b, S As Integer
Введення даних	{ a = InputBox ("a=") b = InputBox ("b=")
Обробка даних	— S = a * b
Виведення результату	— MsgBox("S=" & S)

Рис. 6.18. Складові програмного коду для розв'язання задачі

7. Збережіть проект під назвою **Вправа_40_Прізвище**.

Зверніть увагу, що в більшості випадків, код програми з використанням змінних складається з чотирьох частин (рис. 6.18): оголошення змінних, введення даних, обробка даних, виведення результату. В залежності від умови задачі деякі частини можуть бути опущені.

34.2.

ЗАДАЧІ НА ЦІЛОЧИСЕЛЬНЕ ДІЛЕННЯ

Розв'язуючи різноманітні задачі з програмування, часто виникає потреба у визначенні цифр цілого числа. Використовуючи арифметичні операції ділення націло та визначення остачі від ділення (**Mod**), можна вивести формули, що дозволять визначити цифру числа будь-якої розрядності. В таблиці 6.6 наведено, як знаходити цифри чисел до четвертого розряду. Пропоновані формули стануть у нагоді при подальшому розв'язуванні задач.

Таблиця 6.6. Знаходження цифр цілого числа a

Необхідно знайти	Вираз	Приклади
Остання цифра числа	$a \text{ Mod } 10$	$18 \text{ Mod } 10 = 8$ $325 \text{ Mod } 10 = 5$ $1789 \text{ Mod } 10 = 9$
Знаходження першої цифри двоцифрового числа	$a \setminus 10$	$18 \setminus 10 = 1$ $32 \setminus 10 = 3$
Знаходження першої цифри трицифрового числа	$a \setminus 100$	$185 \setminus 100 = 1$ $327 \setminus 100 = 3$
Знаходження першої цифри довільного числа	$a \setminus 10^{(n-1)}$, де n — розрядність числа	$123456 \setminus 10^5 = 1$ $78934555 \setminus 10^7 = 7$
Знаходження другої цифри трицифрового числа	$a \setminus 10 \text{ Mod } 10$	$736 \setminus 10 = 73 \rightarrow 73 \text{ Mod } 10 = 3$
Знаходження другої цифри чотирицифрового числа	$a \setminus 100 \text{ Mod } 10$	$7362 \setminus 100 = 73 \rightarrow 73 \text{ Mod } 10 = 3$
Знаходження третьої цифри чотирицифрового числа	$a \setminus 10 \text{ Mod } 10$	$7362 \setminus 10 = 736 \rightarrow 736 \text{ Mod } 10 = 6$

ВПРАВА 41



Завдання. Створити проект обчислення суми квадратів цифр цілого трицифрового числа.

- Визначте вхідні та вихідні дані.
Вхідні: a — трицифрове число.
Вихідні: y — сума квадратів цифр числа.
- Визначте математичну модель розв'язання задачі.
 - Знайти цифри числа:
 $a1 = a \setminus 100$ — перша цифра;
 $a2 = a \setminus 10 \text{ Mod } 10$ — друга цифра;
 $a3 = a \text{ Mod } 10$ — третя цифра.
 - Знайти суму квадратів цифр числа:
 $y = a1^2 + a2^2 + a3^2$.
- Створіть форму для реалізації проекту за зразком (рис. 6.19).
- Надайте імена текстовому полю T , а мітці — L .
- Запишіть програмний код для кнопки:


```
'оголошення змінних
Dim a, a1, a2, a3, y As Integer
'введення числа
a = T.Text
'визначення першої цифри числа
a1 = a \ 100
```

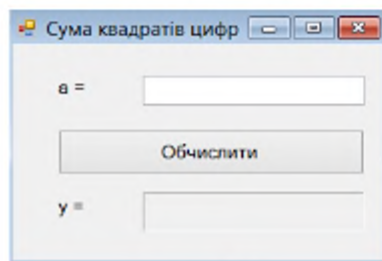


Рис. 6.19



```
'визначення другої цифри числа  
a2 = a \ 10 Mod 10  
'визначення третьої цифри числа  
a1 = a Mod 10  
'знаходження суми квадратів цифр  
y = a1^2 + a2^2 + a3^2  
L.Text = y
```

- Запустіть проект та переконайтеся в правильності його виконання. (При $a = 234$ вийде $y = 29$).
- Збережіть проект під назвою **Вправа_41_Прізвище**.

ВПРАВА 42**САМОСТІЙНО**

Завдання. Створити проекти для розв'язування задач.

- Обчислити добуток другої та третьої цифр цілого чотирицифрового числа.
- Обчислити суму цифр чотирицифрового числа.
- Знайти суму першої та п'ятої цифр п'ятицифрового числа.
- Знайти добуток цифр, що стоять на парних місцях шестицифрового числа.
- Нехай в будинку k поверхів та m квартир на поверсі. Дано R — номер квартири. Визначте, на якому поверсі та в якому під'їзді.

**Висновки**

Щоб записати арифметичний вираз, використовують такі арифметичні операції: «+» (додавання), «-» (віднімання), «*» (множення), «/» — ділення, «\» — ділення націло, «Mod» — остача від ділення, «^» — піднесення до степеня.

Щоб знайти останню цифру будь-якого числа, використовують вираз $a \text{ Mod } 10$.

Щоб знайти першу цифру довільного числа, використовують вираз $a \setminus 10^{(n-1)}$, де n — розрядність числа.

**Контрольні запитання та завдання**

- Назвіть типи числових величин.
- Вкажіть пріоритет арифметичних операцій.
- Що таке тестування програми?
- Як знайти першу цифру двоцифрового числа? трицифрового числа? довільного числа?
- Як знайти останню цифру довільного числа?



Питання для роздумів

- 1*. Який тип треба встановити для змінної, яка є результатом добутку трьох чотирицифрових цілих чисел?
- 2*. Для розв'язування яких задач можна використати арифметичні операції цілочисельного ділення?



Завдання для досліджень

- 1*. Вкажіть вирази, за допомогою яких можна визначити цифри чотирицифрового числа.
- 2*. Визначте загальну формулу для знаходження другої цифри довільного числа.

§35

Стандартні математичні функції

Для виконання більш складних обчислень у математиці використовують різноманітні функції. В мові програмування Visual Basic.NET також є власні *математичні функції*. Аргумент функції завжди беруть у дужки. Аргументом може бути константа, змінна, арифметичний вираз або інша функція.

Математичні функції в кодах програми записують разом з префіксом **Math**.

Наприклад, щоб присвоїти змінній a значення модуля числа b , треба записати код:

$$a = \text{Math.Abs}(b),$$

де **Abs** — функція обчислення модуля числа.

У таблиці 6.7 наведено деякі основні математичні функції класу **Math**. Можливо деякі з них ви ще навіть не вивчали на уроках математики, проте згодом вони вам стануть у нагоді при розв'язуванні різноманітних задач. Щоб дізнатися про призначення деяких функцій, які не представлені в таблиці, потрібно після введення в програмному коді префікса **Math** навести курсор на будь-яку функцію. Відразу з'явиться спливаюча довідка з поясненням призначення та синтаксису даної функції.



Таблиця 6.7. Основні математичні функції

Функція	Математичний запис	Пояснення	Приклад
Abs(x)	$ x $	модуль числа	<code>Math.Abs(-4.5) = 4,5</code>
Ceiling(x)		найменше ціле, яке більше від заданого дробового числа	<code>Math.Ceiling(1.1) = 2</code>
Cos(x)	$\cos x$	косинус числа (кут вимірюється в радіанах)	<code>Math.Cos(3.14) = 0,999</code>
Floor(x)		найбільше ціле, яке менше від заданого дробового числа	<code>Math.Floor(1.1) = 1</code>
PI	π	число π	<code>Math.PI = 3,1415926</code>
Round(x, n)		округлення числа, де x — число, n — кількість цифр у дробовій частині	<code>Math.Round(10.56, 1) = 10,6</code>
Sin(x)	$\sin x$	синус числа	<code>Math.Sin(1.3) \approx 0,96</code>
Sqrt(x)	\sqrt{x}	корінь квадратний з числа	<code>Math.Sqrt(100) = 10</code>
Tan(x)	$tg x$	тангенс числа	<code>Math.Tan(1.5) \approx 14,1</code>

Виконуючи наступні вправи, ви будете мати справу не лише з цілими, а й з дійсними числами. Зверніть особливу увагу при визначенні типів змінних, які є результатами обчислень. Так, якщо в формулі було використано дію ділення, функцію кореня тощо, результат буде дійсним числом, отже змінні будуть приймати тип **Single**.

! Також зверніть увагу на написання дійсних чисел: у програмному кодї для розділення цілої та дробової частини використовують крапку, а в елементах керування — кому.

ВПРАВА 43



Завдання. Дано цілі значення a , b , c . Створити проект для обчислення значень x та y за вказаними нижче формулами.

Рис. 6.20

$$x = \frac{\sqrt{a+1} - |c|}{b^2 + 1}$$

$$y = \sqrt{abc + \frac{a}{b}} + |a|$$

1. Створіть форму за зразком (рис. 6.20).
2. Надайте імена текстовим полям T1, T2 і T3 відповідно, а міткам, що мають містити результат, — L1 і L2.
3. Запишіть програмний код для кнопки **Обчислити**.



‘оголошення змінних

```
Dim a, b, c As Integer, x, y As Single
```

‘введення даних

```
a = T1.Text
```

```
b = T2.Text
```

```
c = T3.Text
```

‘введення формули обчислення значення x

```
x = (Math.Sqrt(a + 1) - Math.Abs(c)) / (b ^ 2 + 1)
```

‘введення формули обчислення значення y

```
y = Math.Sqrt(a * b * c + a / b) + Math.Abs(a)
```

‘виведення значень x та y на мітки

```
L1.Text = x
```

```
L2.Text = y
```

Оскільки результатом виконання програми є дійсне число, і велика кількість знаків після коми не завжди є суттєвою, можна при виведенні результату зменшити кількість знаків після коми, застосувавши функцію форматування.

Так, замість коду `L1.Text = x` можна записати

`L1.Text = Format(x, «0.000»)`. У цьому записі визначено, що після коми в числі буде три знаки.

4. Змініть програмний код, застосувавши до змінних, що містять результат, функцію форматування.
5. Запустіть проект та переконайтеся в правильності його виконання. (При $a = 1, b = 1, c = 1$ ви маєте отримати: $x = 0,207; y = 2,414$).
6. Збережіть проект під назвою **Вправа_43_Прізвище**.



Висновки

Для виконання деяких обчислень використовують математичні функції. *Аргументом* функції може бути константа, змінна, арифметичний вираз або інша функція.

Математичні функції в кодах програми записують разом із префіксом **Math**.

Основні математичні функції: **Abs(x)** — модуль числа; **Round(x,n)** — округлення числа, **Sqrt(x)** — корінь квадратний з числа.



Контрольні запитання та завдання

1. Як правильно записати математичні функції в програмуванні?
2. За допомогою якої функції можна знайти корінь з числа або виразу?
3. Як виконати округлення числа до двох знаків після коми?



Питання для роздумів

- 1*. Як у середовищі програмування дізнатися про призначення математичної функції?
- 2*. Який тип буде мати результат виконання функції `Math.Abs(x)`? `Math.Sqrt(x)`?



Завдання для досліджень

- 1*. Дізнайтеся більш докладно про можливості використання функції `Format()`.

§36

Величини текстового типу

Комп'ютер може виконувати різноманітні дії не лише з числовими величинами, а й з текстовими. Нагадаємо, що текстові величини в середовищі програмування мають тип **String**. Змінна типу **String** — це рядок символів досить великої довжини, приблизно до двох мільярдів символів. Значення рядкових змінних записують у лапках, наприклад, «Приклад тексту».

Над текстовими даними визначена лише операція додавання, для цього використовують символи «+» або «&». Так рівносильними будуть програмні коди:

`a = «КОМ» + «ПОТ»` та `a = «КОМ» & «ПОТ»`

Для виконання інших дій з текстовими величинами використовують спеціально призначені для цього функції. Перелік основних функцій наведено в таблиці 6.8.

Таблиця 6.8. Основні функції для роботи з рядковими величинами

Функція	Призначення	Приклад	
		Вираз	Результат
<code>Asc</code> (рядок)	Визначає код символу	<code>x = Asc(«д»)</code>	228
<code>Chr</code> (код символу)	Визначає символ за його кодом	<code>a = Chr(228)</code>	«д»
<code>Space</code> (довжина)	Виводить рядок пробілів заданої довжини	<code>a = Space(5)</code>	« »
<code>Len</code> (рядок)	Визначає довжину рядка без урахування лапок	<code>x = Len(«Привіт»)</code>	6



LSet(рядок, довжина)	Визначає задану кількість символів з початку рядка	a = LSet(«КОМПОТ», 3)	«КОМ»
Mid(рядок, позиція, довжина)	Визначає задану кількість символів вибраної з рядка тексту, починаючи із заданої позиції	a = Mid(«трапеція», 1, 4)	«трап»
Ltrim(рядок)	Вилучає пробіли на початку рядка	a = Ltrim(« привіт »)	«привіт »
Rtrim(рядок)	Вилучає пробіли наприкінці рядка	a = Rtrim(« привіт »)	«привіт»
Trim(рядок)	Вилучає пробіли на початку та наприкінці рядка	a = Trim(« привіт »)	«привіт»
InStr(рядок, підрядок)	Номер позиції першого входження підрядка в рядок	x = InStr(«Інформатика», «форма»)	3
Replace(рядок, підрядок1, підрядок2, початок)	Знаходить у рядку підрядок1 та змінює на підрядок2, починаючи із заданої позиції	a = Replace(«КОТИК», «КОТ», «РОТ», 1)	«РОТИК»
LCase(рядок)	Перетворює всі великі літери на малі	a = LCase(«ПрограМа»)	«програма»
UCase(рядок)	Перетворює всі малі літери на великі	a = UCase(«ПрограМа»)	«ПРОГРАМА»
StrReverse(рядок)	Розташовує символи в зворотному порядку	a = StrReverse(«форма»)	«амроф»

ВПРАВА 44



Завдання. Створити проект, у якому замінити в тексті всі літери «а» на «о».

- Визначте вхідні та вихідні дані. Визначте імена змінних, яким вони будуть присвоєні. Вхідні: *x* — даний текст.
Вихідні: *y* — змінений текст.
- Задайте програмний код для події клацання на формі.

```
Dim x, y As String
x = InputBox(«Введіть текст»)
'Заміна в заданому рядку одного символу на інший, починаючи з 1-го
y = Replace(x, «а», «о», 1)
MsgBox(y)
```
- Збережіть проект під назвою **Вправа_44_Прізвище**.



ВПРАВА 45



Завдання. Створити проект, у якому із слова «програмування» утворюються нові слова: «амур», «пара», «варан», «ураган». Дані слова вивести в текстовому повідомленні в стовпчик.

1. Визначте вхідні, вихідні дані та проміжні змінні.
Визначте імена змінних, яким вони будуть присвоєні.
Вхідні: a — даний текст.
Вихідні: f — нові слова, розташовані в стовпчик.
Проміжні: b, c, d, g — нові слова.
2. Задайте програмний код для події клацання на формі.

```
Dim a, b, c, d, g, f As String
a = «програмування»
'амур
b = Mid(a, 6, 2) + Mid(a, 8, 1) + Mid(a, 5, 1)
'пара
c = Mid(a, 1, 1) + Mid(a, 6, 1) + Mid(a, 5, 1) + Mid(a, 10, 1)
'варан
d = Mid(a, 9, 2) + Mid(a, 5, 2) + Mid(a, 10, 2)
'ураган
g = Mid(a, 8, 1) + Mid(a, 5, 2) + Mid(a, 4, 1) + Mid(a, 10, 2)
'поєднання слів в одну змінну з переходом на новий рядок після_
кожного слова
f = b + vbNewLine + c + vbNewLine + d + vbNewLine + g
MsgBox(f)
```

3. Самостійно додайте принаймні ще три слова.
4. Збережіть проект під назвою **Вправа_45_Прізвище**.

ВПРАВА 46



Завдання. Створіть проект, у якому користувач має ввести своє ім'я маленькими літерами, а в текстовому повідомленні має бути виведено це ж ім'я з великої літери.

1. Визначте вхідні, вихідні дані та проміжні змінні.
Визначте імена змінних, яким вони будуть присвоєні.
Вхідні: a — ім'я, введене маленькими літерами.
Вихідні: x — ім'я, у якому перша літера велика, інші — маленькі.
Проміжні: b, c, d, g — нові слова.
2. Задайте програмний код для події клацання на формі.

```
Dim a, b As String
a = InputBox(«Введіть ім'я маленькими літерами»)
'Одержання першої літери
b = LSet(a, 1)
'Перетворення маленької літери на велику
b = UCase(b)
```



‘Відкидання у даному імені першої літери

```
a = Mid(a, 2, Len(a))
```

‘З’єднання першої великої літери з іншою частиною імені

```
b = b + a
```

```
MsgBox(b)
```

- Організуйте введення також прізвища та по батькові маленькими літерами. Здійсніть виведення прізвища, імені та по батькові в стовпчик у текстовому повідомленні, починаючи з великої літери.
- Збережіть проект під назвою **Вправа_46_Прізвище**.



Висновки

Текстові величини мають тип **String**.

Значення рядкових змінних записують у лапках.

Над текстовими даними визначена операція додавання, для якої використовують символи «+» або «&».

Щоб виконувати дії з текстовими величинами, використовують спеціальні функції.



Контрольні запитання та завдання

- Який тип мають текстові величини?
- Як виконати додавання текстових значень?
- За допомогою якої функції можна знайти довжину рядка?
- Як вилучити зайві пробіли в тексті?
- Які функції дозволяють із заданого тексту виділити його частину?



Питання для роздумів

- Як у заданому тексті вилучити деякий текст?
- Як встановити пробіл між двома текстовими величинами a і b , значення яких необхідно розташувати послідовно?



Завдання для досліджень

- Як можна порівняти текстові величини?



ПРАКТИЧНА РОБОТА №10

Складання та виконання лінійних алгоритмів операцювання величин в навчальному середовищі програмування

Завдання 1. Створити проект для розв'язання задачі.

Автобус, що рухався зі швидкістю a км/год, доїхав до міста за b хв. З якою швидкістю повинен рухатись велосипедист, щоб проїхати ту саму відстань до міста за c год та d хв? Відповідь подати в км/год.

1. Визначте вхідні та вихідні дані.

Вхідні: a — швидкість автобуса; b — час, який рухався автобус;

c , d — кількість годин і хвилин, що має рухатися велосипедист.

Вихідні: Vv — швидкість велосипедиста.

2. Визначте математичну модель розв'язання задачі: $c = \sqrt{a^2 + b^2}$.
Для зручності запишемо дані у вигляді таблиці.

	Швидкість	Час	Відстань
Автобус	a (км/год)	b (хв)	
Велосипедист	?	c (год) d (хв)	

$Tav = b / 60$ (год) — час, який рухався автобус, у годинах.

$Sav = a \cdot Tav$ (км) — відстань, яку проїхав автобус.

$Tv = c + d / 60$ (год) — час, який рухався велосипедист, у годинах.

$Vv = Sav \cdot Tv$ (км/год) — швидкість велосипедиста.

Створивши математичну модель, ми визначили, що для розв'язання задачі потрібно застосувати проміжні змінні: Tav , Sav , Tv .

3. Створіть форму для реалізації проекту та введіть необхідний програмний код.
4. Збережіть проект під назвою **ПР10_Завдання1_Прізвище**.

Завдання 2. Створити проект для знаходження гіпотенузи прямокутного трикутника за її катетами. Реалізувати введення значень катетів і послідовне виведення у вигляді списку обчислень гіпотенузи при різних значеннях катетів.

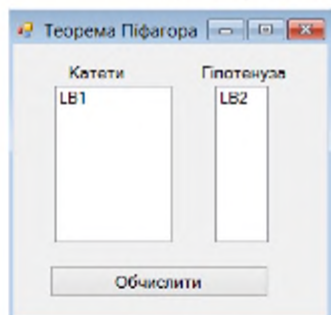


Рис. 6.21

1. Визначте вхідні, вихідні дані.

Визначте імена змінних, яким вони будуть присвоєні.

Вхідні: a , b — катети.

Вихідні: c — гіпотенуза.

2. Визначте математичну модель розв'язання задачі.
3. Створіть форму за зразком (рис. 6.21), надайте спискам імена **LB1** і **LB2**.
4. Введіть програмний код для процедури натиснення на кнопку.



```
Dim a, b, c As Single, x As String
a = InputBox(«a=», «Перший катет»)
b = InputBox(«b=», «Другий катет»)
c = Math.Sqrt(a ^ 2 + b ^ 2)
'Рядкова змінна одержує значення двох катетів відокремлених відступом
x = CStr(a) + vbTab + CStr(b)
'Виведення значень катетів до першого списку
LB1.Items.Add(x)
'Виведення значення гіпотенузи з форматуванням до другого списку
LB2.Items.Add(Format(c, «0.0»))
```

- Запустіть проект та перевірте правильність його виконання, увівши декілька разів значення катетів.
- Самостійно додайте ще один список, у якому реалізуйте виведення результату знаходження синусу кута, що лежить навпроти катета a .
- Збережіть проект під назвою **ПР10_Завдання2_Прізвище**.



Виконати два додаткових завдання до практичної роботи ви зможете на диску.

§37

Величини логічного типу. Програми з розгалуженням

У сьомому класі, вивчаючи мову програмування Scratch, ви ознайомилися з командами розгалуження. Ці команди використовують, коли виконання тієї чи іншої дії залежить від виконання певної умови.

Вибір однієї з двох послідовностей дій залежно від виконання або невиконання деякої умови називають **розгалуженням**.

Алгоритм, що містить одну або кілька умов і, відповідно, дві та більше ще гілок, називають *алгоритмом з розгалуженням*.

37.1. ЛОГІЧНІ ВИРАЗИ

Щоб сформувавши умови в команді розгалуження, використовують логічні вирази. Логічний вираз може набувати значення **True** (істинність) або **False** (хибність).

Простий логічний вираз — це два арифметичні вирази, з'єднані символом відношення (табл. 6.9).

Таблиця 6.9. Символи відношень

Символ	Дія порівняння
=	дорівнює
>	більше
<	менше
>=	більше або дорівнює
<=	менше або дорівнює
<>	не дорівнює



Приклади простих логічних виразів та їх значення наведено в таблиці 6.10.

Таблиця 6.10. Приклади простих логічних виразів

$a = 2, b = 3$	
Логічний вираз	Значення
$a = b$	False
$a + b = 5$	True
$a \geq b$	False
$a <> b$	True
$a + b <> b - a$	True

Вирази, з'єднані між собою логічними операціями **And** (і), **Or** (або), **Not** (не), є складеними логічними виразами (табл. 6.11).

Для визначення істинності або хибності складених логічних виразів використовують так звані таблиці істинності.

Таблиця 6.11. Логічні операції

Оператор	Назва	Пріоритет
Not (не)	Логічне заперечення	1
And (і)	Логічне множення	2
Or (або)	Логічне додавання	3

Таблиця 6.12. Таблиця істинності операції **Not** (логічне заперечення)

X	Not X
True	False
False	True

Логічне заперечення. Операція **Not** (логічне заперечення), що застосовується до виразу X , дає хибний результат, якщо вираз X є істинним, і навпаки (табл. 6.12).

Логічне множення (кон'юнкція). Операція **And** (логічне множення), що застосовується до виразів X та Y , дає істинний результат, якщо обидва вирази є істинними (табл. 6.13).

Таблиця 6.13. Таблиця істинності операції **And** (логічне множення)

X	Y	X And Y
True	True	True
True	False	False
False	True	False
False	False	False

Таблиця 6.14. Таблиця істинності операції **Or** (логічне додавання)

X	Y	X Or Y
True	True	True
True	False	True
False	True	True
False	False	False

Таблиця 6.15. Приклади складених логічних виразів

$a = 2, b = 3$	
Логічний вираз	Значення
Not $a = b$	True
$a + b = 5$ Or $a > 5$	True
$a \leq b$ And $b = 3$	True
$a \geq b$ And $b = 3$	False
$a > b$ Or $a = 7$	False

Логічне додавання (диз'юнкція). Операція **Or** (логічне додавання), що застосовується до виразів X та Y , дає істинний результат, якщо принаймні один вираз є істинним (табл. 6.14).

Приклади складених логічних виразів та їх значення наведено в таблиці 6.15.

Одним з поширених прикладів використання складених умов є подвійні нерівності. Так, нерівність виду $1 < x \leq 5$ у програмному коді слід записувати у вигляді складеного логічного виразу: $1 < x$ And $x \leq 5$.



Значення логічних виразів можна присвоювати логічним величинам.

Наприклад, $x = y < 5$. Значення, яке отримає змінна x буде True або False. Такі величини мають тип **Boolean**.

37.2.

КОМАНДИ РОЗГАЛУЖЕННЯ

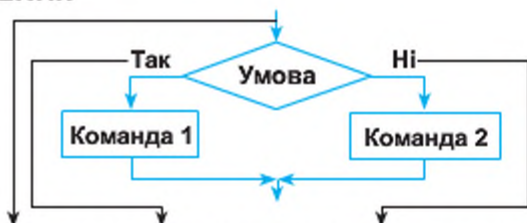
У VB.NET є дві форми умовного оператора — однорядкова та багаторядкова. *Однорядкова форма* використовується в простих випадках, коли в кожній гілці є не більше одного оператора. Вона має вигляд:

```
If <умова> Then <команда 1>_
Else <команда 2>
```

Якщо If значення умови є істинним, то Then виконується команда 1, інакше Else виконується команда 2.

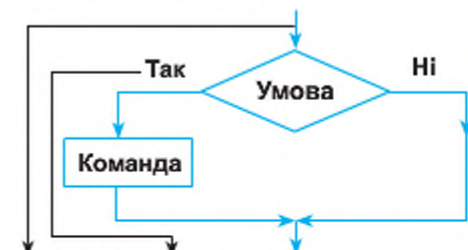
Їй відповідає відома вам із цього класу блок-схема з повним розгалуженням (рис. 6.22).

Із схеми видно, що слову «так» відповідає команда Then, а слову «ні» — команда Else. Даний вид команди розгалуження може бути представлений і в неповній формі, коли гілка Else є відсутньою (рис. 6.23).



```
If <умова> Then <команда 1> Else <команда 2>
```

Рис. 6.22. Блок-схема повного розгалуження і відповідний код мовою програмування VB.NET



```
If <умова> Then <команда>
```

Рис. 6.23. Блок-схема неповного розгалуження і відповідний код мовою програмування VB.NET

ВПРАВА 47



Завдання. Створити проект знаходження максимального числа з двох заданих чисел a і b .

1. Визначте вхідні та вихідні дані.
Вхідні: a , b — дані числа.
Вихідні: max — більше з двох чисел.
2. Побудуйте блок-схему реалізації алгоритму розв'язання задачі (рис. 6.24).
3. Запишіть програмний код об'єкта форма для події Click (подія має відбуватися при клацанні на формі).

```
Dim a, b, max As Integer
```

'Введення двох чисел за допомогою діалогових вікон

```
a = InputBox(«a=», «Введення першого числа»)
```

```
b = InputBox(«b=», «Введення другого числа»)
```

'Перевірка умови та визначення більшого числа

```
If a > b Then max = a Else max = b
```

```
MsgBox(«max = » & max)
```

4. Запустіть проект та перевірте правильність його виконання.
5. Збережіть проект під назвою **Вправа_47_Прізвище**.

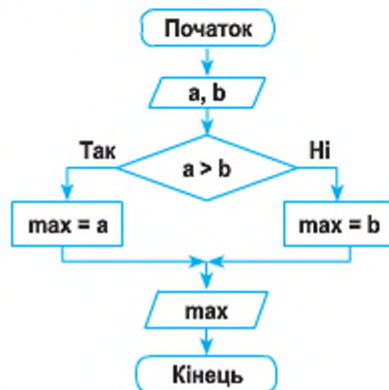


Рис. 6.24



ВПРАВА 48



Рис. 6.25



Завдання. Створити проект, що дозволяє визначити, чи є задане число a парним. Якщо так, то вивести повідомлення про це.

- Визначте вхідні та вихідні дані.
Вхідні: a — дане число.
Вихідні: повідомлення про парність числа.
- Побудуйте блок-схему реалізації алгоритму розв'язання задачі (рис. 6.25).
- Запишіть програмний код для події клацання на формі.

```
Dim a As Integer
```

```
a = InputBox(«a=», «Введення числа»)
```

```
'Перевірка умови парності
```

```
If a Mod 2 = 0 Then MsgBox(«число парне»)
```

- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_48_Прізвище**.



Висновки

Вибір однієї з двох послідовностей дій залежно від виконання або невиконання деякої умови називають *розгалуженням*.

Алгоритм, що містить одну або кілька умов і, відповідно, дві та більше гілок, називають *алгоритмом з розгалуженням*.

Щоб сформулювати умову в команді розгалуження, використовують логічні вирази. Логічний вираз може набувати значення **True** (істинність) або **False** (хибність).

Простий логічний вираз — це два арифметичні вирази, з'єднані символом відношення (=, >, <, >=, <=, <>).

Вирази, з'єднані між собою логічними операціями **And** (і), **Or** (або), **Not** (не), є *складеними логічними виразами*.

Операція **Not** (логічне заперечення), що застосовується до виразу X , дає хибний результат, якщо вираз X є істинним, і навпаки.

Операція **And** (логічне множення), що застосовується до виразів X та Y , дає істинний результат, якщо обидва вирази є істинними.

Операція **Or** (логічне додавання), що застосовується до виразів X та Y , дає істинний результат, якщо принаймні один вираз є істинним.

Логічні величини мають тип **Boolean**.

Однорядкова форма розгалуження має вигляд:

```
If <умова> Then <команда 1> Else <команда 2>
```

**?** Контрольні запитання та завдання

1. Що таке алгоритм з розгалуженням?
2. Що таке простий логічний вираз? Наведіть приклади.
3. Що таке складений логічний вираз? Наведіть приклади.
4. Поясніть принципи виконання операцій логічного заперечення, множення, додавання.
5. Який пріоритет логічних операцій?
6. Який тип мають логічні величини?
7. Опишіть синтаксис однорядкової форми розгалуження.
8. Які бувають види команд розгалуження?

**Питання для роздумів**

1. Яке значення отримає логічний вираз $a > 8$, якщо $a = 5$? $a = 10$?
- 2*. Яке значення отримає логічний вираз $a > 4$ And $a + b <= 4$, якщо $a = 5$, $b = -2$? $a = 10$, $b = 2$?
- 3*. Яке значення отримає логічний вираз $\text{Not}(\text{Not}(a = 5))$, якщо $a = 5$? $a = 0$?
- 4*. Яке значення логічного виразу a Or $\text{Not}(a)$?

**Завдання для досліджень**

- 1*. Як потрібно розставити дужки в логічному виразі a Or b And a Or c And b , щоб його значення було істинним, якщо $a = \text{False}$, $b = \text{True}$, $c = \text{True}$?
- 2*. Визначте значення змінної x після виконання програмного коду:
`a = 5`
`b = 6`
`If a + b mod 10 = 1 Then x = b - a Else x = b + a`
`MsgBox(x)`

§38**Алгоритми з розгалуженнями
для опрацювання величин**

У попередньому параграфі ми розглянули однорядкову форму розгалуження, яка використовується в тому випадку, коли в кожній гілці є не більше одного оператора. Якщо ж у гілці потрібно використати більше одного оператора, використовують *багаторядкову форму розгалуження*, яка має такий вигляд:

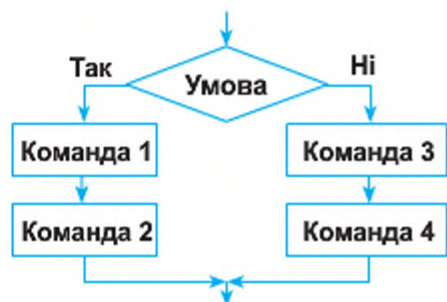


Рис. 6.26

```

If <умова> Then
  <команда 1>
  <команда 2>
...
Else
  <команда 3>
  <команда 4>
...
End If

```

Їй відповідає блок-схема зображена на рис. 6.26.

Звичайно, багаторядковий вигляд може мати і *неповна форма* вказівки розгалуження. В цьому випадку гілка **Else** буде відсутньою і програмний код матиме вигляд:

```

If <умова> Then
  <команда 1>
  <команда 2>
End If

```

Виконаємо наступні вправи, щоб розглянути використання представлених видів програмних кодів.

ВПРАВА 49



Завдання. Розробити проект для знаходження значення виразу: $y = \frac{\sqrt{x}}{x+2}$.

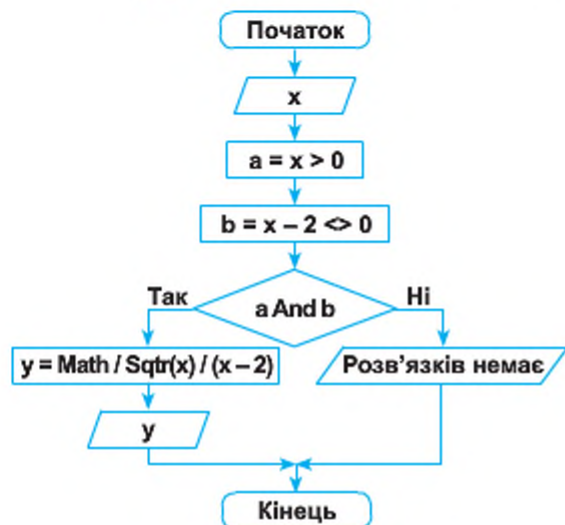


Рис. 6.27

1. Визначте вхідні та вихідні дані.

Вхідні: x — дане число.

Вихідні: y — результат обчислення або повідомлення про відсутність розв'язків.

2. Побудуйте блок-схему реалізації алгоритму розв'язання задачі (рис. 6.27).

Для розв'язання задачі введено дві логічні змінні a та b , які будуть набувати результатів виконання умов, що враховують область допустимих значень представленого виразу.

Так, підкореневий вираз чисельника має бути додатним, отже $a = x > 0$, а знаменник не може дорівнювати нулю — $b = x - 2 <> 0$.

3. Запишіть програмний код для події клацання на формі.

```

Dim x, y As Single, a, b As Boolean
x = InputBox(«x=»)

```

†присвоєння логічній змінній a умови того, що підкореневий вираз є додатним
 $a = x > 0$



'присвоєння логічній змінній b умови того, що знаменник не дорівнює нулю

```
b = x - 2 <> 0
If a And b Then
    y = Math.Sqrt(x)/(x+2)
    MsgBox("y=" & y)
Else
    MsgBox("розв'язків немає")
End If
```

Зауваження. Програмний код для розв'язання задачі можна створити без використання логічних величин. У цьому випадку логічний вираз буде більш громіздким, проте програмний код — дещо коротшим.

```
Dim x, y As Single
x = InputBox("x=")
'врахування в умові області допустимих значень виразу
If x > 0 And x - 2 <> 0 Then
    y = Math.Sqrt(x)/(x+2)
    MsgBox("y=" & y)
Else
    MsgBox("розв'язків немає")
End If
```

- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_49_Прізвище**.

ВПРАВА 50



Завдання. Створити проект для знаходження значення виразу.

$$y = \begin{cases} \frac{4x^2 + 1}{x - 5}, & \text{якщо } x < 5 \\ 3x^2 - 2, & \text{в інших випадках} \end{cases}$$

- Визначте вхідні та вихідні дані.
Вхідні: x — дане число.
Вихідні: y — результат обчислення.
- Побудуйте блок-схему реалізації алгоритму розв'язання задачі (рис. 6.28).

