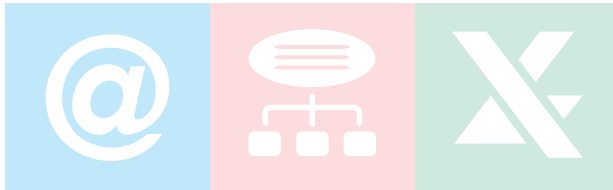


Ольга Казанцева
Ірина Стеценко

ІНФОРМАТИКА

ПІДРУЧНИК ДЛЯ 7 КЛАСУ
закладів загальної середньої освіти

Рекомендовано Міністерством освіти і науки України



ТЕРНОПІЛЬ
НАВЧАЛЬНА КНИГА – БОГДАН
2020

УДК 004(075.3)
К 14

*Рекомендовано Міністерством освіти і науки України
(наказ Міністерства освіти і науки України
від 25.03.2020 р. № 449)*

ВИДАНО ЗА РАХУНОК ДЕРЖАВНИХ КОШТІВ. ПРОДАЖ ЗАБОРОНЕНО

Казанцева О. П.

К 14 Інформатика : підручник для 7 кл. закл. загальн. серед.
освіти / О. П. Казанцева, І. В. Стеценко. — Тернопіль :
Навчальна книга – Богдан, 2020. — 176 с.

ISBN 978-966-10-6113-1

УДК 004(075.3)

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

Завантажуйте безкоштовний інтерактивний додаток,
використовуючи детальну інструкцію, за посиланням:

bohdan-books.com/6113-1



Тех. підтримка: (0352) 52-06-07, 067-352-09-09; admin@bohdan-digital.com

ISBN 978-966-10-6113-1

© Казанцева О. П., Стеценко І. В., 2020
© Навчальна книга – Богдан, виключна
ліцензія на видання, оригінал-макет,
2020

УМОВНЕ ПОЗНАЧЕННЯ



інтерактивна
електронна складова

Дорогі семикласники та семикласниці!

Цей підручник допоможе вам продовжити вивчення інформатики. Матеріал поділений на три розділи. У підручнику описано хід виконання усіх передбачених програмою практичних робіт, створено файли-заготовки до вправ та практичних робіт (знаходяться під відповідним ім'ям на диску інтерактивної складової підручника в папці **FILES**). У першому розділі ви ознайомитеся з сучасними службами Інтернету. У другому розділі ви дізнаєтесь про електронні таблиці та принципи роботи в табличному процесорі. У третьому розділі продовжите працювати в навчальному середовищі виконання алгоритмів Scratch. Завершенням вивчення інформатики в сьомому класі стане створення комп'ютерних моделей у вивчених вами програмних середовищах.

ПУТІВНИК ПІДРУЧНИКОМ

У рубриці підсумовано матеріал параграфу і виокремлено основні поняття і терміни

Пропоновані питання вимагають творчого пошуку і поглиблених знань:

* — достатній рівень,
** — високий рівень навчальних досягнень

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

Висновки

1. Для упорядкування і зберігання адрес електронної пошти використовують адресну книгу. Контакти в ній можна групувати, а потім швидко розсилати листи членам групи.
2. *Спам* — це небажане повідомлення з рекламною або іншою інформацією.
3. *Фішинг* — це вид інтернет-шахрайства, метою якого є отримання доступу до конфіденційних даних користувачів.
4. Учасники електронного листування мають дотримуватися правил етикету, і зокрема: обов'язково зазначати тему листа, дотримуватися зазначеної теми, бути ввічливими, підписувати листи тощо.

Контрольні запитання та завдання

1. Для чого призначено адресну книгу? Яку інформацію в ній можна зберігати?
2. Що таке група контактів? Для чого вона призначена?
- 3*. Опишіть загальний порядок розсилання листів багатьом адресатам.
4. Що означає слово «спам»? Яку небезпеку він несе?
5. Що таке фішинг? Які ознаки фішингу?
6. Яких правил слід дотримуватися під час електронного листування?

Питання для роздумів

1. Як за темою цей лист є спамом?
- 2**. Чому листи, що надіслані...

§12

Практична робота №2. «Уведення, редагування та форматування даних у середовищі табличного процесора»



Завдання. У даній практичній роботі ви маєте створити календар поточного року відповідно до наведеного зразка календаря за 2016 рік.

Хід роботи

1. Заповніть діапазон клітинок **A2:A13** назвами місяців, використовуючи автозаповнення (для цього у вас має бути в даному середовищі MS Excel попередньо створений користувацький список із назвами місяців).

ВПРАВА 3



Завдання. Виконати редагування даних у таблиці **Прайс-лист**.

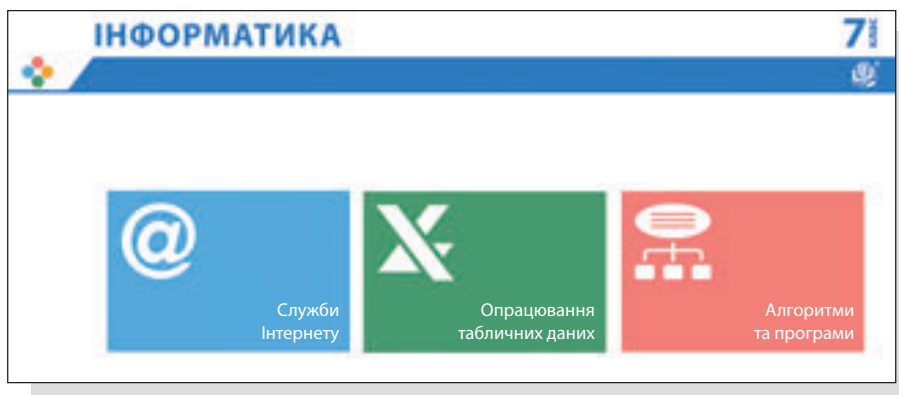
1. Завантажте файл-заготовку до вправи **Прайс.xlsx**.
2. На аркуші **Прайс-лист** видаліть рядки із товаром, що відсутній на складі.
3. Змініть деякі написи у стовпці **Найменування**, переклавши їх українською мовою.
4. Ціну всіх магнітів збільшіть на 10 грн.

Вміщені запитання та завдання дають змогу перевірити засвоєні знання

ПУТІВНИК ІНТЕРАКТИВНОЮ ЕЛЕКТРОННОЮ СКЛАДОВОЮ ПІДРУЧНИКА

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

Головне вікно програми



Розділ 1

Служби Інтернету

Ви дізнаєтесь...

- що таке електронна пошта і як отримати доступ до цієї служби Інтернету
- як формується адреса електронної пошти
- які є особливості використання електронної пошти
- як створити електронну поштову скриньку
- які дії можна виконувати з повідомленнями електронної пошти
- як використовувати адресну книгу
- яких правил безпеки та етикету слід дотримуватися при електронному листуванні
- що таке хмарні технології
- про можливості електронних перекладачів
- що таке Інтернет речей

§1.

Поштові служби Інтернету

1.1.

ЗАГАЛЬНІ ВІДОМОСТІ ПРО ЕЛЕКТРОННУ ПОШТУ

Сучасний світ неможливо уявити без Інтернету, адже ця всесвітня мережа супроводжує нас усюди: вдома, у школі, на роботі, у магазині, під час розваг.



Рис. 1.1

У 5 класі ви отримали основне поняття про комп'ютерну мережу, глобальну мережу Інтернет і таку її службу, як Всесвітня павутина (Веб). Згадаємо деякий матеріал.

- **Комп'ютерна мережа** — це система зв'язку між двома чи більшою кількістю комп'ютерів.
- Комп'ютери у мережі взаємодіють один з одним для того, щоб отримати доступ до певного ресурсу (файлу, програми, вебсторінки, принтера тощо).
- Споживача ресурсу традиційно називають **клієнтом**, а постачальника — **сервером** (від *англ.* server — той, хто обслуговує).
- Сервер у комп'ютерній мережі — потужний комп'ютер, який обслуговує клієнтів, які до нього звертаються.
- **Інтернет** — глобальна комп'ютерна мережа, у якій є сервери, що підтримують роботу певних служб.



Пригадати, що таке вебсторінка, вебсайт, гіперпосилання та інші терміни, ви зможете, виконавши вправи на диску.



Панівне місце серед служб Інтернету займає Веб. Частиною цієї служби є вебсайти, пошукові системи, YouTube (Ютуб), соціальні мережі.

Надзвичайно зручні та поширені також *служби обміну миттєвими повідомленнями*, такі як Skype, Viber та інші. Вони дають змогу дуже швидко обмінюватися текстовими повідомленнями, знімками, іншими файлами. І найголовніше — робити відеовиклики.

Задовго до цих служб в Інтернеті з'явилася і продовжує успішно існувати *служба електронної пошти*. Саме її ви навчитеся використовувати у цьому навчальному році.

Електронна пошта — це служба Інтернету, призначена для організації обміну повідомленнями між користувачами глобальної мережі.



Електронна пошта (від *англ.* e-mail, electronic mail) є певним аналогом звичайної пошти. Зокрема, її повідомлення також називають листами, а для їх отримання необхідно мати поштову скриньку та адресу, звичайно ж електронну. Має служба і свої особливості (рис. 1.2).

- Один лист можна відправити багатьом адресатам.
- До листа можна додавати файли, але не дуже великі за обсягом.
- Відомостей про те, чи є адресат зараз у мережі, служба не надає.
- Читати листи і відповідати на них можна у зручний для себе час.
- Адреси можна зберігати у книзі контактів.
- Відеовиклики, зокрема з абонентами мобільного зв'язку, неможливі.
- Надійність доставки та зберігання кореспонденції дуже висока.
- Є загроза отримати з листом комп'ютерний вірус.

Рис. 1.2. Особливості електронної пошти як служби обміну повідомленнями

Основою служби є поштові сервери, на яких розміщено поштові скриньки та які приймають, надсилають і зберігають електро-

нні листи. Щоб користуватися електронною поштою, потрібно мати електронну скриньку на одному з поштових серверів.

Електронна поштова скринька — це ділянка пам'яті на поштовому сервері. Їй надається унікальна адреса, яка складається з двох розділених спеціальним символом частин — *імені користувача* та *імені поштового сервера*. Цю адресу називають *адресою електронної пошти*.

Розглянемо як приклад адресу: leoant@ukr.net. Тут перша частина **leoant** — це ім'я користувача, друга частина — **ukr.net** — це ім'я поштового сервера.

адреса електронної пошти

LEOANT@UKR.NET

ім'я користувача ім'я поштового сервера

Рис. 1.3. Приклад адреси електронної скриньки

Адреса електронної пошти не може містити пробілів; великі й малі букви в ній не розрізняються.

~~Sergiy Petrenko@ukr.net~~

SergiyPetrenko@ukr.net = sergiypetrenko@ukr.net

Символ @

українською мовою називають «равлик», «вухо». В англomовних країнах на цей символ кажуть «ет», в інших країнах Європи — «мавпа», «черв'як» тощо.



1.2.

ПОШТОВІ СЛУЖБИ

На ранніх етапах існування електронної пошти її послугами могли користуватися лише працівники і працівниці установ та організацій, що мали власні поштові сервери. З часом з'явилися безкоштовні поштові сервери, доступ до яких можна здійснювати через вебінтерфейс з допомогою браузера. Це сприяло тому, що електронна пошта стала доступною практично для всіх.

Сьогодні у світі є багато мережових служб, які надають користувачам послуги вебпошти. В Україні це роблять майже всі великі вебпортали, зокрема **Ukr.net** (<http://www.ukr.net/>), **Мета** (<http://www.meta.ua/>), **Україна онлайн** (<http://www.online.ua/>), **I.UA** (<http://www.mail.i.ua/>).

На рис. 1.4 відображено інтерфейс поштової скриньки системи Ukr.net у вікні браузера.



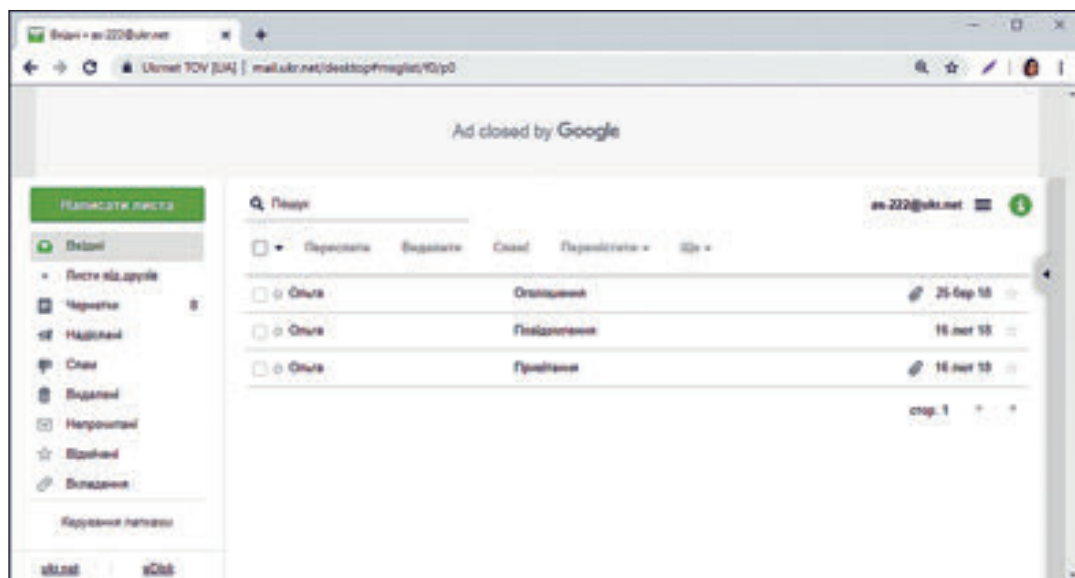


Рис. 1.4. Вікно вебінтерфейсу електронної пошти

Можна звернутися і до закордонних поштових систем, зокрема таких, що надають інтерфейс українською та іншими мовами. Це **Gmail** (<http://gmail.com/>) — поштова система компанії **Google** — та **Outlook** (<http://outlook.com/>) — поштова система компанії Microsoft. Послуги всіх зазначених служб є безкоштовними.



1.3.

ПРИНЦИПИ ФУНКЦІОНУВАННЯ ЕЛЕКТРОННОЇ ПОШТИ

Як зазначалося вище, основу служби електронної пошти становлять поштові сервери, що зберігають електронні повідомлення і керують ними.

Ці сервери незалежні один від одного, проте користуються єдиною розгалуженою системою доставки пошти. Принципи функціонування електронної пошти розглянемо на прикладі обміну листами між двома користувачами — Олегом та Марією.

Адреса скриньки користувача Марії — **mariya@school.edu**, а користувача Олега — **oleg@univer.org**. Як видно з другої частини обох адрес,



Історичні відомості

Електронна пошта — одна з перших служб Інтернету. Цей термін виник 1965 року, а основний внесок у розвиток служби зробила комп'ютерна мережа ARPANET, з якої розпочався розвиток Інтернету. Оскільки тоді далеко не всі мережі були безпосередньо пов'язані між собою, в адресах електронної пошти зазначали ще й маршрут доставлення повідомлення. На щастя, це вже в минулому, і сьогодні маршрути визначаються автоматично.



скриньки цих користувачів містяться на різних поштових серверах (**school.edu** та **univer.org**). Марія на своєму комп'ютері пише листа. Коли вона дає команду його відправити, її комп'ютер зв'язується з поштовим сервером **school.edu** і направляє листа йому. Через низку проміжних серверів електронний лист передається мережею доти, доки не потрапить на поштовий сервер **univer.org**. Отримавши лист, поштовий сервер **univer.org** перевірить у списку своїх скриньок, чи є серед них скринька **oleg**, і, знайшовши таку, помістить листа до неї. Коли Олег почне читати електронну пошту на своєму комп'ютері, той звернеться до скриньки **oleg** на поштовому сервері, знайде в ній новий лист і покаже його Олегові.

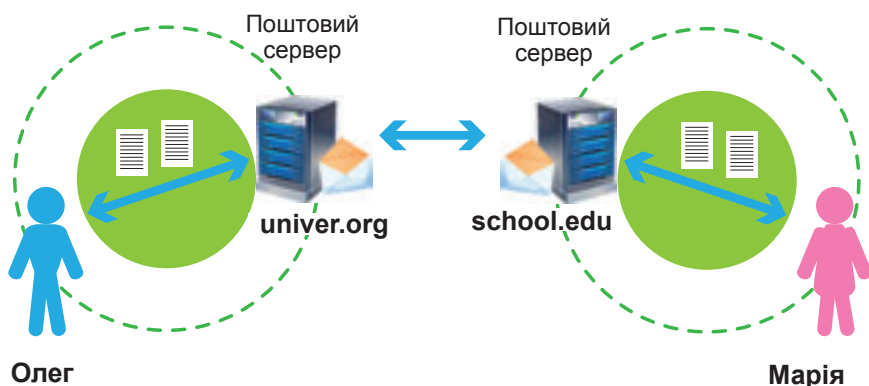


Рис. 1.5. Процес обміну повідомленнями електронної пошти

Лист **не буде доставлено**, якщо:

- ✘ неправильно вказано адресу одержувача;
- ✘ обсяг файлу перевищує допустимі норми.

У разі, якщо адреса одержувача вказана неправильно, служба електронної пошти поверне листа відправникові. Загалом ця служба працює надійно, отож, листи дорогою не губляться. Однак лист не буде доставлений, коли його обсяг перевищує заданий на поштовому сервері обсяг для вхідних листів або коли поштовий сервер одержувача не працює.

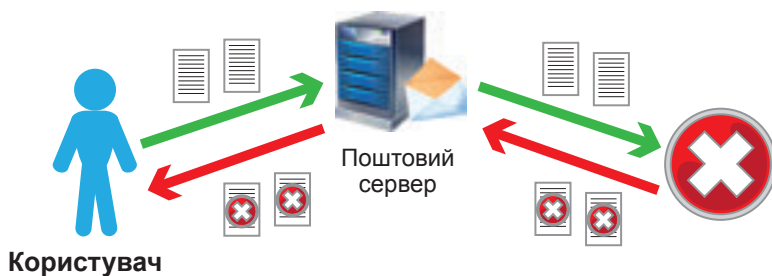


Рис. 1.6. Помилки при передачі листів

У таких випадках служба надішле відправникові повідомлення з описом проблеми.

**Висновки**

Електронна пошта — служба Інтернету, яка забезпечує обмін поштовими повідомленнями між користувачами всесвітньої мережі.

Адреса електронної пошти складається з двох розділених символом «@» частин — імені користувача та доменного імені поштового сервера.

Основу служби становлять поштові сервери, які приймають, надсилають і зберігають електронні листи. Щоб отримати можливість користуватися цією службою, потрібно мати електронну скриньку на поштовому сервері.

Доступ до поштової скриньки онлайнової поштової системи можна отримати через вебінтерфейс, який дає змогу приймати, надсилати і створювати повідомлення з використанням браузера.

Лист не буде доставлено, якщо адреса одержувача вказана неправильно, його обсяг перевищує задане на поштовому сервері значення для вхідних листів або якщо поштовий сервер одержувача не працює.

**Контрольні запитання та завдання**

1. Які є служби Інтернету?
1. Яке призначення електронної пошти? Чому вона так називається?
2. З яких частин складається адреса електронної пошти? Що вони позначають?
3. Що таке вебпошта? Хто надає цю послугу?
4. Опишіть процес обміну повідомленнями електронною поштою між двома користувачами.
5. У яких випадках електронний лист не буде доставлений?
- 6*. Що потрібно мати на комп'ютері, аби працювати з електронною поштою? Опишіть різні варіанти.

**Питання для роздумів**

- 1*. Чому власники порталів надають безкоштовний доступ до поштової служби?
- 2*. Чи можна зареєструвати скриньки з адресами **petro_oliynik@ukr.net** та **petro_oliynik@gmail.com**, у яких імена користувача однакові? Відповідь обґрунтуйте.
- 3**. Як ви думаєте, де фізично на поштовому сервері зберігаються повідомлення? Що собою являє поштова скринька?

§2.

Створення електронної скриньки. Робота з електронною поштою

2.1.

СТРУКТУРА ЕЛЕКТРОННОГО ЛИСТА

Ви вже познайомилися з поштовими службами. Розглянемо, як працювати в одній із таких служб, як створити власну електронну скриньку, як одержати та надіслати листа.

Повідомлення електронної пошти подібне до звичайного поштового листа. В нього є заголовок — аналог адреси на конверті, і тіло повідомлення — аналог власне листа.



Рис. 1.7. Поштове (а) та електронне (б) повідомлення

Заголовок — це набір полів (записів), у яких містяться електронні адреси відправника та одержувачів (їх може бути багато), *тема* повідомлення, час його надсилання тощо. У тілі повідомлення зберігаються його текст та приєднані файли.

На відміну від традиційної пошти, де один екземпляр повідомлення отримує один адресат, електронне повідомлення може надійти кільком людям. При цьому адресату можна відправити *копію* або *приховану копію* листа. У першому випадку всі одержувачі, переглянувши заголовки, будуть знати, хто отримав копію листа, а в другому — вони не знатимуть, кому її ще надіслано.

Електронні листи зберігаються в поштовій скриньці, але на відміну від традиційної скриньки, куди потрапляють лише отримувані вами листи, в електронній містяться всі листи — отримані, відправлені, незавершені тощо. Для впорядкування листів у скриньці використовуються стандартні папки, а також папки, створені власником скриньки у відповідності зі своїми потребами.

Один користувач може мати декілька поштових скриньок. Це можуть бути скриньки на безкоштовних загальнодоступних серверах вебпошти, на поштовому сервері підприємства чи навчального закладу. Скриньку у вебпошті може зареєструвати кожен, натомість в організаціях поштові скриньки створюють лише для їхніх працівників і працівниць.

Таблиця 1. Стандартні папки електронної пошти та їхнє призначення

Папка	Призначення
Вхідні	Стандартне місце зберігання одержаних повідомлень
Чернетки	Місце зберігання неадресованих або незавершених повідомлень
Відправлені	Місце зберігання відправлених повідомлень
Спам	Небажані повідомлення рекламного змісту
Видалені	Місце тимчасового зберігання повідомлень, що видаляються

Доступ до поштової скриньки вебпошти надається через вебінтерфейс, який дає змогу приймати, надсилати і створювати повідомлення з використанням браузера. Перевага такого інтерфейсу в тому, що доступ до поштової скриньки можна здійснювати з будь-якого підключеного до Інтернету комп'ютера за допомогою браузера (рис. 1.4).

Для роботи з електронною поштою також використовують програму *поштового клієнта* (наприклад, Outlook, Mozilla Thunderbird, The Bat!, Opera Mail та ін.). Встановивши її на комп'ютері та налаштувавши на один чи кілька серверів електронної пошти, можна завантажувати пошту на свій комп'ютер і читати її без підключення до Інтернету.



Дізнатися більше

про принцип роботи та інтерфейс поштового клієнта можна в електронному додатку.



2.2.

ПОРЯДОК СТВОРЕННЯ ПОШТОВОЇ СКРИНЬКИ

Для реєстрації поштової скриньки в будь-якій безкоштовній поштовій службі, треба зробити кілька стандартних кроків. Розглянемо їх на прикладі поштової служби ukr.net.

1. Спочатку необхідно відкрити в браузері сторінку поштової служби (наприклад, ukr.net).
2. Обрати посилання **Створити скриньку**.

Пошта

Ім'я скриньки

Пароль

☐ Чужий комп'ютер

Увійти Не вдається увійти?
Створити скриньку

Рис. 1.8. Панель входу та реєстрації електронної скриньки в службі Ukr.net

3. Заповнити запропоновану службою анкету. Зауважимо, що у всіх службах перелік полів анкети приблизно однаковий (нове ім'я електронної скриньки, пароль, прізвище, ім'я, дата народження, мобільний телефон тощо).
4. Завершити реєстрацію, натиснувши кнопку **Зареєструвати скриньку**. Далі система перевірить заповнення анкети на наявність помилок і у разі необхідності запропонує їх виправити.

Отже, останній крок алгоритму створення електронної скриньки — перехід до роботи з нею.

Нижче наведено загальний алгоритм створення нової електронної скриньки.

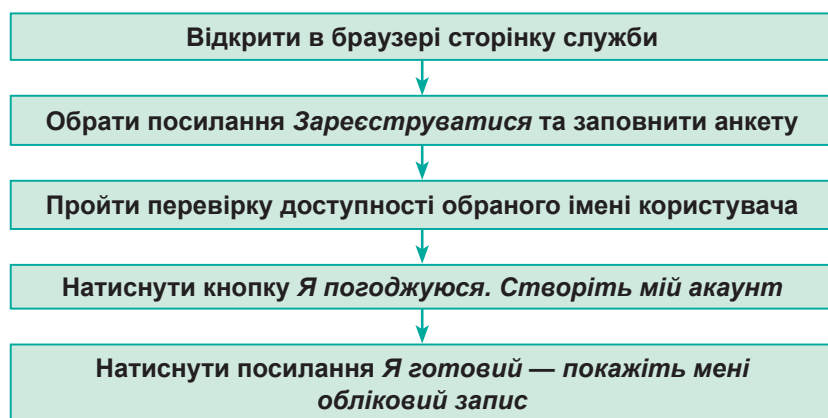


Рис. 1.9. Алгоритм реєстрації поштової скриньки

Якщо ви працюєте не на домашньому комп'ютері, то після роботи з електронною поштою з неї необхідно вийти, натиснувши кнопку **Вихід**.

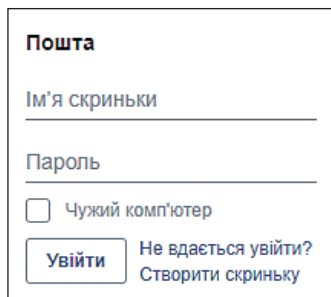
Щоб відновити свою роботу з електронною поштою, необхідно увійти на сайт, що надає послуги електронної пошти, — у нашому випадку це **ukr.net**. Після цього увести адресу електронної скриньки та пароль і знову працювати із власною електронною скринькою.

ВПРАВА 1

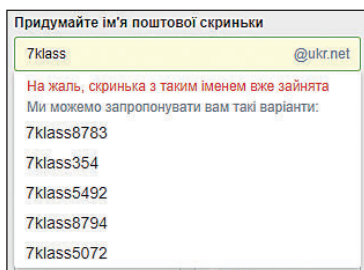


Завдання. Зареєструвати поштову скриньку в безкоштовній службі Ukr.net.

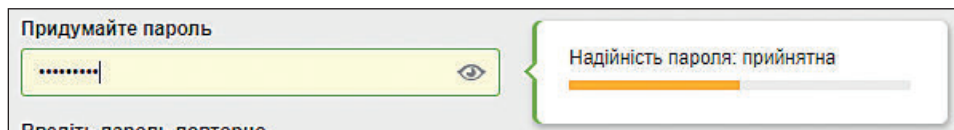
1. Відкрийте стартову сторінку порталу Ukr.net (www.ukr.net).
2. Натисніть посилання **Створити скриньку** для створення власної поштової скриньки.



3. Заповніть усі поля реєстраційної форми.
4. До першого поля введіть першу частину вашої електронної скриньки (Якщо бажаний вами логін уже існує, система повідомить про це і запропонує інші можливі варіанти. Ви можете погодитися із запропонованими системою варіантами або вигадати інший логін.)



5. До поля **Пароль** уведіть пароль (при введенні пароля з'явиться можливість переглянути введений пароль, клацнувши на зображення ока, також система запропонує інструкцію щодо створення надійного пароля).



6. Уведіть своє прізвище та ім'я, вкажіть стать.
7. Уведіть свою дату народження, мобільний телефон та (або) іншу адресу електронної пошти (ці дані необхідні для відновлення доступу до електронної скриньки у разі втрати пароля доступу).
8. Надайте згоду на обробку персональних даних.
9. Клацніть кнопку **Зареєструвати скриньку**. Якщо система видала повідомлення про невдале завершення реєстрації, уважно перегляньте сторінку, заповніть обов'язкові поля і знову клацніть по кнопці **Зареєструвати скриньку**. Якщо усе введено правильно, система повідомить про успішне завершення реєстрації.

! Запам'ятайте або запишіть у потаємному місці пароль, без якого увійти в систему буде неможливо.

2.3.

ВЕБІНТЕРФЕЙС ДЛЯ РОБОТИ З ПОШТОЮ

Веб-сторінки для роботи з поштою в усіх системах подібні. Зліва на сторінці зазвичай розміщено посилання на папки **Вхідні**, **Чернетки**, **Відправлені**, **Видалені**, де зберігаються електронні листи, які надійшли, а також незавершені, відправлені та видалені листи. Якщо клацнути посилання на якусь папку, на екрані відобразиться перелік повідомлень, які в ній містяться.

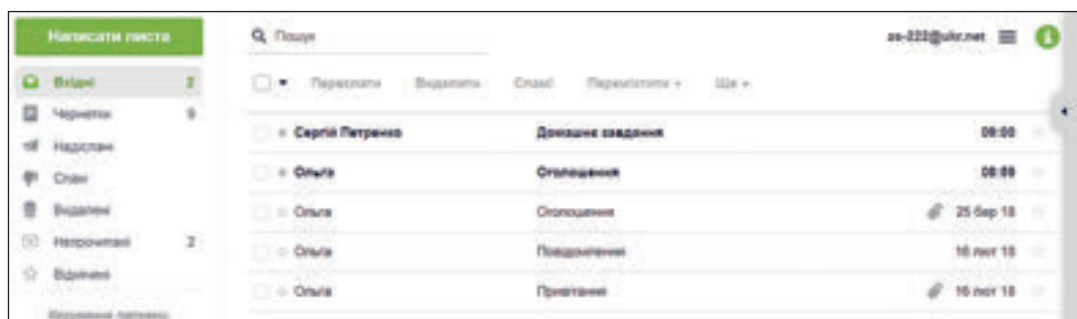


Рис. 1.10. Фрагмент вебсторінки для роботи з поштою служби Ukr.net

Зліва біля кожного повідомлення є прапорці. Встановлюючи їх, можна вибрати одне або кілька повідомлень, а потім, скориставшись кнопками **Переслати**, **Видалити**, **Спам!**, виконати з повідомленнями певні дії.

