

# ИНФОРМАТИКА

Учебник для 7 класса  
общеобразовательных учебных заведений  
с обучением на русском языке

*Рекомендовано Министерством освіти і науки України*

Львів  
Видавництво «Світ»  
2015

УДК 004(075.3)  
ББК 32.973-018я721.6  
І-74

Авторський колектив:

*А. М. Гуржій, Л. А. Карташова, В. В. Лапінський, В. Д. Руденко*

*Рекомендовано Міністерством освіти і науки України  
(наказ МОН України від 20.07.2015 р. № 777)*

**Видано за рахунок державних коштів. Продаж заборонено**

**Інформатика** : підруч. для 7 кл. загальноосв.  
І-74 навч. закл. з навч. російською мовою / А.М. Гуржій,  
Л.А. Карташова, В.В. Лапінський, В.Д. Руденко;  
пер. В.В. Лапінський, Л.А. Карташова. – Львів :  
Світ, 2015. – 176 с. : іл., табл.

ISBN 978-966-603-948-7

Підручник призначений для навчання інформатики у сьомому класі загальноосвітніх навчальних закладів. Зміст підручника повністю відповідає навчальній програмі «Інформатика. 5–9 класи», рекомендованій Міністерством освіти і науки України (наказ від 06.06.2012 р. № 664).

УДК 004(075.3)  
ББК 32.973-018я721.6

ISBN 978-966-603-948-7

© Гуржій А.М., Карташова Л.А.,  
Лапінський В.В., Руденко В.Д., 2015  
© Лапінський В.В., Карташова Л.А.,  
переклад, 2015  
© Видавництво «Світ», 2015

## *Дорогие ученики!*

Вы уже не первый год изучаете информатику, в повседневной жизни используете средства информационных технологий (мобильный телефон, планшет, другие современные гаджеты).

Вы учитесь пользоваться ими, не только читая инструкцию для пользователя (нынче ко многим аппаратным средствам дают только адрес в сети Интернет, по которому ее можно найти).

Вы (и мы также, потому что нет никого, кто знал бы и умел абсолютно все!) спрашиваете у друзей, как и что делать, вспоминаете собственный опыт использования подобных гаджетов.

Информационные технологии в настоящее время развиваются настолько быстро, что их технические средства обновляются почти ежедневно. Двадцать лет назад, покупая телевизор, радиоприемник, телефон, мы надеялись, что приобретенная вещь будет служить нам лет 10 и больше – и так оно и было. А теперь представьте себя с мобильным телефоном, выпущенным пять или шесть лет назад...

Развитие технологий настолько стремительно, что некоторые вещи перестают удовлетворять потребности пользователей уже через год-два после приобретения, оставаясь полностью исправными.

Однако в области информационных технологий, как и во всех областях науки, техники, искусства, есть знания, которые не устаревают. Поэтому их называют основами наук. Овладеть основами определенной науки необходимо для того, чтобы стало возможным дальнейшее развитие, дальнейшее обучение, освоение новых средств деятельности.

В этом году вы будете продолжать изучение информатики. Часть учебного материала будет для вас новым, часть – казаться знакомым. Но имейте в виду: то, что вам кажется знакомым и недостойным внимательного изучения, может содержать очень важные знания, которые станут частью вашей профессиональной компетентности в будущем.

Вы будете на практике усваивать правила безопасной жизнедеятельности при выполнении работ с использованием компьютера и других средств информационных технологий.









*Здоровья и успехов вам в учебе и в жизни!*  
*Авторы*

## КАК РАБОТАТЬ С ЭТОЙ КНИГОЙ


Для обучения по этому учебнику обязательно в вашем распоряжении наличие персонального компьютера. Желательно также подключить к компьютеру устройства, которые описаны в учебнике. На нем нужно установить комплект программных средств, описанных в учебнике.

Учебный материал представлен в четырех основных разделах: «Электронная переписка», «Моделирование», «Алгоритмы с повторением и разветвлением», «Обработка данных в таблицах» и двух дополнительных: «Компетентностные задачи» и «Задачи для учебных проектов». Каждый раздел содержит несколько пунктов. Новые термины в пунктах отмечены **жирным шрифтом**.

В тексте применяются такие обозначения и подзаголовки:

	«Это вы уже знаете» – краткое изложение знаний, необходимых для усвоения материала раздела или пункта
	«Что изучать» – под этим заголовком в начале каждого раздела и пункта приведен краткий перечень того, чему вы научитесь, усвоив материал раздела
	«Важное положение. Желательно запомнить»
	«Обратите особое внимание»
	«Для любознательных» – дополнительные сведения
	«Словарь» – толкование терминов, используемых в разделе
	Рекомендуется выполнить (обсудить) в коллективе
	Рекомендуется выполнить дома

Уровни сложности заданий и вопросов обозначены следующим образом:

	первый
	второй
	третий

## РАЗДЕЛ 1. ЭЛЕКТРОННАЯ ПЕРЕПИСКА



Компьютерные сети и их назначение. Понятие о глобальной сети Интернет. Основные службы Интернета. Поиск сведений в Интернете. Сохранение изображений, веб-страниц и их фрагментов. Интернет-энциклопедии, словари и онлайн-переводчики. Авторское право и Интернет.



Почтовая служба Интернета. Электронный почтовый ящик и электронная переписка. Адрес почтового ящика. Создание электронного почтового ящика. Передача, прием, перенаправление сообщений. Операции с папками и письмами. Вложение файлов. Использование адресной книги и списков рассылки. Этикет электронной переписки. Правила безопасного пользования электронным ящиком.

### 1.1. Электронная переписка. Электронный почтовый ящик

Одной из первых услуг после появления общедоступных компьютерных сетей была **электронная почта**, или e-mail (англ.: *electronic mail* – электронная почта). Эта услуга обеспечивала практически мгновенную передачу сообщений, причем она осуществлялась «из рук в руки». Автору письма не нужно было выходить из дома, чтобы бросить письмо в почтовый ящик, покупать марки (или маркированные конверты) для оплаты пересылки, получателю письма – ждать, когда почтальон бросит письмо в почтовый ящик, или идти в отделение связи, чтобы его получить.

И не важно, что первые электронные письма больше напоминали телеграммы, потому что могли быть только короткими текстовыми сообщениями. Сейчас же достаточно много сервисов Интернета, которые предоставляют услуги почты, в том числе – бесплатные.



*Электронное письмо – сообщение, которое передается средствами компьютерной сети.*

Работа электронной почты обеспечивается программными и аппаратными средствами. Почтовые программы, то есть программы, предназначенные для хранения и обработки **электронных писем**, хранятся на **компьютере-сервере**, а выполняются частично на компьютере-сервере и частично – на **компьютере-клиенте** (компьютере пользователя).



*Компьютер-сервер может состоять из нескольких реальных компьютеров, к тому же расположенных в разных местах.*

Каждый пользователь электронной почты имеет свой **«почтовый ящик»**, в котором хранятся электронные письма, поступающие на его имя.

✓ *Запись, определяющую путь к электронному почтовому ящику, называют электронным адресом.*

Электронный адрес состоит из двух частей, разделенных знаком @ (читается – «эт»), в которых содержатся: название почтового ящика пользователя и название почтовой службы. Левая часть электронного адреса (до знака @) должна быть оригинальной в пределах почтовой службы, например, комбинацией букв имени и фамилии пользователя. Эта часть адреса создается владельцем почтового ящика. Для этого можно использовать буквы только латинского алфавита, цифры, символы «-» (дефис) и «\_» (нижнее подчеркивание), однако адрес не может начинаться с этих двух символов.

Правая часть электронного адреса (после знака @) обозначает почтовую службу, программно-аппаратное обеспечение которой осуществляет пересылку и хранение электронных писем. На сервере, обслуживающем службу, хранятся сведения обо всех клиентах почтовой службы (служб), имена и закодированные электронные адреса всех почтовых ящиков, открытых в почтовой службе, которую обслуживает сервер.

Например, в электронном адресе serg25@km.kiev.ua обозначение ua – название страны (Украина), km.kiev.ua – название почтового сервера, serg25 – имя владельца почтового ящика. Другие примеры электронных адресов: Ganna05@ukrsat.com, lpi09@polyteh.lviv.ua, olia\_2001@kiev.ua.

✓ *Один пользователь может иметь несколько электронных адресов (почтовых ящиков), к которым он может обращаться с любого места Земли, где есть доступ к Интернету.*

Как и в большинстве сервисов сети Интернет, пользователям почтовой службы предоставляются услуги только после регистрации. Каждый пользователь в службе имеет учетную запись, содержащую данные о нем, необходимые как для авторизации вхождения, так и для восстановления данных (например, если пользователь забыл пароль). Эту запись называют «акаунт» или «екаунт» (от англ.: *account* – счет – в банке, например).

Для того чтобы воспользоваться услугой электронной почты, пользователь должен помнить логин (англ.: *login, logon*) – буквенно-цифровой набор символов, который идентифицирует пользователя, и пароль (англ.: *password*) (рис. 1.1).

✓ *Логин в большинстве почтовых служб является левая часть (до символа @) названия почтового ящика.*

Каждый почтовый сервис имеет свое клиентское программное обеспечение, к которому необходимо привыкнуть, хотя все они имеют много общего (рис. 1.2–1.5).

Клиентская часть программного обеспечения почтового сервиса пересылается на компьютер пользователя (клиента) после того, как пользователь выполнит в форме входа (рис. 1.1) определенные действия – введет свой логин и пароль.

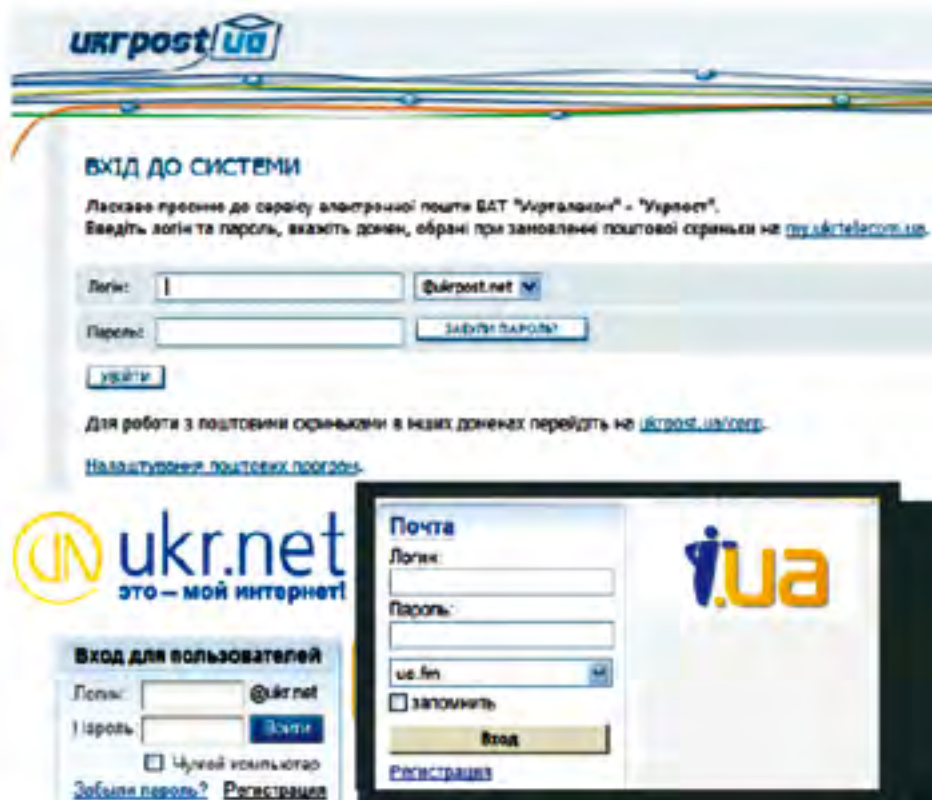


Рис. 1.1. Формы входа некоторых украинских почтовых служб: Укрпост, Freemail (ukr.net) и i.ua

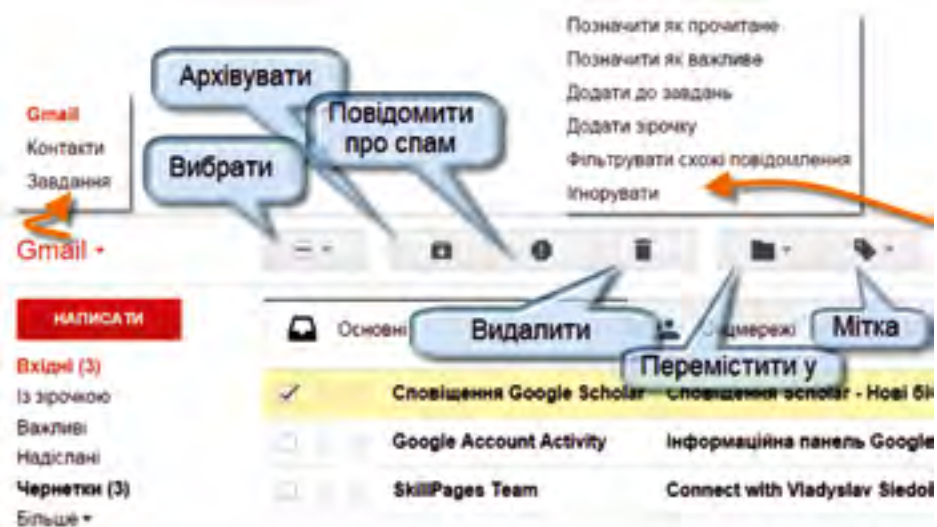


Рис. 1.2. Меню главной страницы клиентской части почтовой службы Gmail.com



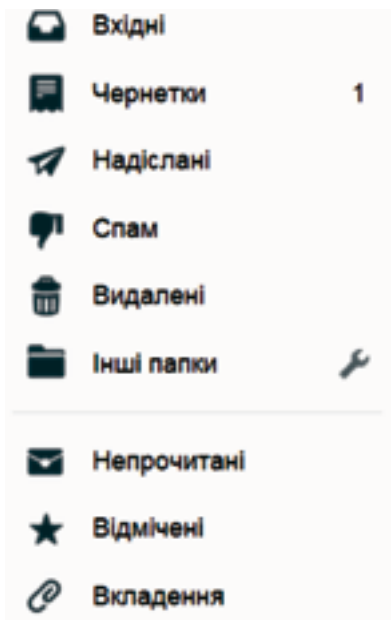


Рис. 1.3. Панель разделов для хранения писем почтовой службы Gmail

Встроенные редакторы большинства почтовых систем предоставляют пользователю возможность создавать письма в форме электронных документов, подобных веб-страницам, то есть форматированного текста с встроенными рисунками или другими объектами (смайликами, например).

Клиентское программное обеспечение кроме редактора сообщений обязательно содержит следующие разделы: **Адресная книга (Контакты)** для хранения электронных адресов и дополнительных сведений (как в мобильном телефоне); разделы (папки) для хранения писем: **Полученные (Входящие)**, **Отправленные**, **Черновики**, **Удаленные (Корзина)**, **Нежелательные сообщения (Спам)** (рис. 1.3, 1.4).

Назначение элементов управления окна **Входящие письма** почтового сервиса Freemail понятно из рис. 1.4.

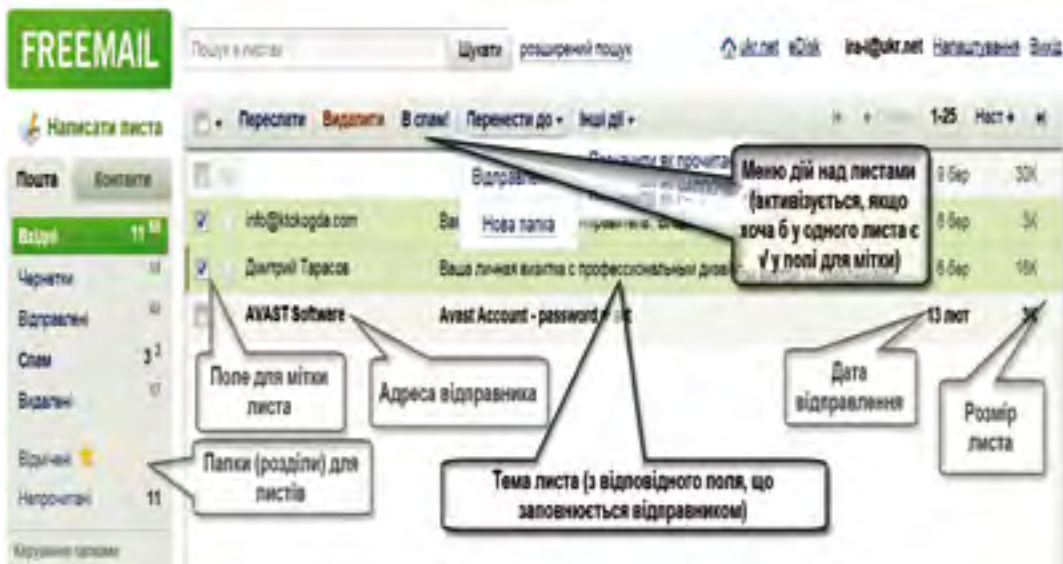


Рис. 1.4. Рис. 1.4. Воспроизведение на экране списка содержания раздела **Входящие (Полученные)** письма почтовой службы Freemail



## Проверяем себя

1. Какие основные виды услуг получает пользователь Интернета? 🌱
2. Для чего предназначена электронная почта? 🌱
3. Какие виды сообщений может содержать электронное письмо? ✦
4. Назовите программы, необходимые для работы с электронной почтой. ✦
5. Для чего предназначен электронный почтовый ящик? 🌱 ✦
6. Что называют электронным адресом? Какова его структура? ✦
7. Как электронное письмо «находит» своего адресата в сети? ✦
8. 🏠 👤 В предложенных электронных адресах найдите имя почтового ящика пользователя и название сервера: vita@school\_2.ua.net, group@kiev.com.ua, semurenko@ukr.net, semurenko@i.ua, olena@ua.fm. 🌱
9. 👤 👤 Какие преимущества электронной почты по сравнению с обычной почтой? 🌱
10. С помощью каких редакторов можно подготовить электронное письмо? ✦
11. 👤 👤 Каково назначение **Адресной книги** в почтовых программах? Как она еще может называться? В каких еще средствах приходится иметь дело с подобной формой хранения сведений? 🌱
12. Приведите примеры адресов, на которые по электронной почте отправляются уведомления. ✦
13. Где устанавливаются программы «почтовый клиент» и «почтовый сервер»? ✦

## Выполняем

### Создание электронного почтового ящика

Большинство общедоступных почтовых служб в настоящее время предоставляют не только услуги электронной почты: они имеют также разделы новостей, хранилища для хранения данных, программные средства для обработки фотографий, фильмов и тому подобное. Некоторые почтовые службы, благодаря наличию хранилища для хранения файлов пользователей, программ для перевода текстов, графических и мультимедийных редакторов, услуг «электронный секретарь» и т.д., являются составляющими **облачных сервисов**. Таковы, например, почтовые службы: Gmail, i.ua, meta.ua.

✓ ***Облачными сервисами** называют системы предоставления услуг, которые обеспечиваются средствами сети Интернет.*

При выборе почтовой службы для создания электронного почтового ящика следует определиться, будете ли вы использовать сервис только для переписки или для хранения документов, фотографий, общения с использованием служб мгновенных сообщений, видео- и голосовой связи.

✓ Желательно пользоваться только отечественными почтовыми сервисами (*ukr.net, i.ua* и т.п.) или почтовыми сервисами крупных сетевых компаний (*Google, Yahoo!* и др.), главные серверы которых расположены в Европе или США, то есть в странах с устоявшимися нормами соблюдения прав личности.

Процесс создания почтового ящика поддерживается программным обеспечением, которое загружается на компьютер пользователя. При входе в любую почтовую службу пользователю предлагают зарегистрироваться, после чего на его компьютер загружается программа, воспроизводящая на экране форму, заполняя поля ввода которой, пользователь регистрируется в системе (рис. 1.5, 1.6).

Весь Google  
Создайте один-единственный аккаунт и пользуйтесь любимыми сервисами Google!

**Служби, які стануть доступними після створення екаунта**

Мобильный Google  
Синхронизируйте файлы, закладки, контакты и другие данные на всех своих устройствах.

Как вас зовут  
Имя Фамилия

Придумайте имя пользователя

Придумайте пароль

Подтвердите пароль

Дата рождения  
день месяц год

Пол

Мобильный телефон  
+380

Залпасний адрес зп. пошти

Докажите, что вы не робот  
Пропустить эту проверку. Может потребоваться проверка по телефону

5320

Введите текст:

Страна  
Украина (Україна)

✓ Я принимаю Условия использования и соглашаюсь с политикой конфиденциальности Google

Далее

**САРТСНА (англ.: completely automated public Turing test to tell computers and humans apart - повністю автоматизований публічний тест Тюринга для розрізнення комп'ютерів і людей).**

Рис. 1.5. Окно облачных сервисов Google, предназначенное для создания почтового ящика

✓ Тщательно и разборчиво записывайте все, что вы вводите в поля регистрационной формы – даже номер своего мобильного телефона – это поможет избежать проблем, связанных с потерей паролей, логинов.

В процессе заполнения формы необходимо внимательно следить за сообщениями программы. Если программа выдает сообщение типа «Пользователь с таким логином уже существует», следует изменить логин или выбрать его из тех, которые будут предложены почтовой системой.

✓ *Пароль должен содержать шесть и более символов. Надо быть очень внимательным при его первом вводе, поскольку нужно учитывать не только последовательность нажатия клавиш букв и цифр, но и состояние клавиш Shift, Caps Lock и др., язык ввода.*


Очень важным моментом процесса регистрации в системе является проверка на осознанность деятельности, то есть на то, является ли пользователь человеком. Для этого используется тест на распознавание символов (CAPTCHA, произносится «капча», рис. 1.5, 1.6).

The image shows a registration form for ukr.net with several annotations in blue speech bubbles and orange arrows. At the top, the text reads: "Зареєструйтесь та отримайте пошту скриньку @ukr.net | 4ГБ простору на хмарнолу сервісі vДиск". The form includes fields for "Логін" (Login) with "@ukr.net" suffix, "Пароль" (Password), "Ім'я" (Name), "Прізвище" (Surname), and gender selection ("Чоловік" / "Жінка"). A red error message "Це поле має бути заповнено" is shown below the password field. Below these are fields for "День народження" (Date of birth) with "День" and "Січня" dropdowns, "Рік" (Year), "Мобільний телефон" (Mobile phone), and "Поточний email (якщо є)" (Current email). A CAPTCHA section shows the text "УТТ8Н5" with a note "Ми хочемо переконатися, що ви - людина" and "Введіть символи, відобразили на малюнку." A "Вводимо УТТ8Н5" bubble points to the CAPTCHA input. At the bottom, there is a green "Отримати скриньку" button and a red "Прочитати!" button. A footer note states: "Регистрація означає вашу згоду з Умовами використання скриньки та Угодою про конфідційність".

Рис. 1.6. Создание собственного почтового ящика на почтовом сервисе ukr.net

Обязательной частью процесса регистрации в системе является ознакомление пользователя с условиями пользования услугой (услугами). Это действие, по своему значению, соответствует чтению будущим пользователем инструкции по правилам эксплуатации прибора, который он хочет приобрести.

После создания учетной записи следует выйти из соответствующей программы и зайти в нее уже с правами зарегистрированного пользователя.

 Несмотря на то что на этапе создания учетной записи никакие данные не проверяются на соответствие реальным (кроме номера мобильного телефона и адреса дополнительного почтового ящика), всегда можно определить, с какого компьютера было отправлено сообщение.

## Выполняем

### Создание и отправка простого письма

Для того чтобы отправить кому-либо электронное письмо, необходимо знать его адрес и иметь доступ к электронному почтовому ящику, с которого это письмо передается. Если у пользователя есть доступ к определенному почтовому ящику (логин и пароль), то создание письма начинается с загрузки программного обеспечения определенной почтовой службы и ввода логина и пароля в форме входа.

Некоторые почтовые службы являются частью порталов облачных сервисов, поэтому этап входа в почтовую службу имеет несколько дополнительных шагов. Например, чтобы воспользоваться почтовой службой одного из самых известных украинских Интернет-порталов – i.ua, необходимо не просто вызвать его страницу, но и выбрать одну из почтовых служб (рис. 1.7).

Если пользователь не помнит пароль, большинство почтовых служб предложат воспользоваться сервисом напоминания пароля. Для того чтобы воспользоваться этим сервисом, нужно вызвать его, ответить на некоторые вопросы и получить временный пароль по электронной почте, сообщением (SMS) на мобильный телефон или иным образом.

После входа в собственный почтовый ящик можно создать письмо, подав команду **Создать письмо** (i.ua), или **Написать письмо** (ukr.net), или **Написать** (gmail.com). После этого следует заполнить строки **Кому:** и **Тема:**

Строку **Кому:** во всех почтовых службах можно заполнять вручную, вводя адрес электронной почты получателя письма. Если адрес получателя уже есть в **Адресной книге (Контактах)**, следует нажать левую кнопку мыши на значке @ в строке **Кому:** (i.ua), или нажать ее правую кнопку на **Кому:** (ukr.net, gmail.com). После этого следует найти нужный адрес и ввести его в строку.



Рис. 1.7. Вхід в поштовий ящик на порталі, сервери якого обслуговують декілька поштових служб

Создавать текст письма следует с соблюдением тех же правил, по которым бы создавалось обычное письмо.


В письмо можно вставлять растровые изображения, смайлики.

После создания письма подается команда **Отправить**.

## Проверяем себя

1. Зачем необходимо записывать все данные, вводимые при регистрации и создании аккаунта? 🟢
2. Что называют учетной записью пользователя? 🟢
3. Чем отличаются формы ввода данных, представленные на рис. 1.5 и 1.6? Что в них общего? ✚
4. Какие данные, используемые на этапе создания учетной записи, являются личными данными? ✚
5. Почему в процессе создания учетной записи необходимо тщательно записывать все данные, которые вы вводите? 🟢
6. 🧑🏻🧑🏻 Что такое CAPTCHA, для чего этот метод используется? Какие еще случаи использования CAPTCHA можете назвать? 🟡
7. Какие основные услуги предоставляет текстовый редактор почтовой программы? ✚



8.  Что такое «смайлики»? Когда и зачем их используют? Приведите примеры смайликов и их значений. ✦
9. Какие свойства объекта «электронное письмо» отображены на экране в режиме воспроизведения содержимого папки **Входящие** (рис. 1.4)? ✦
10. Правильно ли утверждение: «электронным почтовым ящиком является часть памяти внешнего запоминающего устройства сервера»? Объясните. ★
11. Что нужно для того, чтобы отправить кому-то электронное письмо? ★
12. Проанализируйте рис. 1.7. Как можно избежать обращения к услуге «Напоминание пароля»? ★
13. Проанализируйте и сравните процедуры создания электронной почты в нескольких почтовых сервисах. ★

## 1.2. Добавление файлов к письмам. Перенаправление писем

Вместе с письмом можно передавать файлы, которые добавляют к письму (англ.: *to add files*). Файлы (например, фотографии, рисунки, текстовые документы) должны быть созданы заранее. Ограничение на размер вложений для большинства почтовых сервисов Украины сейчас составляет 20 МБ (5–6 фотографий среднего качества, один видеоклип продолжительностью 5–10 минут с качеством обычного телевидения и т.п.). Благодаря развитию технологий размер файлов, которые можно присоединить к электронному письму, постоянно увеличивается.

Обмен письмами с вложениями между почтовыми ящиками, которые принадлежат одной почтовой службе, имеет менее жесткие ограничения – таким путем можно передавать письма с присоединением файлов размером от нескольких до десятков гигабайт.

В пределах большинства почтовых служб можно передавать файлы, присоединяя к сообщению не сам файл, а ссылку на него. Например, из почтового ящика `viktor@ukr.net` на ящик `marichka@ukr.net` можно передать видеофильм объемом 1 ГБ и более, разместив его сначала в облачном хранилище eДиск, и после этого передать в письме полученную ссылку. Более того, иногда можно просто открыть доступ к папке на облачном хранилище, а в этой папке разместить файлы, которые должны быть переданы. Адрес этой папки можно передать в коротком письме или в сообщении SMS.

Недостатком передачи больших файлов с использованием ссылок на них является то, что ссылки, передаваемые электронным письмом, имеют ограниченный срок действия.

Иногда возникает необходимость ознакомить с полученным письмом нескольких друзей. Для этого существует услуга «Переслать» (рис. 1.8). Этой услугой следует пользоваться, учитывая, что, пересылая письма с одной почтовой службы на другую, не всегда можно гарантировать, что

получателю будет доступно все присоединенное, в частности, присоединенные большие файлы.

Большинство современных почтовых программ имеет встроенные средства борьбы с вирусами и нежелательными письмами.

Нежелательные письма, содержащие рекламный характер, называются **спамом**. Такие письма большинство почтовых программ пересылает в специальное хранилище, доступное пользователю. Его следует просматривать, поскольку довольно часто в него могут попадать и нужные письма, в частности, от корреспондентов, адреса которых отсутствуют в адресной книге.

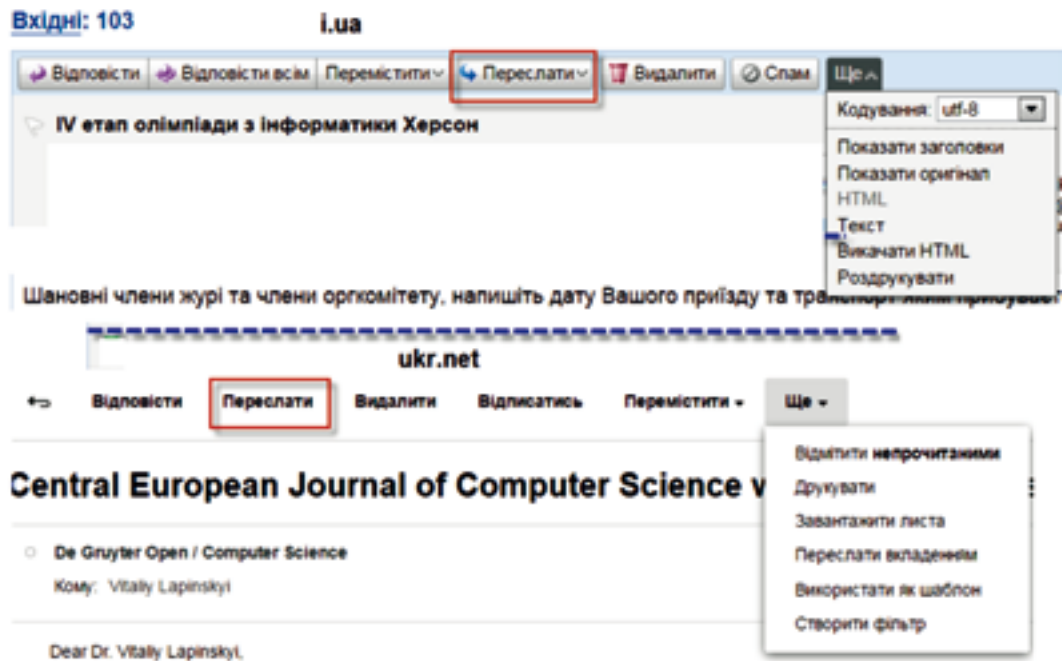






Рис. 1.8. Пересылка письма в сервисах i.ua (вверху) та ukr.net (внизу)

## Проверяем себя

1. Где хранятся электронные письма до получения их адресатом? После получения? 🧑🏻
2. Какого размера хранилище файлов доступно для пользователей почтовой службы i.ua? 🧑🏻
3. Чем отличается пересылка растрового изображения, вставленного в текст письма, от пересылки его как присоединенного файла? ➕
4. 🧑🏻 Почему для пересылки больших файлов целесообразно использовать облачные хранилища данных? ➕



5.  Вам нужно передать много небольших файлов нескольким корреспондентам, которые используют ту же почтовую службу, что и вы. Как это сделать лучше всего? ★
6. Что нужно сделать для того, чтобы ваше письмо прочитали несколько респондентов? 📧
7.  Будет ли одинаково выглядеть одно и то же письмо, принятое респондентами различных почтовых служб? Почему? Проверьте, если вы имеете адреса нескольких респондентов на разных почтовых службах. 📧
8.   Вам нужно переслать 2 ГБ фотографий своему товарищу. Как это сделать лучше всего? Рассмотреть варианты: а) правая часть адреса почтового ящика совпадает с правой частью адреса вашего товарища; б) правые части адресов разные. ★

### 1.3. Правила и этикет электронной переписки.

#### Правила безопасного пользования электронным почтовым ящиком

Общение, поиск и получение различных сведений занимают важное место в становлении личности человека. Однако всегда следует помнить, что **самым важным для человека является его здоровье: физическое, психическое, социальное.**

✓ *Пользуясь Интернетом, следует соблюдать очень простые правила.*

1. Никогда не засиживаться в сети дольше, чем это нужно для того, чтобы найти нужные сведения, написать письмо, пообщаться с друзьями. Общее время пребывания в сети не должно превышать 20–40 минут в день.

2. Никогда, ни при каких обстоятельствах, не передавать в сеть персональные данные (свои, своих родителей, кого-либо другого).

3. Не играть в компьютерные игры в сети, даже если очень хочется – подумайте, что полезнее для вашего здоровья – просидеть час в кресле у компьютера или погулять на свежем воздухе.

4. Никак не реагировать на сообщения из сети, которыми кто-то пытается вас обидеть, – сделать вид, что вы этих сообщений не видите, не заходить больше на соответствующий сайт, внести пользователя в «черный список».

5. Следить за тем, чтобы на компьютере было установлено антивирусное программное обеспечение.

6. Не использовать нелицензионное программное обеспечение – кроме того, что это противозаконно, в основном именно с таким программным обеспечением в компьютер попадают нежелательные программные средства.



## Правила безопасной работы с электронной почтой

Как и пользование другими службами (сервисами) Интернета, работа с электронной почтой требует соблюдения определенных правил безопасности.

Неукоснительное соблюдение простых правил поможет избежать различных неприятностей: от заражения компьютера вирусом, установки на нем нежелательных программ до материального ущерба, морального дискомфорта.

✓ *Общайтесь по электронной почте так и с теми, как и с кем вы бы общались в реальной жизни.*

При написании письма не забывайте подписаться.

Используйте только нормативную лексику.

Не пытайтесь настойчиво убеждать корреспондента в чем-либо. Всегда, хотя бы коротко, но вежливо и приветливо, ответьте на поздравления.

Научившись пользоваться Адресной книгой и имея в ней несколько десятков адресов, не злоупотребляйте групповыми рассылками – во-первых, вы не служба новостей, а во-вторых, не всем вашим друзьям приятно видеть свой адрес рядом с адресом кого-то, с кем не хочется видаться. Лучше не ленитесь и отправьте каждое письмо в отдельности.

Будьте внимательны, отвечая на письма, – если вы нажмете «Ответить всем», ваше письмо смогут прочитать все, кто получал письмо вместе с вами (адреса чьих ящиков были в адресной строке при отправке письма).

✓ *До сих пор встречаются люди, которые пытаются с использованием электронной почты получить материальную выгоду нечестным путем.*

**Поэтому:**

☞ не передавайте через сеть частную информацию, которая может быть использована злоумышленником против вас или вашей семьи;

☞ никогда и никому не сообщайте пароль к своему почтовому ящику;

☞ никогда и никому не пересылайте сведения коммерческого характера (номер кредитной карты, код сейфа и тому подобное);

☞ не отвечайте на письма с сообщениями о том, что вы как будто выиграли в лотерею, у вас умер дальний родственник и оставил наследство и тому подобное.


✓ *Некоторые злоумышленники пытаются войти в доверие с целью завладеть не только материальными вещами, но и привлечь людей к разного рода противоправной деятельности.*

**Поэтому:**

☞ не передавайте через Интернет свой домашний адрес и телефон, имена членов семьи, частные семейные новости;

☞ при работе на чужом компьютере (даже если это компьютер вашего друга) всегда ставьте флажок при входе в свою почту и выходите из почтового сервиса после завершения работы;

☞ не отвечайте и не реагируйте на письма от незнакомых (в реальном мире) людей, особенно в случаях, когда в них сообщают о каких-то массовых мероприятиях.

 При появлении любой угрозы со стороны сети Интернет не стесняйтесь обращаться к родителям, учителям и другим взрослым, которым вы доверяете.

### Практическая работа № 1

<b>Тема:</b>	Электронная переписка с использованием веб-интерфейса. Вложения файлов
<b>Цель:</b>	Научиться использовать почтовые сервисы сети Интернет

### Задания

Создайте несколько текстовых файлов. Запишите их расположение и названия (чтобы не искать долго в процессе прикрепления их к письму).

Войдите в свою почту, напишите учителю письмо следующего содержания:

*Здравствуйте, Иван Петрович!*

*Я уже создал собственный почтовый ящик.*

*К письму прикреплены два файла с моими рисунками.*

*С уважением*

*Ученик 7 Б класса*

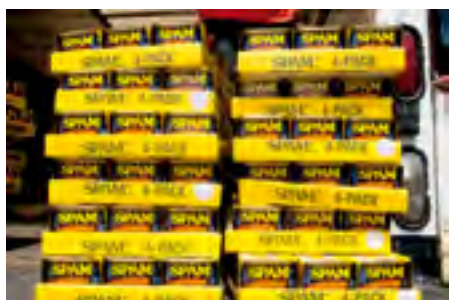
*Иващенко Николай*

Обратите внимание: некоторые почтовые сервисы имеют опции хранения отправленного письма, запроса ответа о прочтении письма. Установите переключатели в нужное положение. ★

Прикрепите к письму созданные файлы и отправьте.



### ДЛЯ ЛЮБОЗНАТЕЛЬНЫХ



#### Что такое СПАМ?

Слово «SPAM» появилось в 1936 году. Оно расшифровывалось как SPiced hAM (острая ветчина) и было товарным знаком для мясных консервов компании Hormel Foods – острого колбасного фарша из свинины.

После Второй мировой войны остались огромные запасы консервов, которыми

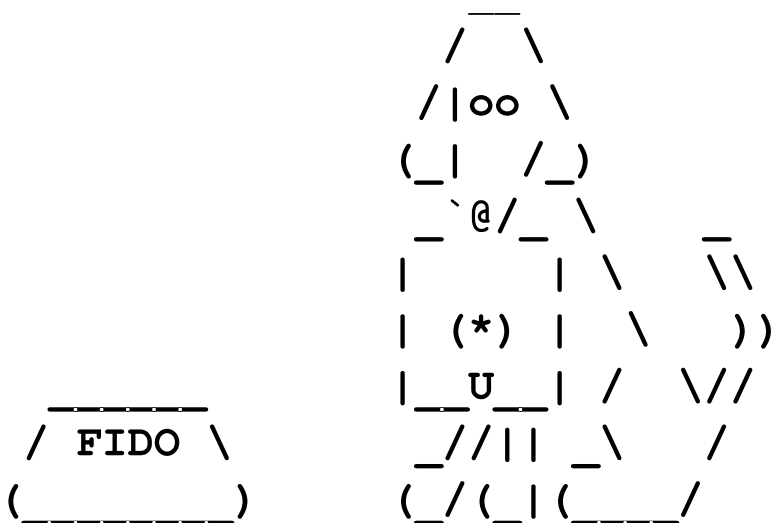
обеспечивались американские военные. Для того чтобы избавиться от этой залежавшейся продукции, компания Hormel Foods провела первую в своем роде рекламную кампанию.

Слово «SPAM» бросалось в глаза на каждом углу, с витрин всех магазинов, оно было написано на бортах автобусов и трамваев. Это слово можно было прочесть на фасадах домов и в газетах. Реклама консервов «SPAM» беспрерывно транслировалась по радио.

А уже в 1986 году в конференциях Usenet появилось множество одинаковых сообщений от некоего Дэйва Родеса, который рекламировал новую финансовую пирамиду. Заголовок гласил: «Заработай кучу денег», а в письмах содержалась инструкция, как это сделать. Автор упорно продолжал дублировать свои тексты, и они настолько надоели респондентам, что их начали сравнивать с рекламой консервов «SPAM». Таким образом за словом «спам» закрепилось новое значение, которое перешло в компьютерную терминологию для обозначения назойливых рекламных рассылок.

### Рождение Смайлика 🐶 ~:-) \*;-)

Смайл, смайлик (от англ.: *smile* – улыбка), а также эмотикон, эмограмма (англ.: *emoticon*) – схематическое изображение человеческого лица, используемое для передачи эмоций. Возник он, когда сетевое общение было ограничено только текстовыми сообщениями. Примером такого изображения является собачка – логотип компьютерной сети Фидонет (существует с 1984 года, англ. FidoNet, /'faiðənet/).



Смайл может состоять из различных символов – букв алфавита, знаков препинания, служебных символов. Особую популярность смайлы получили с распространением Интернета (чаты, форумы, ICQ и тому подобное) и мобильных телефонов (короткие сообщения – SMS), где широко используются для обмена сообщениями.

Считается, что первый смайлик – двоеточие с дефисом и круглой скобкой :-)) – использован 19 сентября 1982 года профессором психологии Университета Карнеги-Меллон Скоттом Фалманом в сообщении на онлайн-форуме.

Этот день считается **Всемирным днем Смайлера** ☺

В среде украинских пользователей используют украинские патристические смайлы:

мужские – с «козацким чубом» – символом, который называют тильдой ( ~ ):

~:-) ~;-P ~;-| ~8-) ~8-D

женские – с «цветком» ( \* ):

\*;-) \*;-P \*;-D \*;-| \*' \_ '\*



## СЛОВАРЬ

**Почтовый сервис** – служба Интернета, которая предоставляет пользователям возможность передавать электронные письма.

**Присоединение к письму** – способ передачи электронного документа вместе с письмом.

**Электронная почта (e-mail)** – совокупность средств для обмена сообщениями между абонентами с помощью компьютерной сети.

**Электронный адрес:** *имя\_почтового\_ящика\_пользователя@название\_почтовой\_службы.*



## РАЗДЕЛ 2. МОДЕЛИРОВАНИЕ



Электронный документ – сведения, хранящиеся в виде, который обеспечивает их обработку на компьютере. Данные – сведения об определенном объекте или процессе, представленные в форме цифр или текста.



Понятие модели. Типы моделей. Понятие предметной области. Формы представления информационной модели: описание, таблица, формулы, схемы и др. Этапы построения информационной модели. Карты знаний, их назначения. Редактор карт знаний.

### 2.1. Модель. Предметная область

Окружающая среда – это совокупность большого количества живых и неживых объектов.

Научное познание, исследование мира, окружающего человека, заключается в выявлении определенных закономерностей, присущих явлениям, которые происходят с объектами. Закономерности обнаруживают, исследуя явления путем измерения некоторых величин (параметров объекта) и анализа полученных наборов значений.

Среди многих параметров объекта есть главные, совокупность значений которых существенно определяет его свойства. Например, для компьютера такими параметрами являются: тактовая частота процессора, объем оперативной памяти, длина диагонали и разрешение монитора, объем памяти винчестера.

Количество главных параметров обычно находится в пределах от единиц до нескольких десятков и более.

✓ *Чем больше значений параметров известно для объекта, тем точнее описываются его свойства.*

✓ *Для изучения реальных объектов часто используют их модели.*

Моделями являются, например, карта автомобильных дорог местности, которую мы изучаем прежде, чем отправиться в путешествие (рис. 2.1), макет застройки микрорайона, схема радиоприемника, глобус, макет гидроэлектростанции.

Можно создать модели объектов, существующих в воображении человека. Например, строя график функции, мы создаем модель зависимости между величинами, которые находятся в отношении: аргумент  $\Rightarrow$  функция  $\Rightarrow$  результат (значение функции).

Понятие «отношение» касательно модели и объекта моделирования является очень важным, поскольку отображает обязательность наличия сходства объекта и его *отображения* – модели.



**!** *Моделью* называют воспроизведение какого-нибудь объекта в упрощенном виде с целью отображения и/или исследования его свойств или процессов, которые с ним происходят.

Модель является отображением объекта, в котором воспроизведены важнейшие (для определенного применения) его свойства.

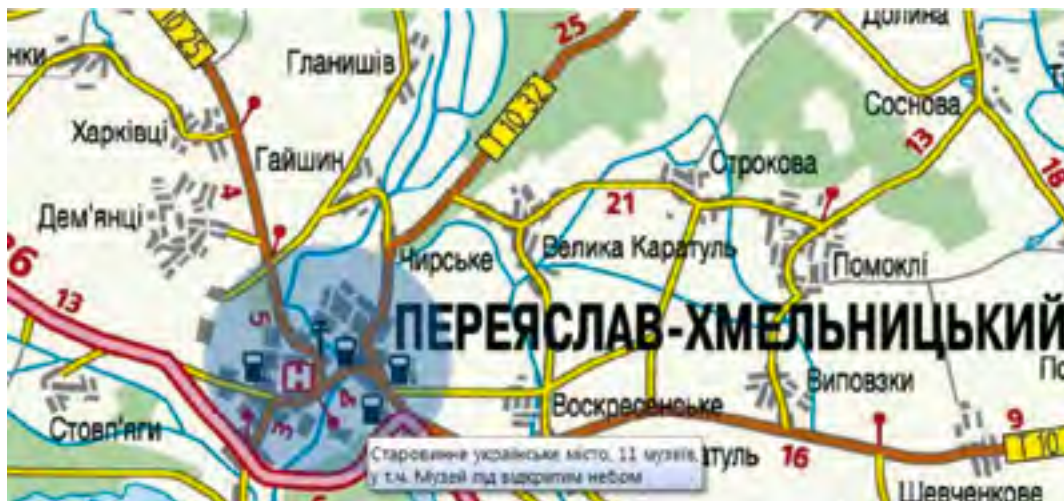


Рис. 2.1. Карта – модель местности

**✓** *Процесс создания моделей называется моделированием.*

Моделирование – один из основных современных методов исследования объектов природы, общества и явлений. Модели создают с целью изучения, исследования и анализа процессов, происходящих с объектами при определенных условиях.

В процессе создания многих реальных объектов (самолеты, космические и морские корабли и т.д.) нужно обязательно разрабатывать и исследовать их модели.

Моделирование и использование моделей невозможно без установления определенных правил, договоренностей относительно значений определенных слов – *терминов*. Эти правила зависят от объектов моделирования и задач, которые ставит перед собой исследователь.

Вычисляя расстояния на местности с целью размещения определенного здания, создавая план расположения, архитектор использует определенные правила создания изображения. Это делается для того, чтобы созданное изображение было понятным для строителей.

Врач, описывая процесс лечения болезни в рецепте, использует латинские термины, понятные работнику аптеки – провизору. Однако не всегда врач может понять, что изображено на электрической схеме, а электрик – что изображено нотами (рис. 2.2). Их профессиональные знания находятся в разных предметных областях.





Рис. 2.2. Нотная запись и принципиальная электрическая схема




✓ *Итак, каждая предметная область не только имеет различные объекты изучения, но и использует различные способы и средства их описания.*

Способы описания моделей в каждой предметной области разные, приспособленные для наиболее эффективного использования объектов исследования.

### Проверяем себя

1. Приведите примеры моделей, которые вы используете в повседневной жизни. 🌱
2. 👤 Моделью какого процесса является расписание уроков? Приведите примеры подобных моделей, обсудите, зачем нужны подобные модели. 🌱
3. Моделью или объектом является глобус? Какие свойства объекта на нем воспроизведены, а какие – проигнорированы? ✚
4. 👤 Чем отличается электронный документ от бумажного? В каком случае электронный документ может считаться только моделью бумажного, а в каком – наоборот? ✚
5. Можно ли считать моделью правило перехода улицы? Объясните, почему. ✚
6. На уроках каких предметов вы чаще всего оперируете моделями? ✚
7. Сравните значение слова «круг» в выражениях: круг друзей, круг общения и площадь круга, диаметр круга. К каким предметным областям относятся эти выражения? ✚
8. Моделью какого реального процесса может быть математическое выражение  $y = v \cdot x$ , если  $v$  измеряется в километрах в час, а  $x$  – в часах? ✚
9. Моделью какого реального процесса может быть математическое выражение  $y = a \cdot x - b \cdot x$ , если  $a$  и  $b$  измеряются в дециметрах кубических в минуту, а  $x$  – в минутах? ✚
10. В результате ежесуточных измерений дневной и ночной температур в течение месяца получили набор данных. В какой форме лучше всего подать этот набор? ✚
11. 👤 Что общего у бумажного самолетика и крупнейшего в мире грузового самолета АН-225 «Мрия»? ✚

## Выполняем


-  Внимательно рассмотрите рис. 2.1. Найдите ответы на такие вопросы:
  - Моделью какого объекта является карта?
  - Какие сведения можно получить, исследуя эту модель?
  - Можно ли проложить автомобильный маршрут с использованием подобной карты? Что будет и что не будет учтено при этом?
  - Можно ли найти в населенном пункте, изображенном на подобной карте, дом по указанному адресу? Объясните. ✚
-  Найдите в сети Интернет карту вашего населенного пункта и его окрестностей. Какие сведения отражены на карте и какие проигнорированы? ★
-  Опишите десятью словами объект (домашнее животное, предмет, находящийся у вас на письменном столе, цветок на подоконнике, дерево, мимо которого вы проходите по дороге в школу) таким образом, чтобы о нем можно было рассказать товарищу, а он смог бы его распознать по вашему тексту. Обсудите с друзьями, попробуйте найти те слова, по которым объект, словесную модель которого вы создали, был бы однозначно идентифицирован. ★

## 2.2. Типы моделей. Формы представления информационной модели

Модель объекта должна отражать его важнейшие свойства. Если необходимо исследование изменений, которые происходят с объектом, в процессе моделирования нужно предусмотреть изменение параметров модели со временем или вследствие определенных внешних воздействий, которые тоже следует смоделировать.

- ✓ *Модели, параметры которых изменяются со временем, принято называть динамическими моделями.*
- ✓ *Статическими моделями называют модели, вид которых не изменяется во времени.*

Таким образом, карта, напечатанная на бумаге, плакат, фотография, чертеж, выполненный на бумаге, могут быть только статическими моделями. Электронная карта, которая может обновляться со временем, уже является динамической моделью. Динамической моделью может быть и чертеж, созданный системой автоматизированного проектирования. Более того, этот чертеж можно использовать для определения и прогнозирования определенных свойств объекта, которого еще не существует.