- Запишіть програмний код для події клацання на формі.

```
Dim x, y As Single
x = InputBox("x=")
If x < 5 Then
    y = (4 * x ^ 2 + 1) / (x - 5)
Else
    y = 3 * x ^ 2 - 2
End If
MsgBox("y=" & y)
```

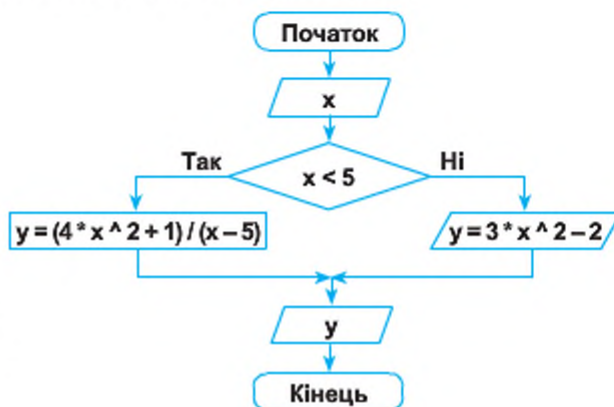


Рис. 6.28



З наведеного програмного коду видно, що коли виконується умова $x < 5$, значення y буде знайдено за першою формулою, якщо умова не виконується, то для визначення y використовується друга формула.

- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_50_Прізвище**.

ВПРАВА 51

Завдання. Створити проекти для знаходження значення виразів:

**Виконати**

додаткові вправи ви зможете на диску.

$$\text{а) } y = \begin{cases} \frac{9x^2+5}{3x+12}, & \text{якщо } x < -4 \\ 4x^2 - 7, & \text{в інших випадках} \end{cases}$$

$$\text{б) } y = \begin{cases} x^2 + 4x + 5, & \text{якщо } x \leq -2 \\ \frac{1}{x^2+4x+5}, & \text{в інших випадках} \end{cases}$$

**Висновки**

Якщо в гілці вказівки розгалуження потрібно використати більше одного оператора, використовують *багаторядкову форму розгалуження*, що має такий вигляд:

```
If <умова> Then
    <команда 1>
    <команда 2>
Else
    <команда 3>
    <команда 4>
End If
```

Багаторядковий вигляд може мати і *неповна* форма вказівки розгалуження. В цьому випадку гілка **Else** буде відсутньою і програмний код буде такий:

```
If <умова> Then
    <команда 1>
    <команда 2>
End If
```

**Контрольні запитання та завдання**

- У яких випадках використовують багаторядкову форму розгалуження?
- Опишіть загальний вигляд багаторядкової форми розгалуження.

**Питання для роздумів**

- Чим відрізняється однорядкова форма розгалуження від багаторядкової?
- У яких випадках доцільно використовувати логічні змінні в програмному коді?



Завдання для досліджень

- 1*. Як зміниться програмний код у вправі 49, якщо $a = x \leq 0$: $b = x - 2 = 0$?
- 2*. Як зміниться програмний код у вправі 50, якщо після слова If буде вказано умову $a > 5$?

§39

Вкладені розгалуження

Багаторядкова форма розгалуження має також *розширену* модифікацію, якщо в програмі розглядається більше однієї умови.

```
If <умова1> Then
  <команда 1>
ElseIf <умова2> Then
  <команда 2>
Else
  <команда 3>
End If
```

Їй відповідає блок-схема, зображена на рис. 6.29.

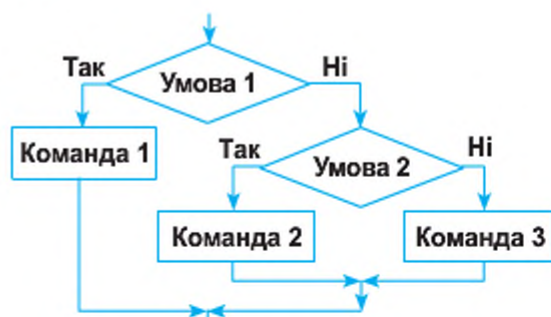


Рис. 6.29

ВПРАВА 52



Завдання. Створити проект розв'язання задачі.

Дано ціле число. Якщо воно додатне, то додати до нього 1; якщо від'ємне, то додати до нього 2; якщо дорівнює 0, то додати 3. Вивести змінене число.

1. Визначте вхідні та вихідні дані.
Вхідні: a — дане число.
Вихідні: a — змінене число.
2. Побудуйте блок-схему реалізації алгоритму розв'язання задачі (рис. 6.30).

Зауваження. Як видно з блок-схеми, тут розглядається лише дві умови $a < 0$ та $a > 0$, оскільки умова $a = 0$ є протилежною до двох попередніх. Отже, її не потрібно вказувати, бо вона враховується в гілці «інакше» другої умови.

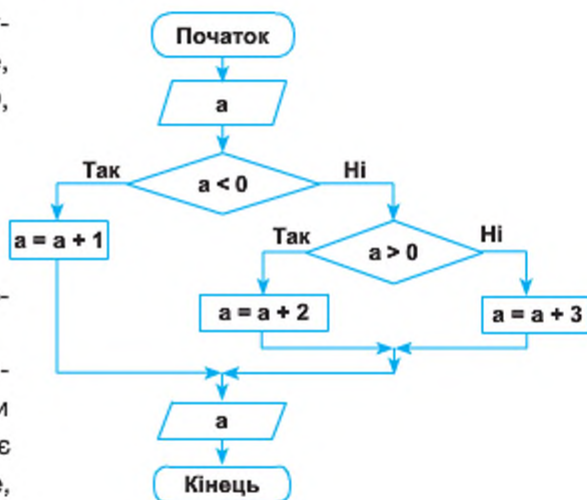


Рис. 6.30



3. Запишіть програмний код для події клацання на формі.

```
Dim a As Integer
a = InputBox(«a =»)
'перевірка умови 1
If a > 0 Then
    a = a + 1
'перевірка умови 2
ElseIf a < 0 Then
    a = a + 2
Else
    a = a + 3
End If
MsgBox(«змінене число =» & a)
```

4. Запустіть проект та перевірте правильність його виконання.
5. Збережіть проект під назвою **Вправа_52_Прізвище**.

ВПРАВА 53



Завдання. Створити проект для знаходження значення виразу.

$$y = \begin{cases} 2x - 5, & x < -3 \\ x^2, & -3 \leq x \leq 3 \\ |10 - x|, & x > 3 \end{cases}$$

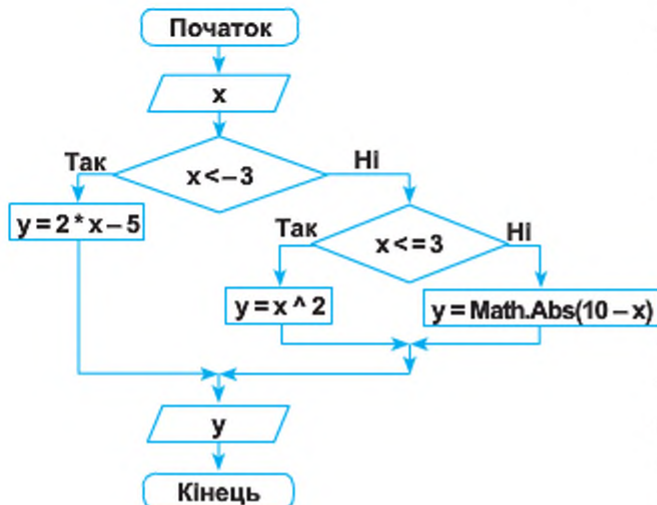


Рис. 6.31

1. Визначте вхідні та вихідні дані.
Вхідні: x — дане число.
Вихідні: y — результат обчислення.
2. Побудуйте блок-схему реалізації алгоритму розв'язання задачі (рис. 6.31).

Зауваження. Як видно з блок-схеми, умова 2 ($x \leq 3$) має тільки другу частину умови подвійної нерівності ($-3 \leq x \leq 3$), вказаної в завданні. Це пов'язано з тим, що умова ($-3 \leq x$) вже врахована, оскільки вона є протилежною до вказаної умови 1 ($x > -3$).



Виконати проект «Тестування» ви зможете на диску.

3. Запишіть програмний код для події клацання на формі.

```
Dim x, y As Single
If x < - 3 Then
    y = 2 * x - 5
ElseIf x <= 3 Then
    y = x ^ 2
```



```
Else
    y = Math.Abs(10 - x)
End If
MsgBox("y=" & y)
```

- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_53_Прізвище**.

**Перевірити**

свої знання

і пройти тест ви зможете на диску.

**Висновки**

При застосуванні *вкладених умов* використовують багаторядкову форму розгалуження, що має вигляд:

```
If <умова1> Then
    <команда 1>
ElseIf <умова2> Then
    <команда 2>
Else
    <команда 3>
End If
```

**Контрольні запитання та завдання**

- Які особливості має багаторядкова форма розгалуження з вкладеними умовами?
- Наведіть приклади використання вкладених умов.

**Питання для роздумів**

- Як можна записати умову: $-5 \leq x \leq 5$?

**Завдання для досліджень**

- Як зміниться програмний код у вправі 52, якщо вказати $a = 0$ як першу умову, а як другу — $a < 0$?
- Як зміниться програмний код у вправі 53, якщо як першу умову вказати $-3 \leq x \text{ And } x \leq 3$, а як другу — $x > 3$?

§40**Складання алгоритмів опрацювання величин, їх налагодження і виконання**

Якщо припуститися помилки в написанні програмного коду, то система визначить її автоматично та повідомить про це. І навіть підкаже, як виправити помилку. Проте існують випадки, коли програма буде складена без помилок, але результат її роботи буде не правильним. Це

означає, що є помилка в логіці побудови програмного коду. Такі помилки може усунути тільки програміст. Щоб відслідкувати місце, де знаходиться помилка, зручно виконати програму в покроковому режимі, одержуючи при цьому значення змінних на кожному кроці виконання програми.

Щоб запустити покрокове виконання програми, треба натиснути клавішу **F11**. Після чого перший рядок програмного коду буде підсвічено жовтим кольором і навпроти нього буде розташована жовта стрілка (рис. 6.32).

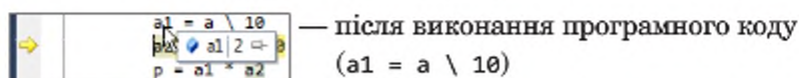
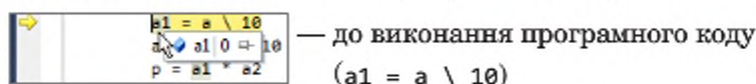
```
Public Class Form1
    Private Sub Button1_Click(sender As System.Object, e As System.EventArgs) Handles Button1.Click
        Dim i, x, S As Integer
        S = 0
    End Sub
End Class
```

Рис. 6.32

Наступне натиснення на клавішу **F11** здійснює виконання програмного коду підсвіченого рядка і переходить до наступного рядка, підсвічуючи його також жовтим кольором.

У будь-який момент покрокового виконання програми користувач може переглянути поточне значення будь-якої величини в програмному коді, навівши на неї курсор миші.

Так при тестуванні програмного коду доцільним буде визначення значення змінної до виконання програмного коду і після.



Проте кожний раз наводити курсор на змінну не дуже зручно. Щоб вирішити цю проблему, в середовищі передбачено закріплення бажаних змінних таким чином, аби їх значення було постійно відображено при покроковому виконанні програми. Це можна зробити після наведення курсора на змінну та натиснення значка у вигляді канцелярської кнопки. До того ж у цьому режимі можна ввести коментар до змінної. Це зручно коли потрібно вказати її призначення. Закріплена змінна буде мати такий вигляд, як на рис. 6.33.

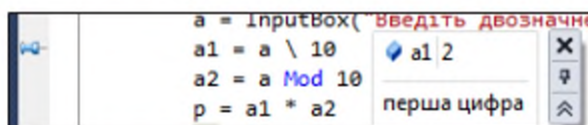


Рис. 6.33

Відмінити закріплення змінних можна, навівши на змінну та обравши у відповідному меню змінної команду **Открепить от источника** (Від'єднати від джерела).

Також є можливість перегляду поточного значення усіх змінних, визначених у проєкті в окремому вікні, яке можна вивести, обравши в меню **Отладка**



(Налагодження) команди **Окна** ▶ **Локальные** (Вікна ▶ Локальні). У вікні **Локальные** (Локальні) відображено імена змінних, їх значення та тип (рис. 6.34).

Имя	Значение	Тип
Me	{WindowsApplication1.F	WindowsApplication1.Form1
a	28	Integer
a1	2	Integer
a2	8	Integer
e	{X = 51 Y = 61 Button = Left	System.EventArgs
p	16	Integer
sender	{WindowsApplication1.F	Object

Рис. 6.34. Вікно **Локальные** (Локальні)

Якщо код програми громіздкий, і покрокова перевірка всього коду недоцільна, можна встановити покажчик зупинки, починаючи з якого, буде здійснено покрокове виконання коду. Для цього треба обрати в контекстному меню потрібного рядка команду **Точка останова** ▶ **Вставити точку останова** (Точка зупинки ▶ Додати точку зупинки) або просто клацнути мишею на сірому стовпчику зліва навпроти рядка. Після чого рядок обраного коду буде виділено червоним і навпроти даного рядка буде встановлено червоний кружечок (рис. 6.35).

У цьому випадку необхідно здійснити запуск проекту в звичайному режимі за допомогою кнопки **Начать отладку** (F5). Після чого код

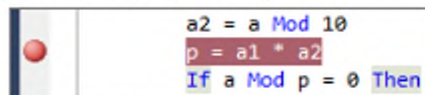


Рис. 6.35

буде виконано до виділеного рядка і вже потім, натискаючи послідовно **F11**, ви зможете покроково виконати бажаний фрагмент програми.

Щоб зняти покажчик зупинки, треба натиснути на червоний кружечок, який встановлено навпроти відповідного рядка коду.



Висновки

Для перевірки роботи програми зручно використовувати покрокове виконання програми, натискаючи клавішу **F11**.

Щоб дізнатись значення змінної при покроковому виконанні програми, треба навести на неї покажчик миші. Для закріплення змінної слід натиснути на значок у вигляді канцелярської кнопки.

Встановлення покажчика зупинки дозволяє здійснити покрокове виконання програми в будь-якій її частині.



Контрольні запитання та завдання

1. Як запустити покрокове виконання програми?
2. Як закріпити змінну при покроковому виконанні програми?
3. Як виконати покрокове виконання частини програми?



Питання для роздумів

1*. Для чого використовують покрокове виконання програми?



Завдання для досліджень

1*. Як буде здійснюватися покрокове виконання програми, якщо встановити декілька покажчиків зупинки?

2**. Дослідіть призначення команд у меню **Отладка** (Налагодження). Поясніть їх призначення.

ПРАКТИЧНА РОБОТА №11

Налагодження готової програми

Завдання. Створити проект для розв'язання задачі.

Дано двоцифрове число. Визначити, чи кратне число до добутку його цифр.

Здійснити покрокову перевірку виконання програмного коду.

1. Визначте вхідні, вихідні та проміжні змінні.

Вхідні: a — дане число.

Вихідні: текстове повідомлення.

Проміжні: a_1 , a_2 — цифри числа, p — добуток цифр числа.

2. Введіть програмний код для реалізації процедури натиснення на форму.

```
Dim a, a1, a2, p As Integer
```

```
a = InputBox("Введіть двоцифрове число")
```

```
'Знаходження першої цифри числа
```

```
a1 = a \ 10
```

```
'Знаходження другої цифри числа
```

```
a2 = a Mod 10
```

```
'Знаходження добутку цифр числа
```

```
p = a1 * a2
```

```
If a Mod p = 0 Then MsgBox ("Кратне") Else MsgBox ("Не кратне")
```

3. Натисніть клавішу F11. У результаті з'явиться форма, на якій необхідно клацнути по кнопці. Після чого виконання програми буде передано заголовку процедури клацання на формі (рис. 6.36).

```
Form1.vb x Form1.vb (Конструктор)
(События Form1)
Public Class Form1
Private Sub Form1_Click(sender As Object, e As System.EventArgs) Handles Me.Click
    Dim a, a1, a2, p As Integer
    a = InputBox("Введіть двоцифрове число")
    a1 = a \ 10
    a2 = a Mod 10
    p = a1 * a2
    If a Mod p = 0 Then MsgBox("Кратне") Else MsgBox("Не кратне")
End Sub
End Class
```

Рис. 6.36



4. Натисніть клавішу F11. Буде виконано рядок попереднього коду і здійснено перехід до наступного коду. Наведіть курсор на змінну a . Її значення дорівнює 0 (рис. 6.37).

```
Private Sub Form1_Click(sender As Object, e As EventArgs)
    Dim a, a1, a2, p As Integer
    a = InputBox("Введіть двоцифрове число")
    a 0 0 ⇐
    a2 = a Mod 10
```

Рис. 6.37

5. Натисніть клавішу F11. Буде виконано рядок попереднього коду ($a = \text{InputBox}(\text{«Введіть двоцифрове число»})$), з'явиться вікно для введення двоцифрового числа. Введіть число, наприклад, 24 та натиснення на кнопку **Ок**. Буде підсвічено наступний код. Наведіть курсор на змінну a . Її значення буде 24 (рис. 6.38). Наведіть курсор на змінну $a1$, її значення дорівнює 0 (рис. 6.39).

```
a = InputBox("Введіть двоцифрове число")
a 24 ⇐
a2 = a Mod 10
p = a1 * a2
```

Рис. 6.38

```
a = InputBox("Введіть двоцифрове число")
a1 = a \ 10
a2 a1 0 ⇐
p = a1 * a2
```

Рис. 6.39

6. Натисніть клавішу F11. Буде виконано рядок попереднього коду ($a1 = a \setminus 10$) і здійснено перехід до наступного коду. Наведіть курсор послідовно на змінні $a1$ та $a2$; $a1$ одержить значення 2 (рис. 6.40), а значення змінної $a2$ дорівнюватиме 0 (рис. 6.41).

```
a1 = a \ 10
a1 2 ⇐
p = a1 * a2
```

Рис. 6.40

```
a1 = a \ 10
a2 = a Mod 10
p a2 0 ⇐
```

Рис. 6.41

7. Натисніть клавішу F11. Буде виконано рядок попереднього коду ($a2 = a \text{ Mod } 10$) і здійснено перехід до наступного рядка. Наведіть курсор послідовно на змінні $a2$ і p ; $a2$ одержить значення 4 (рис. 6.42), а значення змінної p дорівнюватиме 0 (рис. 6.43).

```
a2 = a Mod 10
p a2 4 ⇐
If a Mod p = 0 Then
```

Рис. 6.42

```
a2 = a Mod 10
p = a1 * a2
p 0 ⇐ = 0 Then
```

Рис. 6.43

8. Натисніть клавішу F11. Буде виконано рядок попереднього коду ($p = a1 * a2$) і здійснено перехід до перевірки умови ($\text{If } a \text{ Mod } p = 0$). Наведіть курсор на змінну p , її значення стало дорівнювати 8 (рис. 6.44).

```
p = a1 * a2
p 8 ⇐ = 0 Then MsgBox("Кратне") Else MsgBox("Не кратне")
End Sub
```

Рис. 6.44



9. Натисніть клавішу **F11**. Буде здійснено перевірку умови, і оскільки значення логічного виразу має значення `True`, відбудеться перехід до виразу, що стоїть після команди `Then`, а саме `MsgBox("Кратне")` (рис. 6.45).

```
p = a1 * a2
If a Mod p = 0 Then MsgBox("Кратне") Else MsgBox("Не кратне")
End Sub
```

Рис. 6.45

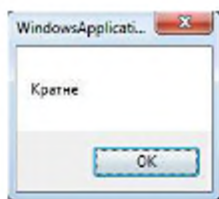


Рис. 6.46

10. Натисніть клавішу **F11**. Буде виведено текстове повідомлення (рис. 6.46).
11. Зупиніть виконання програми.
12. Закріпіть змінні, щоб не треба було кожного разу наводити на них вказівник миші. Для цього наведіть на деяку змінну курсор миші та клацніть по значку у вигляді канцелярської кнопки. Після чого перемістіть значок із змінною в зручне для вас місце (рис. 6.47).

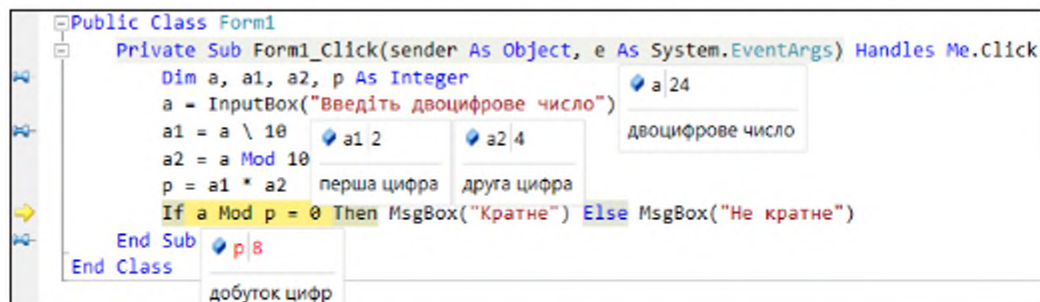


Рис. 6.47

13. Здійсніть покрокове виконання програми із закріпленими змінними, ввівши двоцифрове число, для якого не буде виконуватися умова, наприклад 27.
14. Здійсніть покрокове виконання програми, починаючи від команди $p = a1 * a2$. Для цього створіть покажчик зупинки для даного рядка. Після чого натисніть клавішу **F5**. Коли відбудеться зупинка роботи програми натисніть **F11** необхідну кількість разів до завершення роботи програми.
15. Змініть програмний код для випадку введення трицифрового числа і перевірки кратності добутку його цифр до самого числа.
16. Змініть усі покажчики та запустіть проект на виконання у звичайному режимі.
17. Збережіть проект під назвою **PR11_Прізвище**.



§41

Алгоритми з повтореннями. Цикл з лічильником

41.1.

ЦИКЛ З ПАРАМЕТРОМ

З сьомого класу ви пам'ятаєте, що багаторазове виконання тієї самої послідовності команд доти, доки виконується визначена наперед умова, називають *циклом* (повторенням). Алгоритм, що містить цикли, називають *алгоритмом з повторенням*.

Майже всі мови програмування містять оператори циклів. Не є винятком і VB.NET. Однією з властивостей комп'ютера є його здатність багато разів виконувати ті самі дії. З повторюваними обчисленнями програмісти стикаються повсякчас. Є цикли, в яких кількість повторювань відома заздалегідь, і цикли, де кількість повторень залежить від виконання певної умови.

Цикли, в яких кількість виконуваних повторювань відома заздалегідь, називають *циклами з лічильником*. Тому, що для організації повторень використовують змінну (лічильник), яка змінюється від певного початкового значення до кінцевого значення з заданим кроком.

Програмний код такого циклу має загальний вигляд:

```
For <лічильник>=<початкове значення> To <кінцеве значення>_
```

```
Step <крок> <оператори тіла циклу>
```

```
Next <лічильник>,
```

де <лічильник> — числова змінна;

<початкове значення>, <кінцеве значення> і <крок> — вирази того самого типу, що й лічильник.

Якщо конструкцію Step <крок> пропущено, то вважається, що значення кроку дорівнює +1.

Після службового слова Next лічильник не є обов'язковим параметром, його доцільно вказувати, коли програма містить багато циклів з різними лічильниками.

Блок-схема, що відповідає даному виду циклу має вигляд, як зображено на рис. 6.48.

Алгоритм виконання програмного коду такий:

- лічильнику присвоюють початкове значення. Якщо воно менше від кінцевого значення, то здійснюється перехід до виконання операторів тіла циклу;
- після виконання операторів тіла циклу команда Next змінює значення лічильника на значення кроку;



Рис. 6.48



- 3) якщо значення лічильника менше від кінцевого значення, програма знову переходить до виконання операторів тіла циклу. Такі порівняння виконуються доти, доки значення лічильника знаходиться в діапазоні між початковим і кінцевим значеннями. Коли значення лічильника виходить за межі діапазону, виконання оператора циклу закінчується і здійснюється перехід до команд, що знаходяться після команди **Next**.

Якщо початкове значення циклу є більшим за кінцеве, тіло циклу жодного разу не виконається. Винятком є ситуація, коли значення кроку від'ємне. Проте в цьому випадку і алгоритм виконання циклу буде здійснюватися з врахуванням такого значення кроку.

ВПРАВА 54



Завдання. Створити проект виведення у стовпчик n перших натуральних чисел у прямому та зворотному порядку.

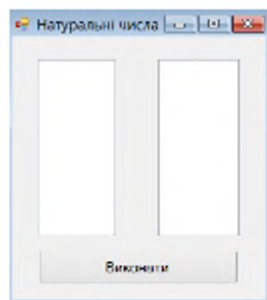


Рис. 6.49

1. Створіть форму за зразком (рис. 6.49).
2. Надайте текстовим полям імена **T1** і **T2**, змініть значення властивості **Multiline** = **True**.
3. Введіть код для виведення послідовностей чисел при натисненні на кнопку **Виконати**.

```
Dim i, n As Integer
'Визначення кількості чисел
n = InputBox(«N=»)
'Очищення текстових полів від попереднього вмісту
T1.Text = Nothing
T2.Text = Nothing
'Виведення натуральних чисел від 1 до N
For i = 1 To n
    T1.Text = T1.Text + CStr(i) + vbCrLf
Next
'Виведення натуральних чисел від N до 1
For i = n To 1 Step -1
    T2.Text = T2.Text + CStr(i) + vbCrLf
Next
```

Пояснення. Найчастіше як лічильник використовують змінну i , як i в представленому коді.

`CStr(i)` — функція, що переводить числовий тип значення змінної в рядковий. Оскільки текстове поле, куди здійснюється запис, може набувати значення лише рядкового типу.

`vbCrLf` — команда для переходу на новий рядок.

Для представлення чисел у зворотному порядку було використано цикл, що містить від'ємний крок (-1).

4. Запустіть проект та перевірте правильність його виконання.
5. Збережіть проект під назвою **Вправа_54_Прізвище**.



41.2.

ЗНАХОДЖЕННЯ СУМИ, ДОБУТКУ, КІЛЬКОСТІ ТА СЕРЕДЬНОГО АРИФМЕТИЧНОГО

Досить часто розв'язуючи різноманітні задачі, виникає потреба визначити суму, добуток, кількість або середнє арифметичне послідовності деяких чисел. Загальний вигляд програмних кодів для виконання цих дій розглянемо далі.

Знаходження суми

‘Початкове значення

$s = 0$

For...To...

‘Знаходження суми, де a будь-яке число (деякий вираз)

$s = s + a$

Next

Знаходження добутку

‘Початкове значення

$p = 1$

For...To...

‘Знаходження добутку, де a будь-яке число (деякий вираз)

$p = p * a$

Next

Знаходження кількості

‘Початкове значення

$k = 0$

For...To...

‘Знаходження кількості, де a будь-яке число (деякий вираз)

$k = k + 1$

Next

У цих формулах перший оператор присвоєння (початкове значення) виконується один раз перед циклом, другий оператор виконується декілька разів у циклі, при цьому число a (це може бути й інша змінна або вираз) кожного разу інше.

Знаходження середнього арифметичного

‘Початкове значення

$Sa = 0$

‘Знаходження середнього арифметичного, де s – сума чисел, а k – їх кількість

If $k <> 0$ Then $Sa = s / k$

Ця формула використовується один раз після знаходження суми та кількості. Перевірка If $k <> 0$ потрібна, бо якщо $k = 0$, отримаємо ділення на 0, що неприпустимо.

ВПРАВА 55



Завдання. Створити проект знаходження суми, кількості та середнього арифметичного всіх натуральних двоцифрових чисел, у яких перша цифра менша від другої. Усі числа, що задовольняють умову, вивести в стовпчик у текстовому полі.

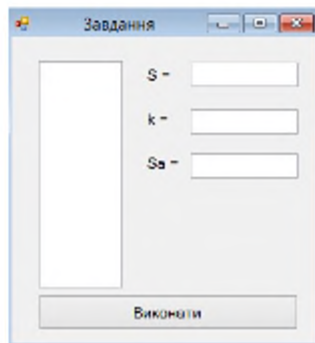


Рис. 6.50

1. Визначте вхідні та вихідні дані, проміжні змінні.

Вхідні: немає.

Вихідні: S , k , Sa — сума, кількість та середнє арифметичне, послідовність чисел, що задовольняє умову.

Проміжні змінні: i — лічильник, x_1 , x_2 — перша та друга цифри числа.

2. Створіть форму за зразком (рис. 6.50).

3. Текстове поле, що має містити послідовність чисел, назвіть **Pos1**, поля, в яких буде визначено S , k , Sa — **TS**, **Tk**, **TSa** відповідно.

4. Введіть код для виконання обчислень при натисненні на кнопку **Обчислити**.

```
Dim i, x1, x2, S, k As Integer, Sa As Single
S = 0: k = 0
For i = 10 To 99
    'Визначення першої цифри числа
    x1 = i \ 10
    'Визначення другої цифри числа
    x2 = i Mod 10
    'Перевірка умови, чи є перша цифра числа менша від другої
    If x1 < x2 Then
        'Виведення числа, що задовольняє умову
        Pos1.Text = Pos1.Text + Cstr(i) + vbNewLine
        'Знаходження суми
        S = S + i
        'Знаходження кількості
        k = k + 1
    End If
Next
'Знаходження середнього арифметичного
If k <> 0 Then Sa = S / k
'Виведення суми, кількості, середнього арифметичного
TS.Text = S
Tk.Text = k
TSa.Text = Sa
```

5. Запустіть проект та перевірте правильність його виконання.

6. Збережіть проект під назвою **Вправа_55_Прізвище**.

Розглянемо в таблиці поданій нижче покрокове виконання програмного коду вправі 64 для i від 10 до 15.

№ кроку	i	$i \leq 15$	x_1	x_2	$x_1 < x_2$	S	k	$k <> 0$	Sa
1	10	True	1	0	False	0	0		
2	11	True	1	1	False	0	0		
3	12	True	1	2	True	12	1		
4	13	True	1	3	True	12+13=25	2		
5	14	True	1	4	True	25+14=39	3		
6	15	True	1	5	True	39+15=54	4		
7	16	False						True	54 / 4 = 13,5



У таблиці в стовпчиках записано значення змінних на кожному кроці виконання програми та результати виконання умов, що мають місце на вказаних кроках.

ВПРАВА 56**САМОСТІЙНО**

Завдання. Створити проекти розв'язання завдань.

- Знайти середнє арифметичне всіх натуральних двоцифрових чисел кратних 13.
- Знайти суму всіх натуральних двоцифрових чисел, що закінчуються на цифру 3.
- Знайти суму, кількість і середнє арифметичне всіх натуральних двоцифрових чисел, які складаються з парних цифр.
- Знайти добуток усіх натуральних двоцифрових чисел, у яких перша цифра вдвічі більша від другої.

41.3.**СТВОРЕННЯ ТАБЛИЦЬ ЗНАЧЕНЬ ФУНКЦІЙ**

У задачах на створення таблиць значень функцій потрібно обчислити значення функції на якомусь інтервалі із заданим постійним кроком. *Крок* — це різниця між значеннями аргумента, для яких обчислюється значення функції.

ВПРАВА 57

Завдання. Створити проект розв'язання задачі.

Знайти значення функції для x з інтервалу $[-2, 2]$ з кроком 0,2.

Тобто потрібно обчислити значення функції для: $x = -2, 0, -1, 8, \dots, 1, 8, 2$.

- Визначте вхідні та вихідні дані, проміжні змінні.
Вхідні: немає.
Вихідні: y — значення функції при різних значеннях x (буде виведено 21 раз).
Проміжні змінні: x — аргумент функції.
- Розташуйте на формі текстове поле та кнопку **Обчислити**.
- Текстове поле назвіть **T**. Змініть значення властивості **Multiline** = True.
- Введіть код для виконання обчислень при натисненні на кнопку **Обчислити**.

```
Dim x, y As Single
```

```
T1.Text = Nothing
```

```
T2.Text = Nothing
```

```
For x = -2 To 2 Step 0.2
```

```
    'Обчислення значення функції
```

```
    y = x ^ 2
```

```
    'Виведення до текстового поля значень X та Y
```

```
    T1.Text = T1.Text + CStr(Format(x, «0.00»)) + vbTab + _
```

```
    CStr(Format(y, «0.00»)) + vbNewLine
```

```
Next
```

**Дізнатися більше**

про програмування з використанням випадкових чисел та **ВИКОНАТИ** додаткові вправи ви зможете на диску.



Оскільки x та y змінні дійсного типу, то для зручного їх відображення застосовано функцію формату з виведенням двох знаків після коми.

`vbTab` — константа для здійснення відступу між числами.

- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_57_Прізвище**.



Висновки

Багаторазове виконання тієї самої послідовності команд доти, доки виконується визначена наперед умова, називають *циклом*.

Алгоритм, що містить цикли, називають *алгоритмом з повторенням*.

Цикли, в яких кількість виконуваних повторювань відома заздалегідь, називають *циклами з лічильником*.

Програмний код циклу з лічильником має вигляд:

```
For <лічильник>=<початкове значення> To <кінцеве значення>_  
Step <крок>  
  <оператори тіла циклу>  
Next <лічильник>
```

Якщо початкове значення циклу більше від кінцевого, то тіло циклу жодного разу не виконається.



Контрольні запитання та завдання

- Що таке цикл у програмуванні?
- Які бувають види циклів?
- Які цикли називають циклами з лічильником?
- Яке призначення службового слова **Step**?
- Опишіть алгоритм виконання циклу з лічильником.
- Як виконати додавання (множення) послідовності чисел?
- Як знайти кількість чисел послідовності?



Питання для роздумів

- Наведіть приклади використання циклів у повсякденному житті.
- Скільки разів буде виведено цикл у заданих командах?
 - `For i = 1 To 10 Step 2`
 - `For i = 1 To 10 Step -2`
 - `For i = 15 To 1 Step -3`
 - `For i = 15 To 25`
 - `For i = 25 To 10`
- Якого типу має бути змінна лічильника в циклі **For**?
- Що відбувається, коли початкове значення лічильника циклу `For...To` більше від кінцевого?



Завдання для досліджень

- 1*. Яким буде програмний код для знаходження значення виразу:
 $S = 1 + 2 + 3 + \dots + N$.
- 2*. Визначте, як можна вивести таблицю значень функцій (вправа 66)
за допомогою елемента керування **ListBox**.

§42

Цикли з умовами

Як було зазначено в попередньому параграфі, при виконанні циклів кількість кроків у них не завжди є відомою, а залежить від певної умови. Саме для таких випадків використовують *цикли з умовами*.

Цикл, тіло якого виконується доти, доки виконується умова, вказана перед його початком, називають *циклом з передумовою*. Якщо задана умовою подія не настає ніколи, тіло циклу не виконується жодного разу.

Цикл з післяумовою — це цикл, у якому умова перевіряється після виконання його тіла.

Програмні коди представлених видів циклів та відповідні до них блок-схеми наведено в таблиці 6.25.