У верхній частині вебсторінки розташовано поле пошуку з кнопкою **Шукати**, посилання на онлайн-диск **eDisk**, **Налаштування** та **Вихід**, призначення яких відповідає їхній назві. Зазначимо, що на загальнодоступних комп'ютерах (зокрема, на шкільних) для закінчення роботи з вебпоштою слід вибирати посилання **Вихід**, а не просто закривати вікно браузера, оскільки подія закриття браузера не означає вихід з вашого облікового запису електронної пошти. Щоб розпочати новий сеанс, треба зайти на головну сторінку служби (в нашому прикладі Ukr.net), у поля **Логін** та **Пароль** ввести потрібні дані та клацнути кнопку **Увійти** (рис. 1.11).

Пошта

Ім'я скриньки

7klass_a

Пароль

.....

☐ Чужий комп'ютер

Увійти Не вдається увійти?
Створити скриньку

Рис. 1.11. Панель входу до електронної скриньки поштової служби Ukr.net

2.4.

НАДСИЛАННЯ ПОВІДОМЛЕНЬ І ФАЙЛІВ

Щоб надіслати кому-небудь листа, треба виконати такі дії.

1. Створити новий електронний лист, клацнувши посилання **Написати листа** на сторінці для роботи з поштою. Після цього відкриється сторінка, подібна до наведеної на рис. 1.12.
2. У вікні для набору тексту листа ввести текст.
3. У поле **Кому** ввести адресу одержувача (тут можна ввести кілька адрес через кому).
4. Відкрити поля **Копія** і **Прихована копія** і ввести в них адреси одержувачів звичайної та прихованої копій листа, якщо це потрібно.
5. У поле **Тема** записати тему повідомлення.
6. Якщо до повідомлення потрібно докласти файли, клацнути посилання **Файл** і вибрати файли у вікні, що відкриється.
7. Для надсилення листа клацнути кнопку **Надіслати**. Копію листа буде збережено в папці **Відправлені**.

Під час написання листа він автоматично зберігається у папці **Чернетки**. Після вдалого відправлення листа він з папки **Чернетки** автоматично буде видалений. У папці **Чернетки** також можна зберегти незавершений лист, клацнувши кнопку **Зберегти**.

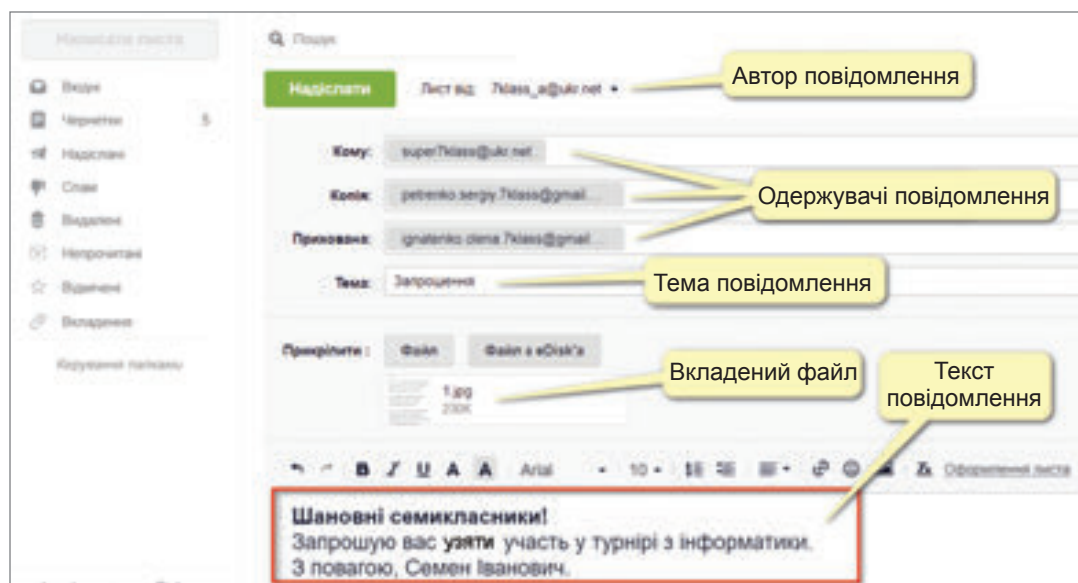


Рис. 1.12. Вебсторінка створення нового листа в службі Ukr.net

Електронною поштою можна пересилати файли будь-яких форматів, проте низка поштових служб (зокрема, Gmail) з міркувань безпеки не приймає файли з розширенням імені **.exe**, оскільки саме такі файли схильні до ураження вірусами. Окрім того, в поштової службі може бути встановлене обмеження на обсяг вкладень як для листів, що відправляються, так і для листів, що надходять. Це слід урахувувати під час пересилання файлів електронною поштою.



Рис. 1.13. Можливі помилки при додаванні до повідомлень файлів

2.5.

ПЕРЕГЛЯД ПОВІДОМЛЕНЬ І НАДСИЛАННЯ ВІДПОВІДЕЙ

Для перегляду отриманих повідомлень потрібно клацнути посилання **Вхідні**, після чого на сторінці буде відображено вміст папки вхідних листів (рис. 1.14). Непрочитані листи подаються в цій папці напівжирним шрифтом, а прочитані — звичайним. Листи із вкладенням зазвичай позначаються значком у вигляді скріпки.



Рис. 1.14. Перевірка надходжень

Щоб прочитати лист, достатньо клацнути його тему або ім'я відправника. Після цього відкриється вікно із вмістом вибраного повідомлення (рис. 1.15).

Якщо лист містить вкладення, для його відкриття і збереження на диску вашого комп'ютера потрібно клацнути посилання **Завантажити**, для збереження на е-диску — посилання **Збереження на eDisk**, а для перегляду — посилання **Переглянути**. Зазначимо, що поштові служби Gmail та Outlook надають можливість переглядати файли деяких форматів он-лайн за допомогою онлайн-ових версій поширених офісних прикладних програм.

Щоб відповісти на лист, слід клацнути кнопку **Відповісти**, після чого відкриється сторінка для створення листа, на якій поле з адресою одержувача буде заповнено. Також цей лист буде мати ту саму тему, але з позначкою **Re**, що означає відповідь на лист. Область повідомлення буде містити місце для введення тексту, а в нижній частині — інформацію про одержаний лист та вміст одержаного листа.

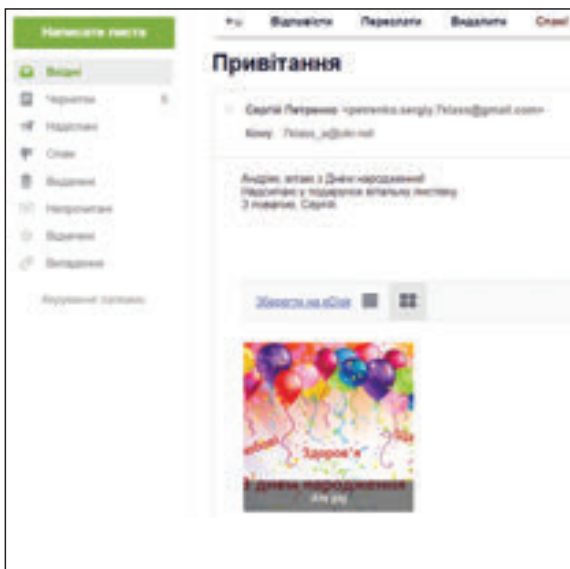


Рис. 1.15. Одержаний лист



Потренуватися

реєструвати
поштову скриньку

та виконати дії
з електронними листами
ви зможете на поштовому
тренажері на диску.

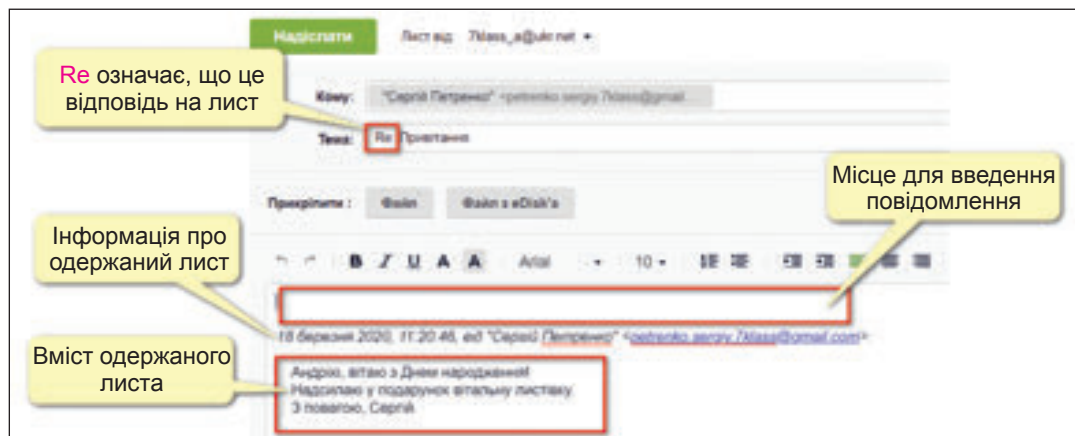


Рис. 1.16. Відповідь на лист

**Висновки**

Найпростіший спосіб отримати можливість користуватися електронною поштою — створити власну скриньку на одному з порталів, які пропонують цю послугу.

Доступ до такої скриньки можна отримати з будь-якого комп'ютера, підключеного до Інтернету, на якому встановлено браузер.

Процедура реєстрації поштової скриньки на всіх порталах подібна, основна дія — **задати логін, пароль та ім'я користувача**. Щоб забезпечити свою пошту від зловмисників, зберігайте в секреті її логін і пароль.

Поштове повідомлення має **адресу одержувача, тему повідомлення, саме повідомлення**. До повідомлення електронної пошти можна **приєднати файли** майже усіх форматів.

**Контрольні запитання та завдання**

1. Опишіть загальний порядок створення електронної скриньки на безкоштовному поштовому сервері.
2. Що потрібно для того, аби отримати можливість користуватися електронною поштою?
3. З якими стандартними папками працюють поштові служби? Яке їхнє призначення?
- 4*. Як передивитися вкладені файли до електронного листа?
5. Як правильно закінчувати роботу з вебінтерфейсом електронної пошти?
- 6*. Які додаткові послуги, крім електронної пошти, пропонують портали?

**Питання для роздумів**

1. У яких випадках доцільно надсилати звичайну копію повідомлення, а в яких — приховану? Наведіть приклади.
- 2*. Лист можна надіслати також самому собі. Яку користь приносить така можливість?

§3.

Адресна книга. Етикет електронного листування. Правила безпечного користування електронною скринькою

3.1.

АДРЕСНА КНИГА

Електронна пошта широко застосовується як засіб ділового листування. Працівники і працівниці, для яких така комунікація є невід'ємною частиною роботи, отримують десятки листів за день і листуються з десятками і сотнями людей. За таких умов особливого значення набуває проблема впорядкування електронних адрес та швидкий пошук потрібної адреси.

Як вебінтерфейс, так і поштові клієнти містять засоби, що прискорюють введення адрес електронної пошти, дають змогу зберігати їх та керувати ними. Наприклад, щойно ви почнете вводити адресу одержувача, вам буде запропоновано на вибір адреси, які трапляються у вашій пошті та починаються із введеної літери (рис. 1.17).

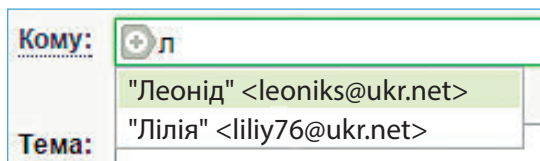
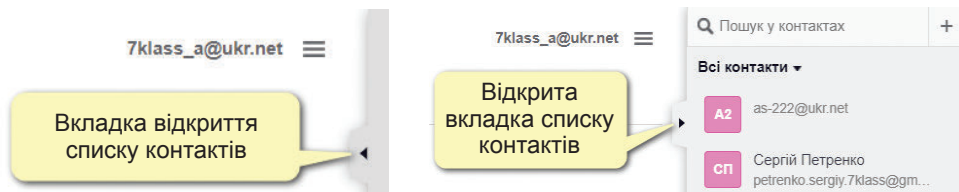


Рис. 1.17. Список контактів деякої поштової скриньки, що починаються на літеру «Л»

У системах електронної пошти інформація про людину зазвичай позначається терміном «контакт». До цієї інформації належать ім'я та прізвище особи, адреса електронної пошти, телефони, місце роботи, дата народження тощо. Усі ці відомості зберігаються в адресній книзі, яку іноді ще називають книгою контактів. Для її відкриття потрібно клацнути посилання **Контакти** або натиснути ледь помітну вкладку зі стрілочкою у правій частині вікна електронної скриньки.



Адресна книга — це список адрес електронної пошти та імен користувачів, з якого обирають одержувачів повідомлень.

Контакти в адресну книгу потрібно заносити вручну, але багато систем вебпошти і поштових клієнтів допомагають вам у цьому, формуючи списки адрес людей, з якими ви найчастіше

спілкуєтесь. Клацнувши елемент у такому списку, ви можете доповнити інформацію про контакт і включити його у відповідну групу (наприклад, **Друзі**, **Однокласники**).

Для створення нового контакту потрібно натиснути кнопку із зображенням плюса, після чого відкриється форма **Додавання контактів**.

Як бачимо, засобами даної форми можна зберегти інформацію про прізвище та ім'я особи, її електронну адресу, занести її до певної групи контактів, увести номер телефону, місце роботи, дату народження тощо.

Після заповнення форми, слід натиснути кнопку **Зберегти**.

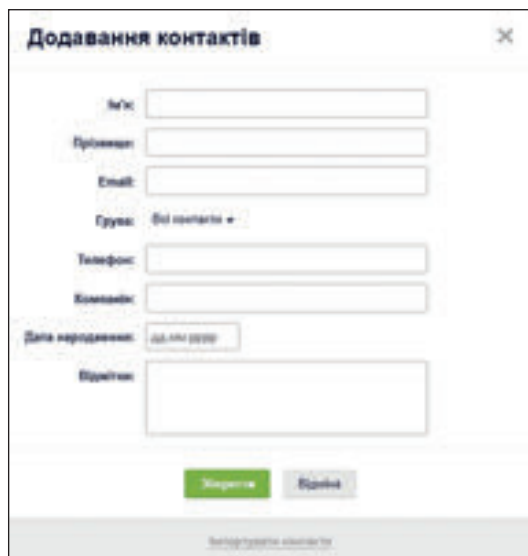


Рис. 1.18. Форма для створення нового контакту в службі Ukr.net

3.2.

РОБОТА З ГРУПАМИ КОНТАКТІВ

Як ви вже знаєте, електронний лист можна надіслати кільком адресатам, указавши їх у полях **Кому**, **Копія** та **Прихована копія**. А якщо адресатів більше десятка? Як уникнути монотонної роботи з вибиранням адрес у разі, коли, наприклад, учитель періодично надсилає учням й ученицям електронні листи із завданням? Для цього зручно користуватися групами, які можна створювати в адресних книгах.

Загальний порядок дій при цьому такий.

1. Відкрити вкладку контактів. Відкрити список **Всі контакти**.
2. Обрати посилання **Нова група**. У полі, що з'явилося, ввести назву нової групи (наприклад, з іменем **Друзі**, **Учні 7-а**, **Мої однокласники** тощо).
3. Натиснути кнопку **Додати**. Після чого у списку буде відображено назву нової групи.

Щоб відобразити список усіх груп, слід натиснути стрілку біля назви групи.

При наведенні на назву групи з'являються інструменти, що дозволяють відредагувати її назву, а також видалити цю групу.

Щоб до групи додати контакт, необхідно обрати цей контакт у групі **Всі контакти** або в будь-якій іншій групі й натиснути посилання **Редагувати**.

У результаті відкриється вікно **Редагування контакту**. Обрати в цьому вікні у полі **Група** назву групи, до якої слід віднести контакт.

Щоб надіслати листа усім контактам групи, необхідно натиснути кнопку **Написати листа**. У полі **Кому** почати вводити назву групи, обрати необхідну групу. Після цього у полі **Кому** буде відображено усі контакти групи. У разі необхідності можна прибрати непотрібні контакти з даного листа, навівши курсор миші на їхню адресу та натиснувши хрестик.

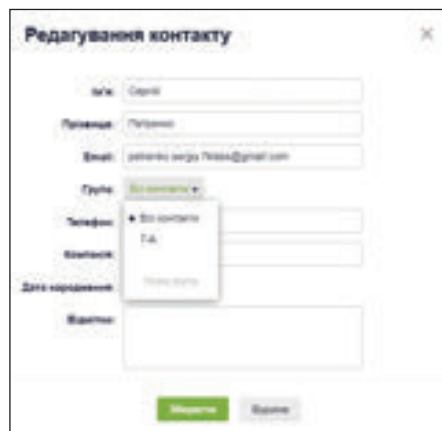
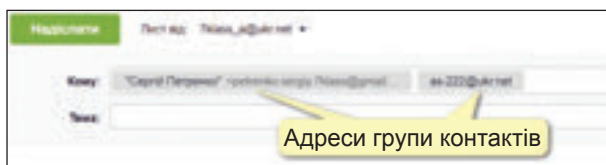


Рис. 1.19. Зміна групи в режимі редагування контакту



3.3.

ЩО ТАКЕ СПАМ ТА ФІШИНГ

Електронна пошта постійно піддається інформаційним атакам рекламного характеру. Листи, що містять таку інформацію, називають *спамом*, а тих, хто його розсилає, *спамерами*. Адреси поштових скриньок для надсилання листів спамери вгадують з допомогою спеціальних програм, які генерують адреси за певними правилами, або беруть із форумів та сайтів, де їх залишають необережні користувачі.

Ще один із видів шкідливих повідомлень — це *фішинг*.

Фішинг — це один із видів інтернет-шахрайства, метою якого є отримання доступу до конфіденційних даних користувачів: логінів, паролів, даних особових рахунків і банківських карт.

Слово *спам* (англ. *spam*) походить від назви м'ясного продукту, реклама якого була дуже нав'язливою.



Дізнатися більше
про спам та фішинг
ви зможете в елек-
тронному додатку.

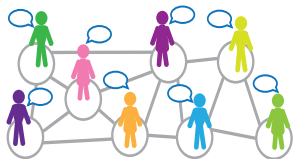


Фішинг, в основному, використовує метод проведення масових розсилок від імені популярних компаній або організацій,

які містять посилання на помилкові сайти, які зовні відрізнити від справжніх неможливо (рисунок з ноутбуком і крючечками залиште навпроти інформації про фішинг).

3.4.

ЕТИКЕТ ЕЛЕКТРОННОГО ЛИСТУВАННЯ



Завдяки електронній пошті люди, що живуть у різних куточках світу, мають змогу спілкуватися між собою, витрачаючи на це мінімум зусиль, часу і коштів. Щоб таке листування було приємним, потрібно дотримуватися правил етикету електронного спілкування.

- **Завжди зазначаєте тему листа.** Тема має якомога точніше відображати зміст повідомлення.
- **У тексті листа дотримуйтеся зазначеної теми.** Пам'ятайте, що багато людей щодня одержують десятки листів, тому викладайте свої думки чітко та лаконічно.
- **Не надсилайте листів із вкладеними файлами великих обсягів.** Це не лише сповільнює пересилання, а й може стати причиною повернення вашого листа, якщо на поштовому сервері встановлено обмеження на обсяг вкладеного файла.
- **У відповіді наводьте уривки з листа, на який відповідаєте, а не весь лист.** Цитувати потрібно лише ті фрази, що безпосередньо стосуються змісту вашої відповіді.
- **Не пишіть увесь текст великими літерами.** Такий текст дуже важко читати, застосовуйте великі літери до окремих слів, які необхідно виділити.
- **Розбивайте текст на логічні абзаци.** Це насамперед стосується листів з великим обсягом тексту.



Для надання електронним листам емоційного забарвлення використовують смайлики. **Дізнатися більше** про їх використання ви зможете в електронному додатку. 

- **Перевіряйте помилки.** По-перше, з грамотною людиною приємно мати справу, а по-друге, помилки в словах або при розставленні пунктуаційних знаків можуть призвести до двоякого тлумачення змісту повідомлення.
- **Підписуйте лист.** У підписі рекомендовано зазначати своє ім'я та адресу електронної пошти, а в діловому листуванні бажано додавати контактний телефон.

- **Не затягуйте з відповіддю.** Відповідати на листа слід упродовж доби.
- **Будьте ввічливі.** Це правило не потребує коментарів.



Висновки

1. Для упорядкування і зберігання адрес електронної пошти використовують адресну книгу. Контакти в ній можна групувати, а потім швидко розсилати листи членам групи.
2. *Спам* — це небажане повідомлення з рекламною або іншою інформацією.
3. *Фішинг* — це вид інтернет-шахрайства, метою якого є отримання доступу до конфіденційних даних користувачів.
4. Учасники електронного листування мають дотримуватися правил етикету, і зокрема: обов'язково зазначати тему листа, дотримуватися зазначеної теми, бути ввічливими, підписувати листи тощо.



Контрольні запитання та завдання

1. Для чого призначено адресну книгу? Яку інформацію в ній можна зберігати?
2. Що таке група контактів? Для чого вона призначена?
- 3*. Опишіть загальний порядок розсилання листів багатьом адресатам.
4. Що означає слово «спам»? Яку небезпеку він несе?
5. Що таке фішинг? Які ознаки фішингу?
6. Яких правил слід дотримуватися під час електронного листування?



Питання для роздумів

1. Як за темою та адресою відправника листа можна визначити, що цей лист є спамом?
- 2**. Чому листи, які не є спамом, іноді потрапляють у папку зі спамом?

§4.

Практична робота №1. Електронне листування з використанням вебінтерфейсу. Вкладені файли

Завдання. Виконавши практичну роботу, ви навчитеся користуватися електронною поштою за допомогою вебінтерфейсу безкоштовної поштової служби. Ви створите електронне повідомлення, прочитаете надіслані електронні повідомлення і відповісте на листи від однокласників та однокласниць із вкладеними до них файлами, а також дізнаєтесь, як видаляти листи і переміщувати їх до іншої папки.

Хід роботи

1. Відкрийте зареєстровану поштову скриньку, створіть та надішліть електронний лист.
 - а) Перейдіть на стартову сторінку порталу Ukr.net. У поле **Ім'я скриньки** уведіть назву електронної скриньки, а в поле **Пароль** — свій пароль.
 - б) Натисніть кнопку **Увійти**, а потім посилання **Вхідні**, щоб відкрити сторінку для роботи зі скринькою та потрапити у папку **Вхідні**.
 - в) Натисніть посилання **Написати листа** у верхній лівій частині вікна. На сторінці, що відкриється, введіть у поле **Кому** електронну адресу одержувача (заздалегідь дізнайтеся її у свого друга), у поле **Тема** — тему повідомлення, у вікно нижче (рис. 1.20) — його текст.

Рис. 1.20. Вікно створення нового повідомлення в поштовій службі Ukr.net

- г) Клацніть посилання **Надіслати**. Якщо все гаразд, система просигналізує про це відповідним повідомленням. Лист буде переміщено до папки **Відправлені**.

2. Прочитайте електронний лист від друга чи подруги та надішліть відповідь на нього, приєднавши до листа графічний файл.
- a) Перевірте, чи не надійшов вам лист. Для цього клацніть посилання **Вхідні**. Нові листи відображаються у вікні вхідних повідомлень. На панелі папок ліворуч біля назви **Вхідні** число означає кількість непрочитаних повідомлень. Префікс **Re:** біля повідомлення означає, що це відповідь на ваш лист із зазначеною далі темою (рис. 1.21).

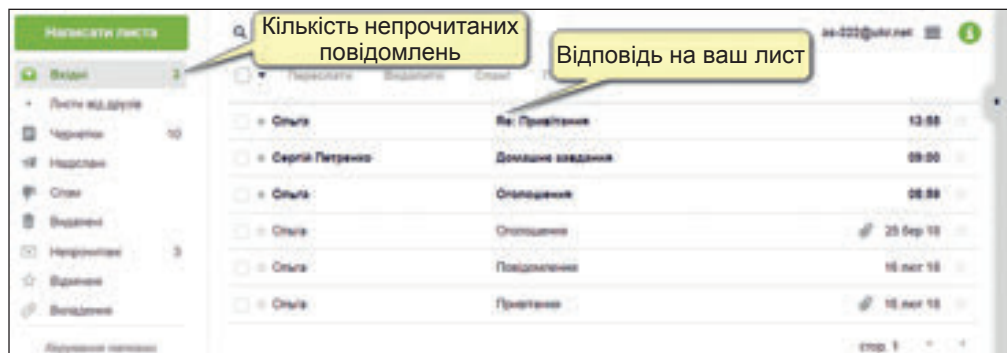


Рис. 1.21. Папка Вхідні в поштовій службі Ukr.net

- б) Щоб прочитати новий лист, клацніть його тему. Відкриється вікно із текстом повідомлення.
- в) Для надсилання відповіді на лист, клацніть посилання **Відповісти**. Після чого відкриється вікно, де в полі **Кому** буде розміщено адресу автора вихідного листа, в полі **Тема** — попередню тему із префіксом **Re:**, а у вікні введення тексту повідомлення буде наведено текст вихідного листа.
- г) Введіть відповідь, залишивши тільки ті цитати, на які ви реагуєте (рис. 1.22).

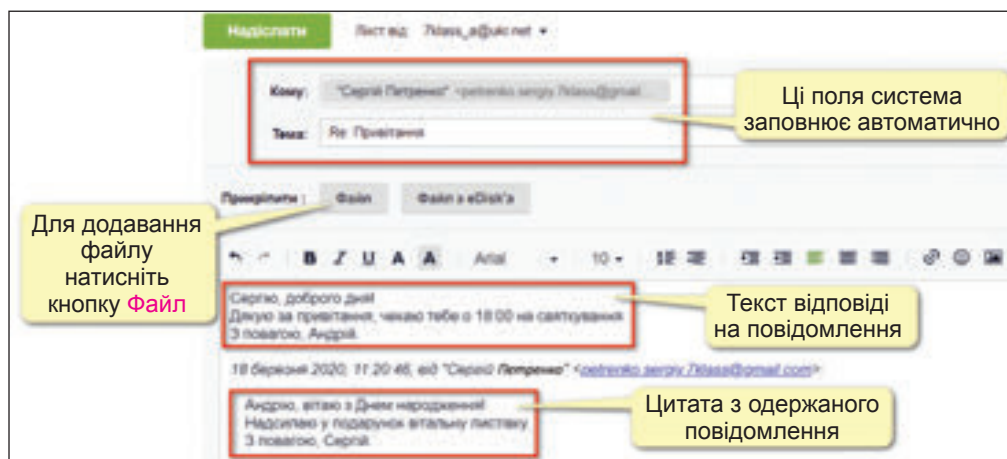


Рис. 1.22. Вікно створення відповіді на одержаний лист в Ukr.net

- г) Додайте будь-який графічний файл до повідомлення, натиснувши кнопку **Файл** та обравши потрібний файл на диску (зауважте, що обсяг файла має бути не дуже великим, до 1 Мбайт).
 - д) Клацніть посилання **Надіслати**, щоб надіслати лист-відповідь.
3. Видаліть будь-який лист до папки **Видалені**, після чого відновіть його до папки **Вхідні**.
- а) Зайдіть до папки **Вхідні**, підготуйте до видалення будь-який лист, установивши прапорець ліворуч від теми повідомлення.
 - б) Клацніть кнопку **Видалити**. Після чого повідомлення потрапить до папки **Видалені**.
 - в) Увійдіть до папки **Видалені**, підготуйте до переміщення будь-який лист, установивши прапорець ліворуч від теми повідомлення.
 - г) Оберіть команду **Відновити в** ► **Вхідні**. Після чого повідомлення потрапить до папки **Вхідні**.
- Примітка.** У разі потреби видалити повідомлення назавжди в папці **Видалені** слід клацнути команду **Видалити назавжди**.
4. Створіть власну папку **Листи від друзів** та перемістіть туди всі листи, які ви отримали від друзів.
- а) Натисніть посилання **Керування папками**.
 - б) У вікні, що відкрилося, натисніть знак «+» біля напису **Створити папку**. Введіть назву **Листи від друзів** та натисніть кнопку **Зберегти**. В результаті буде створено нову папку з указаною назвою.
 - в) Зайдіть до папки **Вхідні** та перемістіть з неї усі листи від друзів до папки **Листи від друзів** (спосіб переміщення листів з однієї папки до іншої описано в попередньому пункті).

Додаткове завдання. Створіть книгу контактів з електронними адресами своїх однокласників й однокласниць.

§5.

Хмарні технології. Онлайнові перекладачі

5.1.

ОСНОВНІ ПОНЯТТЯ

Одним із сучасних напрямів розвитку інформаційних технологій є так звані «хмарні» технології. Звичайно, про те, що технології знаходяться десь у хмарах, не йдеться. Навпаки, ці технології підтримуються великою кількістю потужних комп'ютерів (серверів), об'єднаних у так звані *датацентри*. Як виглядає датацентр ізсередини, показано на малюнку.



Подібні датацентри обслуговують такі відомі вам служби, як YouTube, Google, Фейсбук та інші. Особливістю хмарних технологій є те, що програми та дані розташовуються на обладнанні постачальника (у датацентрі) і надаються користувачам (окремим особам, підприємствам та установам) через мережу у вигляді платних або безкоштовних вебслужб.

**Дізнатися більше**

про історію походження "хмарних"

технологій ви зможете в електронному додатку.



Назву «хмарні» технології отримали через те, що у вигляді хмари на схемах комп'ютерних мереж позначали Інтернет або іншу складну частину, яку детально не описували.