 *Выделяются три основных типа моделей:*

- *физические модели;*
- *математические модели;*
- *информационные модели.*

✓ **Физическая модель** – это материальная модель, создание которой основывается на принципах масштабирования или упрощения. Такие модели иногда называют **натурными**.

Примерами **физических моделей** могут быть: изготовленный из пластика глобус; модели зданий микрорайона, вырезанные из пенопласта; модель космического корабля, созданная из папье-маше; модель электрической цепи, где использованы учебные приборы. В физической модели часто воспроизводятся закономерности, свойственные реальному объекту, но в меньших масштабах. Например, физической моделью молнии может считаться разряд между электродами электрофорной машины, физической моделью землетрясения – явления, происходящие в сооружении, установленном на специальной площадке, которая совершает колебания, подобные колебаниям земной поверхности. Физической моделью взаимодействия самолета и воздуха является испытание уменьшенной во много раз его модели в аэродинамической трубе – аппарате, в котором создается сильный поток воздуха.

✓ **Математическая модель** подается в виде одного или нескольких математических уравнений, которые отражают закономерности, свойственные объекту или явлению.

Многие задачи по физике, биологии, химии решаются с помощью уравнений и неравенств. Например, уравнение  $v = \frac{s}{t}$  описывает скорость равномерного прямолинейного движения объекта. Различные движения могут описываться разными уравнениями, которые являются математическими моделями движений. Математические модели особенно важны в строительных работах, аэрокосмической промышленности, астрономии и т.п.

✓ **Информационная модель** – это совокупность данных и связей между ними, которые описывают объект.

Например, информационной моделью предприятия могут быть данные о количестве работников, видах продукции, средняя заработная плата, годовой доход и т. п. Информационные модели могут сочетаться с математическими. Без математических моделей иногда вообще нельзя обойтись. Например, без них невозможно определить последствия прорыва плотины гидроэлектростанции, место и время посадки космического корабля, возможное состояние домов после землетрясения.

✓ Обычно информационные модели объекта отражают самые существенные параметры, внутренние связи в объекте и закономерности изменений значений параметров в зависимости от внешних воздействий.

ⓘ **Информационная модель** – это модель, основной составляющей которой являются сведения о свойствах и состояниях объекта моделирования, процесса, явления.

Объединив информационную и математическую модели, можно построить **компьютерную модель**.

✓ *Представление объекта в форме компьютерной модели является его отображением в компьютерной программе, которая описывает свойства объекта, выделенные как важные.*

Воспроизведение модели объекта компьютерной программой может осуществляться с разной точностью, в зависимости от требований пользователя, возможностей, которые обеспечивают программно-аппаратные средства, и наличия данных, описывающих объект моделирования.

Например, для того чтобы указать местонахождение морского судна, в большинстве случаев достаточно указать его координаты – широту и долготу, с погрешностью в несколько километров (примерно 0,1 градуса). Если же судно находится вблизи берега, то такая погрешность уже довольно велика. Следовательно, и количество данных, необходимых для моделирования земной поверхности, должно быть больше, для того чтобы судоводитель мог знать глубину, точное расстояние до препятствий.









Для расчета траектории движения спутника Земли или баллистической ракеты (построения компьютерной модели движения тела) необходимо иметь значения их координат и скорости в нескольких точках.

Для того чтобы смоделировать с использованием компьютерной программы платье для конкретной клиентки, необходимо ввести в программу результаты определенных измерений.

Для того чтобы сделать надежный прогноз перспектив развития аграрного предприятия, необходимо иметь данные о перспективах на урожай определенной сельскохозяйственной культуры в мире и на посевных площадях хозяйства.

Расчет возможных угроз для жизни на Земле в результате падения астероида может быть осуществлен только при наличии данных астрономических наблюдений.

### Проверяем себя

1.  Можно ли, исследуя модель, получить новые знания об объекте? 
2. Какой моделью – статической или динамической – является расписание уроков? 
3.  Раньше каждый только что сконструированный самолет строили в количестве не менее двух образцов, один из которых ни разу не поднимался в небо. Что делали с этим образцом и зачем? 
4. Можно ли считать самолет, который не взлетал в воздух (см. предыдущий вопрос), моделью? Если да, то какой? 
5. Почему разработчики механизмов, транспортных средств и т.п. в большинстве случаев предпочитают компьютерное моделирование? 
6. Можно ли считать компьютер моделью мозга человека? Объясните, почему, до какого предела. 

7. Какие свойства чемодана, сундука, спичечной коробки отображаются в чертеже параллелепипеда, выполненном на бумаге? ✦
8. 🏠 Всемирно известная фирма «Тетра Пак» ведет историю возникновения своего названия от геометрического тела «тетраэдр». В каких реальных объектах нашел отражение абстрактный объект «тетраэдр»? Почему? ★
9. 👤 🏠 Попробуйте на словах описать ваши требования к динамической модели автомобиля, на которой человек должен научиться водить реальный автомобиль, сидя в кресле водителя, находящемся в помещении. ★

### Выполняем


1. 👤 Продолжите выражение: «Пчела и инженер строят совершенные конструкции – соты и дома, но самый плохой инженер отличается от наилучшей пчелы тем, что прежде чем строить...». Обсудите. ✦
2. 🏠 👤 Создайте математические модели для таких задач.
- А). Бассейн объемом  $6 \text{ м}^3$  наполняется через две трубы. Через одну из них вода поступает по  $0,7 \text{ м}^3$  в час, а через другую – по  $1 \text{ м}^3$  в час. За сколько часов наполнится бассейн?
- Б). Участок земли  $0,06$  га вскапывают двое работников. Первый из них вскапывает за час  $0,007$  га, а второй –  $0,01$  га. За сколько часов они вскопают участок, работая вместе?
- Решите задачи. 🌱 Объясните, что общего в этих задачах. 🌱 Какие еще модели можно описать подобными уравнениями, то есть отобразить с помощью такой же математической модели? ✦
- Почему модель, описанную математически, почти всегда можно считать динамической? ★

### 2.3. Построение информационных моделей

Для каждой области знаний, науки и техники существуют свои способы моделирования. Для портного моделью человеческой фигуры является манекен, для конструктора кресла космонавта – гипсовый отпечаток тела космонавта, для врача – скелет человека, рисунки, на которых изображено строение мышц и других органов.

Моделировать, как мы уже знаем, можно не только реальные объекты, но и те, которые существуют в воображении человека. Художник, рисуя портрет, натюрморт, пейзаж, воспроизводит в рисунке не только форму и цвета объекта, но и передает свое отношение к модели, свое восприятие действительности. Композитор, создавая мелодию и записывая ее нотами, создает сообщение, описывающее состояние его сознания, его чувств. Таким образом, художественные произведения тоже можно считать моделями реальных объектов и отношения к ним человека.

Каждая область деятельности человека имеет свои особенности, которые определяются особенностями объектов изучения.

 *Результатом изучения человеком окружающего мира и самого себя являются знания, которые отражаются в определенном образом поданных сведениях об объектах и процессах.*

В естественных науках основные закономерности описываются в форме математических моделей. Там, где невозможно использовать математические модели (в искусстве, общественных науках), применяют словесные описания, другие формы представления знаний.

Чтобы облегчить восприятие и запоминание знаний, было создано программное средство, действие которого основывается на том, что человек лучше воспринимает и запоминает то, что видит, – **карта знаний**.

**Карта знаний** (англ.: *Mind map* – карта знаний) – изображения, наглядно представляющие знания. Такая карта позволяет отобразить определенный процесс или идею полностью, а также хранить одновременно в памяти значительное количество данных, находить связи между отдельными частями, запоминать материал и воспроизводить его, даже спустя длительное время, как систему знаний об определенном объекте.

Карты знаний – удобный и эффективный способ иллюстрации знаний и процесса мышления. Их используют при формулировании новых идей, классификации и структурировании данных, анализе и составлении данных, принятии решений и тому подобное. Этот способ обладает большим преимуществом перед обычными общепринятыми способами записи. В отличие от текста, карты знаний не только аккумулируют факты, но и демонстрируют взаимосвязь между ними, обеспечивая тем самым быстрое и более глубокое понимание материала.

Карты знаний используют при проведении презентаций, мозговых штурмов, планировании деятельности, запоминании больших объемов данных, в процессе самоанализа, разработки сложных проектов, собственного обучения и развития.

Карты знаний могут быть представлены в виде диаграммы, на которой изображены слова, идеи, задачи или другие понятия, связанные ветвями, отходящими от центрального понятия или идеи.

Для создания электронных моделей знаний (карт знаний) могут быть применены растровые и векторные редакторы изображений. При создании карт знаний используется незначительное (по сравнению с созданием других типов изображений) количество визуальных объектов и вариантов их закрашивания, а важным и существенным при этом является установление связей между ними. Поэтому для рисования карт знаний было создано достаточно большое количество программных средств.

Одним из таких средств является MindMap, используя который, можно построить как простые, так и достаточно сложные карты знаний (рис. 2.3).



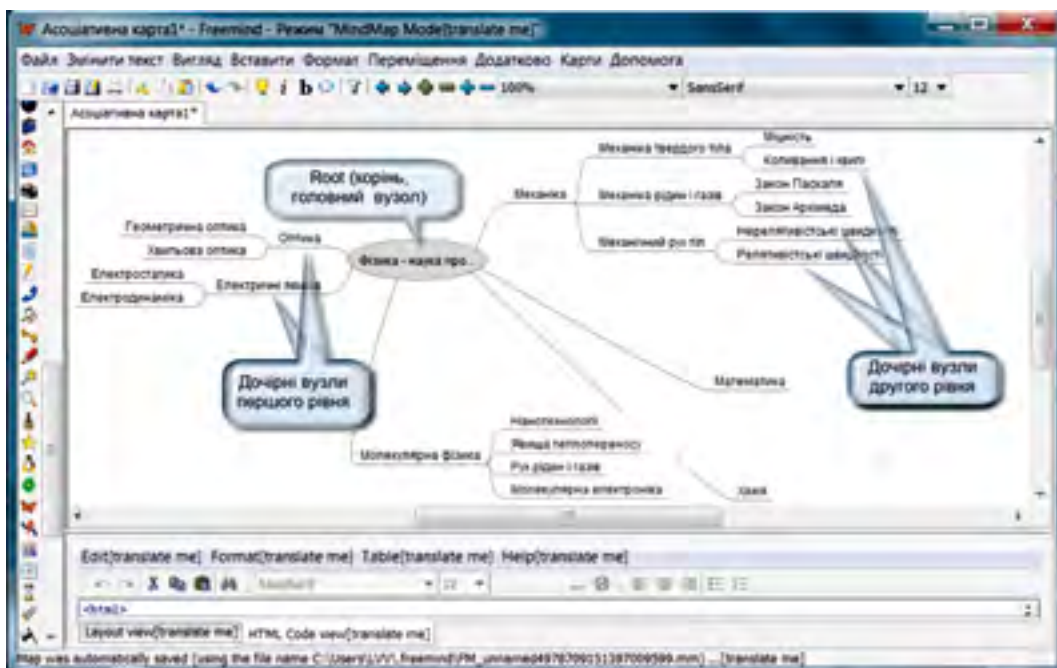


Рис. 2.3. Карта знаний предметной области «Физика»

✓ *Карты знаний можно считать моделями объектов, знания о которых в них отображаются.*

### Проверяем себя

1. Какие объекты исследования или изучения требуют создания моделей? 🌱
2. 👤 Всегда ли является правильным утверждение, что данным, полученным в процессе моделирования, следует доверять меньше, чем данным, полученным непосредственно от объекта исследования? ➕
3. Почему для человека важно не просто прослушать сообщение, но и увидеть модель, представленную графически? 🌱
4. Каким образом могут быть описаны знания об объектах природы? В чем проявляются различия определенных областей знания? 🌱
5. 🏠 Готовясь к контрольной работе по геометрии, ученик выписал на листе бумаги основные формулы и теоремы. Можно ли назвать созданное им сообщение моделью области знания? 🌟
6. Приведите пример использования в быту изображений, которые можно считать ментальными картами. ➕
7. 🏠 👤 Музыку можно записать с помощью нот, а можно и с помощью наборов чисел, обозначающих высоту, продолжительность и громкость звуков. Не означает ли это, что все на свете можно записать таким образом? 🌟



## Выполняем

1. 👤 Попробуйте отразить в графическом редакторе отношение Объект (Свойства) → Модель (Свойства) (рис. 2.4). Как отобразить то, что свойств объекта может быть гораздо больше, чем свойств модели? ✦
2. 🏠 В редакторе FreeMind или другом графическом редакторе создайте карту, показанную на рис. 2.3, дополните ее с максимально возможным учетом полученных знаний о процессе моделирования. ✦

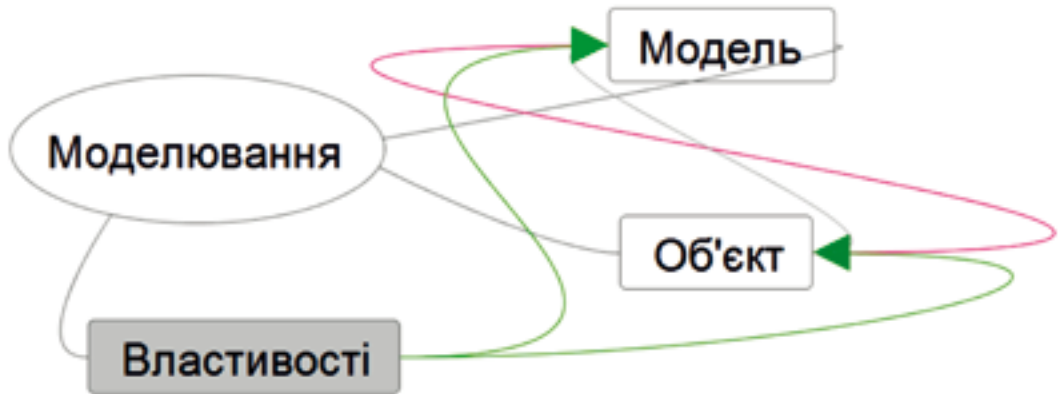


Рис. 2.4. Отношение Объект ↔ Модель и их отображение в свойствах

### Практическая работа № 2

<b>Тема:</b>	Построение информационных моделей в различных программных средах
<b>Цель:</b>	Приобрести практические навыки в разработке моделей

### Указания к работе по созданию карт знаний

1. Расположить основное понятие по центру карты знаний. Сформулировать его кратко и точно, добавить изображение.
2. Создать разветвления к главным составляющим понятиям, от которых, в свою очередь, отходят разветвления к другим, подчиненным. Отслеживать, чтобы на одном разветвлении находились понятия одного уровня.
3. На ветвях разместить слова или изображения, иллюстрирующие понятия соответствующего уровня.
4. На всех ответвлениях указывать ключевые слова, которые характеризуют и помогают вспомнить то или иное понятие.
5. Найти все ассоциации (связи) и представить их на карте.
6. Расставить приоритеты (цветом, стрелками).

### Задание 1

Изобразить в виде ментальной карты структуру диска C: компьютера, за которым вы работаете.

**Указание.** Выполнение задания начать с получения на экране списка корневых каталогов, копирования содержимого экрана в графический редактор. Затем подобным образом получить содержимое каталога Пользователи, назначение подкаталогов которого изложить в коротких (несколько слов) текстовых пояснениях.

### Задание 2

Изобразить в виде ментальной карты отношение понятий: компьютер – принтер – клавиатура – мышь – дисплей – сканер – модем, дополнить карту необходимыми составляющими, не указанными в перечне.

### Задание 3

Изобразить в виде ментальной карты отношение понятий: природа – живая природа – неживая природа – биология – физика – химия – астрономия – Марс – Венера – Земля – человек – планеты. Необходимые изображения найти в Интернете.

### Задание 4

Изобразить в виде ментальной карты отношение понятий: компьютерная сеть – компьютер – сервер – роутер – домен, дополнить карту необходимыми составляющими, не указанными в перечне.



## ДЛЯ ЛЮБОЗНАТЕЛЬНЫХ

Всю историю человечества можно считать историей развития и уточнения моделей.

Известно, что Николай Коперник в 1543 г., имея информационную модель движения планет Солнечной системы, поданную в форме таблиц данных, построил математическое описание гелиоцентрической модели Солнечной системы, согласно которой планеты движутся вокруг Солнца по определенным орбитам. Это позволило точнее вычислять местонахождение планет. До этого существовала другая модель, согласно которой все планеты, Солнце и Луна движутся вокруг Земли. Сейчас ученые пользуются еще более точной математической моделью Солнечной системы, благодаря которой можно вычислять массу и законы движения планет.

Таким образом, процесс познания природы, любого явления, с использованием его модели, охватывает следующие этапы.

1. Постановка задачи (описание задачи, цель моделирования, анализ объекта, формализация задачи).
2. Разработка модели (информационная модель, компьютерная модель).
3. Эксперимент с моделью (план эксперимента, проведение исследования).
4. Анализ результатов моделирования (соответствуют результаты цели или нет).
5. Корректировка модели.

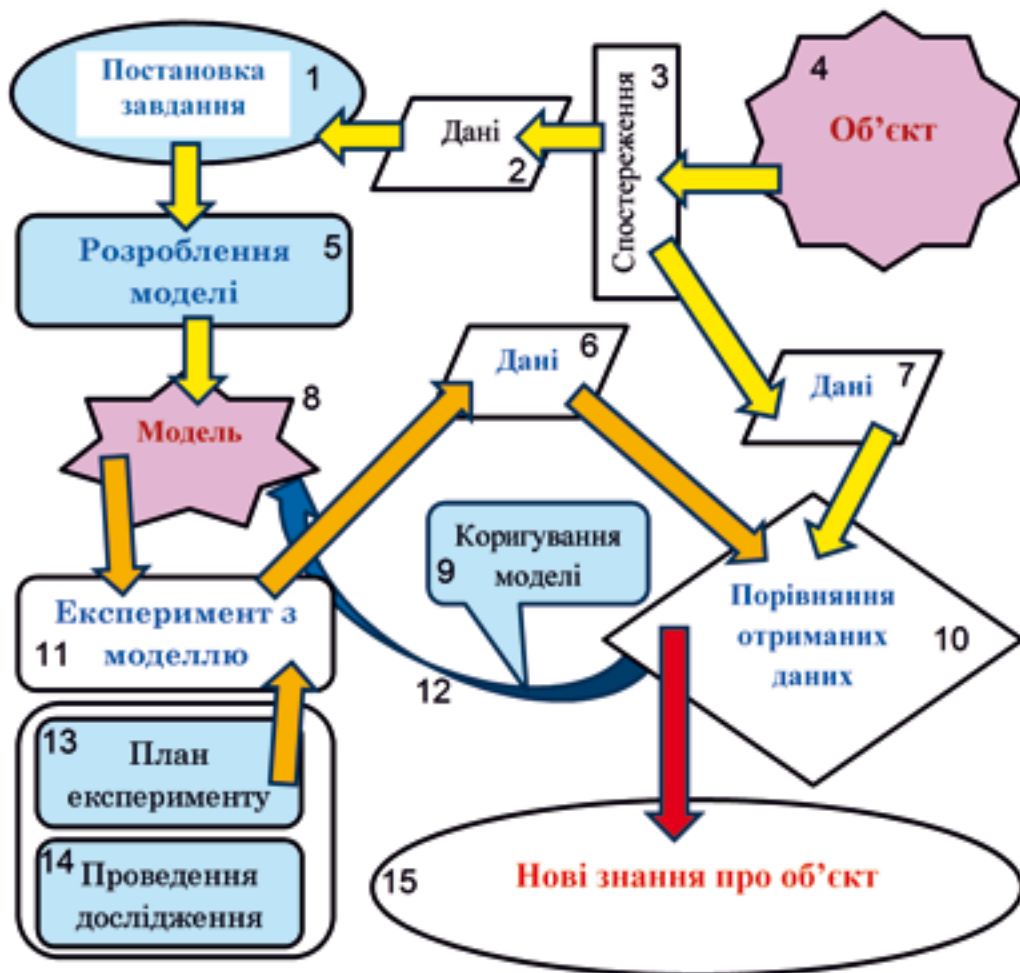


Рис. 2.5. Етапи дослідження об'єкта з використанням його моделі

Ети етапи можна отобразити в виде ментальної карти, котра одночасно буде зображенням алгоритму процесів створення моделі і її використання з метою отримання нових знань об'єкті.

Поскольку обозначения на рис. 2.5 пронумерованы, для описания этапов процесса удобно использовать последовательности их номеров. Такие рисунки довольно часто используют для планирования не только научной деятельности, но и для создания бизнес-планов, планирования схем управления внедрением инноваций и тому подобное.

#### **1-й этап (4,3,2,1). Постановка задачи**

Получение первичных сведений об объекте. Производится анализ объекта. Определение возможного способа постановки задачи. В результате анализа данных формулируются требования к модели. Формулируется цель моделирования. Выявляются существенные факторы, от которых зависит поведение модели. Осуществляется первая попытка с целью определения связей между простыми объектами, входящими в состав моделируемого объекта.

#### **2-й этап (1,5,8). Разработка модели**

По результатам анализа объекта создается информационная модель (совокупность сведений об объекте). Информационная модель должна быть описана в форме совокупности данных, выраженных определенными кодами. Если предполагается осуществить компьютерное моделирование, то информационную модель в большинстве случаев необходимо превратить в математическую.

#### **3-й этап (11 (13,14), 6,10,15). Компьютерный эксперимент**

После создания компьютерной модели проводится ее тестирование. Очень важно предусмотреть все возможные варианты получения результатов, создать план проведения экспериментов (13,14). На этом этапе, как и на последующих, возможна корректировка модели (путь дугой 12), поскольку в процессе исследования почти всегда возникают новые данные, которые требуют уточнения модели.

#### **4-й этап (10,15 и т.д.). Анализ результатов моделирования**

По полученным данным проверяется, насколько результаты соответствуют целям моделирования. Исследователь должен увидеть реальный объект или процесс в его модели, представленной в виде числовых описаний.



## **СЛОВАРЬ**

**Динамическая модель** – модель, созданная таким образом, чтобы можно было изменять некоторые ее свойства и наблюдать за изменениями других.

**Информационная модель** – модель, основной составляющей которой являются сведения о свойствах и состоянии объекта моделирования, процесса, явления.

**Математическая модель** – модель, созданная с помощью символично представленных описаний связей между величинами (уравнения, неравенства, системы уравнений и т.д.).

**Ментальная карта** – обычно статическая, модель некоторого объекта, представленная в форме рисунка и предназначена для иллюстрации его структуры, внутренних связей.

**Модель** – отражение объекта, в котором воспроизведены важнейшие (для определенного применения) его свойства.

**Область знаний** – часть знаний человечества, которая характеризуется определенными объектами изучения, средствами деятельности.

**Статическая модель** – модель, в которой воспроизведен объект в неизменном состоянии.

**Физическая модель** – модель, воссоздана как физический объект (натурная модель).



## РАЗДЕЛ 3. АЛГОРИТМЫ С ПОВТОРЕНИЕМ И РАЗВЕТВЛЕНИЕМ



В шестом классе мы рассматривали два способа представления алгоритмов – словесный и графический, а также учились создавать простейшие проекты в среде Скретч. Словесный и графический способы представления алгоритма понятны человеку. Для того чтобы алгоритм выполнялся компьютером, его нужно описать на специальном языке. Такой язык называется языком программирования.



Базовые алгоритмические структуры: повторение и разветвление. Алгоритмы с повторением. Описание и выполнение алгоритмов с повторением в учебной среде выполнения алгоритмов. Высказывания. Истинные и ложные высказывания. Условное высказывание «если – то – иначе». Алгоритмы с разветвлением. Описание и выполнение в среде Скретч алгоритмов с повторением и разветвлением.

### 3.1. Основные сведения о среде учебного программирования Скретч (повторение изученного)



**Язык программирования** – это язык, предназначенный для описания алгоритмов и данных в виде, пригодном для обработки компьютером.

Язык программирования содержит команды, с помощью которых можно описать алгоритм. Обязательными для многих языков являются команды ввода данных в память компьютера, предоставление переменным определенных значений, вывод результатов на устройства вывода.

Чтобы быть выполненной компьютером, программа, описанная на языке программирования, должна быть преобразована в последовательность команд машинного языка.



**Машинный язык** – это язык, в котором команды и данные записываются комбинацией символов 0 и 1 (машинными кодами).

Для первых компьютеров программист записывал в память компьютера команды и данные в машинных кодах.

Разработка человеком программ в машинных кодах – очень сложный и трудоемкий процесс. Чтобы программист мог быстро описать алгоритм, он должен использовать язык, на котором говорит, по крайней мере, язык программирования должен быть похож на язык человеческого общения.

Такие языки называются **языками высокого уровня**. Сейчас их создано много. Среди первых были разработаны такие языки высокого уровня, как Фортран, Алгол, Паскаль.

Чтобы программы, описанные на языках высокого уровня, могли выполняться компьютером, нужно записать (перевести, **транслировать**) эти программы на машинном языке. Кроме этого, нужны средства для ввода, редактирования, наладки и запуска программ. С этой целью были разработаны специальные программные средства, которые называются **средами (системами) программирования**.

✓ *Среда программирования – это одна или несколько программ, обеспечивающих выполнение всех работ по созданию и выполнению программ.*

К таким работам относятся:

- ввод и редактирование текста программы;
- автоматическое преобразование программы с языка высокого уровня в программу в машинных кодах (**трансляция** программы) и автоматический поиск ошибок;
- настройка программы (пошаговое выполнение команд, просмотр значений переменных и т.п.);
- запуск программ на выполнение;
- наладка системы программирования с учетом потребностей пользователя.

Существует два основных способа **трансляции** программ с языка высокого уровня в машинные коды.

✓ *Компиляция – преобразование программы, описанной командами языка высокого уровня, в программу в машинных кодах.*


При этом создаются одна или несколько программ в машинных кодах, которые могут храниться на внешнем запоминающем устройстве (ВЗУ) (в файле) и запускаться на выполнение пользователем. Программа, которая выполняет такие трансляции, называется **компилятором**.

✓ *Интерпретация – пошаговое преобразование команд языка высокого уровня в последовательности машинных кодов, которые сразу же передаются к исполнению.*

Программа, выполняющая такие преобразования, называется **интерпретатором**.

Одной из систем программирования, предназначенных для обучения, является Скретч (Scratch).



 **Скретч** – это среда программирования, в которой можно создавать программы (проекты), поддерживающие анимации, интерактивные истории, игры, модели и т.д.

Среда Скретч содержит **интерпретатор** языка, текстовый и графический редакторы, справочную систему, образцы проектов, библиотеку рисунков (различных объектов и фонов сцены). Главная страница программы Скретч представлена на рис. 3.1.

Верхняя строка – это заголовок программы. В нем справа расположены три кнопки, предназначенные для свертывания, разворачивания и закрытия окна. Ниже расположены: главное меню, панель инструментов и кнопки для установления размера сцены.

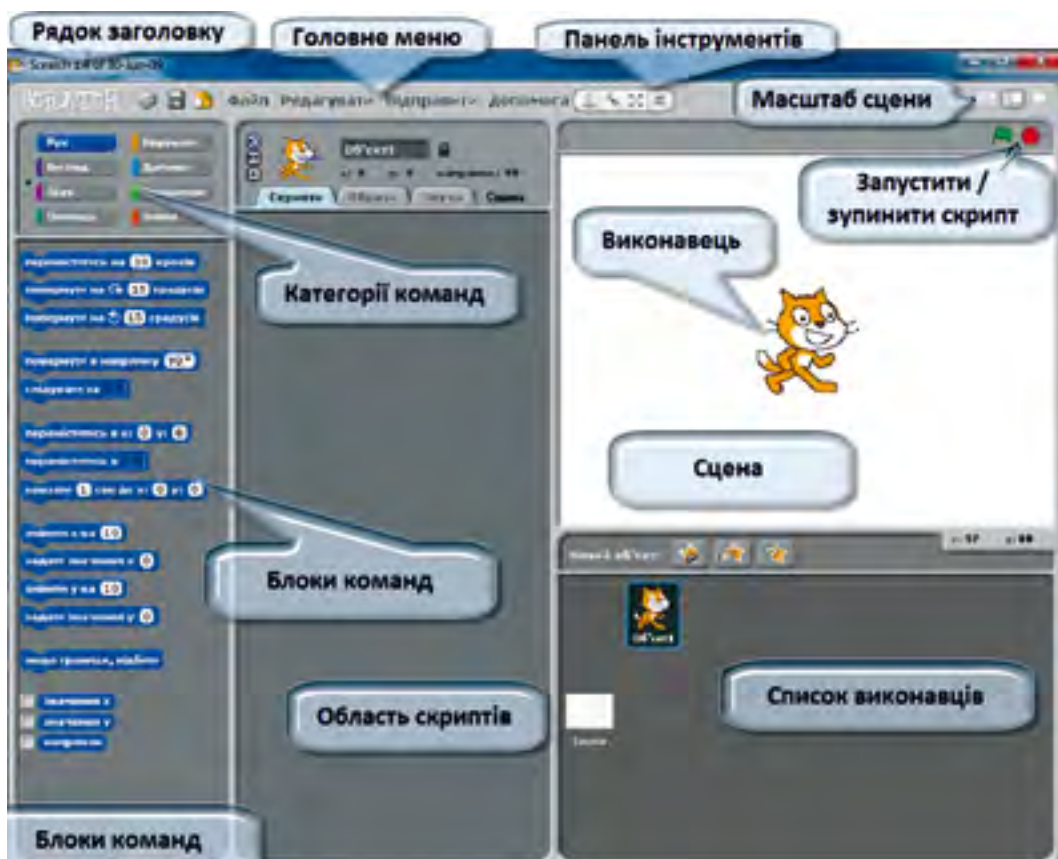


Рис. 3.1. Главная страница программы Скретч

Другая часть окна разделена на три вертикальные полосы, каждая из которых состоит из верхней и нижней частей. Слева сверху находится 8 названий (категорий) кнопок команд (рис. 3.2), одна из которых является активной. Активная кнопка выделена другим цветом.

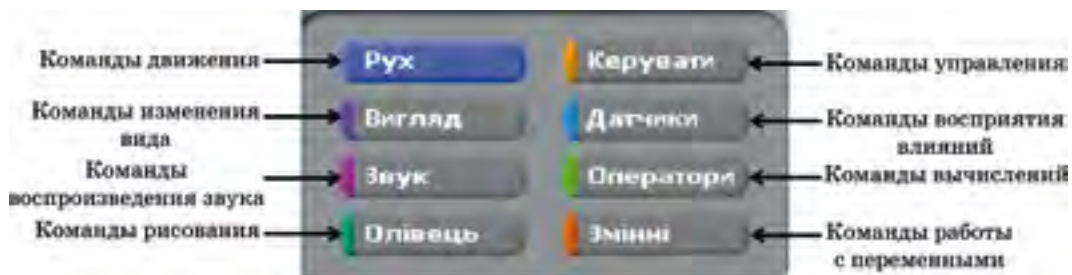


Рис. 3.2. Категории команд Скретч

На рис. 3.2 выделена кнопка **Движение**. Чтобы активировать кнопку, следует нажать на нее левой кнопкой мыши.

Справа вверху (рис. 3.1) белый фон с рыжим котом – это сцена. Сразу после загрузки Скретч сцена всегда имеет светлый фон, на котором находится объект – Рыжий кот. На сцене объекты выполняют запрограммированные действия (двигаются, изменяют размеры, выдают сообщения и т.д.).

✓ *Объект, движущийся по сцене, называют спрайтом.*

На сцене можно разместить несколько объектов. Их месторасположение можно менять. Для этого указатель мыши устанавливают на объект, нажимают левую кнопку мыши и, не отпуская ее, перетягивают объект на новое место. После этого кнопку мыши отпускают. Размер сцены, ее фон и объекты на ней можно менять. Изменить размер сцены можно с помощью кнопок **В уменьшенный размер** и **В полный размер**, расположенных в поле **Размер сцены** (рис. 3.3).

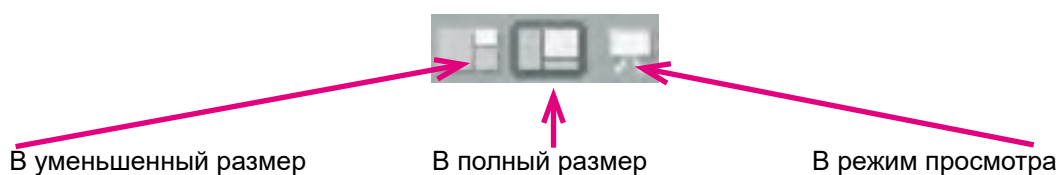



Рис. 3.3. Кнопки для установления размера сцены

При помощи кнопки **В режим просмотра** сцена разворачивается на весь экран. Чтобы вернуться к предыдущему виду сцены, нажимают кнопку , расположенную в левом верхнем углу экрана.

Изменить размер объекта можно с помощью кнопок **Увеличить объект** и **Уменьшить объект** панели инструментов (рис. 3.4). При помощи кнопки **Удалить** объект удаляется со сцены, а с помощью кнопки **Дублировать** на сцене появляется еще один такой же объект.



Рис. 3.4. Кнопки панели инструментов

Рассмотренные действия над объектом можно выполнять также с помощью команд контекстного меню объекта (рис. 3.5). Для открытия этого меню указатель мыши устанавливается на объект и нажимают правую кнопку мыши.

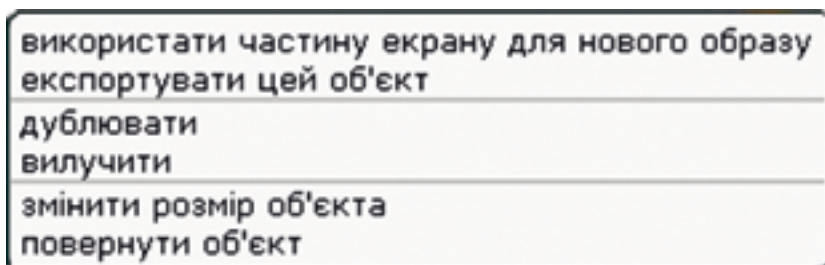



Рис. 3.5. Команды контекстного меню объекта

 *Среда Скретч имеет библиотеку собственных объектов, которые можно размещать на сцене, а также можно создавать собственные объекты с помощью встроенного в среду графического редактора.*

Для этого предназначены кнопки, содержащиеся в поле **Новый объект** (рис. 3.6).

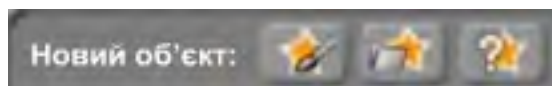


Рис. 3.6. Кнопки для создания и размещения на сцене новых объектов

Все объекты, которые размещаются на сцене, имеют имена: Объект 1, Объект 2 и т. д. Однако им можно давать и другие имена (Коля, Футболист). Изменить можно только имя активного объекта. Активный объект всегда выделен другим цветом. Чтобы объект был активным, достаточно щелкнуть на нем кнопкой мыши. После этого следует установить указатель мыши в поле Объект 1 и при помощи клавиатуры ввести новое имя.

Сцена по горизонтали имеет длину 480 точек (пикселей), по вертикали – 360 точек. Сцена имеет собственную систему координат. Начало координат – в центре сцены (рис. 3.7).

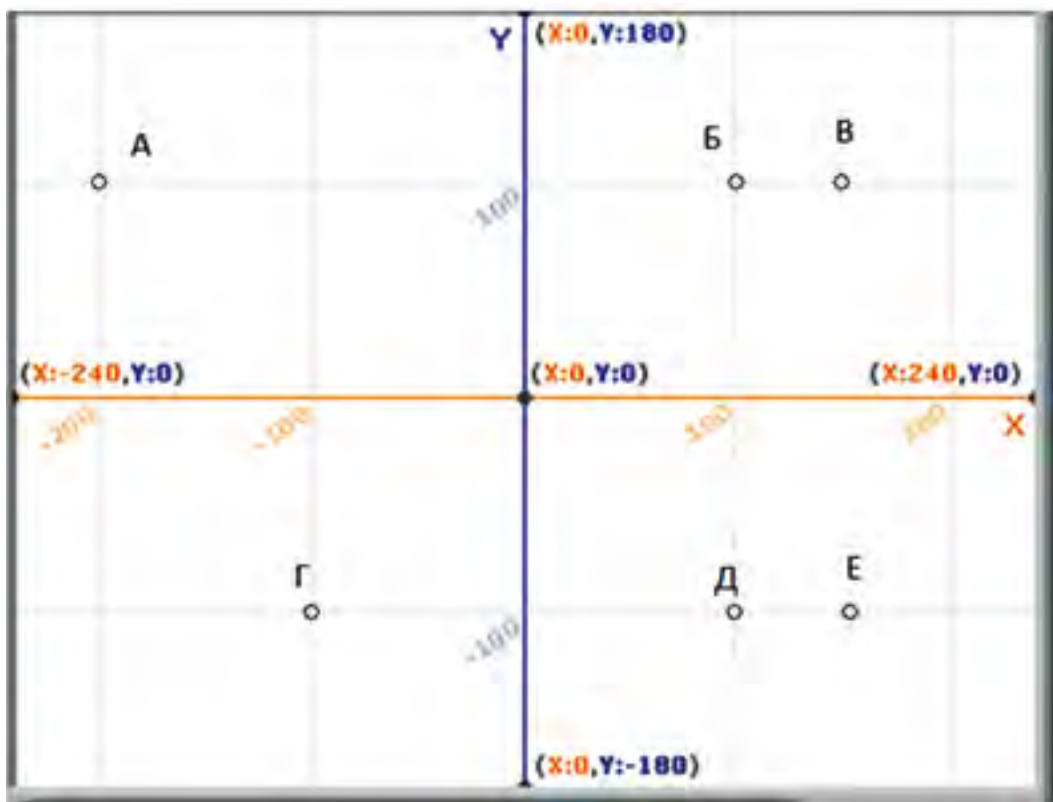


Рис. 3.7. Система координат сцены

В нижней части сцены находится поле, в котором указаны координаты  $x$  и  $y$  положения курсора на сцене. Если положение указателя мыши на сцене меняется, автоматически изменяются и его координаты.

Нижняя часть правой полосы – это поле, в котором содержатся все исполнители, то есть перечень всех объектов проекта, а также сцена. В поле, изображенном на рис. 3.1, есть только два объекта – Рыжий кот и Сцена. Если активна кнопка **Сцена**, в средней части окна сверху появляется окошко с кнопками, которыми можно менять фон сцены и звуки (рис. 3.8).

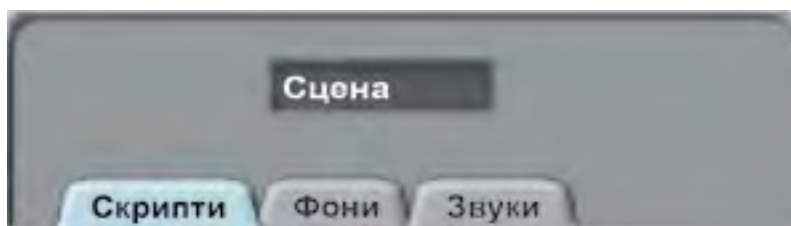


Рис. 3.8. Окошко активной сцены со скриптами

Если сделать активным любой объект, то это окошко приобретет другой вид (как на рис. 3.9). В нем содержатся кнопки, непосредственно предназначенные для работы с объектами и **программами**, которые в среде Скретч называют **скриптами**.



Рис. 3.9. Окошко активного объекта

✓ *Главное меню (рис. 3.10) предназначено для работы с проектом. С помощью его команд можно сохранять проект, отправлять его в Интернет, устанавливать необходимый язык и тому подобное.*

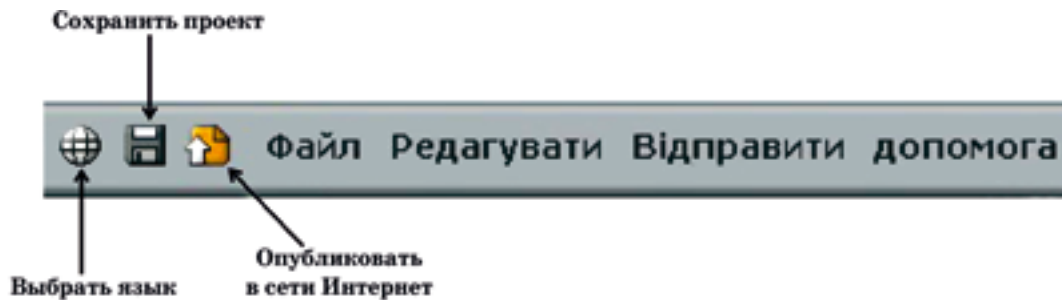


Рис. 3.10. Главное меню

## Выполняем

1. Включите кнопки команд категорий **Движение**, **Управление**, **Вид**. Назначения каких команд вам уже известны? 🌱
2. Удалите со сцены текущий объект (контекстное меню объекта → **Удалить**). Разместите на сцене объект dragon2 папки Fantasy. Для этого нажмите кнопку **Выбрать новый объект** из файла. Откроется окно **Новый объект** (рис. 3.11).



Рис. 3.11. Окно **Новый объект**

В окне содержатся папки объектов **Animals** (животные), **Fantasy** (сказочные персонажи), **Letters** (буквы и цифры), **People** (люди), **Things** (вещи), **Transportation** (транспорт).

Выполните последовательность команд: **Fantasy** → **ОК** → **dragon2** → **ОК**. По аналогии разместите на этой сцене второй объект под названием **fairly** с этой самой папки. Расположите объекты на сцене так, как показано на рис. 3.12. ✦



Рис. 3.12. Два объекта на сцене



3. Увеличьте и уменьшите размеры объектов dragon 2 и fairy, изображенных на рис. 3.12 (контекстное меню объекта → Изменить размер объекта). Появится двойная стрелка. Установите на ней указатель мыши, нажмите левую кнопку мыши и, не отпуская ее, переместите указатель мыши влево или вправо. Одновременно с перемещением указателя будет меняться размер объекта). 🌱

Воспользуйтесь другим способом изменения размера объекта. Для этого нажмите кнопку мыши на кнопке **Увеличить размер** или **Уменьшить размер**. После этого установите курсор на объект и щелкните кнопкой мыши. С каждым последующим нажатием кнопки мыши объект будет увеличиваться или уменьшаться.

4. Уменьшите размер сцены (нажмите кнопку **В уменьшенный размер**). Восстановите предыдущий размер сцены (нажмите кнопку **В полный размер**). Разверните сцену на весь экран (нажмите кнопку **В режим просмотра**). Восстановите прежний вид и размер сцены. 🌱

5. Предоставьте объекту Рыжий кот имя Пусик (установите курсор на поле Объект 1 (рис. 3.9), щелкните кнопкой мыши, введите имя Пусик и нажмите клавишу Enter). 🌱

6. Установите фон сцены flowers папки Nature (Сцена → Фоны → Импортировать). Откроется окно **Импортировать фон** (рис. 3.13), в котором содержатся такие папки фонов: Indoors, Nature, Outdoors, Sports. ✦

В этом окне выберите папку Nature и нажмите кнопку **ОК**. В открывшемся окне выберите фон flowers и нажмите кнопку **ОК**.

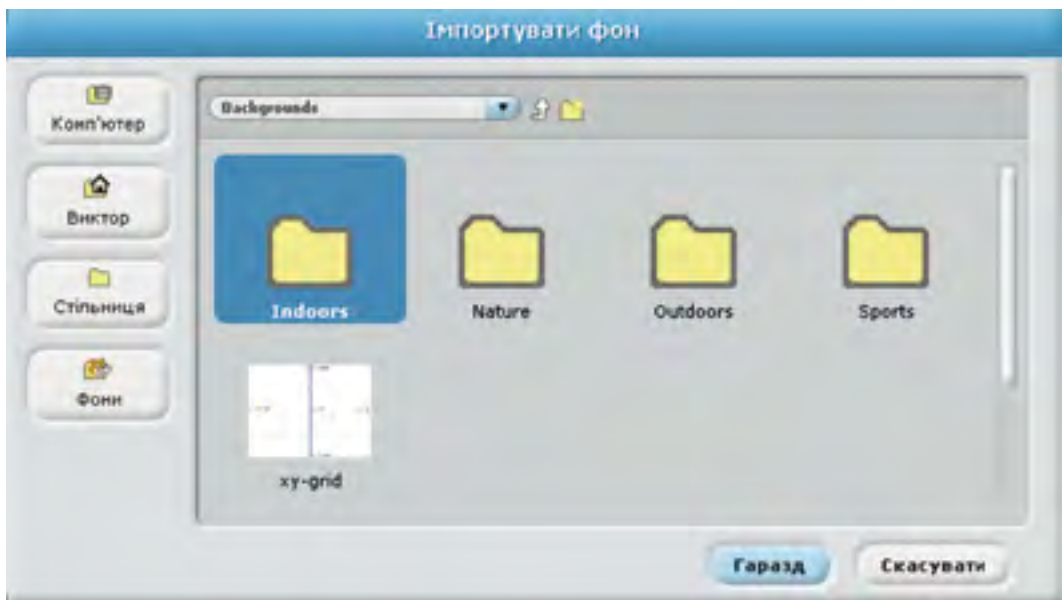




Рис. 3.13. Окно Импортировать фон



7.  Запустите встроенный в Скретч проект Playground, находящийся в папке Animation (Файл → Открыть → Образцы). Откроется окно, представленное на рис. 3.14. 

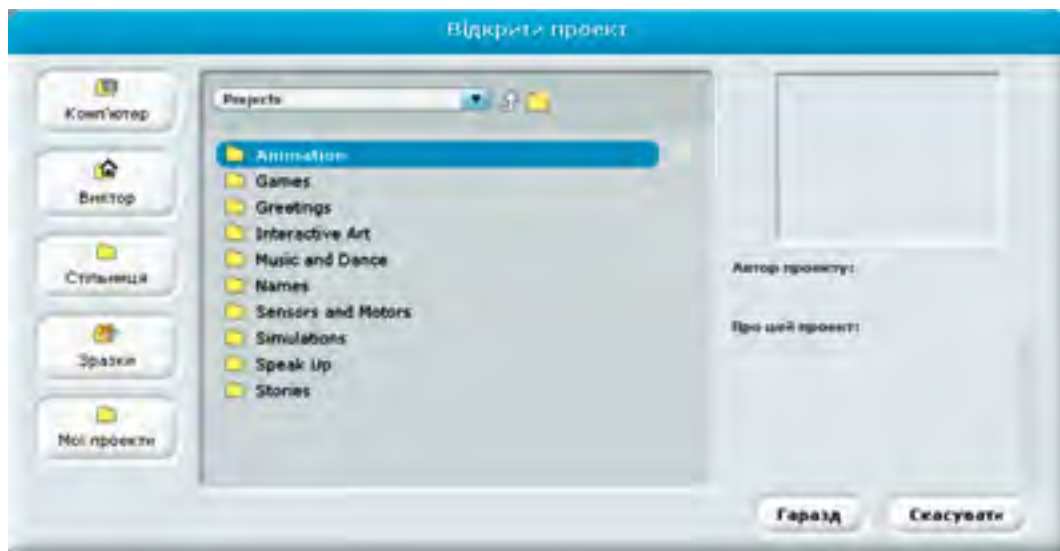


Рис. 3.14. Окно Открыть проект

В этом окне выполняются команды Animation → ОК → Playground → ОК. Откроется проект Playground, содержащий несколько отдельных программ для различных объектов. Для объекта swing программа представлена на рис. 3.15.

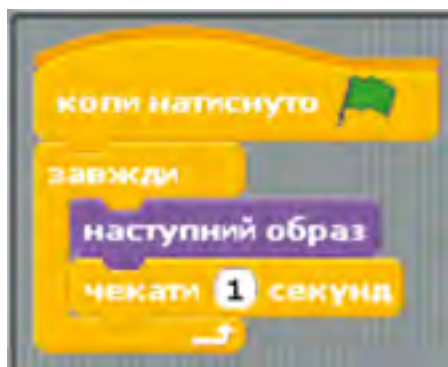


Рис. 3.15. Программа проекта Playground для объекта swing

Если нажать зеленый флажок представленной программы, объект swing будет двигаться непрерывно. Все объекты этого проекта находятся на сцене и изображены на рис. 3.16. Чтобы остановить выполнение программы, нажимают красную кнопку.

Если нажать зеленый флажок среды Скретч (в правом верхнем окне), начнут выполняться все программы проекта Playground, в результате чего будут двигаться все объекты, представленные на рис. 3.16.














Рис. 3.16. Объекты проекта Playground

8. 👤 Поочередно активируйте объекты проекта Playground. На экране будут появляться соответствующие программы. Запустите на выполнение каждую из этих программ отдельно и наблюдайте за действиями, происходящими на сцене. ★
9. 👤 Самостоятельно загрузите и выполните один из проектов, содержащихся на сайте <http://scratch.mit.edu>. После открытия главной страницы сайта нажмите на кнопку треугольника Language и перейдите на украинский язык, для чего щелкните кнопкой мыши на слове Ukrainian. Зарегистрируйтесь, выберите и запустите на выполнение один из проектов. ★
10. 🏠 Опишите словами алгоритм запуска встроенного в Скретч проекта. ✚
11. 🏠 Разработайте графическую схему алгоритма установления нового фона сцены. ✚

### Проверяем себя

1. Какие основные элементы размещены на главном меню Скретч? 👤
2. 👤 Для чего предназначен язык программирования? 👤
3. Что называют средой программирования? 👤
4. Какие основные составляющие содержит среда Скретч? 👤
5. Что называют спрайтом? 👤

6. Какой фон имеет сцена после загрузки Скретч? 
7. Какие размеры может иметь сцена? 
8. Как можно изменить размер объекта? 
9.  Опишите последовательность действий по размещению на сцене нового объекта. 
10. Что называют скриптом? Как его еще можно назвать? 
11.  Объясните назначение кнопки Новый объект. 
12.  Объясните систему координат среды Скретч. 
13. Запишите координаты точек А–Е на рис. 3.7. 