Таблиця 6.25. Цикли з передумовою та післяумовою

	Цикл з передумовою	Цикл з післяумовою
Програмний код	<pre>While <умова> <оператори тіла циклу> End While</pre>	<pre>Do <оператори тіла циклу> Loop</pre>
Блок-схема		
Пояснення	<p>Спочатку перевіряється умова. Якщо вона є істинною, виконуються оператори тіла циклу, після чого весь процес повторюється, доки виконується умова. Якщо спочатку умова не виконується, то тіло циклу жодного разу виконано не буде.</p>	<p>Виконуються оператори, після чого перевіряється умова. Якщо умова не виконується, здійснюється перехід до виконання операторів циклу. Якщо умова виконується — відбувається вихід із циклу. Із блок-схеми видно, що тіло циклу за будь-яких обставин має бути виконане принаймні один раз.</p>



ВПРАВА 58



Завдання. Створити проект обчислення значення виразу:

$$S = 2 + 4 + 6 + \dots (n \text{ разів})$$

- Визначте вхідні та вихідні дані, проміжні змінні.
Вхідні: n — кількість доданків.
Вихідні: S — значення виразу.
Проміжні змінні: i — лічильник.
- Введіть один із запропонованих кодів для виконання обчислень при натисненні на форму.
Для порівняння наведемо приклади трьох способів розв'язання задачі з використанням різних видів циклів.

Цикл з передумовою	Цикл з післяумовою	Цикл з лічильником
<pre>Dim i, n, S As Integer n = InputBox(«n=») S = 0 i = 1 While i <= n S = S + i * 2 i = i + 1 End While MsgBox(«S=» & S)</pre>	<pre>Dim i, n, S As Integer n = InputBox(«n=») S = 0 i = 1 Do S = S + i * 2 i = i + 1 Loop i > n MsgBox(«S=» & S)</pre>	<pre>Dim i, n, S As Integer n = InputBox(«n=») S = 0 For i = 1 to n S = S + i * 2 Next MsgBox(«S=» & S)</pre>

Як видно з програмних кодів, у першому і другому способі умови є протилежними. В першому коді цикл буде виконуватися, доки лічильник i буде менший або дорівнює n . У другому — цикл буде завершено, коли i стане більше від n .

- Запустіть проект та перевірте правильність його виконання.
- Збережіть проект під назвою **Вправа_58_Прізвище**.

ВПРАВА 59



Завдання. Створити проект визначення кількості та суми цифр даного числа.

Вхідні дані	$a = 12456$	$a = 965$
Вихідні дані	$k = 5; S = 18$	$k = 3; S = 20$

- Визначте вхідні та вихідні дані, проміжні змінні.
Вхідні: a — дане число.
Вихідні: S, k — сума та кількість цифр числа.
Проміжні змінні: c — поточна остання цифра числа.



2. Введіть код для виконання обчислень при натисненні на форму.

```
Dim k, S, a, c As Integer
'Введення даного числа
a = InputBox("a=")
k = 0
S = 0
While a <> 0
    'Відокремлення останньої цифри числа
    c = a Mod 10
    'Визначення кількості цифр
    k = k + 1
    'Визначення суми цифр
    S = S + c
    'Відкидання останньої цифри числа
    a = a \ 10
End While
MsgBox("S=" & S)
MsgBox("k=" & k)
```

3. Запустіть проект та перевірте правильність його виконання.

4. Збережіть проект під назвою **Вправа_59_Прізвище**.

ВПРАВА 60



Завдання. Створити проект визначення кількості входжень першої цифри даного числа.

Вхідні дані	Вихідні дані
$a = 1231$	$kp = 2$
$a = 5105058$	$kp = 3$

1. Визначте вхідні та вихідні дані, проміжні змінні.

Вхідні: a — дане число.

Вихідні: kp — кількість входжень першої цифри.

Проміжні змінні: a_1 — перша цифра числа; c — поточна цифра числа; k — кількість цифр числа; b — набуває початкового значення даного числа, використовується для відокремлення цифр.

2. Введіть код для виконання обчислень при натисненні на форму.

```
Dim k, c, x1, b, kp As Integer, a As Long
a = InputBox("a=")
k = 0
b = a
'Визначення кількості цифр числа
While b <> 0
    c = b Mod 10
    k = k + 1
    b = b \ 10
End While
'Визначення першої цифри числа
x1 = a \ 10 ^ (k - 1)
```



```
'Визначення кількості входжень першої цифри
```

```
While a <> 0  
    c = a Mod 10  
    If c = x1 Then kp = kp + 1  
    a = a \ 10  
End While  
MsgBox(«kp=» & kp)
```

3. Запустіть проект та перевірте правильність його виконання.
4. Збережіть проект під назвою **Вправа_60_Прізвище**.



Висновки

Якщо кількість кроків у циклі є заздалегідь невідомою, використовують *цикли з умовами*.

Цикл, тіло якого виконується доти, доки істинною залишається вказана перед його початком умова, називають *циклом з передумовою*.

Програмний код циклу з передумовою має вигляд:

```
While <умова>  
    <оператори тіла циклу>  
End While
```

Цикл з післяумовою — це цикл, у якому умова перевіряється після виконання його тіла.

Програмний код циклу з лічильником має вигляд:

```
Do  
    <оператори тіла циклу>  
Loop
```



Контрольні запитання та завдання

1. Що таке цикл з передумовою?
2. Що таке цикл з післяумовою?
3. Які службові слова використовують для реалізації циклів з умовою?



Питання для роздумів

1. Чому всі сучасні мови програмування використовують команди реалізації циклів?
- 2*. Чим відрізняються цикли з передумовою від циклів з післяумовою.
- 3*. Опишіть та порівняйте принципи виконання циклів з лічильником та циклів з умовами.



ПРАКТИЧНА РОБОТА №12

Складання і виконання алгоритмів з повтореннями та розгалуженнями для опрацювання величин

Завдання. Розробити проект реалізації найпростішого калькулятора.

- Створіть форму за зразком на рис. 6.51 (розташування кнопок і кольорове оформлення можете виконати на власний розсуд).
- Надайте імена об'єктам.



Рис. 6.51

Об'єкт	Ім'я	Об'єкт	Ім'я
Текстове вікно	T1	Кнопка «+»	Bsum
Кнопки з цифрами	B0, B1, B2 ... (відповідно до зображеної цифри)	Кнопка «-»	Briz
Кнопка «*»	Bdob	Кнопка «/»	Bdil
Кнопка «=»	Bdor	Кнопка «C»	BC
Кнопка «,»	Bt		

Опишіть змінні a і b , де a — змінна, що визначає значення введеного числа, b — змінна, що запам'ятовує дію, яку було виконано.

`Dim a As Single, b As String`

- Здайте коди процедур натиснення на кнопки із зображенням цифр та знака «,».

 - Введіть код для процедури натиснення на кнопку «0».

'В текстовому вікні дописується цифра 0

```
T1.Text = T1.Text + «0»
```
 - Для інших кнопок коди створіть аналогічно.

- Здайте коди для процедури натиснення на знак «+».

'Праворуч від попередньої цифри дописується наступна

```
a = a + T1.Text
```

'Змінна b запам'ятовує дію, що було виконано

```
b = «+»
```

- Здайте коди для очищення текстового вікна при переході від кнопок, що позначають арифметичну дію до інших кнопок.

а) У вікні коду як об'єкт оберіть кнопку **Bsum**, а в списку подій оберіть

LostFocus. У результаті буде сформовано код:

```
Private Sub Bsum_LostFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles Bsum.LostFocus
```

```
End Sub
```

б) Між даними рядками введіть код для звільнення текстового поля від написів. `T1.Text = Nothing`

в) Аналогічно введіть коди і для інших кнопок, що позначають арифметичні дії.



6. Задайте код процедури натиснення на клавішу «=».
- a) Введіть код для процедури натиснення на клавішу «=» для виконання дії додавання.
- ```
'якщо змінна b має значення «+», то до попереднього значення змінної a
'додається значення розташоване в текстовому вікні
If b = «+» Then a = a + T1.Text
'В текстовому вікні відображається результат обчислення
T1.Text = a
```
- б) У коді цієї ж процедури додайте коди для виконання інших арифметичних дій.
7. Задайте код для процедури натиснення на кнопку **C** (очищення).
- ```
'очищення текстового поля  
T1.Text = Nothing  
'встановлення нульових значень для змінних  
a = 0  
b = 0
```
8. Додайте кнопку **Bin** для переведення введеного числа в двійкову (Binary) систему числення. Після введення числа та натиснення на кнопку **Bin** у текстовому полі має з'явитися дане число, записане в двійковій системі числення.
- ```
Dim q, x As Integer, y As String
'Одержання числа з текстового поля
q = T1.Text
'Виконувати, поки число більше від нуля
While q > 0
'Знаходження остачі від ділення числа на 2
x = q Mod 2
'Дописання однієї цифри перед значенням рядкової змінної,
'яка має одержати результат
y = CStr(x) + y
'Знаходження цілої частини від ділення
q = q \ 2
End While
'Виведення результату в текстове поле
T1.Text = y
```
9. Додайте кнопки та відповідні до них коди для виконання дій піднесення до степеня та добування кореня.
10. Спробуйте вдосконалити власний калькулятор, щоб з його допомогою можна було виконати послідовно декілька дій.
11. Збережіть проект під назвою **PR12\_Прізвище**.

## §43

### Створення графічних зображень

На формі, крім елементів керування, можна виводити зображення, створені у вигляді сукупності відрізків, прямокутників, кіл, еліпсів тощо. Саме побудові цих та інших графічних примітивів ми присвяtimo матеріал даного параграфу.



Будують фігури відносно умовної координатної площини, що має початок у верхньому лівому куті форми і визначається її розмірами. Так, на рис. 6.52 представлено значення, що мають точки розташовані в різних кутах форми, властивість **Size** якої має значення 300;300.

Рисунок, що створюється на формі, є графічним об'єктом, для оголошення якого на початку програмного коду використовують команду:

```
Dim G As Graphics = MyBase.CreateGraphics
```

У даній команді вказано, що оголошено графічний об'єкт **G**, що володіє методом малювання **CreateGraphics**.

Для побудови контурів геометричних фігур використовують об'єкт перо, яке спочатку оголошують, як нове перо деякого кольору, а потім вже використовують.

```
Наприклад, Dim P As New Pen(Color.Blue)
```

Колір записують у вигляді сталої константи, що з'являється у випадковому списку при наборі коду (**Color.Red**, **Color.Blue** тощо)

Перо може також мати власну ширину, яка задається властивістю **Width**.

```
Наприклад, P.Width = 5
```

У таблиці 6.26 наведено, як створювати графічні примітиви, що можна побудувати за допомогою об'єкта «перо». Зазначимо, якщо перо не оголошено до побудови фігури, його можна записати безпосередньо в коді побудови фігури замість змінної **P** у такому форматі: **Pens.<колір>**. У наведених у табл. 6.26 командах змінною **G** позначено оголошений графічний об'єкт, змінною **P** — перо.

Таблиця 6.26. Команди побудови графічних примітивів за допомогою пера

| Фігура             | Команда мовою програмування                 | Пояснення                                                                            |
|--------------------|---------------------------------------------|--------------------------------------------------------------------------------------|
| <b>Відрізок</b>    | <code>G.DrawLine(P, x1, y1, x2, y2)</code>  | (x1, y1) — початкові координати відрізка,<br>(x2, y2) — кінцеві координати           |
| <b>Прямокутник</b> | <code>G.DrawRectangle(P, x, y, w, h)</code> | (x, y) — координати верхнього лівого кута прямокутника,<br>w — ширина,<br>h — висота |

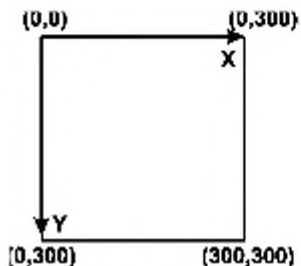
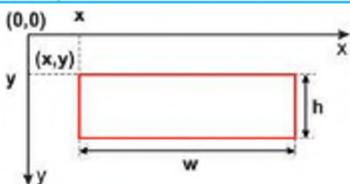
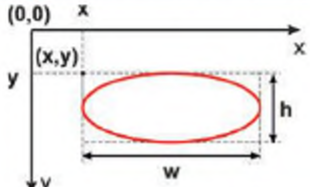
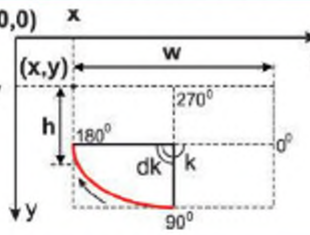
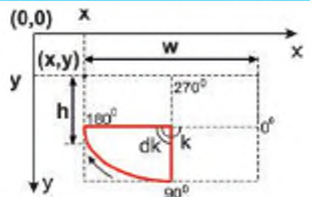


Рис. 6.52



|                                                                                    |                                              |                                                                                                                                                                                                                                                         |
|------------------------------------------------------------------------------------|----------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Еліпс</b>                                                                       | <code>G.DrawEllipse(P, x, y, w, h)</code>    | $(x, y)$ — координати верхнього лівого кута прямокутника описаного навколо еліпса,<br>$w$ — ширина,<br>$h$ — висота прямокутника                                                                                                                        |
|   |                                              |                                                                                                                                                                                                                                                         |
| <b>Дуга</b>                                                                        | <code>G.DrawArc(P, x, y, w, h, k, dk)</code> | $(x, y)$ — координати верхнього лівого кута прямокутника описаного навколо еліпса,<br>$w$ — ширина,<br>$h$ — висота прямокутника,<br>$k$ — початковий кут дуги записаний в градусах,<br>$dk$ — кут зміщення в градусах по колу за годинниковою стрілкою |
|   |                                              |                                                                                                                                                                                                                                                         |
| <b>Сектор</b>                                                                      | <code>G.DrawPie(P, x, y, w, h, k, dk)</code> | $(x, y)$ — координати верхнього лівого кута прямокутника описаного навколо еліпса,<br>$w$ — ширина,<br>$h$ — висота прямокутника,<br>$k, dk$ — ті самі параметри, що й для дуги                                                                         |
|  |                                              |                                                                                                                                                                                                                                                         |

Зафарбовані фігури будують пензлем, який спочатку оголошують, як і перо.

Наприклад, `Dim B As New SolidBrush(Color.Blue)`

За допомогою пензля можна побудувати зафарбовані прямокутник, еліпс і сектор, що мають відповідні команди мовою програмування: `G.FillRectangle(B, x, y, w, h)`, `G.FillEllipse(B, x, y, w, h)`, `G.FillPie(B, x, y, w, h, k, dk)`, де змінною  $B$  позначено пензель, а інші змінні мають те саме призначення, що й для незафарбованих фігур.

Зазначимо, якщо пензель не оголошено до побудови фігури, його можна записати безпосередньо в коді побудови фігури замість змінної  $B$  у наступному форматі: `Brushes.<колір>`.

Для очищення форми від зображення використовують програмний код: `G.Clear(Color.FromName(«Control»))`

### ВПРАВА 61



**Завдання.** Створити проект для побудови зображення за зразком (рис. 6.53).

1. Створіть спочатку зображення на координатній площині в зошиті (рис. 6.54).
2. Встановіть для форми властивість **Size** — 400;300.

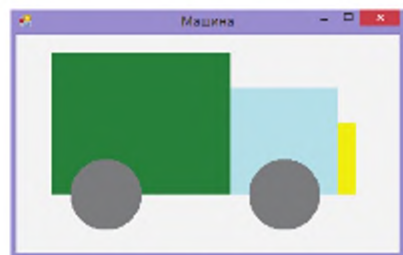


Рис. 6.53

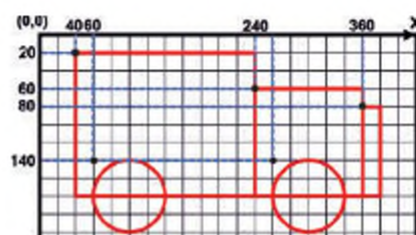


Рис. 6.54

3. Створіть код для опрацювання події клацання по формі, враховуючи розташування зображення відносно координатної площини.

‘Оголошення графічного об’єкта

```
Dim g As Graphics = MyBase.CreateGraphics
```

‘Побудова зеленого прямокутника

```
g.FillRectangle(Brushes.Green, 40, 20, 200, 160)
```

‘Побудова блакитного прямокутника

```
g.FillRectangle(Brushes.LightBlue, 240, 60, 120, 120)
```

‘Побудова жовтого прямокутника

```
g.FillRectangle(Brushes.Yellow, 360, 100, 20, 80)
```

‘Побудова коліс

```
g.FillEllipse(Brushes.Gray, 60, 140, 80, 80)
```

```
g.FillEllipse(Brushes.Gray, 260, 140, 80, 80)
```

4. Збережіть проект під назвою **Вправа\_61\_Прізвище**.



**Виконати**

додаткове завдання  
ви зможете на диску.



**Дізнатися більше**

про встановлення  
стилів контуру ви  
зможете на диску.



### Висновки

Рисунок, що створюється на формі, — це графічний об’єкт, для оголошення якого використовують команду:

```
Dim <ім'я об'єкта> As Graphics = MyBase.CreateGraphics
```

Для побудови контурів геометричних фігур використовують об’єкт «перо», яке оголошують за допомогою команди:

```
Dim <ім'я пера> As New Pen(Color.Blue)
```

У VB визначено методи для побудови таких графічних примітивів, як відрізок, прямокутник, еліпс, дуга та сектор. Прямокутник, еліпс та сектор можна побудувати зафарбованими за допомогою об’єкта «пензель», який оголошують командою:

```
Dim <ім'я пензля> As New SolidBrush(Color.Blue)
```



### Контрольні запитання та завдання

1. Які фігури можна побудувати в графічному режимі VB?
2. Для чого призначені об’єкти «перо» та «пензель»?



3. Які параметри потрібно знати для побудови прямокутника?
4. Яку команду необхідно обрати, щоб побудувати коло? Поясніть принцип його побудови.
5. Чим відрізняється побудова дуги від сектора?



## Питання для роздумів

1. Чим відрізняється побудова контуру прямокутника від зафарбованого прямокутника?
2. Чи можна в програмі описати два об'єкта «перо»?



## Завдання для досліджень

1. Дослідіть, як можна встановити стиль контуру.
2. Визначте, як у графічному режимі можна додати до зображення текст.

## §44

## Побудова зображень з використанням команди повторення

Коли потрібно відтворити кілька однакових фігур, це призводить до багаторазового повторення введення однієї команди. Вміння застосувати в цьому випадку цикл з повторенням дозволяє значно скоротити програмний код, а відповідно і час на його введення. Також це, звичайно, дозволяє уникнути помилок, що часто виникають при роботі з кодами великих обсягів.

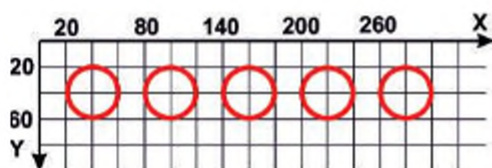


Рис. 6.55

Створимо проект для виведення зображення п'яти кіл, розташованих по горизонталі. Спочатку зробимо заготовку, розташувавши фігури на координатній площині на аркуші паперу (рис. 6.55).

Для виведення даного зображення на формі без використання вказівок повторення необхідно ввести код:

```
'Оголошення графічного об'єкта
Dim g As Graphics = MyBase.CreateGraphics
'Оголошення пера
Dim p As New Pen(Color.Red)
'Побудова кіл
g.DrawEllipse(p, 20, 20, 40, 40)
```





```
g.DrawEllipse(p, 80, 20, 40, 40)
g.DrawEllipse(p, 140, 20, 40, 40)
g.DrawEllipse(p, 200, 20, 40, 40)
g.DrawEllipse(p, 260, 20, 40, 40)
```

Як бачимо, в рядках програмного коду побудови кіл змінюється тільки значення координати  $x$  на сталие число 60. Також для формування коду з використанням циклів врахуємо, що таких кіл має бути 5.

Отже, програмний код для побудови таких кіл з командою повторення буде мати вигляд:

```
'Побудова кіл
For i = 1 To 5
g.DrawEllipse(p, 20 + (i - 1) * 60, 20, 40, 40)
Next
```

Розглянувши код  $(20 + (i - 1) * 60)$  для визначення координати  $x$ , виведемо загальну формулу, яку можна буде використовувати при дублюванні інших зображень, як по горизонталі, так і по вертикалі.

Це може бути формула:  $n + (i - 1) * d$ . Тут  $n$  — початкове значення координати;  $i$  — кількість зображень,  $d$  — відстань між зображеннями.

## ПРАКТИЧНА РОБОТА №13

### Складання та виконання алгоритмів з графічним відображенням даних

**Завдання.** Створити проект побудови стовпчастої діаграми (гістограми), в якому користувач самостійно вводить кількість стовпчиків (від 1 до 10) та значення, що мають відобразити стовпчики (від 0 до 260).

1. Створіть схематично зображення двох стовпчиків діаграми на координатній площині в зошиті (рис. 6.56).
2. Встановіть для форми властивість **Size** — 600;300.
3. Введіть програмний код для процедури натиснення на форму.

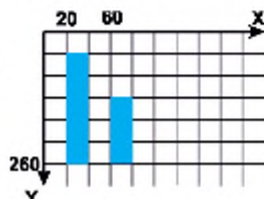


Рис. 6.56

```
Dim i, n, y As Integer
'Введення кількості стовпчиків
n = InputBox(«Введіть кількість стовпчиків від 1 до 10»)
'Оголошення графічного об'єкта та пензля
Dim g As Graphics = MyBase.CreateGraphics
Dim B As New SolidBrush(Color.Blue)
'Очищення форми від попереднього зображення
g.Clear(Color.FromName(«Control»))
For i = 1 To n
'Введення висоти стовпчика
n = InputBox(«Введіть висоту » & i & «-го стовпчика»)
'Побудова стовпчика
g.FillRectangle(B, 20 + (i - 1) * 100, 260 - y, 20, y)
Next
```

**Пояснення.** У коді `g.FillRectangle(x, 20 + (i - 1) * 40, y, 20, 260 - y)` перше число 20 — значення координати  $x$  першого стовпчика;  $40 = 60 - 20$  — різниця між координатами  $x$  першого і другого стовпчика; друге число 20 — ширина стовпчика, 260 — значення координати  $y$ ,  $y$  — висота стовпчика.

4. Самостійно додайте горизонтальну та вертикальну осі для діаграми.
- 5\*. Створіть аналогічний проект з відображенням кругової діаграми.
6. Збережіть проект під назвою **ПР13\_Прізвище**.

## §45

### Відображення рисунків із зовнішніх файлів

Крім створення власного зображення, на форму можна виводити зображення з файла, яке розміщується завдяки елементові керування **PictureBox**. Створюючи деякі проекти, ми вже додавали цей елемент керування, але використовували його лише в режимі конструювання. Розглянемо більш докладно можливості цього елемента керування.

Основною властивістю **PictureBox** є властивість **Image**, що визначає зображення, яке буде використано. Зображення можна завантажити і з зовнішнього носія або додати до ресурсів проекту, а вже потім завантажити до **PictureBox**. Як це зробити, розглянемо при виконанні практичних вправ.

Ще одна властивість — **SizeMode** — може набувати таких значень:

**Normal** — можна вручну змінювати розміри елемента;

**StretchImage** — автоматично розтягує зображення до заданих користувачем розмірів елемента керування;

**AutoSize** — розміри елемента керування розтягуються до розмірів зображення;

**CenterImage** — розташування зображення по центру елемента;

**Zoom** — пропорційне масштабування зображення відповідно до розмірів елемента керування.

Ці ж властивості зручно встановлювати, використовуючи власне меню елемента керування **PictureBox**. Щоб викликати це меню, потрібно натиснути на невеличкий трикутник, розташований у верхньому правому куті елемента (рис. 6.57).

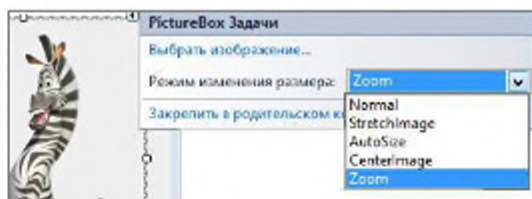


Рис. 6.57

Так посилання **Вибрати зображення...** (Обрати зображення) рівносильне до властивості **Image**, а випадаючий список **Режим изменения размера:** (Режим зміни розміру) дублює властивість **SizeMode**.



## ВПРАВА 62



**Завдання.** Створити проект «Зоопарк», у якому мають відобразитися зображення тварин при наведенні курсору на їх назви. Якщо курсор за межами назви, то зображення повинно зникати.

Зображення для проекту можна взяти в папці **Вправа\_62** на диску або завантажити з мережі Інтернет (рис. 6.58).

- Встановіть для форми заголовок «Зоопарк».
- Додайте зображення іконки.
  - У вікні властивостей форми оберіть властивість **Icon**.
  - Оберіть як зображення файл на комп'ютері **Panda.ico**.

3. Розташуйте на формі елементи керування **Label** та встановіть необхідні властивості.

а) Додайте спочатку один елемент керування **Label** та встановіть для нього властивості, як у даній таблиці.

б) Після встановлення властивостей створіть три копії даного об'єкта, перетягуючи його вниз з утриманням лівої клавіші миші **+Ctrl**.

в) Встановіть для копій значення властивості **Name** — **L\_Z**, **L\_L** і **L\_H** відповідно.

г) Змініть значення властивості **Text** відповідно до вказаних на зразку форми.

4. Додайте до форми зображення та встановіть необхідні властивості.

а) Навпроти написів додайте на форму елемент керування **PictureBox**.

б) Надайте об'єктові ім'я **P\_G**.

в) У меню елемента керування **PictureBox** (верхній правий кут об'єкта) у полі **Режим змінення розміра**: (Режим зміни розміру) оберіть значення **Zoom** (Масштабування), як зображено на рис. 6.59.

г) Для вибору зображення клацніть на посиланні **Вибрати зображення...** (Обрати зображення). В результаті чого з'явиться діалогове вікно **Вибір ресурса** (Вибір ресурсу), як на рис. 6.60.

г) У даному вікні встановіть перемикач у положення **Локальний ресурс** (Локальний ресурс) та натисніть кнопку **Импорт...** (Імпорт). У результаті відкриється вікно для вибору зображення з файлу на комп'ютері.

д) Завантажте до першого елемента керування **PictureBox** файл **giraffe.png**.

е) Виконайте копіювання об'єкта **P\_G**, розташовуючи копію зверху на даний об'єкт.



Рис. 6.58

| Властивість | Значення     |
|-------------|--------------|
| Name        | L_G          |
| Text        | Жираф        |
| AutoSize    | False        |
| Size        | 150; 30      |
| TextAlign   | MiddleCenter |
| Font.Size   | 12           |
| BackColor   | Салатовий    |
| Visible     | False        |

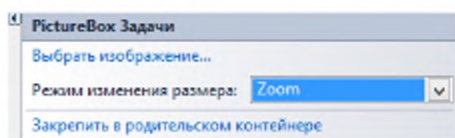


Рис. 6.59

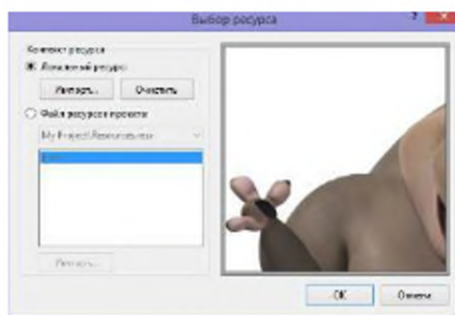


Рис. 6.60



є) Змініть ім'я створеної копії на **P\_Z** та поміняйте зображення, обравши файл **zebra.png**.

ж) Аналогічно створіть копії ще двох елементів керування **PictureBox**, назвавши їх **P\_L** і **P\_H** відповідно. Для об'єкта **P\_L** встановіть зображення **lion.png**, а для **P\_H** — **hippo.png**.

5. Задайте код для події, що має відбуватися у випадку наведення курсору на об'єкт **L\_G** (при наведенні на напис «Жираф» має з'явитися зображення жирафа).

а) Перейдіть у вікно коду. У списку праворуч оберіть об'єкт **L\_G**, у списку праворуч — подію **MouseMove**. В результаті чого мають бути сформовані два рядки програмного коду: заголовок і кінець процедури.

```
Private Sub L_G_MouseMove(sender As Object, e As MouseEventArgs) _
Handles L_G.MouseMove
End Sub
```

б) Між даними рядками введіть код наведений нижче. Тут властивість видимості в елемента керування **P\_G** (зображення жирафа) стає істинною, а в інших об'єктів, що мають зображення, — хибною.

```
P_G.Visible = True
P_H.Visible = False
P_L.Visible = False
P_Z.Visible = False
```

6. Аналогічно створіть коди для наведення курсору миші на інші текстові написи.

7. Задайте програмні коди для наведення курсору миші на форму (зауважте, що в цьому випадку мають зникнути всі зображення).

8. Запустіть проект та перевірте правильність його виконання.

9. Збережіть проект під назвою **Вправа\_62\_Прізвище**.

У даній вправі було розглянуто використання декількох зображень, що були розташовані в різних елементах керування **PictureBox** та були завантажені з файлів, розташованих на диску комп'ютера. В наступній вправі ми також будемо мати справу з декількома зображеннями, які спочатку завантажимо до ресурсів проекту, а потім використаємо в програмному коді. До того ж різні зображення будуть завантажуватися до одного елемента керування **PictureBox** залежно від умови.

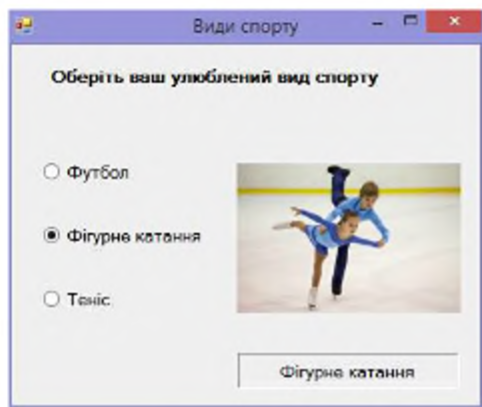
### ВПРАВА 63



**Завдання.** Створити проект «Види спорту», в якому зробити можливим вибір одного з видів спорту за допомогою перемикачів. Після вибору одного з видів спорту з'являється зображення, яке йому відповідає, та відповідний напис.

Зображення для проекту можна взяти в папці **Вправа\_63** на диску або завантажити з мережі Інтернет.

1. Розташуйте на формі елементи керування за зразком (рис. 6.61). Для відображення рисунка скористайтеся елементом керування **PictureBox**.



**Перевірити** свої знання, розгадати кросворд і пройти тест ви зможете на диску.



Рис. 6.61

2. Задайте перемикачам імена **R1**, **R2** і **R3** відповідно. Для мітки, в якій буде розташовано назву виду спорту, надайте ім'я — **L**, для зображення — **P**.
3. Для **PictureBox** властивість **Image** залишіть порожньою, значення властивості **SizeMode** (Зміна розміру) встановіть **Zoom** (Масштабування). Властивості інших елементів керування встановіть самостійно.
4. Додайте до ресурсів проекту зображення з необхідними видами спорту.
  - а) У вікні **Обозреватель решений** клацніть два рази на **My Project**.
  - б) У вікні, що відкрилося оберіть **Ресурси** (Ресурси).
  - в) У меню команди **Добавить ресурс** (Додати ресурс) оберіть **Добавить существующий файл** (Додати файл, що існує).
  - г) Додайте бажані файли з диска комп'ютера (**Футбол**, **Фіг\_кат**, **Теніс**). У вікні **Обозреватель решений** мають відобразитися обрані файли в папці **Resources**.
5. Задайте коди, щоб при виборі за допомогою перемикача одного з видів спорту відображалася його зображення, а знизу — відповідний напис із назвою цього виду спорту.
  - а) Клацніть два рази на перемикачі **R1**. У результаті має з'явитися вікно коду, де буде відображено заголовок процедури **CheckedChanged** (Зміна значення властивості **Checked**), що є подією за умовчанням для елемента керування **RadioButton**.
    - б) Введіть код для даної процедури.  
 'Надання напису тексту, що має R1, а саме слова "футбол"  
`L.Text = R1.Text`  
 'Завантаження зображення файлу "футбол" із папки Resources  
`P.Image = My.Resources.футбол`
    - в) Для інших перемикачів задайте коди самостійно.
6. Запустіть проект та перевірте правильність його виконання.
7. Збережіть проект під назвою **Вправа\_63\_Прізвище**.



### Висновки

Виведення зображення з файла здійснюється завдяки елементові керування **PictureBox**.

Основна властивість **PictureBox** — **Image**, що визначає саме зображення **SizeMode**, яка встановлює спосіб розміщення рисунка.

Вибір властивостей елемента керування **PictureBox** зручно встановлювати за допомогою меню.



### Контрольні запитання та завдання

1. Який елемент керування дозволяє виводити зображення з файла?
2. Яка властивість визначає, яке зображення буде виведено?
3. Як змінити розміри зображення?



### Питання для роздумів

- 1\*. Для чого використовуються ресурси проекту?
- 2\*. Чим відрізняється можливість завантаження рисунків з файлів на диску від завантаження з ресурсів проекту?



### Завдання для досліджень

- 1\*. Дослідіть, де зберігаються файли ресурсів проекту.
- 2\*\*. Властивість **Image** має певний перелік складових. Дослідіть призначення складових цієї властивості.

## Розділ 7

# Технології опрацювання числових даних у середовищі табличного процесора



**§46**

### Абсолютні, відносні й мішані посилання

**46.1.**

#### ПОВТОРЕННЯ

У даному розділі ми продовжимо ознайомлення з табличним процесором. Як ви пам'ятаєте з 7-го класу, в ньому можна виконувати обчислення, подавати дані у вигляді діаграм і графіків, працювати зі списками.

Нагадаємо, що робоче поле табличного процесора MS Excel представлено у вигляді таблиці називають робочим аркушем. Клітинки таблиці мають власні адреси, що складаються з назв стовпців і номерів рядків. Кожна клітинка може містити число, текст або формулу.

Якщо в таблиці потрібно застосувати ту саму формулу декілька разів в одному стовпчику чи рядку, зручно використовувати автозаповнення, при якому копіюються формули з урахуванням нових даних.



#### Пригадати

основні поняття, що пов'язані з табличним процесором, ви зможете на диску.



**46.2.**

#### ПОСИЛАННЯ НА АДРЕСИ КЛІТИНОК

Як вам уже відомо, копіюючи формулу, адреса клітинки, що входить до неї, змінюється залежно від того, куди буде здійснено копіювання. Хоча це не завжди зручно. Іноді в завданні використовують дані, які є сталими і не потребують змін. А отже у формулі вони також мають бути сталими. Розглянемо такий випадок на прикладі наступного завдання.

Відомо ціни товарів у гривнях. Потрібно перевести гривні в холінги (рис. 7.1).