Щоб краще зрозуміти особливості хмарних технологій, розглянемо електронну пошту. З нею можна працювати, встановивши на комп'ютер поштовий клієнт, такий як Outlook. При цьому його можна налаштувати так, що повідомлення з поштового сервера будуть завантажуватися на комп'ютер і зберігатися на його жорсткому диску. Після цього їх можна читати і без підключення до Інтернету.

Натомість вебпошта на зразок **ukr.net** — це вже *хмарна технологія*. На комп'ютер



користувача ніякі спеціальні програми для роботи з поштою не встановлюють, доступ до пошти отримують через браузер, а самі повідомлення на комп'ютері не зберігаються. Звичайно, при цьому читати повідомлення можна лише за наявності інтернет-підключення.

Не всі хмарні технології є безкоштовними. Є послуги, за які користувачеві доведеться сплатити кошти.

Так, плату можуть знімати:

- за надання віддаленого доступу до сховища;
- за доступ до спеціального програмного забезпечення для обробки даних;
- за доступ до комплексу програмного забезпечення.

Використання хмарних технологій має ряд переваг, які є досить суттєвими.

Хмарні технології — це можливість отримати доступ до необхідної інформації за допомогою звичайного браузера з будь-якої точки планети. Роботоздатність обладнання більше не буде хвилювати клієнта, оскільки за нею стежать ті, кому платить користувач за хмарне сховище. Подібні системи користуються попитом у корпоративних користувачів, яким необхідно налагодити документообіг у мережі підприємств.

Молодим підприємствам дуже вигідно використовувати хмарні сервери. Їм не доведеться піклуватися про покупку свого серверного обладнання, витратитися на побудову локальної мережі, наймати системного адміністратора. Досить просто вибрати один із хмарних серверів, який ідеально підходить за обсягом пам'яті, кількістю клієнтів й іншими характеристиками, і платити раз на місяць абонентську плату.

Для звичайних користувачів, які просто не хочуть засмічувати комп'ютер зайвою інформацією, існують безкоштовні хмарні сервіси, що надають дисковий простір.

Найпотужніші хмарні технології для підтримки діяльності підприємств та вебсайтів сьогодні пропонують лідери ІТ-індустрії **Google, Microsoft, Amazon**.

Що стосується звичайних користувачів, то вони для безкоштовного зберігання своїх даних у хмарі можуть скористатися дисковим простором у таких службах, як **ukr.net** або **Dropbox**.



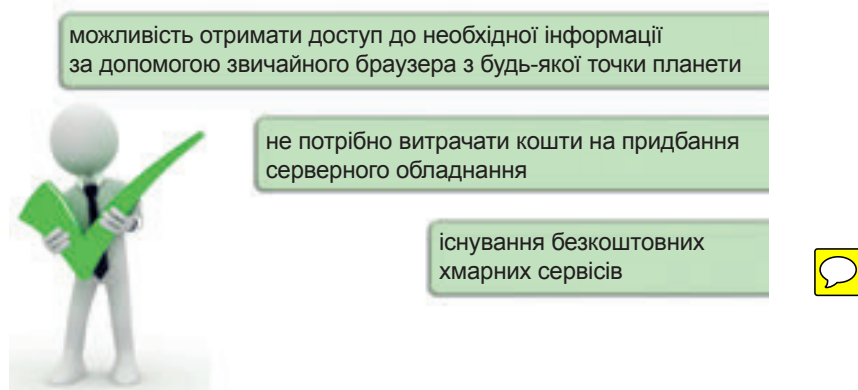


Рис. 1.23. Переваги використання хмарних технологій

Для Dropbox є додаток для смартфона, служба забезпечує синхронізацію даних на різних комп'ютерах.

Ще одна дуже корисна послуга хмарних технологій — тимчасове зберігання файлів та надання посилань на них. У такий спосіб ви можете обмінюватися файлами великого обсягу, посилаючи не сам файл, а посилання на нього.

Звичайно, хмарні технології мають свої недоліки, основні з них перелічені на рис. 1.24.

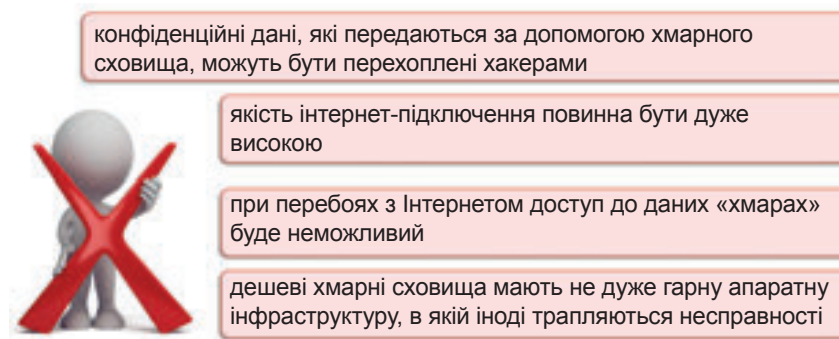


Рис. 1.24. Недоліки використання хмарних технологій

Роботу з хмарними технологіями ви вивчатимете на прикладі служб **OneDrive** та **Google Drive**. Ці служби є схожими за своїм функціоналом. Вони надають змогу, використовуючи лише браузер, працювати з текстовими документами, презентаціями, електронними таблицями та іншими документами. За наявності таких служб у користувачів відпадає потреба постійно переносити файли на флешдисках або пересилати їх електронною поштою, з'являється можливість спільної роботи над документами.

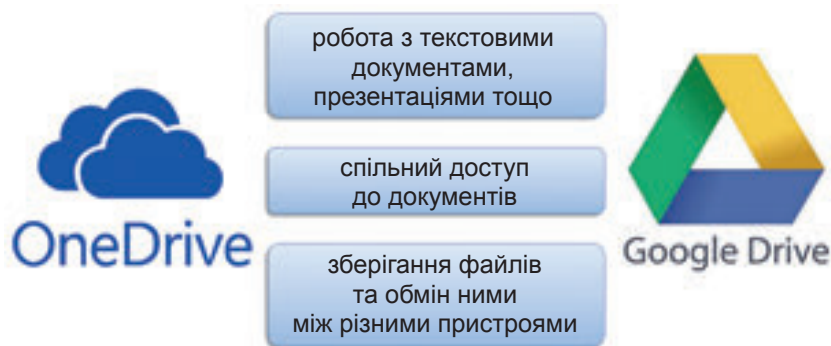


Рис. 1.25. Можливості платформ OneDrive та Google Drive

Розглянемо можливості служби OneDrive.

5.2.

ОГЛЯД СЛУЖБИ ONEDRIVE

OneDrive — це хмарне сховище від компанії Microsoft, основним завданням якого є забезпечення обміну файлами між різними пристроями, а також надання додаткового місця для зберігання інформації.

На даний момент компанія Microsoft пропонує кілька варіантів використання Windows OneDrive:

- **безкоштовний план** — в користування надається 5 Гб вільного місця;
- **платні плани** в залежності від їхньої вартості надають до 1 Тб вільного місця та використання офісу 365 домашньої версії.



Як створити власний обліковий запис Microsoft, ви можете **дізнатись** в електронному додатку.

Для отримання доступу до хмарного сховища OneDrive необхідно мати власний обліковий запис Microsoft. Для входу до OneDrive достатньо перейти на сайт **onedrive.live.com**, натиснути кнопку **Вхід**, увести свою електронну адресу, натиснути кнопку **OneDrive** та ввести свій пароль з облікового запису Microsoft.

За умовчанням в операційних системах Windows 8 та Windows 10 програму OneDrive вже встановлено на комп'ютерах, і у разі входу до операційної системи за обліковим записом Microsoft усі дані з хмарного диска OneDrive будуть синхронізовані з відповідною папкою на вашому комп'ютері. Синхронізувати вміст цієї папки можна на кількох комп'ютерах і смартфоні, якщо встановити на них додатки OneDrive.



Рис. 1.26. Синхронізація даних з хмарного диска OneDrive

5.3.

ІНТЕРФЕЙС ПРОГРАМИ ONEDRIVE

Розглянемо призначення елементів інтерфейсу вікна програми OneDrive.

Ліворуч у вікні розташовано меню. За умовчанням відкрито розділ **Файли**, що містить файли користувача, розміщені на диску OneDrive.

Розділ **Останні** відображає розглянуті за останній час документи.

Фото — сервіс зручної роботи з фотографіями, розміщеними на диску.

Розділ **Спільні** відображає папки та файли, до яких вами надано спільний доступ для використання, а також документи, до яких ви отримали спільний доступ від інших користувачів.

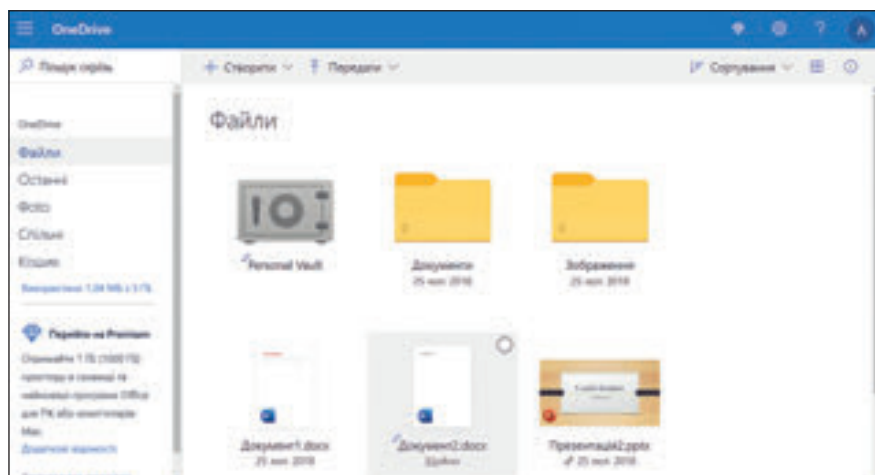


Рис. 1.27. Інтерфейс програми OneDrive

Кошик — папка для тимчасового збереження видалених файлів.

У вікні програми за умовчанням створено дві папки: **Документи** та **Зображення**. Користувач може легко створити й інші папки, обравши команду **Папка** із меню **Створити**. Після чого система запропонує ввести ім'я нової папки.



Як створити та виконати редагування документа в Office Online, ви можете дізнатися в електронному додатку.



В контекстному меню папки можна обрати необхідні дії для роботи з нею. Так, папку можна видалити, перемістити або копіювати до іншої папки, перейменувати.

Меню **Створити** містить можливість створити не тільки папку, але й документи різних типів: текстові документи, презентації, електронні таблиці тощо.

До диска також можна завантажити файли або папки, обравши відповідну команду із меню **Передати**. Потім достатньо обрати необхідний файл або папку на диску комп'ютера, після чого їх буде завантажено до диска OneDrive.

5.4.

СПІЛЬНА РОБОТА

Створені в Office Online документи можна зробити відкритими для інших. Щоб надати доступ до файлів певним користувачам, потрібно встановити біля файлів прапорці і вибрати команду **Спільний доступ**.

Після чого відкриється вікно, в якому можна встановити налаштування спільного доступу:

- надати можливість редагування документа, або документ буде доступний лише для перегляду;
- установити дату завершення дії спільного доступу.

Після натискання на кнопку **Отримати посилання** буде надано посилання, яке можна скопіювати, а потім вставити у будь-яке місце для надання спільного доступу за цим посиланням. Посилання можна вставити до листа електронної пошти, на сторінку вебсайту, просто в текстовий документ. Після цього усі, хто будуть мати це посилання, можуть завантажити позначений до-

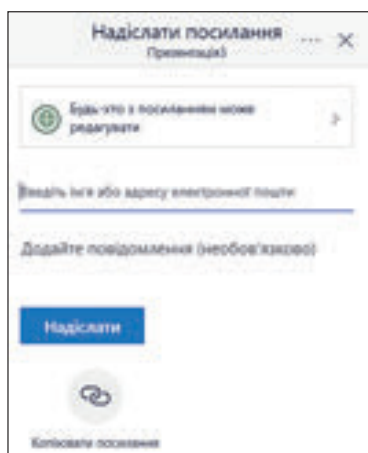


Рис. 1.28. Вікно надання спільного доступу

кумент та виконати його редагування, якщо для нього не було встановлено опцію заборони редагування.

Також у вікні надання спільного доступу можна відразу надіслати посилання на електронну пошту, обравши відповідну кнопку **Електронна пошта**.

Надайте спільний доступ до створеної вами презентації.

- 1) Позначте презентацію прапорцем.
- 2) Натисніть кнопку **Спільний доступ**.
- 3) Надішліть посилання собі на електронну пошту — для цього введіть адресу електронної скриньки та введіть деяке повідомлення. Натисніть кнопку **Спільний доступ**.
- 4) Переконайтесь, що посилання надійшло до електронної скриньки, увійшовши до неї.

ВПРАВА 2



Завдання. З допомогою служби OneDrive створити презентацію з теми «Історія виникнення Інтернету» та надати спільний доступ до неї учителю.

1. Зайдіть до сайту onedrive.live.com під власним обліковим записом.
2. Створіть нову презентацію.
3. На першому слайді додайте заголовок «Історія виникнення Інтернету». В підзаголовок введіть назву навчального закладу, клас, ваше прізвище та ім'я.
4. На другому та наступних слайдах розташуйте інформацію з історії виникнення Інтернету. Як заголовки вкажіть роки події (це мають бути роки 1962, 1969, 1971, 1973, 1983, 1984, 1988, 1989, 1990, 1995, 1997).
5. Знайдіть інформацію в Інтернеті відповідно до кожного року події та розташуйте на слайдах текстову та графічну інформацію.
6. Текстову інформацію можна копіювати з інших сайтів, виконуючи копіювання виділеного фрагменту з використанням контекстного меню. Додавання на слайд здійснюється з використанням команди **Вставити** контекстного меню слайду. (Пам'ятайте, що інформація на сайтах може міститися у вигляді таблиць і безпосереднє копіювання інформації із сайту може призвести до некоректного відображення тексту на слайді. Отже, рекомендуємо виконувати копіювання фрагменту тексту зі слайду до текстового редактора, встановленого на комп'ютері, а потім після його обробки до онлайнвої презентації.)
7. Вставку тексту можна також виконати, використовуючи спеціально призначений для цього інструмент **Текстове поле**, який розташований у вкладці **Вставлення**.

8. Після додавання тексту виконайте його форматування, що можна здійснити, використовуючи відповідні інструменти у вкладці **Формат**, що з'являється у разі виділення рамки тексту.
9. Додайте необхідні зображення. Для цього скористайтеся контекстним меню сайту, звідки здійснюється копіювання зображення. Щоб додати зображення, збережене на диску, слід обрати інструмент **Зображення** на вкладці **Вставлення**.
10. Надайте презентації ім'я **Історія Інтернету**.
11. Надайте спільний доступ до створеної вами презентації та надішліть посилання вчителю для перевірки.

5.5.

ОНЛАЙНОВІ ПЕРЕКЛАДАЧІ



Дізнатися більше
про роботу онлайн-перекладачів
ви можете дізнатися в
електронному додатку.



Ще одним корисним і потужним онлайн-сервісом є онлайнві перекладачі.

Онлайн-перекладачі — це системи, що дозволяють здійснювати переклади текстів, документів і навіть цілих сайтів за один клік. Щоб скористатися онлайн-перекладачами, зазвичай потрібно виконати усього декілька кроків:

- 1) увести текст для перекладу в онлайн-перекладач;
- 2) вибрати мову вихідного тексту;
- 3) вибрати мову тексту, на яку потрібно здійснити переклад.

Практично відразу ви отримаєте готовий переклад. Таким чином, онлайн-перекладачі покликані «зламати» мовний бар'єр між людьми, дати їм можливість спілкуватися, читати новини, блоги, іноземні сайти, вивчати культуру інших країн, подорожувати.



Висновки

1. Хмарні технології — це можливість мати доступ до даних у режимі онлайн, не встановлюючи спеціальних додатків на пристрої.
2. Основним недоліком використання хмарних технологій є високі вимоги до якості інтернет-підключення.
3. Найбільш поширеними платформами, що надають «хмарні» послуги, є OneDrive та Google Drive.
4. Одним із зручних інтернет-сервісів є онлайнві перекладачі. Найбільшого поширення здобув Google-перекладач.

**Контрольні запитання та завдання**

1. Що таке хмарні технології?
2. Вкажіть переваги та недоліки використання хмарних технологій.
3. Які ви знаєте платформи надання послуг хмарних технологій?
4. Які послуги надає служба OneDrive?
5. Опишіть принцип роботи онлайн-перекладачів.

**Питання для роздумів**

1. Які онлайн-програми служби OneDrive відповідають програмам, установленим на вашому комп'ютері?
2. Проаналізуйте можливості служб OneDrive та Google Drive. Вкажіть, що у них є спільного та чим вони відрізняються.

§6.**Інтернет речей****6.1.****ІСТОРІЯ ПОХОДЖЕННЯ ТЕРМІНА**

Інтернет став справжнім проривом у людській цивілізації. З його допомогою виникло багато нових напрямів в економіці та соціальному житті. Одне з них — це Інтернет речей. Що це таке? У чому його суть? Прогрес це чи ні? Розберемося із цим усім далі.

Термін «інтернет речей» (англ. «internet of things», скорочено IoT) уперше був сформульований ще у далекому за технологічними мірками 1999 році. Його сформулював американський науковець Кевін Ештон у своїй праці, яка описувала роботу системи, у якій об'єкти можуть спілкуватися між собою за допомогою Інтернету. І основне, на що зверталась увага, що усе це здійснювалось без втручання людини.

**Дізнатися більше**

про історію походження Інтернету речей ви зможете в електронному додатку. 

**Кевін Ештон
1999 р.**

Робота, в якій фізичні об'єкти комунікували між собою без втручання людини



6.2.

ЩО ТАКЕ ІНТЕРНЕТ РЕЧЕЙ



Інтернет речей — це певна мережа, в яку об'єднано речі. Причому речами може бути все що завгодно: автомобіль, холодильник, кондиціонер. Вони мають бути оснащені давачами, що сприймають інформацію з навколишнього середовища, обмінюватися нею та виконувати різні дії залежно від цієї інформації. Тобто йдеться про взаємодію за схемою «машина–машина» з мінімальною участю людини.

6.3.

ЯК ЦЕ МАЄ ПРАЦЮВАТИ

Інтернет речей має на увазі, що людина визначає мету, а не задає програму з досягнення цієї мети. Ще краще, якщо система сама аналізує дані і передбачає бажання людини.

Одним із яскравих прикладів Інтернету речей є розумний будинок. Розумний будинок «вгадує» ваші бажання, «знає» ваші звички і робить усе так, щоб вас оточували комфорт та затишок. Розумний будинок просипається раніше за вас, переводячи усі системи в необхідний режим після довгої ночі, коли з метою економії було знижено температуру, потім система будить вас улюбленою музикою з нарощуванням рівня гучності, в цей час на кухні вже приготовлено улюблену каву. Ви їдете на роботу, а розумний будинок відключає освітлення та побутові прилади, закриває двері, вмикає сигналізацію, відеоспостереження, систему охорони периметра території будинку. Розумний будинок щохвилини буде підлаштовуватися під ваш режим використання різних систем і приладів, які він містить.



Головні особливості Інтернету речей:

- постійний супровід повсякденних дій людини;
- усе відбувається прозоро, ненав'язливо, з орієнтацією на результат;
- людина вказує, що повинно бути зроблено, а не як це зробити.

Скажете, фантастика? Ні, це найближче майбутнє, але, щоб домогтися таких результатів, необхідно ще багато чого зробити.

6.4.

ЯК ЦЬОГО ДОСЯГТИ

Як же цього досягти?

Єдиний центр. Необхідно створити деякий єдиний центр. Звичайно, що в центрі всіх цих речей повинна стояти не людина, а якийсь керуючий пристрій — контролер, який і буде передавати програму з досягнення мети. Він контролюватиме інші пристрої й виконання завдань, а також збиратиме дані. Такий контролер повинен стояти в кожному будинку, офісі та інших місцях. Їх об'єднає єдина мережа, через яку вони будуть обмінюватися даними і допомагати людині в будь-якому місці.

Зачатки таких центрів розробляються вже зараз, вони частково можуть виконувати роль центрів розумного будинку, хоча їхні можливості поки що обмежені.

Єдині стандарти. Для повнофункціональної реалізації ідеї Інтернету речей мають бути відпрацьовані єдині стандарти.

Це стане, мабуть, головною перешкодою на шляху до глобального Інтернету речей. Для масштабної роботи системи має бути єдина мова. Над своєю екосистемою зараз працюють Apple, Google, Microsoft. Але всі вони рухаються окремо, в різні боки, а отже, в кращому випадку, ми отримаємо локальні системи, які складно об'єднати навіть на рівні міста.

Можливо, з часом якась із систем стане стандартом, або кожна мережа так і залишиться локальною і не переросте в щось глобальне.

Безпека. Природно, розробляючи таку систему, необхідно подбати про захист даних.

Якщо мережу зламає хакер, він буде знати про вас абсолютно все. Розумні речі «розкажуть» зловмисникам про вас усе, а отже, над шифруванням даних, що передаються, потрібно багато попрацювати. Звичайно, над цим працюють уже й зараз, але до ідеальної безпеки ще далеко.



6.5.

ДЕ ВИКОРИСТОВУЄТЬСЯ ІНТЕРНЕТ РЕЧЕЙ

Побут. Телевізор, холодильник, принтер і вся система «розумний дім» можуть бути доведені до повної автоматизації. З підключенням Інтернету речей людина позбавиться нав'язливих думок (чи не забула вона закрити двері або вимкнути плиту) —



всю інформацію можна буде побачити в смартфоні і дистанційно запуснути незавершений алгоритм. А «розумний» пілосос сповістить про знайдену мамину сережку, яка закотилася за крісло.

Безпека. Впровадження технології Інтернету речей в охоронну систему дозволить сканувати й відправляти фото- і відеодані, а штучний інтелект буде здатний запам'ятовувати і розпізнавати людей.



Промисловість. Виробництво стане автоматизованим: системи моніторингу сповістять про можливі проблеми і збої в роботі обладнання, а клієнт зможе реалізувати замовлення на відстані.



Медицина. Інтернет речей потрібен для того, аби персоналізувати пристрої, що допомагають вивчати і контролювати життєві показники людини, а при необхідності викликати лікаря.



Транспорт. У світі Інтернету речей не буде пробок — мережа давачів і сенсорів розпізнає завантаженість та оптимізує роботу транспортних магістралей.



Рітейл. Звичайні магазини знову стануть конкурувати з онлайн-бутіками — будуть налагоджені автоматична передача та аналіз інформації про клієнта через платіжний термінал.

6.6.

ПРОБЛЕМИ ТА НЕДОЛІКИ ІНТЕРНЕТУ РЕЧЕЙ

Які ж виникають проблеми та недоліки впровадження Інтернету речей?

Відсутність єдиної системи. Як уже було зазначено раніше, проблема інтеграції Інтернету речей — у відсутності загальних правил і стандартів. Поки не буде розуміння загальної картини, складно впровадити універсальне рішення.

Енерговитратність. Для повноцінної роботи Інтернету речей потрібно домогтися автономності мережі й отримувати енергію з навколишнього середовища.

Відсутність приватності. Основний ризик — у відкритій базі даних. У шахраїв з'явиться можливість зламувати не тільки рахунки і комп'ютери, але навіть холодильники.

Вартість. Техніка дорога, незважаючи на те, що її використання окупиться в майбутньому: система «розумний дім» допо-

може заощадити на електриці і водопостачанні; обладнання на виробництві завчасно сповістить про ризик поломки; кухонна техніка дозволить уникнути псування продуктів.

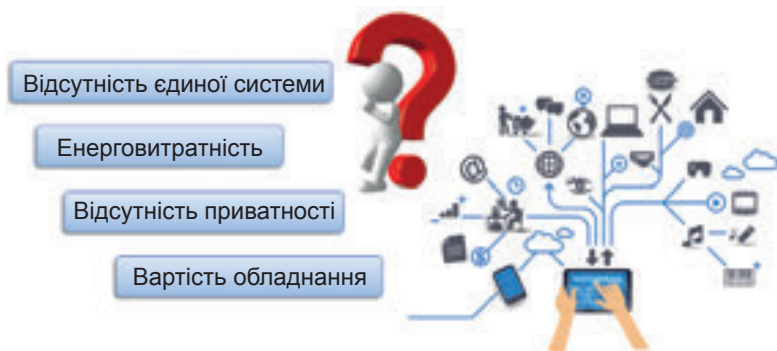


Рис. 1.29. Проблеми впровадження Інтернету речей

6.7.

НОВІ МОЖЛИВОСТІ В МАЙБУТНЬОМУ

У найближчому майбутньому нас чекають розумні будинки, які будуть самі відкривати двері для власників при наближенні, підтримувати комфортний мікроклімат, самостійно поповнювати холодильник і замовляти необхідні ліки, якщо людина захворіла. Перед цим будинок отримає показники з розумного браслета і відправить їх лікарю. По дорогах їздитимуть безпілотні автомобілі, а на самих дорогах більше не залишиться пробок. Інтернет речей дозволить розробити більш просунуту систему контролю трафіка, яка зможе запобігати появі пробок і заторів на дорогах.

Поки складно сказати, яким буде Інтернет речей через десять років, але не залишається сумнівів, що застосування інноваційного алгоритму буде активно використовуватися на ринку послуг і сприятиме підвищенню якості життя, відкриє перспективні можливості в бізнесі й принесе, загалом, чимало користі.

Уже зараз багато пристроїв працюють у зв'язці з різними системами, і в найближчі 5-10 років нас чекає справжній бум розвитку Інтернету речей.

Завдяки цьому полегшиться праця людей у побуті, з'явиться більше можливостей для саморозвитку, подорожей, спілкування з близькими і друзями, поліпшиться якість медичних послуг, розширяться можливості для дотримання здорового способу життя.



Дізнатися більше про сучасні можливості використання Інтернету речей ви зможете в електронному додатку.



ВПРАВА 3



Завдання. Створити презентацію на одну із запропонованих тем:

1. Розумний будинок.
2. Пристрої Інтернету речей, які ми використовуємо сьогодні.
3. Майбутнє Інтернету речей.

Розташуйте презентацію на OneDrive та надішліть посилання вчителю для перевірки.



Висновки

1. Інтернет речей — це певна мережа, в яку об'єднано речі, які можуть взаємодіяти між собою без участі людини за допомогою переданих даних.
2. Для повноцінного використання Інтернету речей необхідні пристрої, оснащені датчиками, центральний контролер, який збиратиме дані та керуватиме пристроями, та мережа, що все це буде поєднувати. Важливими є розробка єдиних стандартів та забезпечення безпеки.
3. Одним із яскравих прикладів впровадження Інтернету речей є «розумний будинок».
4. Інтернет речей є однією з найперспективніших технологій останніх років.



Контрольні запитання та завдання

1. Що таке Інтернет речей?
2. Які пристрої Інтернету речей існують на сьогоднішній день?
3. Що являє собою «розумний будинок»?
4. Опишіть, які виникають проблеми для повноцінного використання технології Інтернету речей.



Питання для роздумів

1. Проаналізуйте, яких технологій не вистачає для того, щоб Інтернет речей став реальністю вже сьогодні.
2. Подумайте, до яких побутових приладів можна застосувати технологію Інтернету речей.

Розділ 2

Опрацювання табличних даних

Ви дізнаєтесь...

- про табличний процесор як програму для опрацювання табличних даних
- що таке електронні таблиці і яке їхнє призначення
- про інтерфейс програми табличного процесора
- які є об'єкти робочої книги
- про формати даних в електронних таблицях
- як вводити та редагувати дані в середовищі табличного процесора
- правила запису формул
- як виконати обчислення з використанням вбудованих функцій
- що таке модель
- причини створення та галузі використання моделей
- про форми подання інформаційних моделей
- які є етапи створення комп'ютерних моделей



§7.

Ознайомлення з електронними таблицями

7.1.

ОЗНАЙОМЛЕННЯ З ПРОГРАМОЮ MICROSOFT EXCEL

У повсякденному житті досить часто виникає потреба в різноманітних обчисленнях. Якщо обчислень небагато, в нагоді може стати звичайний калькулятор. А коли йдеться про велику кількість розрахунків, то самим калькулятором уже не обійтись, оскільки обчислення займуть дуже багато часу.

Для опрацювання табличних даних і розв'язання багатьох задач були створені спеціальні програми, які називали *табличними процесорами* або програмами обробки електронних таблиць.

Табличний процесор — це прикладна програма, яка призначена для опрацювання даних, поданих у таблицях. Таблиці, які опрацьовує табличний процесор, називають **електронними таблицями**.

У таблицях подають багато інформації, зокрема такої, що стосується економіки та статистики. До речі, відомості про успішність учнів також зручно подавати у вигляді таблиці.

Назва Excel походить від англ. *executable cells*, що означає «виконувані клітинки».

Часто виникає потреба зробити на основі табличних даних розрахунки — скажімо, визначити загальну вартість придбаного товару або підсумкову оцінку за рік. Тому і було створено спеціальні програми — табличні процесори.

Обчислення за допомогою електронних таблиць дозволяє виконувати програмний засіб Microsoft Excel, що входить до вже знайомого вам пакета Microsoft Office.

**Історичні відомості**

Ідею електронних таблиць уперше сформулював американський учений Річард Маттессич, опублікувавши в 1961 р. дослідження під назвою «*Budgeting Models and System Simulation*».

**Основні функції Excel:**

- введення, редагування і форматування даних;
- виконання обчислень за формулами;
- графічне зображення даних (побудова графіків, діаграм, вставлення малюнків, географічних карт);
- робота зі списками;
- колективна робота з таблицями (обмін файлами в мережі, спільне використання і захист даних).

Щоб завантажити програму Microsoft Excel на своєму комп'ютері, слід у меню кнопки **Пуск** обрати програму **Excel 2016**. Якщо на комп'ютері встановлено версію 2013 або 2010, то спочатку в меню кнопки **Пуск** слід обрати папку **Microsoft Office**, а вже потім посилання на програму Excel.