### 3.2. Простейшие алгоритмы с повторением


На практике человеку приходится работать с достаточно сложными алгоритмами. Примерами таких алгоритмов являются посадка авиалайнера, плавление металла в доменной печи, разработка расписания движения поездов на железной дороге, прогнозирование погоды на месяц, вычисление значения сложного математического выражения. При этом любой алгоритм состоит из трех основных структур:

- линейной (следования);
- повторение;
- разветвления.

Линейные структуры, называемые структурами следования, рассматривались в шестом классе.

Рассмотрим сущность второй базовой структуры – алгоритмов с повторением. Отметим, что в жизни, науке, практике мы часто встречаемся с действиями, выполнение которых повторяется неоднократно. Например, подъемный кран неоднократно загружает контейнеры на пароход, водитель трамвая в течение рабочего дня несколько раз водит трамвай по одному маршруту, экскаватор последовательно друг за другом загружает в автомобиль необходимое количество ковшей грунта. Для описания одних и тех же действий, выполняемых многократно, используются алгоритмы с повторением.

Примером алгоритма с повторением является умножение заданного числа на последовательность натуральных чисел от 1 до 10.

 **Алгоритм с повторением** – это алгоритм, в котором одна и та же последовательность команд выполняется подряд более одного раза.

Общая структура алгоритма с повторением изображена на рис. 3.17.

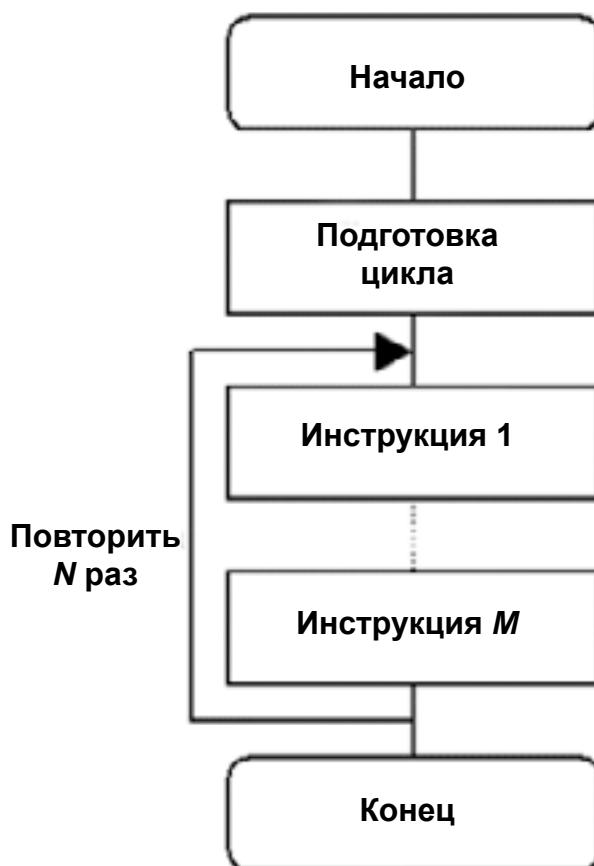


Рис. 3.17. Общая структура алгоритма с повторением

✓ Алгоритмы с повторением называют **циклическими**.

Примером алгоритма с повторением является нахождение суммы чисел натурального ряда. В словесной форме алгоритм нахождения суммы чисел от 1 до 6 можно записать так:

1. Начало.
2. Текущее число равно 0.
3. Текущая сумма равна 0.
4. Увеличить текущее число на единицу.
5. К текущей сумме добавить текущее число.
6. Повторить пункты 4 и 5 шесть раз.

В графической форме этот алгоритм представлен на рис. 3.18 ( $S$  – текущее значение суммы,  $a$  – текущее значение числа).

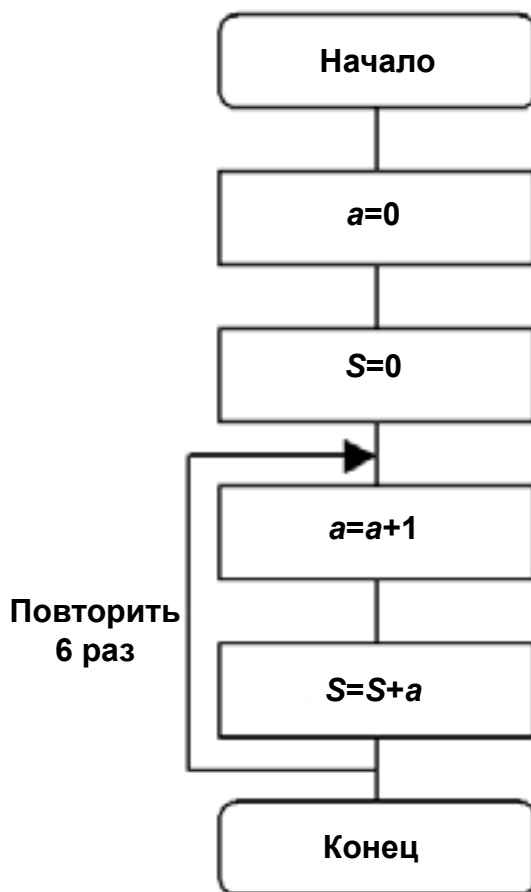



Рис. 3.18. Алгоритм определения суммы чисел натурального ряда

 **Инструкции (команды), которые повторяются, называются телом цикла. Циклом называют одновременное выполнение команд тела цикла.**

Для реализации алгоритмов с повторением в среде Скретч имеются две простые команды (рис. 3.19).



Рис. 3.19. Блоки команд повторения

*Инструкции, которые содержит команда **всегда**, выполняются бесконечно, а инструкции команды **повторить** – указанное количество раз.*

**Пример.** На рис. 3.20 вертолет (объект helicopter1, папка Transportation) находится на фоне сцены hay\_field (папка Outdoors).



Рис. 3.20. Вертолет над полем

Вертолет кружит над полем. Программа моделирования его движения представлена на рис. 3.21.

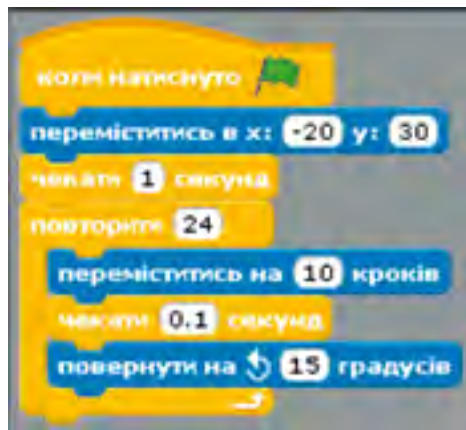


Рис. 3.21. Программа движения вертолета

✓ Телом цикла в представленной программе являются блоки команд: «переместиться на 10 шагов, ждать 0,1 секунды и повернуть на 15°».

**Пример.** Попугай (объект parrot1-а, папка Animals) непрерывно летает от левой границы каньона к правой и обратно (фон canyon, папка Nature). Фрагмент полета представлен на рис. 3.22.



Рис. 3.22. Попугай над каньоном



Для разработки программы моделирования полета попугая размещаем на сцене объект parrot1-a, затем импортируем образ parrot1-b. Программа, управляющая полетом попугая, представлена на рис. 3.23. Команда **наступний образ** поочередно выбирает образы parrot1-a и parrot1-b. Тем самым создается впечатление, что попугай машет крыльями. Команда **якщо границя, відбити** обеспечивает поворот попугая от края поля. Чтобы после отражения от края сцены попугай не переворачивался, следует нажать кнопку **Принимать только слева направо** (рис. 3.9).

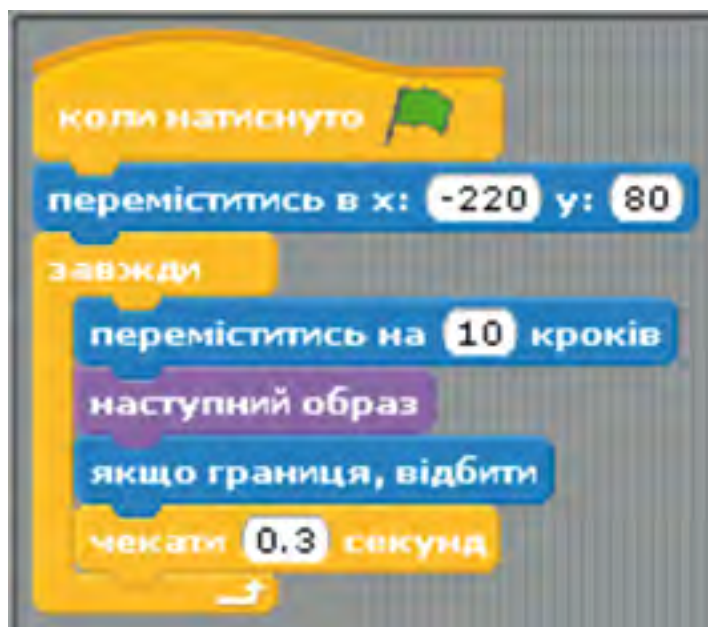



Рис. 3.23. Программа полета попугая

Среда Скретч позволяет создавать и одновременно выполнять для одного объекта проект, содержащий несколько программ.

Одновременное выполнение в компьютере двух и более программ способствует повышению эффективности использования его аппаратных средств и производительности.

 *В теории программирования технологию одновременного выполнения нескольких программ называют **параллельным программированием**.*

**Пример.** В бассейне (фон pool, папка Outdoors) от его левого края к правому плывет объект fantasy5, папка Fantasy (рис. 3.24).



Рис. 3.24. Объект fantasy5 плавает в бассейне

Каждую секунду случайным образом меняется цвет объекта. Объект *fantasy5* спустя некоторое время останавливается в бассейне, а его цвет продолжает изменяться. Изменение цвета объекта осуществляется с помощью команд **Установить эффект** и **Очистить графические эффекты**. Программы проекта, реализующие моделирование описанного выше процесса, представлены на рис. 3.25.



Рис. 3.25. Программы проекта

Первая программа описывает процесс перемещения объекта в бассейне, а вторая – изменение его цвета. Каждую программу проекта можно запустить отдельно, нажав на соответствующую программе кнопку зеленого флажка. Для одновременного выполнения обеих программ проекта нажимают кнопку зеленого флажка над сценой. Для сохранения проекта открывают меню **Файл**, список команд которого представлен на рис. 3.26.

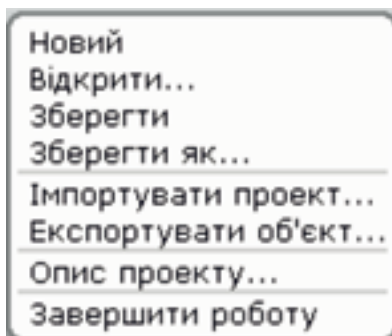


Рис. 3.26. Список команд меню **Файл**

В этом меню выполняют команду **Сохранить как...**. Откроется окно **Сохранить проект**, в котором нажимают кнопку треугольника в поле Scratch Projects.

В результате откроется перечень внешних запоминающих устройств. Выбираем, например, диск **F:**. Окно примет вид, примерное содержание которого представлено на рис. 3.27.



Рис. 3.27. Окно **Сохранить проект**

В поле **Новое имя файла** вводим имя, например, *Проект1*, в поле **Об этом проекте** – его характеристику, например, *в бассейне плавает объект fantasy5*, а в поле **Автор проекта** – имя автора. После этого нажимаем кнопку **ОК**. В результате *Проект1* будет сохранен в корневом каталоге диска **F:**. Для открытия проекта открывают меню **Файл**, выполняют команду **Открыть...**, находят искомое имя файла и нажимают кнопку **ОК**.

В открытый проект можно вносить изменения, а для его сохранения нажимают кнопку **Сохранить**.



Рис. 3.28. Объекты на сцене Wooden-house (деревянный дом)

### Выполняем

1. Удалите все объекты со сцены. Установите фон сцены **Wooden-house** с папки **Outdoors** (**Сцена** → **Фоны** → **Импортировать** → **Outdoors** → **ОК** → **Wooden-house** → **ОК**). Разместите на сцене объекты **horse1-b** и **cat4** (папка **Animals**), как показано на рис. 3.28. Предоставьте объекту **horse1-b** имя **Барс**, а объекту **cat4** – имя **Кукс**. 🐾

2. На рис. 3.29 представлена программа движения объекта на сцене. Разработайте графическую схему алгоритма, который реализует эту программу.

Выполните программу и убедитесь, что алгоритм разработан правильно. ✚

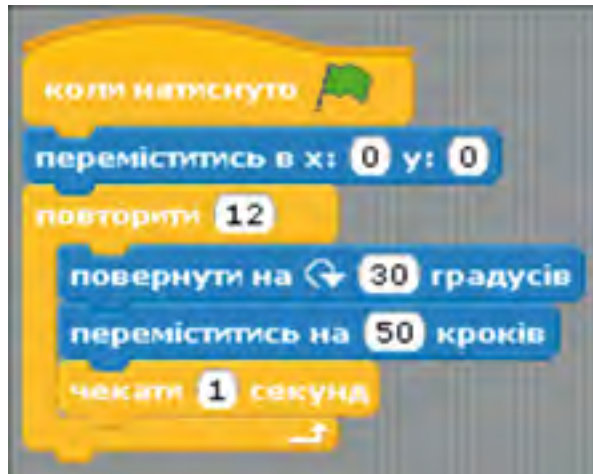


Рис. 3.29. Программа движения объекта

3. Программа, представленная на рис. 3.30, моделирует перемещения объекта от центра сцены слева направо на 80 шагов, затем объект возвращается и движется вверх на 100 шагов. После этого он говорит: «Конец».

Выполните программу и убедитесь, что она функционирует правильно. ★

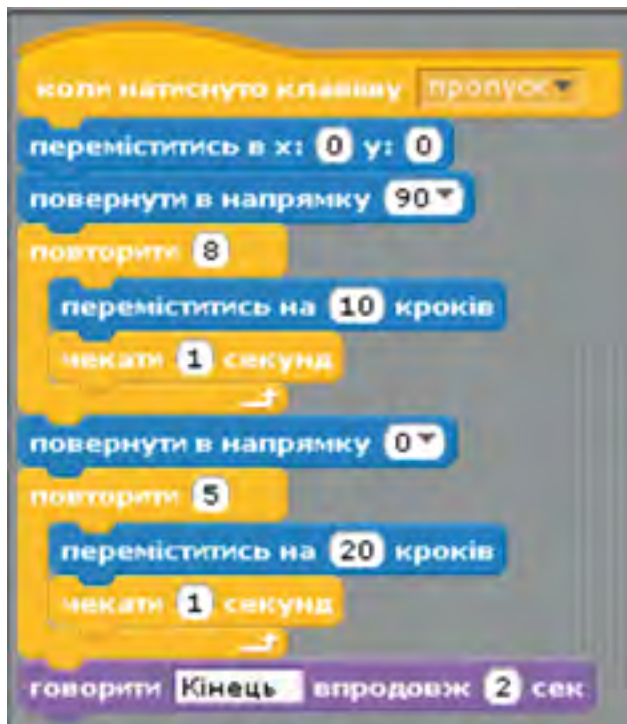






Рис. 3.30. Программа изменения положения объекта



4.  На сцене с фоном grand-canyon (папка Nature) находится объект lion1-b (папка Animals). Из центра сцены он движется вниз на 15 шагов, останавливается на 1 с, затем еще 5 раз повторяет такое же движение. После этого он рычит – rrrrrr. Разработайте программу моделирования этого процесса. ✦
5.  Объект gobo2 (папка Fantasy) с точки сцены с координатами  $x = -50, y = 50$  осуществляет движение по квадрату длиной 100, возвращаясь в начальную точку. На каждой грани квадрата меняется цвет объекта. В каждом углу квадрата объект останавливается на 1 с. Разработайте для этого объекта две программы. Первая программа моделирует движение объекта, а вторая – изменяет его цвет. ★
6.  Пять учеников соревновались в скорости набора символов на клавиатуре. Первый за одну минуту напечатал 81 символ, второй – 78, третий – 75, четвертый – 72, пятый – 69 символов. Разработайте графическую схему алгоритма подсчета общего количества напечатанных символов. ✦
7.  Четверо детей собирали грибы. Первый ребенок собрал 1,6 кг грибов, а каждый следующий – на 0,6 кг больше предыдущего. Дети ссыпали грибы вместе и поделили их между собой поровну. Разработайте графическую схему алгоритма определения массы грибов, которая досталась каждому ребенку. ★

### Проверяем себя

1. Какие команды используются в среде Скретч для описания алгоритмов с повторением? 🟢
2. Для чего предназначена команда **Следующий образ**? 🟢
3. Как осуществляется сохранение проекта? 🟢
4. Как запускается проект в среде Скретч? 🟢
5. Дайте определение алгоритма с повторением. ✦
6. Что называют телом цикла? ✦
7. Что называют циклом? ✦
8. Как осуществляется сохранение проекта в среде Скретч? ✦
9. Нарисуйте общую структуру алгоритма с повторением. ★
10. Приведите пример алгоритма с повторением. ★
11. Объясните назначение кнопки **Принимать только слева направо**. ★
12. Что понимают под термином «параллельное программирование»? ★




### 3.3. Типы данных среды Скретч

В современных языках программирования обрабатываются звуковые, графические, символьные и другие типы данных. В среде Скретч также используются различные типы данных.


**Графические данные** в среде Скретч – это изображение объектов (спрайтов) и фона сцены, содержащиеся в самой среде.

Графические объекты можно создавать также при помощи встроенного графического редактора.

**Звуковые данные** содержатся в библиотеке среды, могут записываться с микрофона или импортироваться из других источников.


 **Числовые данные.** В Скретч могут использоваться целые и дробные числа. Целая часть отделяется от дробной части точкой, например, 234.75. Числа можно вводить с клавиатуры в поля соответствующих команд.

Например, в поля команды **Переместиться** в можно записать такие числа: .

 *Над числами можно выполнять такие операции.*

1. **Арифметические** (сложение, вычитание, умножение, деление). Блоки команд соответствующих операций находятся в категории **Операторы**. В овальные окошки этих команд можно вводить числа, имена переменных, выражения, элементы списка. Эти блоки команд самостоятельно не используются (их называют репортерами).



Они вставляются в поля других блоков команд, например, в поля блоков стека: .

На рис. 3.31 представлена программа, в результате выполнения которой объект сначала поворачивается на  $80^\circ$ , затем перемещается на 50 шагов.

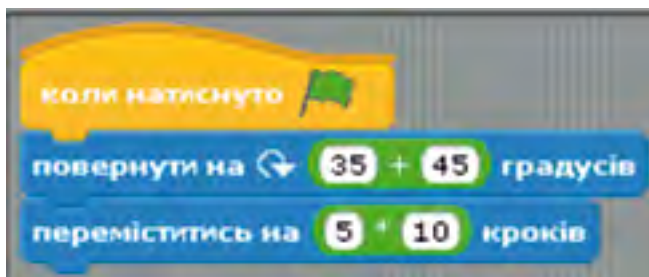
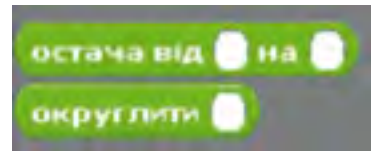


Рис. 3.31. Программа перемещения объекта

2. Операция *округления дробей* до ближайшего целого числа и операция *нахождения остатка* от деления целых чисел.

Блоки команд соответствующей операции отображены в категории **Операторы**.



Например, в результате каждого выполнения программы, представленной на рис. 3.32, объект перемещается в центр сцены и поворачивается на угол  $5^\circ$ . Значение угла поворота в градусах указывает показатель **Направление**.

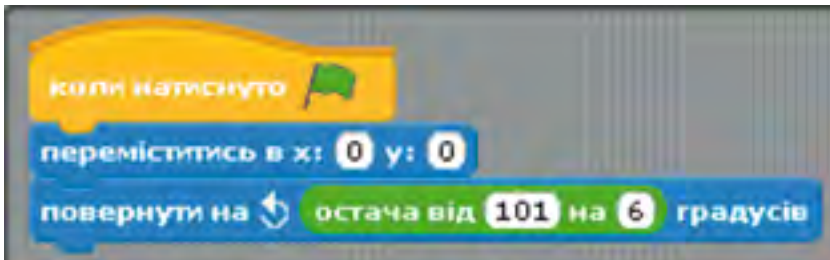


Рис. 3.32. Программа изменения направления движения объекта

3. Операция получения *случайного числа* в заданном диапазоне. Блок команды этой операции имеет следующую структуру:



Например, в результате каждого выполнения программы, представленной на рис. 3.33, объект будет перемещаться на сцене в точку, в которой находится указатель мыши, а затем, через 1 с – на случайное количество шагов в диапазоне от 20 до 50.

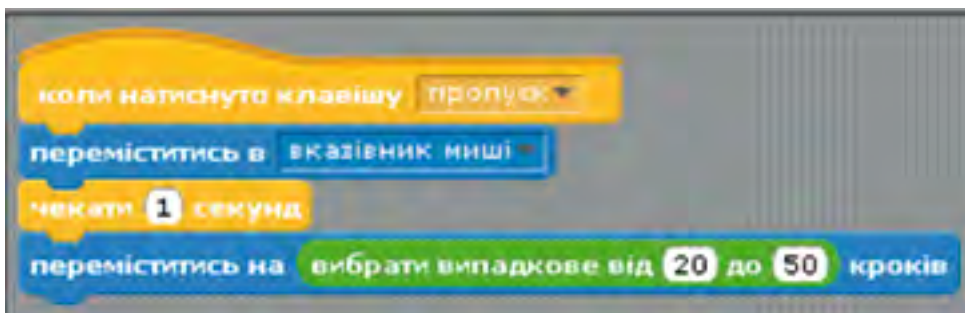



Рис. 3.33. Программа перемещения объекта на случайное расстояние

 **Данные строчного типа.** Эти данные в Скретч состоят из букв алфавита, цифр и специальных символов (#, &, +, :).

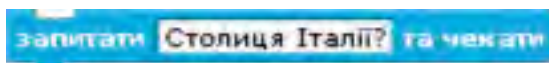
Строчные данные могут содержаться в полях четырех команд категории **Вид**.

Пример заполнения полей этих команд представлен на рис. 3.34.



Рис. 3.34. Команды для ввода строчных данных

Строчные данные можно вводить также в поле команды **спросить** категории **Датчики**:



✓ Над строчными данными можно выполнять следующие операции:

- соединить символы в одну строку;
- определить символ строки по месту его расположения;
- определить длину строки.

Блоки команд этих операций содержатся в категории **Операторы** и представлены на рис. 3.35.

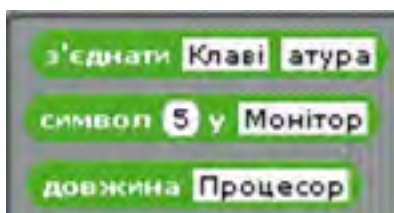


Рис. 3.35. Команды выполнения операций над строчными данными

Например, в результате выполнения программы, представленной на рис. 3.36, на сцене возле объекта появится слово «принтер» (рис. 3.37).

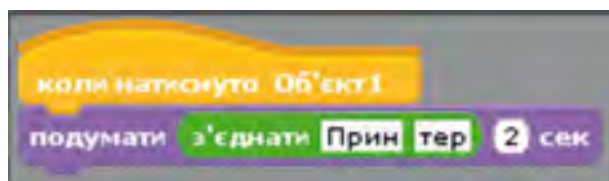


Рис. 3.36. Программа генерирования слова «принтер»



Рис. 3.37. Результат выполнения программы

После запуска программы, представленной на рис. 3.38, возле объекта в течение 2 с подсвечивается цифра 7.

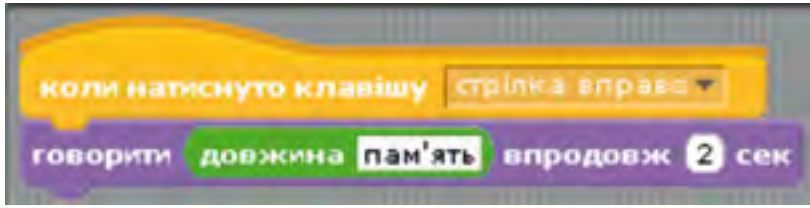




Рис. 3.38. Программа определения длины слова



 **Данные логического типа бывают двух значений: «истинно» и «ложно».**

Эти значения возникают в результате выполнения действий, описанных логическими выражениями.

 **Логическое выражение может содержать операции сравнения: больше ( $>$ ), меньше ( $<$ ), равно ( $=$ ).**



Блоки команд для этих операций находятся в категории **Операторы**.

Результат выполнения этих команд зависит от значений данных в их полях. Например, результат выполнения команды  имеет значение «истинно», а результат выполнения команды  – «ложно».

Показанные блоки команд имеют форму шестиугольников. Они самостоятельно не используются и вставляются в поля такой же формы других команд.


 **Над логическими данными можно выполнять логические операции: и (*and*), или (*or*), нет (*not*). В табл. 3.1 приведены результаты выполнения этих операций над логическими величинами  $x$  и  $y$**


Таблица 3.1

$x$	$y$	$x$ and $y$	$x$ or $y$	not $x$
ложно	ложно	ложно	ложно	истинно
ложно	истинно	ложно	истинно	истинно
истинно	ложно	ложно	истинно	ложно
истинно	истинно	истинно	истинно	ложно

Результат операции **и (and)** имеет значение «истинно», если оба логические выражения имеют значение «истинно».


Результат операции **или (or)** имеет значение «истинно», если значение «истинно» приобретает хотя бы одно из выражений.


Операция **нет** (**not**) выполняется только над одним логическим выражением и возвращает значение, противоположное его значению.


 Логические операции в Скретч реализуются такими блоками команд категории **Операторы**:



Примеры логических операций в среде Скретч и результаты их выполнения:

 – «истинно»;

 – «ложно»;

 – «ложно».

Последовательность выполнения операций в последнем примере такова: сначала выполняется операция  $3 = 3$  (результат – «истинно»), затем – операция  $47 < 52$  (результат – «истинно»), далее – операция **или** (результат – «истинно») и наконец – операция **нет** (результат – «ложно»).

### Выполняем

1. Проанализируйте программу, представленную на рис. 3.39, и определите слово, которое произносит объект. Выполните программу и проверьте правильность своего ответа. 🧑

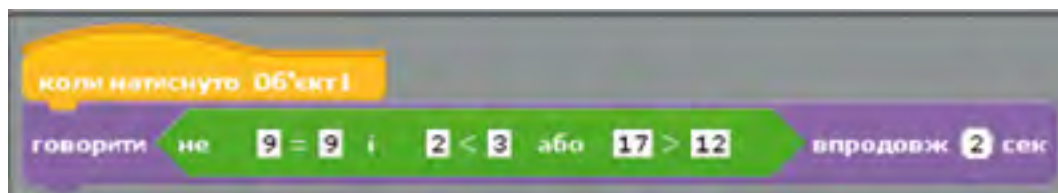


Рис. 3.39. Программа с логическим выражением

2. Заполните пустые поля программы, представленной на рис. 3.40, так, чтобы результат приобретал значение «истинно». Выполните программу и убедитесь, что вы правильно заполнили поля. 🧑





Рис. 3.40. Незавершенная программа с логическим выражением

3. Докажите, что независимо от значений пустых полей команды **подумать** программы, представленной на рис. 3.41, результат всегда будет иметь значение «истинно». Выполните программу и убедитесь в этом. ✚



Рис. 3.41. Программа для самостоятельного анализа

4. Определите, при каком значении пустого поля команды **говорить** программы, представленной на рис. 3.42, выражение будет иметь значение «истинно». Выполните программу и проверьте свою гипотезу. ★

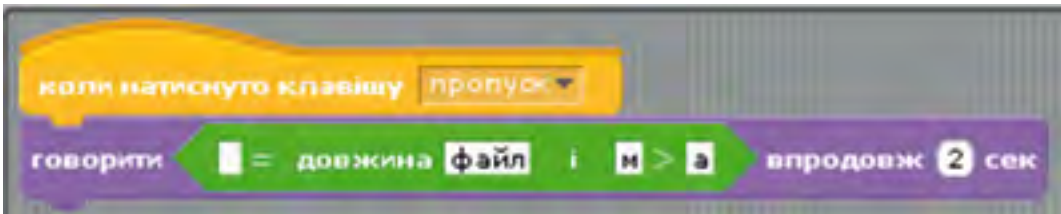


Рис. 3.42. Программа для определения значения пустого поля

5. 👤 Определите, какие сообщения появятся возле объекта после выполнения программы, представленной на рис. 3.43. ✚

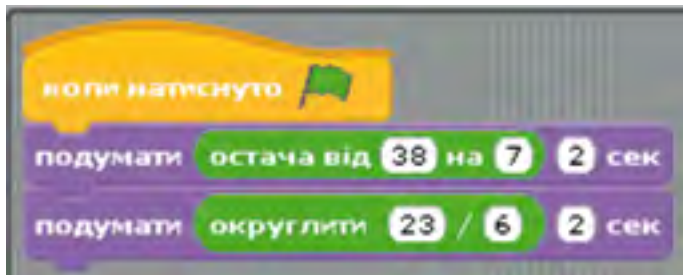



Рис. 3.43. Программа определения значения остатка и округления чисел



6.  Выполните программу, представленную на рис. 3.44. Проанализируйте числа, генерируемые программой. Соответствуют ли их значения содержанию программы? ★

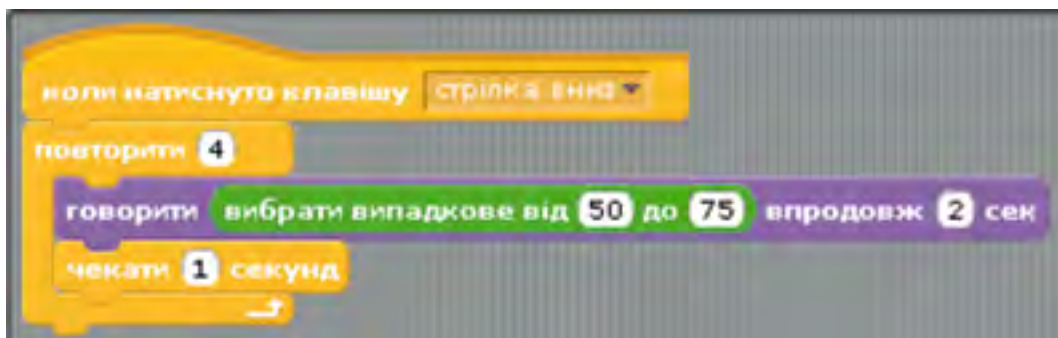



Рис. 3.44. Программа генерирования случайных чисел

7.  Разместите на сцене объект baby. Импортируйте объекты appa-1 и ballerina-c. Для объекта baby создайте программу, представленную на рис. 3.45. Выполните несколько раз программу и наблюдайте, как изменяются объекты, появляющиеся на сцене. Убедитесь, что объекты появляются на сцене в случайной последовательности. ★

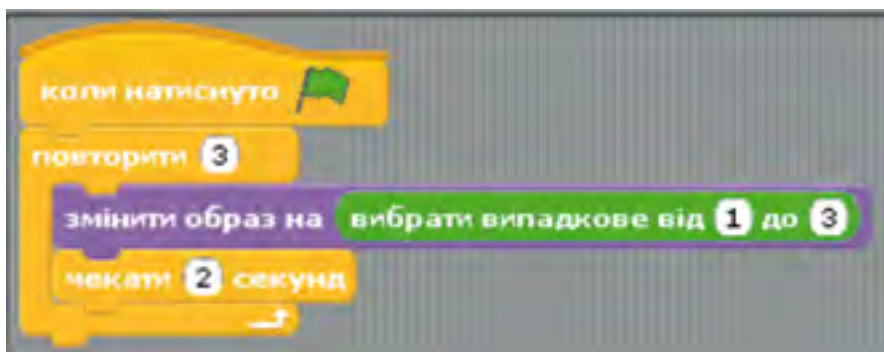



Рис. 3.45. Программа выбора случайного образа

8.  Разработайте программу, по которой объект из центра сцены четыре раза перемещается на случайное количество шагов в диапазоне от 10 до 40 через каждые 2 секунды. +

### Проверяем себя

1. Какие числовые данные используются в среде Скретч? 🌱
2. Какие операции могут выполняться над числами в среде Скретч? 🌱
3. Из чего состоят строчные данные в среде Скретч? 🌱
4. Какие значения принимают логические данные? 🌱

5. Приведите пример команды, в которой используются арифметические операции над числами. ✦

6. Какие операции могут выполняться над строчными данными в Скретч? ✦

7. Какое слово скажет объект после выполнения команды

говорити з'єднати зображення впродовж 2 сек ? ✦

8. Какие операции могут выполняться над логическими данными? ✦

9. Какой символ назовет объект после выполнения команды

говорити символ 4 у довжина впродовж 2 сек ? ✦

10. На сколько шагов переместится объект после выполнения команды

переміститись на остача від 53 на 8 кроків ? ★

11. На сколько шагов переместится объект после выполнения команды

переміститись на округлити 57 / 9 кроків ? ★

12. Что скажет объект после выполнения команды

говорити довжина клавіатура впродовж 2 сек ? ★

13. Какое значение имеет выражение  $5 < 4$  або  $7 > 19$  ? ★

14. Какое значение имеет выражение  $7 = 7$  і  $18 < 5$  ? ★

### 3.4. Константы, переменные и списки

✓ Данные, используемые в программе, сохраняются в памяти компьютера, для чего выделяются определенные ее участки. В процессе выполнения программы в одних участках данные могут неоднократно изменяться, в других – нет.

Данные в тех участках, которые в процессе выполнения программы не изменяются, называются **константами**, а в тех, которые изменяются, – **переменными**.

В среде Скретч константы, как числовые, так и строчные, пользователь может вводить в поля некоторых команд. Кроме этого, в некоторых командах могут использоваться собственные константы самой среды Скретч (их называют **зарезервированными константами**). Например, в поле команды **установить эффект** можно установить зарезервированные строчные константы «цвет», «вздутие» и др. (рис. 3.46).

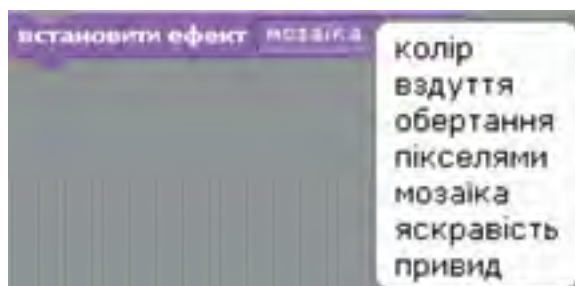



Рис. 3.46. Зарезервированные строчные константы команды установить эффект

 **Переменная в программировании** – это участок памяти с присвоенным ему именем, в который можно записывать различные значения данных.

Переменные в среде Скретч пользователь создает в категории **Переменные**. Они могут быть как локальными, так и глобальными. Локальные переменные связаны только с одним (текущим) спрайтом, а глобальные – со всеми спрайтами проекта. Для создания переменной выполняют команду **Создать переменную** в категории **Переменные**. В результате откроется окно, изображенное на рис. 3.47.

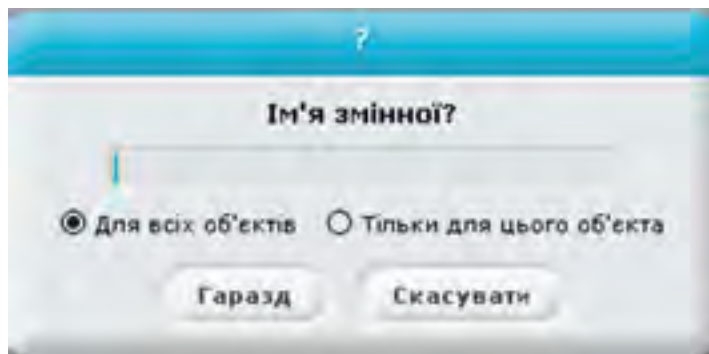


Рис. 3.47. Окно для ввода имени и типа переменной

В пустое поле окна необходимо ввести имя. Имя может состоять из одной или нескольких букв, цифр и знаков подчеркивания ( \_ ). В именах можно использовать как латинские, так и кириллические буквы. Примеры имен переменных: *a\_1*, *refals*, *гипотенуза*. Если проект планируется разместить в Интернете, необходимо использовать только латинские буквы. Для локальной переменной включается переключатель **Только для этого объекта**, а для глобальной – **Для всех объектов**. После ввода имени переменной нажимают кнопку **ОК**. В результате в категории **Переменные** высветятся команды для работы с переменными, приведенные на рис. 3.48.

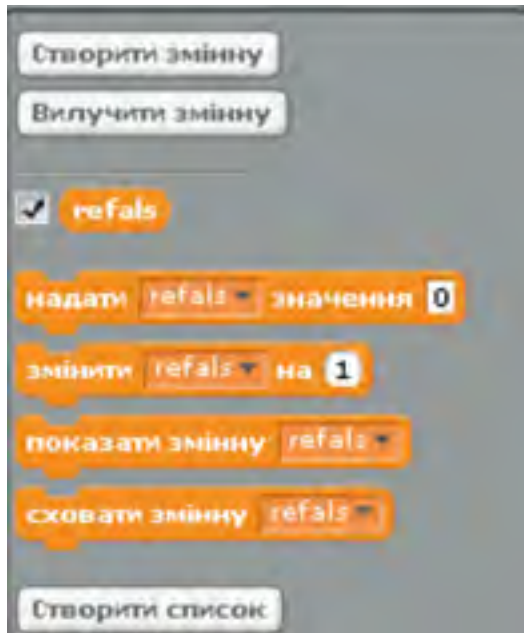


Рис. 3.48. Команды для работы с переменными

✓ Обратите внимание на то, что слева от имени переменной расположен прямоугольник. Если флажок в нем включен, то на сцене высвечивается окошко-монитор с текущим значением переменной.

Мониторы переменных можно показывать или скрывать программно с помощью соответствующих команд категории **Переменные**. Вид монитора переменной можно изменять при помощи команд его контекстного меню. Содержание меню представлено на рис. 3.49. Для открытия контекстного меню нажимают правую кнопку мыши на мониторе соответствующей переменной.

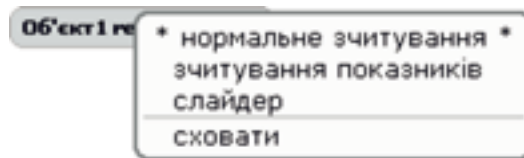


Рис. 3.49. Контекстное меню монитора переменной

Слайдер позволяет при помощи ползунка плавно изменять значение переменной. С помощью команды **нормальное считывание** в окошке-мониторе отображается имя переменной и ее значение, а с помощью команды **считывание показателей** – только ее значение.

Для сообщения значения переменной можно использовать команду **спросить и ждать** и репортер **ответ**, содержащиеся в категории **Датчики**. Команда **спросить и ждать** выводит на сцену текст, содержащийся в ней,

приостанавливает выполнение программы и выводит на сцену специальное поле для ввода данных при помощи клавиатуры. После ввода данных (чисел или текста) нажимают клавишу Enter. В результате данные попадают в репортер **ответ**. Затем данные с репортера можно присвоить другой переменной с помощью команды **присвоить значение** категории **Датчики**. Итак, команда **спросить и ждать** и репортер **ответ** работают вместе.

На рис. 3.50 представлен пример программы присваивания переменной **a\_1** значения, которое вводится с клавиатуры. После запуска программы на сцене возле объекта появится сообщение **Введите значение переменной a\_1**, а внизу сцены – поле для ввода данных. Введем, например, текст «Играю в шахматы» и нажмем клавишу Enter. В результате на сцене появятся имя переменной и введенный текст.

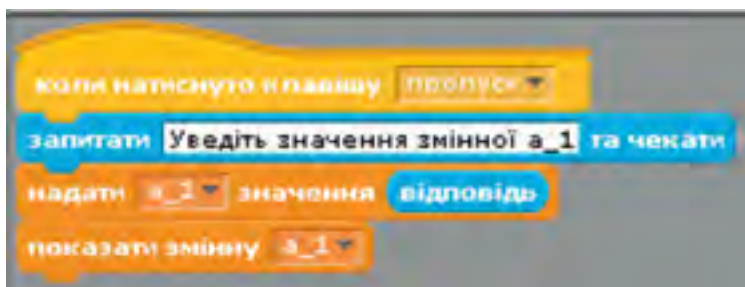



Рис. 3.50. Программа для ввода значения переменной

На рис. 3.51 представлена программа, с помощью которой с клавиатуры вводятся два числа, складываются и результат выводится на сцену.

В среде Скретч кроме простых переменных применяется также структурированный тип данных – списки.

 **Список** состоит из определенной совокупности чисел или строк. Он имеет имя. Каждое значение списка имеет свой порядковый номер (индекс).

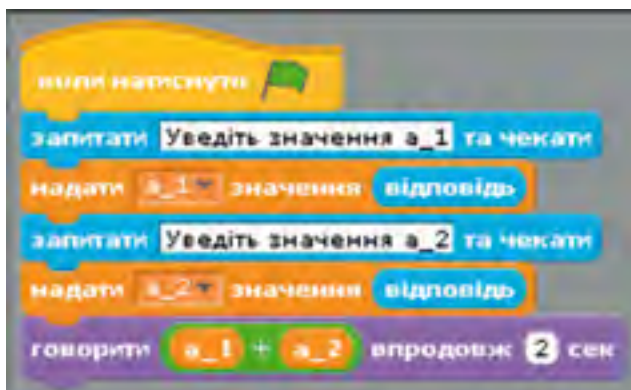


Рис. 3.51. Программа сложения двух чисел



Например, список уроков в понедельник может иметь имя **Уроки** и содержать такие предметы: *Физика, История, Информатика, Физкультура*. В этом списке предмет *Физика* имеет первый номер (индекс), *История* – второй и т. д. К каждому элементу списка можно обращаться по его номеру. Например, по номеру 3 можно обратиться к предмету *Информатика*.

Списки, как и переменные, могут быть **локальными и глобальными**. Для создания списка нажимают кнопку **Переменные**, затем кнопку **Создать список**, вводят имя списка и нажимают кнопку **ОК**.

Создадим, например, список с именем **Друзья** локального типа. После его создания появятся команды для работы со списком, представленные на рис. 3.52.

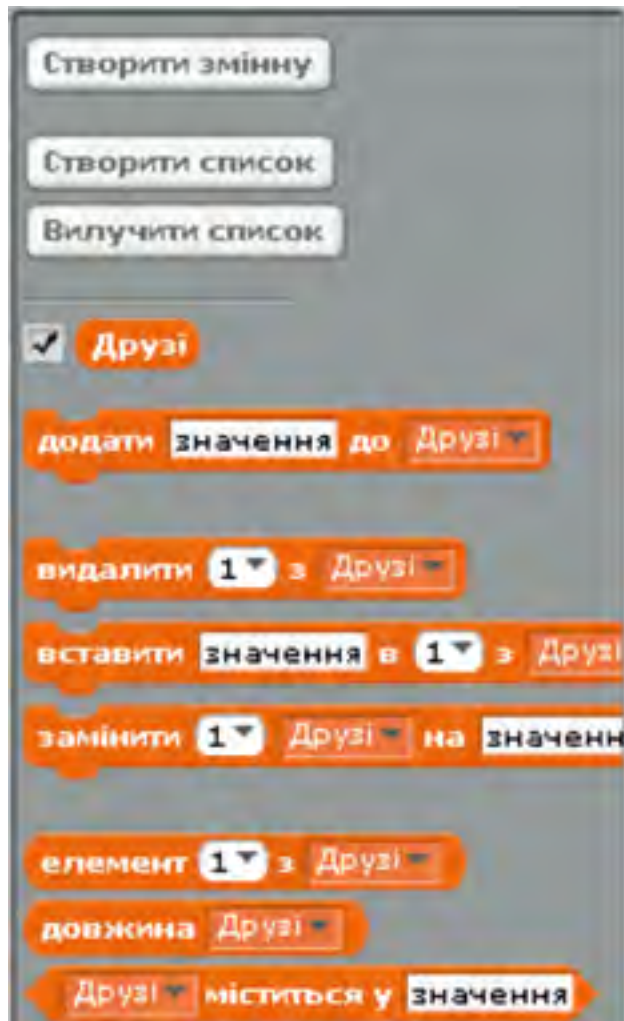


Рис. 3.52. Команды для работы со списками

Если включить флажок, расположенный рядом с названием **Друзья**, пустой список отобразится на сцене проекта (рис. 3.53).

Для ввода элементов списка нажимают кнопку **добавить (+)** и в поля окошка вводят элементы списка.

После ввода каждого элемента нажимают клавишу **Enter**.

Введем, например, в список **Друзья** следующие имена: Ваня, Коля, Вова. На рис. 3.54 представлен список **Друзья** с введенными именами. Если элементы списка не помещаются в окошке, его можно растянуть. Для этого указатель мыши устанавливают в его правый нижний угол, нажимают кнопку мыши и, не отпуская ее, изменяют размер до нужного.

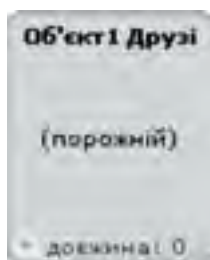


Рис. 3.53. Окошко для ввода значений элементов списка

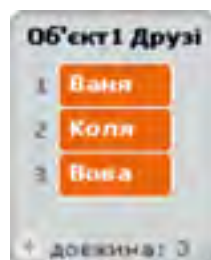


Рис. 3.54. Значение элементов списка **Друзья**

В созданный список можно добавить новый элемент в конец списка с помощью, например, команды **дати Толя до Друзі**. Из списка можно удалить любой элемент по его номеру. Например, с помощью команды **видалити 2 з Друзі** удаляется второй элемент из списка **Друзья**. С помощью команды **вставити Сашко в 2 з Друзі** в список **Друзья** будет вставлено во вторую строку имя Саша, а с помощью команды **замінити 2 Друзі на Петрик** имя Коля будет заменено на имя Петя. Команда-репортер **Содержится в** выдает логическое значение «истинно», если указанный элемент содержится в списке, и значение «ложно» – в противном случае. Например, в результате выполнения команды **Друзі міститься у Вова** будет получено значение «истинно», потому что имя Вова есть в списке **Друзья**. Команда-репортер **Друзі** позволяет вывести на экран все элементы списка. Команда-репортер **дошкина Друзі** выдает количество элементов в списке, а команда **елемент 3 з Друзі** выбирает из списка третье имя.



**Пример.** Разработаем проект, который реализует такой сценарий. На фоне сцены driveway (папка Outdoors) встретились друзья Сергей (объект boy1-standing, папка People) и Андрей (объект boy4-langhing, папка People), которые изображены на рис. 3.55.



Рис. 3.55. Беседа друзей о футболе

Сергей спрашивает у Андрея: «Кто на втором месте?». Через несколько секунд Андрей отвечает: «Динамо». Сергей благодарит Андрея.

Для реализации проекта создадим список с именем **Призеры**, элементами которого являются названия команд *Шахтер*, *Динамо* и *Днепр*. Обратим внимание, что при создании списка необходимо включить переключатель **Для всех объектов**. Программа, реализующая слова Сергея, показана на рис. 3.56, а программа, реализующая слова Андрея, – на рис. 3.57.

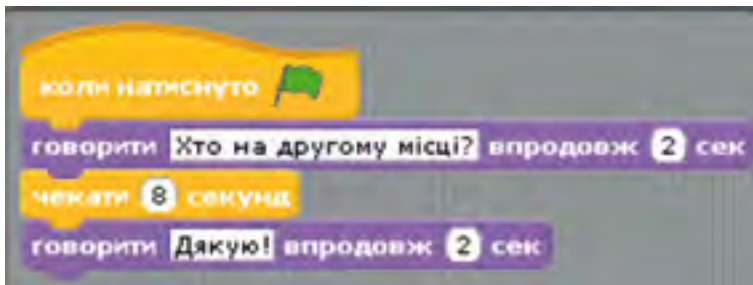


Рис. 3.56. Программа, реализующая слова Сергея

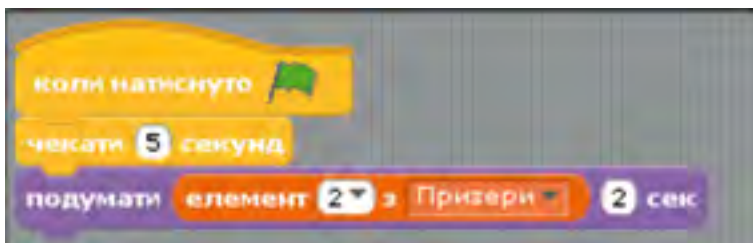


Рис. 3.57. Программа, реализующая слова Андрея

Для одновременного выполнения разработанных программ следует нажать зеленую кнопку в среде Скретч. Если нажать на зеленый флажок программы, будет выполняться только эта программа. Для имитации диалога произведена задержка выполнения соответствующих команд на определенное количество секунд.