Зрозуміло, що для виконання поставленого завдання, в клітинку C4 треба записати формулу  $=B4/B1$ .

|   | A                   | B         | C          |
|---|---------------------|-----------|------------|
| 1 | Курс гривні/холінги | 10        |            |
| 2 |                     |           |            |
| 3 | Найменування        | Ціна, грн | Ціна, хол. |
| 4 | Пенал               | 100       |            |
| 5 | Рюкзак              | 350       |            |
| 6 | Ручка               | 20        |            |

Рис. 7.1

|   | A                   | B         | C          |
|---|---------------------|-----------|------------|
| 1 | Курс гривні/холінги | 10        |            |
| 2 |                     |           |            |
| 3 | Найменування        | Ціна, грн | Ціна, хол. |
| 4 | Пенал               | 100       | 10,00      |
| 5 | Рюкзак              | 350       | #DIV/0!    |
| 6 | Ручка               | 20        | #VALUE!    |

Рис. 7.2

|   | A                   | B         | C          |
|---|---------------------|-----------|------------|
| 1 | Курс гривні/холінги | 10        |            |
| 2 |                     |           |            |
| 3 | Найменування        | Ціна, грн | Ціна, хол. |
| 4 | Пенал               | 100       | =B4/B1     |
| 5 | Рюкзак              | 350       | =B5/B2     |
| 6 | Ручка               | 20        | =B6/B3     |

Рис. 7.3

|   | A                   | B         | C          |
|---|---------------------|-----------|------------|
| 1 | Курс гривні/холінги | 10        |            |
| 2 |                     |           |            |
| 3 | Найменування        | Ціна, грн | Ціна, хол. |
| 4 | Пенал               | 100       | =B4/B1     |
| 5 | Рюкзак              | 350       | =B5/B1     |
| 6 | Ручка               | 20        | =B6/B1     |

Рис. 7.4

посилання на ту саму клітинку B1. Щоб формула в клітинці C4 у результаті копіювання не перетворювалася на неправильну, адресу клітинки B1 слід «зафіксувати». У програмі Excel це роблять за допомогою символу \$. Його можна розташувати перед назвою стовпця і перед номером рядка. Залежно від того, яку частину адреси зафіксовано та чи зафіксовано її взагалі, розрізняють *відносні*, *абсолютні* та *змішані* посилання на адреси клітинок (табл. 7.1).

Таблиця 7.1. Типи посилань

| Адреса | Змінення адреси після її копіювання                   | Тип посилання |
|--------|-------------------------------------------------------|---------------|
| A1     | Змінюється і назва стовпця, і номер рядка             | Відносне      |
| \$A\$1 | Адреса не змінюється                                  | Абсолютне     |
| \$A1   | Назва стовпця не змінюється, а номер рядка змінюється | Змішане       |
| A\$1   | Назва стовпця змінюється, а номер рядка — ні          | Змішане       |

|   | A                   | B          | C          |
|---|---------------------|------------|------------|
| 1 | Курс гривні/холінги | 10         |            |
| 2 |                     |            |            |
| 3 | Найменування        | Ціна, грн. | Ціна, хол. |
| 4 | Пенал               | 100        | =B4/\$B\$1 |
| 5 | Рюкзак              | 350        | =B5/\$B\$1 |
| 6 | Ручка               | 20         | =B6/\$B\$1 |

Рис. 7.5

Але, якщо ми використаємо дію автозаповнення та скопіюємо її вміст до клітинок C5 і C6, у цих клітинках виникнуть помилки (рис. 7.2).

З'ясуємо, в чому річ, і поглянемо, які формули було одержано в цих клітинках у результаті копіювання формули з клітинки C4 (рис. 7.3)

Отже, ми бачимо, що в клітинках C5 і C6 ділення відбувається не на значення клітинки B1, у якій розташовано курс валют, а на клітинки розташовані нижче. Звичайно, ви отримаєте правильний результат, якщо зміните ці формули, щоб ділення виконувалося правильно (рис. 7.4).

А якщо таких формул потрібно ввести не три, а двадцять, або навіть сто і більше? Що тоді робити? Як не припуститися помилки і скоротити час на введення формул?

Якщо уважно поглянути на останні, вже правильно введені формули, то всі вони містять

Отже, для розглянутого вище завдання, формула в клітинці C4 повинна мати вигляд: =B4/B\$1 (якщо її копіювати лише в межах зазначеного в завданні стовпчика) або =B4/\$B\$1 (якщо копіювати при потребі в будь-яку клітинку таблиці). Застосуємо дію автозаповне-





ння до даної формули. Після цього діапазон клітинок **C4:C6** буде містити такі формули, як на рис. 7.5.

Зазначимо, що під час введення та редагування формул ви можете швидко змінювати тип посилання на адреси, щоразу натискаючи клавішу **F4**.

Покажемо це на прикладі адреси клітинки **A1**:  $A1 \xrightarrow{F4} \$A\$1 \xrightarrow{F4} A\$1 \xrightarrow{F4} \$A1 \xrightarrow{F4} A1$

## ВПРАВА 64



**Завдання.** Створити таблицю за зразком (рис. 7.6). Ввести формули для виконання необхідних розрахунків.

|   | A        | B                                        | C   | D                        | E   | F                                    | G   |
|---|----------|------------------------------------------|-----|--------------------------|-----|--------------------------------------|-----|
|   |          | Площа території,<br>тис. км <sup>2</sup> | У % | Чисельність,<br>тис. ос. | У % | Густота,<br>тис. ос./км <sup>2</sup> | У % |
| 1 | Країна   |                                          |     |                          |     |                                      |     |
| 2 | Ірландія | 70,3                                     |     | 4609                     |     | 65,3                                 |     |
| 3 | Іспанія  | 504,9                                    |     | 45200                    |     | 89,57                                |     |
| 4 | Італія   | 301,2                                    |     | 61261                    |     | 201,2                                |     |
| 5 | Всього   |                                          |     |                          |     |                                      |     |

Рис. 7.6.

1. Створіть таблицю та виконайте потрібне форматування.
2. Заповніть формулами клітинки виділені жовтим кольором.
3. Заповніть формулами клітинки виділені зеленим кольором. До клітинки **C2** введіть формулу: **=B2/\$B\$5**. Для клітинок **E2** і **G2** визначте формули самостійно.
4. Скопіюйте формули із зелених клітинок, застосувавши автозаповнення.
5. Встановіть у діапазонах клітинок **C2:C4**, **E2:E4**, **G2:G4** відсотковий формат.
6. Збережіть файл у власній папці під назвою **Вправа\_64\_Прізвище**.

## ВПРАВА 65



**Завдання.** Виконати обчислення в даній таблиці (рис. 7.7).

1. Створіть таблицю за зразком (рис. 7.7) або завантажте файл **Вправа\_65** з папки **Files** на диску та виконайте потрібне форматування.
2. До клітинок **D13** і **D14** введіть поточні курси валют.

|    | A  | B                 | C          | D         | E         | F        | G          | H         | I        |
|----|----|-------------------|------------|-----------|-----------|----------|------------|-----------|----------|
| 1  | №  | Найменування      | Ціна в грн | Кількість | Ціна в \$ | Ціна в € | Сума в грн | Сума в \$ | Сума в € |
| 2  | 1  | Пенал             | 100        | 1         |           |          |            |           |          |
| 3  | 2  | Портфель          | 400        | 1         |           |          |            |           |          |
| 4  | 3  | Ручка             | 7          | 5         |           |          |            |           |          |
| 5  | 4  | Олівець           | 6          | 12        |           |          |            |           |          |
| 6  | 5  | Обкладинка        | 43         | 20        |           |          |            |           |          |
| 7  | 6  | Щоденник          | 55         | 1         |           |          |            |           |          |
| 8  | 7  | Зошит             | 12         | 40        |           |          |            |           |          |
| 9  | 8  | Папка для зошитів | 29         | 1         |           |          |            |           |          |
| 10 | 9  | Ластик            | 5          | 2         |           |          |            |           |          |
| 11 | 10 | Лінійка           | 15         | 2         |           |          |            |           |          |
| 12 |    |                   |            |           |           |          |            |           |          |
| 13 |    | Курси валют:      | \$         |           |           |          |            |           |          |
| 14 |    |                   | €          |           |           |          |            |           |          |

Рис. 7.7

3. Заповніть формулами клітинки виділені зеленим кольором.
4. Скопіюйте формули з зелених клітинок, застосувавши автозаповнення.
5. Встановіть грошовий формат відповідно до зазначеної валюти.
6. Збережіть файл у власній папці під назвою **Вправа\_65\_Прізвище**.



### Висновки

Залежно від того, яку частину адреси зафіксовано та чи зафіксовано її взагалі, розрізняють *відносні*, *абсолютні* та *змішані* посилання на адреси клітинок.

При копіюванні формули з *відносним посиланням* змінюється назва стовпця і номер рядка. При копіюванні формули з *абсолютним посиланням* назва стовпця і номер рядка не змінюються. При копіюванні формули із *змішаним посиланням* змінюється або назва стовпця, або номер рядка.

Для фіксування елементів адреси клітинки використовують знак \$, який встановлюють перед тим елементом, який потрібно зафіксувати.



### Контрольні запитання та завдання

1. Яке призначення табличного процесора?
2. Що таке адреса клітинки?
3. Які дані може містити клітинка?
4. Які існують види посилань на адреси клітинок?
5. Як виконати перехід від одного виду посилання до іншого при введенні формули?



### Питання для роздумів

- 1\*. У яких випадках можна використовувати відносні посилання, а коли абсолютні чи змішані? Наведіть приклади.



### Завдання для досліджень

- 1\*. Дослідіть, для чого використовують у MS Excel стиль посилань R1C1.

## §47

### Імена клітинок та їх діапазонів. Розв'язування задач на обчислення

Якщо клітинка або діапазон клітинок часто використовуються у формулах, зручно їх називати. Перед тим, як ім'я можна буде використовувати, його спочатку потрібно присвоїти клітинці або діапазону клітинок.



Ім'я може починатися з літери, знака підкреслення або знака «\» і містити, крім літер, цифри, питання і крапки. Довжина імені має бути не більше 255 символів і не може містити пробілів. За умовчанням імена клітинок або діапазонів клітинок вважаються *абсолютними посиланнями*.

Щоб присвоїти ім'я клітинці чи діапазону клітинок, потрібно виділити клітинку чи діапазон клітинок, у вкладці **Формули** обрати команду **Визначити ім'я**, у результаті чого з'явиться діалогове вікно **Нове ім'я** (рис. 7.8).

У цьому вікні необхідно:

- ввести ім'я клітинки;
- обрати область застосування (уся книга або лише даний аркуш);
- за потреби ввести певне пояснення у полі **Примітка**;
- змінити або залишити незмінним посилання на клітинку або діапазон, для яких встановлюється ім'я.

Значно швидшим є спосіб введення нового імені для виділеного діапазону клітинок до поля **Ім'я**, розташованого ліворуч від рядка формул (рис. 7.9).

Проте в цьому випадку всі значення імені діапазону клітинок створюються з параметрами за умовчанням і будуть діяти в межах усієї книги.

У цьому ж полі можна переглянути всі імена клітинок, які є, і завдяки вибору імені здійснити перехід до нього. Також відображення імен з їхніми значеннями та властивостями можна відкрити в діалоговому вікні **Диспетчер імен**, обравши у вкладці **Формула** інструмент **Диспетчер імен** (рис. 7.10).

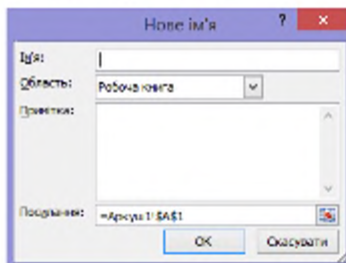


Рис. 7.8

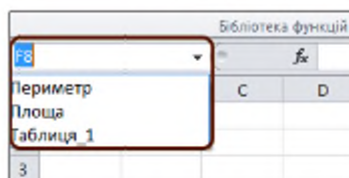


Рис. 7.9

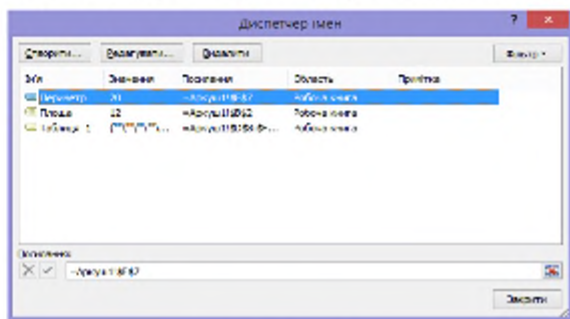


Рис. 7.10

|   | A                  | B          | C          |
|---|--------------------|------------|------------|
| 1 | Курс гривні/холінг | 10         |            |
| 2 |                    |            |            |
| 3 | Найменування       | Ціна, грн. | Ціна, хол. |
| 4 | Попол              | 100        | =B4/холінг |
| 5 | Рюкзак             | 350        | =B5/холінг |
| 6 | Ручка              | 20         | =B6/холінг |

Рис. 7.11

У цьому вікні відображено перелік імен, у якому зазначено ім'я, значення, посилання, області застосування та примітки. Засобами цього вікна можна також створити нове ім'я, змінити або видалити те, що є.

Так, якщо в завданні, розглянутому в попередньому параграфі щодо визначення ціни товарів у холінгах, для клітинки **B1** встановити певне ім'я (наприклад, **холінг**), то у формулах не потрібно буде фіксувати адреси. Замість адреси клітинки у формулах буде вказано її ім'я (рис. 7.11).

## ВПРАВА 66



**Завдання.** Виконати в таблиці необхідні розрахунки, надавши імена клітинкам E3, E4, ..., E8.

|    | A               | B                                      | C                               | D                     | E      | F | G                   | H                      | I        | J                   | K                      |                     |            |                   |
|----|-----------------|----------------------------------------|---------------------------------|-----------------------|--------|---|---------------------|------------------------|----------|---------------------|------------------------|---------------------|------------|-------------------|
| 1  |                 |                                        |                                 |                       |        |   |                     |                        |          |                     |                        |                     |            |                   |
| 2  |                 | Ціни на комунальні послуги             |                                 |                       |        |   |                     |                        |          |                     |                        |                     |            |                   |
| 3  |                 | Опалення, 1 м <sup>2</sup>             |                                 |                       | 22,658 |   |                     |                        |          |                     |                        |                     |            |                   |
| 4  |                 | Водопостачання, 1 м <sup>3</sup>       |                                 |                       | 8,368  |   |                     |                        |          |                     |                        |                     |            |                   |
| 5  |                 | Електропостачання до 100 пот. 1 кВт    |                                 |                       | 0,428  |   |                     |                        |          |                     |                        |                     |            |                   |
| 6  |                 | Електропостачання понад 100 пот. 1 кВт |                                 |                       | 0,728  |   |                     |                        |          |                     |                        |                     |            |                   |
| 7  |                 | Газопостачання, на 1 особу             |                                 |                       | 32,858 |   |                     |                        |          |                     |                        |                     |            |                   |
| 8  |                 | Квартплата, 1 м <sup>2</sup>           |                                 |                       | 13,508 |   |                     |                        |          |                     |                        |                     |            |                   |
| 9  |                 |                                        |                                 |                       |        |   |                     |                        |          |                     |                        |                     |            |                   |
| 10 |                 |                                        |                                 |                       |        |   | Вартість за місяць  |                        |          |                     |                        |                     |            |                   |
|    | № квар-<br>тири | Площа<br>квартири,<br>м <sup>2</sup>   | Кількість<br>прописаних<br>осіб | Показники лічильників |        |   | Водо-<br>постачання | Електро-<br>постачання | Опалення | Водо-<br>постачання | Електро-<br>постачання | Газо-<br>постачання | Квартплата | Сума до<br>сплати |
| 11 |                 |                                        |                                 |                       |        |   |                     |                        |          |                     |                        |                     |            |                   |
| 12 | 1               | 61                                     | 2                               | 3,5                   | 155    |   |                     |                        |          |                     |                        |                     |            |                   |
| 13 | 2               | 48                                     | 4                               | 2,8                   | 320    |   |                     |                        |          |                     |                        |                     |            |                   |
| 14 | 3               | 33                                     | 1                               | 0,9                   | 85     |   |                     |                        |          |                     |                        |                     |            |                   |
| 15 | 4               | 61                                     | 3                               | 4,8                   | 206    |   |                     |                        |          |                     |                        |                     |            |                   |
| 16 | 5               | 61                                     | 4                               | 5,2                   | 333    |   |                     |                        |          |                     |                        |                     |            |                   |
| 17 | 6               | 48                                     | 2                               | 1,8                   | 95     |   |                     |                        |          |                     |                        |                     |            |                   |
| 18 | 7               | 33                                     | 3                               | 2,8                   | 222    |   |                     |                        |          |                     |                        |                     |            |                   |
| 19 | 8               | 61                                     | 7                               | 10,5                  | 580    |   |                     |                        |          |                     |                        |                     |            |                   |
| 20 |                 |                                        |                                 |                       |        |   |                     |                        |          |                     |                        |                     |            | Усього            |

Рис. 7.12



**Виконати** вправу  
«Дерева» ви  
зможете на диску.

- Створіть таблицю за зразком (рис. 7.12) або завантажте файл **Вправа\_66** з папки **Files** на диску відформатуйте її.
- Надайте імена клітинкам E3, E4, ..., E8.
  - Виділіть клітинку E3.
  - У вкладці **Формули** оберіть команду **Визначити ім'я**.
    - У діалоговому вікні **Нове ім'я** введіть ім'я **Опалення** та натисніть **ОК**.
    - Аналогічно надайте імена іншим клітинкам.
- Обчисліть вартість опалення в кожній квартирі за місяць.
  - До клітинки F12 введіть формулу для розрахунку вартості опалення за місяць: (ціна за 1 м<sup>2</sup>) × (площа квартири). Для цього:
    - виділіть клітинку F12;
    - введіть знак «=» і клацніть на клітинку E3, натисніть знак «\*» і клацніть на клітинку B12;
    - натисніть **Enter**. У результаті формула буде мати вигляд:  
**=Опалення\*B12**.
  - Виконайте копіювання формули до клітинок знизу, використовуючи дію автозаповнення.
- До інших клітинок формули введіть самостійно.
- Збережіть файл під назвою **Вправа\_80\_Прізвище**.

## ВПРАВА 67



**Завдання.** Створити таблицю за зразком (рис. 7.13). Обчислити значення у за формулою:  $y = \frac{P + 2x}{Q} - \frac{x^2}{(T + 1)}$ , де  $Q = 3P$ ,  $T = 2P^2$ .



1. Створіть таблицю за зразком.
2. Обчисліть значення змінних Q, T, увівши відповідні формули до клітинок B2 і C2.
3. Надайте імена клітинкам P, Q і T, що містять значення змінних P, Q, T.
4. До клітинки B5 введіть формулу для знаходження значення змінної у:  

$$=(P+2*A5)/Q-A5^2/(R+1)$$
5. Застосуйте до попередньої формули дію автозаповнення, виконавши її копіювання в діапазон B5:B14.
6. Збережіть файл під назвою Вправа\_67\_Прізвище.

|    | A   | B | C |
|----|-----|---|---|
| 1  | P   | Q | T |
| 2  | 0,1 |   |   |
| 3  |     |   |   |
| 4  | x   | y |   |
| 5  | 1   |   |   |
| 6  | 3   |   |   |
| 7  | 5   |   |   |
| 8  | 7   |   |   |
| 9  | 9   |   |   |
| 10 | 11  |   |   |
| 11 | 13  |   |   |
| 12 | 15  |   |   |
| 13 | 17  |   |   |
| 14 | 19  |   |   |

Рис. 7.13



### Висновки

Якщо клітинка або діапазон клітинок часто використовуються у формулах, зручно їх називати.

За умовчанням імена клітинок або діапазонів клітинок вважаються абсолютними посиланнями.

Для надання імені потрібно виділити клітинку або діапазон клітинок і у вкладці **Формули** обрати команду **Визначити ім'я**.

За допомогою діалогового вікна **Диспетчер імен** можна додавати, редагувати та видаляти імена.



### Контрольні запитання та завдання

1. Що таке ім'я діапазону клітинок?
2. Для чого клітинці надають ім'я?
3. Як створити ім'я діапазону клітинок?
4. Як змінити ім'я клітинки?
5. Де можна переглянути імена клітинок, які вже названі?



### Питання для роздумів

1. Як задати ім'я клітинці тільки для даного аркуша?
- 2\*. Чи можна для створення нового імені діапазону використовувати клітинки, яким уже надано імена?
- 3\*. Чи може та сама клітинка мати декілька імен?



### Завдання для досліджень

1. Визначте, як надати ім'я клітинці за допомогою її контекстного меню.
- 2\*. Визначте, як виділити одночасно декілька діапазонів за допомогою їхніх імен.

## ПРАКТИЧНА РОБОТА №14

## Розв'язування задач на обчислення

|   | A                     | B      | C         | D        | E      | F      | G        | H      | I      | J      | K               |
|---|-----------------------|--------|-----------|----------|--------|--------|----------|--------|--------|--------|-----------------|
| 1 | Найменування морозива | Ціна   | Понеділок | Вівторок | Середа | Четвер | П'ятниця | Субота | Неділя | Всього | Сума за тиждень |
| 2 | Каштан                | 12,50€ | 3         | 4        | 10     | 6      | 3        | 6      | 6      |        |                 |
| 3 | Імперія               | 13,75€ | 5         | 2        | 11     | 5      | 8        | 5      | 2      |        |                 |
| 4 | Ласка                 | 11,80€ | 7         | 6        | 7      | 1      | 9        | 7      | 3      |        |                 |
| 5 | Всього морозива       |        |           |          |        |        |          |        |        |        |                 |
| 6 | Всього сума           |        |           |          |        |        |          |        |        |        |                 |

Рис. 7.14

- У клітинці **K6** введіть формулу для обчислення загального прибутку.
- Надайте аркушу ім'я **Завдання1** та збережіть файл під назвою **ПР14\_Прізвище**.

|    | A | B | C | D | E | F | G | H | I | J |
|----|---|---|---|---|---|---|---|---|---|---|
| 1  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |   |
| 2  | 1 |   |   |   |   |   |   |   |   |   |
| 3  | 2 |   |   |   |   |   |   |   |   |   |
| 4  | 3 |   |   |   |   |   |   |   |   |   |
| 5  | 4 |   |   |   |   |   |   |   |   |   |
| 6  | 5 |   |   |   |   |   |   |   |   |   |
| 7  | 6 |   |   |   |   |   |   |   |   |   |
| 8  | 7 |   |   |   |   |   |   |   |   |   |
| 9  | 8 |   |   |   |   |   |   |   |   |   |
| 10 | 9 |   |   |   |   |   |   |   |   |   |

Рис. 7.15

**Завдання 2.** Створити таблицю множення чисел від 1 до 9 на другому аркуші.


- Створіть таблицю за зразком (рис. 7.15).
- Уведіть до клітинки **B2** формулу **=B\$1\*\$A2**, що перемножує число в рядку на число у стовпці, які розташовані ліворуч та зверху над даною клітинкою.

**Пояснення.** Записуючи формулу, було враховано, що при копіюванні її праворуч має незмінним залишатися стовпець **A**, а при копіюванні вниз сталим є перший рядок.

- Виконайте копіювання формули з клітинки **B2** до діапазону **C2:J2**, застосувавши автозаповнення.
- Виділіть діапазон **B2:J2** та застосуйте до нього дію автозаповнення до діапазону **B3:J10**. У результаті ви маєте отримати таблицю множення.
- Надайте аркушу ім'я **Завдання2** та збережіть файл.
- На третьому аркуші реалізуйте таку ж таблицю множення із створенням імен діапазонів.
- Надайте аркушу ім'я **Завдання2\*** та збережіть файл.

**Завдання 1.** Створити таблицю на першому аркуші за зразком (рис. 7.14) та виконати обчислення.

- Створіть таблицю та виконайте необхідне форматування.

**Примітка.** Щоб змінити напрямок тексту, слід обрати інструмент  **Орієнтація** в групі **Вирівнювання** на вкладці **Основне**.

- Введіть необхідні формули до клітинок **J2**, **K2**, **C5** і **C6**. Скопіюйте їх до відповідних діапазонів, використовуючи автозаповнення.

**Функції** — це заздалегідь створені формули, які виконують обчислення за введеними величинами в зазначеному порядку.



У сьомому класі ви вже ознайомилися з деякими найбільш уживаними функціями такими, як **SUM**, **MAX**, **MIN**, **AVERAGE**, **COUNT**, що призначені для знаходження суми, максимального і мінімального значень, середнього арифметичного та знаходження кількості числових значень.

Нагадаємо основні елементи, що має містити функція:

- власне ім'я, наприклад **SUM**;
- аргументи, що відокремлені крапкою з комою та обмежені дужками.

*Аргументом функції* може бути число, текст, посилання на клітинку або діапазон клітинок, формула, інша функція тощо.

Кожна функція має свій синтаксис і порядок дій, якого слід дотримуватися, оскільки в протилежному випадку результати обчислень будуть хибними. Використовуючи функції, також треба дотримуватися типу аргументів.

MS Excel містить математичні, статистичні, логічні та інші функції.

#### 48.1. МАТЕМАТИЧНІ ФУНКЦІЇ

Функції, які найчастіше використовуються на практиці, наведено в таблиці 7.2. Оскільки значне коло користувачів використовує російський інтерфейс програми MS Excel, то в другому стовпчику таблиці подано позначення функцій для даної версії табличного процесора.

Таблиця 7.2. Перелік основних математичних функцій

| Функція                              | Функція в MS Excel з російським інтерфейсом | Призначення                                             |
|--------------------------------------|---------------------------------------------|---------------------------------------------------------|
| ABS(число)                           | ABS                                         | модуль числа                                            |
| COS(число)                           | COS                                         | косинус кута (аргумент має бути в радіанах)             |
| INT(число)                           | ЦЕЛОЕ                                       | округлення числа до найменшого цілого                   |
| MOD(число; дільник)                  | ОСТАТ                                       | остача від ділення                                      |
| PRODUCT(число1; число2;...)          | ПРОИЗВЕД                                    | добуток чисел                                           |
| QUOTIENT(чисельник; знаменник)       | ЧАСТНОЕ                                     | ціла частина від ділення                                |
| RADIANS(кут)                         | РАДИАНЫ                                     | перетворення градусів на радіани                        |
| ROUND(число; кільк_розрядів)         | ОКРУГЛ                                      | округлення числа до заданої кількості десяткових знаків |
| SIN(число)                           | SIN                                         | синус кута (аргумент має бути в радіанах)               |
| SUMPRODUCT(діапазон1; діапазон2;...) | СУММПРОИЗВ                                  | визначення суми добутків відповідних діапазонів         |
| SQRT(число)                          | КОРЕНЬ                                      | квадратний корінь числа                                 |
| TAN(число)                           | TAN                                         | тангенс кута (аргумент має бути в радіанах)             |

## 48.2.

## СТАТИСТИЧНІ ФУНКЦІЇ

**Історичні відомості**

про різні версії  
MS Excel ви зможете  
дізнатися на диску.

Для здійснення статистичного аналізу даних у MS Excel використовують функції з категорії **Статистичні**. Більшість у цій категорії функцій використовує вузьке коло спеціалістів, проте частина з них є досить часто уживаними. Так, до статистичних функцій відносяться вже відомі вам **MAX**, **MIN**, **AVERAGE** та **COUNT**.

У таблиці 7.3 наведено ще деякі функції, які використовують при обчисленнях з категорії **Статистичні**.

Таблиця 7.3. Перелік деяких статистичних функцій

| Функція                          | Функція в MS Excel з російським інтерфейсом | Призначення                                                                   |
|----------------------------------|---------------------------------------------|-------------------------------------------------------------------------------|
| COUNTA(значення1; значення2;...) | СЧЁТЗ                                       | визначення кількості непорожніх клітинок                                      |
| COUNTIF(діапазон; критерій)      | СЧЁТЕСЛИ                                    | визначення кількості непорожніх клітинок, які задовольняють вказаний критерій |
| RANK(число; посилання; порядок)  | РАНГ                                        | визначення рангу числа в переліку чисел                                       |

*Ранг числа* — це його величина відносно інших значень у списку. Якщо посортувати список, то ранг буде його позицією. Цю функцію часто використовують для визначення рейтингу навчання учнів, виступу спортсменів на змаганнях тощо.

Аргументи функції **RANK**:

- *число* — число, для якого визначається ранг;
- *посилання* — діапазон чисел, серед яких визначається ранг;
- *порядок* — визначення способу впорядкування даних. Якщо порядок дорівнює нулю або його не вказано, то число, що є найбільшим значенням діапазону посилання, отримає значення рангу 1. Якщо порядок дорівнює одиниці, то значення рангу 1 отримає число, що є найменшим значенням діапазону посилання.


Починаючи з 2010 версії MS Excel, ця функція була замінена новими функціями: **RANK.AVG** (рос. **РАНГ.СР**) і **RANK.EQ** (рос. **РАНГ.РВ**), які забезпечують більшу точність.

## 48.3.

## ВИКОРИСТАННЯ МАЙСТРА ФУНКЦІЙ

Функції можна вставляти як на початку обчислень, так і під час створення власної формули. Це можна зробити, ввівши формулу вручну в рядок формул або за допомогою майстра функцій.

Потрібну функцію можна вибрати в діалоговому вікні **Вставлення функції**, яке викликається декількома способами:

- натиснути кнопку , що розташована ліворуч від рядка формул;





- у меню інструмента  $\Sigma$  - Сума обрати команду **Інші функції**;
- на вкладці **Формули** обрати команду **Вставити функцію**;
- натиснути один з інструментів у групі **Бібліотека функцій** на вкладці **Формули** (рис. 7.16).



**Дізнатися більше**  
про функції категорій  
**Дата та час** та **Текстова** ви  
зможете на диску.

Після виклику діалогового вікна користувач може обрати в полі **Категорія** одну з категорій функцій: **Нещодавно використані** (містить останні 10 функцій); **Усі**; **Математичні**; **Статистичні**; **Логічні** тощо. На рис. 7.17 представлено вікно, в якому вибрано категорію **Математичні**.

Наступним кроком користувача є вибір функції. Далі з'являється вікно **Аргументи функції**, призначене для формування обраної функції. В даному вікні необхідно ввести всі обов'язкові аргументи та натиснути **OK**. На рис. 7.18 представлено вікно функції **ROUND** (рос. **ОКРУГЛ**).

Якщо у вікні було правильно записано вказані аргументи, то в клітинці, до якої було введено формулу з даною функцією, буде одержано результат обчислень.

Нагадаємо, що вікно **Аргументи функції** має подібний вигляд для різних функцій, але з різною кількістю аргументів.

Щоб відредагувати формули, що містять вбудовані функції, у рядку формул потрібно клацнути на назві функції, а потім на значку  $f_x$  (ліворуч від рядка формул), у результаті буде виведено вікно **Аргументи функції** для обраної функції.

#### 48.4.

#### ВВЕДЕННЯ ФУНКЦІЙ З КЛАВІАТУРИ

Якщо вводити в рядок формул перші літери функції, відкривається список функцій для автозавершення (рис. 7.19).

Для швидкого введення функції необхідно клацнути на ній два рази. В результаті буде вставлено бажану функцію

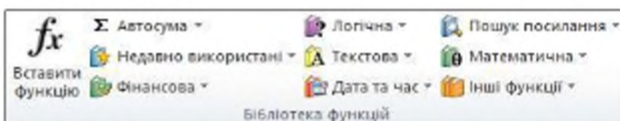


Рис. 7.16

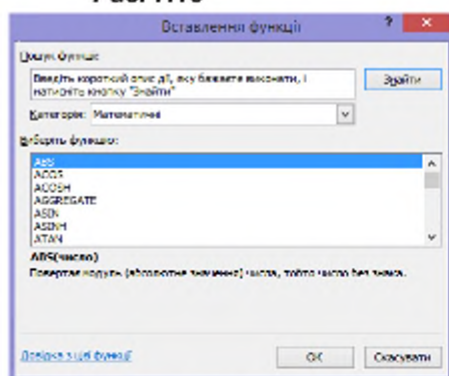


Рис. 7.17

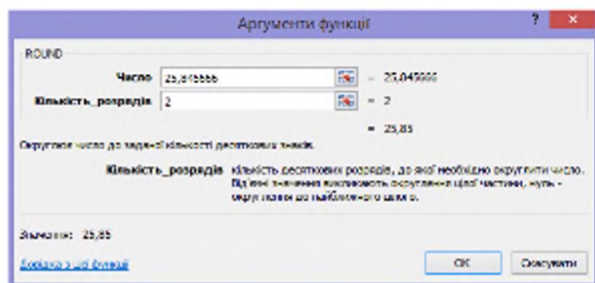


Рис. 7.18

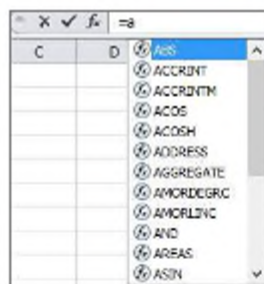


Рис. 7.19

|   |   |                |            |
|---|---|----------------|------------|
| x | ✓ | f <sub>x</sub> | =ABS(      |
| C | D |                | ABS(число) |

Рис. 7.20

та відкрито дужку для введення аргументів. Також з'явиться підказка щодо переліку аргументів, які має містити функція (рис. 7.20).

Після введення аргументів потрібно закрити дужку. Якщо формулу завершено, то клавішу **Enter**, якщо ні, то ввести інші знаки для продовження формули. Нагадаємо, що для відокремлення аргументів використовують «;».

## ВПРАВА 68



**Завдання.** У клітинках **A2**, **B2**, **C2** дано числові значення. Знайти значення інших змінних за вказаними нижче формулами.

|   | A  | B  | C  |
|---|----|----|----|
| 1 | a  | b  | c  |
| 2 | 12 | 13 | 14 |
| 3 |    |    |    |
| 4 | X  |    |    |
| 5 | P  |    |    |
| 6 | S  |    |    |
| 7 | M  |    |    |
| 8 | Q  |    |    |
| 9 | T  |    |    |

Рис. 7.21

$$X = \sqrt{\frac{a}{b}} + |b - c|$$

$$M = \frac{1}{2} \sqrt{2b^2 + 2c^2 - a^2}$$

$$P = \frac{a+b+c}{2}$$

$$Q = P \cdot S + |b - c|$$

$$S = \sqrt{P(P-a)(P-b)(P-c)} \quad T = \sqrt{|a+b-2|} - \frac{a-c}{\sqrt{b}}$$

- Створіть таблицю за зразком (рис. 7.21).
- Введіть потрібну формулу до клітинки **B4** для обчислення значення виразу:  $X = \sqrt{\frac{a}{b}} + |b - c|$ .
  - Виділіть клітинку **B4**.
  - Ліворуч від рядка формул оберіть значок **f<sub>x</sub>** для виклику майстра функцій.
  - У діалоговому вікні **Вставка функції** оберіть категорію **Математичні**.
  - Оберіть функцію **SQRT** (рос. **КОРЕНЬ**) та натисніть **ОК**.
  - У діалоговому вікні **Аргументи функції** у полі **Число** введіть аргумент **A2/B2**.
  - У рядку формул встановіть курсор після останньої дужки.
  - Поставте знак «+».
  - Клацніть на значку **f<sub>x</sub>** два рази для введення наступної функції.
  - Знову оберіть категорію **Математичні**, виберіть функцію **ABS** і натисніть **ОК**.
  - Введіть аргумент даної функції **B2-C2**. Завершіть введення формули, натиснувши **ОК**.
- Формули для обчислення значень у клітинках **B5:B9** введіть самостійно.
- Збережіть файл під назвою **Вправа\_68\_Прізвище**.