Ознайомимося з основними елементами вікна програми.

Як і будь-яка інша програма пакета MS Office, вікно Excel містить вкладки з лініями інструментів. Деякі з них уже відомі після роботи з такими програмами, як Microsoft Word і Microsoft PowerPoint. Переважно це інструменти редагування та форматування тексту або рисунків.

Проте програма Excel має свої особливості. Наприклад, створений у цій програмі документ називають *робочою книгою*, оскільки подібно до книжки він складається з аркушів, які називають *робочими аркушами*.

Ярлики робочих аркушів розташовано у нижньому лівому вікні програми. Для створення нового аркуша достатньо клацнути на ярлику із позначкою «+».

Файл, створений у програмі Excel, має розширення **.xls, .xlsx**.

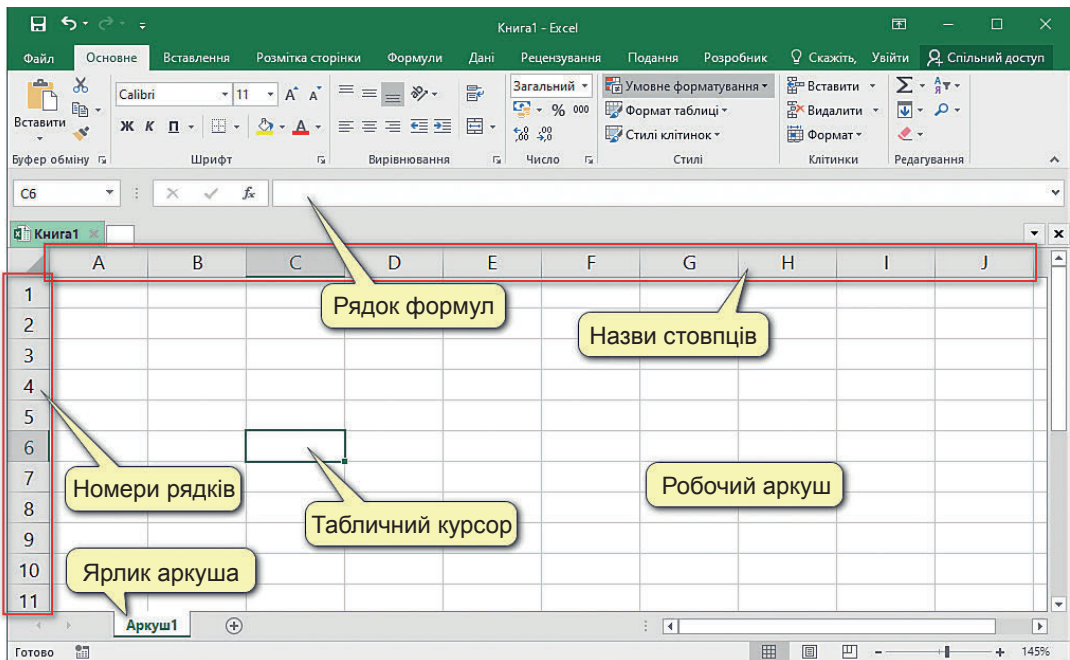


Рис. 2.1. Вікно програми Microsoft Excel

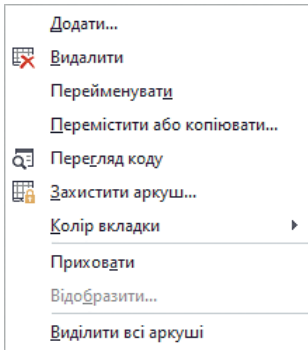


Рис. 2. 

Аркуші можна також видаляти, перейменовувати, переміщувати, копіювати тощо. Усе це можна зробити, обравши контекстне меню відповідного аркуша (рис. 2.2).

Аркуші можна переміщувати, утримуючи ліву клавішу миші, та копіювати, утримуючи ліву клавішу миші та кнопку **Ctrl**.

Робочий аркуш складається із клітинок, для збільшення або зменшення їхнього розміру використовують засоби масштабування, розміщені у нижньому правому куті вікна програми.

Робочий аркуш — це ділянка, поділена вертикальними і горизонтальними лініями на стовпці та рядки.



Рис. 2.3.

Стовпці позначено латинськими літерами **A, B, C, D, ...**. Коли літери латинського алфавіту закінчуються, починається використання подвійної назви, що складається з двох літер, а потім і з трьох. Наприклад, **AA, AB, ABC** тощо. Рядки позначають числами (**1, 2, 3, ...**).

На перетині будь-якого стовпця і рядка розташовано *клітинку*. Дані в електронній таблиці містяться саме в клітинках.

Кожна клітинка в програмі має *свою адресу* (рис. 2.3), яка вказує на те, в якому *стовпці* і в якому *рядку* вона міститься. Наприклад, **C5, F12, K5**.

Щоб визначити місце для введення даних, в Excel використовують *табличний курсор*, який має вигляд рамки темно-зеленого кольору. Табличний курсор позначає одну з клітинок. Виділена клітинка називається *активною*, і саме сюди вводять дані. Щоб зробити активною іншу клітинку (встановити в ній табличний курсор), достатньо підвести вказівник і клацнути по клітинці лівою кнопкою миші. Інший спосіб — перемістити табличний курсор у потрібне місце з допомогою клавіш керування курсором.

Зауважте, що адреса активної клітинки відображається в полі **Ім'я**, а її вміст — у *рядку формул*. Саме тут і будуть відображатись формули, за якими здійснюють обчислення в клітинках.

Зверніть увагу, що формули відображаються в рядку формул, а в клітинці — результат обчислень.

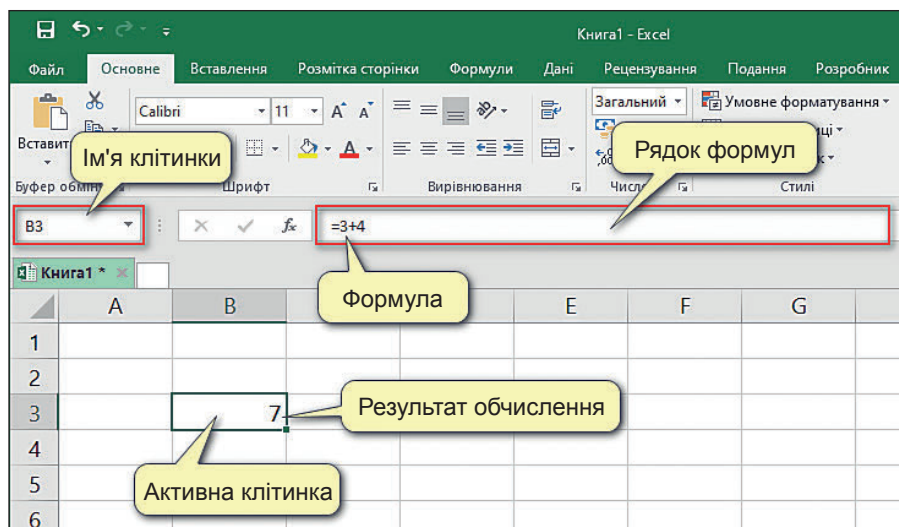


Рис. 2.4.

Так, на рис. 2.4 показано, що в рядку формул відображається формула, що задає додавання чисел 3 і 4, а на робочому аркуші в клітинці відображено результат — 7.

7.2.

УВЕДЕННЯ ДАНИХ

До активної клітинки можна ввести дані трьох видів — це *числа, текст та формули*.

Виділяючи будь-які дані, біля рядка формул з'являються піктограми у вигляді хрестика та галочки: .

Натиснення на галочку означає підтвердження введення значення до клітинки і рівносильне до натиснення клавіші **Enter** на клавіатурі. А при виборі хрестика користувач повідомляє про відміну введення даних, що рівносильне до натискання клавіші **Esc** на клавіатурі.

Щоб користувачу легко було орієнтуватися, з якими даними він має справу, з числами чи текстом, за умовчанням числові дані вирівнюються в клітинці по *правому краю*, а текстові — по *лівому*.

Довжина тексту однієї клітинки може сягати 255 символів. Хоча зовні клітинка є маленькою і весь текст може бути не видно в ній, але він увесь буде відображатися в рядку формул.

Існує декілька способів відображення текстових рядків, більших ніж розміри клітинки.

текст
7

Рис. 2.5

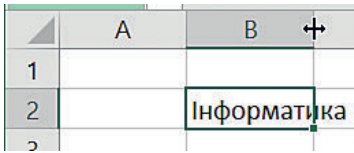


Рис. 2.6



Які ще існують способи перенесення тексту по словах, ви **зможете дізнатися** в електронному додатку.



Щоб розширити клітинку, треба встановити курсор між назвами стовпців і потягнути за стрілку або просто клацнути два рази на межі стовпців.

Дуже довгий текст незручно відображати в один рядок клітинки. Для цього його бажано розташувати у декілька рядків. Щоб розмістити текст у два рядки, необхідно:

1. Виділити клітинку.
2. Обрати вкладку **Основне**.
3. На панелі **Вирівнювання** обрати інструмент **Перенести текст**.

Перенести текст

В Excel доволі часто використовують сукупність суміжних клітинок, яку називають *діапазоном клітинок*.

	A	B	C	D	E	F	G	H	I
1									
2									
3									
4									

Рис. 2.7

Діапазон клітинок позначають адресами двох діагонально протилежних клітинок, розділених символом двокрапки.

Так, на аркуші (рис. 2.7) зараз виділено діапазони **B2:D3** та **G1:H4**.

ВПРАВА 1



Завдання. Створити таблицю відповідно до наведеного зразка (рис. 2.8).

	A	B	C	D
1	Планети Сонячної системи			
2	№	Назва	Діаметр (тис. км)	
3	1	Меркурій	4,87	
4	2	Венера	12,1	
5	3	Земля	12,756	
6	4	Марс	6,67	
7	5	Юпітер	143,76	
8	6	Сатурн	120,42	
9	7	Уран	51,3	
10	8	Нептун	49,5	
11				

Рис. 2.8

1. Запустити на виконання програму Excel.
2. Замінити ім'я робочого аркуша. Для цього клацніть два рази на ярлику аркуша та введіть ім'я **Вправа_1**.

3. Зробіть активною клітинку **A1** і введіть заголовок таблиці — **Планети Сонячної системи**. Натисніть клавішу **Enter**.
4. Уведіть шапку таблиці.
 - а) Клацніть клітинку **A2**, уведіть символ **№** і натисніть на клавішу **→** або **Enter**.
 - б) Уведіть текст **Назва** і натисніть клавішу **→** або **Enter**.
 - в) Уведіть текст **Діаметр (тис. км)** і клацніть клітинку **A3**.
5. Уведіть у клітинки **A3:A10** числа від 1 до 8. Після введення кожного числа натискайте клавішу **Enter**.
6. Клацніть клітинку **B3** і введіть у клітинки **B3:B10** назви планет.
7. Клацніть клітинку **C3** та введіть у клітинки **C3:C10** дані про діаметри планет.
8. Збережіть документ з ім'ям **Вправа_1.xlsx**, він вам ще знадобиться на наступних уроках.

Порада. Якщо ви у якійсь із клітинок припустилися помилки, клацніть у клітинці два рази і зробіть у ній необхідні виправлення, після чого натисніть **Enter**.



Виконати вправу 2 ви зможете в електронному додатку.



Висновки

Табличний процесор — це прикладна програма, яка призначена для опрацювання даних, поданих у таблицях.

Таблиці, які опрацьовує табличний процесор, називають *електронними таблицями*.

Обчислення за допомогою електронних таблиць виконують у програмному засобі Microsoft Excel, що входить до пакета Microsoft Office.

Документ у MS Excel називають *робочою книгою*, оскільки він складається з робочих аркушів.

Робочий аркуш — це ділянка, поділена вертикальними і горизонтальними лініями на стовпці та рядки. Стовпці позначено латинськими літерами, а рядки — числами.

На перетині будь-якого стовпця і рядка розташовано клітинку.

Кожна клітинка у програмі має свою адресу, яка вказує на те, в якому стовпці і в якому рядку вона міститься. Будь-яка клітинка може містити *текст, число, формулу*.

Сукупність виділених клітинок називають діапазоном клітинок.

**Контрольні запитання та завдання**

1. Що таке електронна таблиця?
2. Назвіть елементи головного вікна Microsoft Excel та поясніть їхнє призначення.
3. Які дані може містити клітинка?
4. Що таке адреса клітинки?
5. Що таке діапазон клітинок? Як він записується?

**Питання для роздумів**

- 1*. Назвіть професії, представникам яких для роботи може стати в нагоді програма Excel.
- 2*. Спробуйте визначити, яким буде найбільше значення рядка та стовпця робочої книги MS Excel.

§8.**Редагування електронних таблиць****8.1.****ВИДІЛЕННЯ КЛІТИНОК**

Далі ми познайомимося з тим, як можна редагувати вміст клітинок.

Усі дії з клітинками можна виконувати лише тоді, коли вони виділені. Отже, розглянемо, як виділити клітинки та їхні діапазони.

Для виділення однієї клітинки достатньо зробити її активною, тобто просто клацнути на ній.

Щоб виділити діапазон клітинок, достатньо протягнути по них курсором, утримуючи ліву клавішу миші.

Щоб виділити рядок, треба клацнути на його номері. А щоб виділити стовпець — на його назві.



а)



б)

Рис. 2.9. Виділення: а) рядка; б) стовпця

Щоб виділити сусідні стовпці або рядки, треба протягнути курсором по їхніх назвах, утримуючи ліву клавішу миші.

А щоб виділити всю таблицю, слід клацнути прямокутник, який міститься над номерами рядків та перед назвами стовпців.

Щоб виділити несуміжні клітинки, треба клацати по них, утримуючи клавішу **Ctrl**.



Рис. 2.10. Виділення несуміжних клітинок

8.2.

РЕДАГУВАННЯ ЕЛЕКТРОННИХ ТАБЛИЦЬ

Працюючи з табличним процесором, виникає питання, чи можна використовувати способи копіювання та переміщення даних, які відомі з текстового редактора?

Звичайно, для копіювання потрібно спочатку виділити клітинки і **перемістити їх до буфера обміну** одним із способів:

- за допомогою контекстного меню **Копіювати**;
- клацнути інструмент  **Копіювати** на вкладці **Основне**;
- натиснути комбінацію клавіш **Ctrl+C**.

В результаті застосування однієї із дій копіювання навколо виділеної області з'явиться пунктирна лінія.

Після копіювання вмісту клітинок до буфера обміну необхідно зробити активною верхню ліву клітинку діапазону, куди вставлятимуться дані, та вставити дані з буфера обміну на вказане місце одним із способів:

- за допомогою контекстного меню **Вставити**;
- клацнути інструмент  на вкладці **Основне**;
- натиснути комбінацію клавіш **Ctrl+V**.

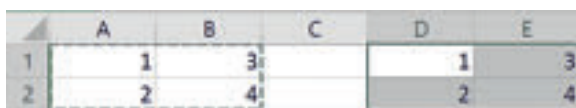


Рис. 2.11

Діапазон, виділений для копіювання, буде мати пунктирну рамку до виконання наступних дій у середовищі Excel або до скасування виділення діапазону. Щоб зняти пунктирну рамку виділення діапазону, треба натиснути клавішу **Esc**.

Якщо ми маємо не скопіювати, а перемістити дані, їх необхідно вирізати до буфера обміну одним із способів:

- за допомогою контекстного меню **Вирізати**;
- клацнути інструмент  **Вирізати** на вкладці **Основне**;
- натиснути комбінацію клавіш **Ctrl+X**.

	Копіювання	Вирізання	Вставлення
Контекстне меню	Копіювати	Вирізати	Вставити
Панель інструментів			
Комбінації клавіш	Ctrl + C	Ctrl + X	Ctrl + V

Рис. 2.12

Область буде залишатися виділеною до переміщення її в бажане місце.

Розглянуті способи копіювання, вирізання та вставлення представлені у таблиці (рис. 2.12).

Розглянемо ще декілька способів редагування вмісту об'єктів, що використовуються в електронних таблицях Excel.

Щоб перемістити виділений діапазон за допомогою миші, потрібно встановити на його межі курсор миші та, коли курсор змінить вигляд на чотиристоронню стрілку , перетягнути діапазон, утримуючи ліву клавішу миші.

Щоб скопіювати діапазон, під час перетягування треба утримувати клавішу **Ctrl**.

Щоб видалити будь-які дані в клітинках, слід виділити їх і натиснути клавішу **Delete**.

Якщо необхідно видалити весь рядок (стовпець), в контекстному меню рядка (стовпця) слід обрати команду **Видалити**.

Щоб вставити рядок над виділеним рядком, необхідно в контекстному меню обрати команду **Додати клітинки**. Ця ж команда дозволяє додати новий стовпець перед виділеним.

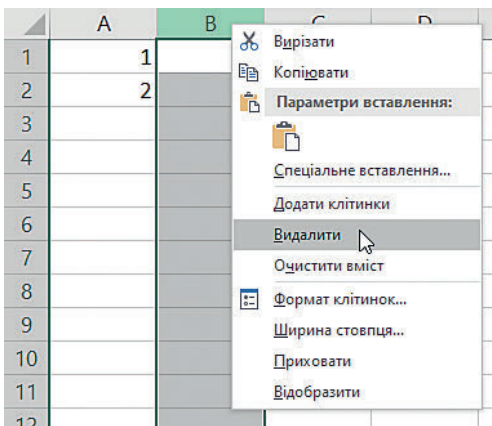


Рис. 2.13. Видалення стовпця B

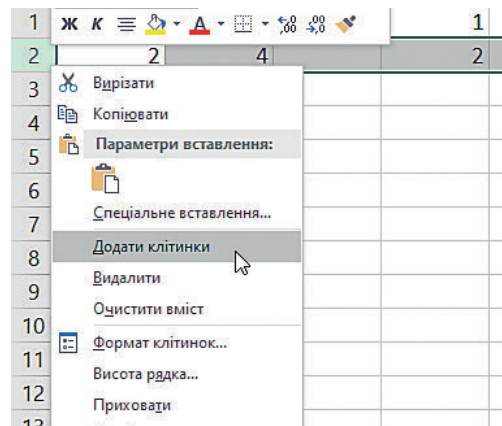


Рис. 2.14. Додавання нового рядка перед другим рядком

8.3.

РЕДАГУВАННЯ ЧАСТИНИ КЛІТИНКИ

Редагування частини тексту, що містить клітинка, здійснюється в рядку формул. Видалення частин тексту здійснюється за допомогою клавіш **Backspace** або **Delete**, як і в будь-якому текстовому редакторі.

Також це можна зробити, клацнувши два рази на необхідній клітинці, після чого у клітинці з'явиться текстовий курсор і текст можна буде змінити.

8.4.

ПОШУК ТА ЗАМІНА ТЕКСТОВИХ ФРАГМЕНТІВ

Іноді виникає необхідність знайти деякий текстовий фрагмент та замінити його на інший.

Так, наприклад, в наведеній нижче таблиці (рис. 2.15) необхідно замінити всі написи, що містять значення «6А» на «7А».



	A	B	C	D	E
	№ особової справи	Прізвище ім'я	Клас	Дата народження	Телефон
2	A1	Антоненко Іван	6А	01.01.2005	063 45 78 932
3	B1	Безпідкова Дарина	6А	02.01.2005	025 47 89 933
4	B2	Белова Богдана	6А	03.04.2005	066 88 99 333
5	B3	Бойцар Максим	6Б	14.05.2005	020 49 90 202
6	B4	Бондар Христина	6А	16.07.2005	029 38 28 282
7	B1	Василевський Максим	6А	06.01.2005	083 02 97 483
8	B2	Ворченко Вікторія	6Б	18.01.2005	090 20 94 859
9	G1	Гуцаленко Кирило	6А	08.01.2005	093 00 22 292
10	D3	Демілін Дмитро	6А	19.01.2005	099 29 99 292
11	D2	Людарино Владислав	6А	10.02.2005	038 83 82 838
12	B1	Зелених Владислав	6Б	11.12.2004	028 88 98 382

	A	B	C	D	E
	№ особової справи	Прізвище ім'я	Клас	Дата народження	Телефон
2	A1	Антоненко Іван	7А	01.01.2005	063 45 78 932
3	B1	Безпідкова Дарина	7А	02.01.2005	025 47 89 933
4	B2	Белова Богдана	7А	03.04.2005	066 88 99 333
5	B3	Бойцар Максим	6Б	14.05.2005	020 49 90 202
6	B4	Бондар Христина	7А	16.07.2005	029 38 28 282
7	B1	Василевський Максим	7А	06.01.2005	083 02 97 483
8	B2	Ворченко Вікторія	6Б	18.01.2005	090 20 94 859
9	G1	Гуцаленко Кирило	7А	08.01.2005	093 00 22 292
10	D3	Демілін Дмитро	7А	19.01.2005	099 29 99 292
11	D2	Людарино Владислав	7А	10.02.2005	038 83 82 838
12	B1	Зелених Владислав	6Б	11.12.2004	028 88 98 382

Рис. 2.15.

Для цього необхідно:

- на вкладці **Основне** обрати інструмент **Знайти й виділити** → **Замінити**;
- у полі **Знайти** ввести текст **6А** у полі **Замінити на** — **7А**;
- натиснути кнопку **Замінити все**.

У результаті всі клітинки, що мали напис **6А**, буде замінено на **7А**.

ВПРАВА 3



Завдання. Виконати редагування даних у таблиці **Прайс-лист**.

1. Завантажте файл-заготовку до вправи **Прайс.xlsx**.
2. На аркуші **Прайс-лист** видаліть рядки із товаром, що відсутній на складі.
3. Змініть деякі написи у стовпці **Найменування**, переклавши їх українською мовою.
4. Ціну всіх магнітів збільшіть на 10 грн.

5. Додайте ще один стовпець після стовпця **Е** та надайте йому назву **Замовлення**.
6. Запишіть довільним чином значення у стовпці **Замовлення**, але не більше, ніж є на складі.
7. Збережіть файл з ім'ям **Вправа_3.xlsx**.



Висновки

Щоб виконати будь-які дії з даними електронних таблиць, їх необхідно виділити.

Виділення однієї клітинки здійснюється клацанням по ній, суміжні клітинки виділяються з утриманням лівої клавіші миші.

Для виділення стовпця або рядка необхідно клацнути відповідну назву стовпця або номер рядка. Виділення діапазону стовпців чи рядків здійснюється з утриманням лівої клавіші миші.

Для виділення несуміжних клітинок слід утримувати клавішу **Ctrl**.

Для редагування даних у таблицях використовують стандартні способи переміщення виділеного фрагмента до буфера обміну та подальшого вставлення (за допомогою контекстного меню, інструментів лінійки **Основне** та комбінацій клавіш **Ctrl+C**, **Ctrl+X**, **Ctrl+V**).



Контрольні запитання та завдання

1. Укажіть відомі вам способи копіювання та переміщення даних з одного діапазону клітинок до іншого.
- 2*. Опишіть способи виділення стовпців, рядків таблиці, суміжних та несуміжних клітинок.
3. Як видалити стовпець або рядок?
4. Як вставити новий стовпець або рядок?
3. Як виправити помилку в тексті клітинки?
4. Як здійснити пошук клітинки за її вмістом?



Питання для роздумів

- 1*. Як поміняти місцями вміст двох стовпців? Дані стовпця **А** розмістити у стовпці **В**, а дані стовпця **В** — у стовпці **А**.
- 2*. В таблиці є відомості про учнів та учениць 6-А та 7-А класу. Оскільки учні та учениці перейшли до наступного класу, всі написи 6-А необхідно замінити на 7-А, відповідно 7-А замінити на 8-А. Опишіть алгоритм виконання цих дій.



§9.

Форматування електронних таблиць

9.1.

ПОНЯТТЯ ФОРМАТУВАННЯ

Щоб інформація в таблиці була зручною для використання, можна змінити її зовнішній вигляд, тобто відформатувати.

На рисунку зображено таблицю, що містить дані електронного журналу учнів/учениць 7-А класу. Проте в даному вигляді вона не дуже зручна для сприйняття.

Рис. 2.16

№	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W												
1			Служби інтернету										Організація табличних даних																						
2	№	Прізвище Ім'я	02.09	09.09	16.09	23.09	30.09	06.10	13.10	20.10	27.10	тематична													03.11	10.11	17.11	24.11	01.12	08.12	15.12	22.12	29.12	тематична	І семестр
3	1	Зеленко Владислав	10				8				11	10	н	н	8			8			8	8	9												
4	2	Коваль Ана	10				9				10	10			9			8			10	9	10												
5	3	Мухомов Максим	10				7	н	н	н		10	9			7		7	7		7	7	8												
6	4	Овчаренко Аліна	н	11			10				9	10			10			10			10	10	10												
7	5	Петровченко Діана		11				8			11	10			8			8			8	8	9												
8	6	Полова Єлизавета	н	9			10				8	9	н	н	10			10			10	10	10												
9	7	Проданко Руслан		10			н	н			10	10			н			7			9	8	9												
10	8	Самойлов Ілля		10			9				9	9			9			9			9	9	9												
11	9	Станка Валерій		10		10	10				8	10	10		8			10			10	9	10												
12	10	Степанюкова Іванна		10			8				8	9			8			8			8	8	9												
13	11	Титок Ігор		11			8				10	10	н	н	10			9			8	9	10												
14	12	Холост Артем		11			9				10	10			9			9		8	9	9	10												
15	13	Шендрик Олег		10			10				9	10	н	н	10			10			10	10	10												
16	14	Широко Дмитро		11			11				10	11			11			11			11	11	11												
17	15	Шумілов Владислав		10			10				10	10			10			10			10	10	10												
18			пр1			пр2					к/р				пр3			пр4				к/р													
		ТА																																	

Значно зручніше переглядати інформацію у таблиці, яка є відформатованою, ось, наприклад, як таблиця з подібними даними електронного журналу учнів/учениць 7-Б класу.

Рис. 2.17

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W		
1			Служби Інтернету										тематична	Опрацювання табличних даних										тематична	І семестр
2	№	Прізвище Ім'я	02.09	09.09	16.09	23.09	30.09	06.10	13.10	20.10	27.10	03.11		10.11	17.11	24.11	01.01	08.12	15.12	22.12	29.12				
3	1	Антоненко Іван		10			8				11	10	н	н	8			8			8	8	9	9	
4	2	Безніцька Дарина		9			10				10	10			9			8			10	9	10	10	
5	3	Бельза Богдана		9			7	н	н	н	10	9			7		7	7			7	7	8	8	
6	4	Бойправ Максим	н	11			9				9	10			10			10			10	10	10	10	
7	5	Бондар Христина		11			8			6	11	9			8			8			8	8	9	9	
8	6	Василевський Микита	н	9			10				8	9	н	н	10			10			10	10	10	10	
9	7	Вовченко Вікторія		10			н	н			10	10			н			7			9	8	9	9	
10	8	Гульченко Кирило		10			9				9	9			9			9			9	9	9	9	
11	9	Данілін Дмитро		10			10				10	10	6		10			10			10	10	10	10	
12	10	Демченко Ілля		10			8				8	9			8			8			8	8	9	9	
13	11	Дударенко Владислав		9			8				10	9	н	н	10			9			8	9	9	9	
14	12	Килимчук Денис		9			9				10	9			9			9		8	9	9	9	9	
15	13	Койдан Анастасія		10			10				9	10	н	н	10			10			10	10	10	10	
16	14	Кондратюк Дмитро		11			7				10	9			11			11			11	11	10	10	
17	15	Мосійчук Анатолій		10			10				10	10			10			10			10	10	10	10	
18	16	Самойлов Владислав		9			8				7	8			6			4			8	6	7	7	
19			пр1			пр2					к/р				пр3			пр4				к/р			
		75																							

Далі розглянемо різні способи здійснення форматування в електронних таблицях Excel.

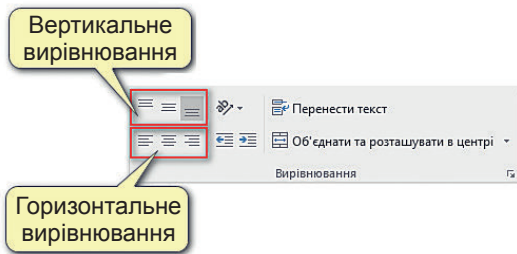


Рис. 2.18

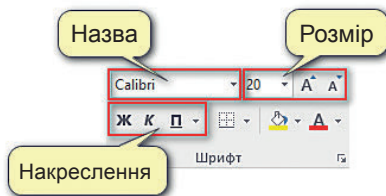


Рис. 2.19

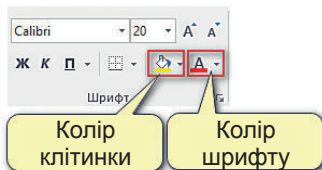


Рис. 2.20

Раніше ми дізналися, що для зручності сприйняття даних в Excel за умовчанням числа вирівнюються по правому краю, а текст — по лівому.

При необхідності вирівнювання як числових, так і текстових даних можна змінити, використовуючи інструменти вирівнювання, що містяться в області **Вирівнювання** на вкладці **Основне**.

Під форматуванням також розуміють зміну параметрів шрифту: назву, розмір, накреслення, які можна змінити в області **Шрифт** вкладки **Основне**.

Звичайно, в клітинках можна змінити колір фону та колір шрифту. Це можна зробити, застосувавши відповідні інструменти в області **Шрифт** на вкладці **Основне**.

9.2.

ВСТАНОВЛЕННЯ МЕЖ

Для відображення меж клітинок використовують інструмент **Межі**, розташований в області **Шрифт** вкладки **Основне**. На рисунку показано найбільш уживані способи використання інструменту **Межі**.



Рис. 2.21

Цей інструмент надає можливості не тільки встановлення контурів клітинок, а також налаштування кольору меж та їхніх

стилів. Усі можливості цього інструменту можна отримати, відкривши його меню.