### Выполняем

1. Создайте переменную с именем  $b_1$ . Включите и выключите флажок слева от имени переменной. Какие изменения происходят на сцене? Предоставьте переменной  $b_1$  случайное значение числа в диапазоне от 5 до 22. Удалите переменную. 🟢
2. Создайте список с именем **Компьютер**, содержащий названия следующих устройств: *память, монитор, принтер, мышь*. Добавьте в конец списка название *процессор*. Замените название *мышь* на название *винчестер*. Удалите из списка название *принтер*. 🟢
3. В программе, представленной на рис. 3.58,  $c_1$  – это переменная. Разработайте графическую схему алгоритма, который реализует эта программа. Какое значение приобретет переменная  $c_1$  после выполнения программы? Выполните программу и проверьте правильность алгоритма. ✚

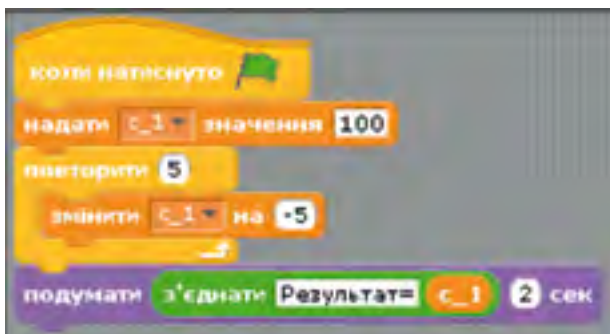


Рис. 3.58. Программа изменения значения переменной

4. На рис. 3.59 представлена программа анализа списка с именем **Город**, элементами которого являются: Житомир, Харьков, Сумы, Черкассы. Проанализируйте программу. Какие результаты она выдает? Запустите программу на выполнение и проверьте правильность своей гипотезы. ✚

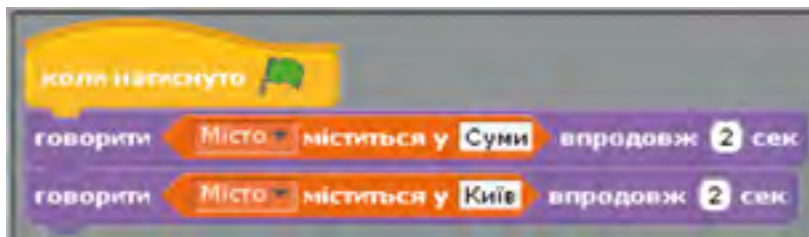


Рис. 3.59. Программа анализа содержания списка

5. На рис. 3.60 изображен аквариум (фон сцены underwater, папка Nature), в котором находится рыбка (объект fish2, папка Animals).



Рис. 3.60. Аквариум с рыбкой

На рис. 3.61 приведена программа, с помощью которой рыбка четыре раза перемещается в точки со случайными значениями координат, хранящимися в переменных  $a_1$  и  $b_1$ . Переменной  $a_1$  сообщаются случайные значения в диапазоне от 20 до 60, а переменной  $b_1$  – в диапазоне от 30 до 80. Проанализируйте программу и разработайте для нее графическую схему алгоритма. Выполните программу и убедитесь, что алгоритм разработан правильно. ★



Рис. 3.61. Программа перемещения объекта на случайные значения расстояний

6. На рис. 3.62 показана программа, в которой  $a$  и  $p$  – переменные. Выполните программу, проанализируйте результаты, которые она выдает, и сформулируйте условие задачи, которую реализует эта программа. ✦

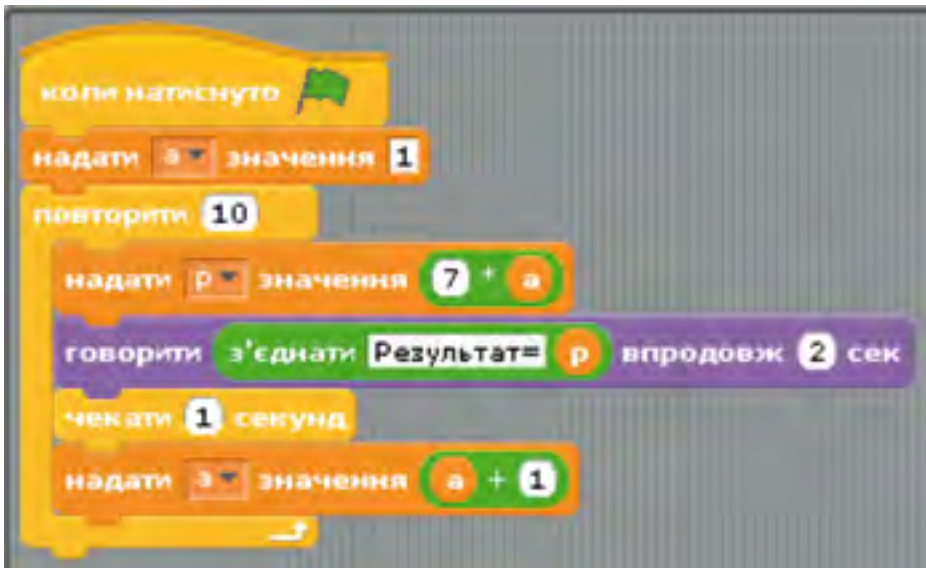


Рис. 3.62. Программа для самостоятельного анализа

7. 🏠 В программе, представленной на рис. 3.63,  $p$  – это список, в котором хранятся числа 32, 12, 44, 21;  $a$  и  $s$  – переменные. Выполните программу, проанализируйте полученные результаты и разработайте графическую схему алгоритма, реализуемого этой программой. ➕

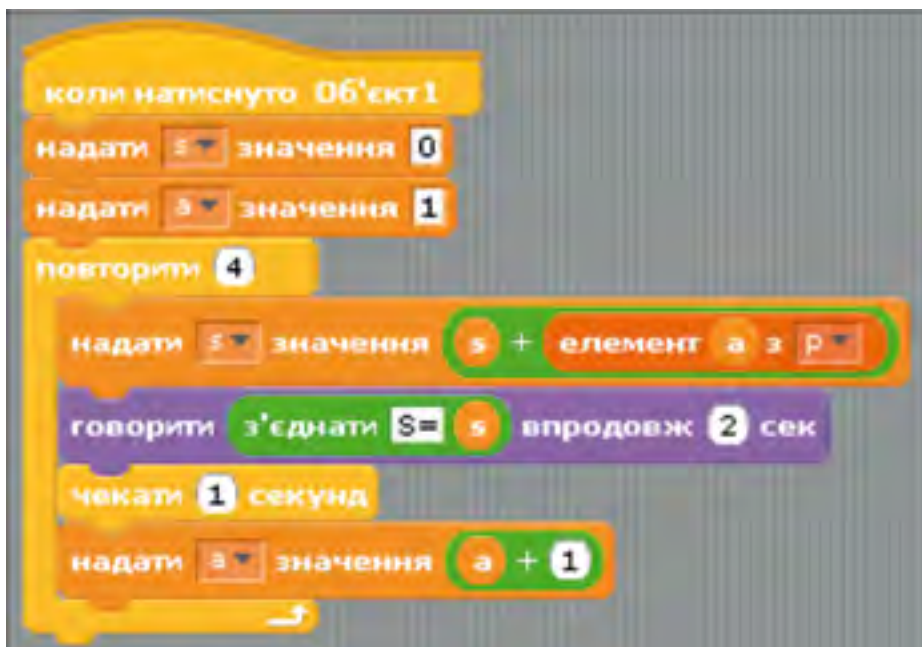


Рис. 3.63. Программа для самостоятельного анализа




8.  На рис. 3.64 изображены футбольное поле (фон сцены football-field, папка Sports), судья (объект referee1, папка People) и футболист (объект football-stand, папка People). Разработайте для каждого объекта программу, моделирующую возможную предварительную беседу между ними. ★



Рис. 3.64. Сценарий для разработки проекта


9.  Создайте проект, с помощью которого вычисляется значение выражения  $(a \cdot c - b) / b$ . Значения переменных вводятся с клавиатуры. ✚
10.  Разработайте программу, которая формирует список из 10 элементов. Элементами списка являются числа натурального ряда. После формирования списка его элементы выводятся на сцену. ★

### Проверяем себя

1. Какие данные называют константами? 🟢
2. Какие данные называют переменными? 🟢
3. Приведите примеры зарезервированных констант в среде Скретч. 🟢
4. Из каких символов может состоять имя переменной? 🟢
5. Объясните порядок создания переменных в среде Скретч. ✚
6. С помощью каких команд можно придать значения переменным? ✚
7. Назовите команды среды Скретч для работы с переменными. ✚
8. Для чего используются мониторы переменных? ✚
9. Что называется списком в среде Скретч? ★
10. Назовите команды для работы со списками. ★
11. Объясните порядок создания списка. ★
12. Каким образом наполнить список его элементами? ★



### 3.5. Алгоритмы с разветвлением

 **Алгоритмы с разветвлением** – это алгоритмы, в которых, в зависимости от результата проверки условия, выполняется одна группа команд или другая группа команд.

#### Примеры.

**Если** футбольный матч закончился ничейным результатом, **то** обоим командам начисляется по одному очку, **иначе** победитель получит три очка, а проигравшая команда – ноль очков.

**Если** стороны треугольника имеют значение 3, 4, 5, **то** делается вывод, что треугольник прямоугольный, **иначе** – треугольник не прямоугольный.

**Если** на перекрестке в светофоре включен зеленый свет, **то** автомобиль может двигаться, **иначе** он останавливается.

Для описания алгоритмов с разветвлением используется оператор условного перехода, который проверяет условие и выдает значение «истинно» (**да**), если условие выполняется, и значение «ложно» (**нет**), если условие не выполняется. Например, в результате проверки условия  $13 = 21$  будет выдано значение «ложно», а результатом проверки условия  $37 > 24$  является значение «истинно».

Для проверки условия используются два типа операторов условного перехода.


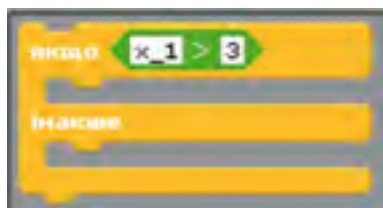
 **Оператор условного перехода первого типа изображен на рис. 3.65.**



Рис. 3.65. Условный оператор первого типа

Если условие подтверждается, то есть выражение приобретает значение «истинно» (**да**), то выполняется одна группа инструкций, если приобретает значение «ложно», (**нет**) – другая группа инструкций.





После выполнения первой или второй группы инструкций продолжается выполнение общих инструкций. Этот тип оператора условного перехода в среде Скретч реализуется таким блоком команд.

После слова **если** устанавливается одна из команд-репортеров проверки условия из категории **Операторы**. В данном случае здесь проверяется условие  $x_1 > 3$ . Если условие выполняется, то есть если значение переменной  $x_1$  больше 3, то выполняются все команды, вставленные в строку после слова **если**. В случае, когда  $x_1$  не больше 3, команды в строке после слова **если** не выполняются, а выполняются все команды, вставленные в строку после слова **иначе**. В любом случае следующей выполняется команда, расположенная непосредственно после этой команды.

**Пример.** С помощью клавиатуры вводится длина стороны квадрата ( $a$  должно быть четным числом). Необходимо разработать программу определения, можно ли в этот квадрат вписать окружность радиусом  $r$ . Ниже приведено описание алгоритма решения этой задачи в словесной форме.

1. Ввести длину стороны квадрата  $a$ .
2. Если  $a / 2 = r$ , выполнить инструкцию 3, иначе – инструкцию 5.
3. Можно вписать.
4. Выполнить инструкцию 6.
5. Вписать невозможно.
6. Конец.

Программа реализации этого алгоритма представлена на рис. 3.66.

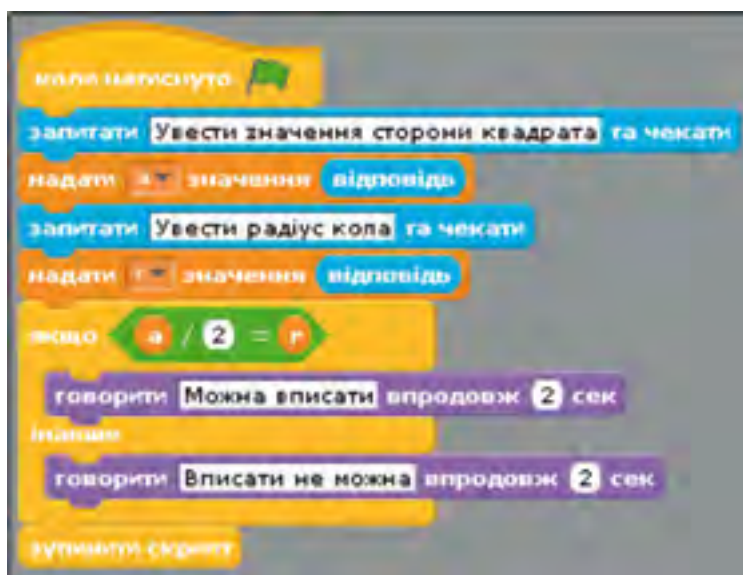


Рис. 3.66. Программа анализа возможности вписать окружность в квадрат

В процессе выполнения программы после высвечивания сообщения «Ввести значение стороны квадрата», а также сообщения «Ввести радиус окружности» программа приостанавливает свое выполнение и ожидает ввода ответов. Ответ вводится в поле, расположенное в нижней части сцены. Для продолжения выполнения программы после введения соответствующих значений следует нажать клавишу Enter.

**Пример.** Провизор спрашивает у покупателя: «Какие Вам нужны лекарства?». Покупатель называет нужный перечень. Провизор просматривает базу данных аптеки. Если он находит необходимые, называет их стоимость, иначе говорит: «Отсутствуют».

Пусть база данных аптеки называется *лекарства* и содержит: *пиносол*, *анальгин*, *панкреатин*, *нитроглицерин*. В переменной *a* будем хранить ответ покупателя. Программа анализа содержания базы данных аптеки и формирования ответов провизора показана на рис. 3.67.

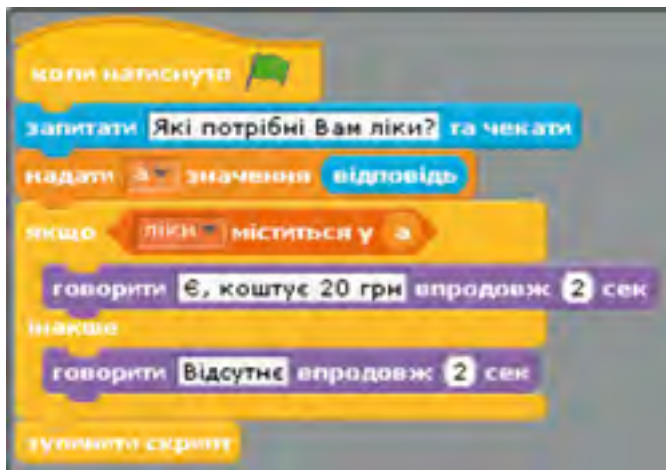
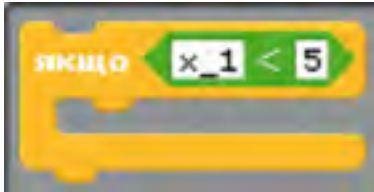


Рис. 3.67. Программа анализа базы аптеки и формирования ответов провизора

✓ Графическая схема условного оператора второго типа представлена на рис. 3.68.



Рис. 3.68. Условный оператор второго типа



Как видим из рис. 3.68, группа инструкций выполняется, если условие истинно (да), и не выполняется, если оно ложно (нет).

В среде Скретч условный оператор этого типа реализуется командой, которая выполняется таким образом: если условие после слова **если** истинно (в данном случае проверяется условие: значение переменной  $x_1$  меньше 5), то выполняются все команды, вставленные в строку после слова **если**. Иначе эти команды не выполняются. В любом случае следующей выполняется команда, расположенная непосредственно после этой команды.

**Пример.** Клиент положил в банк на месяц некоторую сумму денег под 3 % в месяц. Через некоторое время он решил снять деньги со счета. Разработать алгоритм и программу определения полученной суммы денег.

Обозначим сумму взноса переменной  $s$ , а количество дней, прошедших до дня, когда снимается сумма, – переменной  $a$ . Если деньги снимаются ранее чем через 30 дней после внесения, то сумма взноса не меняется, иначе она вычисляется по формуле:  $s = s + s * 0,03$ . Алгоритм определения полученной суммы денег в банке приведен ниже.

1. Ввести значения переменных  $s$  и  $a$ .
2. Если  $a > 30$  – пункт 3, иначе – пункт 4.
3.  $s = s + s * 0,03$ .
4. Вывести значение  $s$ .

Программа, реализующая фрагмент этого алгоритма, представлена на рис. 3.69.

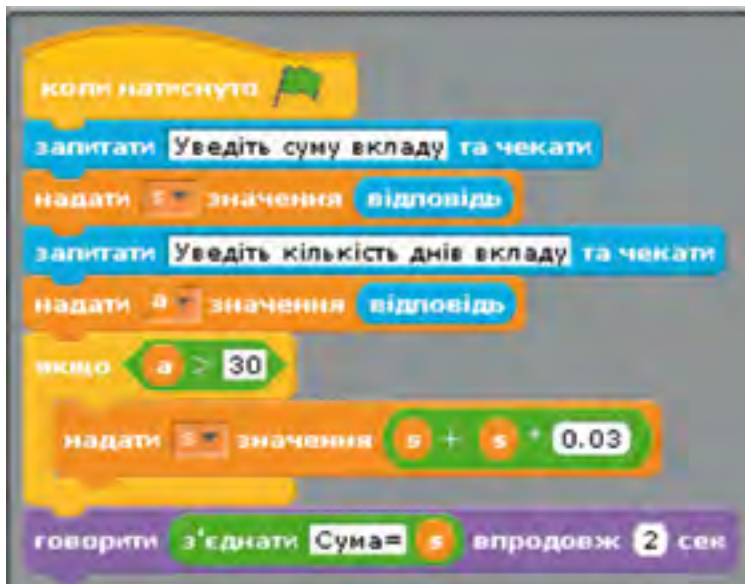


Рис. 3.69. Программа определения суммы взноса

**Пример.** Водитель автобуса Прага–Київ перед каждой автозаправочной станцией спрашивает своего партнера: «Сколько осталось топлива?» Если топлива менее 40 литров, он заезжает на заправку, потом продолжает рейс. Иначе движение продолжается без заправки. Программа моделирования этого процесса представлена на рис. 3.70.

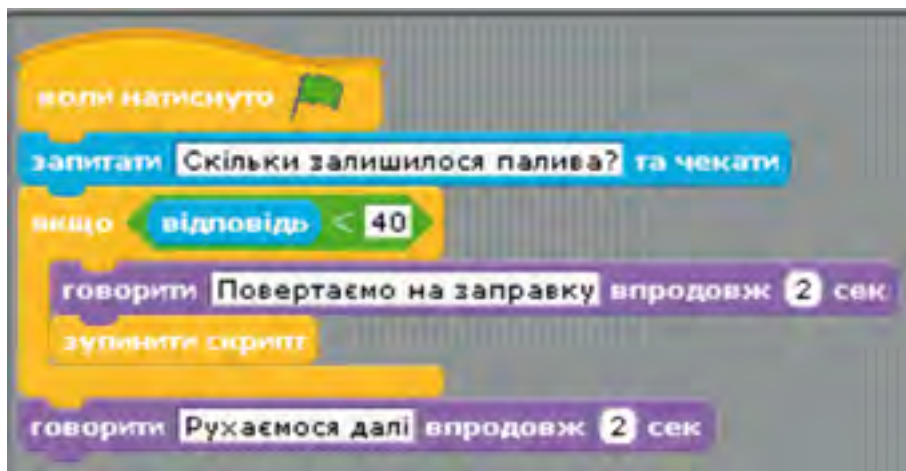


Рис. 3.70. Программа моделирования действий водителя автобуса

В операторах условного перехода часто применяются **логические данные и логические выражения**, что обеспечивает разработку более компактных программ.

**Пример.** Три стороны треугольника вводятся с клавиатуры. Определить, является ли этот треугольник равнобедренным. Обозначим эти стороны переменными  $a$ ,  $b$ ,  $c$ . Ниже приведено описание фрагмента алгоритма решения этой задачи в словесной форме.

1. Ввести значения переменных  $a$ ,  $b$ ,  $c$ .
2. Если  $a = b$ , или  $a = c$ , или  $b = c$ , выполнить инструкцию 3, иначе – инструкцию 4.
3. Вывод – треугольник равнобедренный. Перейти к инструкции 5.
4. Вывод – треугольник не равнобедренный.
5. Конец.

Программа реализации алгоритма представлена на рис. 3.71.

**Пример.** Числа  $a$  и  $c$  вводятся с клавиатуры. Если  $a > 0$  и  $c > 4$ , но  $c < 10$ , то вычисляется значение выражения  $4 * a + 3 * c$ , иначе вычисляется значение выражения  $5 + a * c$ . Программа вычисления значений этих выражений представлена на рис. 3.72.



Рис. 3.71. Програма определения, является ли треугольник равнобедренным



Рис. 3.72. Програма вычисления значения математического выражения



## Выполняем

1. Саша спрашивает у сестры Кати: «Какой сегодня день?». Если она отвечает «Воскресенье», Саша говорит: «Иду на речку», иначе – «Делаю уроки». На рис. 3.73 показана программа моделирования их разговора. Выполните программу и убедитесь, что она функционирует правильно. Разработайте графическую схему алгоритма, который реализует эта программа. 🚩



Рис. 3.73. Программа моделирования разговора Саши и Кати

2. Проанализируйте и выполните программу, представленную на рис. 3.74. Сформулируйте возможное условие задачи для этой программы. ➕

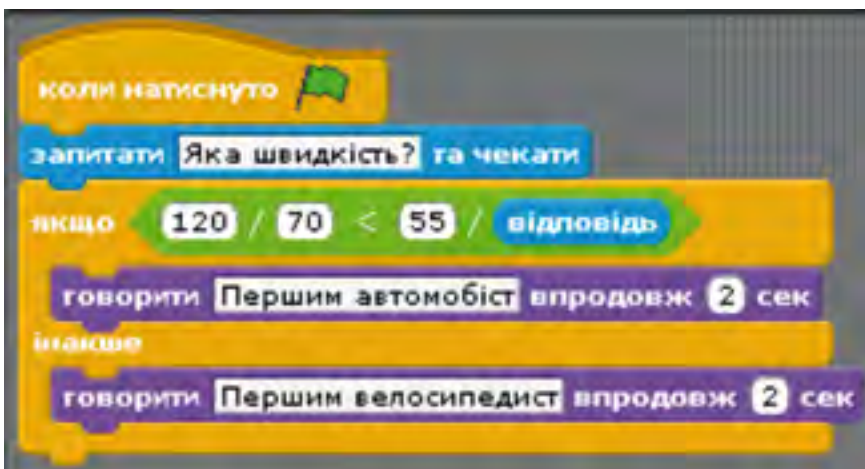


Рис. 3.74. Программа для формулировки условия задачи

3. Нужно назвать пропущенные буквы в слове ПРО\_Е\_ОР, означающем устройство системного блока компьютера. Программа моделирования



отгадывания слова представлена на рис. 3.75. Докажите, что программа работает правильно. ★

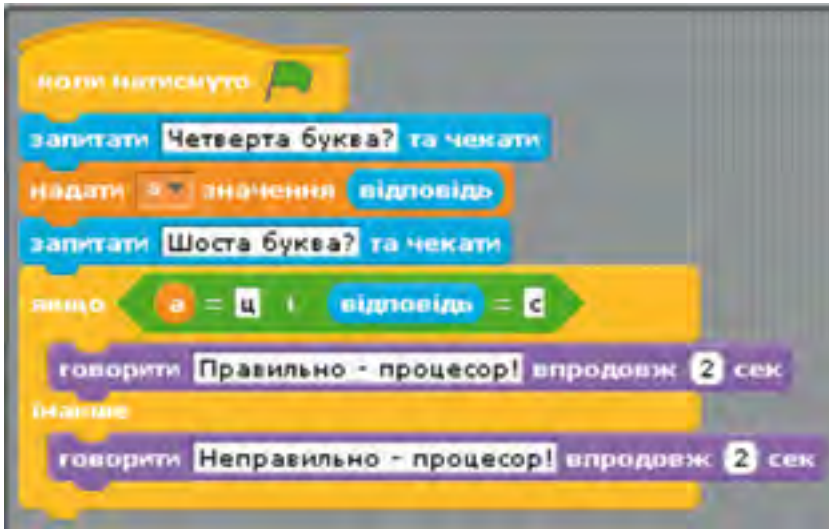


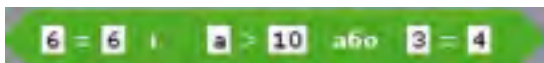
Рис. 3.75. Программа отгадывания слова

4. 🧑🧑 На столе лежат три яблока. Разработайте алгоритм и программу нахождения самого большого среди них путем сравнения их попарно. ✚
5. 🏠 Дано число. Разработайте программу, которая сообщает: «Число положительное», если оно больше нуля, и «Число отрицательное», если оно меньше нуля. ✚
6. 🧑🧑 Петя купил лотерейный билет с шестизначным номером. Он решил так: если в билете одинаковы первая и третья цифры или одинаковы четвертая и шестая цифры, то ему повезет. Разработайте программу, определяющую, повезет ли Пете. Номер лотерейного билета генерируется в программе как случайное число. ★
7. 🏠 С клавиатуры вводятся три числа. Разработайте программу, которая определяет два из них, сумма которых наименьшая. ★

### Проверяем себя

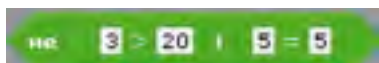
1. Объясните, как выполняется алгоритм с разветвлением. 🌱
2. Начертите графическую схему оператора условного перехода первого типа. 🌱
3. Какой командой Скретч реализуется оператор условного перехода первого типа? 🌱
4. Начертите графическую схему оператора условного перехода второго типа. 🌱

5. Объясните, как выполняется команда среды Скретч если <условие> иначе. ✦
6. Объясните, как выполняется команда среды Скретч если <условие>. ✦
7. Приведите пример алгоритма с разветвлением. ✦
8. Какой командой среды Скретч реализуется оператор условного перехода второго типа? ✦
9. Приведите пример логического выражения в командах среды Скретч. ★
10. При каких значениях переменной  $a$  логическое выражение



примет значение «истинно»? ★

11. Какое значение имеет логическое выражение



? ★

12. Какое значение имеет логическое выражение



? ★

### 3.6. Алгоритмы повышенной сложности

Выше рассматривались простейшие структуры алгоритмов с повторением и с разветвлением. На практике применяются более сложные структуры этих типов алгоритмов. Ниже описаны циклические алгоритмы с разветвлением и алгоритмы с вложенными циклами.

#### 3.6.1. Циклические алгоритмы с разветвлением

В предыдущих разделах рассматривались алгоритмы с фиксированным, наперед известным количеством циклов. Например, алгоритм получения таблицы умножения на 3 содержит 10 циклов. На практике часто используются алгоритмы, в которых количество циклов заранее неизвестно. Оно зависит от текущей ситуации в процессе выполнения алгоритма. Например, неизвестно, сколько раз нужно бросить кубик с цифрами от 1 до 6, чтобы получить сумму чисел больше 45. Неизвестно также, сколько раз необходимо выполнить вычитание из большего числа меньшего, чтобы разница чисел стала меньше меньшего числа. Например, если заданы числа 34 и 9, процесс уменьшения будет происходить таким образом:


$$1\text{-й цикл: } 34 - 9 = 25;$$

$$2\text{-й цикл: } 25 - 9 = 16;$$

$$3\text{-й цикл: } 16 - 9 = 7, \text{ конец.}$$

Циклические алгоритмы с заранее неизвестным количеством повторений делятся на два вида: с предусловием и с постусловием.

## Циклические алгоритмы с предусловием

 В алгоритмах с предусловием сначала проверяется определенное условие. До тех пор, пока результат его проверки имеет значение «истинно», цикл выполняется. Как только в результате проверки условия получается значение «ложно», выполнение цикла прекращается.

На рис. 3.76 представлена схема, объясняющая сущность циклического алгоритма с предусловием. Как видим из рис. 3.76, операции тела цикла могут вообще быть не выполнены ни разу, если результат проверки условия имеет значение «ложно».

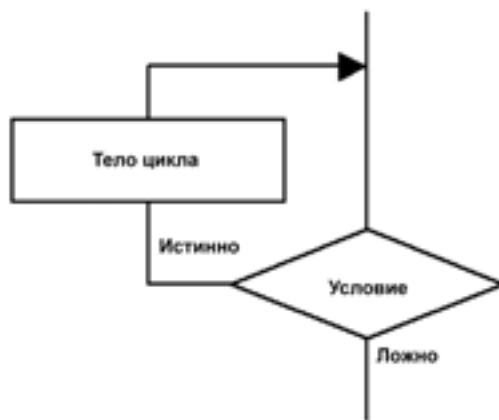
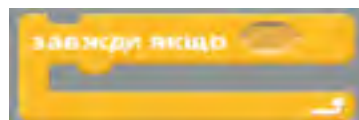


Рис. 3.76. Графическая схема циклического алгоритма с предусловием

Для реализации циклических алгоритмов с предусловием в среде Скретч используется команда:



Инструкции, вставленные в эту команду, будут выполняться в случае, если условие, записанное в команде, имеет значение «истинно».

**Пример.** На рис. 3.77 изображены фон сцены scholl2 (папка Outdoors) и объект boy4-walking-a (папка People).



Рис. 3.77. Движение объекта boy4-walking-a

Для создания эффекта анимации объекта boy4-walking-b импортируем его образы  $b$  и  $c$ . Программа, моделирующая его перемещения, представлена на рис. 3.78. После запуска программы объект начинает двигаться, как только будет нажата кнопка мыши. Движение объекта прекращается, если кнопку мыши отпустить. Итак, команды тела цикла выполняются только при нажатии кнопки мыши.

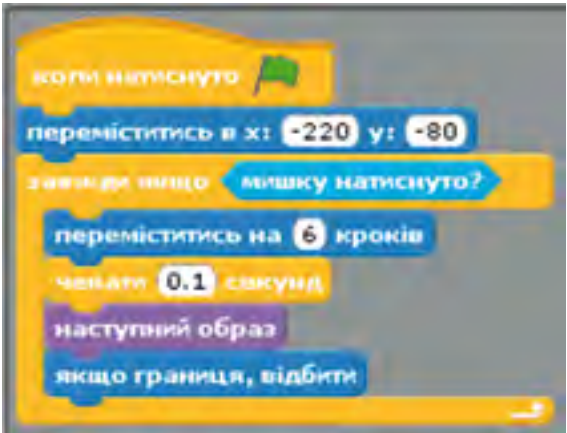


Рис. 3.78. Программа моделирования движения объекта boy4-walking

**Циклические алгоритмы с постусловием**


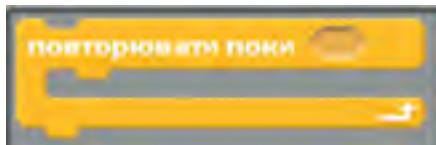
 В циклических алгоритмах с постусловием сначала выполняются операторы тела цикла, а затем проверяется условие. Если условие имеет значение «ложно», операторы тела цикла выполняются еще раз, иначе их выполнение прекращается. В таких алгоритмах тело цикла выполняется по крайней мере один раз.



Рис. 3.79. Графическая схема циклического алгоритма с постусловием

Сущность циклических алгоритмов с постусловием объясняется графической схемой, представленной на рис. 3.79.

В среде Скретч для реализации циклических алгоритмов с постусловием используется такая команда:



С помощью этой команды вставленные в нее инструкции выполняются до тех пор, пока ее логическое выражение имеет значение «ложно».

**Пример.** Перед экзаменом по математике Вася взял кубик с цифрами от 1 до 6 и решил так: если выпадет цифра 5 не позже чем после третьего бросания, то ему повезет, иначе экзамен будет тяжелым. Программа моделирования этого процесса представлена на рис. 3.80. В переменной  $k$  сохраняется количество бросаний кубика.



Рис. 3.80. Программа моделирования игры в кубик

**Пример.** В расписании поездов по станции Киев подсчитать количество поездов, следующих в г. Львов. Обозначим: *Расписание* – общий список поездов по станции Киев (содержащий станции назначения Ровно, Харьков, Львов, Николаев, Львов, Херсон, Львов);  $a$  – текущий номер строки в расписании поездов;  $p$  – содержание текущей строки расписания;  $c$  – количество поездов во Львов. Программа представлена на рис. 3.81.

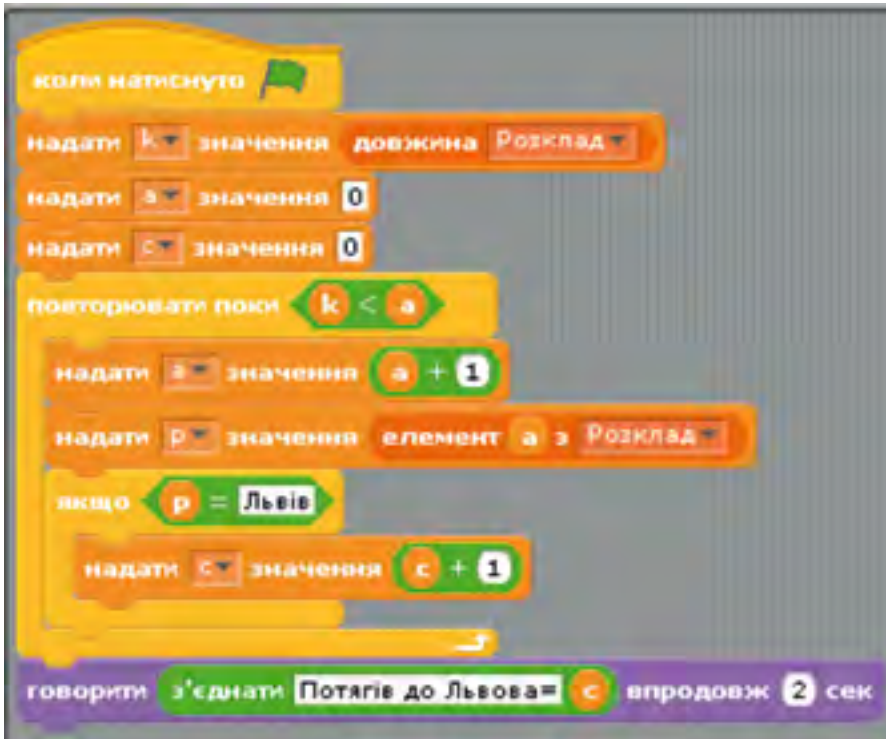


Рис. 3.81. Программа определения количества поездов в списке

### 3.6.2. Алгоритмы с вложенными циклами

**!** *Алгоритмы с вложенными циклами – это такие алгоритмы, в которых инструкции одного цикла содержатся в инструкциях другого цикла.*

Рассмотрим сущность алгоритмов такого типа на примере. Записаны три строки чисел, в каждой из которых находится по пять чисел:

1	3	5	7	9
4	6	8	10	12
7	9	11	13	15

Необходимо найти их общую сумму. Алгоритмы нахождения суммы чисел могут быть разными. Чаще всего применяется метод «последовательного накопления» суммы. Его сущность заключается в том, что берется число первой строки первого столбца, к нему прибавляется число второго столбца этой же строки. К полученной сумме прибавляется число третьего столбца, затем четвертого и пятого. Далее в такой же последовательности к полученной сумме добавляются числа второй, потом третьей строк. Для поданного примера этот процесс



можно записать следующим образом:  $1 + 3 = 4$ ;  $4 + 5 = 9$ ;  $9 + 7 = 16$ ;  $16 + 9 = 25$ ;  $25 + 4 = 29$ ;  $29 + 6 = 35$  и т. д.

Обозначим текущее значение суммы переменной  $s$ ; текущее значение числа, которое прибавляется – переменной  $a$ ;  $b$  – значение числа первого столбца текущей строки. С учетом описанного алгоритм нахождения суммы чисел можно записать таким образом:

1. Начало.
2. Текущее значение суммы равно 0 ( $s = 0$ ).
3. Сделать текущей первую строку.
4. Сделать текущим первый столбец.
5. Выбрать число ( $a$ ) из текущей строки текущего столбца.
6. Добавить к текущей сумме текущее число ( $s = s + a$ ).
7. Увеличить на единицу номер текущего столбца.
8. Повторить пункты 5–7 пять раз.
9. Увеличить на единицу номер текущей строки.
10. Повторить пункты 4–9 три раза.
11. Конец.

Программа реализации этого алгоритма представлена на рис. 3.82. В программе учтено, что каждое последующее число строки больше предыдущего на 2, а первое число каждой последующей строки больше первого числа предыдущей строки – на 3.

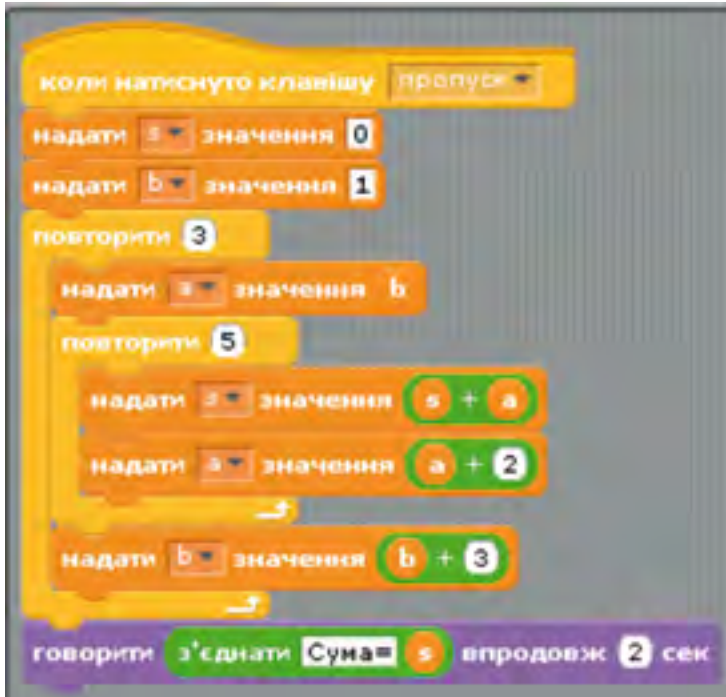


Рис. 3.82. Программа вычисления суммы чисел

**Пример.** На рис. 3.83 представлен орнамент. Его анализ показывает, что изображено 6 равносторонних треугольников различного цвета. Каждый из них сдвинут относительно предыдущего на половину его стороны. Программа рисования орнамента представлена на рис. 3.84.



Рис. 3.83. Орнамент для разработки программы

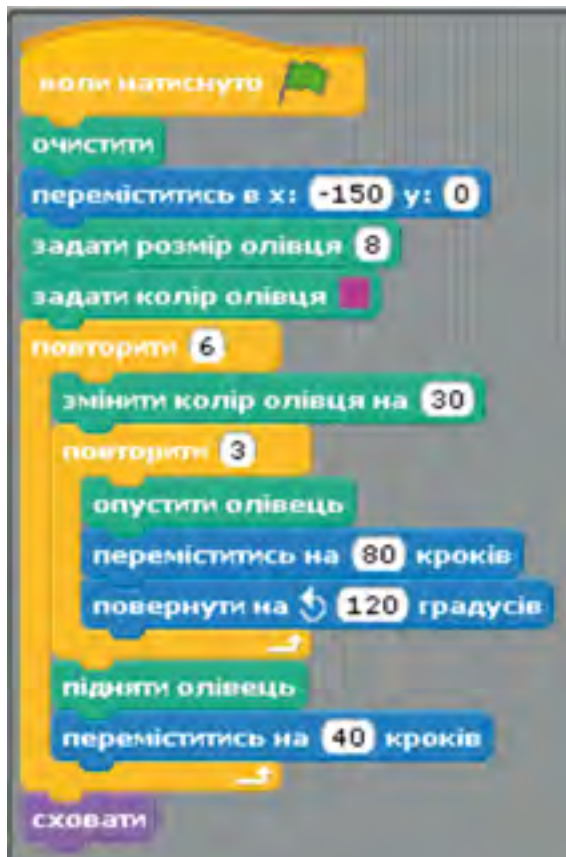


Рис. 3.84. Программа моделирования орнамента

### Выполняем

1. Два целых числа вводятся с клавиатуры. Разработайте программу для создания таблицы умножения для большего числа. 🧑🏫

2. Случайные числа генерируются в диапазоне от 2 до 7 и добавляются одно к другому. Разработайте программу определения количества случайных чисел, сумма которых будет больше 49. +

3. В памяти банкомата сохраняется список купюр 20 и 50 грн. На рис. 3.85 приведена программа вычисления общей суммы денег и необходимого количества купюр каждого номинала.



Рис. 3.85. Программа анализа списка купюр

В программе использованы такие переменные:  $k$  – общее количество купюр в банкомате (длина списка);  $a$  – текущий номер купюры в списке;  $x$  – количество купюр 20 грн;  $y$  – количество купюр 50 грн;  $s$  – общая сумма денег в банкомате, *Банкомат* – список купюр в банкомате. Проанализируйте и выполните программу. Докажите, что она функционирует правильно. Измените список *Банкомата* и проверьте правильность программы. ★

4. 👤 Перед последним этапом экспедиции, который должен длиться 10 дней, запас воды составлял 200 литров. С каждым последующим днем потребность в воде возрастала на 10 % относительно расхода воды за предыдущий день. В первый день расход воды составлял  $x$  литров (значение  $x$  вводится с клавиатуры). Экспедиция продолжается до тех пор, пока есть вода. При каких условиях экспедиция может завершиться без посторонней помощи? Если не хватит воды, то на сколько дней? ★

5. 🏠 На рис. 3.86 представлена программа, которая моделирует табло с демонстрацией шести крупнейших рек мира (Нил, Амазонка, Янцзы, Миссисипи, Енисей, Иртыш). На табло последовательно высвечиваются названия перечисленных рек. Выполните программу и убедитесь, что она функционирует правильно. Внесите дополнения в список рек и проверьте работу программы. ★

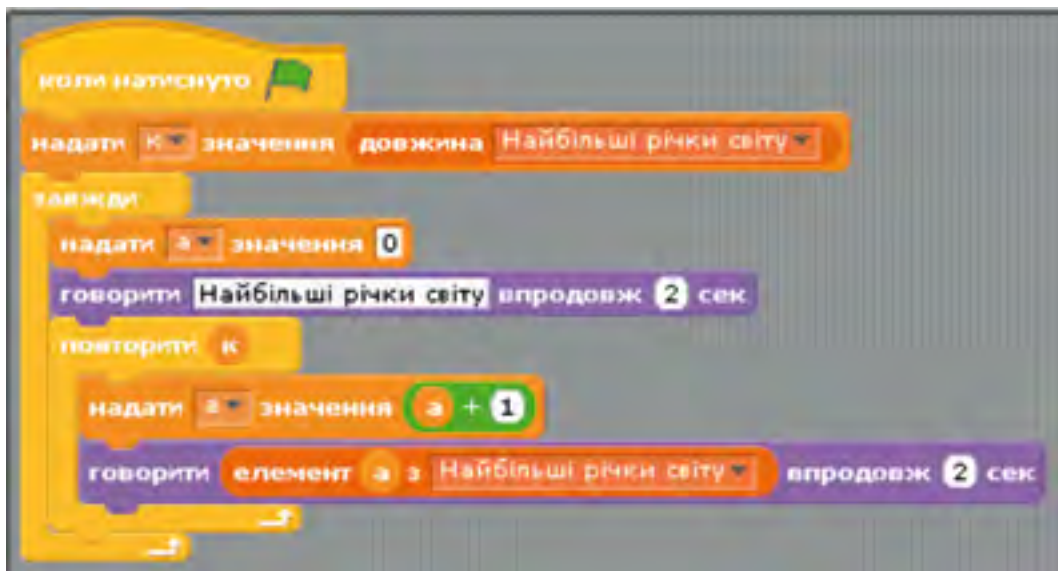


Рис. 3.86. Программа моделирования демонстрации табло

6. 👤 Выполните программу, представленную на рис. 3.87, и проанализируйте ее результаты. Сформулируйте условие задачи, которую может решить эта программа. +

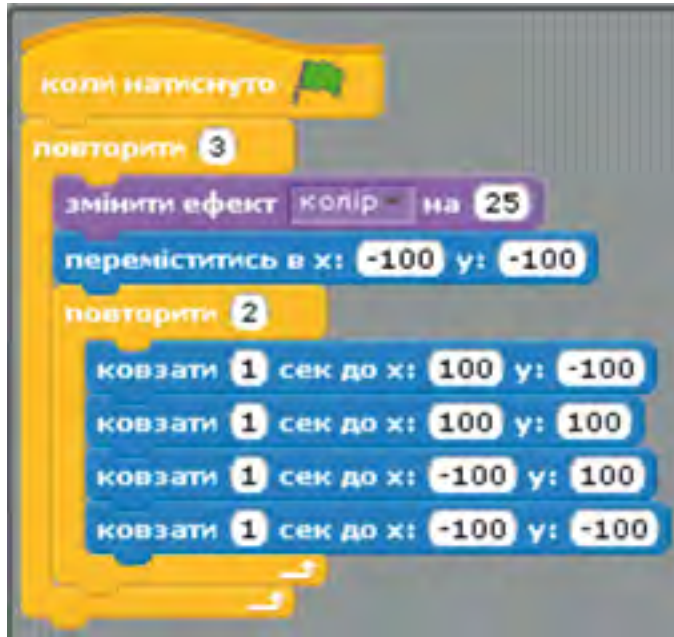


Рис. 3.87. Программа для самостоятельного анализа

7. 🏠 Разработайте программу, которая с помощью команд категории **Карандаш** создает пять квадратов, показанных на рис. 3.88. ✦



Рис. 3.88. Рисунок, создаваемый с помощью программы

### Проверяем себя

1. Какие алгоритмы называются циклическими с предусловием? 🌱
2. С помощью какой команды среды Скретч реализуется цикл с предусловием? 🌱
3. Какие алгоритмы называются циклическими с постусловием? 🌱
4. С помощью какой команды среды Скретч реализуется цикл с постусловием? 🌱
5. Какие алгоритмы называются алгоритмами с вложенными циклами? ✦
6. Начертите графическую схему циклического алгоритма с предусловием. ✦
7. Начертите графическую схему циклического алгоритма с постусловием. ✦



8. Приведите пример циклического алгоритма с предусловием. ★
9. Приведите пример циклического алгоритма с постусловием. ★
10. Приведите пример алгоритма с вложенными циклами. ★

<b>Практическая работа № 3</b>	<b>Тема:</b>	Составление и выполнение алгоритмов с повторением в учебной среде выполнения алгоритмов
	<b>Цель:</b>	Приобрести практические навыки разработки и реализации алгоритмов

**Задание.** С помощью команд группы **Карандаш, Управление и Движение** разработать алгоритм и программу создания двух одинаковых равносторонних треугольников; их стороны имеют разный цвет и толщину. Вариант размещения треугольников изображен на рис. 3.89.



*Рис. 3.89.* Изображения треугольников для их программного рисования

В процессе разработки программы можно придерживаться такой последовательности действий.

Переместить объект в точку сцены, например, с координатами  $x = -100, y = 0$ .

Выбрать размер (толщину линии рисования) и цвет карандаша.

Опустить карандаш.

Переместить карандаш на 100 шагов.

Повернуть объект на  $120^\circ$  налево.

Переместить объект на 100 шагов.

Повернуть объект на  $120^\circ$  налево.

Переместить карандаш на 100 шагов.

Повернуть объект на  $120^\circ$  налево.

Поднять карандаш.

Выбрать размер и цвет карандаша.

Переместить карандаш на 150 шагов.

Опустить карандаш.

Переместить объект на 100 шагов.

Повернуть объект на  $120^\circ$  направо.

Переместить карандаш на 100 шагов.

Повернуть объект на  $120^\circ$  направо.



Переместить объект на 100 шагов.

Повернуть объект на  $120^\circ$  направо.

Поднять карандаш.

Выполните программу и убедитесь, что она функционирует правильно. Измените размер и цвет карандаша. Выполните программу для новых данных.

<b>Практическая работа № 4</b>	<b>Тема:</b>	Составление и выполнение алгоритмов с разветвлением в учебной среде выполнения алгоритмов
	<b>Цель:</b>	Приобрести практические навыки разработки и реализации алгоритмов

**Задание.** В финальных соревнованиях по метанию диска принимали участие три спортсмена. Первым закончил выступать Василий, вторым – Петр и третьим – Николай. Все три спортсмена показали разные результаты. Разработать алгоритм и программу определения победителя соревнований.

Результат метания диска Василием обозначим переменной  $a$ , результат Петра – переменной  $b$  и Николая – переменной  $c$ . Будем считать, что значения этих переменных вводятся с помощью клавиатуры. Условие, согласно которому победителем становится Василий, можно записать таким образом: если выражение  $(a > b)$  and  $(a > c)$  имеет значение «истинно»; Петр становится победителем, если выражение  $(b > a)$  and  $(b > c)$  имеет значение «истинно», а Николай – если выражение  $(c > a)$  and  $(c > b)$  имеет значение «истинно».

Разработайте алгоритм и программу определения победителя. Проверьте программу для различных значений переменных. Докажите, что программа работает правильно.

<b>Практическая работа № 5</b>	<b>Тема:</b>	Создание и выполнение алгоритмов с повторением и разветвлением в учебной среде выполнения алгоритмов.
	<b>Цель:</b>	Приобрести практические навыки разработки алгоритмов с повторением и разветвлением, а также программ в среде Скретч.

**Задание.** В компьютерной базе библиотеки школы ежемесячно после поступления регистрируется 12 журналов: Математика, История, Начальная школа, Физика, География, Компьютер, Родная школа, Искусство, Родной язык, Химия, Иностранный язык, Безопасность. Чтобы узнать, поступил журнал или нет, ученику достаточно ввести его название. Необходимо разработать графическую схему алгоритма

и программу, моделирующую выполнение запроса ученика (название соответствующего журнала) и ответ о поступлении журнала, или о том, что такой журнал школа не выписывает.

Задания желательно выполнять в такой последовательности.

Создать список с названием **Журналы**. Элементами этого списка являются названия перечисленных журналов.

Разработать алгоритм в словесной форме. Пример такого алгоритма приведен ниже.

1. Начало.
2. Спросить: «Какой Вам нужен журнал?».
3. Ввести название нужного журнала.
4. Выбрать текущим первый элемент списка **Журналы**.
5. Если текущий элемент списка **Журналы** совпадает с названием введенного, выполнить п. 6, иначе – п. 8.
6. Говорить: «Да, журнал уже поступил».
7. Перейти к п. 11.
8. Увеличить на 1 номер текущего элемента списка **Журналы**.
9. Повторить п. 5 – п. 8 двенадцать раз.
10. Говорить «Такого журнала в школе нет».
11. Конец.

Создать необходимые переменные для реализации графической схемы алгоритма.

Разработать графическую схему алгоритма.

Разработать программу реализации алгоритма.

Отладить программу и доказать, что она функционирует правильно.

Для реализации циклов целесообразно применить блок команды.

Для проверки условия совпадения введенного названия журнала с названием текущего элемента списка **Журналы** желательно использовать блок команды.

Исследуйте все возможные варианты как запросов, так и ответов.