## ВПРАВА 69



**Завдання.** Виконати переведення числа від 0 до 1023 з десяткової системи числення в двійкову.



- Створіть таблицю за зразком (рис. 7.22).
- У клітинці **C2** введіть формулу для знаходження остачі від ділення числа, що знаходиться в клітинці **B2**, на 2.
  - Виділіть клітинку **C2** і викличте майстра функцій.
  - У категорії **Математичні** оберіть функцію **MOD** (рос. **ОСТАТ**).
  - У діалоговому вікні **Аргументи функції** у полі **Число** оберіть клітинку **B2**, у полі **Дільник** — 2.
  - Натисніть **ОК**. У результаті в рядку формул ви маєте одержати формулу **=MOD(B2;2)**.

|    | A      | B   | C |
|----|--------|-----|---|
| 1  | Розряд |     |   |
| 2  | 0      | 125 |   |
| 3  | 1      |     |   |
| 4  | 2      |     |   |
| 5  | 3      |     |   |
| 6  | 4      |     |   |
| 7  | 5      |     |   |
| 8  | 6      |     |   |
| 9  | 7      |     |   |
| 10 | 8      |     |   |
| 11 | 9      |     |   |

Рис. 7.22

- У клітинці **B3** введіть формулу для знаходження цілої частини від ділення попереднього числа, що знаходиться в клітинці **B2**, на 2. Виконайте дії аналогічно до описаних у попередньому пункті, застосувавши функцію **QUOTIENT** (рос. **ЧАСТНОЕ**). В результаті в клітинці **B3** ви маєте одержати формулу: **=QUOTIENT(B2;2)**.
- Застосуйте до введених формул дію автозаповнення, протягнувши їх до діапазонів **B4:B11** і **C3:C11**. У результаті в клітинках діапазону **C2:C11** буде одержано число записане в двійковій системі числення, якщо переглядати діапазон у зворотному порядку.

**Примітка.** Для з'єднання значень діапазону до однієї клітинки можна скористатися функцією **CONCATENATE** (рос. **СЦЕПИТЬ**). Для з'єднання клітинок діапазону **C2:C11** у зворотному порядку формула буде мати вигляд: **=CONCATENATE(C11;C10;C9;C8;C7;C6;C5;C4;C3;C2)**.

- Збережіть файл під назвою **Вправа\_69\_Прізвище**.

## ВПРАВА 70



**Завдання.** Відомо результати середнього балу учнів. Визначити рейтинг успішності учнів.

- Створіть таблицю за зразком (рис. 7.23).
- До клітинки **D2** введіть формулу для обчислення рейтингу першого учня серед інших учнів.
  - У вікні майстра функцій в категорії **Статистичні** оберіть функцію **RANK.EQ** (рос. **РАНГ.РВ**) або **RANK** (рос. **РАНГ**).

|    | A  | B           | C            | D                  |
|----|----|-------------|--------------|--------------------|
| 1  | №  | Прізвище    | Середній бал | Рейтинг успішності |
| 2  | 1  | Бондаренко  | 3,9          |                    |
| 3  | 2  | Іванов      | 6,8          |                    |
| 4  | 3  | Ігнатів     | 10,3         |                    |
| 5  | 4  | Карпенко    | 10,1         |                    |
| 6  | 5  | Козленко    | 9,6          |                    |
| 7  | 6  | Крохмаленко | 8,4          |                    |
| 8  | 7  | Кулеш       | 6,8          |                    |
| 9  | 8  | Новіков     | 7,2          |                    |
| 10 | 9  | Слісарчук   | 7,3          |                    |
| 11 | 10 | Шевцов      | 7,9          |                    |

Рис. 7.23

- У вікні діалогового вікна в полі **Число** оберіть клітинку **C2**, а в полі **Посилання** — діапазон **\$C\$2:\$C\$11**, у полі **Порядок** — число 0.

**Пояснення.** Посилання на діапазон є абсолютним, оскільки при копіюванні формули діапазон має залишитися незмінним. Порядок набуває значення 0, оскільки найбільше значення діапазону буде мати 1 місце в рейтингу.

У результаті в рядку формул ви маєте одержати формулу: **=RANK(C2;\$C\$2:\$C\$11;0)**.



**Виконати** вправу «Вартість товару» ви зможете на диску.

3. Виконайте копіювання даної формули до клітинок D3:D11, використовуючи дію автозаповнення.
4. Збережіть файл під назвою **Вправа\_70\_Прізвище**.



### Висновки

**Функції** — це заздалегідь створені формули, які виконують обчислення за введеними величинами в зазначеному порядку.

Щоб виконати математичні обчислення, використовують функції з категорії **Математичні**, а щоб проаналізувати статистичні дані, — функції категорії **Статистичні**.

Функції можна вводити вручну в рядок формул або за допомогою майстра функцій.



### Контрольні запитання та завдання

1. Що таке функція в MS Excel?
2. Які ви знаєте математичні функції? статистичні функції?
3. Яке призначення функції **RANK**?
4. Вкажіть способи виклику майстра функцій.
5. Вкажіть послідовність дій для додавання функції за допомогою майстра.
6. Як увести формулу вручну?



### Питання для роздумів

- 1\*. Чи можна в одній формулі ввести одну функцію за допомогою майстра, а іншу — вручну?
- 2\*. Навіщо в Excel використовують декілька способів виклику майстра функцій?



### Завдання для досліджень

- 1\*. Дослідіть, скільки вкладень може мати одна функція.
- 2\*. Визначте, чим відрізняються функції **RANK.AVG** і **RANK.EQ**.

## §49

## Логічні функції

### 49.1.

### ВИКОРИСТАННЯ ЛОГІЧНИХ ФУНКЦІЙ

У MS Excel можна вирішувати завдання, виконання яких залежить від певної умови. Частину таких завдань можна реалізувати, використовуючи функції з категорії **Логічні**. Однією з основних функцій даної категорії є функція **IF** (рос. **ЕСЛИ**), яка має такий синтаксис:

**=IF(Лог\_вираз;Значення\_якщо\_істина;Значення\_якщо\_хибність).**



Блок-схема на рис. 7.24 демонструє дію логічної функції.



Рис. 7.24

Як і в програмуванні, в табличному процесорі можна використовувати складені логічні вирази, для яких використовують логічні зв'язки **AND** (рос. **И**), **OR** (рос. **ИЛИ**), **NOT** (рос. **НЕ**). У дужках даних функцій логічні вирази записують через крапку з комою. Для порівняння числових значень використовують знаки порівнянь (=, >, < тощо), які записують так само, як і в мові програмування.

## ВПРАВА 71



**Завдання.** Дано сторони  $a$  і  $b$ . Визначити за даними значеннями сторін тип фігури, а саме: квадрат чи прямокутник.

- Створіть таблицю за зразком (рис. 7.25).
- У клітинці **C2** введіть формулу для визначення типу фігури. Виконайте копіювання даної формули до діапазону **C2:C8**, використовуючи автозаповнення.
  - Зробіть активною клітинку **C2** і викличте майстра функцій.
  - Оберіть функцію **IF** (рос. **ЕСЛИ**) з категорії **Логічні**.
  - У діалоговому вікні в полі **Лог\_вираз** введіть умову **A2=B2**. У полі **Значення\_якщо\_істина** — «Квадрат», у полі **Значення\_якщо\_хибність** — «Прямокутник» (рис. 7.26).

|   | A | B | C          |
|---|---|---|------------|
| 1 | a | b | Тип фігури |
| 2 | 5 | 6 |            |
| 3 | 4 | 4 |            |
| 4 | 7 | 8 |            |
| 5 | 2 | 3 |            |
| 6 | 3 | 4 |            |
| 7 | 6 | 6 |            |
| 8 | 3 | 8 |            |

Рис. 7.25

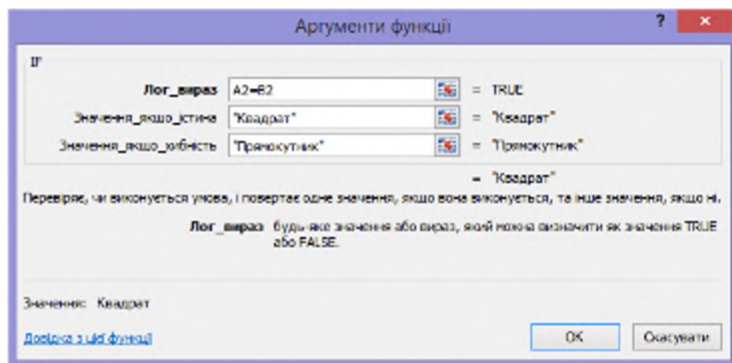


Рис. 7.26

г) Натисніть **ОК**. У результаті для клітинки **C2** у рядку формул має з'явитися формула **=IF(A2=B2;«Квадрат»;«Прямокутник»)**.

- Збережіть файл під назвою **Вправа\_71\_Прізвище**.

## ВПРАВА 72



**Завдання.** Відомо час, за який учні подолали дистанцію на 100 м. Визначте, яке місце посів кожний учень, і надайте відповідно до цього рекомендації: якщо 1, 2 або 3 місце — «Призер», інакше — «Тренуйся далі».

|   | A                                        | B    | C     | D            |
|---|------------------------------------------|------|-------|--------------|
| 1 | Таблиця змагань учнів на дистанцію 100 м |      |       |              |
| 2 | Прізвище                                 | Час  | Місце | Рекомендації |
| 3 | Іваненко                                 | 13,5 |       |              |
| 4 | Петренко                                 | 12,4 |       |              |
| 5 | Сидоренко                                | 13,8 |       |              |
| 6 | Гаврилюк                                 | 12,9 |       |              |
| 7 | Гук                                      | 13,1 |       |              |

Рис. 7.27

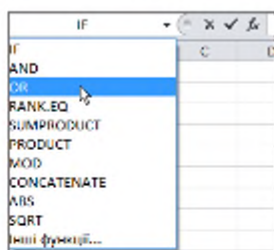


Рис. 7.28



**Виконати** вправу «Дріб» ви зможете на диску.

- Створіть таблицю за зразком (рис. 7.27).
  - У клітинці **C3** введіть формулу для визначення місця, яке посіли учні, скориставшись функцією **RANK** (рос. **РАНГ**). Скопіюйте формулу до клітинок нижче, використовуючи дію автозаповнення.
  - У клітинці **D3** введіть формулу для надання рекомендації. Виконайте копіювання даної формули до діапазону **D4:D7**, використовуючи автозаповнення.
    - Зробіть активною клітинку **D3** та викличте майстра функцій.
    - Оберіть функцію **IF** (рос. **ЕСЛИ**) з категорії **Логічні**.
    - У діалоговому вікні **Аргументи функції**, перебуваючи в полі **Лог\_вираз**, відкрийте список функцій, клацнувши на стрілку в полі, що розташована ліворуч від рядка формул (рис. 7.28).
    - У списку оберіть функцію **OR** (рос. **ИЛИ**) або команду **Інші функції** для подальшого вибору функції **OR**.
    - У полі **Лог\_значення1** введіть умову **C3=1**, у полі **Лог\_значення2** — **C3=2**, у полі **Лог\_значення3** — **C3=3**.
    - У рядку формул після введеної умови **OR(C3=1;C3=2;C3=3)** поставте крапку з комою. У результаті ви маєте повернутися до вікна функції **IF**.
    - У полі **Значення\_якщо\_Істина** введіть «Призер», у полі **Значення\_якщо\_хибність** — «Тренуйся далі».
    - Натисніть **ОК**. У результаті для клітинки **C2** у рядку формул має з'явитися формула: **=IF(OR(C3=1;C3=2;C3=3);«Призер»;«Тренуйся далі»)**.
4. Збережіть файл під назвою **Вправа\_72\_Прізвище**.

## 49.2.

## ВКЛАДЕНІ ФУНКЦІЇ

|   | A     | B     | C     | D            |
|---|-------|-------|-------|--------------|
| 1 | № гри | Квант | Алмаз | Хто переміг? |
| 2 | 1     | 3     | 4     |              |
| 3 | 2     | 0     | 0     |              |
| 4 | 3     | 2     | 1     |              |
| 5 | 4     | 5     | 1     |              |
| 6 | 5     | 1     | 3     |              |

Рис. 7.29

Щоб перевірити більш ніж одну умову, використовують вкладення функції **IF**.

Так, наприклад, відомо результати ігор між двома командами «Квант» і «Алмаз». Потрібно визначити для кожної гри, хто переміг, або вивести повідомлення про нічию (рис. 7.29).



Блок-схема розв'язання цього завдання буде мати вигляд, як на рис. 7.30:

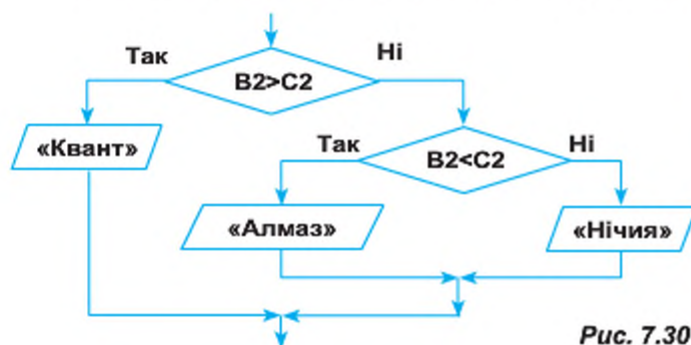


Рис. 7.30

**Дізнатися більше**  
про вкладені логічні  
функції ви зможете на  
диску.

**Перевірити свої**  
знання і пройти тест  
ви зможете на диску.

У MS Excel для цього використовують вкладення функції таким чином:  
=IF(B2>C2;«Квант»;IF(B2<C2;«Алмаз»;«Нічия»)).

Щоб ввести цю вкладену функцію, використовуючи майстра функцій, треба виконати таку послідовність дій.

1. Виділіть клітинку D2. Оберіть функцію IF з категорії Логічні.
2. У діалоговому вікні в полі **Лог\_вираз** введіть умову B2>C2, у полі **Значення\_якщо\_істина** — «Квант».
3. Перебуваючи в полі **Значення\_якщо\_хибність**, клацніть знову на функції IF у полі, яке розташоване ліворуч від рядка формул (рис. 7.31). Після чого має з'явитися вікно **Аргументи функції** для введення наступних значень.
4. У полі **Лог\_вираз** введіть умову B2<C2.
5. У полі **Значення\_якщо\_істина** — «Алмаз», а в полі **Значення\_якщо\_хибність** — «Нічия».
6. Натисніть ОК.

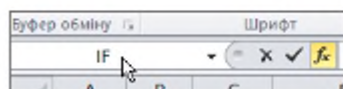


Рис. 7.31

## ВПРАВА 73



**Завдання.** Вивести інформацію про результати змагань між командами «Квант» і «Алмаз», керуючись даними та поясненнями поданими вище. Зберегти таблицю під назвою **Вправа\_73\_Прізвище**.



### Висновки

Для виконання завдань з врахуванням умов у MS Excel використовують функції з категорії **Логічні**.

Однією з основних функцій даної категорії є функція IF (рос. ЕСЛИ), яка має такий синтаксис:

=IF(Лог\_вираз;Значення\_якщо\_істина;Значення\_якщо\_хибність).

Для формування складних логічних виразів використовують функції AND (рос. И), OR (рос. ИЛИ), NOT (рос. НЕ).

Для порівняння числових значень використовують знаки порівнянь.



## Контрольні запитання та завдання

1. Для чого використовують функції з категорії Логічні?
2. Опишіть синтаксис функції IF.
3. Для чого використовують функції AND, OR, NOT?
4. Опишіть синтаксис функції AND.



## Питання для роздумів

- 1\*. У яких випадках використовують вкладені логічні функції?
- 2\*. Скільки логічних виразів може містити функція OR?



## Завдання для досліджень

- 1\*. Порівняйте синтаксис функції IF у MS Excel з цією ж функцією в VB.NET.
- 2\*. Яке призначення і синтаксис функції IFERROR (ЕСЛИОШИБКА)?

## §50

## Умовне форматування

При створенні та форматуванні документів, у яких велика кількість клітинок, можна застосувати таку можливість Excel, як *умовне форматування*. Тобто вносити певний формат до клітинки, виконуючи спеціально створену умову. Це може бути зміна шрифту, його кольору, заливка клітинки тощо.

Наприклад, якщо в деякій клітинці або діапазоні клітинок у результаті введення даних або обчислень виконується задана умова (значення більше або менше деякого числа тощо), то колір тексту, заливки, межі і т. д. змінюються певним чином автоматично.

Створюючи умовне форматування в Excel, широко застосовують готові рішення, такі як кольорові шкали, гістограми, набори значків.

## 50.1.

## ВИКОРИСТАННЯ КОЛЬОРОВОЇ ШКАЛИ

*Кольорова шкала* — це графічне рішення, що дозволяє зрозуміти розподіл даних. За допомогою кольорової шкали, використовуючи градації двох або трьох кольорів, можна порівнювати діапазони клітинок. Водночас рівень яскравості кольору відповідає найбільшим, середнім або найменшим значенням.

Використання умовного форматування з кольоровою шкалою розглянемо, виконуючи наступну вправу.





## ВПРАВА 74



**Завдання.** У таблиці з числовими даними застосуйте умовне форматування з використанням кольірної шкали до діапазону **A1:AL33** для одержання деякого зображення.

1. Відкрийте файл **Вправа\_74** з папки **Files** на диску.
2. Виділіть діапазон **A1:AL33**.
3. У групі **Стилі** на вкладці **Основне** оберіть **Умовне форматування** ▶ **Кольорові шкали** (рис. 7.32).
4. Оберіть першу кольорову схему.
5. Поекспериментуйте з вибором інших схем кольорових шкал.
6. Спробуйте застосувати до даного діапазону команду умовного форматування **Набори піктограм**

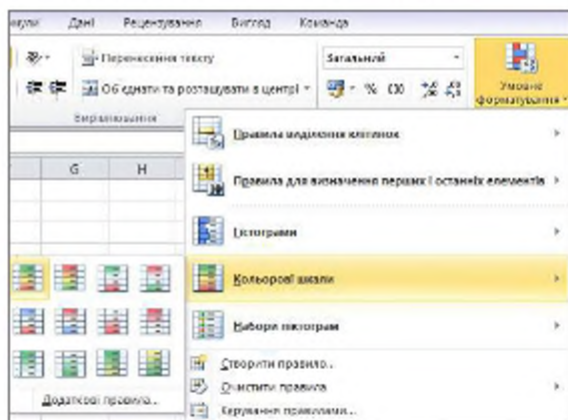


Рис. 7.32

7. Збережіть файл під назвою **Вправа\_74\_Прізвище**.

## 50.2.

### ВИКОРИСТАННЯ ПРАВИЛ ВИДІЛЕНИХ КЛІТИНОК

Форматування за допомогою правил виділених клітинок застосовують до клітинок з текстом, числовими значеннями та значеннями дати або часу. Дане форматування використовують для спрощення пошуку деяких клітинок у діапазоні з застосуванням оператора порівняння.

Для реалізації цього виду умовного форматування потрібно виконати послідовність дій.

1. Виділіть необхідний діапазон.
2. У групі **Стилі** на вкладці **Основне** оберіть **Умовне форматування** ▶ **Правила виділених клітинок**.
3. У меню **Правила виділених клітинок** оберіть потрібну команду порівняння: **більше**, **менше**, **між**, **дорівнює** тощо.
4. Якщо, наприклад, обрати команду **Більше**, то у вікні **Більше** в полі **Форматувати клітинки, які БІЛЬШІ ЗА:** потрібно ввести значення або клацнути на клітинці з необхідним значенням, а у вікні поряд обрати бажаний спосіб форматування клітинки при виконанні вказаної умови (рис. 7.33).

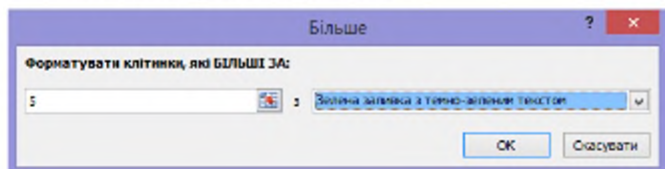


Рис. 7.33



## 50.3.

## ВИКОРИСТАННЯ ГІСТОГРАМ

Гістограми в умовному формативанні використовують для порівняння значення клітинки відносно значень інших клітинок. При цьому довжина гістограми відповідає значенню клітинки (чим більше значення — тим вона довші).

Використовувати такий спосіб умовного формативання доречно до даних, наприклад, такої таблиці, як на рис. 7.37.

Щоб застосувати такий вид умовного формативання для виділеного діапазону, треба обрати команду **Гістограми** з меню інструмента **Умовне формативання**. Після чого вибрати бажаний вид гістограми з запропонованих схем.

|    | A        | B                      |
|----|----------|------------------------|
| 1  | Місяць   | Кількість опадів у мм. |
| 2  | січень   | 37,2                   |
| 3  | лютий    | 11,4                   |
| 4  | березень | 16,5                   |
| 5  | квітень  | 19,5                   |
| 6  | травень  | 11,7                   |
| 7  | червень  | 129,3                  |
| 8  | липень   | 57,1                   |
| 9  | серпень  | 43,8                   |
| 10 | вересень | 8,7                    |
| 11 | жовтень  | 86                     |
| 12 | листопад | 12,5                   |
| 13 | грудень  | 21,2                   |

Рис. 7.37

## 50.4.

## СТВОРЕННЯ ВЛАСНОГО ПРАВИЛА З ВИКОРИСТАННЯМ ФОРМУЛ

Бувають випадки, коли заготовок, що містить інструмент **Умовне формативання**, недостатньо для розв'язання поставленої задачі. У вправі 90 ми вирішимо завдання, в якому треба виконати формативання залежно від значень, що розташовані не в самій клітинці, де здійснюється формативання, а в деякій іншій клітинці.

## ВПРАВА 76



**Завдання.** Дано таблицю, що містить середній бал учнів і значення прохідного бала для вступу до деякого ВНЗ. Виділити рожевим кольором середні бали тих учнів, що мають можливість вступити до даного ВНЗ.

1. Створіть таблицю за зразком (рис. 7.38) або відкрийте файл **Вправа\_76** з папки **Files** на диску.
2. Виділіть діапазон клітинок **C2:C9**.
3. На вкладці **Основне** в групі **Стилі** оберіть команду **Умовне формативання** **Створити правило...**
4. У вікні **Нове правило формативання** оберіть команду **Використовувати формулу для визначення клітинок для формативання**.
5. У полі **Форматувати значення, для яких ця формула має значення ІСТИНА** введіть формулу: **=IF(C2>=\$C\$11;TRUE)** або **=C2>=\$C\$11** (рис. 7.39).

|    | A | B                  | C            |
|----|---|--------------------|--------------|
| 1  | № | Прізвище, ім'я     | Середній бал |
| 2  | 1 | Андрієць Олександр | 7,5          |
| 3  | 2 | Бондар Станіслав   | 9,8          |
| 4  | 3 | Буянова Олеся      | 10,2         |
| 5  | 4 | Городній Микола    | 6,3          |
| 6  | 5 | Здоровець Олексій  | 11,1         |
| 7  | 6 | Іванов Володимир   | 8,9          |
| 8  | 7 | Іванов Сергій      | 5,9          |
| 9  | 8 | Іванченко Артем    | 9,5          |
| 10 |   |                    |              |
| 11 |   | Прохідний бал      | 9,5          |

Рис. 7.38

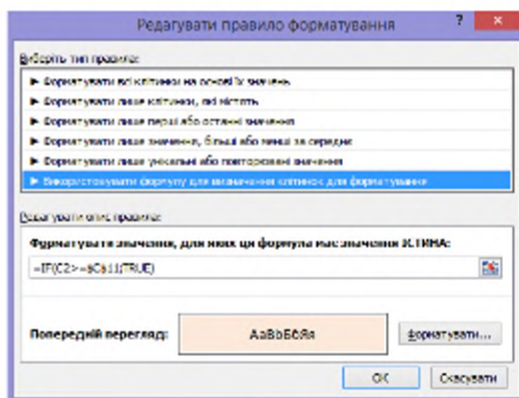


Рис. 7.39

|    | A | B                   | C            |
|----|---|---------------------|--------------|
| 1  | № | Прізвище, ім'я      | Середній бал |
| 2  | 1 | Андрієць Олександра | 7,5          |
| 3  | 2 | Бондар Станіслав    | 9,8          |
| 4  | 3 | Буянова Олеся       | 10,2         |
| 5  | 4 | Городний Микола     | 6,3          |
| 6  | 5 | Здоровець Олексій   | 11,1         |
| 7  | 6 | Іванов Володимир    | 8,9          |
| 8  | 7 | Іванов Сергій       | 5,9          |
| 9  | 8 | Іванченко Артем     | 9,5          |
| 10 |   |                     |              |
| 11 |   | Прохідний бал       | 9,5          |

Рис. 7.40

Тут **C2** є відносним посиланням, оскільки значення в діапазоні змінюються, **\$C\$11** — абсолютне, оскільки воно є сталим для усіх значень діапазону.

- Клацніть на кнопку **Форматувати...** і встановіть рожевий колір зафарбування клітинки.
- Натисніть **ОК**. У результаті вікно таблиці буде виглядати, як на рис. 7.40.

**Примітка.** Якщо потрібно змінити створене правило, виділіть клітинки, до яких застосовано умовне форматування, та виконайте команду **Умове форматування** ▶ **Керування правилами...**, оберіть правило та натисніть кнопку **Редагувати правило...**

- Змініть значення прохідного бала на більше або менше значення, слідкуючи за змінами форматування таблиці.
- Самостійно створіть правило, щоб виділеними були не тільки значення середніх балів, а також прізвище та ім'я учня.
- Збережіть файл під назвою **Вправа\_76\_Прізвище**.



### Висновки

Щоб змінити формат клітинки залежно від деяких умов, використовують *умовне форматування*.

При створенні умовного форматування в Excel використовують такі готові рішення, як кольорові шкали, гістограми, набори піктограм.

Для форматування клітинок, що відповідають певному діапазону значень, використовують *правила виділених клітинок*.

Якщо для умовного форматування клітинок недостатньо готових рішень, користувач має можливість створити власне правило.



### Контрольні запитання та завдання

- Для чого використовують умовне форматування?
- Які способи умовного форматування ви знаєте?
- Як виконати умовне форматування виділеного діапазону?
- Який спосіб умовного форматування потрібно застосувати для виділення клітинок, значення яких належить числовому діапазону?



### Питання для роздумів

- Наведіть приклади завдань, для яких доцільно використовувати умовне форматування за допомогою гістограм.
- Чи можна при створенні власного правила умовного форматування скористатися майстром формул?



## Завдання для досліджень

- 1\*. Вкажіть послідовність дій встановлення умовного форматування для клітинок, які містять дати з тими подіями, що відбулися раніше від поточної дати.
- 2\*. Дослідіть, як можна встановити умовне форматування для даних, які є більшими (меншими) за середнє арифметичне даних деякого діапазону.

## ПРАКТИЧНА РОБОТА №15

## Використання математичних, логічних і статистичних функцій табличного процесора. Умовне форматування

**Завдання 1.** У заданому шаблоні робочого аркуша **Тестування** реалізувати перевірку відповідей на запитання тестів, оцінити відповіді за 12-бальною шкалою оцінювання. У бланку відповідей правильні відповіді підсвітити зеленим кольором, неправильні — рожевим.

1. Завантажте файл **ПР15.xls** з папки **Files** на диску.
2. У файлі **ПР15** на аркуші **Тестування** виконайте форматування таблиці за зразком (рис. 7.41).

**Пояснення.** У клітинках **A3:B22** введено запитання та варіанти відповідей. У клітинках **E2:H2** — правильні відповіді на запитання, згодом вони будуть приховані, щоб той, хто буде давати відповіді на запитання, їх не бачив. У клітинках **E5:H5** залежно від відповіді буде зараховано 1 або 0 балів, а в клітинці **E8** має бути виставлена оцінка за тест.

3. До клітинок **E4:H4** введіть відповіді довільним чином.
4. У клітинці **E5** введіть формулу для перевірки правильності відповіді та зарахування одного бала, якщо

|    | A                                      | B                           | C | D                | E             | F | G | H | I |   |   |   |
|----|----------------------------------------|-----------------------------|---|------------------|---------------|---|---|---|---|---|---|---|
| 1  | Уведіть відповіді до бланку відповідей |                             |   | Бланк відповідей |               |   |   |   |   |   |   |   |
| 2  |                                        |                             |   | етапон           |               |   |   |   |   |   |   |   |
| 3  | 1                                      | Столиця України...          |   |                  | № запитання   |   |   |   | 1 | 2 | 3 | 4 |
| 4  | a                                      | Харків                      |   |                  | Відповіді     |   |   |   |   |   |   |   |
| 5  | b                                      | Львів                       |   |                  | бали          |   |   |   |   |   |   |   |
| 6  | v                                      | Одеса                       |   |                  |               |   |   |   |   |   |   |   |
| 7  | г                                      | Київ                        |   |                  |               |   |   |   |   |   |   |   |
| 8  | 2                                      | Скільки областей в Україні? |   |                  | Усього балів: |   |   |   |   |   |   |   |
| 9  | a                                      | 25                          |   |                  |               |   |   |   |   |   |   |   |
| 10 | b                                      | 24                          |   |                  |               |   |   |   |   |   |   |   |
| 11 | v                                      | 22                          |   |                  |               |   |   |   |   |   |   |   |
| 12 | г                                      | 27                          |   |                  |               |   |   |   |   |   |   |   |
| 13 | 3                                      | Найдовше місто України...   |   |                  |               |   |   |   |   |   |   |   |
| 14 | a                                      | Київ                        |   |                  |               |   |   |   |   |   |   |   |
| 15 | b                                      | Кривий Ріг                  |   |                  |               |   |   |   |   |   |   |   |
| 16 | v                                      | Чернігів                    |   |                  |               |   |   |   |   |   |   |   |
| 17 | г                                      | Дніпропетровськ             |   |                  |               |   |   |   |   |   |   |   |
| 18 | 4                                      | Найдовша річка України      |   |                  |               |   |   |   |   |   |   |   |
| 19 | a                                      | Десна                       |   |                  |               |   |   |   |   |   |   |   |
| 20 | b                                      | Південний Буг               |   |                  |               |   |   |   |   |   |   |   |
| 21 | v                                      | Дніпро                      |   |                  |               |   |   |   |   |   |   |   |
| 22 | г                                      | Дністер                     |   |                  |               |   |   |   |   |   |   |   |

Рис. 7.41

відповідь є правильною: **=IF(E2=E4;1;0)**. Скопіюйте цю формулу до діапазону **F5:H5**, використовуючи автозаповнення.

5. У клітинці **I5** полічіть загальну кількість набраних балів за тестування.



**Дізнатися більше**  
про захист клітинок  
ви зможете на диску.

6. У клітинці **E8** введіть формулу для виставлення оцінки за 12-бальною системою оцінювання. Для цього потрібно загальну кількість балів (**15**) поділити на кількість запитань і помножити на 12. Результат має бути цілим числом.

7. До клітинок **E4:H4** застосуйте умовне форматування: якщо в клітинці відповідь правильна, то вона зафарбована в зелений колір, якщо неправильна, то — в рожевий.

а) Виділіть діапазон **E4:H4**.

б) На вкладці **Основне** в групі **Стил** оберіть команду **Умове форматування** → **Створити правило**.

в) У вікні **Нове правило форматування** оберіть команду **Використовувати формулу для визначення клітинок для форматування**.

г) У полі **Форматувати значення, для яких ця формула має значення ІСТИНА** введіть формулу: **=IF(E5=1;TRUE)** або **=IF(E2=E4;TRUE)**.

г) Клацніть на кнопку **Форматувати** та встановіть зелений колір зафарбування клітинки. Натисніть **ОК**.

д) Самостійно створіть правило для встановлення умовного форматування, якщо відповідь неправильна.

8. Приховайте проміжні обчислення та правильні відповіді, щоб особа, яка проходить тестування, їх не бачила.

а) Виділіть клітинки **D2:H2** і **D5:I5**.

б) Встановіть для них білий колір літер.

**Примітка.** Excel у своєму арсеналі має і більш професійний підхід для приховування даних і формул з подальшим захистом аркуша від змін. При бажанні з цією можливістю ви можете ознайомитися самостійно.

9. Змініть формулу в клітинці **E8**, враховуючи, якщо користувач не надав жодної відповіді, він має замість 0 балів отримати оцінку 1 бал.

**Завдання 2.** Розв'язати квадратне рівняння. Відповідь представити у вигляді двох чисел округлених до двох десяткових знаків або у вигляді повідомлення «немає розв'язків». Повідомлення про відсутність розв'язків підсвітити жовтим кольором, додатні розв'язки — рожевим, а від'ємні — блакитним.

1. Перейдіть до аркуша **Квадратне рівняння**.

2. У клітинках **A2:C2** введіть довільні числа, які є коефіцієнтами квадратного рівняння.

3. У клітинці **E2** обчисліть дискримінант квадратного рівняння за формулою:

$$D = b^2 - 4ac$$

4. До клітинок **B4** і **B5** введіть формули для обчислення коренів квадратного рівняння з урахуванням умови, якщо  $D < 0$ , то рівняння немає коренів.

а) До клітинки **B4** введіть формулу:

**=IF(E2<0;«немає розв'язків»;(-B2+SQRT(E2))/2\*A2)**.

б) До клітинки **B5** введіть формулу самостійно.



5. Виконайте умовне форматування клітинок **B4** і **B5**, вказане в умові завдання.

**Завдання 3.** На аркуші **Заробітна платня** заповнити таблицю відповідно до запропонованих умов.

Нарахування за лікарняні дні залежить від стажу роботи: до 3-х років — 50%; від 3-х до 5-ти років — 60%; від 5-ти до 8-ми років — 70 %; більше 8-ми років — 100%.

Врахувати, що податок становить 15%, а відрахування до пенсійного фонду — 2%.

Ввести формули до клітинок зафарбованих зеленим кольором та виконати їх копіювання, застосувавши дію автозаповнення.

## §51


### Створення та налагодження діаграм різного типу

У цьому класі ви навчилися створювати діаграми в середовищі MS Excel. Ознайомилися з послідовністю дій щодо побудови та налаштування стовпчастих і кругових діаграм, навчилися визначати основні їх елементи.

Нагадаємо, що *діаграма* — це наочне графічне представлення різних даних, що є зручним засобом для порівняння і демонстрації. Діаграму можна розташувати на аркуші з даними, а також на окремому аркуші книги. Щоб створити діаграму, потрібно виділити дані, на основі яких вона буде побудована, вибрати тип діаграми в групі **Діаграми** на вкладці **Вставлення**. Нижче розглянемо типи діаграм, які найбільш широко використовують.

#### 51.1.

#### ОСНОВНІ ТИПИ ДІАГРАМ

*Стовпчасті діаграми* , як правило, використовують для відображення змін даних за певний проміжок часу або демонстрації порівняння різних об'єктів. До того ж категорії в них зазвичай формуються по горизонтальній осі, а значення — по вертикалі (рис. 7.42).


*Лінійчаті діаграми* (або графіки)  дозволяють демонструвати в єдиному масштабі неперервну зміну даних за рівні проміжки часу. В графіку дані категорій розподіляються по горизонтальній осі, а значення — уздовж вертикальної осі (рис. 7.43).



Рис. 7.42



Рис. 7.43



Рис. 7.44.



Рис. 7.45.

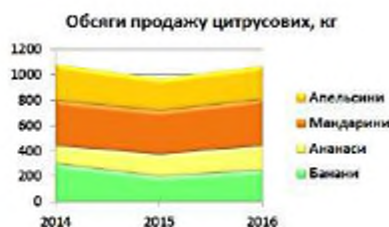


Рис. 7.46.

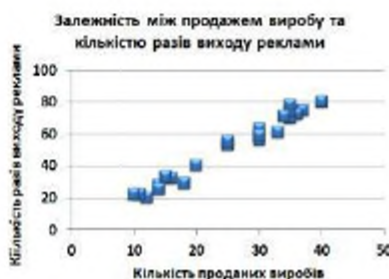







Рис. 7.47.