Усі необхідні параметри меж можна встановити у відповідному діалоговому вікні **Формат клітинок** на вкладці **Межі**, яке викликається з допомогою команди **Додаткові межі**.

Якщо виникає необхідність об'єднати клітинки, використовують інструмент **Об'єднати та розташувати в центрі**. В результаті клітинки буде об'єднано в одну клітинку, а вміст клітинки розташовано по центру. Якщо до об'єднання у клітинках було розміщено дані, то після об'єднання у клітинці залишиться вміст лише клітинки, яка була розміщена у верхньому лівому куті до дії об'єднання. Також об'єднана клітинка буде мати її адресу.

В меню інструменту **Об'єднати та розташувати в центрі** можна обрати більше варіантів об'єднання клітинок (рис. 2.23).

Щоб змінити напрямок розташування тексту, достатньо виділити текст та обрати інструмент **Орієнтація**, після чого обрати бажаний напрямок.

Вибір команди **Формат вирівнювання клітинки** призведе до відкриття діалогового вікна, в якому налаштування вирівнювання тексту можна зробити точнішими.

Щоб кожного разу не встановлювати декілька параметрів форматування для різних клітинок, зручно скористатися копіюванням формату за допомогою інструменту

 **Формат за зразком** **Формат за зразком**.

Для цього спочатку слід виділити клітинку, в якій уже встановлено потрібний формат, клацнути на інструмент  **Формат за зразком** **Формат за зразком** (після чого курсор набуде вигляду пензлика) та виділити клітинки, до яких треба застосувати бажаний формат.

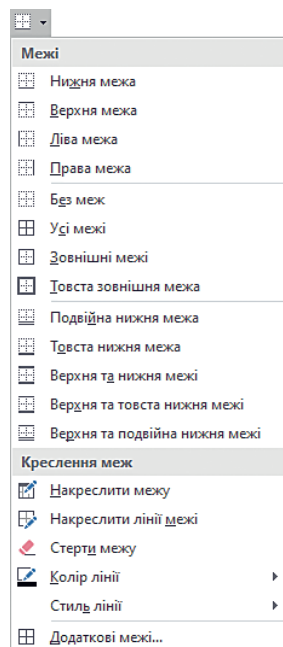


Рис. 2.22. Меню інструменту **Межі**

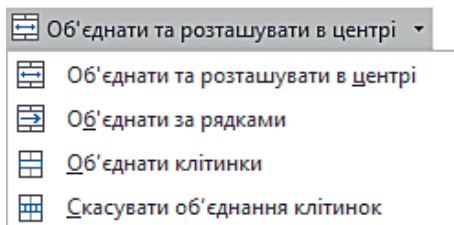


Рис. 2.23. Меню інструменту **Об'єднати та розташувати в центрі**

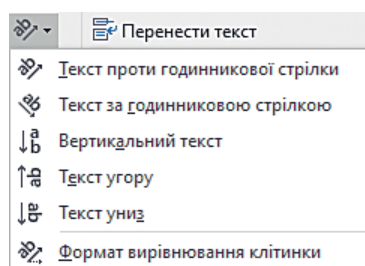


Рис. 2.24. Меню інструменту **Орієнтація**

ВПРАВА 4



Завдання. Отримати таку таблицю, як показано на рисунку.

	A	B	C	D
1	Планети Сонячної системи			
2	№	Назва	Діаметр (тис. км)	
3	1	Меркурій	4,87	
4	2	Венера	12,1	
5	3	Земля	12,756	
6	4	Марс	6,67	
7	5	Юпітер	143,76	
8	6	Сатурн	120,42	
9	7	Уран	51,3	
10	8	Нептун	49,5	
11				

1. Відкрийте файл **Вправа_1.xlsx**.
2. Розташуйте заголовок по центру таблиці. Для цього виділіть діапазон клітинок **A1:C1** і оберіть інструмент **Об'єднати та розташувати в центрі**.
3. Виділіть заголовок таблиці жирним шрифтом та збільшіть розмір шрифту. Для цього клацніть на заголовку, а потім — на інструменті **Ж** на лінійці **Основне** в області **Шрифт**, у цій же області змініть значення шрифту, встановивши значення **14пт**.
4. Виконайте форматування шапки таблиці.

а) Виділіть клітинки **A2:C2** і клацніть кнопки **Ж** та **К**.

б) Клацніть стрілку на кнопці і виберіть з палітри жовтий колір.

в) Вирівняйте текст у клітинках заголовка по центру, натиснувши послідовно кнопки та .

5. Змініть ширину стовпців та розташування тексту.

а) Клацніть на клітинці **C2** та оберіть інструмент **Перенести текст** на вкладці **Основне** в області **Вирівнювання**.

б) Змініть ширину стовпців: перший стовпець звузьте, а другий — розширте. Ширину третього встановіть такою, щоб заголовок стовпця розмістився у два рядки.



Виконати вправу 5 ви зможете в електронному

додатку.



6. Задайте для таблиці межі. Виділіть усі її клітинки (це діапазон **A2:C10**), клацніть на стрілочку біля кнопки та оберіть варіант **Усі межі**.

7. Збережіть файл з ім'ям **Вправа_4.xlsx**.

ВПРАВА 6



Завдання. Виконавши завдання цієї вправи, здійснити форматування даних електронного журналу для учнів/учениць 7-А класу відповідно до зразка таблиці для учнів 7-Б класу (рис. 2.17).

1. Відкрийте файл **Журнал.xlsx**.
2. Виділіть діапазон **C1:K1** та застосуйте до нього інструмент **Об'єднати та розташувати в центрі**.
3. Аналогічно виконайте об'єднання клітинок діапазонів **M1:U1**, **V1:V2** та **W1:W2**.
4. Змініть напрям тексту у клітинках **L1**, **V1** та **W1**, обравши відповідну команду в меню інструменту **Орієнтація**.



5. Зменшіть ширину стовпців **L**, **V** та **W** та збільшіть висоту рядка **2**.
6. Встановіть, де це необхідно, напівжирний шрифт тексту.
7. Встановіть червоний колір літер для назв тем.
8. Встановіть межі для діапазону клітинок **A1:W17**.
9. Для стовпців **№**, **Прізвище Ім'я**, **Тематична**, **I семестр** установіть зовнішню межу, обравши команду **Товста зовнішня межа** в меню інструменту  **Межі**.
10. Встановіть заливку клітинок різними кольорами відповідно до зразка.
11. Збережіть файл з ім'ям **Вправа_6.xlsx**.



Висновки

Форматування шрифту та вирівнювання тексту в клітинках здійснюється на вкладці **Основне** в області **Шрифт** та **Вирівнювання** відповідно.

Встановлення кольорів здійснюється інструментами **Колір заливки** та **Колір шрифту**.

Встановлення та відміна меж клітинок здійснюється за допомогою інструменту **Межі**.

Для об'єднання клітинок використовують інструмент **Об'єднати та розташувати в центрі**.



Контрольні запитання та завдання

1. Які параметри форматування можна встановити для шрифту?
2. Які параметри форматування можна встановити для клітинки?
3. Яку адресу буде мати клітинка після об'єднання діапазону клітинок?
4. Що відбудеться з даними діапазону клітинок після їхнього об'єднання до однієї клітинки?



Питання для роздумів

Назвіть спільні та відмінні команди форматування даних у середовищах текстового редактора MS Word і табличного процесора MS Excel.

§10.

Формати даних

10.1.

ОСНОВНІ ФОРМАТИ

Як було зазначено раніше, клітинка електронної таблиці може містити текст, число або формулу. Excel автоматично розпізнає тип даних і здійснює вирівнювання тексту по лівому краю, а числа — по правому краю клітинки. Проте користувач може і самостійно вказати програмі, який формат повинна мати клітинка.

Числові дані в Excel можуть бути представлені в різних форматах:

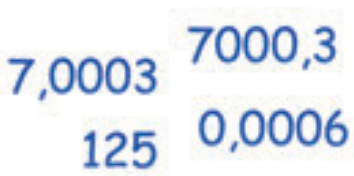
- числовий;
- грошовий;
- дата;
- час;
- відсотковий.

Текстові дані представлені текстовим форматом.

Для зміни формату даних слід відкрити діалогове вікно в області **Число** вкладки **Основне** або обрати команду **Формат клітинок** у контекстному меню клітинки.

В результаті буде відкрито вікно **Формат клітинок**, у якому відображаються можливі формати даних і за умовчанням встановлено **Загальний формат**. При виборі одного із форматів буде змінено формат даних клітинки.

Розглянемо детальніше основні формати.

Числовий формат

7,0003 7000,3
125 0,0006

Рис. 2.25. Приклади даних у числовому форматі

Числовий формат призначений для відображення чисел із заданою кількістю десяткових знаків. Це можуть бути як числа, що містять дробову частину, так і числа, що не містять її.

Щоб змінити кількість знаків, що буде встановлена після коми, необхідно відкрити вкладку **Число** діалогового вікна **Формат клітинок**, змінити кількість знаків після коми в полі **Кількість десяткових розрядів**.

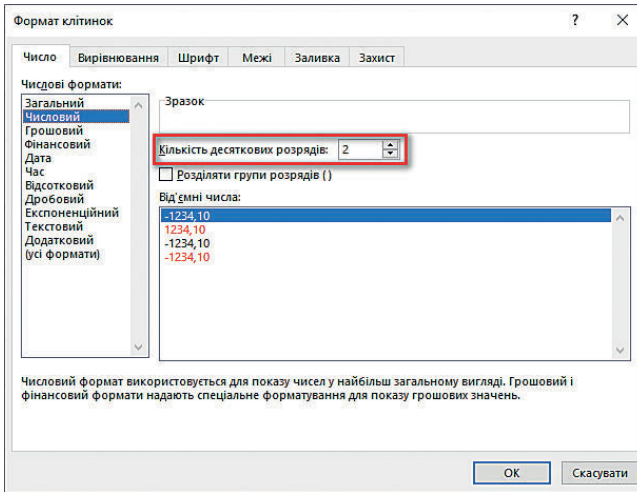


Рис. 2.26. Зміна параметрів числового формату

Грошовий формат

Встановлення грошового формату дозволяє представити грошові дані: ціну, вартість, зарплату тощо із вказуванням грошової одиниці.

У діалоговому вікні **Формат клітинок** можна змінити кількість знаків, що буде встановлена після коми, та тип валюти у полі **Позначення**.



Рис. 2.27. Приклади даних у грошовому форматі

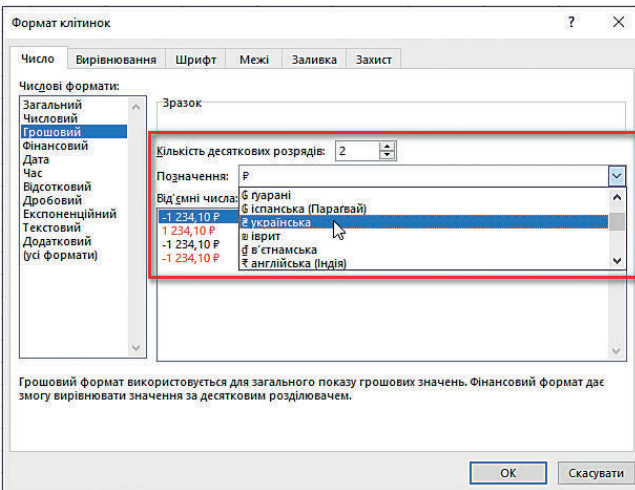


Рис. 2.28. Зміна параметрів грошового формату

Символ гривні (₴) — коротке позначення української гривні. Графічний символ гривні складається з двох елементів. Перший і основний елемент ґрунтується на рукописному варіанті букви «Г» — вона позначає першу букву в назві «гривня». Другим елементом є дві горизонтальні паралельні лінії, які втілюють ідею стабільності грошової одиниці, високий потенціал і зростання національної економіки України. Подібна ідея традиційно використовується для позначення інших валют.

Формати дата та час



Рис. 2.29. Приклади даних у форматі **Дата**



Рис. 2.30. Приклади даних у форматі **Час**

Для відображення дати або часу використовують формат **Дата**. Він може містити тільки дату в різних формах подання, так і дату з часом.

Формат **Час** містить лише час в різних формах подання: скороченій формі, з указуванням часу до полудня або після полудня тощо.

Зазвичай при введенні даних із розділенням їх точкою після кожних двох коректно введених цифр Excel самостійно визначає, що має справу з датами.

Excel надає широкі можливості встановлення різних видів подання дати і часу. Бажаний вид дати можна обрати в діалоговому вікні **Формат клітинок** у полі **Тип**.

Відсотковий формат



Рис. 2.31. Приклади даних у відсотковому форматі

Відсотковий формат використовується для відображення значень у вигляді відсотків (число 1 у клітинці буде визначено, як 100%).

Зауважимо, якщо для значення клітинки буде встановлено відсотковий формат, то число, відображене у клітинці, буде автоматично збільшено у 100 разів. Так, якщо клітинка містила значення 0,5, при встановленні відсоткового формату в клітинці буде відображено 50% (рис. 2.32).

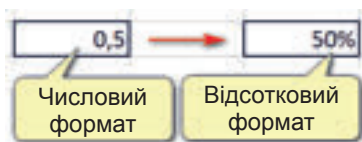


Рис. 2.32.

Звичайно, коли клітинка вже містить відсотковий формат, то необхідно вводити такі числові дані, які ми бажаємо представити у відсотках. Так, для попереднього прикладу це буде число 50.

Текстовий формат



Рис. 2.33. Приклади даних у текстовому форматі

Якщо клітинка містить текст або дані, що не вимагають обчислень, їх можна представляти в текстовому форматі.

У текстовому форматі значення клітинок інтерпретуються як текст, навіть коли вони містять числа.

Так, іноді є необхідність числову інформацію представити у вигляді текстової.

Якщо ви спробуєте ввести номер телефону будь-якого клієнта у форматі, до якого більшість із нас звикла, наприклад, 068 845 55 44, Excel розпізнає ці дані як число і 0 прибере. Тобто залишиться напис 68 845 55 44. Щоб цього не відбувалося, для цієї клітинки необхідно встановити текстовий формат. Тепер після введення числа, що починається на 0, усі цифри буде збережено.

Зауваження. Щоб число було сприйняте програмою як текст, установіть перед ним символ апострофа. У клітинці апостроф відображатися не буде, а число буде вирівняно до лівого краю, що свідчить про те, що в клітинці — дані текстового типу.

Основні формати даних з їхнім описом та прикладами подані у зведеній таблиці, вміщеній на форзаці книги.

10.2.

ЗМІНА ФОРМАТУ НА ПАНЕЛІ ІНСТРУМЕНТІВ

Оскільки в електронних таблицях часто доводиться вводити числові значення, грошові суми та відсотки, для швидкого вибору цих поширених форматів, а також для зміни розрядності чисел у клітинках у вкладці **Основне** на панелі **Число** передбачено спеціальні кнопки.

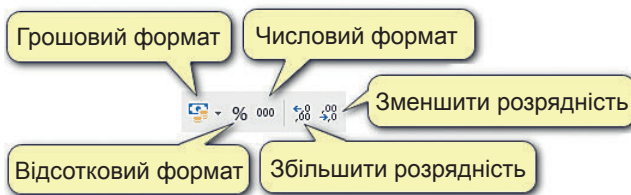


Рис. 2.34. Інструменти встановлення формату

Також для швидкої зміни формату чисел можна скористатися списком, що є на панелі **Число**, обравши у ньому один із розповсюджених форматів.

ВПРАВА 7



Завдання. Заповнити таблицю (рис. 2.35) даними та встановити необхідні числові формати відповідно до зразка.

1. Уведіть необхідні дані до таблиці.
2. Застосуйте до діапазону **C2:C9** формат **Дата**:
 - виділіть діапазон **C2:C9**;
 - відкрийте діалогове вікно **Формат клітинок**;
 - у вкладці **Число** оберіть числовий формат **Дата**;
 - оберіть потрібний тип дати в полі **Тип**.

3. Застосуйте до діапазону **D2:D9** формат **Грошовий**, додатково встановлюючи необхідну валюту та кількість знаків після коми.

	A	B	C	D	E
1	№	Робітник	Дата прийому на роботу	Заробітна плата, грн	Нараховано премії, % від зарплати
2	1	Сергієнко І.С.	01.12.10	4 750₴	20%
3	2	Петриченко Г.П.	12.03.11	2 500₴	
4	3	Гаврилов П.С.	17.01.11	4 900₴	15%
5	4	Іванченко С.Г.	18.09.09	3 700₴	10%
6	5	Кириченко Х.П.	01.07.08	6 500₴	25%
7	6	Степанов І.Г.	24.01.06	3 220₴	10%
8	7	Казаков Б.Ф.	09.10.10	4 050₴	
9	8	Тарасенко П.Х.	11.11.11	3 015₴	20%

Рис. 2.35

4. Застосуйте до діапазону **E2:E9** формат **Відсотковий**. За необхідності змініть налаштування формату в діалоговому вікні **Формат клітинок** або за допомогою лінійки інструментів.



Виконати вправу 8 ви зможете в електронному додатку.



5. Виконайте необхідне форматування таблиці (виділення заголовка жирним шрифтом, заливку клітинок заголовка сірим кольором, перенесення по словах деяких клітинок заголовка, зміну ширини стовпців).

6. Збережіть документ з ім'ям **Вправа_7.xlsx**.



Висновки

Користувач може самостійно визначити, який формат повинні мати дані в клітинці. Це можна зробити за допомогою діалогового вікна **Формат клітинок** у вкладці **Число**.

Числові дані мають такі основні формати: числовий, дата, час, грошовий, відсотковий.

Встановлення основних параметрів форматів даних можна здійснювати в області **Число** на вкладці **Основне**.



Контрольні запитання та завдання

1. Які формати чисел підтримує MS Excel?
2. Опишіть послідовність дій для встановлення такого формату «0,00 ₴».
3. При введенні до клітинки дати 01.01.19 відображається 01.01.2019. Як для цієї клітинки встановити формат «01.01.19»?



Питання для роздумів

1. Як установити користувацький формат числа?
- 2*. Спробуйте визначити, як записати в Excel числа в такому вигляді:
12,56 грн; 12 °C.
- 3**. Як з допомогою діалогового вікна **Формат клітинок** створити такий формат, щоб додатні числа виводилися червоним кольором, а від'ємні — синім?

§11.

Автозаповнення та автозавершення

11.1.

ПОНЯТТЯ АВТОЗАПОВНЕННЯ, МАРКЕР АВТОЗАПОВНЕННЯ

У програмі Excel є ще один зручний спосіб копіювання — автозаповнення.

Цей спосіб має кілька секретів. Щоб скопіювати таким чином, достатньо навести вказівник на нижній правий кут клітинки (курсор має набути вигляду тонкого чорного плюса, який називають маркером автозаповнення) і перетягнути його в потрібне місце.

Таким способом зручно і швидко можна створювати різноманітні списки.



Рис. 2.36

11.2.

СТВОРЕННЯ СПИСКУ ЧИСЕЛ ВІД 1 ДО 10

Розглянемо створення списку послідовності чисел від 1 до 10 з використанням автозаповнення.

1. У клітинці **A1** уведіть число 1, у клітинці **A2** — число 2.
2. Виділіть клітинки **A1:A2**.
3. Встановіть курсор на нижньому правому куті клітинки **A2** та перетягніть його вниз до клітинки **A10**, утримуючи натисненою ліву клавішу миші.

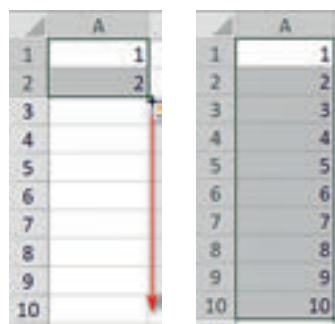


Рис. 2.37

Аналогічно можна створити список чисел по горизонталі.

1. У клітинці **A1** уведіть число 1, у клітинці **B1** — число 2.
2. Виділіть клітинки **A1:B1**.

3. Встановіть курсор на нижньому правому куті клітинки **B1** та перетягніть його праворуч до клітинки **J1**, утримуючи натисненою ліву клавішу миші.

	A	B	C	D	E	F	G	H	I	J
1	1	2								
2										

Рис. 2.38

	A	B	C	D	E	F	G	H	I	J
1	1	2	3	4	5	6	7	8	9	10

Рис. 2.39

11.3.

СТВОРЕННЯ ЧИСЛОВОЇ ПОСЛІДОВНОСТІ

	A	B	C	D	E	F
1		2		25		0,3
2		4		20		0,5
3		6		15		0,7
4		8		10		0,9
5		10		5		1,1

Рис. 2.40

Ми розглянули, як створити список із послідовності чисел, де кожне наступне число є більшим на 1 від попереднього. Так само можна утворити будь-яку числову послідовність, у якій кожне наступне число буде більшим або меншим від попереднього на якесь числове значення (рис. 2.40).

	A
1	2
2	4

Рис. 2.41

	A
1	2
2	4
3	6
4	8
5	10

Рис. 2.42

Розглянемо послідовність дій щодо створення такого вигляду послідовностей на першому прикладі (2, 4, 6, 8, 10).

1. Встановіть початкове значення послідовності в першій клітинці, а наступне — у другій.
2. Виділіть обидві клітинки.
3. Встановіть вказівник до нижнього правого кута другої клітинки.
4. Протягніть вказівник вниз до потрібної клітинки, утримуючи натисненою ліву кнопку миші.

Аналогічно можна створити послідовність і по горизонталі.

Розглянемо ще один спосіб створення таких послідовностей.

1. Уведіть перші два числа послідовності.
2. Виділіть діапазон, у якому плануєте створити послідовність, наприклад, **A1:A5**.
3. На вкладці **Основне** оберіть в меню інструменту **Заповнити** команду **Прогресія**. В результаті буде діалогове ві-

кно **Прогресія** (у цьому вікні можна створювати різного вигляду послідовності).

- У діалоговому вікні оберіть команду **автозаповнення**.

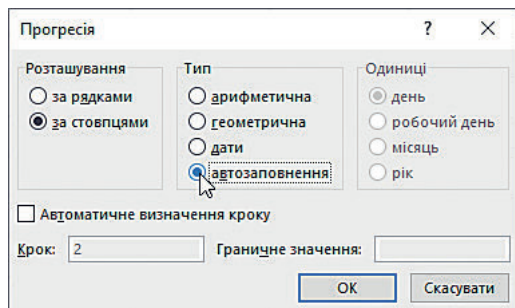


Рис. 2.45

- Натисніть кнопку **OK**. При цьому програма самостійно визначить інтервал зміни чисел і заповнить виділений раніше діапазон.

	A
1	2
2	4
3	
4	
5	

Рис. 2.43

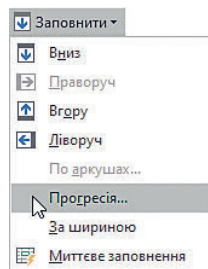


Рис. 2.44

	A
1	2
2	4
3	6
4	8
5	10

Рис. 2.46

11.4.

СТВОРЕННЯ СПИСКУ З ЧИСЛАМИ ТА НАПИСАМИ

В Excel також існує можливість створення списків типу:

- 1) A1, A2, A3 ...;
- 2) B2, B4, B6 ...

Щоб створити список першого типу, достатньо ввести лише першу частину, що містить текст і якесь число, після чого застосувати до цієї клітинки автозаповнення.

Для створення списку другого типу потрібно заповнити значеннями дві клітинки, аби Excel установив різницю між двома числами.

Зверніть увагу, що для створення числової послідовності разом із текстом число має стояти після тексту, але не навпаки.

Так, при застосуванні дії автозаповнення до напису **1A** відбудеться лише його копіювання. Якщо ввести **1A, 2A**, то копіювання відбудеться через рядок (рис. 2.47).

З цих прикладів неважко здогадатися, що спроби застосувати автозаповнення лише до тексту призведуть до його копіювання.

Проте створювати списки, що містять текст, все ж таки можна. Так, в Excel існує ряд вбудованих списків. Це дні

	A	B	C
1	1A		1A
2	1A		2A
3	1A		1A
4	1A		2A
5	1A		1A
6	1A		2A

Рис. 2.47

	A
1	Excel
2	Excel
3	Excel

Рис. 2.48

тижнів, назви місяців та їхні скорочення. Але зверніть увагу, що ці списки представлені у даній версії Excel лише на російській мові.

Для створення такого виду списку достатньо в першій клітинці ввести будь-яке значення у списку і застосувати до нього автозаповнення.

	A	B	C	D
1	янв	январь	пн	понедельник

Рис. 2.49

	A	B	C	D
1	янв	январь	пн	понедельник
2	фев	февраль	вт	вторник
3	мар	март	ср	среда
4	апр	апрель	чт	четверг
5	май	май	пт	пятница
6	июн	июнь	сб	суббота
7	июл	июль	вс	воскресенье
8	авг	август	пн	понедельник
9	сен	сентябрь	вт	вторник
10	окт	октябрь	ср	среда
11	ноя	ноябрь	чт	четверг
12	дек	декабрь	пт	пятница

Рис. 2.50

Після останнього елемента списку буде відображатися знову перший елемент. Це добре видно у випадку створення списку днів тижнів.

11.5.

СТВОРЕННЯ ВЛАСНОГО СПИСКУ

Відсутність списків днів тижнів та назв місяців українською мовою не є великою проблемою, оскільки списки можна створити і самостійно. Наведемо алгоритм створення власного списку.

1. Перейдіть на вкладку **Файл** → **Параметри** → **Додатково**.
2. В області **Загальні** оберіть кнопку **Редагувати користувачькі списки**. У вікні, що відкриється, буде відображено усі списки, що використовуються за умовчанням.
3. Для створення власного списку в полі **Елементи списку** введіть дні тижнів, натискаючи **Enter** після введення кожного елемента списку (рис. 2.51).
4. Натисніть кнопку **Додати**. В результаті новий список з'явиться у полі **Списки**.
5. Натисніть два рази **ОК**.
6. В електронній таблиці до клітинки введіть слово із списку, наприклад, **Понеділок**.



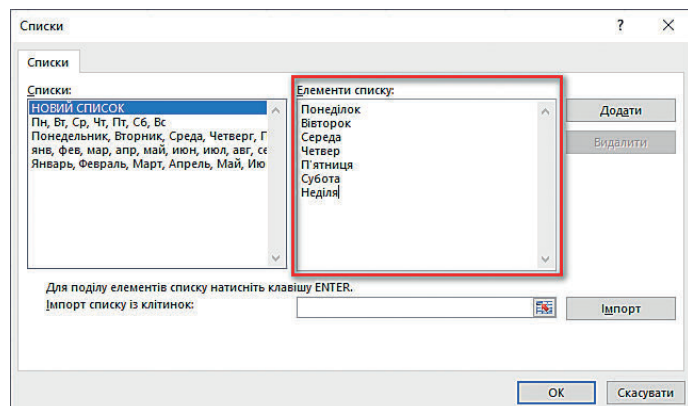


Рис. 2.51

7. Поверніться до клітинки зі словом **Понеділок** та застосуйте до неї автозаповнення. В результаті буде відображено список днів тижнів українською мовою, а саме, список, створений користувачем.

Звичайно, у разі допущення помилки в написанні деяких елементів списку їх у будь-який час можна легко відредагувати у цьому ж вікні. Також створені списки можна видалити, натиснувши кнопку **Видалити**. Зауважимо, що списки, які є вбудованими у середовище, видалити неможливо.

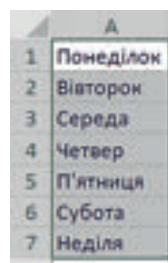


Рис. 2.52

11.6.

АВТОЗАВЕРШЕННЯ

Ще одна зручна можливість в Excel — це *автозавершення*. Використовуючи автозавершення, в Excel можна легко вводити один і той же текст в декілька клітинок. Для цього достатньо ввести декілька перших літер до клітинки, і Excel автоматично завершить її, спираючись на інші записи, які до цього було зроблено у стовпці.

Так, наприклад, нами було вже створено таблицю з переліком предметів для здачі ЗНО із наступними даними (рис. 2.53). Необхідно заповнити інші клітинки таблиці.

	A	B	C	D	E
1	Учень	I предмет	II предмет	III предмет	IV предмет
2	Мазуренко М.	Українська мова та література	Математика	Хімія	Географія
3	Мосійчук В.	Українська мова та література	Історія України	Біологія	Фізика
4	Номерюк І.				
5	Пісковцов А.				

Рис. 2.53



Про ще один спосіб створення власних списків ви **можете дізнатися** в електронному додатку.



Звичайно, при заповненні стовпчика **B** проблем не виникне, адже його значення завжди є однаковими і достатньо скопіювати вміст клітинки **B3** до інших клітинок, використовуючи автозаповнення. У стовпчику **C** клітинки містять одне з уведених в цьому стовпці значень (Математика або Історія України). Для введення наступних значень достатньо розпочати введення предмета, після чого Excel розпізнає слово за початковими буквами та запропонує завершити слово, використовуючи наявні вище слова (рис. 2.54). Для завершення введення слова достатньо натиснути **Enter**.

	A	B	C	D	E
1	Учень	I предмет	II предмет	III предмет	IV предмет
2	Мазуренко М.	Українська мова та література	Математика	Хімія	Географія
3	Мосійчук В.	Українська мова та література	Історія України	Біологія	Фізика
4	Номерюк І.		математика		
5	Пісковцов А.				

Рис. 2.54

До того ж автозавершення також змінює автоматично регістр літер. Якщо у списку вище слово розпочиналося з великої літери «М», а ми будемо вводити з маленької, то після застосування автозавершення слово буде введено з великої літери (рис. 2.55).

	A	B	C	D	E
1	Учень	I предмет	II предмет	III предмет	IV предмет
2	Мазуренко М.	Українська мова та література	Математика	Хімія	Географія
3	Мосійчук В.	Українська мова та література	Історія України	Біологія	Фізика
4	Номерюк І.		Математика		
5	Пісковцов А.				