Сделайте выводы.



## СЛОВАРЬ

**Алгоритм** – описание последовательности определенных действий, выполнение которых за конечное количество шагов приводит к выполнению поставленной задачи.

**Алгоритм с повторением** – алгоритм, в котором одна и та же совокупность инструкций выполняется неоднократно.

**Алгоритм с разветвлением** – алгоритм, в котором в зависимости от результата проверки условия выполняется одна группа инструкций или другая.

**Алгоритмы с вложенными циклами** – алгоритмы, в которых инструкции одного цикла содержатся в инструкциях другого цикла.

**Анимация** – воспроизведение движущихся изображений.

**Интерпретатор языка программирования** – программа, которая по описанию алгоритма, выполненному на языке программирования, формирует последовательность машинных команд.

**Линейный алгоритм** – алгоритм, инструкции которого выполняются последовательно, в порядке их размещения.

**Переменная в программировании** – участок памяти с присвоенным ему именем, в который можно записывать различные значения данных.

**Программа** – алгоритм, представленный на языке программирования.

**Цикл (тело цикла)** – выполняемая многократно последовательность инструкций.

**Циклы с постусловием** – циклы, в которых сначала выполняются операторы тела цикла, а затем проверяется условие.

**Циклы с предусловием** – циклы, в которых сначала проверяется условие, а затем, в зависимости от результата, выполняются или не выполняются операторы тела цикла.



## РАЗДЕЛ 4. ОБРАБОТКА ДАННЫХ В ТАБЛИЦАХ



Зависимость между двумя и более величинами может быть представлена в форме таблицы, графика, математического выражения.

В результате измерений получают таблицу, в которой содержатся данные. Используя эти данные, можно построить график или диаграмму, анализируя которые, можно сделать выводы о ходе некоторого процесса. Почти все явления окружающего мира можно представить в форме информационной и математической моделей. Таблица с размещенными в ней данными, диаграмма или график, математическая формула являются моделями явлений, объектов.

### 4.1. Электронные таблицы, табличный процессор



Таблицы, электронные таблицы. Табличный процессор, его предназначение. Электронная таблица как объект. Объекты электронной таблицы, их свойства. Открывание, просмотр и сохранение электронной книги. Способы навигации на листе и в книге табличного процессора. Адресации в среде табличного процессора. Именованные ячейки и диапазоны.

Табличное представление данных является наиболее распространенным способом их хранения и передачи в виде сообщений. Особенность табличного представления состоит в том, что в таблице кратко можно отобразить достаточно подробные сведения об объекте. Например, табель с оценками ученика за год отображает (на одном небольшом листе бумаги) его учебную деятельность в течение года. Развернутые страницы дневника, на которых запечатлены расписание, домашние задания и отметки, являются моделью деятельности ученика в течение недели.

Использование электронных таблиц еще более информативно, поскольку можно создать электронные таблицы, в которых будет происходить автоматическое вычисление определенных значений величин, отображение их в виде рисунков (диаграмм, графиков), на основе которых человек будет принимать решения. Электронными таблицами можно пользоваться как источником данных с автоматизированным поиском.



*Электронными таблицами называют программные средства, данные в которых представляются в виде совокупности ячеек (клеток), каждая из которых имеет две координаты.*

Общим для всех видов электронных таблиц является способ обращения к данным как к содержимому ячейки (клетки) таблицы, то есть

с использованием номера строки и номера столбца, рассматриваемых как ссылки на ячейку, и, в результате, на ее содержимое.

✓ *Особо следует отметить, что для обработки порции данных, меньшей чем содержимое ячейки, необходимо использовать средства встроенного языка программирования.*

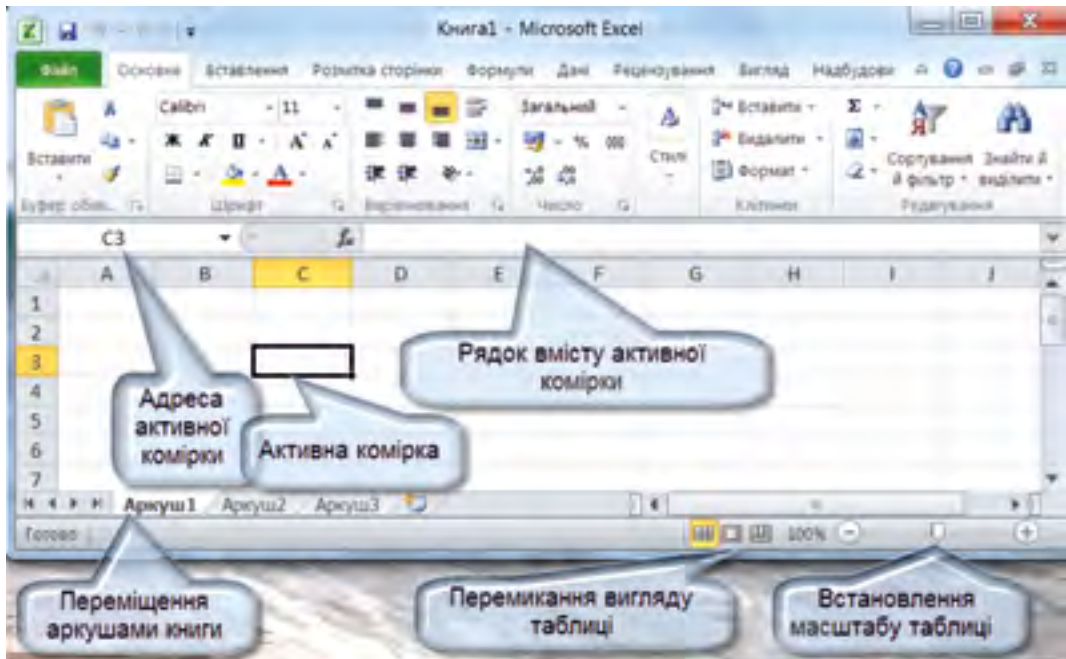


Рис. 4.1. Электронные таблицы Excel

✓ *Программные средства, обеспечивающие математическую обработку данных, поиск данных по определенным признакам в больших их наборах, создание сложных электронных документов и т.п. называют табличными процессорами.*

Основными признаками таких программных средств является представление данных в форме таблиц и создание документов в виде таблиц. Так же, как и текстовые редакторы, электронные таблицы (ЭТ) предоставляют возможность пользователям создавать текстовые сообщения, внедрять в них изображения. Однако, в отличие от текстовых редакторов, данные и изображения, представленные с использованием электронных таблиц, могут отображать достаточно сложные закономерности, присущие наборам данных. Рисунки (диаграммы) могут отображать только внесенные изменения данных.

✓ *Важным признаком ЭТ является возможность создания электронных документов, в которых, на основе определенной математической модели, выполняется обработка введенных данных.*

Наиболее распространенные ЭТ в настоящее время – MS Excel (рис. 4.1) и Libre Office Calc (рис. 4.2).

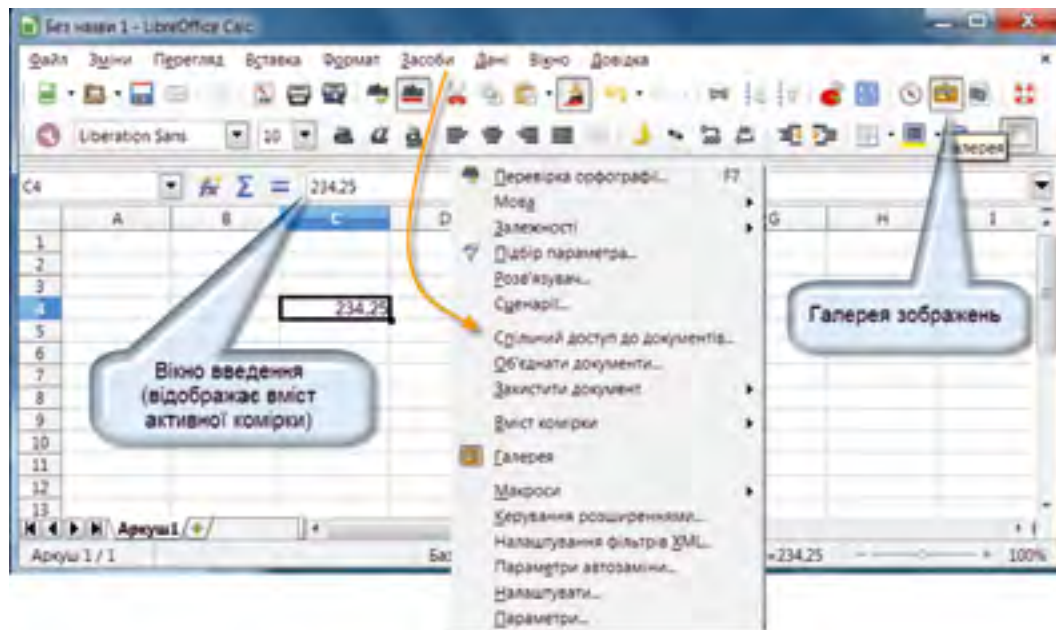


Рис. 4.2. Электронные таблицы Libre Office Calc

Электронные таблицы (или табличный процессор) MS Excel имеют интерфейс пользователя, подобный интерфейсу текстового процессора MS Word. Важным отличием ЭТ является наличие строки, в которой отображается содержимое ячейки (**Строка формул**), и наличие окна, в котором отображается ее адрес.

Интерфейс ЭТ Libre Office Calc построен без использования ленточного меню – основное меню не имеет выделенных блоков, а подменю вызываются «нажатием» на соответствующие пиктограммы. Интересная особенность этих таблиц – наличие большой галереи достаточно качественных изображений, которые можно использовать для оформления документов.

Электронные таблицы располагаются на **Листах**, образующих документ ЭТ, или **Книгу**. Перемещение по листам книги осуществляется специальным элементом интерфейса, предназначенным для перемещения по листам книги, добавления листов (см. рис. 4.1). Перемещение между ячейками (клетками) таблицы выполняется с помощью мыши, стрелок клавиатуры или тактильного экрана.

Наименьшей единицей данных в ЭТ является содержимое ячейки (клетки). Каждая ячейка (клетка) имеет свой адрес в виде латинской буквы (или нескольких букв), которыми обозначен столбец, и номера строки.



Группа (диапазон) ячеек обозначается как B6: E8, то есть используются обозначения первой и последней ячеек группы через двоеточие. В окне с адресом или именем ячейки в этом случае отображается адрес левой верхней ячейки диапазона.

Ячейка таблицы, с которой работает пользователь (вводит, изымает данные или формулы, форматирует символы и т.д.) называется **активной**. Ее содержимое отображается в **Строке формул** (рис. 4.3).

Ячейке или группе ячеек можно присвоить имя, которое будет использоваться в дальнейшем для обращения к их содержимому (рис. 4.4).

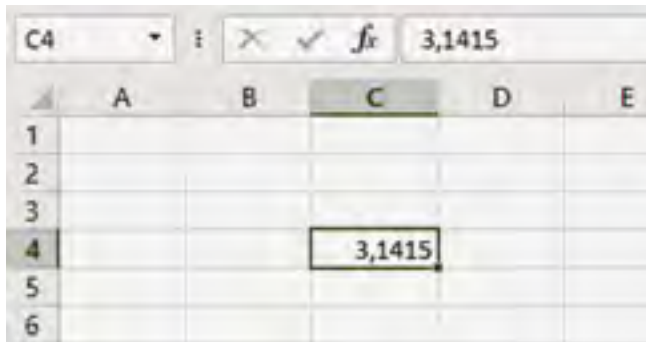


Рис. 4.3. Активная ячейка, ее адрес и содержимое

✓ *Имена ячеек могут содержать буквы и цифры, но не могут начинаться так, как адрес ячейки. Имена ячеек не должны повторяться даже на разных листах одной книги.*

Для того чтобы лучше ориентироваться в таблице, к некоторым ячейкам можно добавить примечания (рис. 4.4).



Рис. 4.4. Присвоение имени ячейке, добавление примечания (MS Excel 2010)

Подобным образом предоставляют имя группе ячеек в MS Excel 2007 (рис. 4.5).

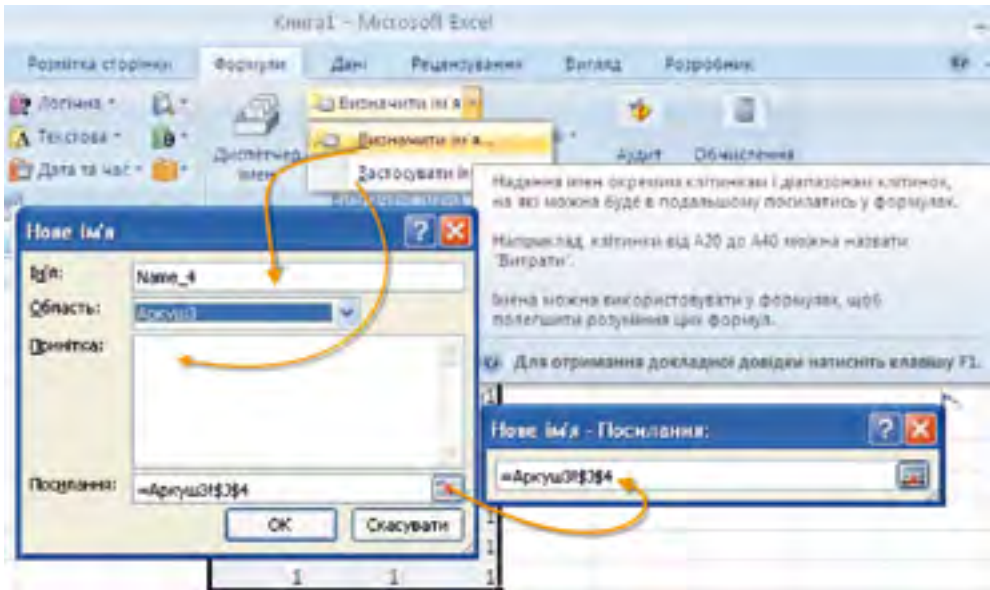


Рис. 4.5. Предоставление имени группе ячеек в MS Excel 2007

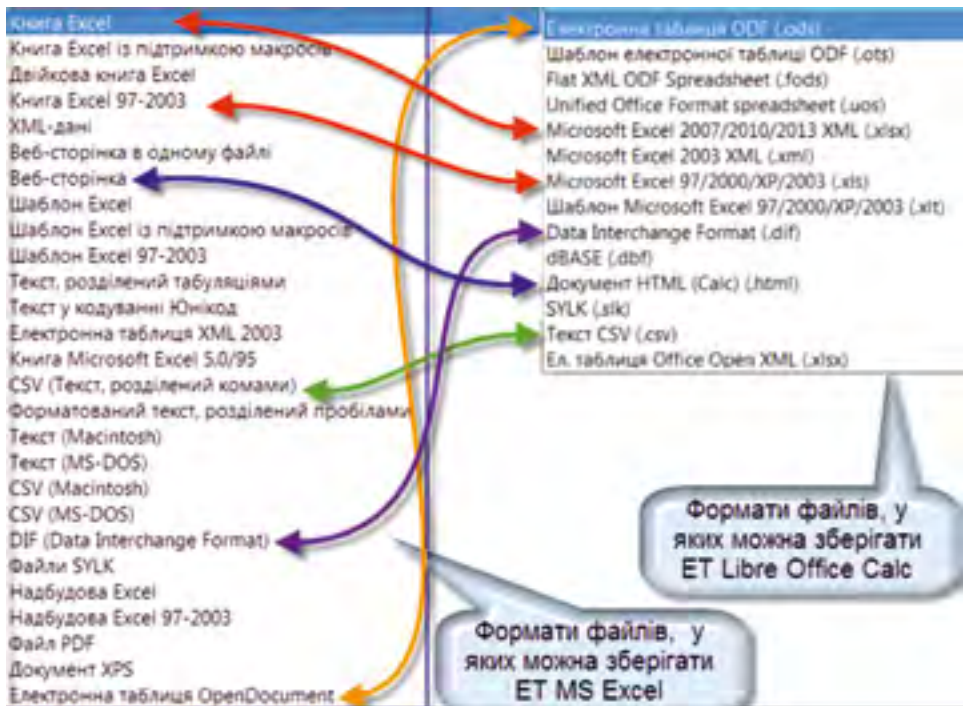


Рис. 4.6. Форматы, в которых могут быть сохранены документы, содержащие электронные таблицы (слева – MS Excel, справа – Libre Office Calc)



Важным форматом хранения данных в ЭТ является формат \* .csv (англ.: Comma Separated Values – разделенные запятыми значения). В файле этого формата хранятся только значения, без форматирования и формул для вычислений, но он используется для хранения данных очень часто. В указанном формате хранятся адресные книги практически всех мобильных телефонов.

Последовательность выполнения команд для сохранения файла с ЭТ в приложении Microsoft Excel 2007 показана на рис. 4.7.

Экспорт файлов предусмотрен также на компьютеры Macintosh и в операционную систему MS-DOS (текстовые форматы \* .csv, \* .txt).

Электронные таблицы можно воспроизвести на экране и на бумаге без линий, которыми лист расчерчен на клетки, но обычно используют режим с расчерчиванием (**Сетка**), как показано на рис. 4.8.

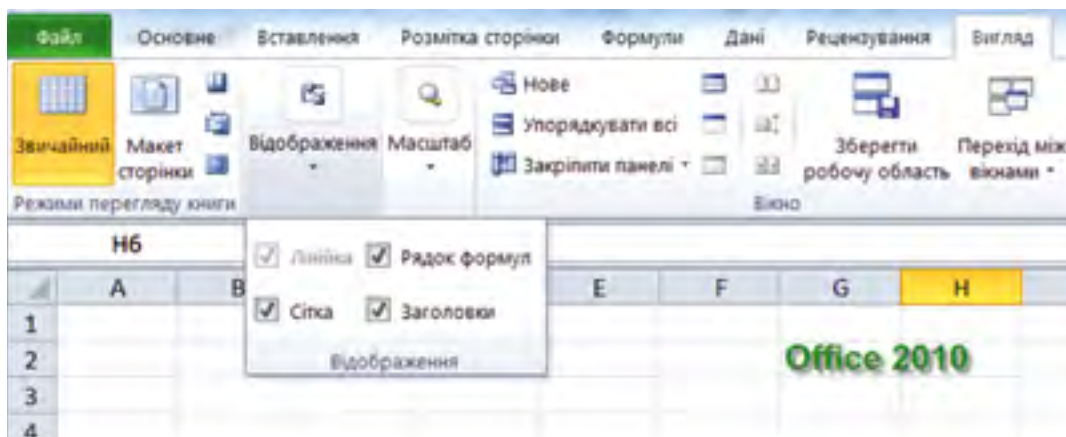


Рис. 4.8. Меню группы **Отображение** ленточного меню Вид табличного процессора MS Office 2010

Создать новую книгу электронных таблиц можно как средствами операционной системы (в контекстном меню выбрать команду **Создать**, а в подменю – **Лист Microsoft Excel**), так и с помощью самого табличного процессора.

Для создания нового документа (книги электронных таблиц) достаточно скачать программу Microsoft Excel, в результате откроется окно нового документа с именем Книга1 (Book1) в строке заголовка. По умолчанию Excel предлагает стандартное имя Книга [порядковый номер] (Book), которое пользователь при сохранении книги на внешнем запоминающем устройстве может заменить на любой другой из предоставленного при вызове команды **Сохранить как** списка.

Работая в Excel, можно создать новую книгу. Программа предлагает несколько вариантов создания книги: новой, из книги, которая уже существует, и с использованием шаблонов.

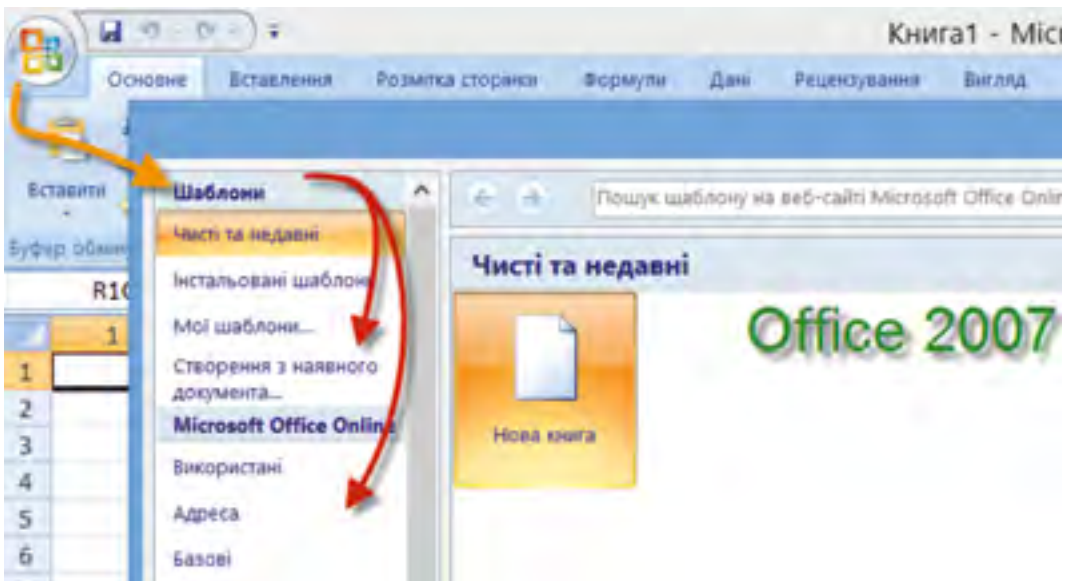


✓ *Каждый документ Microsoft Excel основывается на шаблоне.*

Шаблон определяет основную структуру документа. Общие шаблоны содержат настройки, используемые для всех документов. Шаблоны отдельных документов, например, шаблон личного бюджета на месяц, содержат специальные настройки, которые используются только для документов, основанных на этом шаблоне.

✓ *Для пользователя всегда доступны шаблоны, созданные им, и стандартные шаблоны, которые уже размещены на его компьютере.*

Если табличный процессор Excel 2007 уже загружен, для создания новой книги следует в меню кнопки Office вызвать подменю **Создать**, на экране появится меню со списком, разделенным на две группы: **Чистые и недавние** и **Microsoft Office online** (рис. 4.9).



*Рис. 4.9. Создание новой книги на основе шаблонов в приложении Office Excel 2007*

Для табличного процессора Excel 2010 возможные варианты создания новой книги представлены на рис. 4.10.

Для сохранения документа используют средства и правила, общие для приложений Microsoft Office и Windows.

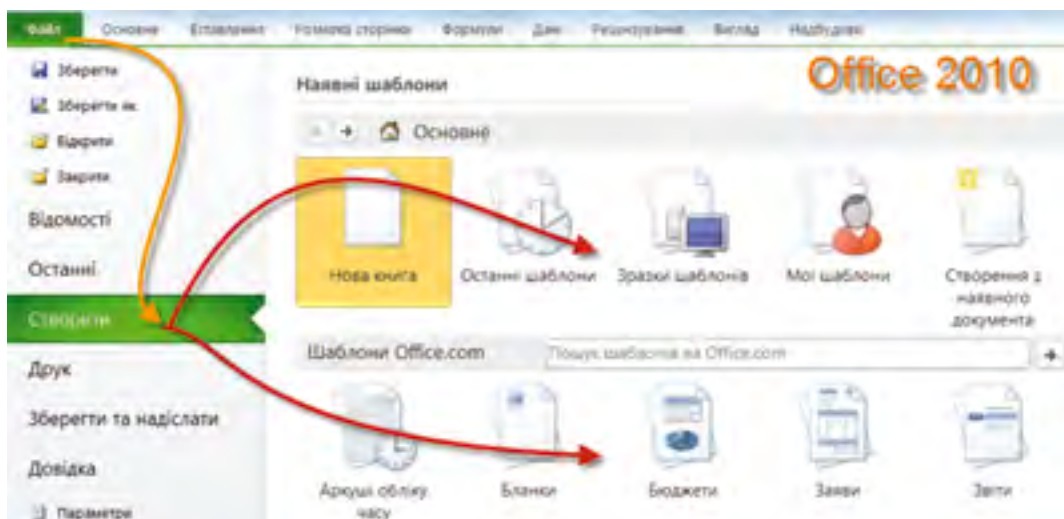


Рис. 4.10. Создание новой книги на основе шаблонов в приложении MS Office Excel 2010

По умолчанию программами Excel 2007 и Excel 2010 создается файл с расширением \* .xlsx, предназначенный для обработки в версии ЭТ, не более старой, чем та, в которой он создан.

Если предполагается просмотр созданной ЭТ в приложениях Office 2003 и более старых, при сохранении используют формат \* .xls.

## Выполняем

1. Откройте файл с ЭТ (по указанию учителя). Какие свойства этого файла отличаются от свойств текстовых документов, содержащих таблицы? 🌱
2. 👤 Запустите на выполнение программы MS Word и MS Excel. Какие подменю основного меню (вызываемых через закладки основного ленточного меню) меньше всего отличаются в этих программах, какие – больше всего? Почему? 🌱
3. 👤 Сравните содержимое ленточных меню Основное MS Word и MS Excel. Какие новые (по сравнению с меню текстового процессора) группы появились в меню Excel? Почему и зачем? ✚
4. 🏠 Сохраните электронную таблицу в форматах, отличных от того, в котором она была подана. ✚
5. 🏠 Найдите в Интернете объяснение особенностей форматов файлов \* .xml, \* .csv и \* .dif, \* .xlt, \* .xls. Запишите основные отличия между форматами файлов. ★



## Проверяем себя

1. Назовите основные форматы хранения ЭТ табличного процессора Microsoft Excel 2007. 🟢
2. Для чего предназначен формат \* .xml? Найдите в справке Excel 2010 необходимые сведения. 🟢
3. Для чего предназначены форматы \* .xlt и \* .xls? ➕
4. В какие программные средства возможен экспорт электронных документов, созданных в Excel 2010? ➕
5. Чем отличаются **Шаблоны** текстового процессора, презентационной системы и табличного процессора? Чем обусловлены эти отличия? ➕
6. 🏠 Найдите в справке Excel 2010 сведения по форматированию и свойствам ЭТ, созданных в приложении Excel 2010, которые не хранятся в файлах других форматов. ➕
7. 🏠 👤 Выполните по возможности пересылку телефонной книги мобильного телефона на компьютер (в файл \* .csv) и откройте этот файл текстовым редактором Блокнот, текстовым процессором и табличным процессором. Сделайте выводы. ★
8. Какие документы целесообразно выполнять в форме ЭТ? Почему? ➕
9. 👤 Откройте файл, содержащий электронную таблицу, в которой имеются макросы или подпрограммы, описанные языком Visual Basic for Applications. Какие ответы нужно давать на вопросы программы Excel 2007 для того, чтобы обеспечить выполнение макросов и подпрограмм? ★
10. Какими свойствами характеризуются объекты электронной таблицы: электронная книга, листы, строки, столбцы, ячейки (клетки), диапазон ячеек, диаграммы? ★

## 4.2. Данные в электронных таблицах



Типы данных. Форматы данных: числовой, текстовый, формат даты. Введение данных в ячейки (клетки). Автозаполнение. Редактирование данных таблицы. Форматирование данных, ячеек и диапазонов ячеек (клеток). Копирование, перемещение и удаление данных.



*Данные можно записывать непосредственно в ячейку или в строку формул.*

Прежде чем выполнять действия ввода, редактирования содержимого, копирование, перемещение ячеек ЭТ и т.п., необходимо выделить ячейки или их группы (блоки ячеек), то есть сделать их активными.

В Excel можно сразу заполнить содержимым диапазон ячеек. Заполнение может происходить как одинаковыми значениями, так и наборами

данных (списками), например, последовательными номерами или датами, названиями дней недели или месяцев. Кроме имеющихся в Excel встроенных списков, например, полных и сокращенных наименований для дней недели и месяцев, можно создавать и использовать при заполнении собственные списки.

✓ Для заполнения ячеек одинаковым содержимым нужно выполнить следующие действия. Ввести нужное содержимое в первую ячейку; переместить указатель мыши на ячейку с введенным значением. В правом нижнем углу выделенной ячейки появляется маленький черный прямоугольник – **маркер заполнения**. Переместить указатель мыши на маркер заполнения (указатель примет форму маленького крестика) и, удерживая нажатой левую кнопку мыши, протянуть **маркер заполнения** в нужном направлении через ячейки, которые должны быть заполнены (вправо, влево, вниз или вверх).

✓ Если содержимое выделенного сначала диапазона воспринимается Excel как начальные значения ряда данных (списка), то в случае использования данной процедуры заполнения ячейки будут заполнены не одинаковым содержимым, а последовательными значениями. Чтобы отключить заполнение ячеек значениями ряда при перемещении маркера заполнения, необходимо дополнительно удерживать нажатой клавишу Ctrl.

✓ Важным признаком, общим для всех электронных таблиц и табличных процессоров, является *представление данных в определенном, заранее описанном, виде*. Такое предварительное описание данных называют *типизацией данных, или описанием типов*. Его используют в большинстве языков программирования.

Зачем это делается? Мы знаем, что текст и числовые данные в памяти компьютера хранятся в виде последовательности кодов, которые являются числами. Однако, если кодами представлен текст, то к этим кодам, как и к тексту в целом, не могут быть применены арифметические действия. А из кодов, которыми описаны числа, нельзя составить текст. Поэтому предварительное описание данных требуется для **определения множества возможных действий над ними**. Над числовыми данными можно выполнять арифметические действия. Данные, представленные в виде текста, можно сравнивать с другими данными такого же типа, определять длину последовательности символов, добавлять и удалять фрагменты текста и т.п.

Однако для хранения данных, которые представляют номер дня недели (числа от 0 – воскресенье до 6 – суббота), достаточно одного байта, для данных, описывающих день месяца или код символа, – двух, для хранения значения физической величины, в зависимости от точности

измерений, нужно четыре байта и больше. Итак, предварительное указание типа данных дает возможность рационально использовать память компьютера.

**!** *Предварительное описание типов данных необходимо для рационального распределения памяти компьютера и определения действий, которые можно выполнять над конкретными данными.*

**!** *Содержимое ячейки следует рассматривать как объект, свойствами которого являются: тип, значение и формат отображения (рис. 4.11).*

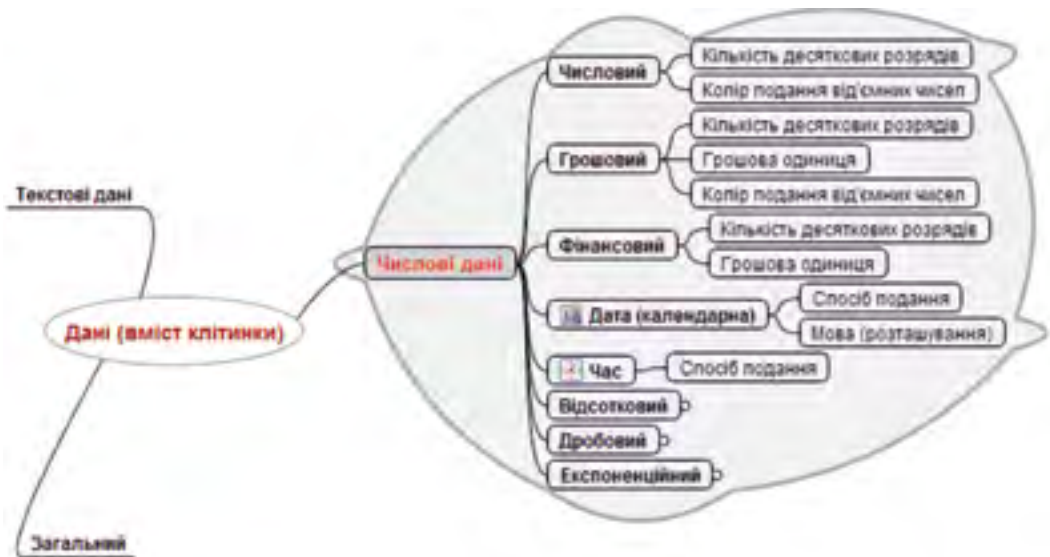


Рис. 4.11. Карта свойств объекта содержимое ячейки для типа числовые данные

Основные типы числовых данных и форматы их отображения показаны на рис. 4.12.

Для ЭТ важно не только установление форматов представления значений, но и форматирование ячеек.

**!** *Формат ячейки – это описание типа данных и формы их представления. Установление определенного формата ячеек называется их форматированием.*

**Форматирование абзацев и символов в ЭТ** имеет свои особенности, так как нужно не только располагать абзац определенным образом, но и учитывать возможность его размещения в ячейке под определенным углом, использовать форматирование, которое зависит от значения (например, для числового типа можно долг обозначить красным цветом).



При создании документов средствами ЭТ важны возможности использования различных способов заливки ячеек и начертания их границ (рис. 4.13).

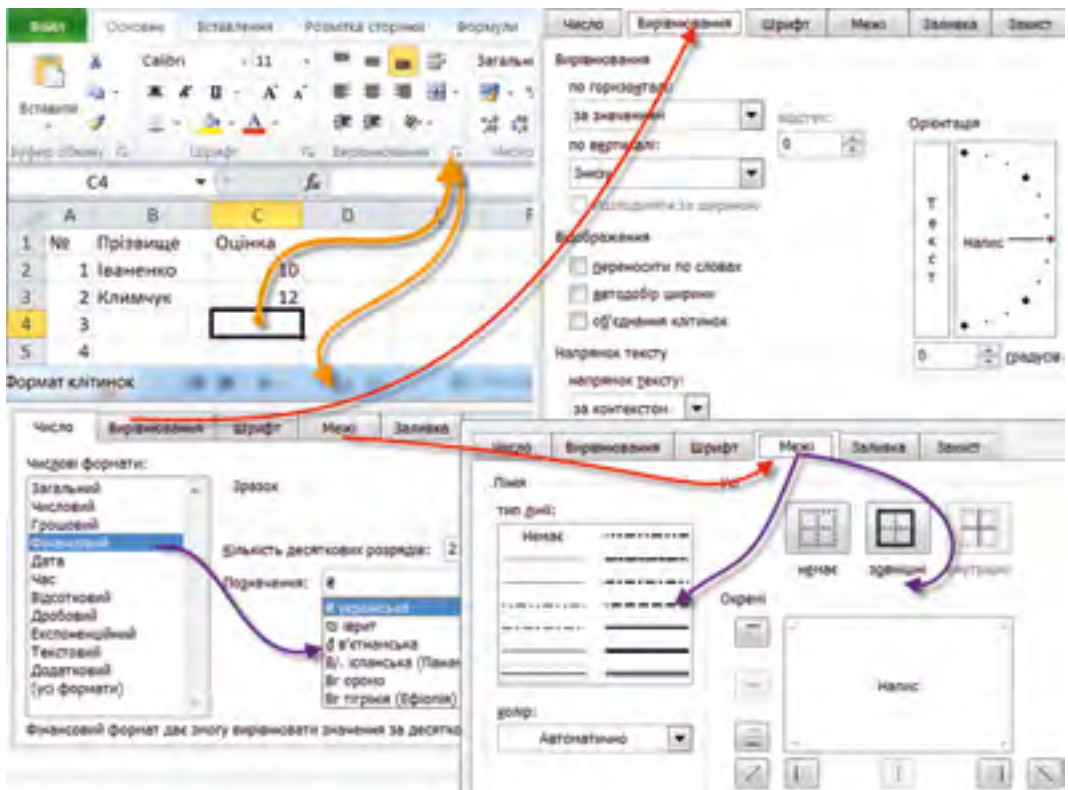


Рис. 4.13. Некоторые средства форматирования ячеек и их содержимого

✓ Некоторые форматы, в том числе формат **Дата**, распознаются автоматически, поэтому чтобы избежать искажения данных, желательно устанавливать нужные форматы заранее, до начала ввода данных.

По умолчанию программой Excel для содержимого ячеек устанавливается общий формат. Тип данных (числовой или текстовый) в этом случае определяется в процессе ввода.

В программах Excel 2007 и Excel 2010 команды форматирования содержатся в группах **Шрифт**, **Выравнивание**, **Число**, **Стили**, **Ячейки** ленточного меню **Основное**. Эти команды можно вызвать также из динамического меню.

Ячейки таблицы, которые необходимо форматировать, предварительно следует выделить – сделать ячейку активной или выделить диапазон ячеек. Различные способы выделения ячеек и диапазонов представлены в табл. 4.1.

Таблица 4.1

## Способы выделения ячеек и групп

№ п\п	Выделение	Действие
1	Текст в ячейке	Если режим редактирования в ячейке включен, выбрать ячейку. Если режим редактирования в ячейке выключен, выбрать ячейку и выделить текст в строке формул
2	Отдельная ячейка	Щелкнуть на ячейке левой кнопкой мыши или перейти к ней с помощью клавиш со стрелками
3	Диапазон ячеек	Нажать на первой ячейке диапазона левую кнопку мыши и перетащить указатель к последней ячейке
4	Большой диапазон ячеек	Щелкнуть на первой ячейке диапазона левой кнопкой мыши, нажать клавишу <b>Shift</b> и, удерживая ее, щелкнуть на последней ячейке диапазона. Для перехода к последней ячейке можно использовать полосы прокрутки
5	Все ячейки листа	Нажать кнопку <b>Выделить все (Ctrl + A)</b>
6	Несмежные ячейки или диапазоны ячеек	Выделить первую ячейку или первый диапазон ячеек, нажать клавишу <b>Ctrl</b> и, удерживая ее, выделить другие ячейки или диапазоны
7	Вся строка или столбец	Щелкнуть левой кнопкой мыши на заголовке строки или столбца
8	Смежные строки или столбцы	Протащить указатель по заголовкам строк или столбцов или выделить первую строку или столбец, а затем, удерживая клавишу <b>Shift</b> , выделить последнюю строку или столбец
9	Несмежные строки или столбцы	Выделить первую строку или столбец, нажать клавишу <b>Ctrl</b> и, удерживая ее, выделить другие строки или столбцы
10	Больше или меньше ячеек, чем охватывает текущее выделение	Нажать клавишу <b>Shift</b> и щелкнуть на последней ячейке, которую нужно включить к новому выделению. Будет выделена прямоугольная область между активной ячейкой и ячейкой, на которой щелкнули




После этого выполнить следующие действия: открыть правой кнопкой мыши контекстное меню и вызвать в нем команду **Формат ячеек** или вызвать команду **Формат ячеек** из группы **Ячейки**. На экране появится диалоговое окно **Формат ячеек**. По умолчанию в верхнем поле этого окна будет выбрана закладка **Число**, содержащая средства для форматирования чисел. В поле **Числовые форматы** приведены различные типы форматов, которые могут быть использованы (см. рис. 4.12). Щелчком левой кнопки мыши или с помощью клавиш управления курсором предоставляется возможность указать формат, обозначения, количество разрядов числа и т.п. В нижней части окна содержится поле просмотра **Образец**, в котором показано, как будет выглядеть содержимое ячейки в выбранном формате.

✓ *Количество разрядов числа, с которыми оно выводится на экран, не распространяется на число, которое сохраняется в памяти, и не влияет, таким образом, на точность вычислений.*

Для **перемещения данных** при помощи мыши необходимо: выделить ячейки (только смежные), установить указатель мыши в любом месте на границе выбранного диапазона (курсор примет форму четырех стрелок) и, удерживая нажатой левую кнопку мыши, перетащить весь диапазон на новое место.

**Копирование данных** можно осуществить аналогично, но дополнительно следует удерживать нажатой клавишу **Ctrl** (указатель мыши – стрелка будет дополнен знаком плюс +).

**Копирование или перемещение данных** возможно через Буфер обмена с использованием команды меню или кнопок группы Буфер обмена. Во время копирования: в группе **Буфер обмена** выполнить команду **Копировать** или нажать сочетание клавиш **Ctrl + C**, или воспользоваться кнопкой в группе **Буфер обмена**, или выбрать пункт **Копировать** из контекстного меню. Выделенные ячейки будут обозначены в рабочем листе рамкой, линия которой имеет вид подвижного пунктира. При перемещении: в группе **Буфер обмена** выполнить команду **Вырезать**, или нажать сочетание клавиш **Ctrl + X**, или воспользоваться кнопкой панели инструментов или выбрать пункт **Вырезать** из контекстного меню. Выделенные ячейки будут обозначены в рабочем листе «бегущей» рамкой.

Переместить указатель мыши и обозначить левую верхнюю ячейку диапазона, в который должна быть выполнена вставка ячеек. В группе **Буфер обмена** выполнить команду **Вставить** или нажать сочетание клавиш **Ctrl + V**, или воспользоваться кнопками , или выбрать пункт **Вставить** из контекстного меню.

Содержимое активной ячейки или выделенной области можно скопировать в соседние ячейки и путем перетаскивания **маркера заполнения** через нужные ячейки. При этом можно получить несколько копий, если количество ячеек, указывающих место для копии, будет кратным размеру выделенного диапазона.

Способы, использующие **Буфер обмена**, дают возможность при необходимости сделать несколько копий выделенных данных, выполняя несколько раз вставки в нужных позициях рабочего листа.

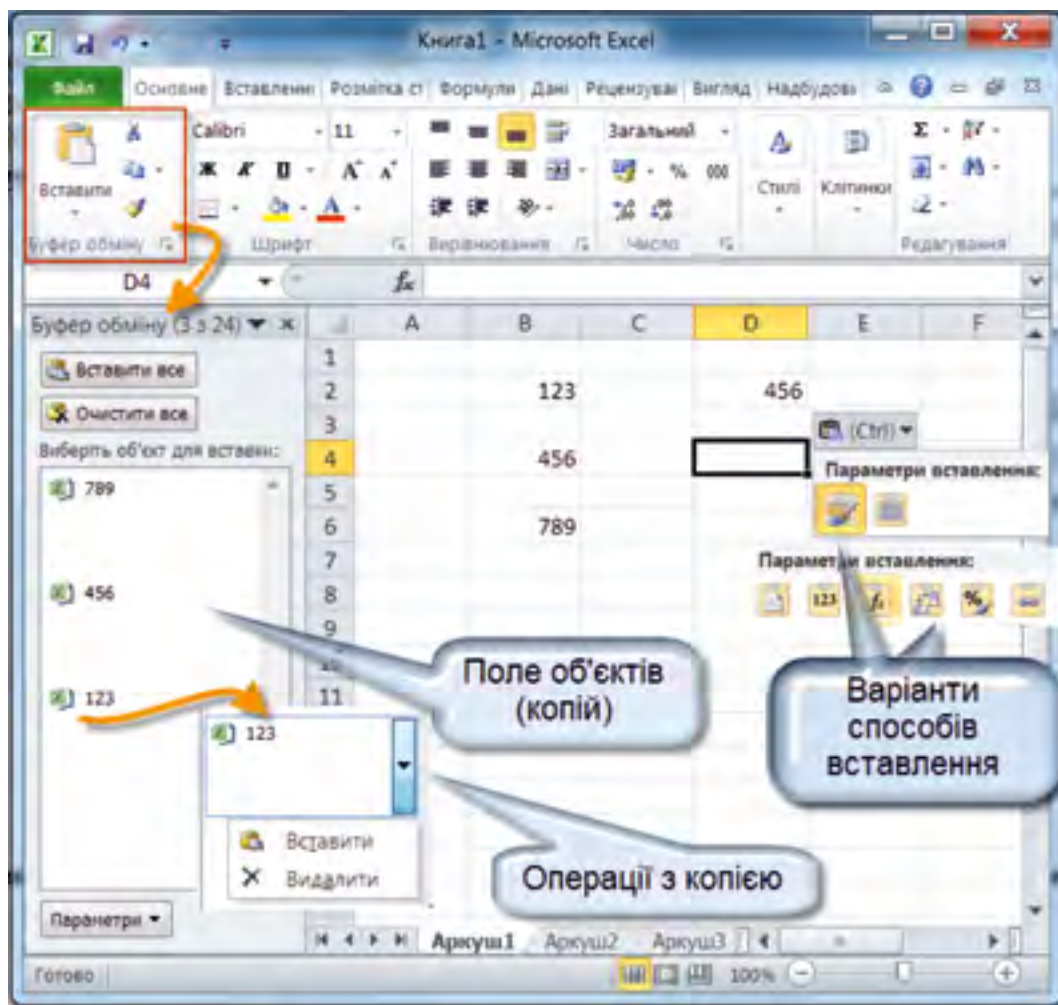


Рис. 4.14. Некоторые средства копирования содержимого ячеек

После выполнения операции копирования копия объекта находится в буфере обмена. При необходимости содержимое буфера можно открыть и выбрать один из необходимых для вставки объектов (рис. 4.14). Поскольку свойства содержимого копии в ЭТ могут быть разные, вариан-

тов вставки копии тоже может быть несколько. Эти варианты можно выбрать из меню **Параметры вставки** панели **Форматирование**.

Отдельные ячейки, а также целые строки или столбцы могут быть удалены из рабочего листа или вставлены в рабочий лист. При вставке пустых ячеек прилегающие ячейки будут смещены в заданном направлении.

Для вставки пустых ячеек с помощью меню: выделить столько ячеек, сколько должно быть вставлено (ячейки будут вставлены в позиции текущего выделения), ⇨ в меню **Вставка** ⇨ выполнить команду **Ячейки** ⇨ в появившемся окне диалога **Вставка** отметить направление сдвига прилегающих ячеек ⇨, установив переключатель в положение **Строка** или **Столбец**, вставить сверху или слева от выделенного диапазона столько пустых строк или столбцов, сколько содержит выделенный диапазон (рис. 4.15).

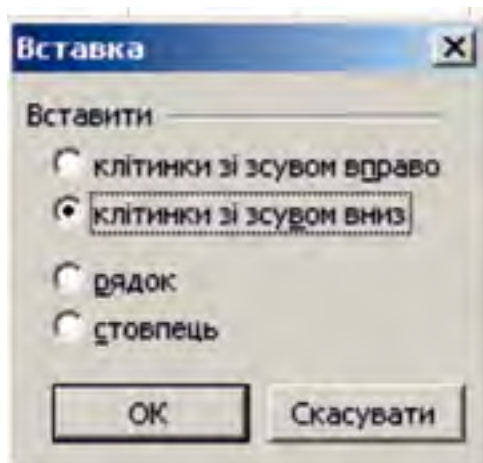






Рис. 4.15. Установка ячеек с использованием меню




Способы копирования, использующие буфер обмена, дают возможность копировать или перемещать данные также в другие листы рабочей книги.


### Проверяем себя

1. Зачем используют типизацию данных? ✦
2. Какие действия можно выполнять с текстовыми данными? ✦
3. В каких случаях целесообразно использовать автоматическое заполнение ячеек? ✦
4. Какие типы данных могут быть размещены в ячейках ЭТ табличного процессора Microsoft Excel 2007 (2010)? Найдите в справке Excel необходимые сведения. 📄

5.  Какие свойства может иметь объект *содержимое ячейки*, если в нее занесено значение температуры воздуха в г. Киеве? Каким, ориентировочно, будет интервал значений? ✚
6. Какие свойства может иметь объект *содержимое ячейки*, если в нее занесено значение расстояния между городами? Каким, ориентировочно, будет интервал значений? ✚
7. Чем отличаются Денежный и Финансовый форматы представления чисел? 🌱
8.  Сколько десятичных знаков целесообразно указать для числа, которым выражены: расстояние между городами в километрах, стоимость товара в гривнах, рост человека в сантиметрах, масса автомобиля в килограммах? Обоснуйте выбор для каждого случая. ✚
9. Назовите по крайней мере три способа выделения группы ячеек. 🌱
10.  Какие типы данных вы использовали, описывая алгоритмы в среде Скретч? Сравните их с теми, которые используются в ЭТ. Сделайте выводы. ✚
11. Какой тип имеет содержимое ячейки ЭТ по умолчанию? 🌱
12.  При изменении количества воспроизводимых десятичных разрядов в числе 28,6713 до одного в ячейке выведено значение 28,7. Почему? ✚
13. Назовите по крайней мере три способа копирования группы ячеек. 🌱
14. Как применить определенный формат к нескольким ячейкам? 🌱
15. Чем отличается очистка содержимого ячеек от их удаления? 🌱
16. Чем отличается копирование содержимого ячеек от его перемещения? 🌱
17. Что происходит с содержимым ячейки после его вырезания (рассмотрите два случая)? ✚

### Выполняем

1.   В поисковой системе выполните запрос: найти сведения по ключевым словам *типы данных*. Сравните найденные сведения с теми, которые вы получили, программируя в среде Скретч и обучаясь работать с электронными таблицами. Сделайте выводы. ✚
2.  Введите, начиная с ячейки A2 до A11, числа от 1 до 10. Установите для этих ячеек числовой формат с количеством десятичных знаков, равным десяти. Что происходит с шириной столбца? Уменьшите ширину столбца до девяти знакомест, перемещая мышью маркер ширины столбца в строке с буквами столбцов. Что происходит в ячейках? Какое сообщение возникает в них? ✚

3.  Как в нескольких ячейках ЭТ можно разместить текстовое название таблицы и выровнять его по центру (вкладка **Главное**, группа команд **Выравнивание**). Назовите таблицу «Мои учебники». Создайте строку заголовков: №, Название, Издательство, Год издания, Количество страниц. Столбик № заполните числами от 1 до 12, используя Автозаполнение. Заполните таблицу, используя, при необходимости, операции копирования и редактирования. ★
4. Создайте новый лист книги, назовите его «Информатика», перейдите на него (рис. 4.16.).