Для побудови інших, менш уживаних на практиці, діаграм у MS Excel використовують меню інструмента  **Інші діаграми**.

Якщо тип обраної діаграми вас не влаштовує, його можна змінити в будь-який момент. Для цього у вкладці **Конструктор**, треба обрати інструмент  **Змінити тип діаграми**, далі в діалоговому вікні, що відкриється, вибрати бажаний тип діаграми.

**Секторні діаграми**  відображають величину розміру елементів одного ряду даних пропорційно до суми всіх елементів. Точки даних у секторній діаграмі відображаються у вигляді відсоткової частки від цілого круга (рис. 7.44).

**Гістограми**  мають те ж призначення, що й стовпчасті діаграми, але з іншою орієнтацією осей. На відміну від стовпчастих діаграм, тут вісь x (вісь категорій) розташована вертикально, а вісь y (вісь значень) — горизонтально. Горизонтально розташовані осі є більш зручними для порівняння різних величин у межах одного часового періоду (рис. 7.45).

**Діаграми з областями**  демонструють величину змін між кількома рядами даних залежно від часу. Наприклад, на основі даних, які показують прибуток у часі, можна побудувати діаграму з ділянками (областями), щоб підкреслити загальну суму прибутку. Відображаючи суму даних, діаграма з областями також показує відношення частин до цілого (рис. 7.46).

**Точкові діаграми**  використовують для визначення потенційних зв'язків між двома змінними. На точковій діаграмі значення двох осей об'єднуються в одну точку даних і розміщуються з різними інтервалами. Такі діаграми, як правило, використовують для виведення та порівняння наукових, статистичних або інженерних даних (рис. 7.47). Завдяки цьому типу діаграм можна будувати графіки функцій, здійснювати графічне розв'язання системи рівнянь тощо.



Зазначимо, що кожний тип діаграм має свої різновиди. Так **Стовпчасті діаграми** можуть бути плоскими, об'ємними, циліндричними, конічними, пірамідальними (рис. 7.48). Залежно від мети подання даних використовують звичайну діаграму і з накопиченням. У *звичайній стовпчастій діаграмі* графічні зображення числових даних з різних наборів розташовано поруч один з одним. У *діаграмі з накопиченням* — один на одному. Це дає змогу оцінити сумарні дані і внесок кожної складової в загальну суму. Користувач може обрати потрібний різновид з меню інструмента відповідного типу діаграм.

## 51.2.

### ВИБІР ТИПУ ДІАГРАМИ

Основне завдання побудови діаграм — вибрати правильно тип, щоб коректно відобразити дані. Найбільш широко використовують графіки, стовпчасті та кругові діаграми.

Для побудови графіків враховують:

- чи є залежність між двома числовими величинами;
- значення числових величин визначені через однакові проміжки часу.

Використання графіка передбачає демонстрацію деякого неперервного процесу. Також графіки слід застосовувати, коли потрібно відобразити дуже велику кількість значень.

Побудова стовпчастих діаграм доцільна, коли:

- є залежність між двома числовими величинами;
- дані, що мають бути відображені на діаграмі, не демонструють неперервний процес.

Побудову кругових діаграм здійснюють, якщо:

- необхідно відобразити дані лише одного ряду;
- жодне із значень не є нульовим або від'ємним;
- категорії відповідають частинам єдиного цілого.

Розглянемо вправу, що передбачає побудову нового типу діаграм — графіка, і пригадаємо послідовність дій щодо створення та налаштування діаграм.

## ВПРАВА 77



**Завдання.** Дано таблицю одержання та відправлення електронних повідомлень деяким користувачем упродовж тижня (рис. 7.49). Побудувати діаграму, що має наочно показати представлені дані.

1. Оберіть тип діаграми для відображення даних.

Оскільки є залежність між двома величинами (залежно від дати надіслано або відправлено певну кількість листів), а також значення відтворюють певний процес через однаковий проміжок часу, то для демонстрації цих даних доцільно побудувати графіки.



Рис. 7.48. Меню інструмента **Стовпчаста**



|   | A     | B           | C        |
|---|-------|-------------|----------|
| 1 | Дата  | Відправлено | Одержано |
| 2 | 01.02 | 20          | 11       |
| 3 | 02.02 | 10          | 5        |
| 4 | 03.02 | 5           | 8        |
| 5 | 04.02 | 4           | 9        |
| 6 | 05.02 | 10          | 15       |
| 7 | 06.02 | 4           | 6        |
| 8 | 07.02 | 7           | 10       |

**Рис. 7.49**



**Рис. 7.50**

встановить у розділах **Параметри маркера**, **Заливка маркера** та **Колір лінії маркера**;

- у відповідних розділах встановить колір, товщину і тінь лінії;
- аналогічно налаштуйте зовнішній вигляд другого графіка.

в) Додайте підпис вертикальної осі, обравши інструмент **Назви осей** на вкладці **Макет**.

г) Додайте підписи значень на графіку, обравши інструмент **Підписи даних** на вкладці **Макет**.

г) Змініть за потреби розмір шрифту написів на діаграмі, обираючи необхідний елемент і встановлюючи параметри шрифту в групі **Шрифт** на вкладці **Основне**.

4. Збережіть файл під назвою **Вправа\_77\_Прізвище**.

### ВПРАВА 78



**Завдання.** Таблиця містить три набори даних. Проаналізувати представлені дані, обрати необхідний тип діаграм для відображення цих даних і побудувати діаграми.


|    | A                                | B                 | C                     | D            | E             | F                     | G     | H        |
|----|----------------------------------|-------------------|-----------------------|--------------|---------------|-----------------------|-------|----------|
| 1  |                                  |                   |                       |              |               |                       |       |          |
| 2  | Розподіл часу учня протягом доби |                   | Продаж товару за день |              |               | Котивання курсу валют |       |          |
| 3  |                                  |                   |                       |              |               |                       |       |          |
| 4  | Вид діяльності                   | Виграти часу, год |                       | Найменування | Кількість, кг |                       | Дата  | Значення |
| 5  | Навчання                         | 6                 |                       | Зефір        | 25,5          |                       | 01.02 | 24,58    |
| 6  | Приєм їжі                        | 2                 |                       | Тістечко     | 36,7          |                       | 02.02 | 24,67    |
| 7  | Сон                              | 8                 |                       | Морозиво     | 75,4          |                       | 03.02 | 25,11    |
| 8  | Відпочинок                       | 5,5               |                       | Мафін        | 23,3          |                       | 04.02 | 25,33    |
| 9  | Заняття спортом                  | 2,5               |                       | Круасан      | 16,5          |                       | 05.02 | 24,82    |
| 10 |                                  |                   |                       |              |               |                       |       |          |

**Рис. 7.51**



1. Введіть дані до аркуша електронної таблиці за зразком (рис. 7.51).
2. Розгляньте дані розподілу часу учня протягом доби. Виберіть тип діаграми.  
Оскільки дані містять усього один ряд, не містять від'ємних і нульових значень, а також є складовими єдиного цілого, а саме доби, то для даного набору даних доцільною буде побудова секторної діаграми.
3. Побудуйте діаграму розподілу часу учня протягом доби.
  - а) Виділіть діапазон **A5:B9**.
  - б) На вкладці **Вставлення** оберіть інструмент **Секторна** та бажаний вид діаграми. В результаті діаграму буде побудовано.
4. Виконайте оформлення діаграми, додавши заголовок і значення біля її секторів.
  - а) На вкладці **Макет** оберіть інструмент **Назва діаграми**, обравши бажаний спосіб розташування назви діаграми. Введіть назву діаграми.
  - б) Додайте підписи даних у відсотках:
    - на вкладці **Макет** оберіть інструмент **Підписи даних** і клацніть на команді **Інші параметри підпису даних**;
    - у розділі **Підписи даних** зніміть прапорець з параметра **Значення** та встановіть біля параметра **Відсотки**;
    - у цьому ж розділі встановіть розташування підписів **Коло вершини, ззовні**.
  - в) Змініть кольорове оформлення секторів за власним бажанням.
5. Розгляньте дані продажу товарів за день. Виберіть тип діаграми.  
Оскільки між двома числовими величинами немає залежності, дані не є часткою деякого цілого значення, то для їх представлення доцільно побудувати стовпчасту діаграму або гістограму.
6. Самостійно побудуйте стовпчасту діаграму продажу товарів за день, виконайте її оформлення, додайте назву та вилучте легенду.
7. Розгляньте дані коливання курсу валют. Виберіть тип діаграми.  
Оскільки є залежність між двома величинами (залежно від дати встановлено певний курс валют), а також значення відтворюють певний процес через однаковий проміжок часу, то для демонстрації цих даних доцільно побудувати графік.
8. Побудуйте графік коливання курсу валют без використання маркерів, оформіть його, надавши йому червоний колір з рожевою підсвіткою, додайте назву і вилучте легенду.
9. Збережіть файл під назвою **Вправа\_78\_Прізвище**.



**Виконати** вправу «Графіки функцій» ви зможете на диску. 



### Висновки

*Діаграма* — це наочне графічне представлення різних даних, що є зручним засобом для порівняння і демонстрації.

У MS Excel можна побудувати такі основні типи діаграм: стовпчасті, секторні, лінійчаті, точкові, гістограми, діаграми з областями, для побудови яких треба обрати відповідний інструмент на вкладці **Вставлення**.

Для вибору інших типів діаграм необхідно обрати інструмент **Інші діаграми** на вкладці **Вставлення**.

Для зміни типу діаграми необхідно обрати інструмент **Змінити тип діаграми** на вкладці **Конструктор**.

Правильний вибір типу діаграми — одне з основних завдань у процесі їх побудови.



#### Контрольні запитання та завдання

1. Які є типи діаграм?
2. У яких випадках використовують секторні діаграми?
3. Які типи діаграм демонструють неперервну зміну даних за рівні проміжки часу?
4. Для чого використовують діаграми з областями?
5. Як змінити тип побудованої діаграми?



#### Питання для роздумів

- 1\*. Який тип діаграм краще підходить для побудови графіків функцій?
- 2\*\*. Розробіть алгоритм вибору типу діаграми залежно від числових даних.



#### Завдання для досліджень

- 1\*. Визначте, як побудувати діаграми різних типів на одній ділянці побудови. Наприклад, стовпчасту діаграму та графік.
- 2\*\*. Дослідіть призначення типів діаграм, доступ до яких можна отримати в меню інструмента **Інші діаграми**.

## §52

### Упорядкування даних у таблицях. Автоматичні та розширені фільтри. Проміжні підсумки

#### 52.1.

#### УПОРЯДКУВАННЯ ДАНИХ У ТАБЛИЦЯХ

Якщо користувач працює з великою кількістю даних, то зручно їх сортувати.



*Сортування* — це розташування даних таблиці в потрібному порядку. Наприклад, від меншого до більшого (за зростанням) або навпаки від більшого до меншого (за спаданням).





Дізнатися більше про сортування ви зможете на диску.

MS Excel має можливості сортування не тільки за значенням даних, а й за кольором клітинки або шрифту. Впорядковувати можна текст, числа, дати та час. Як правило, сортування даних здійснюється по стовпцях. До того ж заголовки стовпців мають бути розташовані тільки в одному рядку.

Для сортування текстових даних потрібно звернути увагу, щоб усі клітинки мали текстовий формат і на початку тексту не містили пробілів. Якщо деякі клітинки будуть мати числовий формат, то при сортуванні за зростанням вони опиняться вище клітинок, що мають числові дані, збережені в текстовому форматі. Текст з пробілами також буде вище від того, який не містить пробілів.

Щоб посортувати дані, треба: 1) клацнути в будь-якій клітинці стовпця з даними для сортування; 2) скористатися одним із способів сортування.

*1 спосіб.* На вкладці **Основне** обрати в меню інструмента **Сортування й фільтр** команду **Сортування від А до Я** або **Сортування від Я до А**.

*2 спосіб.* На вкладці **Дані** обрати один з інструментів  **Сортування від А до Я** або  **Сортування від Я до А**.

У результаті буде здійснено сортування даних у тому стовпці, в якому було виділено клітинку, а значення, що знаходяться у відповідних рядках будуть також переміщені.

Щоб посортувати дані в межах тільки одного стовпця, спочатку треба виділити його і лише тоді здійснити сортування.

Excel має більш ширші можливості сортування. Щоб їх реалізувати, потрібно клацнути в будь-якій клітинці таблиці та викликати діалогове вікно **Сортування** одним із способів.

*1 спосіб.* На вкладці **Основне** обрати в меню інструмента **Сортування й фільтр** команду **Настроюване сортування...**

*2 спосіб.* На вкладці **Дані** обрати інструмент  **Сортувати**.

У діалоговому вікні, що з'явилося, обрати потрібні стовпці в першому полі. У другому полі обрати, за якими даними буде сортуватися (за значенням, кольором тощо), у третьому полі вказати порядок сортування. Якщо збігаються дані в результаті сортування першого вказаного стовпця, то можна посортувати ще по одному стовпцю, обравши кнопку **Додати рівень**.

На рис. 7.52 показано приклад сортування даних спочатку за стовпцем **Видавництво**, а в разі збігу — по стовпцю **Назва**.

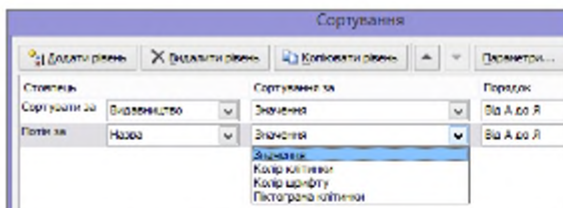


Рис. 7.52. Вікно Сортування

## 52.2.

## ФІЛЬТРАЦІЯ ДАНИХ

Сортування не завжди спрощує роботу з обробкою великих обсягів інформації. Іноді зручно здійснити пошук даних, що відповідають деякому критерію. В таких випадках треба скористатися можливостями фільтрації даних. *Фільтрація* дозволяє виводити на екран лише ті значення, які відповідають заданим умовам відбору для стовпців, не змінюючи сам документ.



Дізнатися більше про «розумні» таблиці ви зможете на диску.

Дізнатися більше про «розумні» таблиці ви зможете на диску.

Дізнатися більше про «розумні» таблиці ви зможете на диску.



Відфільтровані дані можна змінювати, редагувати, використовувати для створення діаграм, але водночас не можна змінювати та переміщувати їх порядок.

Є два типи фільтрації списків. Для створення простих умов відбору використовують *автофільтр*, а для більш складних — *розширений фільтр*.

Автофільтр використовують, коли умова вибору стосується тільки одного стовпця або коли необхідно одночасно врахувати умови декількох стовпців.

Для застосування автофільтра треба виконати такі кроки:

1. Клацнути на будь-якій клітинці таблиці даних.
2. Застосувати до таблиці режим автофільтра одним із способів.

*1 спосіб.* На вкладці **Основне** обрати в меню інструмента **Сортування й фільтр** команду **Фільтр**.



Фільтр

*2 спосіб.* На вкладці **Дані** обрати інструмент **Фільтр**.

Після чого в клітинках першого рядка буде відображено кнопки з трикутниками (рис. 7.53).

|   | A                         | B              | C    | D           | E           | F                  |
|---|---------------------------|----------------|------|-------------|-------------|--------------------|
| 1 | Назва                     | Автор          | Ціна | Видавництво | Рік випуску | Кількість сторінок |
| 2 | Зірка з ліжка та сніданку | Джаклін Вілсон | 21   | Богдан      | 2011        | 224                |
| 3 | П'ятеро дітей та ельф     | Несбіт Е.      | 45   | Богдан      | 2015        | 280                |

Рис. 7.53

3. Клацнути на кнопці з трикутником у стовпці, який треба відфільтрувати. Зі списку, що відкриється, вибрати умову фільтрації.

Список фільтрації містить перелік усіх даних (без повторень), що є у стовпці, команди сортування і команду переходу до текстових або числових фільтрів залежно від даних стовпця (рис. 7.54).

Для вибору одного або декількох значень із запропонованих даних списку необхідно залишити навпроти них прапорці, а інші забрати. Після натиснення кнопки **ОК** буде виведено лише вибрані значення. Якщо натиснути на прапорець **(Виділити все)**, буде знято всі прапорці.

Для встановлення умов фільтрації використовують команди **Текстові фільтри**, **Фільтри чисел** або **Фільтри дат** залежно від типів даних.

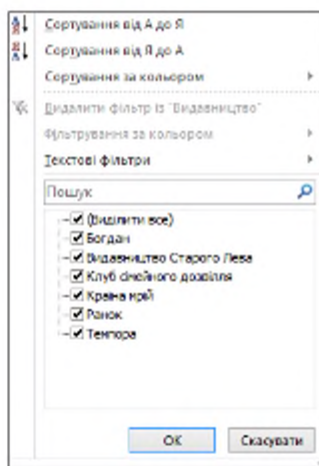


Рис. 7.54



Розглянемо приклад, коли у стовпці з числовими даними потрібно вивести всі значення більші від 100 та менші від 200.

Для цього в меню команди **Фільтри чисел** треба обрати команду **Між**. У результаті має відкритися вікно **Користувацький автофільтр**. Для даного прикладу в цьому вікні треба встановити параметри, як на рис. 7.55.

При фільтруванні текстових даних зручно використовувати символи «?» і «\*», що дозволяють здійснювати пошук за неповними даними. Тут знак «?» визначає один будь-який символ, а «\*» — будь-яку кількість довільних символів. Так запис «a\*» буде означати, що необхідно вивести всі дані, що починаються на літеру «а».

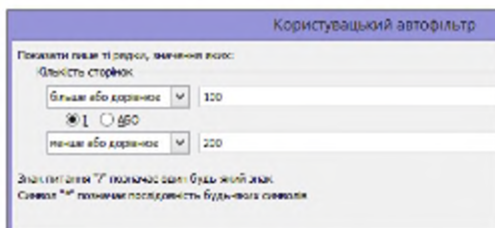


Рис. 7.55

Щоб зняти фільтр з даного стовпчика, треба клацнути на трикутник біля назви стовпчика та обрати команду **Видалити фільтр із**. Щоб відмінити фільтр на всьому аркуші, потрібно на вкладці **Дані** зняти виділення з інструмента **Фільтр**, клацнувши на ньому.

|             | I | J           |
|-------------|---|-------------|
| Автор       |   | Рік випуску |
| Дара Корній |   |             |
|             |   | 2012        |

Рис. 7.56

Якщо значення деякого параметра має задовольняти умову, що складається більш ніж з двох частин або частини умови, які стосуються різних параметрів, з'єднані сполучником «або», використовують розширені фільтри.

Щоб застосувати розширений фільтр, треба виконати наступні кроки.

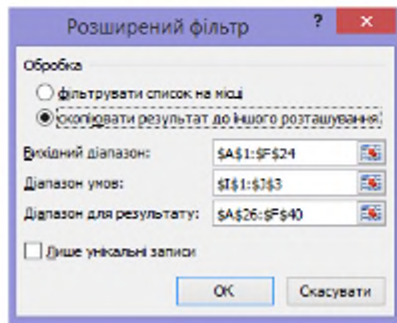


Рис. 7.57

1. На аркуші створити критерій відбору даних, вказавши назви стовпців і значення пошуку. Так, на рис. 7.56 представлено критерій пошуку книжок за ім'ям автора — Дара Корній, або за 2012 роком видання.
2. На вкладці **Дані** обрати в групі **Сортування й фільтр** вибрати кнопку **Додатково**. В результаті з'явиться діалогове вікно **Розширений фільтр** (рис. 7.57).
3. У полі **Вихідний діапазон** встановити діапазон, у якому розташована таблиця з даними. У полі **Діапазон умов** — діапазон з критерієм. Натиснути **ОК**.

Після чого буде відображено лише дані, що задовольняють критерій відбору (рис. 7.58).

|    | A                    | B            | C    | D                       | E           | F                  |
|----|----------------------|--------------|------|-------------------------|-------------|--------------------|
| 1  | Назва                | Автор        | Ціна | Видавництво             | Рік випуску | Кількість сторінок |
| 12 | Зворотній бік світла | Дара Корній  | 37   | Клуб сімейного дозвілля | 2012        | 320                |
| 13 | Зірка для тебе       | Дара Корній  | 36   | Клуб сімейного дозвілля | 2013        | 304                |
| 14 | Ключі від ліфта      | Іванцова М.  | 44   | Клуб сімейного дозвілля | 2012        | 288                |
| 16 | Робінзон Крузо       | Даніель Дефо | 29   | Країна мрій             | 2012        | 272                |

Рис. 7.58

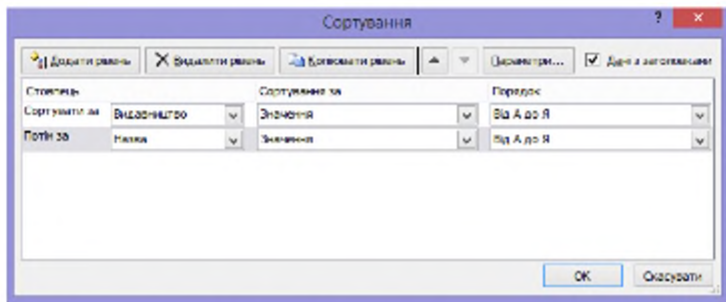


Рис. 7.59

4. Застосуйте до таблиці проміжні підсумки для визначення кількості книжок кожного видавництва.
- Клацніть у довільному місці таблиці.
  - На вкладці **Дані** оберіть інструмент **Проміжні підсумки**.

в) У діалоговому вікні, що відкриється, встановіть дані згідно рис. 7.60 і натисніть **ОК**.

- 5\*. На другому аркуші самостійно визначте кількість книжок, виданих за кожний рік, використовуючи проміжні підсумки.

6. На третьому аркуші визначте всі книжки, прізвища авторів яких починаються з літери «К».
- Клацніть у довільному місці таблиці.
  - На вкладці **Дані** оберіть інструмент **Фільтр**.
  - Клацніть на кнопці з трикутником у стовпці **Автор**.
  - У полі пошуку введіть запис «к\*» і натисніть **ОК**.

7. На четвертому аркуші виведіть інформацію про книжки видавництва «Богдан» або ті, що видані в 2015 році, застосувавши розширений фільтр.

а) У діапазоні I1:J3 створіть критерій відбору, як на рис. 7.61.

**Пояснення.** Оскільки значення умов розташовані в різних рядках, до них буде застосовано сполучник «або». Для застосування сполучника «і» дані розташовують в одному рядку.

б) Клацніть у довільному місці таблиці з даними.

в) На вкладці **Дані** в групі **Сортування й фільтр** оберіть кнопку **Додатково**. В результаті з'явиться діалогове вікно **Розширений фільтр**.

г) У полі **Вихідний діапазон** встановіть діапазон, у якому розташована таблиця з даними.

ґ) У полі **Діапазон умов** — діапазон з критерієм I1:J3.

д) Встановіть опцію **Скопіювати результат в інше місце** та в полі **Помістити результат в діапазон** вкажіть діапазон **A26:A40**. Натисніть **ОК**. У вказаному діапазоні буде розміщено результати фільтрації.

8\*. На п'ятому аркуші виведіть самостійно інформацію про п'ять найтовстіших книжок.

9. Збережіть файл під назвою **Вправа\_79\_Прізвище**.

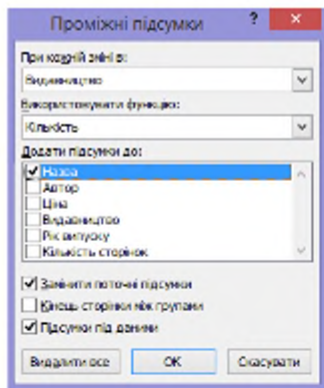


Рис. 7.60

|   | I           | J           |
|---|-------------|-------------|
| 1 | Видавництво | Рік випуску |
| 2 | Богдан      |             |
| 3 |             | 2015        |

Рис. 7.61

**Висновки**

*Сортування* — це розташування даних таблиці в заданому порядку.

Для пошуку даних за деякою умовою використовують *фільтрацію даних*. Для створення простих умов відбору використовують *автофільтр*, а для більш складних — *розширений фільтр*.

Для автоматичних розрахунків спільного підсумку для декількох рядків взаємопов'язаних даних використовують *проміжні підсумки*.

Проміжні підсумки визначають кількість, суму, добуток, середнє, максимальне та мінімальне значення.

**Контрольні запитання та завдання**

1. Які є способи сортування даних?
2. За якими параметрами можна здійснити сортування в MS Excel?
3. Для чого використовують фільтрацію даних?
4. Як застосувати автофільтр?
5. Для чого використовують проміжні підсумки?
- 6\*. Як забрати проміжні підсумки з аркуша?

**Питання для роздумів**

- 1\*. У яких випадках використовують автофільтр, а в яких — розширений фільтр?
- 2\*. Чому при обчисленні проміжних підсумків один із стовпців має бути відсортованим?

**Завдання для досліджень**

- 1\*. Дослідіть і поясніть призначення інструментів групи **Знаряддя даних** на вкладці **Дані**.
- 2\*. Визначте, яким буде критерій для розширеного автофільтра, якщо у вправі 79 потрібно вивести всі книжки, що мають більше 100 сторінок та видані у видавництві «Богдан» або «Ранок».



## §53

Встановлення параметрів сторінки.  
Друкування електронної таблиці

## 53.1.

## ВСТАНОВЛЕННЯ ПАРАМЕТРІВ СТОРІНКИ

На відміну від MS Word в Excel параметри сторінки задають, щоб вивести дані на друк, і по суті це є складовою частиною встановлення налаштувань друку.

Для налаштування параметрів сторінки потрібно скористатися одним з інструментів групи **Параметри сторінки** на вкладці **Розмітка сторінки** або викликати діалогове меню цієї групи.

На вкладці **Сторінка** (рис. 7.62) можна встановити орієнтацію сторінки, масштаб, розмір паперу та якість друку.

Зручною можливістю при налаштуванні параметрів сторінки є встановлення кількості сторінок по горизонталі та вертикалі. Якщо вибрати опцію **гранична кількість сторінок**: і встановити необхідні значення, то можна розраховувати на точну кількість аркушів, яку буде використано для друку таблиці. Дані таблиці буде автоматично стиснуто до вказаних розмірів. Зауважимо, що оскільки таблиця стискається пропорційно до її розмірів, то кількість аркушів, на яких буде розташовано таблицю, може бути меншою, але не більшою, ніж вказано.

На вкладках **Поля** та **Колонтитули** можна задати значення полів та за потреби встановити верхній та нижній колонтитули.

На вкладці **Аркуш** (рис. 7.63) важливими для виведення на друк є такі параметри:

- встановлення діапазону (за умовчанням на друк буде виведено всі дані аркуша);
- встановлення параметрів друку сітки, заголовків рядків і стовпців (за умовчанням при друці вони не виводяться);
- послідовність виведення сторінок.

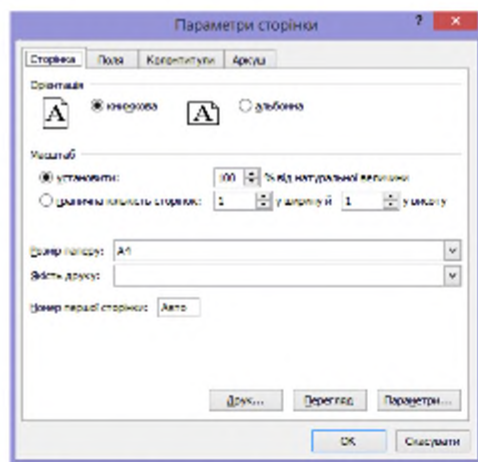


Рис. 7.62. Вкладка **Сторінка** у вікні **Параметри сторінки**

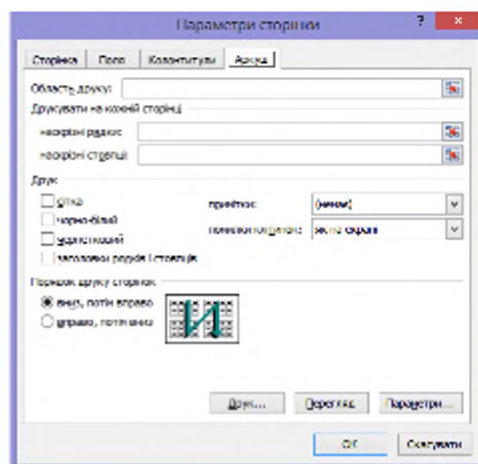


Рис. 7.63. Вкладка **Аркуш** у вікні **Параметри сторінки**

## 53.2.

## НАЛАШТУВАННЯ ДРУКУ

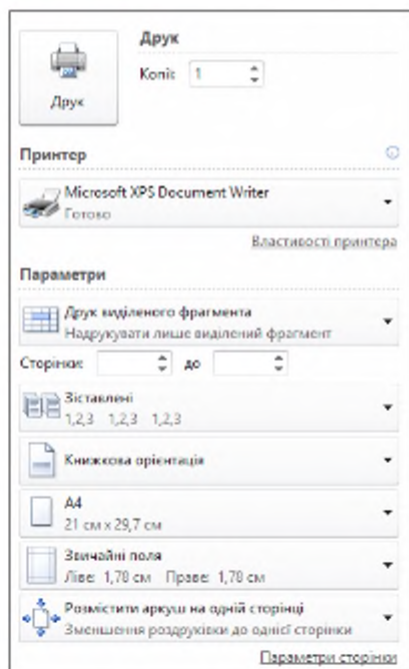


Рис. 7.64. Вікно налаштування параметрів друку



Рис. 7.65. Вікно попереднього друку

Щоб налаштувати параметри друку, потрібно виконати таку послідовність дій.


1. Перейти на вкладку **Файл** та обрати розділ **Друк**.
2. У вікні налаштувань (рис. 7.64) у групі **Друк** встановити кількість копій.
3. У групі **Принтер** обрати один із встановлених принтерів і за потреби налаштувати його, обравши посилання **Властивості принтера**.
4. У групі **Параметри** особливої уваги заслуговують перший та останній списки.
  - а) У першому списку можна обрати ділянки друку: (**Надрукувати активні аркуші**, **Надрукувати всю книгу**, **Надрукувати виділений фрагмент**);
  - б) В останньому списку визначається спосіб виведення даних на друк:

**Без масштабування** — друк аркушів з фактичними розмірами;

**Розмістити аркуш на одній сторінці** — розміри всієї таблиці зменшено в масштабі та розміщено на одній сторінці;

**Розмістити всі стовпці на одній сторінці** — таблиця по ширині буде розміщена на одній сторінці, кількість сторінок залежить від кількості рядків у ній;

**Розмістити всі рядки на одній сторінці** — таблиця по висоті буде розміщена на одній сторінці, кількість сторінок залежить від кількості стовпців у ній.

5. Для виведення документа на друк натиснути кнопку  **Друк**.

Одним з найбільш розповсюджених завдань виведення на друк є друк виділеного діапазону, який потрібно розташувати на одній сторінці. На рис. 7.65 показано встановлені параметри саме для виконання такого завдання.

Зверніть увагу, що при виконанні різноманітних налаштувань друку на ділянці справа розміщено вікно попереднього перегляду, яке демонструє, як будуть розташовані дані на аркуші після друкування.

**Висновки**

Щоб налаштувати параметри сторінки, треба скористатися одним з інструментів групи **Параметри сторінки** на вкладці **Розмітка сторінки** або викликати діалогове меню цієї групи.

Щоб налаштувати параметри друку, потрібно перейти на вкладку **Файл** та обрати розділ **Друк**.

**Контрольні запитання та завдання**

1. Які основні налаштування можна виконати у вікні **Параметри сторінки**?
2. Як вивести на друк виділений діапазон?
3. Як вивести на друк таблицю, розмістивши її на одній сторінці?
4. Де можна побачити попередні налаштування друку?

**Питання для роздумів**

- 1\*. Для чого потрібні колонтитули при виведенні документа на друк?

**Завдання для досліджень**

- 1\*. Дослідіть і поясніть можливості інструментів **Тло** та **Область друку** з групи **Параметри сторінки**.

**ПРАКТИЧНА РОБОТА №16****Упорядкування даних у таблицях.  
Автоматичні та розширені фільтри**

**Завдання.** Використовуючи таблицю з інформацією про прокат мультфільмів у кінотеатрах, вивести дані, застосувавши фільтрацію та проміжні підсумки.

1. Завантажте файл **ПР16.xls** з папки **Files** на диску.
2. Виконайте копіювання даних таблиці ще на шість аркушів.
3. На першому аркуші здійсніть сортування в алфавітному порядку спочатку за назвою мультфільму, а потім — у зворотному порядку за найменшою ціною.
  - а) Клацніть у довільному місці таблиці.
  - б) На вкладці **Дані** оберіть інструмент **Сортувати**.
  - в) У діалоговому вікні, що відкриється, встановіть дані згідно рис. 7.66 та натисніть **ОК**.



**Перевірити** свої  
знання, розгадати  
кросворд і пройти тест  
ви зможете на диску.

Рис. 7.66

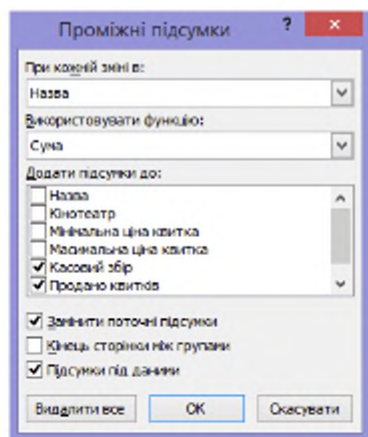
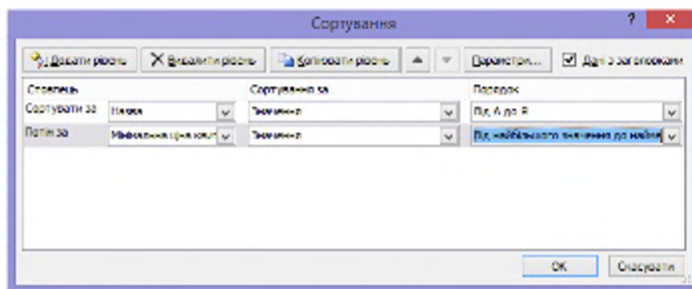


Рис. 7.67

4. Застосуйте до таблиці проміжні підсумки, щоб підрахувати суму касового збору та кількість проданих квитків за кожний мультфільм (рис. 7.67).

5. На другому аркуші визначте мінімальну та максимальну ціну квитка на кожний мультфільм, використовуючи проміжні підсумки.

а) Виконайте сортування таблиці за назвою мультфільмів.

б) Застосуйте проміжні підсумки за стовпцем **Мінімальна ціна квитка**, визначаючи найменше значення.

в) У цій же таблиці застосуйте проміжні підсумки за стовпцем **Максимальна ціна квитка**, визначаючи найбільше значення. Для додавання значень максимальної ціни квитка в діалоговому вікні **Проміжні підсумки** заберіть прапорець з опції **Замінити поточні підсумки**.

6. На третьому аркуші самостійно визначте суму касового збору та кількість проданих квитків у кожному кінотеатрі, використовуючи проміжні підсумки.

7. На четвертому аркуші визначте всі мультфільми, які було показано в кінотеатрах «Старт» і «Салют», застосувавши автофільтр.

8. На п'ятому аркуші виведіть дані про мультфільми, ціна квитків на які становить від 40 грн до 100 грн включно.