Рис. 2.55

До списку попередніх слів можна також отримати доступ, обравши в контекстному меню клітинки команду **Вибрати з розкривного списку**. В результаті Excel відобразить список усіх слів (без повторень), що містить даний стовпець (рис. 2.56). Для заповнення даними з цього списку достатньо обрати один з елементів списку. Даний список можна викликати, також натиснувши **Alt+↓**.

Якщо стовпець містить декілька записів, які відповідають першим декільком символам, Excel не запропонує автозавершення, доки новий запис не буде відповідати одному з попередніх однозначним чином (рис. 2.57).

	A	B	C	D	E
1	Учень	I предмет	II предмет	III предмет	IV предмет
2	Мазуренко М.	Українська мова та література	Математика	Хімія	Географія
3	Мосійчук В.	Українська мова та література	Історія України	Біологія	Фізика
4	Номерюк І.		Математика		
5	Пісковцов А.				
6	Польська Ю.		Математика		
7	Рехлицька С.		Історія України		

Рис. 2.56

Зверніть увагу, що функція автозавершення працює тільки у межах суміжних рядків клітинок. Якщо у стовпці буде присутня порожня клітинка, то автозавершення аналізуватиме лише вміст клітинок, що розташовані під порожньою клітинкою.

	A
1	Географія
2	Геометрія
3	географія

Рис. 2.57

ВПРАВА 9



Завдання. Створити списки по 10 елементів. На першому аркуші розташувати списки по вертикалі, на другому — по горизонталі.

- 1) 3, 7, 11, ...;
- 2) 0,5; 1; 1,5; 2; ...;
- 3) 60, 57, 54, ...;
- 4) -5, -4, -3, ...;
- 5) к2, к5, к8, ...;
- 6) файл1, файл2, файл3,

1. Для створення списків 1) – 5) уведіть спочатку перший елемент списку, нижче — другий.
2. Виділіть введені елементи.
3. Наведіть курсор на нижній правий кут другого елемента та протягніть його вниз, утримуючи ліву клавішу миші, до місця розташування десятого елемента.
4. Для створення списку 6) достатньо ввести лише перший елемент списку та застосувати до нього автозаповнення.
5. Для створення списків по горизонталі виконайте дії, що аналогічно використовують автозаповнення, протягуючи курсор миші праворуч.
6. Збережіть файл з іменем **Вправа_9.xlsx**.

ВПРАВА 10



Завдання. Створити власні списки, що будуть містити:

- 1) текстові написи цифр;
- 2) назви місяців українською мовою.

1. Перейдіть на вкладку **Файл** → **Параметри** → **Додатково**.
2. В області **Загальні** оберіть кнопку **Редагувати користувацькі списки**.
3. У полі **Елементи списку** введіть назви цифр (нуль, один, два, три, чотири, п'ять, шість, сім, вісім, дев'ять). Уведення кожної назви завершуйте натисненням клавіші **Enter**.
4. Натисніть кнопку **Додати**. Список має з'явитися у полі **Списки**.

5. Натисніть два рази **ОК**.

6. В електронній таблиці до клітинки введіть перше слово зі списку.



Виконати вправи 11 та 12 ви зможете в електронному додатку.



7. Поверніться до клітинки з цим словом та застосуйте до неї автозаповнення. В результаті має бути відображено список назв цифр.

8. Аналогічно створіть власний список назв місяців на українській мові.

9. Збережіть файл з іменем **Вправа_10.xlsx**.



Висновки

Особливим способом копіювання даних в Excel є **автозаповнення**, що прискорює процес введення даних до таблиць шляхом створення різних видів списків.

За допомогою автозаповнення можна здійснювати копіювання даних за певними правилами.

В Excel використовуються вбудовані списки, що містять назви днів тижнів та місяців.



Контрольні запитання та завдання

1. Для чого використовують функцію автозаповнення?
2. Які дані можна ввести з допомогою маркера автозаповнення?
3. Для чого використовують функцію автозавершення?



Питання для роздумів

1. Чи можна застосувати функцію автозаповнення для введення в електронну таблицю списку учнів класу?
- 2*. Наведіть приклади списків, які можна створити шляхом автоматичного введення даних.

§12.

Практична робота №2. «Уведення, редагування та форматування даних у середовищі табличного процесора»



Завдання. У даній практичній роботі ви маєте створити календар поточного року відповідно до наведеного зразка календаря за 2016 рік.

Хід роботи

1. Заповніть діапазон клітинок **A2:A13** назвами місяців, використовуючи автозаповнення (для цього у вас має бути в даному середовищі MS Excel попередньо створений користувацький список із назвами місяців).

№	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL		
1		ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ		
2	Січень					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
3	Лютий	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29										
4	Березень		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31							
5	Квітень					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30					
6	Травень						1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
7	Червень			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30							
8	Липень				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31					
9	Серпень	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31								
10	Вересень			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30							
11	Жовтень				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31					
12	Листопад		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30								
13	Грудень			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31						

Рис. 2.58

- Заповніть діапазон клітинок **B1:AL1** скороченими назвами днів тижнів, використовуючи автозаповнення (можна скористатися вбудованими назвами російської версії, змінивши лише деякі значення, або створити користувацький список з українськими скороченими назвами).
- Зменшіть розміри стовпців відповідно до ширини написів.
- Визначте день тижня, на який випадає **1 січня** поточного року.
- У другому рядку введіть число **1** навпроти відповідного дня тижня.
- У клітинці праворуч уведіть число **2**.
- Заповніть інші клітинки (до числа 31) даного рядка, використовуючи автозаповнення до клітинок із числами **1** та **2**.
- У третьому рядку (місяць Лютий) уведіть число **1** в ту клітинку, яка буде відповідати дню тижня, що є наступним після **31 січня**. Далі розташуйте число **2** та застосуйте автозаповнення.
- Числа інших місяців заповніть аналогічно (при заповненні числами пам'ятайте, що такі місяці, як квітень, червень, вересень та листопад, містять по 30 днів, лютий — 28 днів (у високосному році — 29 днів), а усі інші місяці — по 31 дню).
- Встановіть межі для клітинок відповідно до зразка.
- Назви місяців та днів тижня виділіть напівжирним шрифтом.
- Виділіть червоним кольором числа, на які припадають вихідні дні.
- Виконайте заливку клітинок відповідно до зразка.
- Збережіть файл з ім'ям **ПР_2.xlsx**.

§13.

Виконання обчислень

13.1.

ЗНАКИ МАТЕМАТИЧНИХ ОПЕРАЦІЙ

Раніше ми вчилися вводити, редагувати та форматовувати дані різних типів у таблиці. Проте основне призначення електронних таблиць — це спрощення обчислень. Розглянемо, як виконується обчислення в середовищі MS Excel.

Обчислення в таблицях виконують за допомогою формул. Формула може складатися з *математичних операторів*, *констант*, *посилань* та *функцій*. Порядок виконання обчислень або пріоритет операцій визначається звичайними математичними законами. Найпростіші обчислення виконують, використовуючи математичні операції: додавання, віднімання, множення, ділення та піднесення до степеня. Спочатку буде виконуватися дія піднесення до степеня, потім — ділення або множення й уже потім — додавання або віднімання.

Таблиця 2. Знаки математичних операцій

Знак у формулі	Математичний зміст	Пріоритет
+	Додавання	3
-	Віднімання	3
*	Множення	2
/	Ділення	2
^	Піднесення до степеня	1

13.2.

ВИДИ ФОРМУЛ

 Запам'ятайте, що будь-яка формула в Excel починається зі знака «=»!

Формули можна записувати по-різному:

- з допомогою *чисел* (наприклад, **=12+45-9**);
- з допомогою *адрес клітинок* (наприклад, **=A1+B2**);
- з допомогою *вбудованих функцій* (наприклад, **=SUM(A2:A5)**).

У першому прикладі формула дозволяє знайти значення числового виразу.

У другому прикладі у формулі здійснюється додавання вмісту клітинок **A1** та **B2**.

У третьому прикладі у формулі буде знайдено суму значень клітинок, що належать діапазону **A2:A5**.

Зверніть увагу, що формули можна вводити як у рядку формул, так і в клітинці. В клітинці буде відображено результат виконання формули, а формула буде відображена у рядку формул.

Розглянемо, як правильно записувати формули, що містять арифметичні операції з числами, а також послідовність дій уведення таких формул, виконавши наступну вправу.

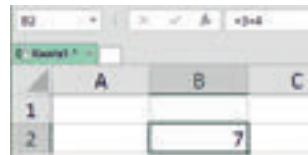


Рис. 2.59

ВПРАВА 13



Завдання. Обчислити значення виразів у середовищі програми MS Excel:

- 1) $25 : 5 + 4 \cdot 2$; 2) $7^2 - 4 \cdot 3$; 3) $\frac{(30 - 10)}{4 - 2}$.

- Створіть у програмі Excel новий документ.
- До клітинки **A1** уведіть першу формулу за правилами запису формул у програмі Excel: **=25/5+4*2**. Натисніть клавішу **Enter**. У результаті в клітинці **A1** одержите відповідь **13**.
- До клітинки **A2** уведіть другу формулу: **=7^2-4*3**. У клітинці **A2** має бути відповідь **37**.
- До клітинки **A3** уведіть третю формулу: **=(30-10)/(4-2)**. У клітинці **A3** результат обчислень — **10**.
- Збережіть документ з ім'ям **Вправа_13.xlsx**.

Формули, що були використані у цій вправі, використовують на практиці не дуже часто, оскільки такі дії можна виконати і на калькуляторі. Частіше для виконання обчислень у формулах використовують посилання на адреси клітинок. По-перше, при такому записі, якщо змінюють числові дані в клітинках, посилання на адреси яких записані у формулі, автоматично перерахується значення цієї формули. По-друге, це дозволяє виконувати обчислення з великою кількістю числових даних.

При введенні формул, що містять посилання на адреси клітинок, необхідно пам'ятати, що назви стовпців треба вводити латинськими літерами. А щоб уникнути помилок при введенні

адреси клітинки, не потрібно набирати її з клавіатури, а достатньо клацнути на відповідну клітинку — її буде обведено пунктирною рамкою, а у формулі відобразиться адреса.

Так, щоб увести таким способом формулу $=A3+C2$, треба:

- увести знак « $=$ »;
- клацнути на клітинці **A3**;
- увести знак « $+$ »;
- клацнути на клітинці **C2**;
- натиснути **Enter**.

В результаті у клітинці ми одержимо результат.

ВПРАВА 14



Завдання. Створити таблицю, як на рис. 2.60, та заповнити її жовті клітинки формулами для обчислення площі прямокутника S за двома сторонами a і b .

	A	B	C	D
1		a	b	S
2	Прямокутник 1	3	7	
3	Прямокутник 2	5	2	
4	Прямокутник 3	4	8	
5	Прямокутник 4	9	4	

Рис. 2.60

1. Створіть у програмі Excel новий документ.
2. Уведіть необхідні написи і значення довжин сторін у таблицю. Відформатуйте таблицю за зразком.
3. Уведіть до клітинки **D2** формулу: $=B2*C2$.
 - а) Виділіть клітинку **D2**.
 - б) Уведіть знак « $=$ ».
 - в) Клацніть на клітинці **B2**.
 - г) Уведіть знак « $*$ ».
 - г) Клацніть на клітинці **C2**.
 - д) Натисніть клавішу **Enter**.

У результаті ви отримаєте потрібну формулу (рис. 2.61).

	A	B	C	D
1		a	b	S
2	Прямокутник 1	3	7	$=B2*C2$
3	Прямокутник 2	5	2	
4	Прямокутник 3	4	8	
5	Прямокутник 4	9	4	

Рис. 2.61



4. Скопіюйте формулу з клітинки **D2** до діапазону клітинок **D3:D5**, використовуючи автозаповнення (рис. 2.62).

а) Зробіть активною клітинку **D2**.

б) Наведіть вказівник на нижній правий кут виділеної клітинки **D2** та протягніть його, утримуючи ліву клавішу миші, до клітинки **D5**.

	A	B	C	D
1		a	b	S
2	Прямокутник 1	3	7	21
3	Прямокутник 2	5	2	
4	Прямокутник 3	4	8	
5	Прямокутник 4	9	4	

Рис. 2.62

5. Змініть у клітинці **B2** значення 2 на 3 та переконайтеся в тому, що результат обчислень буде відразу змінено. Поекспериментуйте зі зміною інших значень довжин сторін **a** і **b**.
6. Збережіть документ у файлі з ім'ям **Вправа_14.xlsx**.

У результаті ви отримаєте таблицю з обчисленими значеннями площ чотирьох прямокутників.

	A	B	C	D
1		a	b	S
2	Прямокутник 1	3	7	21
3	Прямокутник 2	5	2	10
4	Прямокутник 3	4	8	32
5	Прямокутник 4	9	4	36

Рис. 2.63



Проаналізуємо формули, які було одержано в результаті виконання дії автозаповнення в даній вправі.

	A	B	C	D
1		a	b	S
2	Прямокутник 1	3	7	=B2*C2
3	Прямокутник 2	5	2	=B3*C3
4	Прямокутник 3	4	8	=B4*C4
5	Прямокутник 4	9	4	=B5*C5

Рис. 2.64



Як видно з таблиці, після копіювання формули **=B2*C2** на одну клітинку вниз номер рядка в посиланнях на адреси клітинок збільшився на 1 (замість **B2** стало **B3**, замість **C2** — **C3**). Копіювання формули на дві клітинки вниз збільшує номер рядка на 2 і т. д. Відповідні закономірності відбуваються також при копіюванні формул у будь-який бік (вгору, вниз, ліворуч, праворуч).

	A	B	C	D	E
1			=D3		
2			=D4		
3	=B5	=C5	=D5	=E5	=F5
4			=D6		
5			=D7		

Рис. 2.65

Так, на рис. 2.65 показано зміни посилання на адресу клітинки **D5** при копіюванні її вгору, вниз, праворуч, ліворуч.

ВПРАВА 15



Завдання. Знайти значення виразів в MS Excel у відповідних клітинках.

A1: $63 + 5 \cdot 8 - 4 : 2$

A2: $39 + (75 : 5 + 24 : 2) \cdot 6$

A3: $\frac{35+7}{8-6}$

A4: $6^2 - 15 : 5 + 4^3$

A5: $\frac{6^2}{2^3+1} + \frac{5^2}{3^2-4}$

1. Уведіть формули за правилами введення формул в MS Excel. При правильному введенні формул ви маєте одержати наступні результати:

Клітинка	Результат обчислень
A1	101
A2	201
A3	21
A4	97
A5	9

2. Збережіть файл з ім'ям **Вправа_15.xlsx**.

ВПРАВА 16



Завдання. Знайти значення периметра прямокутників у діапазоні **D2:D5**.

	A	B	C	D
1		a	b	S
2	Прямокутник 1	4	7	
3	Прямокутник 2	3	5	
4	Прямокутник 3	5	8	
5	Прямокутник 4	7	6	

Рис. 2.66

1. Створіть таблицю відповідно до зразка.
2. Уведіть до клітинки **D2** необхідну формулу.
3. Скопіюйте утворену формулу до діапазону **D3:D5**, використовуючи автозаповнення.
4. Збережіть документ з ім'ям **Вправа_16.xlsx**.

**Висновки**

Будь-яка формула в Excel починається зі знака «=».

Для написання формул використовують математичні знаки: «+» — додавання; «-» — віднімання; «*» — множення; «/» — ділення; «^» — піднесення до степеня.

Формули можна записувати по-різному: з використанням чисел; з використанням адрес клітинок; з використанням вбудованих функцій.

Формули можна вводити як до рядка формул, так і до клітинок.

**Контрольні запитання та завдання**

1. Які види формул ви знаєте?
2. З якого знака розпочинають вводити формули?
3. У яких елементах вікна табличного процесора можна вводити формулу?
- 4*. Для чого використовують формули в електронних таблицях?

**Питання для роздумів**

1. Що відобразиться в клітинці електронної таблиці, якщо при введенні формули не поставити знак «=»?
2. Що відбувається під час копіювання формули до іншої клітинки?

§14.**Практична робота №3.
«Виконання обчислень»**

Завдання 1. Виконати обчислення у порожніх клітинках файлу **Ромашка.xlsx**. Для обчислення виразів потрібно ввести формули у порожні клітинки, виділені рамкою та блакитним кольором (рис. 2.67).

Хід роботи

1. Завантажте до своєї папки файл **Ромашка.xlsx**.
2. Зробіть активною клітинку **J2**.
3. Уведіть формулу **=F8-H6**:
 - а) уведіть символ «=»;
 - б) клацніть клітинку **F8**, що містить число 1,5;
 - в) уведіть з клавіатури знак віднімання «-»;

- г) клацніть клітинку **H6**, що містить число 8,9 (у рядку формул буде відображено: **=F8-H6**).

	A	B	C	D	E	F	G	H	I	J	K
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											

Рис. 2.67

- Завершіть введення формули, натиснувши **Enter**.
У клітинці **J4** відобразиться результат **-7,4**, а у клітинці **K4** – **Молодець!**
Якщо в клітинці **K4** залишилось повідомлення **Розв'язуй...**, то при виконанні завдання було допущено помилку і необхідно це завдання виконати ще раз.
- Аналогічно заповніть формулами сім інших порожніх клітинок.
- Збережіть файл з ім'ям **ПРЗ_Завдання1.xlsx**.



Завдання 2. Виконати обчислення у порожніх клітинках, виділених рожевим та блакитним кольорами файлу **Ланцюжки.xlsx** на **першому** аркуші.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1														
2														
3														
4														
5														
6														
7														
8														
9														
10														
11														

Рис. 2.68

Хід роботи

- Завантажте файл **Ланцюжки.xlsx**.
- Відкрийте перший аркуш.



3. Зробіть активною клітинку **D5** і введіть у ній формулу **= B5^C4**. У клітинці **D5** з'явиться відповідь **64**.
4. Аналогічно введіть формули в інші виділені рамками клітинки.
У разі правильного виконання завдання у клітинках, що мають напис **Розв'язуй...**, з'явиться напис **Молодець**.
5. Збережіть файл.



Завдання 3. Виконати обчислення у порожніх клітинках, виділених рожевим кольором файлу **Ланцюжки.xlsx** на **другому** аркуші. Як і в попередніх завданнях, при правильності виконання обчислень має з'явитися повідомлення **Молодець**.

Збережіть файл з ім'ям **ПР3_Завдання2_3.xlsx**.



Виконати завдання 4
ви зможете в електронному додатку.

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2												
3												
4			0,24		1,8							
5				2,7			2,1		0,12			Розв'язуй...
6			+		:		x		-			
7												
8												
9			4,2		0,18		2,4		0,87			
10						9						Розв'язуй...
11			x		+		:		-			

Рис. 2.69

§15.

Вбудовані функції

15.1.

ПОНЯТТЯ ФУНКЦІЇ

Ознайомимося ще з одним інструментом електронних таблиць, що дозволяє прискорити виконання обчислень, а саме, із вбудованими функціями.

Функція — це заздалегідь визначена формула, згідно з якою виконуються обчислення за заданими величинами у вказаному порядку.

Наприклад, у таблиці наведено обсяги файлів деякої папки (рис. 2.70). У клітинці **I2** потрібно знайти загальний обсяг папки.

	A	B	C	D	E	F	G	H	I
1		Файл1	Файл2	Файл3	Файл4	Файл5	Файл6	Файл7	Папка
2	Обсяг, Мб	125	15	3	17	38	14	101	

Рис. 2.70

Якщо згадати відомі вам способи обчислень, можна було б, звісно, до клітинки I2 записати формулу:

$$=125+15+3+17+38+14+101.$$

Однак цей спосіб не є раціональним, оскільки при зміні якогось числового значення результат формули не зміниться. Другий відомий вам спосіб подання формул — з використанням посилань на адреси клітинок:

$$=B2+C2+D2+E2+F2+G2+H2.$$

Запис формули цим способом справді забезпечить переобчислення формули в разі зміни якогось числового значення. Проте у випадку, коли необхідно виконати додавання великої кількості значень, цей спосіб також не є раціональним, оскільки введення формул займе багато часу.

Розробники середовища Excel подбали про швидке виконання обчислень і додали до середовища понад сотню вбудованих функцій, що автоматизують різноманітні обчислення.

Отже, для швидкого обчислення суми в запропонованому прикладі необхідно спочатку виділити клітинку I2 та обрати інструмент Σ (Сума).

У цьому випадку програма самостійно визначить суміжний з обраною клітинкою діапазон клітинок, які треба додати (рис. 2.71), і сформує в клітинці I2 формулу із вбудованою функцією **=SUM(B2:H2)**. Якщо діапазон клітинок програмою визначено неправильно, можна завжди його змінити самостійно.

	A	B	C	D	E	F	G	H	I	J
1		Файл1	Файл2	Файл3	Файл4	Файл5	Файл6	Файл7	Папка	
2	Обсяг, Мб	125	15	3	17	38	14	101	=SUM(B2:H2)	
3										

Рис. 2.71

Щоб підтвердити застосування саме цієї формули, треба натиснути клавішу **Enter**. У результаті в рядку формул буде відображено формулу, а в клітинці I2 — результат обчислення (рис. 2.72).

	A	B	C	D	E	F	G	H	I
1		Файл1	Файл2	Файл3	Файл4	Файл5	Файл6	Файл7	Папка
2	Обсяг, Мб	125	15	3	17	38	14	101	313

Рис. 2.72

У попередній формулі ми використали одну із вбудованих в Excel функцій **SUM**, призначену для знаходження суми.

15.2.

ВИКОРИСТАННЯ ПАНЕЛІ ІНСТРУМЕНТІВ

Є випадки, в яких виконати обчислення за три кроки (1-й крок — виділити клітинку; 2-й — обрати інструмент; 3-й — натиснути **Enter**) неможливо. Розглянемо різні способи введення функцій у формули. Щоб уводити формули, використовуючи панель інструментів, потрібно:

- 1) Зробити активною клітинку, до якої треба ввести формулу.
- 2) Натиснути інструмент Σ (для виконання дії додавання) або в меню інструменту (рис. 2.73) обрати бажану команду (призначення команд більш детально описано в табл. 4 на стор. 86).
- 3) Переконаватися, що програма виділила необхідний діапазон клітинок для виконання дії. За потреби змінити діапазон, протягнувши по ньому курсором та утримуючи ліву клавішу миші.
- 4) Натиснути клавішу **Enter**.

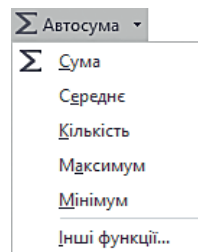


Рис. 2.73

15.3.

ОЗНАЙОМЛЕННЯ ІЗ ВБУДОВАНИМИ ФУНКЦІЯМИ

Розглянемо більш детально використання вбудованих функцій, які розташовані в меню інструменту **Сума** (рис. 2.73).

Перша команда меню, з якою ми вже трохи познайомилися, — це **Сума**.

Їй відповідає функція **SUM** в українській та англійській версіях програми Excel, в російській версії вона має напис **СУММ**.

Неважко здогадатися, що використання цієї функції дозволяє знайти суму значень указаних у дужках клітинок.

	A	B	C	D	E
1	7	10	4	=SUM(A1:C1)	
2				SUM(число1; [число2]; ...)	

	A	B	C	D	E
1	7	10	4	21	

Рис. 2.74

Наступна команда **Середнє**. Їй відповідає функція **AVERAGE** в українській та англійській версіях програми Excel, в російській версії вона має напис **СРЗНАЧ**.

Використання цієї функції дозволяє знайти середнє значення вказаних у дужках клітинок. З математичної точки зору здійснюється знаходження суми значень указаних клітинок, після чого ця сума ділиться на кількість значень. Наприклад, $(5+10+6):3=7$.

	A	B	C	D	E
1	5	10	6	=AVERAGE(A1:C1)	
2				AVERAGE(число1; [число2]; ...)	

	A	B	C	D	E
1	5	10	6	7	

Рис. 2.75

Наступна команда **Кількість**. Їй відповідає функція **COUNT** або **СЧЁТ** в російській версії Excel.

Використання цієї функції дозволяє визначити кількість значень, що є числами. Так, у прикладі, наведеному нижче, результат дорівнює 2, оскільки діапазон містить два числа 5 і 6, а клітинка **B1** містить текст — це літера «А».

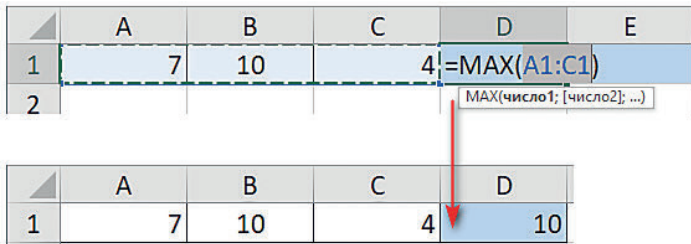
	A	B	C	D	E
1	5	A	6	=COUNT(A1:C1)	
2				COUNT(значення1; [значення2]; ...)	

	A	B	C	D	E
1	5	A	6	2	

Рис. 2.76

Ще одна команда меню — це **Максимум**, і їй відповідає функція **MAX** або **МАКС** в залежності від інтерфейсу Excel.

Завдяки цій функції можна визначити максимальне значення із вказаного діапазону чисел.

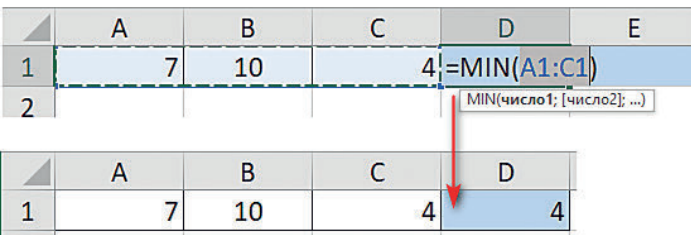


	A	B	C	D	E
1	7	10	4	=MAX(A1:C1)	
2					

	A	B	C	D
1	7	10	4	10

Рис. 2.77

Подібною за своїм призначенням є команда меню **Мінімум**, що визначає мінімальне значення із діапазону чисел, і їй відповідає функція **MIN** або **МИН** в залежності від мови інтерфейсу.



	A	B	C	D	E
1	7	10	4	=MIN(A1:C1)	
2					

	A	B	C	D
1	7	10	4	4

Рис. 2.78

Якщо нам потрібно застосувати одну із вбудованих функцій не до діапазону клітинок, а до несуміжних клітинок, після вибору необхідної команди меню достатньо послідовно клацнути на клітинки, утримуючи клавішу **Ctrl**, при цьому у формулі посилання на адреси клітинок будуть відокремлюватися знаком **крапка з комою**, а не двокрапкою, як у випадку з діапазонами клітинок.

Ми ознайомилися з основними вбудованими функціями, доступ до яких можна легко отримати з використанням меню інструменту **Сума**. В таблиці нижче представлені усі розглянуті функції.

Таблиця 3. Перелік деяких вбудованих функцій

Функція	Функція (рос.)	Команда в меню інструменту Сума	Призначення
SUM (число1;число2;...)	СУММ	Сума	Додавання чисел
AVERAGE (число1;число2;...)	СРЗНАЧ	Середнє	Обчислення середнього арифметичного чисел
COUNT (число1;число2;...)	СЧЁТ	Кількість	Визначення кількості чисел
MAX (число1;число2;...)	МАКС	Максимум	Знаходження максимального числа серед інших чисел
MIN (число1;число2;...)	МИН	Мінімум	Знаходження мінімального числа серед інших чисел

ВПРАВА 17



Завдання. Створити таблицю, щоб обчислити загальну довжину вказаних у ній річок, а також загальну площу їхніх басейнів.

	A	B	C	D
1	Найбільші річки світу			
2	№	Річка	Загальна довжина, км	Площа басейну, тис. км кв.
3	1	Ніл	6671	2870
4	2	Амазонка	6437	7180
5	3	Міссісіпі	6019	3268
6	4	Янцзи	5800	1880
7	5	Волга	3530	1360
8		Всього		

Рис. 2.79

- Створіть таблицю відповідно до зразка, що на рисунку нижче.
- Виконайте обчислення суми загальної довжини річок у клітинці **C8**.
 - Виділіть клітинку **C8**.
 - Оберіть інструмент Σ (Сума).
 - Переконайтеся, що до функції включено необхідний діапазон **C3:C7**. Якщо діапазон вказано неправильно, змініть його значення, протягнувши курсор по необхідному діапазону, утримуючи ліву клавішу миші.



- г) Натисніть клавішу **Enter**. У результаті до клітинки **C8** буде застосовано формулу **=SUM(C3:C7)**.
3. Виконайте копіювання одержаної в клітинці **C8** формули до клітинки **D8**, використовуючи автозаповнення.
- а) Зробіть активною клітинку **C8**.
- б) Наведіть курсор на нижній правий кутик клітинки та протягніть курсор праворуч на одну клітинку. В результаті до клітинки **D8** буде застосовано формулу **=SUM(D3:D7)**.
4. Збережіть документ у файлі з ім'ям **Вправа_17.xlsx**.



Висновки

Середовище Excel містить понад сотню вбудованих функцій, що автоматизують різноманітні обчислення.

Щоб увести формулу, необхідно: зробити **активною клітинку**; в меню інструменту **Сума** обрати відповідну команду; переконатися, що програма виділила необхідний діапазон клітинок (якщо це не так, тоді виділити необхідний діапазон); натиснути клавішу **Enter**.

Призначення деяких вбудованих функцій: **SUM** — додає числа, **AVERAGE** — обчислює середнє арифметичне чисел, **COUNT** — визначає кількість числових значень, **MAX**, **MIN** — визначають максимальне та мінімальне значення з діапазону чисел.



Контрольні запитання та завдання

1. Як у формулу вставити функцію?
2. Які вбудовані функції вам відомі? Опишіть їхнє призначення.
- 3*. Вкажіть послідовність дій для додавання вбудованої функції до формули за допомогою панелі інструментів.

§16.

Використання вбудованих функцій

16.1.