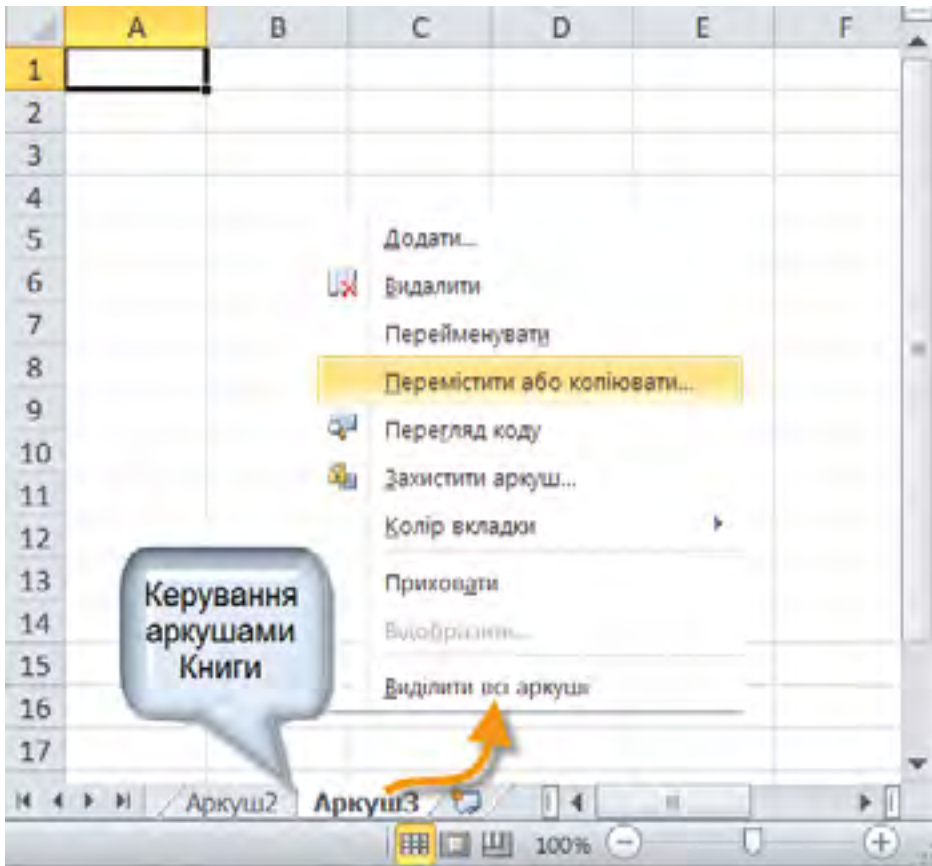


Рис. 4.16. Меню управления листами книги

Скопируйте на этот лист созданную таблицу. Обратите внимание на то, меняется ли при копировании форматирование ячеек и их содержимого. ✦

5. Создайте новый лист книги, назовите его «Данные\_2», перейдите на него (рис. 4.17). Создайте на нем таблицу, содержащую только числовые данные из таблицы «Мои учебники».



Для этого следует выполнить информационное связывание соответствующих ячеек.

Проще всего это выполняется следующим образом. Следует содержимое ячейки скопировать в буфер (с помощью комбинации клавиш Ctrl + C), затем перейти в ячейку, в которой должен появиться результат.

Затем нужно войти в меню **Специальная вставка** и в диалоговом окне этой директивы нажать на кнопку **Вставить связь** (рис. 4.17, 4.18).

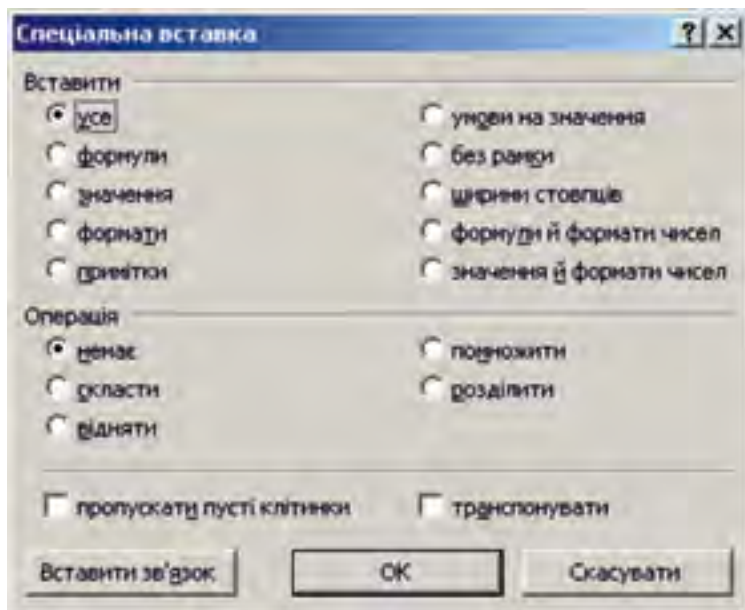


Рис. 4.17. Режим Специальная вставка (Office 2007)

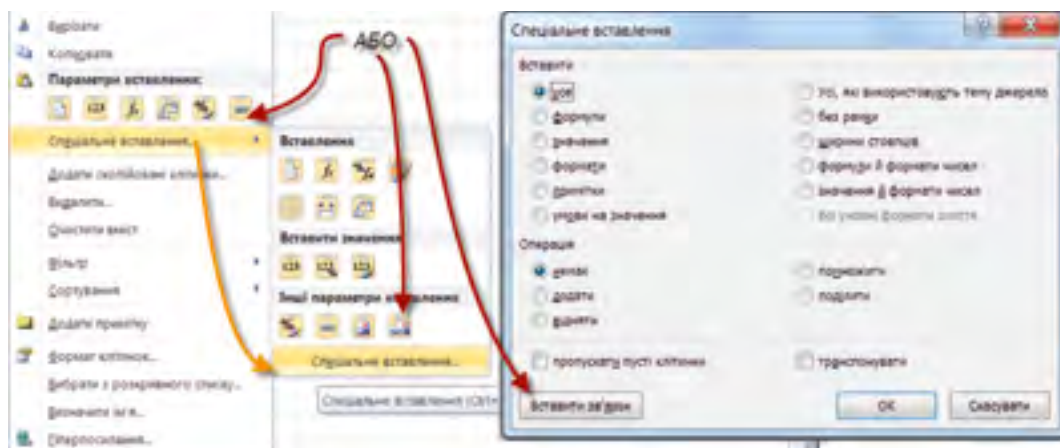


Рис. 4.18. Режим Специальная вставка (Office 2010)



### 4.3. Формулы в электронных таблицах



Правила записи формул в табличном процессоре. Копирование и перемещение формул. Понятие о модификации формул при копировании. Специальные случаи копирования.

Возможность использования формул и функций является одним из важнейших свойств программы обработки электронных таблиц.

Программа Excel автоматически интерпретирует введенные данные или как текст (с выравниванием по левой границе), или как числовое значение (с выравниванием по правой границе).

✓ Для ввода формулы необходимо ввести в ячейку алгебраическое выражение, которое должно начинаться со знака равенства (=).

Пусть в ячейке A1 таблицы содержится число 100, а в ячейке B1 – число 20. Чтобы разделить первое число на второе, а результат поместить в ячейку E1, в ячейку E1 нужно ввести формулу = A1 / B1 и нажать *Enter*.

✓ *Аргументом формулы называют данные (числовые или текстовые), над которыми выполняются определенные действия для вычисления значения функции. В программировании аргументы называют также операндами.*

✓ *Введение формулы можно упростить: в ячейку, где будет размещен результат вычисления, ввести знак равенства (=) → щелкнуть левой кнопкой мыши на ячейке с первым операндом → в формуле появится адрес первой ячейки с данными → ввести знак операции → щелкнуть левой кнопкой мыши на ячейке со вторым операндом → в формуле появится адрес второй ячейки → нажать *Enter*.*

В качестве аргументов в формуле обычно используются числа и адреса ячеек (рис. 4.19).

	A	B	C	D	E	F
1	2	5	4	1	2	11

Рис. 4.19. Пример вычислений с использованием формулы

Для обозначения арифметических операций могут использоваться следующие символы: + (сложение); – (вычитание); \* (умножение); / (деление); ^ (возведение в степень).

Формула может содержать ссылки на ячейки, которые расположены на другом рабочем листе или даже в таблице другого файла. Один раз введенная формула может быть в любое время отредактирована. Про-

программа Excel позволяет работать со сложными формулами, содержащими несколько операций. Для наглядности можно включить текстовый режим, в котором программа Excel будет выводить в ячейках не результат вычисления, а собственно формулу.

Для редактирования содержимого ячейки (ячеек) ее (их) нужно сначала выделить. Далее следует включить режим редактирования, нажав на клавиатуре клавишу F2 или выполнив двойной щелчок левой кнопкой мыши. Редактирование формулы можно осуществлять также в строке содержимого активной ячейки.

Применение сложных формул можно увидеть на примере, поданном на рис. 4.20.

	C	D	E	F	G	H
1						
2		робота	час. год	вартість, грн	сума, грн	
3		Встановлення і налагодження ОС	4	35,00 грн.	140,00 грн.	
4		Підключення до локальної мережі	0,5	20,00 грн.	10,00 грн.	
5		Встановлення офісного програмного забезпечення	3	25,00 грн.	75,00 грн.	
6		<b>СУМА</b>			<b>225,00 грн.</b>	
7		Податок на додану вартість (ПДВ)			45,00 грн.	
8		<b>Сума з ПДВ</b>			<b>270,00 грн.</b>	

Рис. 4.20. Применение сложных формул для вычисления общей стоимости выполненных работ

Пусть в столбце D таблицы указано время (в часах), затраченное на выполнение работы, в столбце E – стоимость одного часа работы, а в столбце F – промежуточная сумма, которую необходимо уплатить за конкретный вид работы.

В ячейке F8 надо вычислить общую стоимость всех работ. Для этого в эту ячейку следует записать соответствующую формулу, аргументами которой будут: время, затраченное на выполнение работы каждого вида (столбец D), стоимость определенного вида работы (столбец E).

Для того чтобы увидеть промежуточные результаты, используют поэтапное выполнение вычислений подобно тому, как бы их выполняли без использования компьютера, на бумаге.

Итак, сначала вычисляем стоимость каждой работы отдельно, находя произведение ячейки столбца D и ячейки столбца E той же строки.

Для вычисления конечной суммы, подлежащей оплате (в ячейке F8), надо сначала сложить промежуточные суммы, вычислить значение налога на добавленную стоимость и полученные значения сложить.

Стрелками на рис. 4.20 показана последовательность выполнения действий и передачи значений из ячейки в ячейку в процессе выполнения вычислений.

Иногда нужно создать таблицу, в строках которой содержатся одинаковые или почти одинаковые формулы. Например, необходимо рассчитать объем накопления для вклада, проценты на который начисляются ежемесячно.

Алгоритм расчета достаточно прост. Нужно к остатку на счете ежемесячно добавлять этот же остаток, умноженный на 1/12 годового процента. Это равносильно тому, чтобы умножать остаток на счете ежемесячно на  $1 + (\text{годовой процент})/1200$ .

Электронные таблицы для такого расчета представлены на рис. 4.21.

f <sub>3</sub> =B3*1,01			
	A	B	C
1	Місяць	Залишок	Початковий внесок, грн
2	1	101,00€	100,00€
3	2	102,01€	
4	3	103,03€	
5	4	104,06€	
6	5	105,10€	
7	6	106,15€	
8	7	107,21€	
9	8	108,29€	
10	9	109,37€	
11	10	110,46€	
12	11	111,57€	
13	12	112,68€	

f <sub>3</sub> =B12*(SD\$2/1200+1)				
	A	B	C	D
1	Місяць	Залишок	Початковий внесок, грн	Річний відсоток
2	1	101,00€	100,00€	12
3	2	102,01€		
4	3	103,03€		
5	4	104,06€		
6	5	105,10€		
7	6	106,15€		
8	7	107,21€		
9	8	108,29€		
10	9	109,37€		
11	10	110,46€		
12	11	111,57€		
13	12	112,68€		

Рис. 4.21. Применение модификации формул при копировании и фиксировании адреса одного из аргументов


Для создания таблицы в столбец B для ячеек с B3 по B13 вводим формулу: «Число из предыдущей ячейки умножить на 1+ (годовой процент)/1200». Пусть годовой процент составляет 12 %. Тогда для B2 нужно ввести формулу = C2 \* 1,01; для B3 – = B2 \* 1,01; для B4 – = B3 \* 1,01 и так далее, до B13 – = B12 \* 1,01.


Для создания таких таблиц предусмотрена модификация формул при их копировании. Если в ячейку В3 ввести формулу  $= B2 * 1,01$ , а после этого скопировать ее в ячейки с В4 по В13, получим нужный результат. Созданная таким образом таблица представлена на рис. 4.21 слева. Определенное неудобство таблицы остается, поскольку множитель для ежемесячного начисления процентов мы вычисляли отдельно. При изменении этого процента таблицу придется переделывать.

На рис. 4.21 справа представлена таблица, созданная таким образом, чтобы пользователь мог ввести значение первоначального взноса в ячейку С2 и годового процента – в ячейку В2, после чего в соответствующих ячейках столбца В получить ежемесячное значение вклада.

Для того чтобы понять, как это сделано, обратите внимание на содержание строки формул, а именно  $= B12 * (\$ D \$ 2/1200 + 1)$ .

Адрес  $\$D\$2$  ячейки, в которую предполагается вводить значение годового процента, записан со знаком  $\$$  перед номерами столбца и строки. Это означает, что при копировании этот адрес не будет модифицироваться (сохранится неизменным).

 *Адреса ячеек (и диапазонов ячеек), которые не изменяются при копировании формул, называются **абсолютными**.*

 *Абсолютной является и адресация с использованием имен.*

При вводе формул и при вычислениях могут возникать ошибки.

Если ошибка заключается в нарушении правил записи, то программа выдаст сообщение, подобное представленному на рис. 4.22, и предложит способ ее исправления (закрывать скобку, добавить знак операции и т.д.).

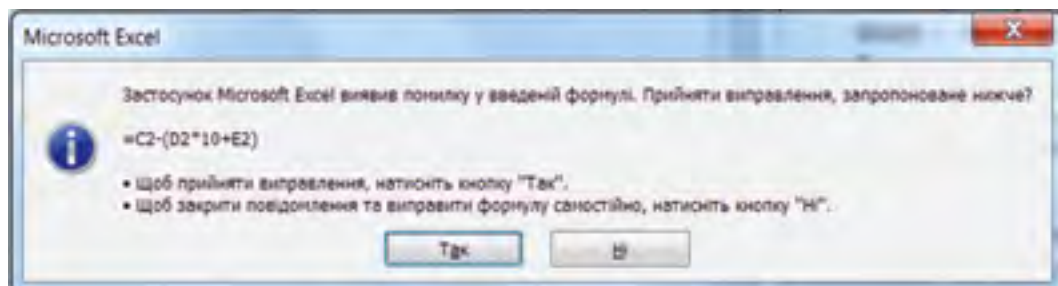


Рис. 4.22. Диалоговое окно с сообщением об ошибке

Если ошибка возникает при выполнении вычисления, в ячейку вместо результата может быть выведено соответствующее сообщение.

# DIV / 0! (# ДЕЛ / 0!) – попытка деления на ноль;

#NAME? (# ИМЯ?) – ошибка в названии функции или адреса;

# NUM! (# ЧИСЛО!) – в формуле аргументом должно быть число;  
# N / A (# Н / Д) – вычисление невозможно из-за отсутствия данных;  
### – ширина ячейки недостаточна для числа.

Для исправления таких ошибок следует проанализировать формулу и отредактировать или внести изменения в ячейки с операндами.


### Проверяем себя


1. Что называется операндом? Что является операндом формулы = A6? 🌱
2. Над данными каких типов допустимо выполнение арифметических операций? 🌱
3. Что называется синтаксической ошибкой (ответ найдите в третьем разделе учебника)? ✚
4. 🧑🧑 Какого типа должно быть содержимое ячеек A9 и B9 для выполнения вычислений по формуле = A9/B9? ✚
5. 🧑🧑 Какие действия необходимо предпринять при возникновении сообщения # DIV / 0! (# ДЕЛ / 0!)? ✚
6. 🧑🧑 Какие действия необходимо предпринять при возникновении сообщения #NAME? (# ИМЯ?)? ✚
7. 🧑🧑 Какие действия необходимо предпринять при возникновении сообщения ## NUM! (# ЧИСЛО!)? ✚
8. 🧑🧑 Какие действия необходимо предпринять при возникновении сообщения # N / A (# Н / Д)? ✚
9. 🧑🧑 Какие действия необходимо предпринять при возникновении сообщения ###? 🌱
10. В каких случаях целесообразно использовать имена как ссылки на данные? ✚
11. В каких случаях целесообразно использовать абсолютные адреса для ссылок на данные? 🌱
12. В каких случаях целесообразно использовать относительные адреса для ссылок на данные? 🌱
13. Что происходит при копировании формулы = \$ B6\*10, расположенной в ячейке B7, в ячейки с C5 по C11? 🌱
14. Что происходит при копировании формулы = \$ B \$ 6\*C1, расположенной в ячейке B7, в ячейки с C5 по C11? ✚

### Выполняем

1. 🧑🧑 Создайте ЭТ для вычисления значений  $y = 10 - 5 \cdot x$  для значений  $x$  от 0 до 10 с шагом 1. Для создания значений  $x$  используйте авто-

матическое заполнения ячеек. Как сделать так, чтобы шаг последовательности данных можно было бы менять без изменения формулы? ✦

2.  Создайте ЭТ для вычисления значений  $y = 2 \cdot x^2 - 5 \cdot x + 10$  для значений  $x$  от  $-10$  до  $+10$  с шагом 1. 🌿

3.  Опишите в словесной форме последовательность действий, которые необходимо выполнить для решения такой задачи: «Для выпечки одной буханки ржаного хлеба нужно 300 г ржаной муки, 200 г пшеничной муки, 10 г соли, 30 г сахара, 5 г дрожжей; для выпечки одной буханки белого хлеба нужно 600 г пшеничной муки, 10 г соли, 60 г сахара, 5 г дрожжей. Сколько нужно каждой из составляющих для выпечки 10 буханок ржаного и 20 – белого хлеба?» Проверьте рассуждения вычислениями, придумайте несколько подобных задач и запишите в тетрадь их условия. Создайте ЭТ для выполнения вычислений. ★

#### 4.4. Встроенные функции



**Использование встроенных функций: сумма, среднее значение, min, max и некоторых других.**

Поскольку некоторые формулы и их комбинации встречаются очень часто, табличный процессор Excel содержит более двухсот заранее запрограммированных формул, которые называются **функциями**.

Например, для того чтобы вычислить среднее арифметическое значение чисел, содержащихся в клетках от B2 до B6, нужно записать формулу  $= (B2 + B3 + B4 + B5 + B6) / 5$ , а с использованием встроенной функции  $= AVERAGE (B2: B6)$ .



**Встроенные функции** – это именованные описания последовательности вычислений с указанными операндами (аргументами).

Аргумент, или операнд, функции может занимать одну ячейку или размещаться в группе ячеек, быть переменной или константой, числом или текстом.

Используются аргументы разных типов: число, текст, логическое значение («истинно» и «ложно»), массив, значение ошибки (например, # N / A) или ссылка на ячейку или группу ячеек. В каждом отдельном случае нужно использовать соответствующий тип аргумента. Константы, формулы или функции также используются как аргументы.

✓ **Синтаксис записи обращения к функции таков:** запись начинается со знака равенства (=), далее указывается имя функции, открывающая скобка, список аргументов, разделенных точками с запятыми, закрывающая скобка (рис. 4.23).



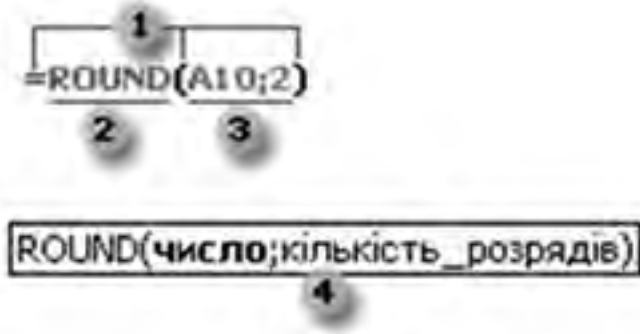


Рис. 4.23. Структура описания функции, осуществляющей округление значений:

- 1 – структура обращения к функции (вызова функции);
- 2 – имя функции; 3 – аргументы;
- 4 – подсказка синтаксиса.

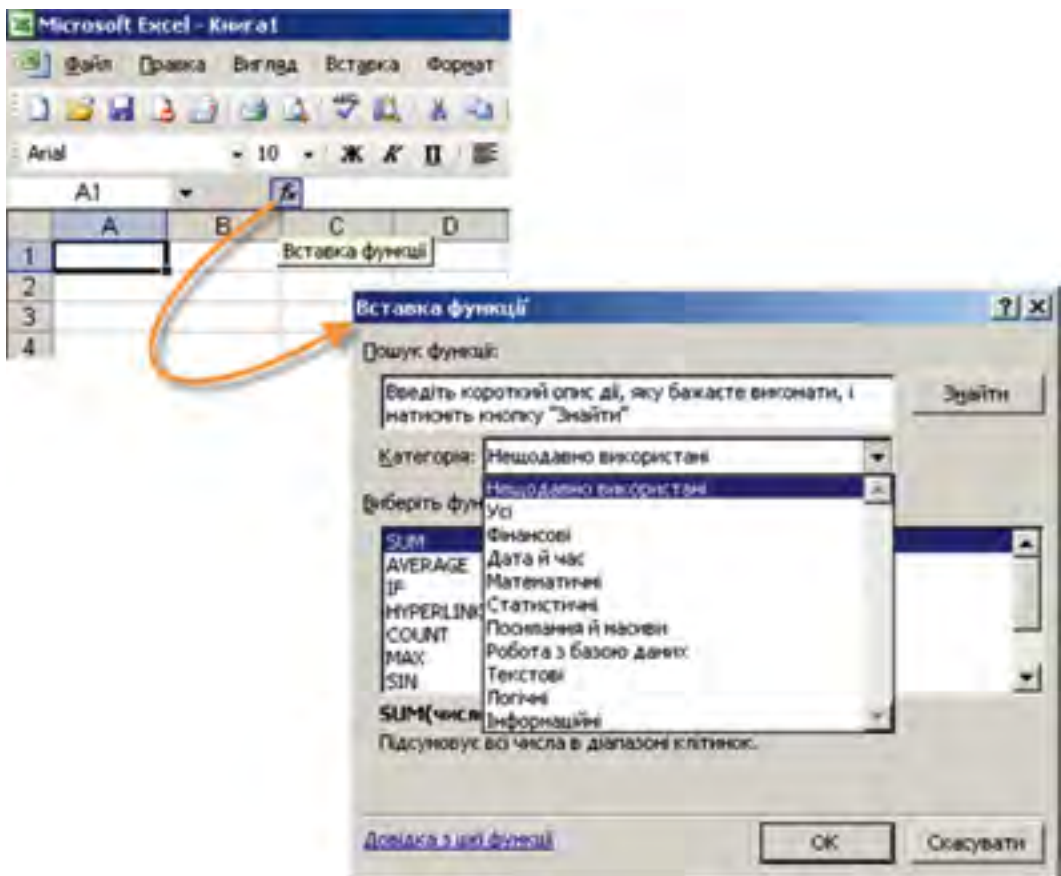


Рис. 4.24. Вставка функции (Office 2007)

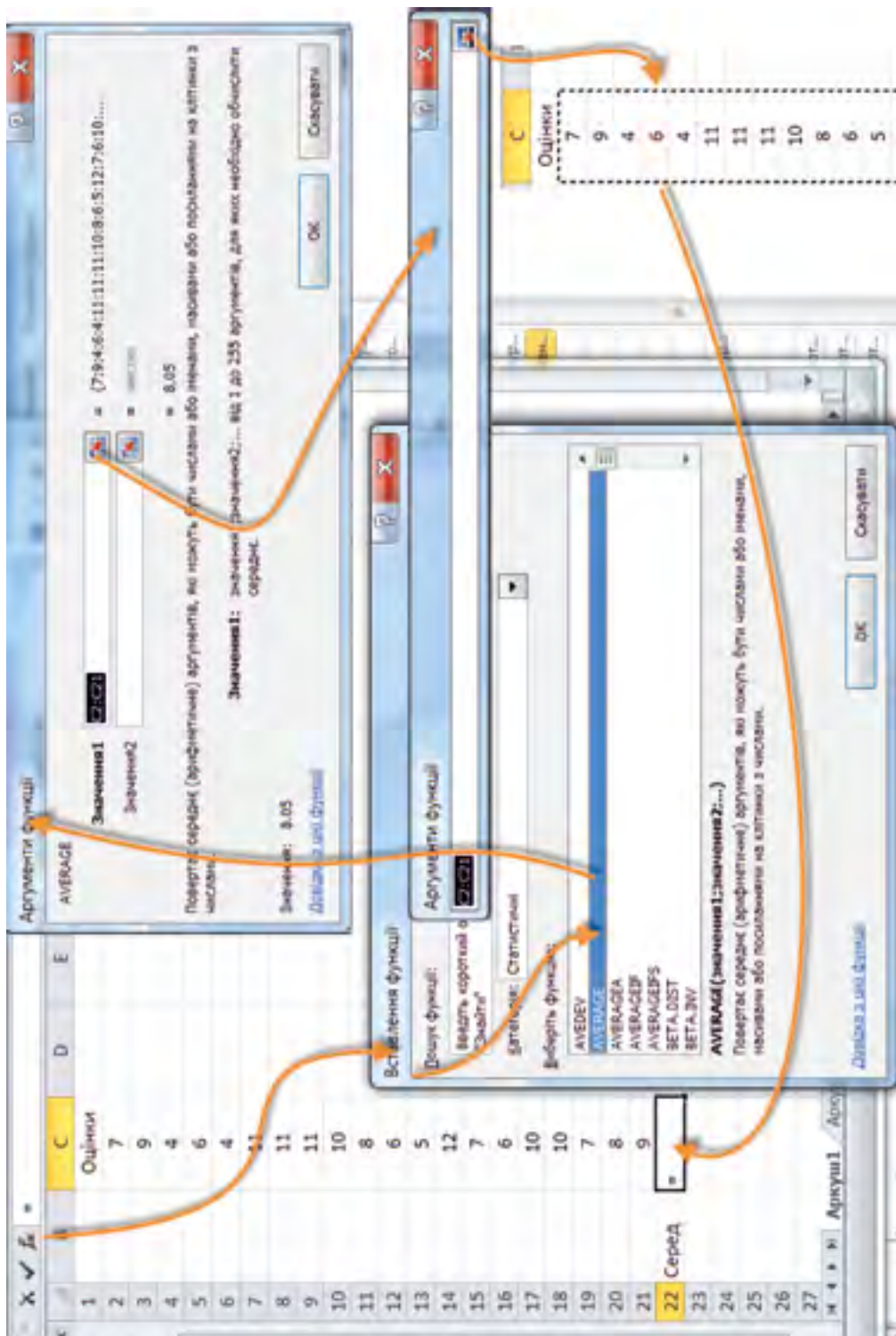


Рис. 4.25. Вставка функції (Office 2010)

- ✓ *Встроенная функция может иметь несколько аргументов или вообще их не иметь.*

Например, встроенная функция округления = ROUND (B1 / C1; 2) имеет два аргумента, функции PI () (возвращает число π) и RAND () (возвращает случайное число от 0 до 1), TODAY () (возвращает текущую дату) не имеют аргументов.

- ✓ *Аргументами функций могут быть значения, которые возвращает другая функция.*

Например, известная вам условная конструкция (если ... то ... иначе) в ЭТ реализуется функцией IF (логическое\_выражение, [значение\_если\_истинно], [значение\_если\_ложно]). Операндом функции является результат вычисления логического выражения, а возвращать она будет, в зависимости от истинности или ложности значения логического выражения, значения, которые также могут быть результатом вычислений.

Для ввода функций используется команда **Вставка функции**, кнопка которой расположена в **Строке формул** (левее окна ввода).

В результате нажатия кнопки **Вставка функции** открывается окно **Вставка функции**, которое содержит упорядоченный по категориям список функций. В этом списке следует найти необходимую категорию (Недавно использованные, Математические, Статистические и т.п.) функции, а в окне **Выберите функцию** – нужную функцию. При наведении курсора на имя функции внизу списка выводится сообщение-подсказка с ее кратким описанием.

Если функция не найдена, ее поиск можно выполнять по категориям, нажав левую кнопку мыши на нужном имени в списке **Категории** или воспользоваться командой **Найти**.

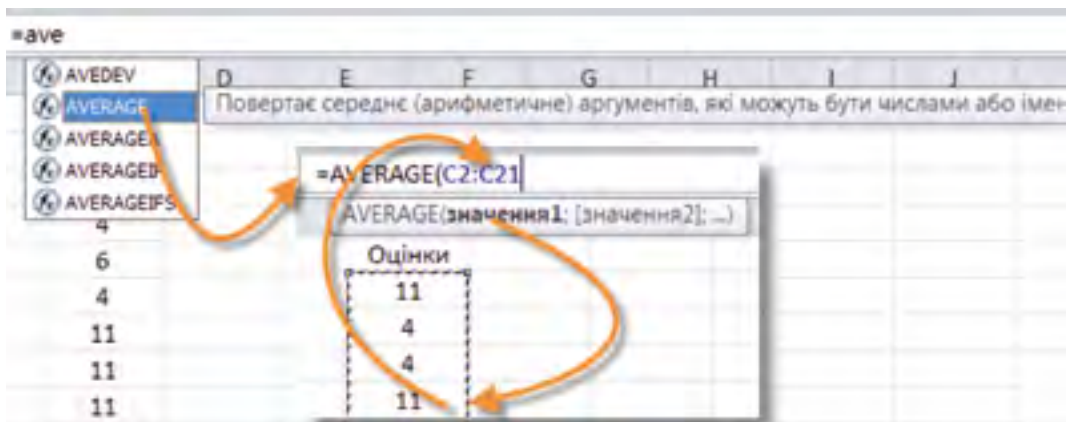


Рис. 4.26. Ввод функции (Office 2010), упрощенная схема ввода

Если вы помните по крайней мере несколько первых букв названия функции, то достаточно ввести их после знака =. После этого появится меню с подсказками, как показано на рис. 4.26. В этом меню можно ввести аргументы функции в виде адресов ячеек или адресов диапазонов ячеек и завершить ввод функции, введя закрывающую скобку.

Особенности применения каждой из встроенных функций можно найти в Справке Excel несколькими путями. Можно, как и во всех приложениях Office, вызвать справку нажатием клавиши F1, как показано на рис. 4.27.

Также в процессе ввода функции можно получить справку о функции, которая вводится. На рис. 4.24 показан процесс введения функции с подсказкой в электронных таблицах Office 2007. В Office 2010 этот процесс несколько усовершенствован, как показано на рис. 4.25.

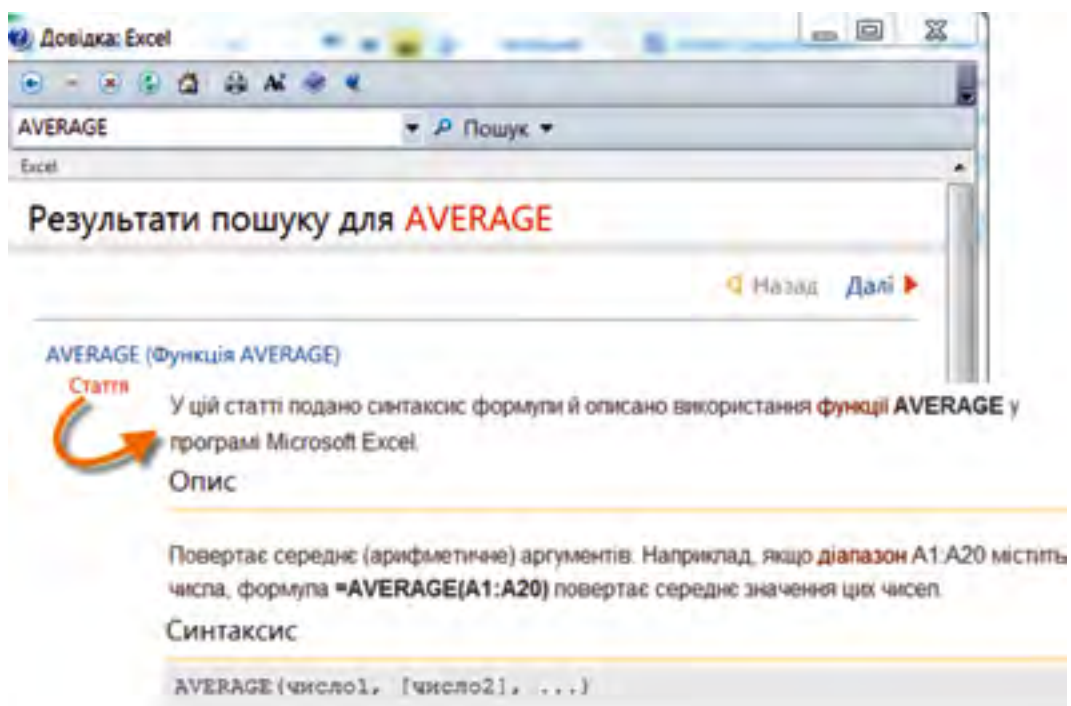


Рис. 4.27. Использование Справки Excel

Некоторые встроенные функции, необходимость обращения к которым чаще всего возникает в реальной работе с электронными таблицами, представлены в табл. 4.2.

Таблица 4.2





**Некоторые встроенные функции, их синтаксис и назначение**

№ п\п	Название и синтаксис	Назначение
1	SUM (аргумент1; аргумент2 ...) или СУММ (...)	Функция добавляет все числа, указывающиеся в качестве аргументов. Каждый аргумент может быть диапазоном или результатом, полученным с помощью другой функции
2	MAX (аргумент1; аргумент2 ...) или МАКС (...)	Возвращает наибольшее значение из набора значений
3	MIN (аргумент1; аргумент2 ...) или МИН (...)	Возвращает наименьшее число в наборе значений
4	ROUND (число, количество разрядов) или ОКРУГЛ (...; ...)	Функция ROUND округляет число до указанного количества разрядов
5	AVERAGE (аргумент1; аргумент2 ...) или СРЗНАЧ (...)	Возвращает среднее (арифметическое) аргументов
6	IF (лог_выражение, [значение_если_истинно], [значение_если_ложно]) или ЕСЛИ (...; ...; ...)	Функция IF возвращает одно значение, если вычисленное значение заданного условия – «истинно», и другое значение, если вычисленное значение заданного условия – «ложно»





**Проверяем себя**

1. С какой целью созданы встроенные функции Excel? 🌟
2. Что называется аргументом функции Excel? Как еще можно называть аргумент функции Excel? 🌟
3. Что может быть операндом? Сколько операндов может иметь функция Excel? ✦
4. Приведите примеры функций, не требующих аргументов. ✦




5. Запишите формулу, которая возвратит наибольшее (наименьшее) число из ячеек A1:A100. ✦
6.  Можно ли в ячейку записать дату создания таблицы? Как сделать так, чтобы эта дата осталась неизменной? ★
7.  Запишите формулу, которая будет возвращать квадрат суммы данных из ячеек от A1 до A20. ★
8. Как вычислить среднее арифметическое чисел из ячеек от E2 до E8? 🌿
9.   Запишите формулу, по которой в ячейки от A1 до A20 будут занесены случайные числа от 0 до 1. Как сделать так, чтобы эти числа не изменялись в дальнейшем? ★

### Выполняем

1.  Введите в ячейку B1 формулу  $=IF(C1 > C2; C1; C2)$ . Вводя в ячейки C1 и C2 разные числа, опишите, как осуществляется формирование значения в ячейке B1. ✦
2.  Создайте таблицу квадратов чисел от 1 до 20 (один столбец). 🌿
3.  Создайте калькулятор степеней введенного числа. ✦
4.  Создайте таблицу квадратов чисел от 1 до 100, как показано на рис. 4.28. Формулы модифицируйте таким образом, чтобы клетки от B2 до L10 заполнить одним копированием. ★

	A	B	C	D	E	F	G	H	I	J	K	L
1		0	1	2	3	4	5	6	7	8	9	10
2	10	100	121	144	169	196	225	256	289	324	361	400
3	20	400	441	484	529	576	625	676	729	784	841	900
4	30	900	961	1024	1089	1156	1225	1296	1369	1444	1521	1600
5	40	1600	1681	1764	1849	1936	2025	2116	2209	2304	2401	2500
6	50	2500	2601	2704	2809	2916	3025	3136	3249	3364	3481	3600
7	60	3600	3721	3844	3969	4096	4225	4356	4489	4624	4761	4900
8	70	4900	5041	5184	5329	5476	5625	5776	5929	6084	6241	6400
9	80	6400	6561	6724	6889	7056	7225	7396	7569	7744	7921	8100
10	90	8100	8281	8464	8649	8836	9025	9216	9409	9604	9801	10000






Рис. 4.28. Таблица квадратов чисел от 10 до 100

5.  Создайте калькулятор степеней числа, как показано на рис. 4.29. Формулы модифицируйте таким образом, чтобы клетки от C2 до C9 заполнить одним копированием. Какие два варианта форматирования возможны для ячеек от C2 до C9? Какой из них был использован и где? ★



	A	B	C
1	Число	Показник	Степень
2	-1,8	0	1
3		1	-1,8
4		2	3,24
5		3	-5,832
6		4	10,4976
7		5	-18,89568
8		6	34,012224
9		7	-61,222003

Рис. 4.29. Таблица степеней чисел

6.  Создайте таблицу, в которой клетки от A1 до A10 будут зафиксированными случайными числами в диапазоне от 0 до 20 с одним десятичным знаком. ✚
7.  Создайте таблицу, в которой для введенного в ячейку A2 числа в ячейке B2 появлялся бы его квадрат, в ячейке C2 – обратное число. Что будет, если в ячейку A2 ввести отрицательное число? Что будет, если в нее ввести ноль? ✚
8.  Дополните таблицу, созданную для выполнения предыдущего задания, формулами, которые вместо стандартного сообщения об ошибке выводили бы сообщение «Делить на ноль невозможно». ★
9.   Создайте таблицу, в которой бы столбец A от второй до двадцать первой строки был заполнен числами, каждое последующее из которых было бы больше предыдущего на постоянное число, которое можно менять, не редактируя формул. Где можно использовать такую заготовку? ✚

## 4.5. Диаграммы



Точечные, столбчатые и секторные диаграммы, их объекты и свойства. Создание и форматирование столбчатых и секторных диаграмм в среде табличного процессора. Анализ данных, представленных на диаграмме.



**Диаграммы** – это средства наглядного представления данных, с помощью которых легче осуществлять сравнение, выявление закономерностей и тенденций изменений данных.

Диаграмма создается на основе числовых и текстовых данных, размещенных в ЭТ. Диаграмму можно расположить рядом с таблицей или поместить на отдельном листе.

Поскольку диаграмма отображает данные ЭТ, то изменение данных приводит к изменению диаграммы. Этим можно воспользоваться для моделирования зависимостей между величинами, исследования процессов и тому подобное.

Современные табличные процессоры обеспечивают построение большого количества типов диаграмм.

✓ **Точечная диаграмма** (рис. 4.30) используется для наглядного представления зависимости между двумя величинами, поскольку обе оси этой диаграммы являются осями значений.

Точечная диаграмма может быть использована для построения графиков функций. Для построения необходимы два ряда значений. Тип диаграммы наиболее подходящий для построения графиков зависимостей между величинами.

✓ **Столбиковая диаграмма** (рис. 4.31) используется для сравнения нескольких рядов значений.

✓ **Горизонтальная ось** является осью категорий, поэтому сравниваемые значения (например, производство какого-либо товара за один день) должны быть определенным образом упорядочены.

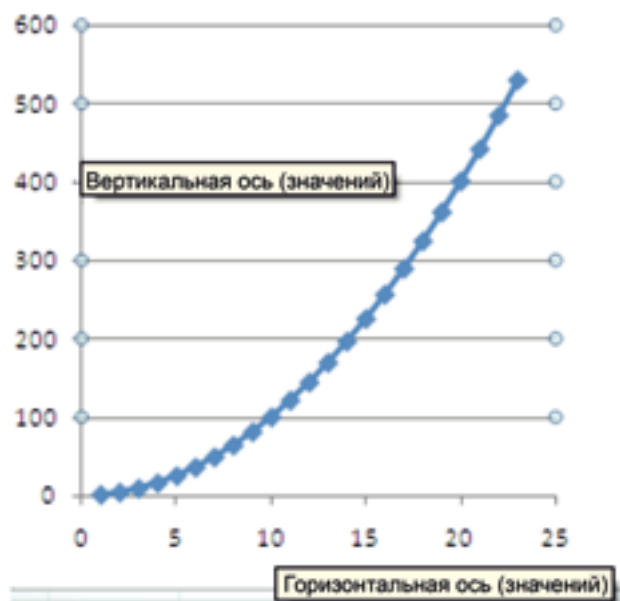


Рис. 4.30. Точечная диаграмма, на которой отображена зависимость  $y = x^2$  в интервале от 1 до 22 с шагом 1

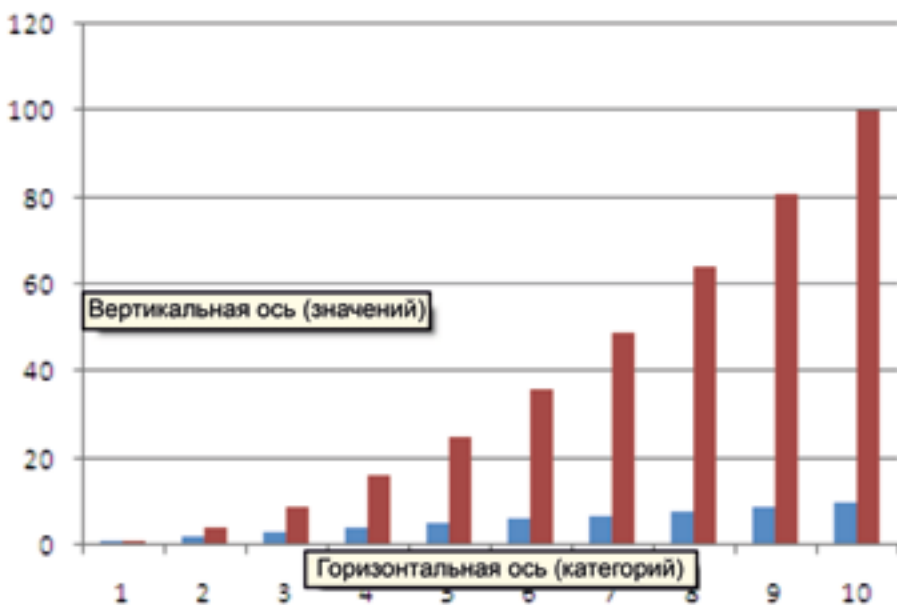


Рис. 4.31. Столбиковая диаграмма (отображены значения двух рядов, наименование делений на оси категорий выполнено автоматически)

✓ Для построения диаграммы достаточно одного ряда значений.

Поскольку одна из осей диаграммы является осью категорий, для нее нужны только данные для создания подписей делений, иначе они будут пронумерованы числами от 1 до  $n$ , где  $n$  – количество значений ряда, в котором их больше (если выполняется построение диаграммы для нескольких рядов значений).

Столбиковая диаграмма может быть создана при различных расположениях осей.

✓ Если необходимо проанализировать как общую динамику изменений определенной величины в зависимости от времени, так и вклад нескольких разных источников в значение этой величины, используют столбиковую диаграмму с накоплением.

Пример столбиковой диаграммы с накоплением представлен на рис. 4.32. Отображены значения четырех рядов, наименование делений на оси категорий выполнено названиями месяцев из ячеек А3: А13 (заголовками строк), наименование рядов данных выполнено по названиям городов, в которых расположены филиалы (ячейки В2: Е2, заголовки строк). Высота каждого столбца на диаграмме отображает суммарную стоимость товара, который ежемесячно производится всеми четырьмя филиалами вместе, а высота составляющих каждого столбца – стоимость товара, производимого каждым филиалом отдельно.

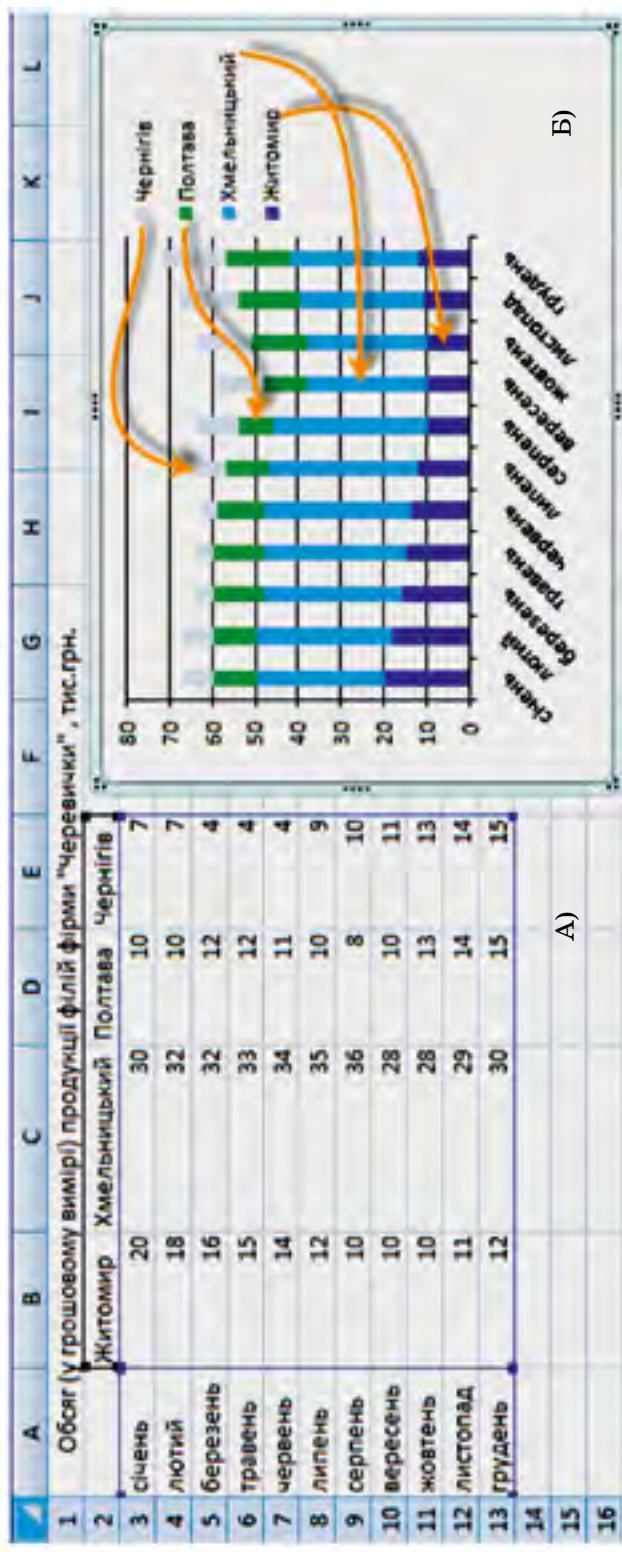


Рис. 4.32. Столбиковая диаграмма с накоплением:  
 А) – таблица значений; Б) – созданная на ее основе диаграмма

✓ *Секторная диаграмма (рис. 4.33) используется для наглядного представления содержания составляющих в целом, например, чистых металлов в сплаве, содержания питательных веществ в определенном пищевом продукте и тому подобное.*

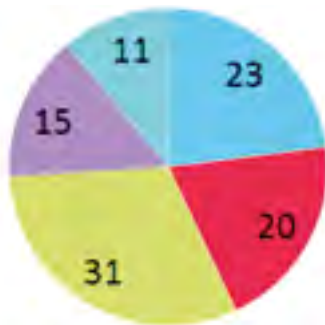


Рис. 4.33. Секторная диаграмма, которая может описывать (в процентах) вклад филиалов определенной фирмы в общую прибыль или содержание компонентов в смеси

✓ *Для построения диаграммы достаточно одного ряда значений.*

Диаграмма связана с данными листа, на основе которых она создана, и в случае изменения данных автоматически обновляется. Графические элементы диаграммы являются векторными изображениями, параметры которых либо зависят от значений, содержащихся в ячейках, либо устанавливаются в процессе создания. Кроме того, что числовые данные используются для построения графических составляющих диаграммы (определение высоты столбцов на столбиковой диаграмме и гистограмме, центрального угла на секторной диаграмме, координат точек  $x$  и  $y$  на точечной диаграмме и тому подобное), их можно использовать для вычислений.

Текстовые данные таблицы могут быть использованы для вывода на поле диаграммы названий рядов данных, названий осей и делений на них и названия диаграммы (см. рис. 4.31). Отметки на осях могут быть числами (числами считаются также дата и время суток, если для них в отдельности не определен текстовый формат) в случае, когда ось является осью значений (см. рис. 4.30). Если ось является осью категорий, то отметки на ней являются названиями категорий (см. рис. 4.31).

Обозначенные определенным образом полоса, область, точка, сектор или другой объект на диаграмме, который представляет одну точку данных или значение ячейки листа, называется **Маркером данных**. Связанные друг с другом маркеры данных на диаграмме отражают **Ряд данных**. Один маркер соответствует одному значению данных таблицы на листе. Каждый ряд данных на диаграмме имеет свой цвет или другой способ обозначения и представлен в легенде диаграммы.



✓ **Легенда** – область, в которой представлены цвета или другие способы обозначения (маркеры), соответствующие рядам данных или категориям на диаграмме.

✓ **Диаграммы всех типов, за исключением круговой, могут содержать несколько рядов данных.**

**Основные линии сетки** – линии, которые можно добавить к диаграмме для улучшения восприятия и оценки отображаемых данных. Линии сетки начинаются от делений на оси и пересекают область построения, обозначая основные интервалы на оси.

На диаграмме можно воспроизвести промежуточные линии, обозначающие интервалы в пределах основных интервалов.