9. На шостому аркуші виведіть відомості про всі мультфільми кінотеатру «Факел» або ті мультфільми, на які було продано більше 1100 квитків.

|   | I         | J               |
|---|-----------|-----------------|
| 1 | Кінотеатр | Продано квитків |
| 2 | Факел     |                 |
| 3 |           | >1100           |

Рис. 7.68

а) У діапазоні **I1:J3** створіть критерій відбору, як на рис. 7.68.

б) Застосуйте до даних таблиці розширений фільтр з урахуванням створеного критерію, розташували результати фільтрації під таблицею з даними.

10. На сьомому аркуші виведіть інформацію про мультфільми кінотеатру «Воонег», що мають касовий збір менше 20000 грн або на них було продано менше 900 квитків.

11. Збережіть файл під назвою **ПР16\_Прізвище**.

## Розділ 8

# Розв'язування компетентнісних задач



У даному розділі ви маєте розв'язати три компетентнісні задачі. Сплануйте свою діяльність так, щоб на розв'язання перших двох задач ви витратили по два уроки, а на розв'язання третьої задачі — один урок.

### ЗАДАЧА 1

#### Комунальні послуги

Уявіть, що ви є представником деякого об'єднання співвласників багатоквартирного будинку (ОСББ). До вас звернувся мешканець будинку з проханням здійснити розрахунок вартості комунальних послуг для його сім'ї на рік. Виконайте потрібні розрахунки згідно вашого варіанта, дотримуючись вимог представлених у таблиці 8.1.



Таблиця 8.1. Вимоги до завдання

| Умова                                                         | Варіанти та відповідні значення |    |    |    |    |    |    |    |    |    |
|---------------------------------------------------------------|---------------------------------|----|----|----|----|----|----|----|----|----|
|                                                               | 1                               | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 |
| Кількість прописаних осіб                                     | 2                               | 3  | 4  | 2  | 3  | 4  | 2  | 3  | 4  | 1  |
| Площа квартири, м <sup>2</sup>                                | 48                              | 60 | 52 | 33 | 62 | 44 | 62 | 45 | 60 | 33 |
| Наявність лічильника електропостачання                        | +                               | +  | +  | +  | +  | +  | +  | +  | +  | +  |
| Наявність лічильника газопостачання                           | +                               | -  | +  | -  | +  | -  | +  | -  | +  | -  |
| Наявність лічильника холодного водопостачання та відводу води | -                               | +  | -  | +  | -  | +  | -  | +  | -  | +  |
| Наявність лічильника опалення                                 | -                               | -  | -  | -  | -  | -  | -  | -  | -  | -  |



Вартість комунальних послуг на даний момент часу завантажте з мережі Інтернет, узявши ціни в місті, яке є вашим обласним центром. Дані показників лічильників завантажте з файла **Розділ\_8\_Задача\_1**, що знаходиться в папці **Files** на диску.

За результатами обчислень побудуйте діаграму споживання електроенергії та холодної води або газу. Також побудуйте діаграму за результатами щомісячної оплати комунальних послуг.

Щоб інформувати щомісяця мешканців свого будинку, створіть шаблон оголошення, в якому потрібно буде змінювати назву місяця та вартість послуг. Окрім звернення до мешканців, оголошення має містити таблицю, зразок якої наведено нижче.

| № | Послуга        | Одиниця виміру   | Ціна в грн |
|---|----------------|------------------|------------|
| 1 | Водопостачання | 1 м <sup>3</sup> |            |
| 2 | ...            | ...              |            |

На основі створеного шаблону складіть оголошення на поточний місяць.

### ВКАЗІВКИ ДО ВИКОНАННЯ

1. Створіть інформаційну модель розв'язання задачі у вигляді схеми, де головними темами будуть поставлені завдання, а підрозділами стануть етапи їх виконання.
2. Запишіть у зошиті, які дані для розв'язання задачі вам потрібно знайти.
3. Вкажіть ключові слова пошуку потрібних відомостей.
4. Здійсніть пошук і вкажіть адреси використаних пошукових служб.
5. Вкажіть адреси сайтів, на яких знайшли потрібні дані.
6. Оберіть середовище для виконання обчислень.
7. Розробіть структуру таблиці для виконання обчислень.
8. Введіть потрібні дані до обраного середовища та виконайте обчислення.
9. Вкажіть, які діаграми доцільно побудувати для представлення результатів.
10. Побудуйте діаграми.
11. Визначте структуру оголошення.
12. Оберіть необхідний програмний засіб для створення шаблону оголошення.
13. Створіть шаблон оголошення.
14. На основі шаблону створіть оголошення на поточний місяць.



15. Представте результати своєї роботи аудиторії, зробивши висновки щодо одержаних результатів розрахунків. Обґрунтуйте наведений вами текст оголошення.
16. Оцініть виконану вами роботу за критеріями, представленими в табл. 8.2.

**Таблиця 8.2.** Критерії оцінювання завдання

| Критерій                                              | Пояснення                                                                                                                    | К-ть балів |
|-------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|------------|
| Створення інформаційної моделі                        | Створено схему послідовності дій розв'язання задачі                                                                          | 2          |
| Розуміння умов завдання                               | Правильно визначено дані необхідні для розв'язання задачі                                                                    | 2          |
| Уміння формулювати критерії відбору даних для пошуку  | Вказано правильно ключові слова для пошуку відомостей, адреси пошукових служб, адреси сайтів, на яких знайдено потрібні дані | 3          |
| Уміння створювати електронну таблицю                  | Розроблено правильно структуру електронної таблиці. Введено необхідні дані. Виконано правильно розрахунки                    | 12         |
| Уміння обирати спосіб візуалізації подання результату | Правильно визначено, які діаграми доцільно побудувати, та здійснено їх побудову, виконано форматування діаграм               | 2          |
| Уміння створювати шаблон                              | Створено шаблон оголошення згідно поставлених умов                                                                           | 3          |
| Уміння створювати документ на основі шаблону          | Створено оголошення на основі шаблону на поточний місяць                                                                     | 2          |
| Правильність вибору програмних засобів                | Правильно вказано програмні середовища для реалізації поставлених завдань                                                    | 2          |
| Уміння робити висновки                                | Зроблено висновки й обґрунтовано правильність виконання результатів розрахунків                                              | 2          |
| Усього                                                |                                                                                                                              | <b>30</b>  |

## ЗАДАЧА 2

### Інформаційний буклет

У вашому обласному центрі відкривається туристична фірма, яка планує проводити туристичні маршрути визначними місцями вашої області. Ви як представник дизайнерського центру одержали завдання розробити логотип і інформаційний буклет цієї туристичної фірми згідно вимог наведених нижче.



**Вимоги до виконання**

Логотип має містити оригінальне накреслення повної або скороченої назви фірми, а також графічні елементи, що відображають напрямок її діяльності.

*Буклет* — вид друкованої рекламної продукції, що зазвичай виконується на аркуші формату А4 з альбомним розташуванням і умовним поділом на шість сторінок. Інформація на буклеті, як правило, друкується з обох сторін.

**Таблиця 8.3.** Вимоги елементів буклета

| Елементи     | Вимоги                                                                                                                                                                                |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Сторінка 1   | Заголовок з використанням об'єкта WordArt, логотип, девіз фірми                                                                                                                       |
| Сторінки 2–5 | Реклама туристичної фірми з переліком послуг, що вона надає залежно від сезону. Опис і фотографії живописних місць, маршрутів, музеїв тощо. Таблиця з вартістю послуг, що надає фірма |
| Сторінка 6   | Фотографії працівників фірми. Контактні дані: адреса, телефон, e-mail, сайт                                                                                                           |

Створюючи буклет, не використовуйте основний розмір шрифту більше 14 пт, інтервал між рядками має бути одинарним. Не розташовуйте текстові написи на зображеннях, оскільки їх важко буде прочитати. Колір літер має бути темним, а тло — світлим. Дотримуйтеся приємної для сприйняття кольорової гами.

**ВКАЗІВКИ ДО ВИКОНАННЯ**

1. Створіть інформаційну модель розв'язання задачі у вигляді схеми, де головними темами будуть поставлені завдання, а підрозділами стануть етапи їх виконання.
2. Намалюйте в зошиті схематично зображення логотипу.
3. Оберіть середовище для створення зображення.
4. Створіть зображення логотипу.
5. Створіть схематично інформаційну модель буклета.
6. Запишіть у зошиті, які дані для розв'язання задачі вам потрібно знайти.
7. Вкажіть ключові слова пошуку потрібних відомостей.
8. Здійсніть пошук і вкажіть адреси використаних пошукових служб.
9. Вкажіть адреси сайтів, на яких знайшли потрібні дані.
10. Оберіть необхідний програмний засіб для створення буклета.
11. Створіть буклет в обраному програмному середовищі.
12. Роздрукуйте створений буклет.





13. Представте результати своєї роботи аудиторії, здійснивши демонстрацію буклета.
14. Оцініть виконану вами роботу за критеріями, представленими в таблиці 8.4.

**Таблиця 8.4.** Критерії оцінювання завдання

| Критерій                                             | Пояснення                                                                                                                               | К-ть балів |
|------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|------------|
| Створення інформаційної моделі розв'язання задачі    | Створено схему послідовності дій розв'язання задачі                                                                                     | 2          |
| Правильність вибору програмних засобів               | Правильно вказані програмні засоби для виконання завдання                                                                               | 2          |
| Уміння створювати зображення                         | При створенні зображень дотримано вимоги згідно поставленої умови                                                                       | 5          |
| Уміння створювати графічну схему                     | Створено схему майбутнього буклета                                                                                                      | 3          |
| Уміння формулювати критерії відбору даних для пошуку | Вказано правильно ключові слова для пошуку потрібних відомостей, адреси пошукових служб, адреси сайтів, на яких знайдено необхідні дані | 3          |
| Уміння виконувати форматування текстового документа  | Здійснено форматування документа згідно зазначених вимог                                                                                | 12         |
| Уміння презентувати результати своєї роботи          | Здійснено презентацію буклета відповідно до зазначених вимог та інших особливостей                                                      | 3          |
| Усього                                               |                                                                                                                                         | <b>30</b>  |

### ЗАДАЧА 3

#### Кросворд

Створіть проект реалізації електронного кросворда з будь-якого навчального предмета. При натисненні на номер біля слова в нижньому полі має з'явитися його означення, при натисненні на кнопку **Перевірка** з сітки кросворда мають бути вилучені неправильні літери. Якщо всі літери введено правильно, то з'являється текстове повідомлення з привітанням щодо завершення правильного розв'язання кросворда.

На рис. 8.1 наведено зразок подібного кросворда.



**Рис. 8.1.**

**ВКАЗІВКИ ДО ВИКОНАННЯ**

1. Створіть інформаційну модель розв'язання задачі у вигляді схеми, де головними темами будуть поставлені завдання, а підрозділами стануть етапи їх виконання.
2. Підготуйте слова та визначення для них.
3. Створіть у зошиті інформаційну модель майбутнього кросворда.
4. Оберіть необхідний програмний засіб для створення кросворда.
5. Створіть інформаційну модель виконання проекту з кросвордом у вигляді блок-схеми.
6. Створіть сітку та інші необхідні елементи кросворда в програмному середовищі.
7. Реалізуйте виконання алгоритму в обраному програмному середовищі.
8. Представте свій кросворд однокласникам, обґрунтувавши правильність реалізації кросворда.
9. Оцініть виконану вами роботу за критеріями, представленими в таблиці 8.5.

**Таблиця 8.5.** Критерії оцінювання завдання

| Критерій                                                       | Пояснення                                                                 | К-ть балів |
|----------------------------------------------------------------|---------------------------------------------------------------------------|------------|
| Створення інформаційної моделі                                 | Створено схему послідовності дій розв'язання задачі                       | 2          |
| Розуміння умови завдання                                       | Правильно визначено дані необхідні для розв'язання задачі                 | 2          |
| Уміння здійснювати пошук необхідних даних                      | Здійснено пошук слів та означень до них                                   | 2          |
| Правильність вибору програмних засобів                         | Правильно вказано програмне середовище для реалізації поставлених завдань | 2          |
| Уміння створювати алгоритм реалізації проекту                  | Розроблено блок-схему реалізації проекту                                  | 5          |
| Уміння здійснювати візуалізацію проекту                        | Створено сітку кросворду та розташовано інші елементи керування           | 5          |
| Уміння реалізувати алгоритм з використанням програмного засобу | Виконано кросворд за допомогою програмного засобу                         | 10         |
| Уміння робити висновки                                         | Зроблено висновки й обґрунтовано правильність реалізації кросворда        |            |
| Усього                                                         |                                                                           | <b>30</b>  |

## Розділ 9

# Виконання індивідуальних і групових навчальних проектів



Навчальні індивідуальні та групові проекти орієнтовані на самостійну роботу учнів — індивідуальну, парну чи групову. У процесі виконання навчальних проектів ви повинні реалізувати науково-дослідну та навчальну мету, а саме: розширити і поглибити теоретичну базу ваших знань, надати результатам практичного значення, тобто довести їх придатність до розв'язування повсякденних життєвих проблем. Під час роботи ви маєте самостійно ознайомитися з додатковою навчальною і науковою літературою, відомостями з інших джерел, зокрема з Інтернету, навчитися аналізувати й критично оцінювати їх.

Нагадаємо основні етапи роботи над проектом.

### ЕТАПИ РОБОТИ НАД КОМП'ЮТЕРНИМ ПРОЕКТОМ

1. Вибір теми.
2. Постановка мети.
3. Створення інформаційної моделі.
4. Підбір і підготовка матеріалу.
5. Вибір засобів реалізації проекту.
6. Створення програмного продукту.
7. Створення презентації проекту.
8. захист проекту.
9. Оцінювання результатів.

Далі вам буде запропоновано приклад розробки комп'ютерного проекту з використанням різноманітних програмних засобів. На кожному етапі виконання проекту не забувайте надсилати проміжні результати своєї роботи вашому вчителю інформатики на електронну пошту.

### ПРОЕКТ «ЗНАЙДИ СВОЮ МИШКУ»

Розробити проект, у якому проаналізувати різні види комп'ютерних мишок, розглянути їх основні характеристики та історію виникнення. Використовуючи інформацію будь-якого інтернет-магазину, зробити аналіз найпопулярніших моделей мишей. На основі одержаних даних створити порівняльні діаграми.

Розглянемо нижче зразок реалізації такого проекту засобами програмного середовища VB.NET.

**1. Вибір теми**

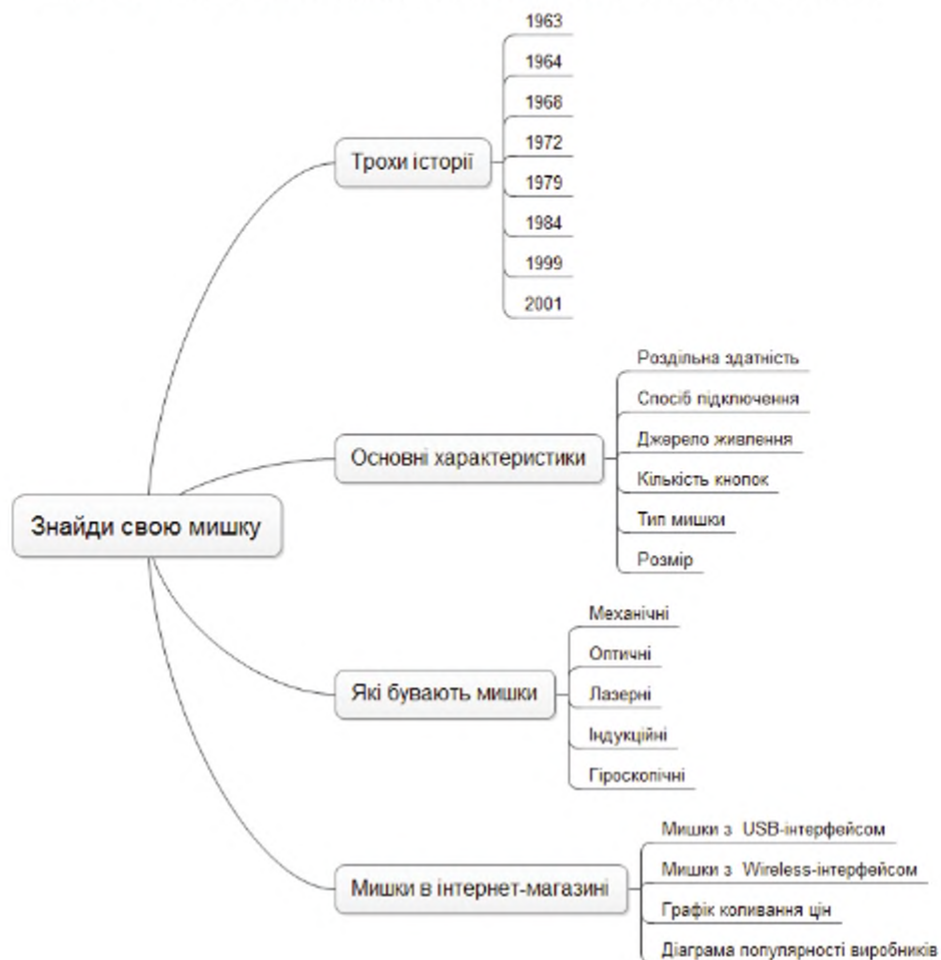
*Тема проекту.* «Знайди свою мишку».

**2. Постановка мети**

*Мета проекту.* Створити комп'ютерний проект, що містить інформацію про різні види комп'ютерних мишок, їх основні характеристики, історію виникнення, порівняльний аналіз найпопулярніших комп'ютерних мишок.

**3. Створення інформаційної моделі**

Інформаційну модель структури проекту представлено на рис. 9.1.



**Рис. 9.1.** Інформаційна модель структури проекту

**4. Підбір матеріалу**

Пошук необхідної інформації здійснено в мережі Інтернет з використанням сайтів, перелік яких представлено в табл. 9.1.

**Таблиця 9.1.** Перелік сайтів, на яких знайдено інформацію для комп'ютерного проекту

| Шукана інформація                          | Назва сайта                  |
|--------------------------------------------|------------------------------|
| Основні характеристики                     | lftexnoall.com, getoptim.com |
| Різновиди                                  | computermouse.ucoz.net       |
| Історія походження                         | vist.kharkov.ua, coob.com.ua |
| Зображення                                 | yandex.ua/images             |
| Каталог комп'ютерних мишок, що є в продажу | hard.rozetka.com.ua          |

**5.** Вибір програмних засобів реалізації проекту

Для реалізації проекту обрано програмне середовище VB.NET, у якому можна наочно представити необхідну інформацію і немає потреби встановлювати на комп'ютер спеціальне програмне забезпечення. Для порівняння основних характеристик комп'ютерних мишок, які є в продажу, обрано програму MS Excel. Обробку графічних зображень виконано засобами програми Paint.

**6.** Створення програмного продукту

Зразок проекту ви можете скачати з папки **Files** на диску. А нижче представлено послідовність дій для його створення в середовищі VB.NET.

**1)** Створіть форму за зразком (рис. 9.2). Розташуйте на формі чотири кнопки, необхідні написи та зображення. Для цієї та усіх наступних форм встановіть початкові властивості, представлені в табл. 9.2.



**Рис. 9.2.** Головне меню проекту

**Таблиця 9.2.** Початкові властивості форми

| Властивість     | Значення        |
|-----------------|-----------------|
| BackColor       | SkyBlue         |
| FormBorderStyle | FixedToolWindow |
| Size            | 600;450         |
| StartPosition   | CenterScreen    |

**2)** Створіть чотири нові форми для відображення інформації відповідно до зазначених на першій формі пунктів.



3) Задайте коди для кнопок першої форми, щоб при натисненні на них відкривалися відповідні форми. Так, для першої кнопки потрібно ввести код:

```
Me.Hide()
Form2.Show()
```

Для інших кнопок задайте коди аналогічно.

4) Виконайте оформлення другої форми за зразком (рис. 9.3). Розташуйте на ній необхідні кнопки, додайте зображення мишки і хмаринки. В середині хмаринки розташуйте елемент керування **Label** та надайте йому ім'я **L**.



Рис. 9.3. Вікно **Основні характеристики** в режимі конструювання та запуску проекту

5) Задайте програмний код, щоб при клацанні на кнопку з назвою характеристики в об'єкті **L** відображався відповідний текст, а кнопка, на яку було натиснуто, змінювала колір. Усі інші кнопки мають бути зафарбовані однаковим кольором, щоби було зрозуміло, інформацію про яку характеристику відображено в хмаринці.

Так, для кнопки **Роздільна здатність** програмний код буде мати вигляд:

**L.Text = «Кількості точок, що відповідає руху на 1 дюйм. Суть цього параметра полягає в тому, що чим більше дозвіл, тим швидше рухається курсор на екрані.»**

```
Button1.BackColor = Color.CornflowerBlue
Button2.BackColor = Color.LightSkyBlue
Button3.BackColor = Color.LightSkyBlue
Button4.BackColor = Color.LightSkyBlue
Button5.BackColor = Color.LightSkyBlue
Button6.BackColor = Color.LightSkyBlue
```

6) Виконайте оформлення третьої форми за зразком (рис. 9.4). Розташуйте на ній потрібні кнопки, додайте елемент керування **PictureBox** і надайте йому ім'я **P**.

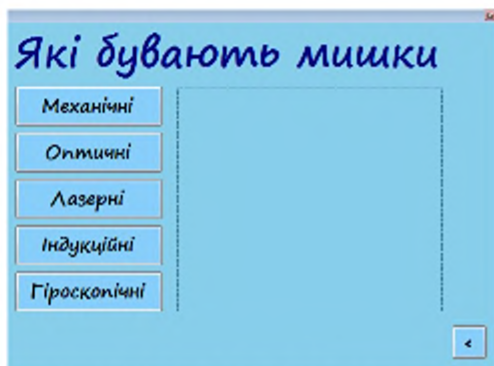


Рис. 9.4. Вікно Які бувають мишки в режимі конструювання та запуску проекту

7) Задайте програмний код, щоб при наведенні на одну з кнопок в об'єкті Р відображалася зображення відповідного типу миші.

Додайте до ресурсів проекту потрібні зображення:

- оберіть у меню **Проект** ▶ **Свойства...** ▶ **Ресурси** (Проект ▶ Властивості... ▶ Ресурси);
- в меню інструмента **Добавить ресурс** (Додати ресурс) оберіть команду **Добавить существующий файл** (Додати існуючий файл) і додайте потрібні файли зображень.

Введіть програмний код для події наведення курсору миші на кнопку:

P.Image = My.Resources.mv1

8) Створіть форми подібні до зразка на рис. 9.5 для відображення більш детальної інформації по кожному типу комп'ютерних мишей. На попередній формі задайте коди для відкриття цих форм при натисненні на відповідні кнопки.

9) Виконайте оформлення четвертої форми згідно зразка для відображення історичних відомостей (рис. 9.6). На формі розташуйте:



Рис. 9.5. Вікно Оптичні мишки в режимі конструювання



Рис. 9.6. Вікно Трохи історії в режимі конструювання та запуску проекту

- кнопки для виведення події, що відповідає обраному року, та повернення на зміст;
- два елементи **PictureBox** (в одному розташуйте зображення миші з виноскою, другий залиште порожнім та надайте йому ім'я **P**);
- п'ять елементів **Label**. У першому введіть заголовок, другий розташуйте у верхньому правому куті для виведення в ньому року (надайте йому ім'я **L**), інші три розташуйте в хмаринці для виведення інформації про історичні події (надайте їм імена **L1**, **L2**, **L3**).

**10)** Задайте програмний код, щоб при клацанні на кнопку із значенням року у верхньому правому куті відображався вказаний рік, у виноскі виводилася текстова інформація щодо події цього року, а в об'єкті **P** — відповідне зображення. Так для кнопки **1963** необхідно ввести код наведений нижче.

L.Text = «1963»

L1.Text = «Зовсім випадково з'явилася ідея створення маніпулятора на основі канадської кулі для боулінгу.»

L2.Text = «Трекбол, створений інженерами, був першим прототипом звичайної»

L3.Text = «комп'ютерної миші.»

P.Image = My.Resources.m1

Тут **m1** — ім'я файла зображення.

Для інших кнопок коди задайте аналогічно.

**11)** В одному з інтернет-магазинів оберіть 20 найпопулярніших видів мишей з USB-інтерфейсом і 20 мишей з Wireless-інтерфейсом. Дані про основні характеристики мишей занесіть до аркушів табличного процесора (рис. 9.7).

| №  | Найменування        | Виробник  | Ціна, грн | Кількість кнопок | Вага, г | Джерело живлення | Гарантія | Додаткові функції |
|----|---------------------|-----------|-----------|------------------|---------|------------------|----------|-------------------|
| 1  | GM200               | Gemix     | 160       | 6                |         | 2 x AAA          | 12       | 800/1200 dpi      |
| 2  | SL-6300-BK          | SPEEDLINK | 179       | 3                | 179     | 2 x AAA          | 12       |                   |
| 3  | NX-7005             | Genius    | 191       | 3                | 79      | 1 x AA           | 12       | 1200 dpi          |
| 4  | RX-420              | Sven      | 199       | 5                | 143     | 2 x AAA          | 12       | 1000 dpi          |
| 5  | Primo               | Trust     | 199       | 4                | 132     | 2 x AAA          | 36       |                   |
| 6  | Traveler 6000Z      | Genius    | 199       | 2                | 55      | 1 x AA           | 12       | 1000 dpi          |
| 7  | G9-500 F -1 V-Track | A4 Tech   | 249       | 3                | 187     | 1 x AA           | 12       |                   |
| 8  | Yvi Mini Mouse      | Trust     | 279       | 3                |         | 1 x AAA          | 36       |                   |
| 9  | M10                 | Rapoo     | 299       | 3                | 117     | 1 x AA           | 24       | 1000 dpi          |
| 10 | 1090p               | Rapoo     | 349       | 3                | 170     | 1 x AA           | 24       | 1000 dpi          |
| 11 | Mini Mouse M187     | Logitech  | 349       | 2                | 144     | 1 x AAA          | 36       |                   |
| 12 | M175                | Logitech  | 379       | 2                | 130     | 1 x AA           | 36       |                   |
| 13 | 3100p               | Rapoo     | 425       | 3                | 200     | 2 x AA           | 24       | 1000 dpi          |
| 14 | M185                | Logitech  | 449       | 2                | 166     | 1 x AA           | 36       |                   |
| 15 | 5G Super Mini 3300p | Rapoo     | 499       | 3                | 125     | 1 x AAA          | 24       | 1000 dpi          |
| 16 | Touch Mouse T120p   | Rapoo     | 549       | 2                | 156     | 2 x AA           | 24       | 1000 dpi          |
| 17 | 7100p               | Rapoo     | 575       | 5                | 117     | 2 x AA           | 24       | 500 / 1000 dpi    |
| 18 | T6                  | Rapoo     | 649       | 2                | 200     | 1 x AA           | 24       | 1000 dpi          |
| 19 | 3710p               | Rapoo     | 699       | 6                | 172     | 2 x AAA          | 24       | 800 / 1600dpi     |
| 20 | 3920p               | Rapoo     | 875       | 5                | 189     | 2 x AA           | 24       | 800 / 1600dpi     |

**Рис. 9.7.** Зразок таблиці 20-ти найпопулярніших мишей з Wireless-інтерфейсом



12) На основі одержаних даних побудуйте діаграми розподілу цін видів мишей та діаграму розподілу виробників з USB-інтерфейсом і Wireless-інтерфейсом (рис. 9.8).

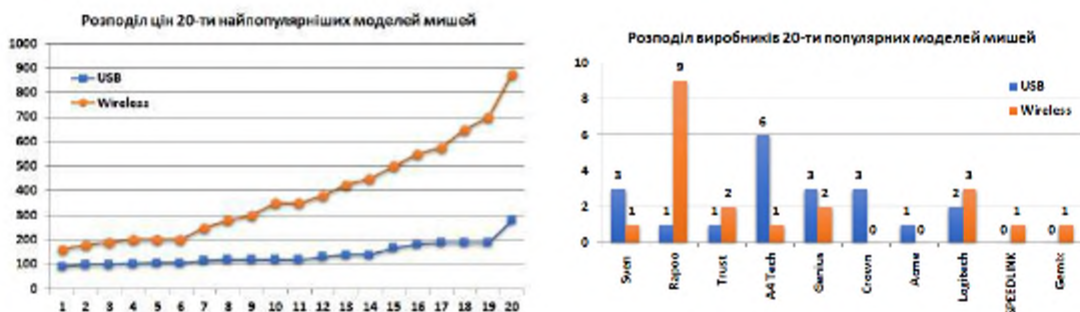


Рис. 9.8. Порівняльні діаграми

13) Розташуйте відповідні таблиці та діаграми на окремих формах проекту.

14) Виконайте оформлення п'ятої форми за зразком (рис. 9.9), розташували на ній кнопки переходу до таблиць і діаграм. Задайте програмні коди для переходу до відповідних форм.



Рис. 9.9. Вікно Мишки в Інтернет-магазині

15) Для всіх форм, де це передбачено, задайте програмний код, щоб повертатися до попередньої форми.

## 7. Створення презентації проекту

Для створення презентації проекту бажано дотримуватися наступних рекомендацій.

### РЕКОМЕНДАЦІЇ ЩОДО СТВОРЕННЯ ПРЕЗЕНТАЦІЇ

- 1) На першому слайді розташуйте назву проекту та інформацію про його виконавців (прізвище та ім'я авторів).
- 2) На другому слайді сформулюйте мету проекту.
- 3) На третьому і наступному слайдах розташуйте:



- інформаційну модель;
- інформацію щодо пошуку необхідного матеріалу (веб-сайти, знайдені зображення, перелік того, що було знайдено, тощо);
- інформацію з обґрунтуванням вибору програмних засобів;
- декілька скріншотів програмного продукту (копії екрана, які можна скопіювати до буфера обміну, натиснувши клавішу **PrtSc**, а потім вставити на слайд, клацнувши **Ctrl+V**);
- висновки (перелік дій, які було здійснено в результаті створення проекту).

### ЗАГАЛЬНІ ВИМОГИ ДО ОФОРМЛЕННЯ ПРЕЗЕНТАЦІЇ

1) Фон слайдів витриманий в одному стилі, приємний для візуального сприйняття, на ньому можна легко прочитати текст, відповідає вмісту презентації.

2) Розмір та колір шрифту зручний для сприйняття.

3) Площа, що займає текст і зображення на слайді, приблизно однакова.

### 8. Захист проекту

Нижче наведено орієнтовний план захисту проекту.

#### ПЛАН ЗАХИСТУ ПРОЕКТУ

1) Представлення авторів проекту, повідомлення теми.

2) Оголошення мети проекту.

3) Представлення етапів виконання комп'ютерного проекту:

- демонстрація інформаційної моделі;
- наведення переліку сайтів, на яких було знайдено потрібну для виконання проекту інформацію;
- представлення та аргументація програмних засобів;
- демонстрація декількох скріншотів створеного проекту.

4) Демонстрація програмного продукту.

5) Висновки:

- рекомендації щодо вибору мишей;
- відповіді на запитання.

6) Самооцінка.

### 9. Оцінювання результатів

Оцінювати проект потрібно за певними критеріями, які представлено в табл. 9.3.

**Таблиця 9.3.** Критерії оцінювання проекту

| Критерій                       | Пояснення                                                           | К-ть балів |
|--------------------------------|---------------------------------------------------------------------|------------|
| Створення інформаційної моделі | Створено схему структури проекту                                    | 5          |
| Розуміння умови завдання       | Правильно визначено дані, які потрібно знайти для створення проекту | 3          |

|                                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                               |
|---------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|
| Уміння здійснювати пошук в Інтернеті                                | Вказано адреси пошукових служб, адреси сайтів, на яких знайдено потрібні дані. Вказано правильні ключові слова для пошуку відомостей                                                                                                                                                                                                                                                                                                                                                          | 3                             |
| Правильність вибору програмних засобів                              | Правильно вказано програмне забезпечення                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 3                             |
| Уміння створювати комп'ютерний проект                               | Здійснено візуальне оформлення проекту. Правильно введено програмні коди                                                                                                                                                                                                                                                                                                                                                                                                                      | 15<br>20                      |
| Уміння систематизувати числові дані та на їх основі робити висновки | Створено таблиці з даними. Правильно обрано тип діаграм та здійснено їх побудову                                                                                                                                                                                                                                                                                                                                                                                                              | 10<br><br>10                  |
| Уміння презентувати проект                                          | Презентацію створено з дотриманням усіх вимог:<br>– презентація має чітку логічну структуру розташування слайдів;<br>– вдало підібрано елементи оформлення (шрифт, тло, графіка, слайди презентації не перевантажено матеріалом тощо);<br>– у презентації наявні всі представлені у вимогах дані (автор, тема, мета, інформаційна модель, перелік даних для пошуку, вибір програмних засобів, скріншоти програмного продукту, висновки);<br>– захист проекту виконано згідно зазначених вимог | 5<br><br>6<br><br>10<br><br>5 |
| Уміння робити висновки                                              | Зроблено висновки й обґрунтовано правильність реалізації проекту                                                                                                                                                                                                                                                                                                                                                                                                                              | 5                             |
| Усього                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 100                           |

## ДЛЯ ВИКОНАННЯ НАВЧАЛЬНОГО ПРОЕКТУ ОБЕРІТЬ ОДНУ ІЗ ЗАПРОПОНОВАНИХ ТЕМ

### 1. Шифрування інформації

Вивчити можливі способи і методи шифрування інформації. Від найпростіших прикладів — шифру Цезаря і Віженера — до найсучасніших методів відкритого шифрування, відкритих американськими математиками Вітфілдом Діффі та Мартіном Геллманом.

### 2. Методи обробки і передачі інформації

Дослідити способи передачі інформації від одного об'єкта до іншого, знайти можливі позитивні і негативні сторони того чи іншого технічного рішення.

### 3. Світ без Інтернету

Проаналізувати внесок Інтернету в наше життя, і яким би міг бути світ без Інтернету. Чи є йому альтернатива? Чому Інтернет називають унікальним винаходом?



#### 4. Інформаційне суспільство

Визначити, що ж таке інформаційне суспільство. Які його характерні риси? Зробити висновки, чи існує воно в Україні.

#### 5. Кращі інформаційні ресурси світу

Представити кращі, на ваш погляд, інформаційні ресурси світу. Обґрунтувати свою думку.

#### 6. Види інформаційних технологій

Визначити, що таке інформаційні технології, і як вони пов'язані з науково-технічним прогресом.

#### 7. Кіберзлочинність

Визначити, хто такі хакери, кіберсквоттери, спамери і т. д. Які існують способи профілактики кіберзлочинності та боротьби з нею?

#### 8. Проблема захисту інтелектуальної власності в Інтернеті

Сьогодні будь-який продукт інтелектуальної власності — музична композиція, роман, стаття тощо — розміщений в Інтернеті, можна безперешкодно вкрасти і незаконно розтиражувувати. Які ви бачите шляхи вирішення цієї проблеми?

#### 9. Чи існує штучний інтелект?

Визначити можливості сучасних комп'ютерів, перспективи їх розвитку з точки зору штучного інтелекту. Комп'ютер — це звичайний інструмент чи самостійний суб'єкт?

#### 10. Комп'ютеризація XXI століття. Перспективи розвитку

Визначити, які сфери людської діяльності ще не комп'ютеризовані, де комп'ютеризація необхідна, а де вона категорично недопустима, і чи потрібна вона взагалі.