УВЕДЕННЯ ФОРМУЛ З КЛАВІАТУРИ

Функції в Excel призначені для прискорення дій обчислення. Їх використовують як частини формул. Записуючи формули із вбудованими функціями, треба дотримуватися певних правил.

1. Після запису назви функції в дужках записують її аргументи (числа, адреси клітинок, діапазони клітинок, інші функції та вирази).

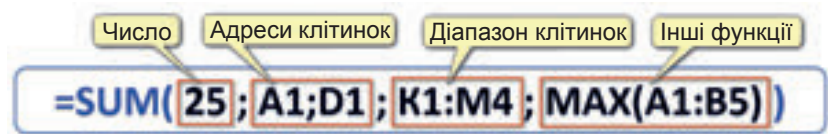


Рис. 2.80

2. Аргументи у функціях відокремлюються один від одного символом крапка з комою.



Рис. 2.81

	A	B	C	D	E
1	1	2	3	4	=SUM(A1;D1)
2					SUM(число1; [число2];

	A	B	C	D	E
1	1	2	3	4	5

Рис. 2.82

	A	B	C	D	E
1	1	2	3	4	=SUM(A1:D1)
2					SUM(число1; [число2];

	A	B	C	D	E
1	1	2	3	4	10

Рис. 2.83

Розглянемо декілька прикладів формул з використанням вбудованих функцій.

Так, в результаті обчислення формули **=SUM(A1;D1)** буде знайдено суму чисел, розташованих у клітинці **A1** та **D1**.

На відміну від попередньої формули, результатом виконання формули **=SUM(A1:D1)** буде сума діапазону клітинок від **A1** до **D1**, а саме, клітинок **A1**, **B1**, **C1** та **D1**.

На перший погляд ці дві формули однакові, але у першій клітинки **A1** і **D1** відокремлені знаком крапка з комою, і це означає, що необхідно врахувати лише значення двох клітинок **A1** та **D1**, у другій формулі адреси відокремлені знаком двокрапки, тобто буде обчислено суму діапазону клітинок від **A1** до **D1**.

Із попередніх пояснень неважко здогадатися, що у формулі **=SUM(A1:D1;K1)** буде знайдено суму чисел, розташованих у клітинах від **A1** до **D1**, та клітинки **K1**.

А у формулі **=MAX(SUM(A1:D1);SUM(A2:D2))** спочатку буде визначено суму в діапазоні **A1:D1**, потім суму в діапазоні **A2:D2**, і вже потім між ними буде визначено максимальне значення.

16.2.

ВИКОРИСТАННЯ МАЙСТРА ВВЕДЕННЯ ФОРМУЛ

Оскільки користувач не завжди може пам'ятати усі вбудовані функції (в Excel їх більше сотні) розробники програми подбали про можливість вибору необхідної функції в діалоговому вікні **Вставлення функції**, яке можна викликати, натиснувши кнопку **fx** , що розташована ліворуч від рядка формул. Також майстра можна викликати, натиснувши інструмент **Вставити функцію** на вкладці **Формули**.

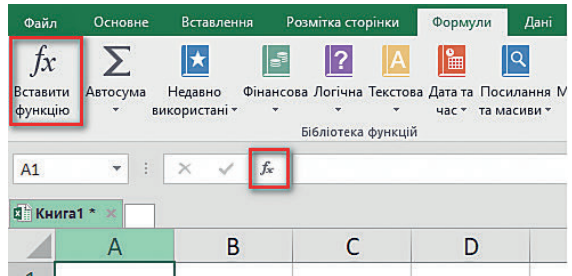


Рис. 2.84

В результаті відкриється діалогове вікно **Вставлення функції**.

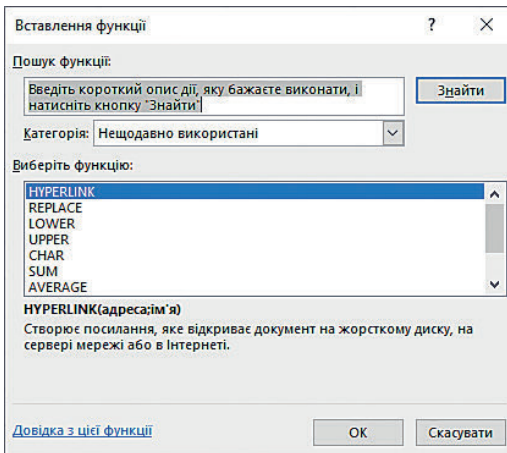


Рис. 2.85

Розглянемо використання майстра функцій для знаходження мінімального значення результатів змагань учнів класу.

1. Зробіть активною клітинку **B9**.
2. Викличте діалогове вікно **Вставлення функції**, натиснувши кнопку **fx** біля рядка формул.

	A	B
1	Прізвище	Час за 100 м
2	Ігнатенко	14,8
3	Петренко	15,2
4	Свірдов	16,1
5	Сердюк	14,5
6	Хижняков	15,5
7	Якущенко	16,3
8		
9	Кращий результат	

Рис. 2.86

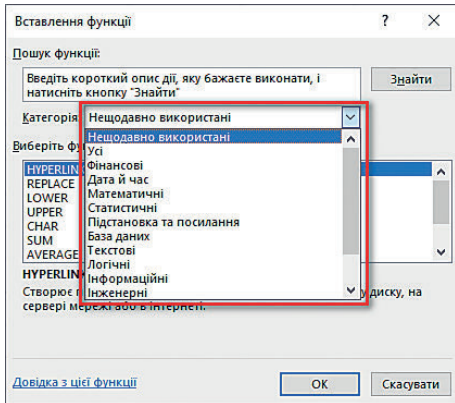


Рис. 2.87

Зауваження. Результат виконання дії до завершення роботи з майстром можна буде побачити під областю введення аргументів.

	A	B	C	D	E	F	G	H
1	Прізвище	Час за 100 м						
2	Ігнатенко	14,8						
3	Петренко	15,2						
4	Свірідов	16,1						
5	Сердюк	14,5						
6	Хижняков	15,5						
7	Якущенко	16,3						
8								
9	Кращий результат	=MIN(B2:B7)						
10								
11								

Рис. 2.88

6. Натисніть **ОК**. В результаті буде виведено мінімальне значення значення діапазону **14,5**.

ВПРАВА 18



Завдання. Створити таблицю оцінювання учнів за I семестр, у якій визначити середнє значення балів кожного учня, а також найвищий, найнижчий та середній результати учнів з кожного предмета, увівши відповідні формули до жовтих клітинок.

1. Створіть таблицю відповідно до зразка або завантажте файл **Вправа_18.xlsx** з електронного додатка до уроків.



	A	B	C	D	E	F	G	H	I
1	№	Прізвище	Математика	Укр. мова	Укр. літ.	Географія	Англ. мова	Інформатика	Середнє значення
2	1	Іванов	8	7	9	6	7	10	
3	2	Петров	12	8	9	7	8	9	
4	3	Сидоров	4	5	6	7	6	8	
5	4	Ліньков	8	9	7	8	6	8	
6	5	Котов	6	9	8	7	6	5	
7	6	Романов	9	6	8	7	4	5	
8	7	Марченко	6	9	8	7	5	7	
9	8	Шевченко	10	11	10	9	10	10	
10	9	Гук	12	5	6	9	5	8	
11	10	Андрієнко	9	8	10	10	9	10	
12	Найвищий результат								
13	Найнижчий результат								
14	Середнє значення								

Рис. 2.89

2. До клітинки I2 введіть формулу визначення середнього значення серед діапазону клітинок C2:H2. Розглянемо спосіб введення формули з клавіатури до рядка формул.
 - а) Зробіть активною клітинку I2.
 - б) Клацніть у рядку формул і почніть вводити формулу =AV, після чого у списку, що з'явиться, оберіть функцію AVERAGE.
 - в) Виділіть діапазон C2:H2.
 - г) Закрийте дужку та натисніть клавішу Enter.
3. Виконайте копіювання формули із клітинки I2 до діапазону клітинок I3:I11, використовуючи автозаповнення.
Наступні формули введіть, використовуючи майстер введення формул.
4. До клітинки C12 введіть формулу визначення максимального значення серед діапазону клітинок C2:C11.
 - а) Зробіть активною клітинку C12.
 - б) Викличте діалогове вікно Вставлення функції, натиснувши кнопку  біля рядка формул.
 - в) Оберіть категорію Статистичні.
 - г) У полі Виберіть функцію оберіть функцію MAX, після чого натисніть ОК.
 - г) У діалоговому вікні Аргументи функції у полі Число 1 введіть як аргумент діапазон клітинок C2:C11.
 - д) Натисніть ОК.

5. Аналогічно в клітинці **C13** обчисліть мінімальне значення серед діапазону клітинок **C2:C11** та середнє значення в клітинці **C14** серед діапазону **C2:C11**.



Виконати вправу 19 ви зможете в електронному додатку.



6. Виконайте копіювання формул із клітинок **C12**, **C13** та **C14** до необхідних діапазонів, використовуючи автозаповнення.

а) Виділіть клітинки **C12:C14**.

б) Наведіть курсор на нижню праву область діапазону клітинок та протягніть курсор праворуч до клітинки **H14**.

7. Збережіть документ з ім'ям **Вправа_18.xlsx**.



Висновки

При використанні вбудованих функцій у формулах слід дотримуватися певних правил:

- після запису назви функції в дужках записуються її аргументи (числа, адреси клітинок, діапазони клітинок, інші функції та вирази);
- аргументи у функціях відокремлюються один від одного символом «;».

Оскільки користувач не завжди може пам'ятати усі вбудовані функції, їх зручно обирати з використанням діалогового вікна **Вставлення функції**, яке можна викликати, натиснувши відповідний інструмент.



Контрольні запитання та завдання

1. Яких правил треба дотримуватися, записуючи вбудовану функцію?
2. Опишіть послідовність дій використання майстра функцій.
3. Чим відрізняється використання символу «:» від «;» у вбудованих функціях?
4. Визначте, до яких категорій майстра функцій належать відомі вам вбудовані функції.



§17. Помилки у формулах

17.1. ВИДИ ПОМИЛОК

Іноді після введення формули в клітинці замість результату обчислень з'являється повідомлення про помилку. В таблиці 5 представлено їхній перелік та причини виникнення. Оскільки багато користувачів використовують російську версію MS Excel, у дужках наводимо запис помилки російською.

Таблиця 4. Повідомлення про помилку у формулі

Помилка	Причина виникнення
#####	Результат обчислення формули не вміщується в клітинці
#DIV/0! (#ДЕЛ/0)	Спроба ділення на нуль
#N/A (#Н/Д)	Значення недоступне
#NAME? (#ИМЯ?)	Неможливо розпізнати ім'я у формулі
#NUM! (#ЧИСЛО!)	Неправильні числові значення
#REF! (#ССЫЛКА)	Неприпустиме посилання на клітинку
#VALUE! (#ЗНАЧ!)	Неприпустимий тип аргументу

17.2. СПОСОБИ УСУНЕННЯ ПОМИЛОК

Перший тип помилки з'являється тоді, коли числове значення не вміщується в клітинці, зазвичай це стосується запису дат або довгих чисел. Для виправлення цієї помилки достатньо збільшити ширину клітинки.

В інших випадках потрібно виправляти формули відповідно до одержаної помилки (наприклад, перевірити формат даних, правильність запису чисел, адресу клітинок тощо).

Щоб дізнатися, на які клітинки посилається формула, треба двічі клацнути на клітинці з формулою — й адреси всіх зазначених у ній клітинок буде виділено різними кольорами, а навколо цих клітинок з'являться рамки таких самих кольорів.

Так, у клітинці **C2** на рис. 2.90 зазначено помилку, що вказує на спробу ділення на 0.

Для аналізу цієї формули слід двічі клацнути на клітинці, що містить формулу (рис. 2.91).

	A	B	C	D
1				
2		1	0	#DIV/0!

Рис. 2.90

	A	B	C	D
1				
2		1	0	=B2/C2

Рис. 2.91

	A	B	C	D	E
1					
2		1	2	3	#NAME?

Рис. 2.92

	A	B	C	D	E
1					
2		1	2	3	=B2+C2+D2

Рис. 2.93

	A	B	C	D	E
1					
2		1	2	A	#VALUE!

Рис. 2.94

	A	B	C	D	E
1					
2		1	2	A	=B2+C2+D2

Рис. 2.95

З формули видно, що знаменником є число 0, який знаходиться у клітинці **C2**. Для виправлення помилки достатньо змінити значення клітинки **C2** на число, відмінне від нуля.

В наступному прикладі помилка виникла в клітинці **E2**. Повідомлення у клітинці вказує на те, що неможливо розпізнати ім'я у формулі.

Переглянемо цю формулу, двічі клацнувши на клітинці.

Як бачимо, підсвічено лише клітинки **B2** і **D2**, а клітинка **C2** є непоміченою, оскільки програма не сприймає її як посилання на адресу клітинки. Напевно, у формулі було введено посилання на адресу клітинки літерами кирилиці. Для виправлення помилки слід змінити написання адреси **C2** на літеру латиницею.

В наступному прикладі в клітинці **E2** повідомлення про помилку вказує на неприпустимий тип аргументу.

Переглянемо цю формулу, двічі клацнувши на клітинці.

Як видно із формули, у ній є спроба додати два числа і значення клітинки, що містить текст. Звичайно, дію додавання можна виконувати лише з числами. Якщо змінити значення клітинки **D2** на число, помилка зникне.



Висновки

Іноколи після введення формули в клітинці замість результату обчислень з'являється повідомлення про помилку.

Щоб її виправити, потрібно визначити причину виникнення помилки та відредагувати формулу.



Контрольні запитання та завдання

1. Які можливі помилки при введенні формул?
- 2*. Опишіть способи усунення різних видів помилок.

§18.

Практична робота №4. «Виконання обчислень за даними електронної таблиці»



Завдання 1. Записати формули до клітинок **D2:D5** для обчислення значення виразу за формулою:

$$y = a^2 + \frac{b - c}{ab - c^2}.$$

Хід роботи

	A	B	C	D
1	a	b	c	y
2	5	11	13	
3	4	5	21	
4	1	7	7	
5	3	10	10	

Рис. 2.96

1. На першому аркуші створіть таблицю відповідно до зразка.
2. Запишіть до клітинки **D2** формулу за правилами запису математичних дій в Microsoft Excel. Зауважте, що замість змінних **a**, **b** і **c** у формулі слід указувати адреси клітинок, у яких знаходяться значення цих змінних, а саме, **A2**, **B2** і **C2** відповідно.
3. Виконайте копіювання формули з клітинки **D2** до діапазону клітинок **D3:D5**.
4. Надайте аркушу ім'я **Завдання_1**.



Завдання 2. На другому аркуші документа створити таблицю для обчислення вартості зернової суміші, заготовленої для птахів у зоопарку, як показано на рис. 2.97.

	A	B	C	D	E	F	G
1	№	Компонент суміші	Кількість, кг	Ціна за 1 кг, грн	Вартість суміші на 1 день, грн	Вартість суміші на 1 тиждень, грн	Вартість суміші на 1 місяць, грн
2	1	Просо	4	6			
3	2	Овес	3	8			
4	3	Насіння бур'янів	2	30			
5	4	Насіння соняшника	1	20			
6		Всього					

Рис. 2.97

1. Додайте до книги Excel новий аркуш та надайте йому ім'я **Завдання_2**.
2. На цьому аркуші створіть таблицю відповідно до зразка.

3. Заповніть формулами клітинки, виділені жовтим кольором.
4. Скопіюйте формули з клітинок **E2**, **F2**, **G2** до діапазонів клітинок **E3:E5**, **F3:F5** та **G3:G5** відповідно, а формулу з клітинки **E6** до діапазону клітинок **F6:G6**.
5. У клітинках з формулами встановіть грошовий формат з відповідною валютою.



Завдання 3. Створити на третьому аркуші робочої книги таблицю, що демонструватиме наявну в інтернет-магазині кількість моделей телевізорів від різних виробників, як на рис. 2.98.

	A	B
1	Виробник	Кількість моделей
2	Samsung	192
3	Panasonic	100
4	Toshiba	46
5	Sony	118
6	Supra	31
7	Sharp	54
8	JVC	4
9	Philips	71
10	Всього моделей	
11	Всього виробників	

Рис. 2.98

1. Додайте до книги Excel новий аркуш та надайте йому ім'я **Завдання_3**.
2. На цьому аркуші створіть таблицю відповідно до зразка.
3. До клітинки **B10** уведіть формулу для визначення суми значень серед діапазону клітинок **B2:B9**.



Виконати завдання 4 ви зможете в електронному

додатку. 

4. До клітинки **B11** уведіть формулу для визначення кількості значень серед діапазону клітинок **B2:B9**. (Для цього в меню інструменту **Сума** оберіть команду **Кількість**.)

§19.

Моделі. Етапи побудови моделей

19.1.

ПОНЯТТЯ МОДЕЛІ

У своїй діяльності людина дуже часто використовує моделі, тобто створює образ того об'єкта, явища або процесу, з яким їй доведеться працювати.

Модель (від латинського *modulus* — міра, зразок, норма) — це певний новий спрощений об'єкт, який відображає суттєві особливості реального об'єкта, процесу або явища.



Початковий об'єкт — **прототип** або **оригінал**



Спрощений об'єкт — **модель**

Аналіз моделі і спостереження за нею дозволяють пізнати суть реально існуючого, більш складного об'єкта, процесу або явища. Початковий об'єкт прийнято називати **прототипом** або **оригіналом**.

Отже, **моделювання** — це метод дослідження предметів, процесів і явищ шляхом заміни конкретного об'єкта досліджень на модель.

У кожній галузі науки досліджують об'єкти і процеси під різними кутами зору та будують різні типи моделей.

У **фізиці** моделюють рух тіла під дією різних сил, у **хімії** вивчають будову, склад і властивості речовин, у **біології** — будову та поведінку живих організмів.



Чому люди створюють моделі оригіналів, ви **зможете дізнатися** в електронному додатку.

19.2.

ІНФОРМАЦІЙНІ МОДЕЛІ

За способом представлення моделі поділяються на матеріальні та інформаційні.

Інформаційна модель — це модель об'єкта, подана у вигляді опису. Основу такої моделі становить цілеспрямовано відібрана інформація про об'єкт, що відображає його найбільш суттєві властивості для дослідника.

Інформаційні моделі можуть бути представлені різними способами. Ці способи (типи) ви бачите на схемі нижче (рис. 2.99).



Які існують ознаки моделей, ви **зможете дізнатися** в електронному додатку.



Рис. 2.99. Класифікація моделей за способом представлення

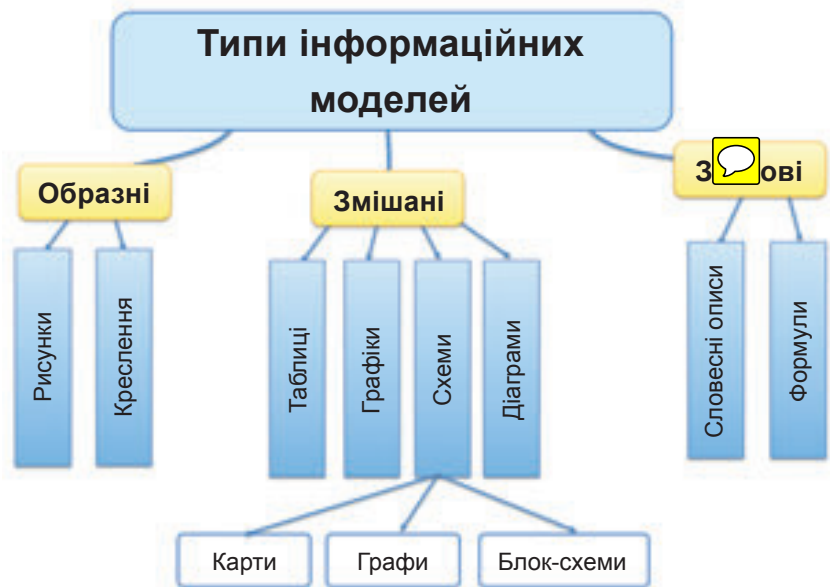


Рис. 2.100

Досить часто при створенні інформаційних моделей звертаються до математичної мови, а саме, подають певні властивості об'єктів у вигляді математичних понять, формул, графіків.

Отже, моделі, що побудовано з використанням математичних понять і формул, називають **математичними моделями**.

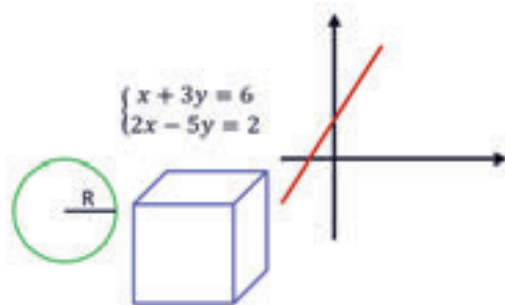


Рис. 2.101. Приклади математичних моделей

19.3.

ЕТАПИ ПОБУДОВИ КОМП'ЮТЕРНОЇ МОДЕЛІ

Комп'ютерне моделювання — це моделювання об'єктів, процесів, явищ засобами спеціальних комп'ютерних програм: графічних та анімаційних редакторів, табличних процесорів, програм для створення баз даних, спеціальних комп'ютерних тренажерів-симуляторів, віртуальних лабораторій тощо.

Якщо модель формулюється таким чином, що її можна обробити на комп'ютері, вона називається *комп'ютерною*.

Отже, *комп'ютерна модель* – це модель, реалізована за допомогою програмних засобів.

Комп'ютерні моделі звичайно розрізняють за програмним забезпеченням, яке застосовується під час роботи з моделлю. Для обробки комп'ютерних моделей використовуються існуючі програмні додатки (математичні пакети, електронні таблиці, графічні редактори тощо) або розробляються оригінальні програми за допомогою мов програмування (Basic, Pascal, Delphi, C++).

Розглянемо основні етапи комп'ютерного моделювання.

1 етап. Постановка задачі та її аналіз.

2 етап. Побудова інформаційної моделі.

3 етап. Розробка алгоритму дослідження моделі.

4 етап. Розробка комп'ютерної моделі.

5 етап. Проведення комп'ютерного експерименту.

Розглянемо сутність цих етапів на прикладі наступної задачі.



Задача. Вкладник хоче покласти на банківський рахунок x грн. Альфа-банк приймає вклад під прості відсотки у розмірі 14 відсотків річних, Бета-банк — під складні відсотки у розмірі 13 відсотків річних. Допоможіть вкладнику визначитись, у який банк вигідніше вкласти гроші, дослідивши його прибуток протягом 10 років. Чи залежить вибір банку вкладника від строку вкладу? Від суми вкладу?



Рис. 2.102

1 етап. Постановка задачі та її аналіз.

На цьому етапі необхідно визначити мету моделювання, що в задачі дано, що необхідно знайти, а також урахувати обмеження на дані.

В нашій задачі **дано** суму x , яку вкладник має покласти на рахунок.

Знайти необхідно суму, яку отримає вкладник через 1, 2, 3 і так далі до 10 років в Альфа- та Бета-банках.

Метою моделювання є вибір банку з найвигіднішими умовами.

2 етап. Побудова інформаційної моделі.

x — значення суми вкладу.

$a_1, a_2, \dots, a_{10}$ — прибуток на кінець кожного року в Альфа-банку.

$b_1, b_2, \dots, b_{10}$ — прибуток на кінець кожного року в Бета-банку. Визначимо формули.

Формула обчислення прибутку за простими відсотками має вигляд:

$$a_n = x(1+r/100 \cdot n),$$

де x — сума вкладу; r — відсоток річних; n — кількість років.

Отже, формула обчислення прибутку в Альфа-банку:

$$a_n = x(1+0,14 \cdot n).$$

Відповідно формула обчислення прибутку за **складними** відсотками має вигляд:

$$b_n = x(1+r/100)^n.$$

А формула обчислення прибутку в Бета-банку:

$$b_n = x(1+0,13)^n.$$

3 етап. Розробка алгоритму дослідження моделі.

Розробимо алгоритм дослідження з допомогою блок-схеми (рис. 2.103). Проаналізуємо кроки блок-схеми:

- 1) введення суми вкладу;
- 2) обчислення прибутку на кінець кожного року в Альфа- та Бета-банках протягом десяти років;
- 3) аналіз одержаних даних та визначення прибутковішого банку;
- 4) визначення прибутковості банку в залежності від кількості років вкладу;
- 5) визначення залежності вибору банку від суми вкладу.

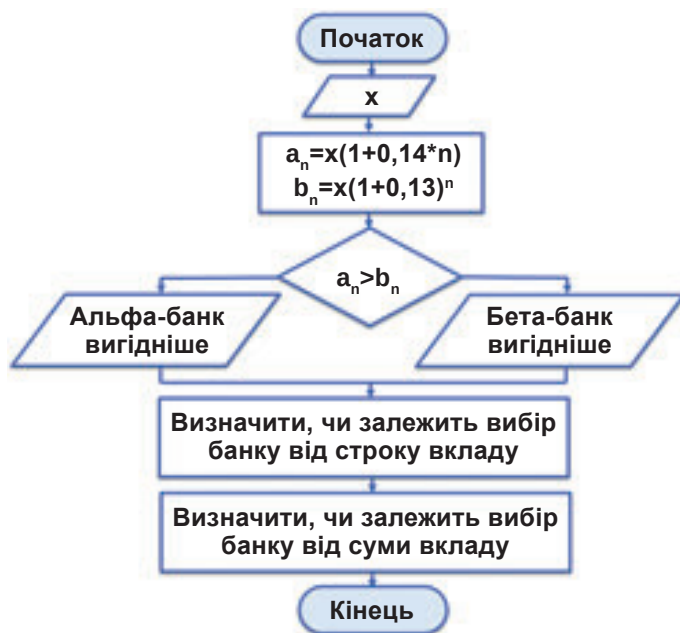


Рис. 2.103

4 етап. Розробка комп'ютерної моделі.

Будувати і досліджувати комп'ютерні моделі можна за допомогою різних програм. Кожне програмне середовище має

	A	B	C	D
1		Сума вкладу	Альфа-банк	Бета-банк
2	рік	x	a	b
3	1			
4	2			
5	3			
6	4			
7	5			
8	6			
9	7			
10	8			
11	9			
12	10			

Рис. 2.104

	A	B	C	D
1		Сума вкладу	Альфа-банк	Бета-банк
2	рік	x	a	b
3	1	1000	1140	1130,00
4	2	1000	1280	1276,90
5	3	1000	1420	1442,90
6	4	1000	1560	1630,47
7	5	1000	1700	1842,44
8	6	1000	1840	2081,95
9	7	1000	1980	2352,61
10	8	1000	2120	2658,44
11	9	1000	2260	3004,04
12	10	1000	2400	3394,57

Рис. 2.105

свій інструментарій і дозволяє працювати з певними видами інформаційних моделей. Ту саму задачу можна розв'язувати, використовуючи різні середовища.

Побудуємо модель розв'язання даної задачі в середовищі табличного процесора Excel у вигляді наступної таблиці.

1. У стовпчику **B** розташуємо значення суми вкладу.
2. У клітинку **B3** введемо суму вкладу, наприклад 1000.
3. У клітинку **B4** введемо формулу **=B3**.
4. Скопіюємо цю формулу нижче, використовуючи автозаповнення.
5. До клітинки **C3** введемо формулу прибутку за простими відсотками для Альфа-банку **=B3*(1+0,14*A3)**.
6. Скопіюємо цю формулу нижче, використовуючи автозаповнення.
7. До клітинки **D3** введемо формулу прибутку за складними відсотками для Бета-банку **=B3*(1+0,14)^A3**.
8. Скопіюємо цю формулу нижче, використовуючи автозаповнення.

5 етап. Проведення комп'ютерного експерименту.

Експеримент повинен супроводжуватися аналізом результатів для прийняття рішення.

Згідно з одержаними даними ми бачимо, що при сумі вкладу 1000 грн на перші два роки вклад робити вигідніше в Альфа-банку, якщо ж робити довгостроковий вклад на 3 і більше років, то кращими є умови в Бета-банку.



	A	B	C	D
1		Сума вкладу	Альфа-банк	Бета-банк
2	рік	x	a	b
3	1	1000	1140	1130,00
4	2	1000	1280	1276,90
5	3	1000	1420	1442,90
6	4	1000	1560	1630,47
7	5	1000	1700	1842,44
8	6	1000	1840	2081,95
9	7	1000	1980	2352,61
10	8	1000	2120	2658,44
11	9	1000	2260	3004,04
12	10	1000	2400	3394,57

Рис. 2.106

Перевіримо, чи вплине на вибір банку сума вкладу.

Збільшимо суму вкладу і до клітинки **B3** введемо 10000. В результаті тенденція залишиться та сама. В перші 2 роки вигіднішим є Альфа-банк, далі – Бета-банк.

	A	B	C	D
1		Сума вкладу	Альфа-банк	Бета-банк
2	рік	x	a	b
3	1	10000	11400	11300,00
4	2	10000	12800	12769,00
5	3	10000	14200	14428,97
6	4	10000	15600	16304,74

Рис. 2.107

Спробуємо зменшити суму, нехай наш вклад буде 100 грн. І знову, як бачимо, вибір банків не зміниться.

	A	B	C	D
1		Сума вкладу	Альфа-банк	Бета-банк
2	рік	x	a	b
3	1	100	114	113,00
4	2	100	128	127,69
5	3	100	142	144,29
6	4	100	156	163,05

Рис. 2.108

Як бачимо, після декількох кроків можна вже з упевненістю зробити висновок, що перші два роки вклад робити вигідніше

в Альфа-банк — Бета-банк є вигідним для довгострокових депозитів.



Рис. 2.109

ВПРАВА 20



Завдання. Створити комп'ютерну модель розв'язання задачі.