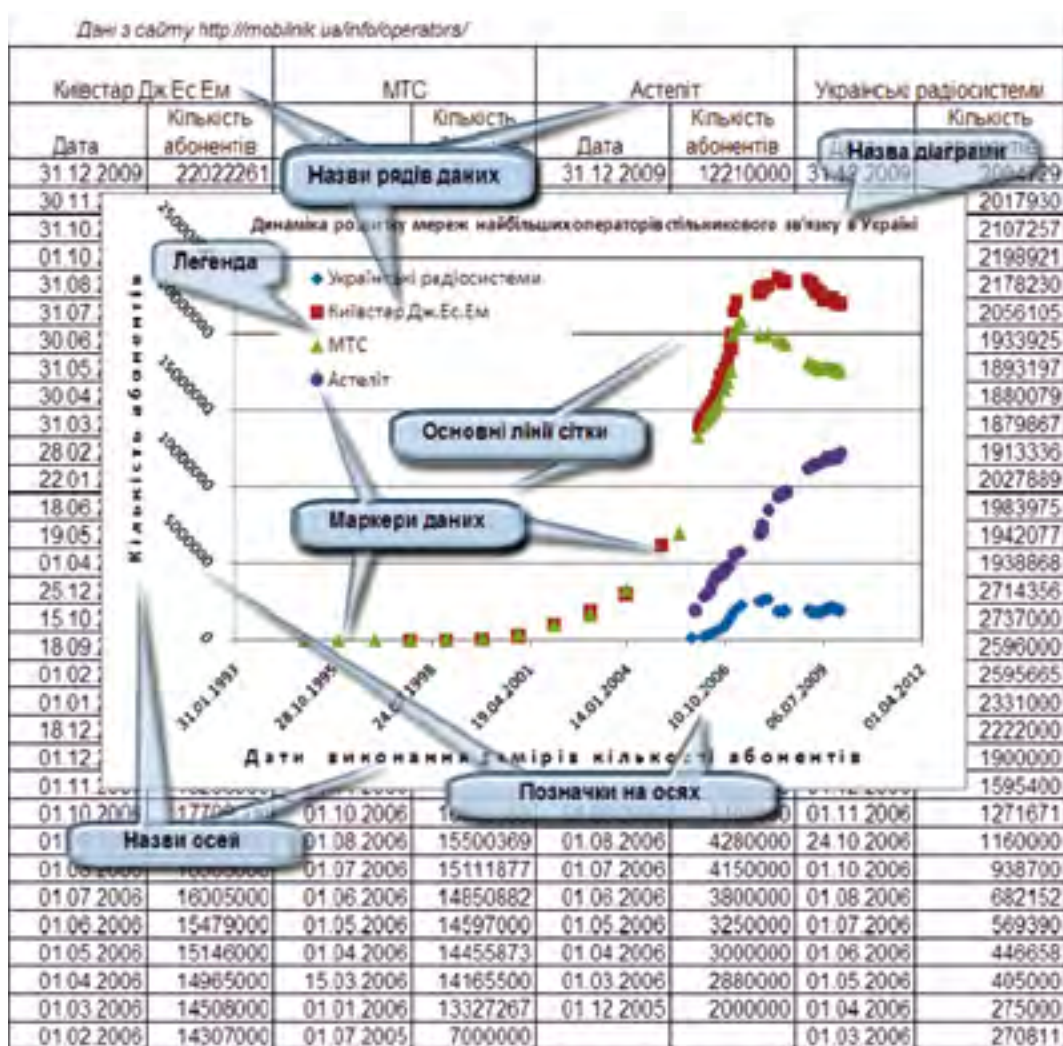


Рис. 4.34. Пример использования точечной диаграммы



В таблицах, построенных с использованием Microsoft Excel, как отметки оси категорий можно использовать заголовки столбцов и строк. Это могут быть названия месяцев, фирм, марки автомобилей, даже оценки, полученные учениками. Microsoft Excel также использует заголовки столбцов и строк как имена рядов данных.

Имена рядов данных и образцы их маркеров показаны в легенде.

В примере на рис. 4.34 названия рядов (названия операторов мобильных сетей) используются как имена рядов данных.

Прежде чем строить диаграмму, необходимо закончить все работы в таблице, включая форматирование. Затем выделить левой кнопкой мыши таблицу, в том числе заголовки строк и столбцов, для того чтобы указать позицию, где содержатся данные, на основе которых строится диаграмма.

Диаграмму можно создать на отдельном листе или расположить как внедренный объект на листе данных. Кроме этого, диаграмму можно опубликовать на веб-странице.

Чтобы создать диаграмму, необходимо сначала ввести для нее данные на лист. После этого, выделив эти данные, следует перейти к пошаговому созданию диаграммы, во время которого выбрать ее тип и различные параметры, или с помощью панели инструментов **Диаграммы** ленточного меню **Вставка** создать базовую диаграмму, которую впоследствии можно будет изменить.

Для создания диаграммы на отдельном листе его следует предварительно создать, используя команду **Добавить** из динамического меню, вызываемую нажатием правой кнопки мыши на закладках **Листы**.

Для просмотра нужной панели инструментов выберите в ленточном меню закладку **Вставка**. В группе **Диаграммы** находят ссылки на все необходимые команды и подменю (рис. 4.35).

Чтобы изменить текст в любом элементе диаграммы, щелкните этот элемент правой кнопкой мыши, а затем введите новый текст или измените имеющийся. Чтобы добавить графический элемент, нажмите кнопку **Добавить фигуру** на панели инструментов **Диаграмма**. К элементам диаграммы, отражающим числовые данные, для большинства типов диаграмм можно добавить числовые значения.

После завершения работы щелкните левой кнопкой мыши за пределами поля диаграммы.

✓ *Если задержать указатель на элементе диаграммы, появится подсказка с названием этого элемента.*

Например, если навести указатель на легенду диаграммы, появится подсказка, которая содержит слово *Легенда*. То есть элемент диаграммы, на который наведен указатель, становится доступным для редактирования. Добавление элементов на диаграмму и их редактирование следует

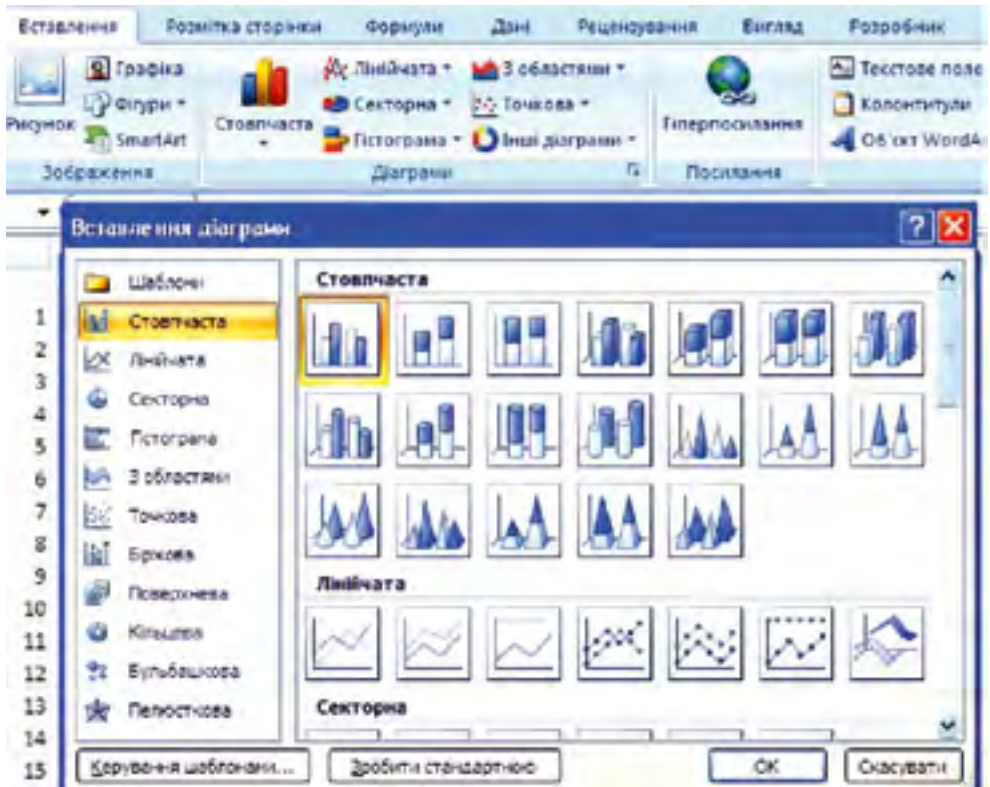


Рис. 4.35. Выбор типа диаграммы

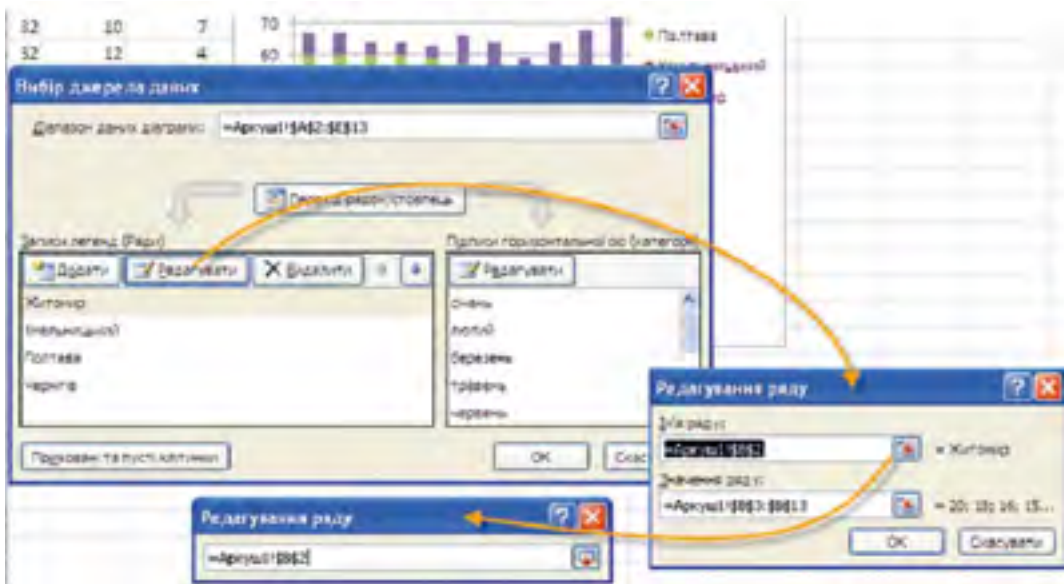


Рис. 4.36. Последовательность действий по выбору данных, отображаемых на диаграмме

выполнять с использованием панели инструментов **Макет** главного ленточного меню.

Редактирование диаграммы в целом (внесение изменений в надписи, изменение источников данных диаграммы, способов отображения графических элементов и т.д.) выполняют, приведя предварительно указатель на свободное поле диаграммы и нажав правую кнопку мыши. После этого можно выбрать в динамическом меню нужное указание, например **Выбор данных** (рис. 4.36).

### Проверяем себя

1. Какие средства используют для отображения данных на диаграмме? 🌟
2. Зачем нужна **Легенда**? В каких случаях без этого компонента диаграммы можно обойтись? 🌟
3. Чем отличается ось значений от оси категорий? ➕
4. Какой тип диаграммы лучше использовать для построения графика зависимости  $y = f(x)$ ? Почему? ★
5. В каких случаях целесообразно дополнять изображение на диаграмме значениями данных? Почему? ➕
6. Какие составляющие диаграммы являются векторными изображениями, а какие – текстом? Что в них общего? ➕
7. Как были созданы отметки горизонтальной оси на рис. 4.28 и 4.29? Почему? ➕
8. Предложите несколько задач, для которых можно использовать диаграммы, подобные показанным на рис. 4.33. ➕

### Выполняем

1. 🧑🧑 Постройте диаграмму, которая отображает результаты почасового измерения температуры воздуха в течение суток. Используйте ориентировочные данные для весенних суток, считая, что минимальная температура ночью составляла  $-8^{\circ}\text{C}$ , а максимальная днем –  $+5^{\circ}\text{C}$ . Нанесите на диаграмму названия осей. Какой тип диаграммы целесообразно использовать? Какой тип диаграммы будет единственно возможным, если измерения выполняют в произвольные моменты времени (не точно каждый час)? Как можно по построенной диаграмме определить температуру в промежутках между ее измерениями? Какой тип диаграммы для этого наиболее подходящий? ★
2. 🧑🧑 Чтобы получить бронзу, в нагревательную печь загрузили 10 кг меди, 1 кг цинка, 3 кг олова и 6 кг свинца. Постройте диаграмму, на которой каждому маркеру поставьте в соответствие название металла в сплаве и его содержание в процентах. Какой тип диаграммы целесообразно использовать? Какие данные достаточно внести в таблицу? 🌟



3.  По данным по валовому сбору зерна в Украине (табл. 4.3) постройте столбиковую диаграмму.

Таблица 4.3

Годы	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
Валовой сбор, млн т	41,8	38,4	34,3	29,3	50,2	45,4	39,3	56,7	46,2	62,9	44,45

Как получено значение 44,45 млн т в последнем столбце, если таблица была создана в 2013 году? Как его можно обозначить на диаграмме? 



4.  В производственном объединении «Вышиванка» четыре производства – в городах Хмельницкий, Ивано-Франковск, Косов, Тернополь. В течение пяти последних лет эти производства дают доход каждое на сумму, показанную в табл. 4.4. Постройте диаграмму, на которой можно было бы увидеть как общие достижения ПО «Вышиванка», так и вклад каждого производства.


Таблица 4.4

#### Прибыль производств объединения «Вышиванка», тыс. грн

Город	Прибыль производств по годам				
	2010	2011	2012	2013	2014
Хмельницкий	55,1	56,2	48,0	51,0	52,0
Ивано-Франковск	48,4	48,0	48,0	47,0	49,8
Косов	5,0	9,0	19,5	22,3	22,0
Тернополь	15,0	17,0	21,0	23,0	25,3
Общая прибыль	123,5	130,2	136,5	143,3	149,1

Как получены числа в нижней строке? Как их можно использовать на диаграмме? Какой тип диаграммы целесообразнее использовать для представления данных, по вашему мнению? Почему? 

## 4.6. Моделирование в электронных таблицах

 Необходимость в моделировании обычно возникает в том случае, если исследование самого объекта невозможно, тяжелое, дорогостоящее или требует много времени. Между моделью и объектом должно существовать отношение отображения, которое может заключаться в сходстве физических характеристик или функций, в тождестве математического описания и тому подобное.



Обработка числовых данных по алгоритму. Описание моделей средствами электронных таблиц.

Информационные технологии и компьютер – это помощники человека при составлении моделей. Использование компьютера для исследования моделей различных объектов позволяет изучить их изменения в зависимости от значения тех или иных параметров.

Для построения компьютерных моделей можно использовать прикладное программное обеспечение общего назначения (например, электронные таблицы) или системы программирования. В некоторых случаях целесообразно использовать специализированные программы (например, программы исследования функций, программы для создания ментальных карт), а также программные модели, которые размещены в Интернете.

Многие объекты и процессы можно описать математическими формулами, связывающими их параметры. Эти формулы и являются математической моделью оригинала. По ним можно выполнить расчеты с разными значениями параметров и получить количественные характеристики модели.

Поскольку табличный процессор обеспечивает обработку данных, представленных и как наборы (например, совокупность значений ячеек), можно достаточно просто и быстро выполнить сложные вычисления.

Табличный процессор является инструментом, с помощью которого можно не только выполнить расчеты количественных характеристик исследуемого объекта или процесса, но и представить их в форме диаграмм.

Для того чтобы некоторое действие считалось моделированием, необходимо определить цель моделирования. Цель моделирования (познание, исследование, проектирование) определяет, зачем проводится моделирование объекта.

Цель моделирования – чаще всего это поиск ответа на вопрос, заданный в формулировке задачи исследования. Выявляются факторы, от которых зависит поведение модели, осуществляется анализ объекта. Определяются внутренние связи простых объектов, входящих в состав моделируемого объекта.

- ✓ По результатам анализа объекта создается **информационная модель** (формируется совокупность сведений об объекте).
- ✓ **Математическая модель** создается как математические описания связей между свойствами объекта.
- ✓ **Компьютерная модель** непосредственно связана с прикладной программой, с помощью которой проводится моделирование, поэтому данные для ее создания подаются в форматах, которые могут быть обработаны табличным процессором, а математические описания – по правилам (синтаксисом), принятым для конкретного табличного процессора.

- ✓ **Компьютерный эксперимент** планируется и проводится в несколько этапов.

На первом этапе выполняется проверка компьютерной модели. Эта проверка заключается в представлении входных данных, результат обработки которых уже известен из наблюдений за объектом. Если в процессе проверки модели получены предполагаемые результаты, считается, что компьютерная модель достаточно правильно отображает объект.

На втором этапе в качестве входных используются данные, описывающие условия, в которых будет находиться реальный объект. Полученные расчетные данные передаются для анализа.

По полученным расчетным данным проверяется, насколько расчеты соответствуют представлению и целям моделирования, возможно ли использовать объект в тех условиях, для которых проводились расчеты на модели.

### Проверяем себя

1. Какие этапы моделирования в электронной таблице можно выделить? 🟢
2. Что означает тестирование модели? ➕
3. Моделью какого реального процесса может быть математическое выражение  $y = a \cdot x - b \cdot t$ , если  $a$  и  $b$  измеряются в километрах в час, а  $t$  – в часах? ➕
4. Что такое параметр объекта? ➕
5. Что такое модель объекта? 🟢
6. Для чего используются модели объектов? 🟢
7. Приведите примеры моделей, которыми вы пользовались при изучении физики, химии, биологии. Могут ли некоторые из них быть преобразованы в информационные? В математические? ★
8. Как, исследуя математическую модель, можно получить новые знания об объекте? Приведите примеры. ★
9. Как, исследуя компьютерную модель, можно получить новые знания об объекте? Приведите примеры. ★

### Выполняем

1. 🧑‍🤝‍🧑 Амеба каждые 3 часа делится надвое. Постройте модель роста численности амеб через 3,6,9,12, ... часов. Факторы, приводящие к гибели амеб, не учитываются. 🟢
2. 🎲 На первую клетку шахматной доски положили одно пшеничное зерно, на вторую – два, на третью – четыре и так далее. Назовите число зерен на 37-й клетке. 🟢



3. 🧑🏫 Формализуйте шуточную задачу из «задачника» Г. Остера: «С двух будок, находящихся на расстоянии 27 км друг от друга, навстречу друг другу выскочили в одно и то же время две драчливые собаки. Первая бежит со скоростью 4 км/ч, а вторая – 5 км/ч. Через сколько времени начнется драка?» Создайте математическую модель. Рассчитайте в таблице зависимость расстояния между собаками от времени, используя заполнение ячеек с различным шагом. ✦

4. Определение возможности построения треугольника. Составьте модель определения возможности построения треугольника по трем заданным сторонам А, В, С. Используйте формулы для сравнения значений. ★

5. 🏠🧑🏫 Исследуйте с использованием табличного процессора модель поверхности Земли, представленную на рис. 4.35. Для каждого из обозначенных маршрутов постройте диаграмму зависимости высоты подъема от горизонтального расстояния. Для этого приложите линейку к прямой, отражающей маршрут, запишите расстояние от начальной точки до пересечения с линией уровня и соответствующую высоту поднятия. Постройте диаграмму. ★

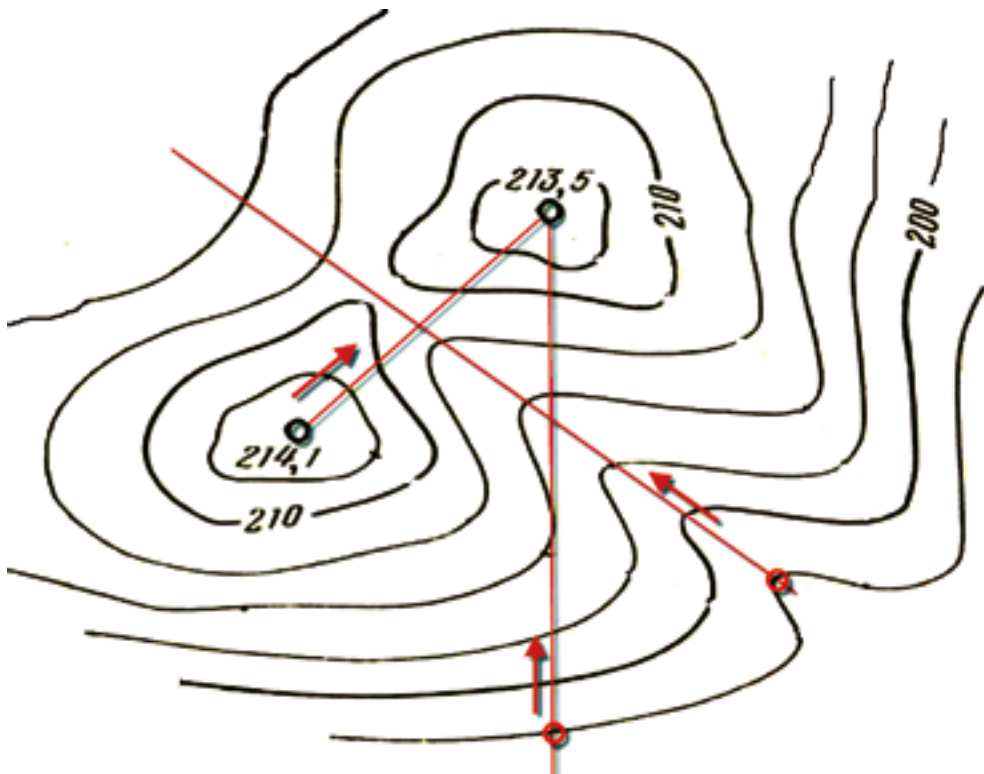


Рис. 4.35. Изображение двух холмов и долины между ними (модель поверхности Земли) в масштабе: в 1 сантиметре 200 метров, расстояние по высоте между линиями уровня 2,5 м

**Практическая  
работа № 6**

<b>Тема:</b>	Ввод, редактирование и форматирование данных в среде табличного процессора
<b>Цель:</b>	Приобрести практические навыки ввода, редактирования и форматирования данных в среде табличного процессора

**Задание 1**

Создать ЭТ для формирования счета-фактуры:

№ п/п	Наименование товара	Ед. изм.	Количество	Стоимость	НДС	Цена с НДС	Сумма
1	2	3	4	5	6	7	8
Всего:							

Столбец 1 должен содержать номер по порядку позиции товара и должен заполняться автоматически.

Столбцы 2–5 заполняются произвольно.

Столбцы 7–8 вычисляются средствами Microsoft Excel.

В столбцах 6 и 7 налог на добавленную стоимость (НДС) составляет 20 % от стоимости товара в столбце 2.

**Задание 2**

Участок производит железобетонные изделия А, Б, В, Г, Д, Е, Ж по разным рецептурам бетона. Создать электронную таблицу для вычисления необходимого на день количества компонентов, если известно дневное количество каждого изделия. Рецептuru смесей:

Название	Масса, т	Цемент 400	Цемент 500	Щебень	Отсев	Песок
		Массовое содержание компонентов в бетонной смеси, %				
А	10	25		30	30	15
Б	5	25				75
В	0,5		10	50		40
Г	7	20			30	50

Название	Масса, т	Цемент 400	Цемент 500	Щебень	Отсев	Песок
		Массовое содержание компонентов в бетонной смеси, %				
Д	1	30		30		40
Е	6		30	40		30
Ж	0,8		25		25	50

### Практическая работа № 7

<b>Тема:</b>	Вычисления над данными электронной таблицы. Использование встроенных функций
<b>Цель:</b>	Приобрести практические навыки выполнения вычислений в среде табличного процессора

### Задания

1. Создать ЭТ для вычисления значений выражения на промежутке  $[-10, 10]$  с шагом 0,5. Значения вычисляются таким образом:

$$(x^2 - 16)/(x - 4), \text{ если } x > 4;$$

$$x, \text{ если } x \leq 4.$$

2. Создать ЭТ для вычисления значений выражения на промежутке  $[-5, 10]$  с шагом 0,5. Значения вычисляются таким образом:

$$(2x^2 - 4x - 1)/2x, \text{ если } x < 0;$$

$$2x - 1, \text{ если } x > 0.$$

3. Создать таблицу для обработки результатов измерения плотности вещества методом обмера и взвешивания.

### Практическая работа № 8

<b>Тема:</b>	Создание диаграмм. Анализ данных, представленных на диаграмме
<b>Цель:</b>	Приобрести навыки работы с диаграммами

### Задание

Путем измерений напряжения, которое дают одинаковые микросхемы, предназначенные для питания устройства памяти компьютера, получили таблицу (см. с. 144).

Надо построить диаграмму, которая дает возможность увидеть, в каких пределах находятся значения напряжения для разных температур.

**Указание:** Для каждой температуры найти большее, меньшее и среднее значение. Построить столбиковую диаграмму разниц напряжений для каждой температуры.

T, °C	Номер экземпляра микросхемы				
	1	2	3	4	5
0	3,953	3,957	3,950	3,980	3,963
10	3,977	3,986	3,961	3,977	3,961
20	3,976	3,977	3,970	3,992	3,984
30	3,999	3,979	3,995	4,000	3,979
40	3,994	3,991	3,992	3,997	3,986
50	3,995	3,996	3,997	3,999	4,000
60	4,002	4,004	4,001	4,001	4,005
70	4,008	4,010	4,005	4,008	4,009
80	4,024	4,025	4,012	4,010	4,009
90	4,027	4,022	4,034	4,009	4,028
100	4,022	4,013	4,015	4,002	4,005
120	4,004	4,004	4,061	4,040	4,013
130	4,023	4,027	4,063	4,052	4,040



## Словарь

**Адрес ячейки** – способ обращения к данным в электронных таблицах.

**Аргумент** – порция данных (значения: переменной, константы, ячейки или группы ячеек, результата вычислений и т.п.), которая передается на обработку функции.

**Диаграмма** – изображение, используемое для иллюстрации числовых значений, зависимостей между ними.

**Компьютерная модель** – модель, которая создана с использованием компьютера, обычно представляющая собой сочетание информационной и математической моделей.

**Операнд** – порция данных (значения: переменной, константы, ячейки или группы ячеек, результата вычислений и т.п.), которая передается на обработку.

**Тип данных** – характеристика, которая явно или неявно присвоена объекту (переменной, функции, полю записи, константе, массиву и т.п.). Тип данных определяет множество допустимых значений, формат их хранения, размер выделенной памяти и набор операций, которые можно выполнять над ними.

**Формат представления данных** – способ, модель представления данных, в которых указаны их тип и способ воспроизведения для пользователя.

**Функция** – описанная определенным образом последовательность действий (с использованием определенных данных, которые называют аргументом, или без них), результатом выполнения которой являются данные.

**Электронные таблицы** – группа приложений, предназначенных для обработки данных, представленных в табличной форме.

**Ячейка (клетка) (в электронных таблицах)** – наименьшая порция данных в электронных таблицах, которая может быть использована как операнд функции, описанной средствами таблиц.



## РАЗДЕЛ 5. КОМПЕТЕНТНОСТНЫЕ ЗАДАЧИ



Искать сведения обо всем, что знает человечество, можно в сети Интернет. Вычислять все, что необходимо, обрабатывать данные – с использованием табличных процессоров и систем программирования, анализировать полученные результаты – с использованием редакторов карт знаний. Описывать полученные результаты – с использованием текстовых процессоров, графических редакторов, редакторов карт знаний. Представлять полученные результаты – с использованием презентационных систем, посылая свои выводы с использованием электронной почты, систем мгновенной передачи сообщений, обнародуя их в социальных сетях, на собственном сайте.



Этапы и средства решения компетентностных задач: содержательный анализ задачи; построение информационной модели; поиск информационных материалов; подбор основного средства обработки данных (редакторы: графические, презентаций, карт знаний; текстовый процессор, табличный процессор, учебная среда выполнения алгоритмов); обработка данных; представление результатов решения задачи.

### 5.1. Когда и как возникают компетентностные задачи

Существуют задачи, которые можно решить и выполнить, только сочетая знания и умения по нескольким предметам, которые вы изучали, привлекая собственный жизненный опыт. Такие задачи возникают в основном в жизненных ситуациях, когда хочется спросить: «Что мне делать, как мне действовать при определенных обстоятельствах для достижения определенной цели?». – А спросить не у кого. Эти задачи (как и большинство других задач, которые стоят перед взрослым человеком) требуют анализа условий, определения обстоятельств деятельности, характеристик объекта, для которого решается задача, построения модели.

Одним из самых простых примеров, когда необходимо пройти все указанные выше этапы, является задача планирования поездки в другой город, которую очень часто приходится решать.

Например, для того чтобы принять решение, ехать семье из четырех человек из Киева в Житомир **собственным автомобилем** или **рейсовым автобусом**, нужно, среди других данных, оценить стоимость поездки.

Анализируем две модели действий и сравниваем их эффективность. Основным критерием будет *стоимость поездки*.



Прежде всего нужно *узнать, сколько стоит билет на автобус* (выходим на сайт автостанции, находим стоимость билета – самый очевидный и самый простой шаг, который, однако, требует умений, необходимых для выполнения этих действий).

Затем следует вычислить стоимость поездки автомобилем. Для этого нужно оценить объем топлива, необходимого для того, чтобы конкретным автомобилем доехать из Киева в Житомир.

Для этого нужно выполнить следующие действия:

1. *Определить* (найти в справочнике, на карте) расстояние  $S$  между городами и задать время  $t$ , за которое хотим преодолеть это расстояние, скорость движения  $v_0$ , которую можно реально поддерживать (значение получим из Правил дорожного движения или спросим у опытного водителя).

2. *Разделить*  $S$  на  $t$ , то есть вычислить, какой должна быть скорость движения  $v$ .

3. *Сравнить значения  $v$  и  $v_0$* ; если полученное значение  $v$  превышает значение  $v_0$ , то выполнить действия пункта 2, присвоив  $t$  большее значение, **иначе** перейти к пункту 4.

4. *Определить* (найти в справочнике, в описании автомобиля), сколько горючего расходует автомобиль на 100 км при скорости движения  $v$ , разделить найденное число на 100, то есть определить значение  $q$  (сколько литров топлива будет расходоваться на один километр).

5. *Умножить  $q$  на  $S$*  и получить объем топлива, необходимого для путешествия, то есть промежуточный ответ.

6. *Умножить* полученный объем топлива на стоимость одного литра, получить стоимость топлива для поездки.

После этого необходимо сравнить стоимость четырех билетов на автобус со стоимостью топлива и принимать решение, учитывая и другие обстоятельства (комфортность поездки, наличие багажа и т.п.).

Другим примером задачи, которая требует знаний из многих отраслей, является задача планирования времени в течение недели. Эту задачу все привыкли решать, не слишком задумываясь. Или не решать вовсе, полагаясь на то, как распределение времени сложится стихийно.



*На самом деле, достаточно лишь раз попробовать рационально составить план своих действий, и вы увидите, что появляется много свободного времени, которое раньше уходило на ожидание, лишние поездки, просмотр ненужных телевизионных передач и тому подобное.*



*Итак, прежде чем принимать какое-либо решение, нужно для себя лично или для команды, с которой вы будете вести совместную деятельность, выполнить такие шаги:*

1. **Проанализировать задачу**, выделить и как можно точнее сформулировать цель деятельности, начальные условия, ресурсы, которые можно использовать.

2. Описать задачу таким образом, чтобы можно было планировать деятельность по ее решению, выделив один или несколько способов достижения результата, описать их словесно, построить **информационную модель** деятельности и объектов. Оценить важность всех и каждого из параметров этой модели, их влияние на результат, выделить важнейшие и пренебречь несущественными.

3. По возможности и потребности найти **математические описания** процессов и объектов, которые будут моделироваться при решении задачи.

4. Спланировать действия при решении задачи, а именно: поиск необходимых данных, **выполнение вычислений**, формулировка и **представление результатов**.

5. Определить необходимый для выполнения каждого этапа **инструментарий**, то есть средства ИТ и источники сведений.

6. Выполнить решение задачи, получить и проанализировать результаты. Представить их.

### Проверяем себя

1. Какие ИТ и программные средства можно использовать на этапе поиска данных для решения задачи? 🌱

2. Какие источники данных и сведений на бумажных носителях можно использовать на этапе поиска данных? 🌱

3. Зачем необходимо выполнять формулировку информационной модели объекта или процесса, рассматриваемых в задаче? ➕

4. 🏠 Каким образом можно выделить важнейшие параметры модели? Приведите примеры описания модели процесса подбора компьютера с целью использования для обучения в домашних условиях. ➕

5. Какие ИТ и программные средства можно использовать на этапе построения модели и планирования деятельности? 🌱

6. Что такое «математическая модель» объекта? Приведите пример. 🌱

7. Какие ИТ и программные средства можно использовать на этапе решения задачи? 🌱



8. 👤 Какие возможности может обеспечить редактор карт памяти и на каких этапах его можно использовать? Можно ли его заменить другим программным средством и каким? ➕


9. Для каких задач целесообразно строить алгоритм решения и использовать учебную систему программирования? ➕


10. Для каких задач достаточно использовать поисковые системы и редактор презентаций? ✦

11. Какие возможности может обеспечивать табличный процессор и на каких этапах его целесообразно использовать? ✦


### Выполняем

1.   Проанализируйте задачу о поездке из Киева в Житомир, представленную в качестве примера. **Ключевые вопросы:** Какие программные средства, по вашему мнению, целесообразно использовать на каждом из этапов решения этой задачи? ✦

2.  Постройте графическое представление алгоритма решения задачи о поездке. ★

3.  Выполните поиск в Интернете по двум словам: *пшеница* и *ремесло*. **Ключевые вопросы:** Какое практическое применение могут иметь результаты поиска? При изучении какого предмета можно использовать эти результаты и в какой форме? Какие программные средства для этого потребуются? ✦

4. В табличном процессоре смоделируйте процесс наполнения бассейна водой, если через одну трубу вливается 10 литров в минуту, а через другую – выливается 3 литра в минуту. Частью какой задачи может быть такая модель? ✦

5.  Спланируйте выполнение проекта: «Безопасность на ежедневном маршруте». **Цель:** Понять, как можно решить проблему безопасности пешеходов (как и кто это может сделать – знаки, разметка, правила, их соблюдение). **Ключевые вопросы:** Какие основные источники опасности для пешехода на улице? Всегда ли кратчайший путь самый безопасный? Кто и что должен делать для того, чтобы путь пешехода был безопасным? ★

## 5.2. Компьютерные сети и Интернет

Если для некоторых задач поиск сведений в Интернете является лишь первым шагом к формулированию информационной модели, то для очень большого количества практически значимых задач этот поиск и является процессом решения задачи.

Для того чтобы поиск дал нужные результаты, следует стараться не пропускать шаги, рассмотренные в предыдущем разделе. Это позволит избежать ошибок и обеспечит надежность полученных результатов.

В частности, если предполагается создание проекта о каком-то техническом изобретении, следует прежде всего создать информационную модель электронного (или бумажного) документа, который предполагается получить в результате поиска.

Пусть таким документом будет компьютерная презентация. Опишем возможные варианты модели презентации, построенной на результатах поиска сведений в Интернете.

Прежде всего необходимо сформулировать требования к материалам, которые необходимо искать, их качеству и количеству.

Количество материалов, которые должны быть включены в презентацию, определяется, исходя из назначения презентации. Если презентацию планируется использовать для сопровождения доклада продолжительностью 10–15 минут, то и количество слайдов не должно превышать 10–15. Если же презентация предназначена для самостоятельного просмотра пользователем или размещения в сети, то количество слайдов определяется, исходя из соображений возможности и целесообразности определенной глубины раскрытия проблемы.

Например, для краткого рассказа о родном городе достаточно презентации из 10 слайдов, на которых можно подать картосхему города, созданные с использованием табличного процессора диаграммы, которые наглядно характеризуют промышленность города, структуру занятости населения и тому подобное. Мультимедийная составляющая этой презентации не может быть длительной – не более двух минут.

Обширная справка даже о небольшом городке, его истории, видах и достопримечательностях может содержать 100 страниц и более, содержать музыкальные произведения, видеофрагменты и тому подобное.

Требования к качеству и форме представления материалов также определяются назначением презентации. Презентации, предназначенные для сопровождения докладов, должны содержать минимально возможное количество текста, а материалы, предназначенные для размещения в сети, могут быть более широкими, содержать такое количество текста, чтобы можно было полностью представить все нужные сведения.

В то же время использование материалов из сети должно быть осуществлено с соблюдением следующих правил:

1. Материал должен быть максимально достоверным. Поэтому, просматривая найденные источники, останавливаемся на тех, которые можно считать наиболее достоверными. Это обычно должны быть официальные источники – сайты государственных органов, научные статьи (рис. 5.1).

2. Всегда, когда используется определенный источник сведений, следует подать также ссылки на него. Это очень просто сделать, наведя курсор на адресную строку и нажав правую кнопку мыши. После копирования адреса в буфер ее можно вставить как в текстовый документ, так и в адресную строку другого браузера.

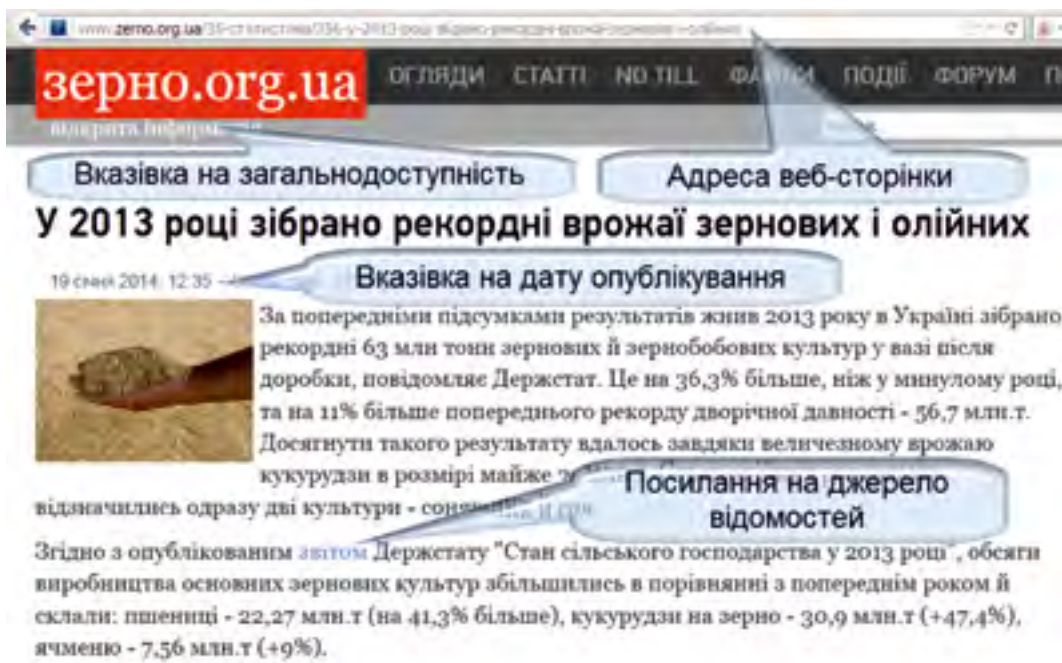


Рис. 5.1. Страница сайта (<http://www.zerno.org.ua>), сведения из которой имеют все необходимые признаки достоверности

## Проверяем себя

1. Какие признаки достоверности должны быть у источника сведений, которые можно использовать? 🌱
2. В чем заключаются правила соблюдения авторских прав при использовании сведений, полученных из сети Интернет, и каким образом это следует реализовывать? 🌱
3. Зачем необходимо выполнять формулировки информационной модели объекта или процесса, рассматриваемых в задаче? ✦
4. 🏠 Каким образом можно выделить важнейшие параметры модели? Приведите примеры для описания модели процесса отбора компьютера для использования в домашних условиях с целью обучения. ✦
5. Какие ИТ и программные средства можно использовать на этапе построения модели и планирования деятельности? 🌱
6. Что такое «математическая модель» объекта? Приведите пример. 🌱
7. 🏠 Пусть объектом моделирования является бассейн, а процессом – его наполнение. Бассейн имеет две трубы. Через одну вода наполняет бассейн, а через другую – вытекает. Постройте математическую модель наполнения бассейна. ★

## Выполняем

Нас окружают устройства, которых 20–30 лет назад не существовало. Они появились благодаря развитию информационных технологий, сокращенно – ИТ (рис. 5.2).



Рис. 5.2. Основные этапы развития мобильных устройств связи

Найдите в сети Интернет характеристики первых мобильных устройств связи и создайте электронную таблицу их характеристик.



### 5.3. Компетентностные задачи по основам алгоритмизации и программирования

#### Выполняем

#### Задача № 1.

Дана последовательность 12 целых чисел. Первое число вводится при помощи клавиатуры. Каждое последующее число больше предыдущего на 3. Разработать алгоритм и программу вычисления суммы первых шести чисел и суммы последних шести чисел. 🌱

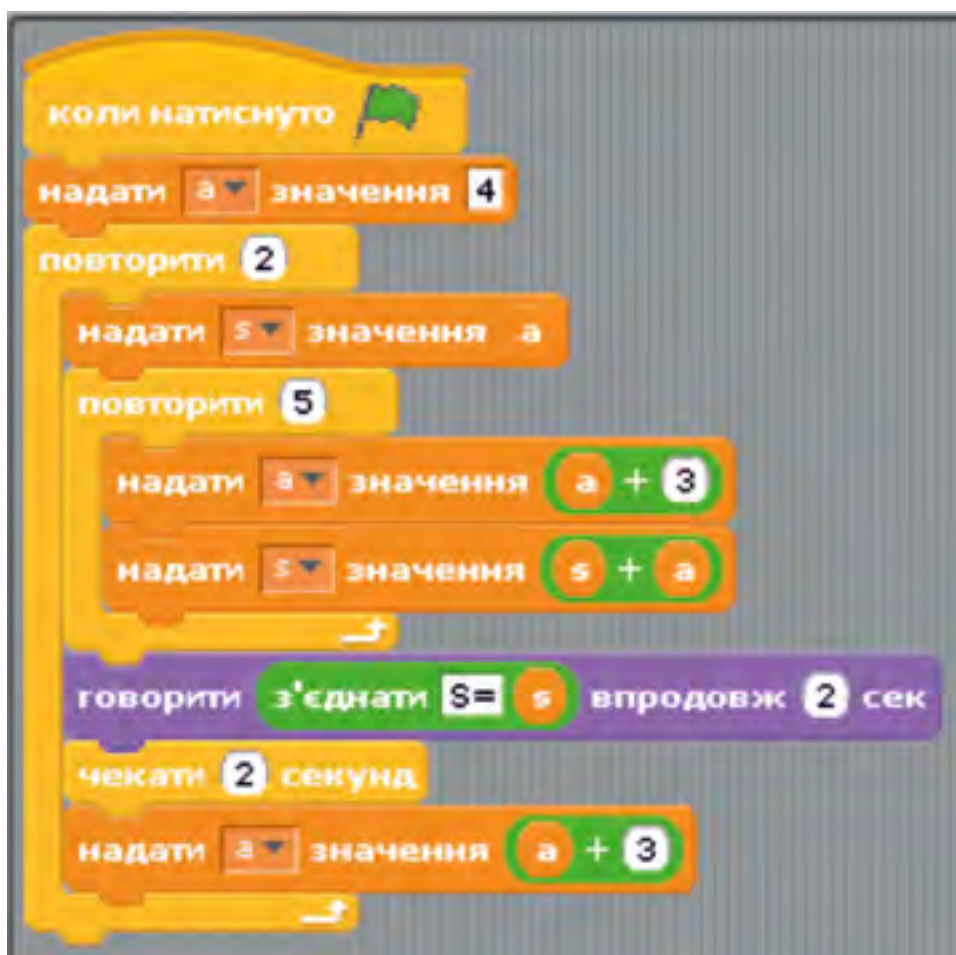


Рис. 5.3. Программа вычисления суммы последовательности чисел

Рассмотрим порядок исчисления сумм чисел на примере. Пусть задана последовательность таких чисел: 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37. Сумму первых шести чисел можно вычислить по сле-

дующей схеме:  $4 + 7 = 11$ ,  $11 + 10 = 21$ ,  $21 + 13 = 34$ ,  $34 + 16 = 50$ ,  $50 + 19 = 69$ . Сумму последних шести чисел можно вычислить таким образом:  $22 + 25 = 47$ ,  $47 + 28 = 75$ ,  $75 + 31 = 106$ ,  $106 + 34 = 140$ ,  $140 + 37 = 177$ . Обозначим текущее значение числа, которое будет добавляться, переменной  $a$ , текущее значение суммы чисел – переменной  $s$ . На основе рассмотренной схемы сложения чисел алгоритм вычисления суммы чисел каждой половины последовательности можно записать таким образом:

1. Предоставить переменной  $a$  значение 4.
2. Предоставить переменной  $s$  значение переменной  $a$ .
3.  $a := a + 3$ .
4.  $s := s + a$ .
5. Повторить п. 3 и п. 4 пять раз.
6. Объявить значение суммы половины чисел.
7.  $a := a + 3$ .
8. Повторить п. 2–п. 7 дважды.
9. Конец.

Программа, реализующая разработанный алгоритм, представлена на рис. 5.3.

Выполните программу.

Убедитесь, что сначала объявляется значение суммы первых шести чисел, затем – суммы последних шести чисел.

Выполните программу для различных значений первого числа, разницы между числами, а также различной длины последовательности и докажите, что программа работает правильно.

## Задача № 2. Кто лучше знает математику – Вася или Коля?

Ученик седьмого класса Вася хорошо знал математику. Он мог также отгадывать числа, которые задумывали другие ученики. Но ученик Коля внимательно проанализировал операции, которые выполнял Вася над числами, и не только распознал «тайну» Васи, но и разработал программный проект, который моделировал операции, выполняемые Васей. Программы этого проекта представлены на рис. 5.4, 5.5. На рис. 5.4 представлены программы для объекта с именем Коля, а на рис. 5.5 – для объекта с именем Вася.

Из приведенного проекта видно, что Коля просил задумать любое четное число, умножить его на 3, разделить на 2 и умножить на 2. Затем нужно сказать полученный результат. После этого Коля называет задуманное число. Итак, Коля фактически угадал, что для определения задуманного числа необходимо результат разделить на 3. ★

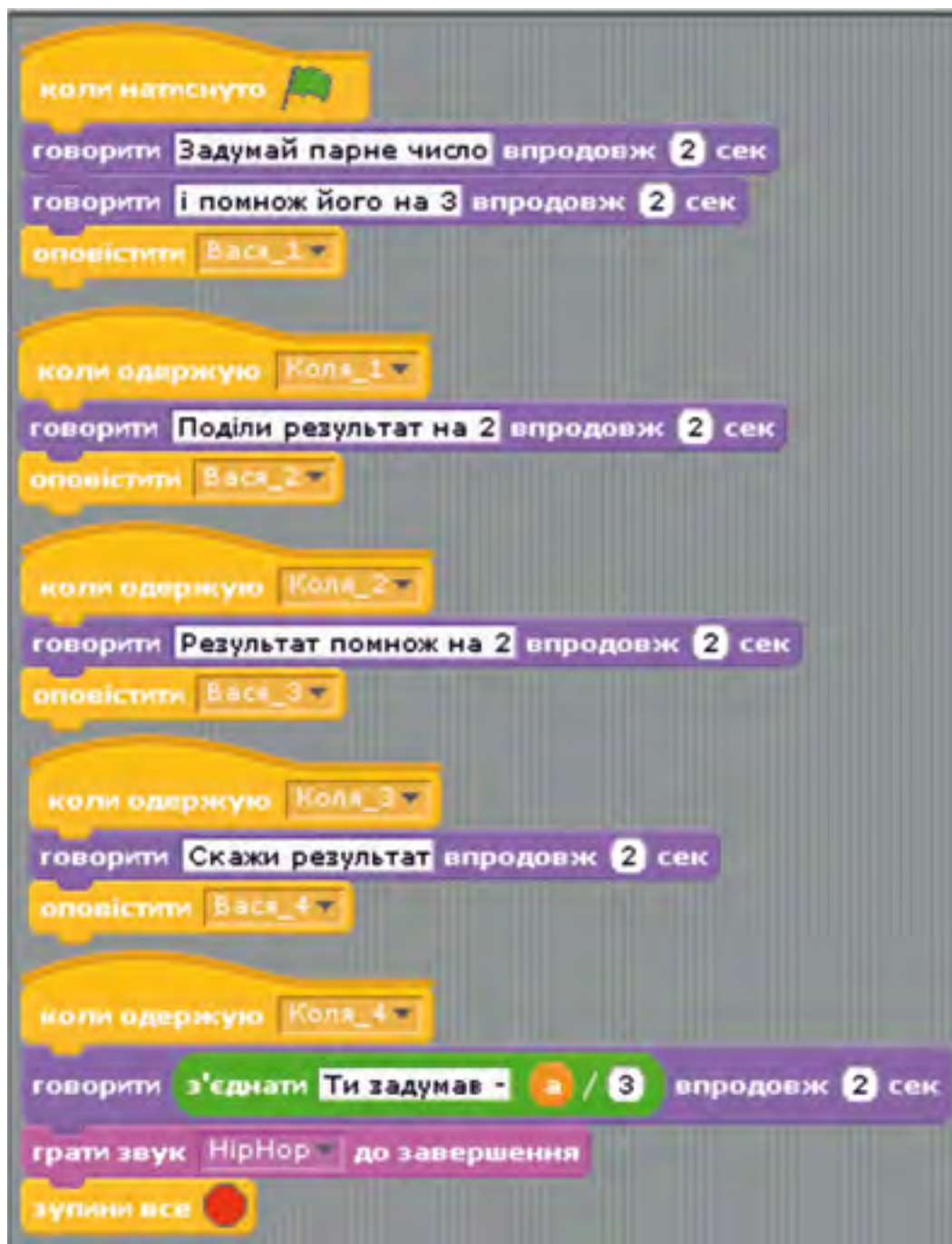


Рис. 5.4. Программы для объекта с именем Коля

Вася выполнил этот проект и убедился, что для приведенных исходных данных он действительно выдает правильный результат. В то же

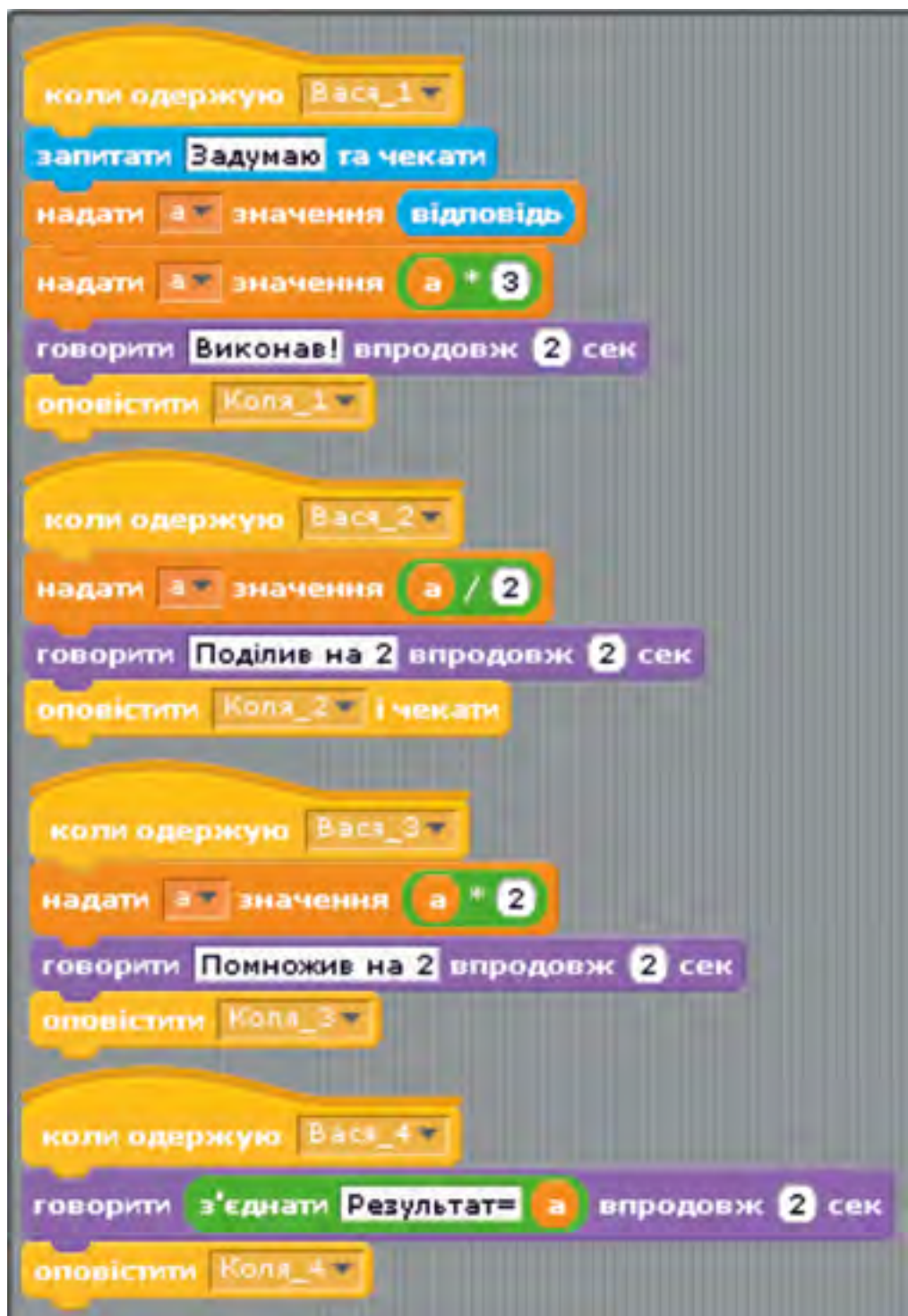


Рис. 5.5. Программы для объекта с именем Вася

время Вася сообщил, что проект разработан некорректно, так как не для всех исходных данных он выдает правильный результат. Попробуйте найти ошибку в проекте и ответить на вопрос: «Кто лучше знает математику – Вася или Коля?»