#### 11. Клавіатура. Історія розвитку

Історія розвитку клавіатури з початку 70-х років і до наших днів. Які клавіші за що відповідають, навіщо були введені і чому клавіші, які вже не виконують тих завдань, для виконання яких були спочатку введені (наприклад, **Scroll Lock**), до сих пір не вилучені.

#### 12. Історія операційних систем для персонального комп'ютера

Порівняти сучасні ОС з давнішими, які вже не використовують. Виділити відмінні та подібні риси.

#### 13. Мобільні віруси — міф чи загроза?

Визначити, що таке мобільні віруси. Чи всім мобільним пристроям вони загрожують? Як ставитися до проблеми мобільних вірусів?

#### 14. Соціальні мережі в житті сучасних школярів

Розглянути вплив використання соціальних мереж на навчання, відпочинок і здоров'я сучасних школярів.

#### 15. Інтернет-залежність — проблема сучасності

Визначити причини виникнення інтернет-залежності та можливі шляхи її подолання.

#### 16. Видатні інформатики

Систематизувати інформацію про видатних вчених, які зробили важливий внесок у розвиток інформатики як науки.

## ПРЕДМЕТНИЙ ПОКАЖЧИК

- Автофільтр 270
- Алгоритм 201
  - з розгалуженням 201
  - з повторенням 217
- Алфавіт 21
- Арифметичні операції 189
- Аргумент 193
- Аргумент функції 247
- Архів 51
  - багатотомний 54
  - саморозпаковуваний 55
- Архіватор 51
- Аудіоряд 120
- Аудіо потокове 124
- Байт 20
- Біт 6, 20
- Бітова частота 102
- Веб-камера 30
- Величина 178
  - Integer 178
  - Long 178
  - Single 178
  - String 178
  - Boolean 178
- Випоска 89
- Відео ефект 118
- Відеозапис 103
- Відеокарта 37
- Відео потокове 124
- Відеоряд 118
- Вкладені розгалуження 209
- Властивість 135, 150
  - Name 151
  - Text 151
  - Size 151
  - AutoSize 151
  - BackColor 151
  - BorderStyle 151
  - Enabled 151
  - ForeColor 151
  - Font 151
  - Image 151
  - Location 151
  - TextAlign 151
  - Visible 151
- Гігабайт 20
- Графічний інтерфейс користувача 135
- Графічні примітиви 228
  - відрізок 229
  - дуга 230
  - еліпс 230
  - прямокутник 229
  - сектор 230
- Gif-анімація 99
- GroupBox 169
- Двійковий код 8
- Декодування 7
- Дискові накопичувачі 29
- Діаграма 263
  - з накопиченням 265
  - з областями 264
  - лінійчатга 263
  - секторна 264
  - стовпчаста 263
  - точкова 264
- Драйвери 42
- Dim 179
- Елементи керування 135, 141
  - Label 141
  - TextBox 141
  - Button 141
  - CheckBox 141
  - RadioButton 141
  - GroupBox 141
  - ListBox 141
  - ComboBox 141
  - PictureBox 141
- Захоплення відео 104
- Збірка кліпів 113
- Змінна 176
- Ініціалізація 181
- Інтерпретатор 133
- Інформаційна вага символу 21
- Inputbox 181
- Каталог 44
- Кілобайт 20
- Кількість інформації
- Клавіатура 29
- Кластер 44
- Кліп 111
- Код 12
  - Морзе 12
  - ASCII 14
  - Unicode 15
- Кодек 105
- Кодування інформації 6
- Колонка 84
- Колонки звукові 31
- Колонитигул 87
- Коментар 146
- Компілятор 133
- Комп'ютер 25
- Комп'ютерна програма 132
- Комунікаційні пристрої 26
- Конвертування 106
- Константа 179
- Ліцензія на ПЗ 47
- Логічне додавання 202

- Логічне заперечення 202
- Логічне множення 202
- Логічний вираз 201
  - простий 201
  - складений 202
- List 173
- Математичні функції 193
- Материнська плата 27
- Мегабайт 20
- Медіаконтейнер 105
- Метод 135, 153
  - Activate 153
  - CenterToScreen 153
  - Close 153
  - Hide 153
  - Show 153
- Миша 29
- Мікрофон 30
- Мова програмування 132
- Монітор 31
- Мультимедіа 97
- Набір символів 14
- Налагоджувач 133
- Обробка інформації 6
- Обробник події 137
- Оператор присвоєння 180
- Операційна система 41
- Пам'ять 25
  - внутрішня 25
  - оперативна 28
  - постійна 28
  - зовнішня 28
- Перемикач 169
- Перехід 119
- Повідомлення 6
- Подія 137
- Показчик 90, 92
- Поля 83
- Посилання 89, 239
  - відносне 240
  - абсолютне 240, 243
  - змішане 240
- Потужність алфавіту 21
- Прапорець 169
- Принтер 31
- Пристрої
  - введення 25
  - виведення 25
  - зберігання 28
- Програвач 106
- Програмне забезпечення 39
  - прикладне 40
  - системне 40,
  - інструментальне 40, 46
  - портативне 48
- Програмний код 137
- Програмний об'єкт 134
- Проект 110, 139
- Проміжні підсумки 272
- Процедура 137
  - MsgBox 156
- PictureBox 234
- Розгалуження 201
- Розділ 85
- Розрив сторінки 85
- Розширений фільтр 270
- RadioButton 169
- Символ 70
  - недрукований
- Система програмування 132
- Система числення 8
- Сканер 30
- Сортування 269
- Список 58, 173
- Стиль 77
- Стиснення даних 50
  - маркований 59
  - нумерований 59
  - багаторівневий 61
- CheckBox 16
- ComboBox 175
- Таблиця 63
- Тактова частота 26
- Текстове поле 165
  - TextBox 165
- Терабайт 20
- Титри 120
- Транслятор 133
- Умовне форматування 256
- Утиліти 42
- Флеш-пам'ять 28
- Файл 43
- Файлова система 43
- Фільтрація 270
- Форма 136
- Форма розгалуження 203
  - однорядкова 203
  - багаторядкова 205
- Форматування 44
- Формат файла 58
- Функція 246
- Центральний процесор 26
- Цикл 217
  - з лічильником 217
  - з передумовою 223
  - з післяумовою 223
- Часова шкала 111
- Шаблон документа 80
- Шифрування 7

## ЗМІСТ

|                                                                              |    |
|------------------------------------------------------------------------------|----|
| <b>РОЗДІЛ 1. Кодування даних</b>                                             | 5  |
| <b>§1</b> Роль і місце кодування в інформаційних процесах                    | 5  |
| 1.1. Зберігання, передавання та обробка інформації                           | 5  |
| 1.2. Кодування інформації                                                    | 6  |
| 1.3. Двійкове кодування                                                      | 8  |
| 1.4. Переведення числа в двійкову систему числення                           | 8  |
| 1.5. Цілі числа в комп'ютері                                                 | 8  |
| Вправа 1                                                                     | 11 |
| <b>§2</b> Кодування символів                                                 | 12 |
| 2.1. Азбука Морзе                                                            | 12 |
| 2.2. Кодування символів у комп'ютерній техніці                               | 13 |
| 2.3. Особливості кодування кирилиці                                          | 15 |
| Вправа 2                                                                     | 16 |
| <b>§3</b> Вимірювання інформації                                             | 19 |
| 3.1. Як виміряти інформацію                                                  | 19 |
| 3.2. Змістовий підхід                                                        | 19 |
| 3.3. Алфавітний підхід                                                       | 21 |
| <b>Практична робота №1</b>                                                   |    |
| Розв'язування задач на визначення довжини двійкового коду даних різних типів | 24 |
| <b>РОЗДІЛ 2. Апаратно-програмне забезпечення комп'ютера</b>                  | 25 |
| <b>§4</b> Апаратне забезпечення                                              | 25 |
| 4.1. Будова комп'ютера                                                       | 25 |
| 4.2. Центральний процесор                                                    | 26 |
| 4.3. Материнська плата                                                       | 27 |
| 4.4. Внутрішня пам'ять                                                       | 27 |
| 4.5. Зовнішня пам'ять                                                        | 28 |
| 4.6. Введення і виведення інформації                                         | 29 |
| 4.7. Мультимедійне обладнання                                                | 31 |
| 4.8. Від абака до суперкомп'ютера                                            | 32 |
| Вправа 3                                                                     | 33 |
| <b>§5</b> Вибір складових комп'ютера під потребу                             | 34 |
| 5.1. Позначення комплектуючих системного блока комп'ютера                    | 34 |
| 5.2. Огляд комп'ютерних комплектуючих                                        | 35 |
| <b>Практична робота №2</b>                                                   |    |
| Конфігурація комп'ютера під потребу                                          | 38 |
| <b>§6</b> Програмне забезпечення                                             | 39 |
| 6.1. Роль програмного забезпечення, його класифікація                        | 39 |
| 6.2. Прикладне програмне забезпечення                                        | 40 |
| 6.3. Системне програмне забезпечення                                         | 41 |
| 6.4. Складові операційної системи                                            | 42 |
| 6.5. Файлова система                                                         | 43 |
| 6.6. Форматування носіїв даних                                               | 44 |
| <b>§7</b> Розробка і розповсюдження програмного забезпечення                 | 46 |
| 7.1. Інструментальне програмне забезпечення                                  | 46 |
| 7.2. Розповсюдження програмного забезпечення                                 | 46 |
| 7.3. Інсталяція та деінсталяція програмного забезпечення                     | 47 |
| 7.4. Сумісність програмного забезпечення                                     | 48 |
| Вправа 4                                                                     | 49 |

|                                                   |                                                                      |    |
|---------------------------------------------------|----------------------------------------------------------------------|----|
| <b>§8</b>                                         | Стиснення та архівування даних.....                                  | 50 |
|                                                   | 8.1. Стиснення даних.....                                            | 50 |
|                                                   | 8.2. Програми-архіватори.....                                        | 51 |
|                                                   | 8.3. Архівування файлів і папок у Windows.....                       | 52 |
|                                                   | 8.4. Програма-архіватор WinRAR.....                                  | 52 |
|                                                   | 8.5. Архівування файлів.....                                         | 53 |
|                                                   | 8.6. Багатотомний архів.....                                         | 54 |
|                                                   | 8.7. Саморозпаковуваний архів.....                                   | 54 |
| <b>Практична робота №3</b>                        |                                                                      |    |
|                                                   | Архівування та розархівування даних.....                             | 56 |
| <b>РОЗДІЛ 3. Опрацювання текстових даних.....</b> |                                                                      |    |
| <b>§9</b>                                         | Списки в текстових документах.....                                   | 57 |
|                                                   | 9.1. Повторення.....                                                 | 57 |
|                                                   | 9.2. Формати файлів текстових документів.....                        | 58 |
|                                                   | 9.3. Види списків.....                                               | 58 |
|                                                   | 9.4. Створення списків.....                                          | 59 |
|                                                   | 9.5. Зміна зовнішнього вигляду списків.....                          | 59 |
|                                                   | 9.6. Багаторівневі списки.....                                       | 60 |
|                                                   | Вправа 5.....                                                        | 61 |
| <b>§10</b>                                        | Таблиці в текстових документах.....                                  | 63 |
|                                                   | 10.1. Основні елементи таблиці.....                                  | 63 |
|                                                   | 10.2. Створення таблиць.....                                         | 63 |
|                                                   | 10.3. Виділення елементів таблиці.....                               | 65 |
|                                                   | 10.4. Введення та редагування тексту.....                            | 65 |
|                                                   | 10.5. Форматування таблиць.....                                      | 66 |
|                                                   | 10.6. Вставлення та видалення рядків і стовпців.....                 | 66 |
|                                                   | 10.7. Змінення ширини стовпців і висоти рядків.....                  | 67 |
|                                                   | 10.8. Об'єднання та поділ клітинок.....                              | 67 |
|                                                   | Вправа 6.....                                                        | 68 |
|                                                   | Вправа 7.....                                                        | 68 |
| <b>§11</b>                                        | Символи, формули, графічні об'єкти в текстовому документі.....       | 70 |
|                                                   | 11.1. Вставлення символів.....                                       | 70 |
|                                                   | Вправа 8. Самостійно.....                                            | 70 |
|                                                   | 11.2. Вставлення формул.....                                         | 71 |
|                                                   | Вправа 9.....                                                        | 71 |
|                                                   | 11.3. Вставлення графічних об'єктів.....                             | 72 |
|                                                   | Вправа 10.....                                                       | 72 |
| <b>Практична робота №4</b>                        |                                                                      |    |
|                                                   | Створення текстового документа, що містить об'єкти різних типів..... | 76 |
| <b>§12</b>                                        | Використання стилів і шаблонів документів.....                       | 76 |
|                                                   | 12.1. Стилї форматування.....                                        | 76 |
|                                                   | 12.2. Типи стилів.....                                               | 77 |
|                                                   | 12.3. Перегляд інформації про стилі.....                             | 77 |
|                                                   | 12.4. Застосування стилів.....                                       | 78 |
|                                                   | 12.5. Створення і зміна стилів.....                                  | 78 |
|                                                   | Вправа 11.....                                                       | 79 |
|                                                   | 12.6. Шаблони документів.....                                        | 80 |
|                                                   | 12.7. Створення документа на основі шаблону.....                     | 80 |
|                                                   | 12.8. Створення шаблону.....                                         | 81 |
|                                                   | Вправа 12.....                                                       | 81 |



|                                                         |                                                                                                         |     |
|---------------------------------------------------------|---------------------------------------------------------------------------------------------------------|-----|
| <b>§13</b>                                              | Верстка сторінок документа. Створення змісту і покажчика.....                                           | 83  |
|                                                         | 13.1. Настроювання параметрів сторінки.....                                                             | 83  |
|                                                         | 13.2. Створення колонок.....                                                                            | 84  |
|                                                         | Вправа 13. <i>Самостійно</i> .....                                                                      | 85  |
|                                                         | 13.3. Верстка сторінок.....                                                                             | 85  |
|                                                         | 13.4. Нумерація сторінок.....                                                                           | 86  |
|                                                         | 13.5. Оформлення колонтитулів.....                                                                      | 87  |
|                                                         | Вправа 14. <i>Самостійно</i> .....                                                                      | 89  |
|                                                         | 13.6. Створення посилань.....                                                                           | 89  |
|                                                         | 13.7. Створення структури документа.....                                                                | 90  |
|                                                         | Вправа 15.....                                                                                          | 90  |
|                                                         | 13.8. Автоматичне створення змісту.....                                                                 | 91  |
|                                                         | 13.9. Створення покажчиків.....                                                                         | 92  |
|                                                         | Вправа 16.....                                                                                          | 92  |
| <b>§14</b>                                              | Створення складних документів.....                                                                      | 94  |
|                                                         | 14.1. Робота з кількома документами.....                                                                | 94  |
|                                                         | 14.2. Алгоритми роботи над складним документом.....                                                     | 94  |
| <b>Практична робота №5</b>                              |                                                                                                         |     |
|                                                         | Структура документа. Автоматизоване створення змісту та покажчиків.....                                 | 95  |
| <b>РОЗДІЛ 4. Опрацювання об'єктів мультимедіа</b> ..... |                                                                                                         |     |
|                                                         |                                                                                                         | 97  |
| <b>§15</b>                                              | Мультимедіа як ознака сучасного інформаційного світу.....                                               | 97  |
|                                                         | 15.1. Поняття мультимедіа.....                                                                          | 97  |
|                                                         | 15.2. Застосування мультимедіа.....                                                                     | 98  |
|                                                         | 15.3. Складові мультимедіа.....                                                                         | 98  |
|                                                         | 15.4. Створення GIF-анімації.....                                                                       | 99  |
|                                                         | Вправа 17.....                                                                                          | 100 |
| <b>§16</b>                                              | Зберігання і відтворення мультимедійних даних.....                                                      | 102 |
|                                                         | 16.1. Аудіодані та аудіоформати.....                                                                    | 102 |
|                                                         | 16.2. Відеодані та відеоформати.....                                                                    | 103 |
|                                                         | 16.3. Кодеки.....                                                                                       | 105 |
|                                                         | 16.4. Медіаконтейнери.....                                                                              | 105 |
|                                                         | 16.5. Конвертування відео та аудіо.....                                                                 | 106 |
|                                                         | 16.6. Мультимедійні програвачі.....                                                                     | 106 |
|                                                         | Вправа 18.....                                                                                          | 107 |
| <b>§17</b>                                              | Основи обробки мультимедійних даних.....                                                                | 109 |
|                                                         | 17.1. Огляд програми Кіностудія Windows.....                                                            | 109 |
|                                                         | 17.2. Документи й об'єкти Кіностудія Windows.....                                                       | 110 |
|                                                         | 17.3. Інтерфейс програми Windows Movie Maker.....                                                       | 111 |
|                                                         | 17.4. Початок роботи над фільмом.....                                                                   | 112 |
|                                                         | 17.5. Основи монтажу фільму.....                                                                        | 113 |
|                                                         | Вправа 19.....                                                                                          | 116 |
| <b>§18</b>                                              | Редагування мультимедійних даних.....                                                                   | 118 |
|                                                         | 18.1. Оформлення відеоряду.....                                                                         | 118 |
|                                                         | 18.2. Керування аудіорядом.....                                                                         | 120 |
| <b>Практична робота №6</b>                              |                                                                                                         |     |
|                                                         | Створення відеокліпу. Додавання відео ефектів, налаштування часових параметрів аудіо- та відеоряду..... | 123 |
| <b>§19</b>                                              | Мультимедіа у Вебі.....                                                                                 | 124 |
|                                                         | 19.1. Роль і місце мультимедіа у Вебі.....                                                              | 124 |
|                                                         | 19.2. Потоків аудіо та відео.....                                                                       | 124 |
|                                                         | 19.3. Розміщення анімації, відео та аудіо на веб-сторінках.....                                         | 124 |

|                                                                               |                                                                                  |     |
|-------------------------------------------------------------------------------|----------------------------------------------------------------------------------|-----|
|                                                                               | Вправа 20 .....                                                                  | 125 |
| 19.4.                                                                         | Онлайнні конвертори відео та аудіо, GIF-редактори .....                          | 126 |
|                                                                               | Вправа 21. <i>Самостійно</i> .....                                               | 126 |
| 19.5.                                                                         | Робота з інтернет-джерелами мультимедійного контенту .....                       | 126 |
|                                                                               | Вправа 22. <i>Самостійно</i> .....                                               | 127 |
| <b>§20</b>                                                                    | Розміщення аудіо- та відеофайлів в Інтернеті .....                               | 128 |
|                                                                               | 20.1. Платформи для розміщення фотографій та відео .....                         | 128 |
|                                                                               | 20.2. Відеохостинг YouTube .....                                                 | 129 |
| <b>Практична робота №7</b>                                                    |                                                                                  |     |
|                                                                               | Розміщення аудіо- та відеоматеріалів в Інтернеті .....                           | 130 |
| <b>РОЗДІЛ 5. Основи подійно-та об'єктно-орієнтованого програмування</b> ..... |                                                                                  |     |
| 131                                                                           |                                                                                  |     |
| <b>§21</b>                                                                    | Мова програмування .....                                                         | 131 |
|                                                                               | 21.1. Поняття мови програмування .....                                           | 131 |
|                                                                               | 21.2. Система програмування .....                                                | 132 |
| <b>§22</b>                                                                    | Основні поняття об'єктно-орієнтованого програмування .....                       | 134 |
|                                                                               | 22.1. Об'єкти. Властивості і методи об'єктів .....                               | 134 |
|                                                                               | 22.2. Графічний інтерфейс користувача .....                                      | 135 |
|                                                                               | 22.3. Події, обробники подій .....                                               | 136 |
| <b>§23</b>                                                                    | Ознайомлення із середовищем програмування .....                                  | 138 |
|                                                                               | 23.1. Початок роботи в середовищі .....                                          | 138 |
|                                                                               | 23.2. Структура головного вікна .....                                            | 140 |
|                                                                               | 23.3. Вікно форми .....                                                          | 141 |
|                                                                               | 23.4. Панель елементів керування .....                                           | 141 |
|                                                                               | 23.5. Вікно властивостей .....                                                   | 142 |
|                                                                               | Вправа 23 .....                                                                  | 142 |
|                                                                               | 23.6. Вікно складових проекту .....                                              | 142 |
| <b>§24</b>                                                                    | Створення найпростішого проекту .....                                            | 144 |
|                                                                               | 24.1. Вікно програмного коду .....                                               | 144 |
|                                                                               | 24.2. Зміна властивостей об'єкта .....                                           | 145 |
|                                                                               | 24.3. Виконання програми .....                                                   | 146 |
|                                                                               | 24.4. Редагування коду обробника подій .....                                     | 146 |
|                                                                               | 24.5. Збереження проекту .....                                                   | 147 |
|                                                                               | Вправа 24 .....                                                                  | 148 |
| <b>§25</b>                                                                    | Властивості та методи елементів керування .....                                  | 150 |
|                                                                               | 25.1. Найбільш уживані властивості елементів керування .....                     | 151 |
|                                                                               | Вправа 25 .....                                                                  | 152 |
|                                                                               | 25.2. Поняття методу .....                                                       | 153 |
|                                                                               | Вправа 26 .....                                                                  | 154 |
| <b>§26</b>                                                                    | Використання вікон повідомлень .....                                             | 156 |
|                                                                               | 26.1. Синтаксис процедури .....                                                  | 156 |
|                                                                               | 26.2. Константи та значення процедури MsgBox .....                               | 157 |
|                                                                               | Вправа 27 .....                                                                  | 157 |
| <b>Практична робота №8</b>                                                    |                                                                                  |     |
|                                                                               | Створення об'єктно-орієнтованої програми, що відображає вікно повідомлення ..... | 158 |
| <b>§27</b>                                                                    | Елементи керування «напис» і «кнопка» .....                                      | 160 |
|                                                                               | Вправа 28 .....                                                                  | 160 |
|                                                                               | Вправа 29 .....                                                                  | 161 |
| <b>Практична робота №9</b>                                                    |                                                                                  |     |
|                                                                               | Створення програми з кнопками та написами .....                                  | 163 |

|                             |                                                                                                       |     |
|-----------------------------|-------------------------------------------------------------------------------------------------------|-----|
|                             | <b>РОЗДІЛ 6. Алгоритми роботи з об'єктами та величинами</b>                                           | 165 |
| <b>§28</b>                  | Елементи для введення даних. Текстове поле                                                            | 165 |
|                             | 28.1. Властивості текстового поля                                                                     | 165 |
|                             | 28.2. Встановлення властивостей в програмному коді                                                    | 166 |
|                             | Вправа 30                                                                                             | 166 |
|                             | Вправа 31                                                                                             | 167 |
|                             | Вправа 32. <i>Самостійно</i>                                                                          | 167 |
| <b>§29</b>                  | Використання перемикачів і прапорців                                                                  | 169 |
|                             | 29.1. Використання перемикачів                                                                        | 169 |
|                             | Вправа 33                                                                                             | 169 |
|                             | 29.2. Використання прапорців                                                                          | 170 |
|                             | Вправа 34                                                                                             | 171 |
| <b>§30</b>                  | Списки                                                                                                | 173 |
|                             | 30.1. Елемент керування ListBox                                                                       | 173 |
|                             | Вправа 35                                                                                             | 174 |
|                             | 30.2. Елемент керування ComboBox                                                                      | 175 |
|                             | Вправа 36                                                                                             | 175 |
| <b>§31</b>                  | Типи величин                                                                                          | 178 |
|                             | 31.1. Основні поняття мови програмування                                                              | 178 |
|                             | 31.2. Поняття величини                                                                                | 178 |
|                             | 31.3. Оголошення змінних у програмі                                                                   | 179 |
| <b>§32</b>                  | Присвоєння значень величинам                                                                          | 180 |
|                             | 32.1. Оператор присвоєння                                                                             | 180 |
|                             | 32.2. Введення даних засобами діалогового вікна                                                       | 181 |
|                             | Вправа 37                                                                                             | 182 |
|                             | Вправа 38                                                                                             | 183 |
| <b>§33</b>                  | Співвідношення типів даних та елементів для введення даних                                            | 184 |
|                             | 33.1. Введення та виведення числових і текстових величин                                              | 184 |
|                             | 33.2. Введення та виведення логічних величин                                                          | 186 |
|                             | Вправа 39                                                                                             | 186 |
| <b>§34</b>                  | Числові величини                                                                                      | 188 |
|                             | 34.1. Арифметичні операції                                                                            | 188 |
|                             | Вправа 40                                                                                             | 189 |
|                             | 34.2. Задачі на цілочисельне ділення                                                                  | 190 |
|                             | Вправа 41                                                                                             | 191 |
|                             | Вправа 42. <i>Самостійно</i>                                                                          | 192 |
| <b>§35</b>                  | Стандартні математичні функції                                                                        | 193 |
|                             | Вправа 43                                                                                             | 194 |
| <b>§36</b>                  | Величини текстового типу                                                                              | 196 |
|                             | Вправа 44                                                                                             | 197 |
|                             | Вправа 45                                                                                             | 198 |
|                             | Вправа 46                                                                                             | 198 |
| <b>Практична робота №10</b> |                                                                                                       |     |
|                             | Складання та виконання лінійних алгоритмів опрацювання величин в навчальному середовищі програмування | 200 |
| <b>§37</b>                  | Величини логічного типу. Програми з розгалуженням                                                     | 201 |
|                             | 37.1. Логічні вирази                                                                                  | 201 |
|                             | 37.2. Команди розгалуження                                                                            | 203 |
|                             | Вправа 47                                                                                             | 203 |
|                             | Вправа 48                                                                                             | 204 |
| <b>§38</b>                  | Алгоритми з розгалуженнями для опрацювання величин                                                    | 205 |

|                                                                                              |                                                                                                |            |
|----------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|------------|
|                                                                                              | Вправа 49.....                                                                                 | 206        |
|                                                                                              | Вправа 50.....                                                                                 | 207        |
|                                                                                              | Вправа 51.....                                                                                 | 208        |
| <b>§39</b>                                                                                   | <b>Вкладені розгалуження.....</b>                                                              | <b>209</b> |
|                                                                                              | Вправа 52.....                                                                                 | 209        |
|                                                                                              | Вправа 53.....                                                                                 | 210        |
| <b>§40</b>                                                                                   | <b>Складання алгоритмів опрацювання величин, їх налагодження і виконання.....</b>              | <b>211</b> |
| <b>Практична робота №11</b>                                                                  |                                                                                                |            |
|                                                                                              | Налагодження готової програми.....                                                             | 214        |
| <b>§41</b>                                                                                   | <b>Алгоритми з повтореннями. Цикл з лічильником.....</b>                                       | <b>217</b> |
|                                                                                              | <b>41.1. Цикл з параметром.....</b>                                                            | <b>217</b> |
|                                                                                              | Вправа 54.....                                                                                 | 218        |
|                                                                                              | <b>41.2. Знаходження суми, добутку, кількості та середнього арифметичного.....</b>             | <b>219</b> |
|                                                                                              | Вправа 55.....                                                                                 | 219        |
|                                                                                              | Вправа 56. <i>Самостійно</i> .....                                                             | 221        |
|                                                                                              | <b>41.3. Створення таблиць значень функцій.....</b>                                            | <b>221</b> |
|                                                                                              | Вправа 57.....                                                                                 | 221        |
| <b>§42</b>                                                                                   | <b>Цикли з умовами.....</b>                                                                    | <b>223</b> |
|                                                                                              | Вправа 58.....                                                                                 | 224        |
|                                                                                              | Вправа 59.....                                                                                 | 224        |
|                                                                                              | Вправа 60.....                                                                                 | 225        |
| <b>Практична робота №12</b>                                                                  |                                                                                                |            |
|                                                                                              | Складання і виконання алгоритмів з повтореннями та розгалуженнями для опрацювання величин..... | 227        |
| <b>§43</b>                                                                                   | <b>Створення графічних зображень.....</b>                                                      | <b>228</b> |
|                                                                                              | Вправа 61.....                                                                                 | 230        |
| <b>§44</b>                                                                                   | <b>Побудова зображень з використанням команди повторення.....</b>                              | <b>232</b> |
| <b>Практична робота №13</b>                                                                  |                                                                                                |            |
|                                                                                              | Складання та виконання алгоритмів з графічним відображенням даних.....                         | 233        |
| <b>§45</b>                                                                                   | <b>Відображення рисунків із зовнішніх файлів.....</b>                                          | <b>234</b> |
|                                                                                              | Вправа 62.....                                                                                 | 235        |
|                                                                                              | Вправа 63.....                                                                                 | 236        |
| <b>РОЗДІЛ 7. Технології опрацювання числових даних у середовищі табличного процесора</b> 239 |                                                                                                |            |
| <b>§46</b>                                                                                   | <b>Абсолютні, відносні й мішані посилання.....</b>                                             | <b>239</b> |
|                                                                                              | <b>46.1. Повторення.....</b>                                                                   | <b>239</b> |
|                                                                                              | <b>46.2. Посилання на адреси клітинок.....</b>                                                 | <b>239</b> |
|                                                                                              | Вправа 64.....                                                                                 | 241        |
|                                                                                              | Вправа 65.....                                                                                 | 241        |
| <b>§47</b>                                                                                   | <b>Імена клітинок та їх діапазонів. Розв'язування задач на обчислення.....</b>                 | <b>242</b> |
|                                                                                              | Вправа 66.....                                                                                 | 244        |
|                                                                                              | Вправа 67.....                                                                                 | 244        |
| <b>Практична робота №14</b>                                                                  |                                                                                                |            |
|                                                                                              | Розв'язування задач на обчислення.....                                                         | 246        |
| <b>§48</b>                                                                                   | <b>Математичні та статистичні функції.....</b>                                                 | <b>246</b> |
|                                                                                              | <b>48.1. Математичні функції.....</b>                                                          | <b>247</b> |
|                                                                                              | <b>48.2. Статистичні функції.....</b>                                                          | <b>248</b> |

|                                                                                |                                                                                                           |     |
|--------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|-----|
| 48.3.                                                                          | Використання майстра функцій .....                                                                        | 248 |
| 48.4.                                                                          | Введення функцій з клавіатури .....                                                                       | 249 |
|                                                                                | Вправа 68 .....                                                                                           | 250 |
|                                                                                | Вправа 69 .....                                                                                           | 250 |
|                                                                                | Вправа 70 .....                                                                                           | 251 |
| <b>§49</b>                                                                     | Логічні функції .....                                                                                     | 252 |
| 49.1.                                                                          | Використання логічних функцій .....                                                                       | 252 |
|                                                                                | Вправа 71 .....                                                                                           | 253 |
|                                                                                | Вправа 72 .....                                                                                           | 254 |
| 49.2.                                                                          | Вкладені функції .....                                                                                    | 254 |
|                                                                                | Вправа 73 .....                                                                                           | 255 |
| <b>§50</b>                                                                     | Умове форматування .....                                                                                  | 256 |
| 50.1.                                                                          | Використання кольорової шкали .....                                                                       | 256 |
|                                                                                | Вправа 74 .....                                                                                           | 257 |
| 50.2.                                                                          | Використання правил виділених клітинок .....                                                              | 257 |
|                                                                                | Вправа 75 .....                                                                                           | 258 |
| 50.3.                                                                          | Використання гістограм .....                                                                              | 259 |
| 50.4.                                                                          | Створення власного правила з використанням формул .....                                                   | 259 |
|                                                                                | Вправа 76 .....                                                                                           | 259 |
| <b>Практична робота №15</b>                                                    |                                                                                                           |     |
|                                                                                | Використання математичних, логічних і статистичних функцій табличного процесора. Умове форматування ..... | 261 |
| <b>§51</b>                                                                     | Створення та налагодження діаграм різного типу .....                                                      | 263 |
| 51.1.                                                                          | Основні типи діаграм .....                                                                                | 263 |
| 51.2.                                                                          | Вибір типу діаграми .....                                                                                 | 265 |
|                                                                                | Вправа 77 .....                                                                                           | 265 |
|                                                                                | Вправа 78 .....                                                                                           | 266 |
| <b>§52</b>                                                                     | Упорядкування даних у таблицях. Автоматичні та розширені фільтри.                                         |     |
|                                                                                | Проміжні підсумки .....                                                                                   | 268 |
| 52.1.                                                                          | Упорядкування даних у таблицях .....                                                                      | 268 |
| 52.2.                                                                          | Фільтрація даних .....                                                                                    | 270 |
| 52.3.                                                                          | Проміжні підсумки .....                                                                                   | 272 |
|                                                                                | Вправа 79 .....                                                                                           | 272 |
| <b>§53</b>                                                                     | Встановлення параметрів сторінки. Друкування електронної таблиці .....                                    | 275 |
| 53.1.                                                                          | Встановлення параметрів сторінки .....                                                                    | 275 |
| 53.2.                                                                          | Налаштування друку .....                                                                                  | 276 |
| <b>Практична робота №16</b>                                                    |                                                                                                           |     |
|                                                                                | Упорядкування даних у таблицях. Автоматичні та розширені фільтри .....                                    | 277 |
| <b>РОЗДІЛ 8. Розв'язування компетентнісних задач</b> .....                     |                                                                                                           |     |
|                                                                                | <b>Задача 1.</b> Комунальні послуги .....                                                                 | 279 |
|                                                                                | <b>Задача 2.</b> Інформаційний буклет .....                                                               | 281 |
|                                                                                | <b>Задача 3.</b> Кросворд .....                                                                           | 283 |
| <b>РОЗДІЛ 9. Виконання індивідуальних і групових навчальних проєктів</b> ..... |                                                                                                           |     |
|                                                                                | Етапи роботи над комп'ютерним проєктом .....                                                              | 285 |
|                                                                                | Проєкт «Знайди свою мишку» .....                                                                          | 285 |
|                                                                                | Рекомендації щодо створення презентації .....                                                             | 291 |
|                                                                                | Загальні вимоги до оформлення презентації .....                                                           | 292 |
|                                                                                | План захисту проєкту .....                                                                                | 292 |
|                                                                                | Для виконання навчального проєкту оберіть одну із запропонованих тем .....                                | 293 |
|                                                                                | Предметний покажчик .....                                                                                 | 295 |

*Навчальне видання*

**КАЗАНЦЕВА Ольга Павлівна**  
**СТЕЦЕНКО Ірина Володимирівна**

## **ІНФОРМАТИКА**

**Підручник для 8 класу**  
**загальноосвітніх навчальних закладів**

Головний редактор *Богдан Будний*  
Редактор *Вікторія Дячук*  
Обкладинка *Ростислава Крамара*  
Комп'ютерна верстка *Зоряни Сидор*  
Художній редактор *Ростислав Крамар*  
Технічний редактор *Оксана Чучук*

Підписано до друку \_\_\_\_\_. Формат 70×100/16. Папір офсетний. Гарнітура CentSchbook Win95BT.  
Друк офсетний. Умовн. друк арк. \_\_\_\_\_. Умовн. фарбо-відб. \_\_\_\_\_.

Видавництво «Навчальна книга – Богдан»

Свідоцтво про внесення суб'єкта видавничої справи до Державного  
реєстру видавців, виготівників і розповсюджувачів видавничої продукції  
ДК № 4221 від 07.12.2011 р.

Навчальна книга – Богдан, просп. С. Бандери, 34а, м. Тернопіль, 46002  
Навчальна книга – Богдан, а/с 529, м. Тернопіль, 46008  
тел./факс (0352)52-06-07; 52-19-66; 52-05-48  
[office@bohdan-books.com](mailto:office@bohdan-books.com) [www.bohdan-books.com](http://www.bohdan-books.com)