### Задача № 3.

На рис. 5.6 изображен кабинет физики (фон room1, папка Indoors), в котором Ира (объект girl4-standing, папка People) проверяет знания Насти (объект girl5, папка People) по физике. Ира предлагает Насте ответить на следующие три вопроса: «Раздел, в котором изучается движение тел без учета сил», «Название раздела об изучении движения тел под действием сил», «Раздел, в котором изучается равновесие тел». ★

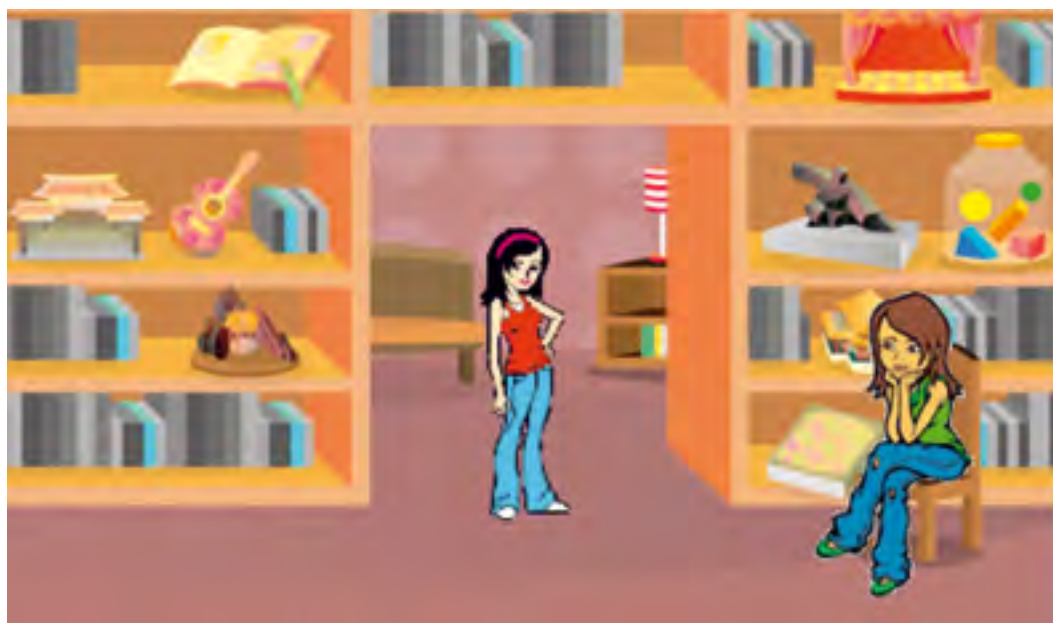


Рис. 5.6. Беседа в кабинете физики

После ответа на каждый вопрос Ира сообщает, правильный он или нет. Получив ответы на все три вопроса, Ира также сообщает Насте общее количество правильных ответов. Совокупность программ для Иры представлена на рис. 5.7, 5.8, а для Насти – на рис. 5.9.

Проанализируйте и выполните проект. Убедитесь, что он работает правильно. Попробуйте адаптировать проект для другого учебного предмета, например, для истории.



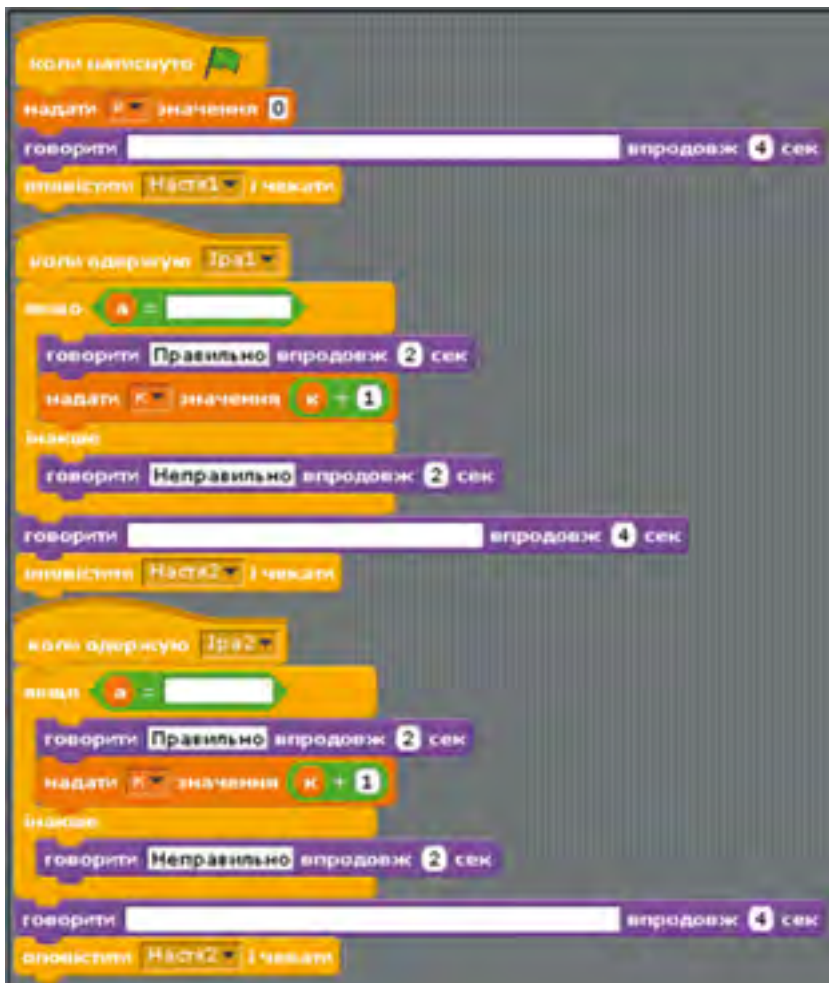


Рис. 5.7. Програма 1 для об'єкта с именем Ира

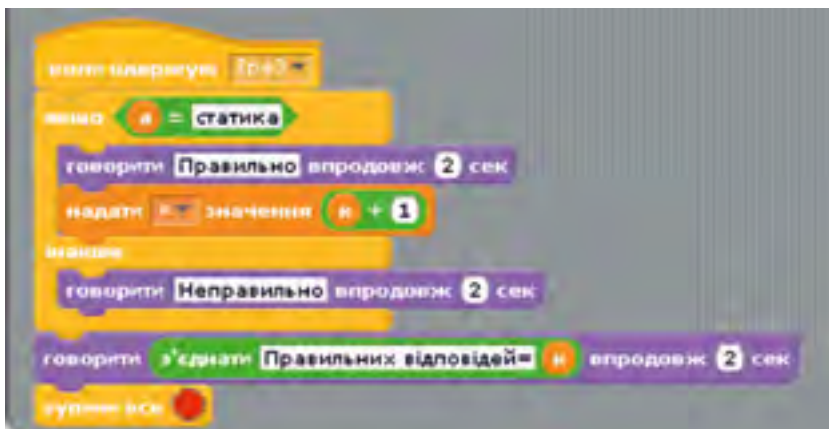


Рис. 5.8. Програма 2 для об'єкта с именем Ира





Рис. 5.9. Программы для объекта с именем Настя

#### Задача № 4.

У Вадима, ученика седьмого класса, возникла идея разработать в среде Скретч универсальную программу для проверки знаний основных терминов по любому учебному предмету. Суть его идеи заключалась в том, что на экране должно освещаться неполное название термина, в котором не хватает двух букв. Необходимо ввести нужные буквы. Если введенные буквы действительно принадлежат этому термину, пользователю программы сообщается о правильности ответа, иначе – о неправильности, а также показывается правильное название термина. Количество терминов может быть произвольным. После ввода букв для последнего термина сообщается о количестве правильных ответов и звучит определенная музыка.

Свою идею Вадим реализовал на примере названий 10 рек Украины. Сначала он создал в среде Скретч список с названием **Реки Украины**: Дніпро, Дністер, Ірпінь, Рось, Горинь, Сейм, Тетерів, Десна, Хорол, Уборть. Затем – список с названием **Неполные названия**: Дн\_\_ро, Дн\_\_тер, Ір\_\_нь, Р\_\_ь, Гор\_\_ь, С\_\_м, Те\_\_рів, Де\_\_а, Хо\_\_л, Уб\_\_ть и список с названием **Две буквы**: іп, іс, пі, ос, ін, ей, те, сн, ро, ор.

Программа проверки знаний названий рек Украины представлена на рис. 5.10. В программе использованы следующие переменные:  $k$  – текущий номер в списке реки, название которой анализируется в данный момент;  $c$  – количество правильных ответов;  $a$  – текущее значение введенных букв. Выполните программу. Докажите, что она функционирует правильно. Внесите в программу такие изменения, чтобы она проверяла знания терминов из любимого вами предмета. ✦



Рис. 5.10. Программа проверки знаний названий рек Украины

### Задача № 5.

На пристани расположены 10 контейнеров с номерами от 1 до 10, которые необходимо загрузить на пароход. Габаритные размеры всех контейнеров одинаковы, а масса каждого из них может достигать от 4 до 50 тонн. Для безопасной транспортировки контейнеров морем существуют определенные правила их загрузки и расположения на пароходе. Поэтому перед загрузкой контейнеров необходимо знать, какой из них имеет наибольшую массу. На рис. 5.11 представлена программа поиска контейнера с наибольшей массой среди 10. ★

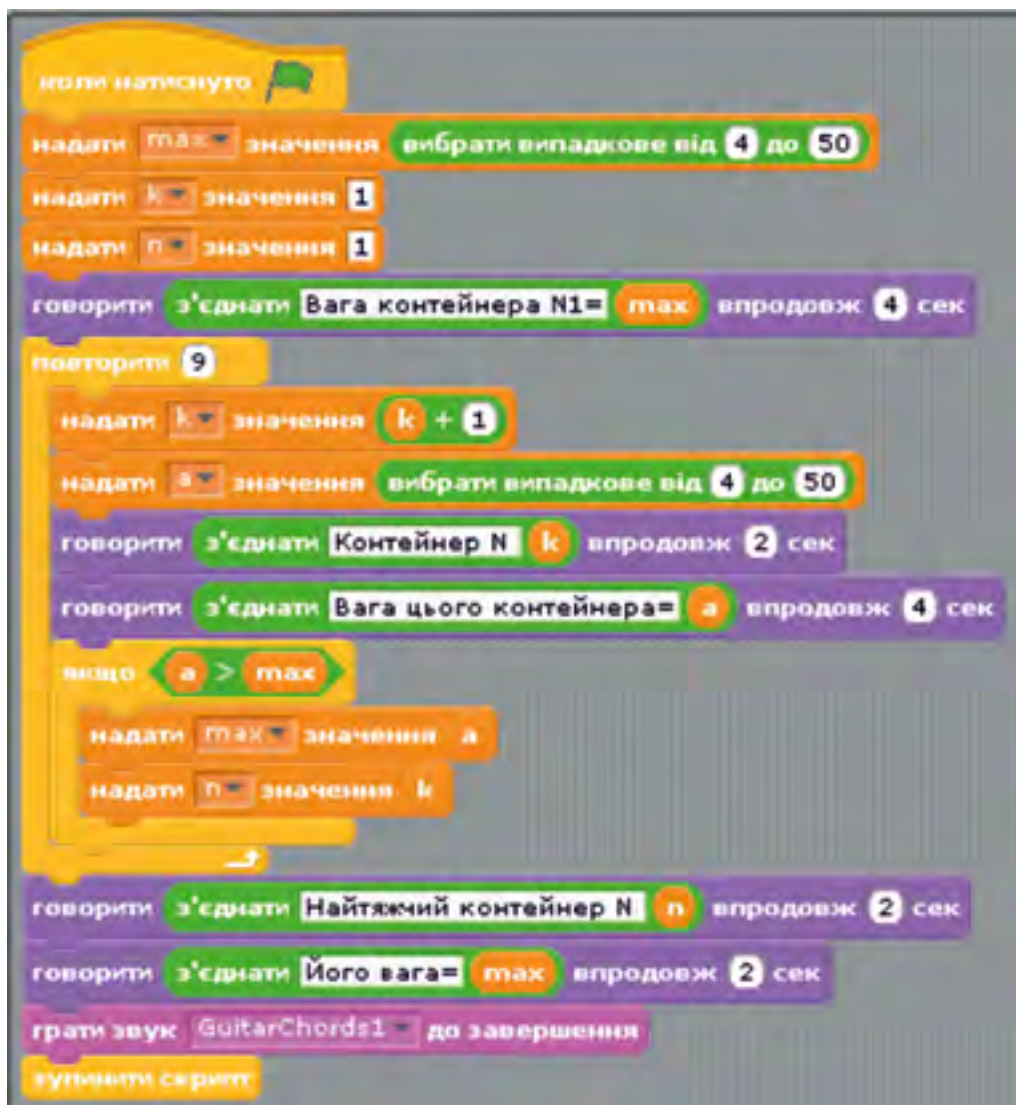


Рис. 5.11. Программа поиска контейнера с наибольшей массой

Решите задачу также и в электронной таблице, сделайте выводы.

В программе использованы следующие переменные: *max* – наибольшая масса контейнера из числа проверенных; *k* – номер текущего контейнера; *n* – номер контейнера с наибольшей массой из числа проверенных; *a* – масса текущего контейнера. Считается, что масса любого контейнера может иметь случайное значение от 4 до 50 тонн. Масса контейнеров определяется в последовательности их номеров.

Сущность выполнения программы объясняется на следующем примере. Пусть 10 контейнеров (в порядке их номеров) имеют такую массу: 20, 12, 7, 48, 10, 33, 35, 17, 40, 9. Для таких значений случайных чисел выполнение программы иллюстрируется данными, отображенными в табл. 5.1.

Таблица 5.1

Начальные значения	Циклы									Результат
	1	2	3	4	5	6	7	8	9	
<i>k</i> =1	<i>k</i> =2	<i>k</i> =3	<i>k</i> =4	<i>k</i> =5	<i>k</i> =6	<i>k</i> =7	<i>k</i> =8	<i>k</i> =9	<i>k</i> =10	<i>max</i> =48 <i>n</i> =4
<i>n</i> =1	<i>a</i> =12	<i>a</i> =7	<i>a</i> =48	<i>a</i> =10	<i>a</i> =33	<i>a</i> =35	<i>a</i> =17	<i>a</i> =40	<i>a</i> =9	
<i>max</i> =20	12>20 нет	7>20 нет	48>20 да	10>48 нет	33>48 нет	35>48 нет	17>48 нет	40>48 нет	9>48 нет	
			<i>max</i> =48							
			<i>n</i> =4							

Проанализируйте и выполните программу. Докажите, что она функционирует правильно. Зафиксируйте случайные числа, генерируемые в процессе выполнения программы. Разработайте таблицу, в которой отображаются изменения значений переменных для зафиксированных случайных чисел.

Сгенерируйте в табличном процессоре новые массы контейнеров с использованием функции = ROUND (RANDBETWEEN (4, 50); 0), скопированной в 10 ячеек. Введите новые значения и повторите выполнение.



## РАЗДЕЛ 6. ЗАДАЧИ ДЛЯ УЧЕБНЫХ ПРОЕКТОВ

### 6.1. Применение табличного процессора для выполнения проектов

**Проект № 1.** Создайте таблицу с названием «Самые известные разработки украинского авиационного конструкторского бюро «Антонов».

1. Переименуйте Лист 1 в Самолеты.
2. На этом листе создайте таблицу и заполните ее данными с сайта <http://www.antonov.com/aircraft>.
3. Отформатируйте заголовки столбцов и строк таблицы.
4. Оставьте пустым столбец «Изображения» (их будем вносить потом).
5. Используя команду **Примечание** меню **Вставка**, сделайте примечания для каждого названия самолета. В примечании укажите главные сведения о самолете (например, АН-124 «Руслан» – тяжелый транспортный самолет).
6. Для ячеек, в которых содержатся числа, установите формат **Общий** во вкладке **Число** меню **Формат ячейки**.
7. Закрепите области заголовков таблицы. Для этого щелкните на заголовке столбца А. Он весь выделится. Нажмите клавишу Ctrl и, не отпуская ее, щелкните на заголовке столбца В и заголовках строк 1, 2, 3. Они также выделятся (нажатая клавиша Ctrl обеспечивает выделение несмежных ячеек, столбцов или строк). Затем выполните команду **Окно / Закрепить области**. После этой команды, куда бы не перемещался курсор, выделенный столбец и строки всегда будут оставаться на экране.
8. Сохраните файл в собственной папке под названием **Airplane.xls**.
9. Откройте таблицу, дополните ее изображениями самолетов.
10. Проверьте, удобно ли пользоваться таблицей.

### 6.2. Создание проектов в среде Скретч



*Проектом в среде Скретч обычно называют одну сложную программу или несколько программ, реализующих единственную задачу.*

В процессе разработки проекта желательно соблюдать следующие правила:

- определить цель создания проекта и его конечный результат;
- выбрать объекты и фон сцены или разработать их с помощью графического редактора;



- разработать алгоритм реализации проекта;
- разработать соответствующую программу (программы) в среде Скретч;
- наладить проект;
- сохранить проект;
- по возможности разместить проект на Скретч-сайте.

Если проект разрабатывается группой учеников, то необходимо распределить обязанности между всеми членами группы. Распределение обязанностей может осуществляться по разным принципам. Например, по видам деятельности: художник, программист, сценарист и тому подобное; по перечню программ: один ученик разрабатывает первую программу, другой – вторую и т. д.

Одним из самых сложных и очень важным этапом является налаживание проекта, в процессе которого отыскиваются и исправляются ошибки.



*Ошибки в программах делятся на синтаксические и логические.*



*Синтаксические ошибки возникают в результате неправильной записи команд, операторов или недопустимой структуры программы.*

Скретч – это среда программирования, в которой не могут возникать синтаксические ошибки. Команды в Скретч используются в виде готовых блоков. Пользователь не может соединить между собой те блоки, соединение которых правилам Скретч не предусмотрено.



*Логические ошибки – это ошибки, возникающие в результате неправильных действий разработчика. Они могут возникать как на этапе разработки алгоритма, так и на этапе разработки программ проекта.*



*Логические ошибки в среде Скретч средствами среды не отслеживаются и не обнаруживаются.*

Ошибки этого типа должен выявить и исправить сам разработчик. Например, если вместо выражения  $a \cdot c$  в программе записано  $a + c$ , это является логической ошибкой.

Логической ошибкой будет и неправильная запись условий выполнения инструкций в команде разветвления.

Если в программе записано количество повторений тела цикла 3, а его инструкции необходимо повторить 5 раз, то это также является логической ошибкой. Логической ошибкой является и появление на сцене в определенный момент времени вместо одного объекта другого.



В среде Скретч для поиска логических ошибок и налаживания проектов можно применять такие методы:

**1. Пошаговое выполнение программ.** Пошаговое выполнение программ – это последовательное выполнение пользователем команд одна за другой и наблюдение за результатом их выполнения. Для пошагового выполнения программ применяются команды **Запустить один шаг** и **Задать одиночный шаг**, содержащиеся в меню редактирования и отладки программы пункта **Редактировать** основного меню (рис. 6.1).

**2. Приостановление выполнения программ.** Приостановить выполнение программ на время можно с помощью блока команды **Ожидать**. За это время можно проанализировать результаты, которые освещаются на экране.

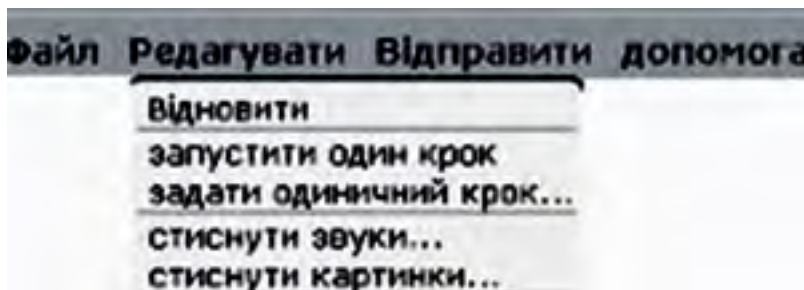


Рис. 6.1. Меню редактирования и отладки программы

**3. Использование контрольных точек.** Для облегчения процесса поиска ошибок в некоторых местах программы целесообразно выводить значения промежуточных переменных. Например, если необходимо вычислить значение выражения  $(a/b + c + a \cdot c) \cdot 5 - 3 \cdot (a \cdot b - 2c + 4)$ , можно отдельно вычислить значение выражения  $(a/b + c + a \cdot c) \cdot 5$  и также отдельно – значение  $3 \cdot (a \cdot b - 2c + 4)$  и вывести эти значения на экран.

**4. Использование комментариев.** Комментарий – это объяснение отдельных частей программы, а также выполняемых действий. Он помогает облегчить понимание программы. Комментарий особенно полезен для крупных проектов. Он может быть указан в любом месте программы и не влияет на результат ее выполнения.

**Проект № 2.** Разработайте и сохраните проект с именем **Экзаменатор**, с помощью которого проверяются знания о больших городах мира. В списке хранятся 10 крупнейших городов мира, расположенных в порядке уменьшения количества жителей. Необходимо четыре раза подряд вывести на экран вопрос: **Назовите город, который по численности населения занимает место № ....** Номер города генерируется как случайное число. После ответа на четвертый вопрос на экран следует

вывести количество правильных ответов, а также некоторые пожелания. Например, если ответы были правильными на все вопросы, на экран выводится сообщение: «Правильно». Если даны 2 или 3 правильных ответа – «Недостаточно», иначе – «Нужно изучить».

**Рекомендации.** В помощь в разработке проекта на рис. 6.2 представлена программа, в которой в случайном порядке предлагается назвать три страны из списка семи крупнейших по площади стран мира. В зависимости от результата ответов ученику выводится соответствующее сообщение. В программе использованы:  $k$  – переменная, в которой хранится количество правильных ответов,  $a$  – переменная, в которой

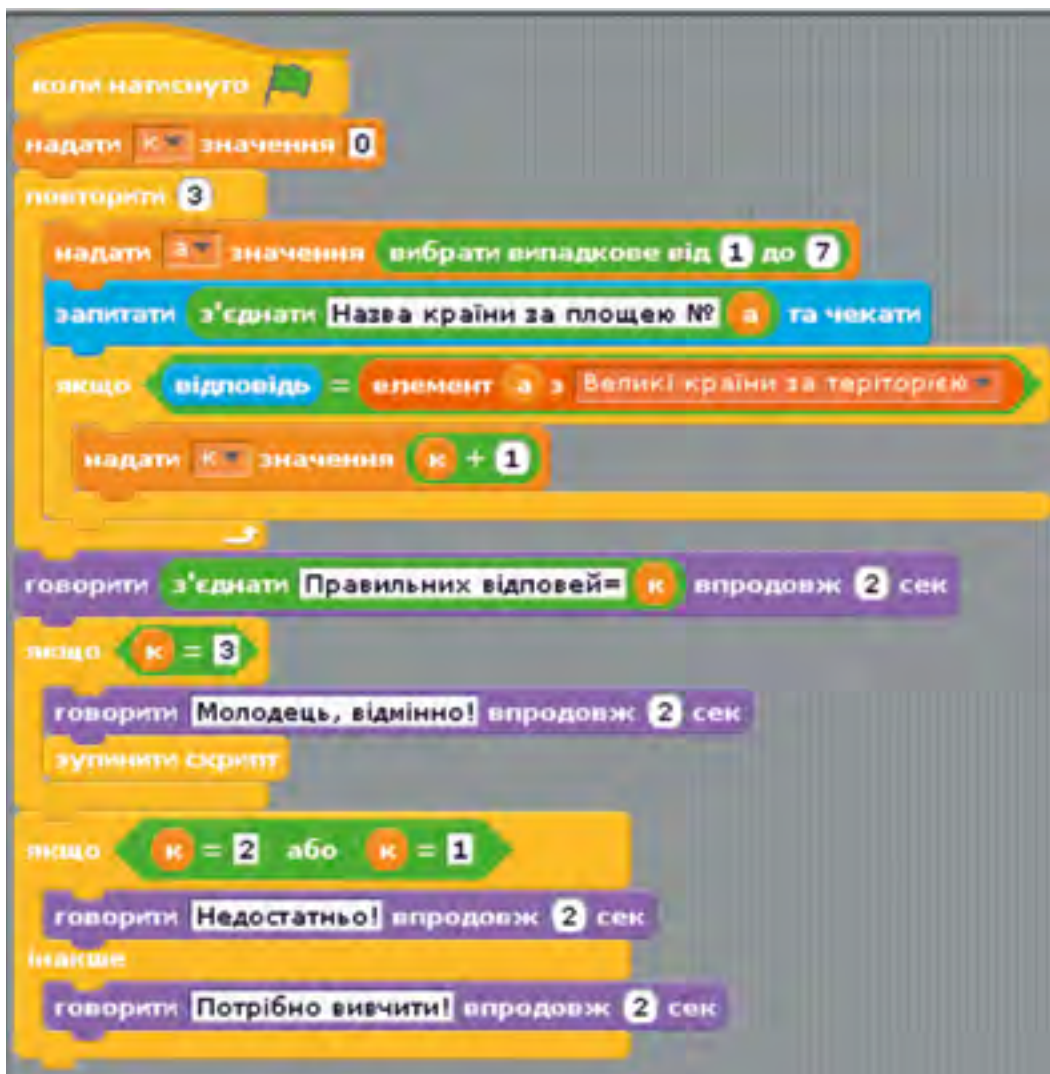


Рис. 6.2. Программа проверки знаний учащихся по географии

хранится текущее случайное число. *Большие по территории страны* – список, содержащий семь крупнейших стран мира.



**Проект № 3.** В соревнованиях учащихся седьмого класса по шахматам перед последними двумя играми сложилось положение, описанное в табл. 6.1.

Таблица 6.1

№ п\п	Имя	1	2	3	4	5	6	Текущее количество очков
1	Коля	⊙	0	1	1/2		1	2,5
2	Толя	1	⊙	0		1	1/2	2,5
3	Витя	0	1	⊙	1/2	1/2	1	3,0
4	Саша	1/2		1/2	⊙	1	1	3,0
5	Игорь		0	1/2	0	⊙	1	1,5
6	Вася	0	1/2	0	0	0	⊙	0,5

Как видим из табл. 6.1, в последних играх соревнуются между собой Коля с Игорем и Толя с Сашей, а Витя и Вася завершили соревнования. Нужно разработать проект, с помощью которого определяется, может ли Саша стать единоличным победителем турнира. Анализ таблицы позволяет сделать вывод, что Саша может стать единоличным победителем в случае его победы над Толей. В этом случае у него будет 4 очка, которые набрать больше никто не в состоянии. Кроме этого, он может закончить игру вничью, набрав 3,5 очка. Однако в этом случае Коля должен проиграть Игорю или завершить игру с ним вничью. ★

Пусть в переменной  $a$  сохраняется результат встречи Саши с Толей (0, 1/2 или 1), в переменной  $b$  – результат встречи Коли с Игорем. С учетом этого условие единоличной победы Саши можно записать так: если  $a = 1$  OR ( $a = 1/2$  AND  $b < 1$ ).

Разработайте и сохраните проект с именем **Шахматы**, реализующий данную задачу.



**Проект № 4.** Разработайте и сохраните проект с именем **Артисты**. Проект должен содержать три программы для объектов anjuli (используются образы 1, 2, 3), анпа (используются образы 1, 2, 3) и

princess1. Все объекты находятся в папке People. Сначала на сцене (фон chalkboard, папка Indoors) появляется объект anjuli, который четыре раза повторяет три акробатических упражнения. Его выступление сопровождается музыкой. После выступления он исчезает со сцены. На сцене появляется второй объект – anna, который пять раз повторяет три акробатических упражнения. Его выступление сопровождается новой музыкой. После завершения выступления этот объект также исчезает со сцены. Наконец появляется объект princess1, движется по сцене 30 секунд, поет и также исчезает со сцены. ✦

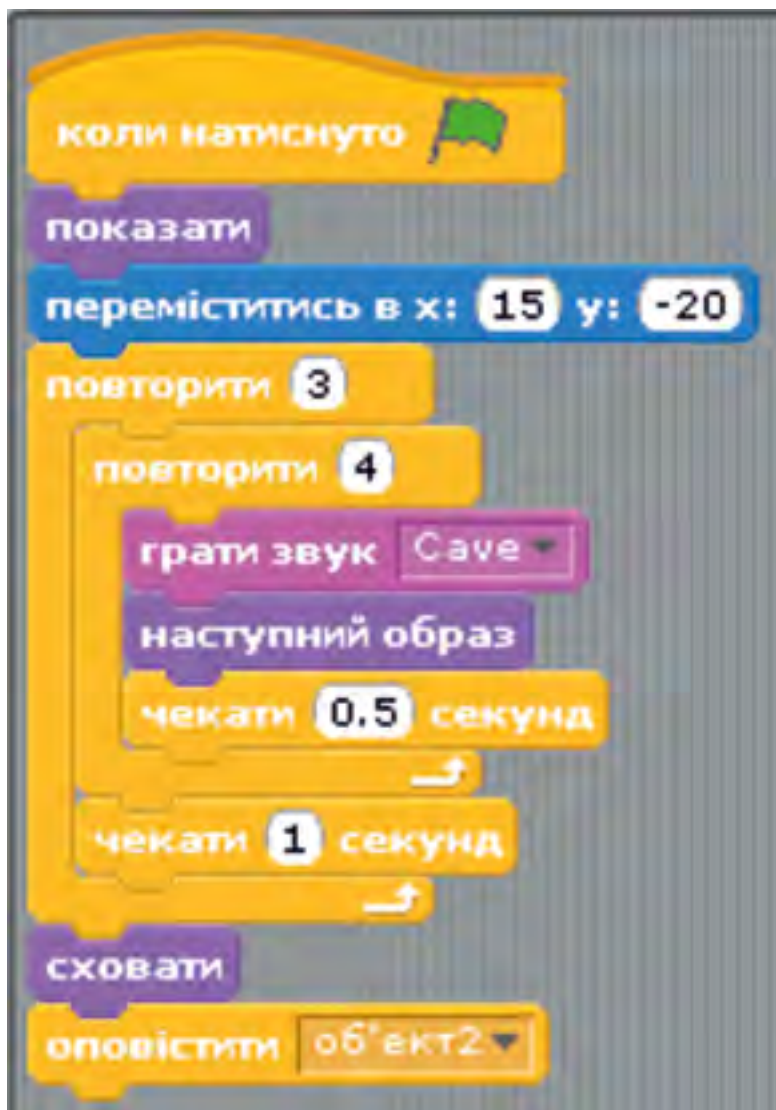


Рис. 6.3. Программа для объекта ballerina

**Рекомендации.** При разработке проекта воспользуйтесь содержанием проекта, приведенным ниже. Он содержит две программы (рис. 6.3 и рис. 6.4). Программа, представленная на рис. 6.3, разработана для первого объекта (ballerina и ее образы – *a, b, c, d*). Этот объект трижды повторяет четыре упражнения. Во время выступления балерины играет музыка Cave. Чтобы выбрать название этой музыки, необходимо выполнить последовательность команд: **Звуки** → **Импортировать** → папка Music Loops → Cave → ОК.

Программа для второго объекта (cassy и ее образов jumping, sitting-1 и standing) представлена на рис. 6.4. Объект трижды выполняет три упражнения.

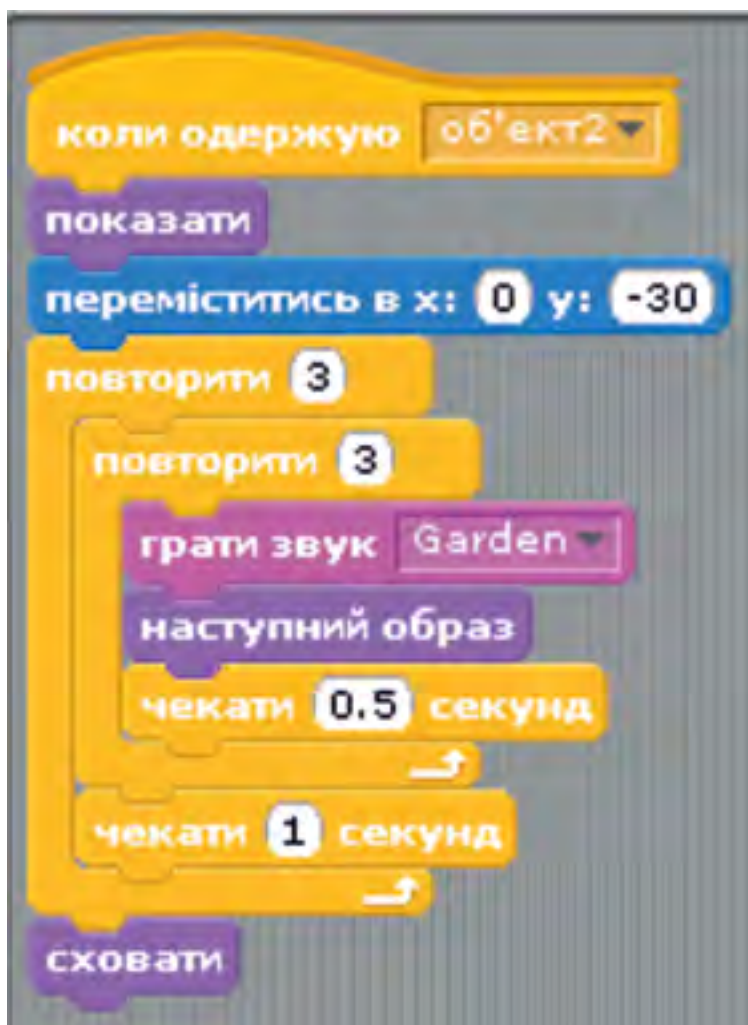
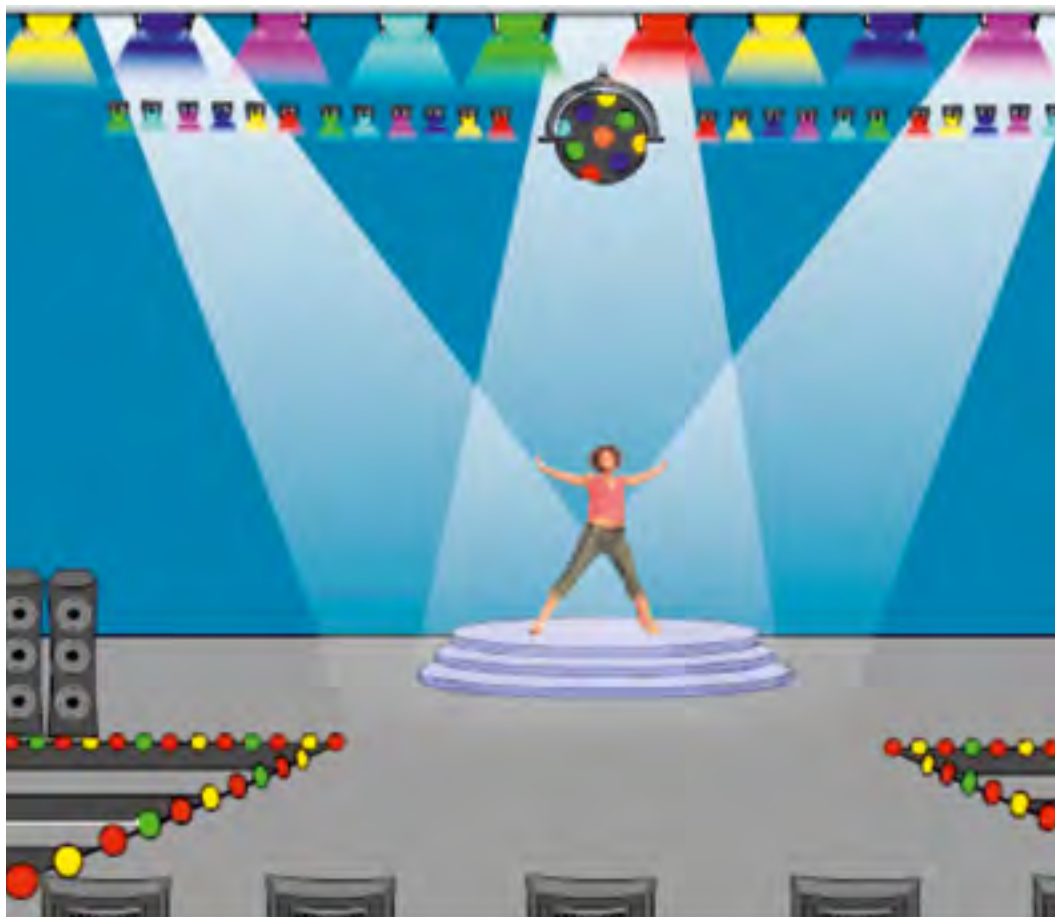


Рис. 6.4. Программа для объекта cassy-jumping



Во время выступления объекта звучит новая музыка Garden, также хранящаяся в папке Musik Loops.

Фрагмент выступления второго объекта представлен на рис. 6.5.



*Рис. 6.5. Фрагмент выступления объекта cassy-jumping*

**Проект № 5. По морю** (фон boardwalk, папка Outdoors) одновременно движутся шхуна (sail-boat) и сухогруз (tug-boat). Над ними в небе летит вертолет (helicopter1). Скорость движения объектов разная. Все объекты находятся в папке Transportation. Фрагмент движения объектов показан на рис. 6.6.

Как только объект касается правой границы, он исчезает из поля зрения. Первым завершает полет вертолет, вторым завершает движение сухогруз, последней – шхуна. Разработайте проект, содержащий отдельную программу для каждого объекта. Сохраните проект с именем **Море.** ✦





Рис. 6.6. Фрагмент движения объектов по морю

**Рекомендации.** Для перемещения объектов желательно использовать команду **Скользить**. Вариант программы перемещения вертолета представлен на рис. 6.7. Вертолет исчезает со сцены, если его координата  $x$  достигает значения 220.

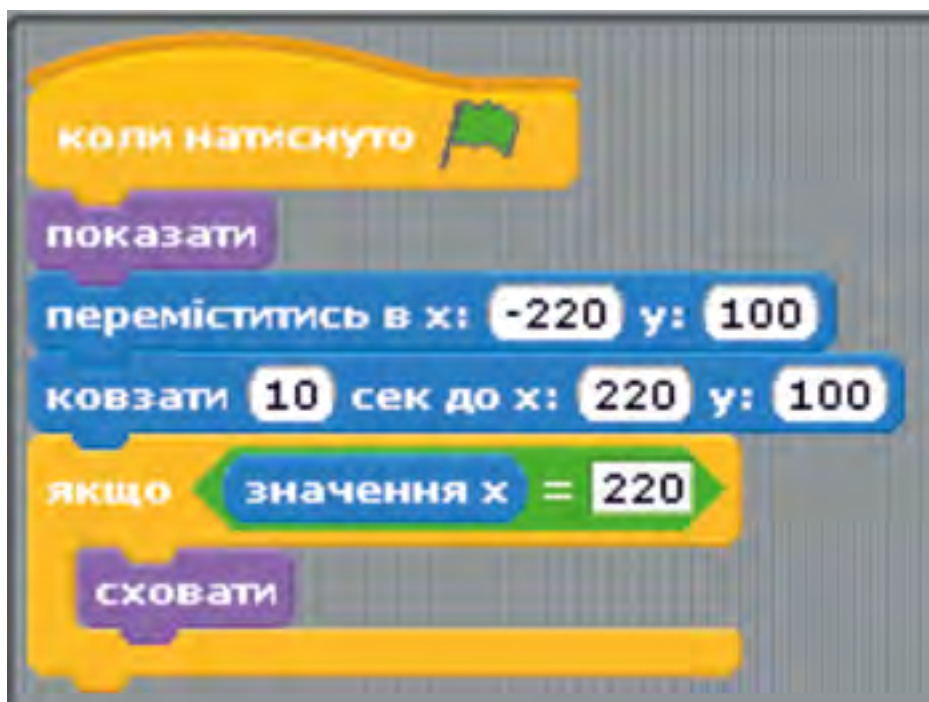


Рис. 6.7. Программа перемещения вертолета



*Рис. 6.8. Фрагмент игры в баскетбол*

**Проект № 6.** На рис. 6.8 приведен фрагмент игры в баскетбол. На рисунке изображены: фон сцены `basketball-court1` (папка `Sports`) и объекты `referee1` (папка `People`), `amon1` (папка `People`) и `beachball1` (папка `Nrings`). Разработайте условие задачи для этого сценария и проект, содержащий программы для каждого изображенного объекта. ★



## СОДЕРЖАНИЕ

Как работать с этой книгой . . . . .	4
<b>РАЗДЕЛ 1. ЭЛЕКТРОННАЯ ПЕРЕПИСКА . . . . .</b>	<b>5</b>
1.1. Электронная переписка. Электронный почтовый ящик . . . . .	5
1.2. Добавление файлов к письмам. Перенаправление писем . . . . .	14
1.3. Правила и этикет электронной переписки.	
Правила безопасного пользования электронным почтовым ящиком	16
<i>Практическая работа № 1 . . . . .</i>	<i>18</i>
<b>РАЗДЕЛ 2. МОДЕЛИРОВАНИЕ . . . . .</b>	<b>21</b>
2.1. Модель. Предметная область . . . . .	21
2.2. Типы моделей. Формы представления информационной модели . . .	24
2.3. Построение информационных моделей . . . . .	27
<i>Практическая работа № 2 . . . . .</i>	<i>30</i>
<b>РАЗДЕЛ 3. АЛГОРИТМЫ С ПОВТОРЕНИЕМ И РАЗВЕТВЛЕНИЕМ . . .</b>	<b>35</b>
3.1. Основные сведения о среде учебного программирования Скретч (повторение изученного) . . . . .	35
3.2. Простейшие алгоритмы с повторением . . . . .	46
3.3. Типы данных среды Скретч . . . . .	56
3.4. Константы, переменные и списки. . . . .	63
3.5. Алгоритмы с разветвлением . . . . .	74
3.6. Алгоритмы повышенной сложности . . . . .	82
3.6.1. Циклические алгоритмы с разветвлением. . . . .	82
3.6.2. Алгоритмы с вложенными циклами . . . . .	86
<i>Практическая работа № 3 . . . . .</i>	<i>92</i>
<i>Практическая работа № 4 . . . . .</i>	<i>93</i>
<i>Практическая работа № 5 . . . . .</i>	<i>93</i>
<b>РАЗДЕЛ 4. ОБРАБОТКА ДАННЫХ В ТАБЛИЦАХ . . . . .</b>	<b>96</b>
4.1. Электронные таблицы, табличный процессор. . . . .	96
4.2. Данные в электронных таблицах . . . . .	105
4.3. Формулы в электронных таблицах . . . . .	117
4.4. Встроенные функции . . . . .	122
4.5. Диаграммы . . . . .	129
4.6. Моделирование в электронных таблицах. . . . .	138
<i>Практическая работа № 6 . . . . .</i>	<i>142</i>
<i>Практическая работа № 7 . . . . .</i>	<i>143</i>
<i>Практическая работа № 8 . . . . .</i>	<i>143</i>

РАЗДЕЛ 5. КОМПЕТЕНТНОСТНЫЕ ЗАДАЧИ . . . . .	146
5.1. Когда и как возникают компетентностные задачи . . . . .	146
5.2. Компьютерные сети и Интернет . . . . .	149
5.3. Компетентностные задачи по основам алгоритмизации и программирования . . . . .	153
РАЗДЕЛ 6. ЗАДАЧИ ДЛЯ УЧЕБНЫХ ПРОЕКТОВ . . . . .	163
6.1. Применение табличного процессора для выполнения проектов . . .	163
6.2. Создание проектов в среде Скретч . . . . .	163

НАВЧАЛЬНЕ ВИДАННЯ

ГУРЖІЙ Андрій Миколайович  
КАРТАШОВА Любов Андріївна  
ЛАПІНСЬКИЙ Віталій Васильович  
РУДЕНКО Віктор Дмитрович

## ІНФОРМАТИКА

Підручник для 7 класу  
загальноосвітніх навчальних закладів  
з навчанням російською мовою

*Рекомендовано Міністерством освіти і науки України*

**Видано за рахунок державних коштів. Продаж заборонено**

Переклад з української мови  
Перекладачі *Лапінський Віталій Васильович,*  
*Карташова Любов Андріївна*

Російською мовою

Редактор *Л. В. Дячишин*  
Художній редактор *І. Б. Шутурма*  
Коректор *О. А. Тростянчин*

Макет, підбір та колажування ілюстрацій *В. В. Лапінський*  
Обкладинка *О. В. Шингур*

Формат 70×100<sup>1</sup>/<sub>16</sub>.  
Ум. друк. арк. 14,256. Обл.-вид. арк. 13,68.  
Тираж 604 пр. Зам. №

Державне підприємство  
«Всеукраїнське спеціалізоване видавництво «Світ»  
79008 м. Львів, вул. Галицька, 21  
Свідоцтво суб'єкта видавничої справи ДК № 4826 від 31.12.2014  
[www.svit.gov.ua](http://www.svit.gov.ua)  
e-mail: [office@svit.gov.ua](mailto:office@svit.gov.ua)  
[svit\\_vydav@ukr.net](mailto:svit_vydav@ukr.net)

Друк ТДВ «Патент»  
88006 м. Ужгород, вул. Гагаріна, 101  
Свідоцтво суб'єкта видавничої справи ДК № 4078 від 31.05.2011



НАВЧАЛЬНЕ ВИДАННЯ

ГУРЖІЙ Андрій Миколайович  
КАРТАШОВА Любов Андріївна  
ЛАПІНСЬКИЙ Віталій Васильович  
РУДЕНКО Віктор Дмитрович

## ІНФОРМАТИКА

Підручник для 7 класу  
загальноосвітніх навчальних закладів  
з навчанням російською мовою

*Рекомендовано Міністерством освіти і науки України*

Переклад з української мови  
Перекладачі *Лапінський Віталій Васильович,*  
*Карташова Любов Андріївна*

Російською мовою

Редактор *Л. В. Дячишин*  
Художній редактор *І. Б. Шутурма*  
Коректор *О. А. Тростянчин*

Макет, підбір та колажування ілюстрацій *В. В. Лапінський*  
Обкладинка *О. В. Шингур*

Формат 70×100<sup>1</sup>/<sub>16</sub>.  
Ум. друк. арк. 14,256. Обл.-вид. арк. 13,68.  
Додатковий тираж 10 пр. Зам. №

Державне підприємство  
«Всеукраїнське спеціалізоване видавництво «Світ»  
79008 м. Львів, вул. Галицька, 21  
Свідоцтво суб'єкта видавничої справи ДК № 4826 від 31.12.2014  
[www.svit.gov.ua](http://www.svit.gov.ua)  
e-mail: [office@svit.gov.ua](mailto:office@svit.gov.ua)  
[svit\\_vydav@ukr.net](mailto:svit_vydav@ukr.net)