Задача. Вкладник хоче покласти на банківський рахунок x грн. Альфа-банк приймає вклад під прості відсотки у розмірі 14 відсотків річних, Бета-банк — під складні відсотки у розмірі 13 відсотків річних. Допоможіть вкладнику визначитись, у який банк вигідніше вкласти гроші, дослідивши його прибуток протягом 10 років. Визначте, чи залежить вибір банку вкладника від строку вкладу; від суми вкладу. Перевірте свої міркування для початкової суми вкладу 100 грн, 1000 грн, 5000 грн, 10000 грн. Збережіть файл з ім'ям **Вправа_20.xlsx**.



Висновки

1. Модель — це спрощений об'єкт, який відображає суттєві особливості реального об'єкта, процесу або явища. Початковий об'єкт називають **прототипом** або **оригіналом**.

2. Моделювання — це метод дослідження предметів, процесів і явищ шляхом заміни конкретного об'єкта досліджень на модель.

3. Інформаційна модель — це модель об'єкта, подана у вигляді опису.

4. Моделі, що побудовано з використанням математичних понять і формул, називають математичними моделями.

5. Побудова комп'ютерної моделі складається з п'яти етапів:

- 1) Постановка задачі та її аналіз.
- 2) Побудова інформаційної моделі.
- 3) Розробка алгоритму дослідження моделі.
- 4) Розробка комп'ютерної моделі.
- 5) Проведення комп'ютерного експерименту.



Контрольні запитання та завдання

1. Що таке модель? Наведіть приклади моделей.
2. Що таке моделювання?
3. Що таке матеріальна модель? Наведіть приклади.
4. Що таке інформаційна модель? Наведіть приклади.
5. Які ви знаєте типи інформаційних моделей?
6. Що таке математична модель? Наведіть приклади.
- 7*. Наведіть приклади інформаційних моделей населеного пункту; спортивного стадіону; класної кімнати.
8. Що таке комп'ютерна модель?
9. Вкажіть етапи побудови комп'ютерної моделі.



Питання для роздумів

1. Чи може об'єкт мати декілька моделей?
- 2**. Чи можуть різні об'єкти описуватися однією моделлю?
- 3**. Чи можна стратегічну комп'ютерну гру назвати ігровою моделлю? Обґрунтуйте свою думку.

§20.

Реалізація математичних моделей

У цьому параграфі ми розв'яжемо задачі, які вимагають побудови математичних моделей та їхньої реалізації з допомогою комп'ютера.

Задача. Необхідно пофарбувати стіни прямокутної кімнати. Скільки необхідно придбати банок фарби, якщо відомо, що розміри кімнати (600х400х260) см; 15% площі кімнати – вікна та двері; одна банка фарби призначена для фарбування площі 5 м².

Переходимо до етапів побудови комп'ютерної моделі.



Рис. 2.110

1 етап. Постановка задачі та її аналіз.

Нам **відомо**, що кімната є прямокутною, відомі розміри кімнати, відсоток площі, яку не потрібно фарбувати, — це площа, яку займають вікна та двері, а також площа фарбування однією банкою фарби.

Знайти, яку кількість банок фарби необхідно придбати для фарбування кімнати.

Також слід зауважити, що кількість банок має бути цілим числом.

При розрахунках слід також звернути увагу, що розміри кімнати вказано в сантиметрах, а площа витрат однієї банки фарби у квадратних метрах.

2 етап. Побудова інформаційної моделі.**Вхідні дані:**

$a = 600$ (см) = 6 (м) — ширина кімнати;

$b = 400$ (см) = 4 (м) — довжина кімнати;

$c = 260$ (см) = 2,6 (м) — висота кімнати;

$n = 15\%$ — відсоток площі, яка не потребує фарбування;

$s_b = 5$ (м²) — площа фарбування однією банкою фарби.

Вихідні дані:

k — кількість банок для фарбування кімнати.

Доповнимо інформаційну модель формулами для визначення результату, тобто створимо математичну модель розв'язання задачі.

Визначимо площу поверхні усієї кімнати за формулою:

$$s_k = 2c(a + b).$$

Для обчислення поверхні, яку потрібно пофарбувати, застосуємо формулу:

$$s_f = s_k(1 - n/100).$$

Кількість банок фарби обчислимо за формулою:

$$k = s_f / s_b.$$

В одержаному результаті відкинемо дробову частину та додамо 1.

3 етап. Розробка алгоритму дослідження моделі.

Алгоритм дослідження реалізуємо з допомогою блок-схеми (рис. 2.111).

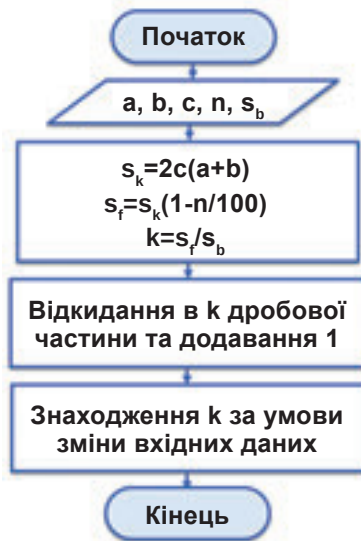


Рис. 2.111

4 етап. Розробка комп'ютерної моделі.

Заносимо дані до електронної таблиці та виконуємо обчислення.

1. До клітинки **C9** уведіть формулу:
 $=2*C5*(C3+C4)$.
2. До клітинки **C10** уведіть формулу:
 $=C9*(1-C7/100)$.
3. У клітинці **C11** уведіть формулу: $=C10/C6$.

В результаті виконання обчислень ви одержуєте значення 8,8, отже, для фарбування стін кімнати із заданими параметрами необхідно придбати 9 банок фарби.

	A	B	C
1	Фарбування кімнати		
2	Вхідні дані		
3	довжина, м	a	6
4	ширина, м	b	4
5	висота, м	c	2,6
6	витрати 1 банки фарби, м ²	s _b	5
7	площа, що не потребує фарбування, %	n	15
8	Розрахунки		
9	площа стін, м ²	s _k	
10	площа стін для фарбування, м ²	s _f	
11	кількість банок фарби, шт	k	

Рис. 2.112

5 етап. Проведення комп'ютерного експерименту.

1. Визначимо, скільки банок знадобиться, якщо фарби вистачить для 3 м², 4 м², 7,5 м².

Для цього до клітинки **C6** послідовно слід увести значення 3, 4 та 7,5. Одержані у клітинці **C11** значення округлюємо до більшого цілого значення.

2. Визначимо, скільки фарби необхідно придбати на другу кімнату, що має розміри (450×350×260) см, а вікна та двері займають 10 % площі кімнати. Значення витрат фарби 5 м².

Для цього слід змінити значення розмірів кімнати: $a=4,5$, $b=3,5$, $c=2,6$ — залишається незмінним. Як значення площі, що не потребує фарбування, ввести 10.

ВПРАВА 21



Завдання. Необхідно пофарбувати стіни прямокутної кімнати. Скільки необхідно придбати банок фарби, якщо відомо, що розміри кімнати (600×400×260) см; 15% площі кімнати — вікна та двері; одна банка фарби призначена для фарбування площі 5 м².

1. Визначте, скільки банок знадобиться, якщо фарби вистачить для 3 м², 4 м², 7,5 м².
2. Визначте, скільки фарби необхідно придбати на другу кімнату, що має розміри (450×350×260) см, а вікна та двері займають 10 % площі кімнати, значення витрат фарби 5 м².
3. Збережіть файл з розрахунками з ім'ям **Вправа_21.xlsx**.

ВПРАВА 22



Завдання. Створити комп'ютерну модель розв'язання задачі з дотриманням п'яти етапів комп'ютерного моделювання.

Збережіть файл із розрахунками з ім'ям **Вправа_22.xlsx**.

Задача. У наявності є сітка довжиною 100 м для огорожі прямокутної території складу. Визначити, яку найбільшу площу прямокутної форми можна обгородити цією сіткою. Вказати довжину та ширину ділянки з найбільшою площею. Визначити, як зміняться ці результати, якщо довжина сітки буде 80 м; 120 м.



Виконати вправу 23 ви зможете в електронному

додатку.



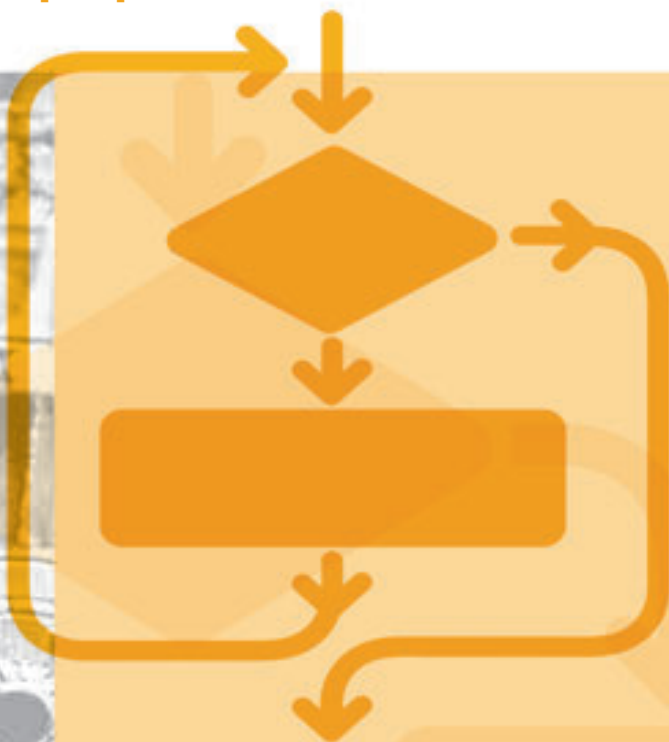
Виконайте розрахунки для випадку, якщо ділянка межує однією стороною з деякою будівлею.



Рис. 2.113

Розділ 3

Алгоритми та програми



Ви дізнаєтесь...

- що таке величини, константи та змінні
- які типи величин використовують у Scratch
- як створювати та використовувати змінні
- як використовувати вбудовані змінні
- як виконувати обчислення в середовищі Scratch
- що таке таблиці істинності і як їх використовувати
- як створювати проекти з використанням команд повторення та розгалуження
- як створювати проекти з використанням вкладених повторень та розгалужень
- як створювати комп'ютерні моделі в середовищі програмування





§21.

Повторення роботи в середовищі Scratch

21.1.

ОСНОВНІ ПОНЯТТЯ

У попередніх класах ви вже мали змогу познайомитися з роботою в середовищі Scratch. Середовище від початку розроблялося для навчання програмування дітей. Працювати у ньому нескладно і дуже цікаво. У цьому році ви навчитеся загальним прийомам і методам складання алгоритмів розв’язання різноманітних задач. Створені вами проєкти матимуть вигляд справжніх програм.

Сьогодні ми розглянемо основні принципи роботи в середовищі програмування Scratch.

Згадаємо основні терміни, пов’язані з використанням середовища Scratch.

Програми, створені в даному середовищі, прийнято називати *проєктами*.

У проєктах можна задавати дії для різноманітних персонажів, які прийнято називати *спрайтами*.

Кожен персонаж може бути розташований у різних позах — ці пози називають *образами*. Також будь-який спрайт можна розмістити на певному фоні (пейзажі, інтер’єрі кімнати, під водою тощо). Цей фон називають *сценою*.

Крім того, що спрайти і сцени можна завантажити з комп’ютера, їх також можна намалювати власноруч з допомогою вбудованого до середовища Scratch *графічного редактора*.

21.2.

СТВОРЕННЯ ВЛАСНОГО ПРОЄКТУ

Згадаємо послідовність дій для створення власного проєкту на прикладі наступного завдання.

Завдання. Створити проєкт логічної гри «Перевізник». Суть гри є такою. Перевізник мусить переправити з одного берега річки на інший вовка, козу і капусту. Через те, що човен невеликий, за один раз він може перевезти лише когось одного: вовка або козу, або капусту. Вовка не можна залишати сам на сам з козою, бо він її з’їсть, а козу — з капустою. Вовка, козу і капусту потрібно переправити з лівого берега річки на правий так, аби всі вони залишилися цілими.



1. Додайте до проєкту спрайти вовка, кози, капусти й перевізника. (Картинки ви зможете знайти в Інтернеті або скачати з файлів для уроку. При скачуванні з Інтернету не забувайте зберігати зображення у форматі **.png** — це саме той формат, який підтримує програма Scratch і дозволяє зробити фон зображення прозорим).
2. Переіменуйте їх відповідно до зображень.
3. Скориставшись інструментами вбудованого у Scratch графічного редактора, намалюйте на сцені річку. Розташуйте всіх героїв проєкту з лівого її боку.

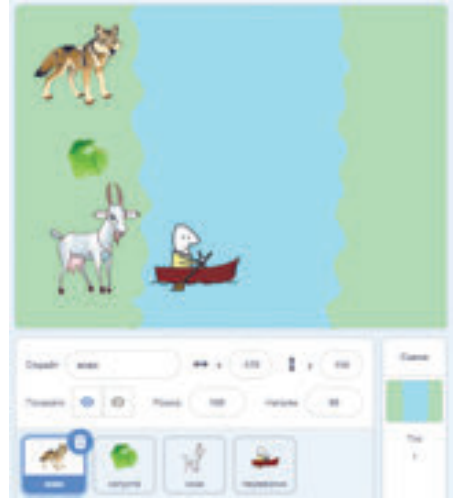


Рис. 3.1

Перед тим, як створити скрипти для наявних у проєкті спрайтів, розглянемо алгоритм, за яким ці спрайти будуть пересуватися згідно з умовою поставленої задачі.

- 1) Спочатку перевеземо козу (вовк залишиться на березі з капустою, оскільки він її не їсть) (рис. 3.2).
- 2) Перевізник повернеться на лівий берег (рис. 3.3).

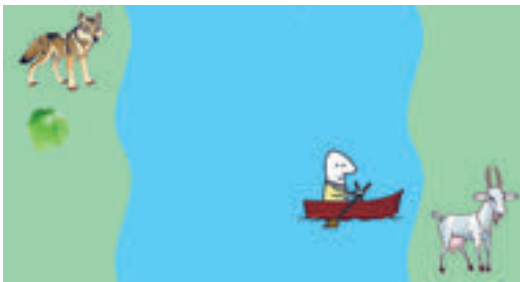


Рис. 3.2

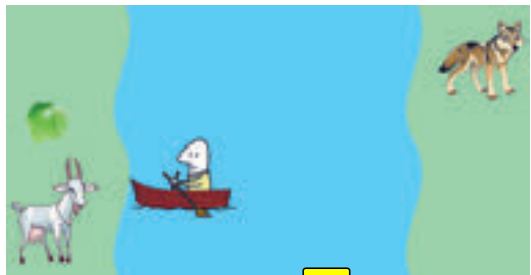


Рис. 3.3

- 3) Потім перевізник перевезе вовка (рис. 3.4).
- 4) Тепер перевізник візьме козу і поверне її на лівий берег (рис. 3.5).



Рис. 3.4

Рис. 



- 5) На лівому березі він візьме капусту і перевезе на правий берег (рис. 3.6).
- 6) Перевізник повернеться порожнім на лівий берег (рис. 3.7).

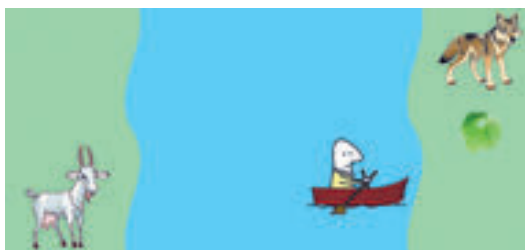


Рис. 3.6

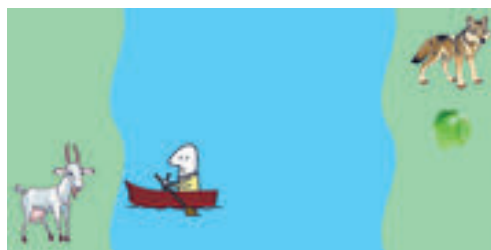


Рис. 3.7



Рис. 3.8

- 7) Останнім кроком буде перевезення кози з лівого берега на правий берег (рис. 3.8).

Алгоритм, описаний вище, є алгоритмом, за яким буде діяти перевізник, звичайно, його діям мають відповідати дії й інших спрайтів. Виконання поданого алгоритму будемо виконувати за умови натискання клавіш 1, 2, 3 і так далі, що відповідають крокам алгоритму.

Отже, перший крок алгоритму буде застосовано до таких спрайтів, як перевізник та коза.

4. Створіть скрипт для перевізника.

а) Оберіть спрайт **Перевізник**. **б)** У групі команд **Керування** оберіть команду, **коли клавішу ... натиснуто**. В списку клавіш оберіть значення **1**. **в)** Додайте команду **оповістити**. У списку введіть назву об'єкта **коза** (рис. 3.9). **г)** У групі **Рух** оберіть команду **ковзати** та встановіть відповідні значення. Їх можна побачити у нижньому куті вікна проєкту, перетягнувши вручну зображення перевізника. **г)** Для завершення події перевезення кози додайте команду **зупинити все** з категорії **Керування**.

5. Створіть скрипт для кози.

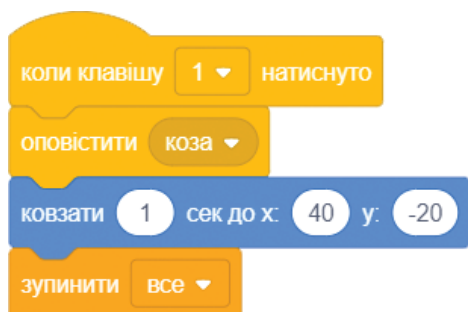


Рис. 3.9

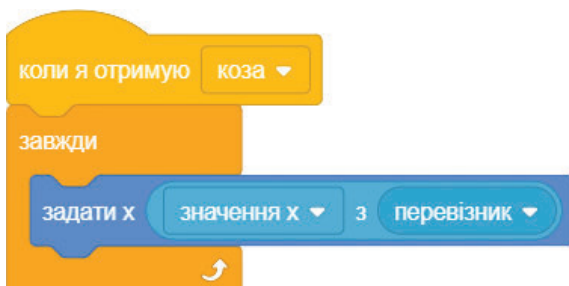


Рис. 3.10



Логіка його виконання має бути наступною: разом з переміщенням перевізника має бути переміщена коза (рис. 3.10).

Отже, коли спрайт кози отримає від перевізника сигнал **коза**, він почне рухатися разом з перевізником.

Якщо запустити проєкт на виконання, то після натискання клавіші **1** човен разом з козою переміщаються на другий берег річки. Проте коза разом з перевізником залишаються на воді.

Аби цього не траплялося і коза все ж таки опинилася на березі річки, збільшіть значення **x** у скрипті кози на 140, змінивши скрипт на наступний.

6. Створіть скрипт для виконання наступного кроку алгоритму: повернення перевізника на лівий берег.

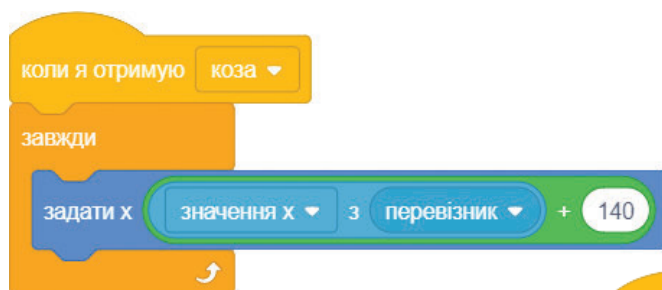


Рис. 3.11

У цьому скрипті, як і в попередньому, першою командою буде **коли клавішу ... натиснуто**. Тільки тепер указуємо, що це буде клавіша **2**. Наступна команда **поворот на 180 градусів** використовується для розвороту човна в протилежному напрямку. Як і для випадку натискання клавіші **1** додайте команди **ковзати** та **зупинити все**. Для команди **ковзати** змініть лише координату **x** (рис. 3.12).

7. Для відображення спрайта зліва направо у полі **Напрямок** оберіть параметр **Зліва-направо** (рис. 3.13).

8. Конструювання наступних скриптів виконайте аналогічно. Так, наступним кроком алгоритму є переїзд на правий берег перевізника разом з вовком.

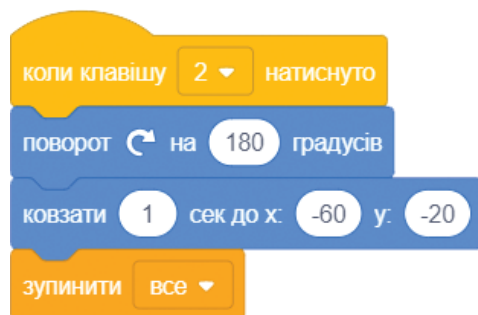


Рис. 3.12

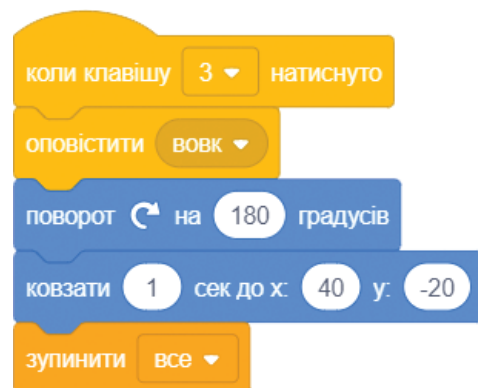


Рис. 3.13



Для створення цього скрипта створіть копію першого скрипта перевізника, змінюючи в ньому назву клавіші на **3** та оповіщення **коза** на **вовк**. Також додайте команду **поверот на 180 градусів**.

9. Створіть скрипт для спрайта **вовк**. Він буде аналогічний до спрайта **коза**. Щоб не створювати його заново, створіть копію скрипта кози, перетягнувши його на спрайт вовка. Замініть в команді **коли я отримую** слово **коза** на **вовк**.
10. Створіть скрипт повернення кози на лівий берег (рис. 3.14). Зверніть увагу, що тут необхідно буде створити ще один скрипт для спрайта **коза**, для випадку переміщення з правого берега на лівий (рис. 3.18).
11. Створіть скрипт п'ятого кроку (рис. 3.15) для перевезення капусти.

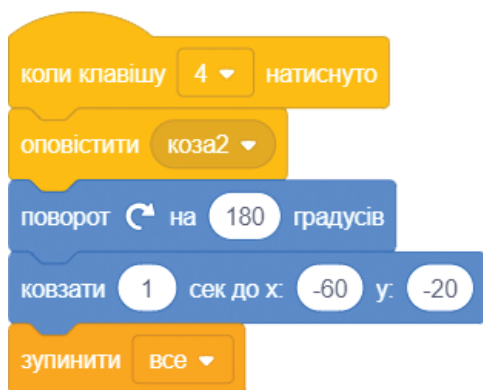


Рис. 3.14

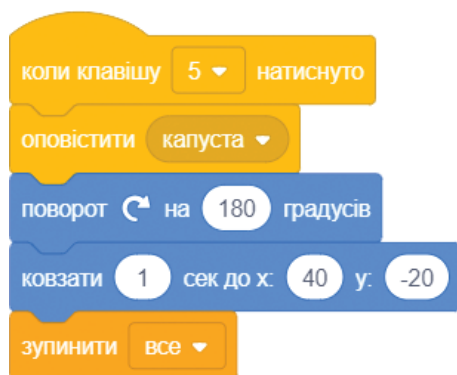


Рис. 3.15

12. Створіть скрипт для спрайта **капуста**. Він буде аналогічний до спрайта **коза** або **вовк**. Отже, для цього створіть копію скрипта кози і замініть в команді **коли я отримую** слово **коза** на **капуста**.
13. Створіть скрипт шостого кроку (рис. 3.16), а саме: повернення перевізника.
14. Створіть скрипт сьомого кроку (рис. 3.17), а саме: перевезення кози.

Ще раз проаналізуємо скрипти кожного спрайта.

Найбільше їх у перевізника. Фактично він у нас є ключовою фігурою.

Перша команда — це завжди умова натискання певної клавіші від 1 до 7.

Якщо перевізник їде не порожній, він має оповістити той спрайт, з яким він переїжджає річку.

Далі у всіх скриптах, окрім першого, він повертається на 180 градусів і ковзає до вказаної координати.

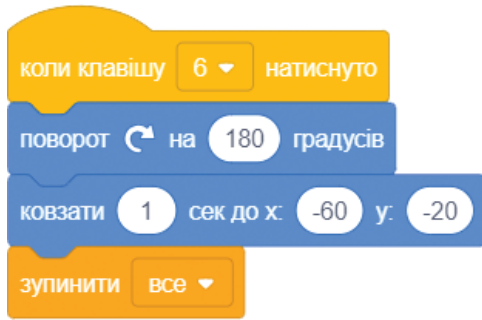


Рис. 3.16



Рис. 3.17

Команда **зупинити все** зупиняє вказаний крок алгоритму.

Що стосується скриптів інших спрайтів, вони фактично скрізь однакові, крім значення команди **коли я отримую**. В ній вказано, що в даний момент цей спрайт має рухатися разом з перевізником (рис. 3.18).

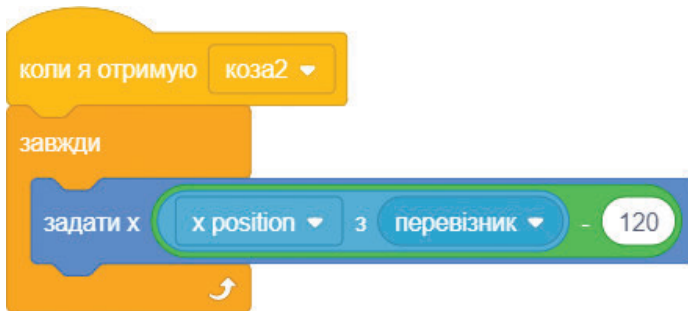


Рис. 3.18

Також відмінним є скрипт кози, що відповідає її поверненню на лівий берег.

ВПРАВА 1



Завдання. Створити проєкт логічної гри «Перевізник».

Збережіть файл проєкту з ім'ям **Вправа_1**.

ВПРАВА 2



Завдання. Створити проєкт логічної гри «Перевізник», у якому запрограмувати другий варіант перевезення кози, вовка та капусти.

Збережіть файл проєкту з ім'ям **Вправа_2**.



Висновки

Scratch — це середовище візуального програмування, створене для дітей.

Програми, створені в даному середовищі, прийнято називати *проєктами*.

У проєктах можна задавати дії для різноманітних персонажів, які прийнято називати *спрайтами*.

Будь-який спрайт можна розмістити на певному фоні — цей фон називають *сценою*.

Команда, яку виконує спрайт, називається *скриптом*.



Контрольні запитання та завдання

1. Що називається проєктом?
2. Що таке спрайт?
3. Що таке образ?
4. Що таке сцена?
5. Що таке скрипт?
6. Як викликати вбудований до Scratch графічний редактор?
7. Як запустити проєкт на виконання?

§22.

Типи даних

22.1.

ПОНЯТТЯ ТИПУ ДАНИХ

Одне з основних понять програмування, що визначає структуру даних, їхні можливі значення і те, як їх буде оброблено й подано в пам'яті комп'ютера — це **тип даних**.

01101110110



Число
чи текст?

Рис. 3.19

Комп'ютер, отримуючи двійковий код, що не має жодних ознак, не здатен визначити його природу та відрізнити, скажімо, числа від тексту (рис. 3.19). Указуючи в програмі тип даних, ми повідомляємо комп'ютер про те, яким чином він має перетворювати інформацію у відповідній ділянці пам'яті.

Поняття типу даних існує майже в усіх мовах програмування, зокрема у Скретчі (Scratch).



Створюючи у цій програмі проєкти, ви прагнете, щоб їхні сценарії виконувались якомога точніше: спрайти опинялися в потрібному місці в потрібний час і промовляли фрази, які ви вкладаєте в їхні уста.

Усе це та багато іншого спрайти роблять завдяки блокам-командам, що передають у програму вказані у спеціальних полях дані.

У Скретчі — це дані трьох типів: числа, рядки та логічні дані.

22.2.

ЧИСЛОВІ ДАНІ

Розглянемо способи подання та використання у Скретчі числових даних.

Числові дані у Скретчі вводять в овальні поля блоків-команд.

Це можуть бути як додатні, так і від’ємні цілі числа та десяткові дроби.

Такі поля мають також блоки-репортери. Репортерами ми називатимемо блоки овальної форми, через те що вони так само, як репортери газет і журналів, збирають важливі для проєктів дані — різні числові та текстові значення. На рис. 3.21 наведено приклади таких блоків.

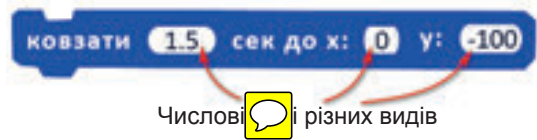


Рис. 3.20



Рис. 3.21

Розглянемо для прикладу, які дані можна вводити до команди **говзати** з категорії **Рух** (рис. 3.20).

Тут можна ввести будь-які додатні та від’ємні цілі числа. Також тут можна вводити дробові числа. Зверніть увагу, що дробову частину числа слід відокремлювати *крапкою*. Якщо ж спробувати ввести тут текст, то Скретч нам не дозволить цього зробити.

Над числами в Скретчі можна виконувати різноманітні дії. Зверніть увагу на блоки-репортери, що належать до категорії **Оператори**. Додаючи їх до своїх скриптів, ви зможете здійснювати з допомогою програми Скретч такі операції над числами:



— виконувати арифметичні дії (додавання, віднімання, множення та ділення);

— обчислювати математичні функції;

— округлювати дріб до найближчого цілого числа;

— знаходити остачу від ділення цілих чисел;

— отримувати випадкове число з будь-якого діапазону значень.



Рис. 3.22

Розглянемо роботу деяких операцій.

Для додавання двох чисел 3 і 4 необхідно створити скрипт, представлений на рис. 3.22. Щоб знайти результат обчислення, достатньо клацнути двічі на скрипті. Після чого буде виведено результат — 7.

22.3.

ДАНИ ТЕКСТОВОГО ТИПУ

Рядковий тип даних у програмуванні — це тип даних, значеннями яких є будь-яка послідовність символів (літер, цифр, розділових знаків) довільної довжини.

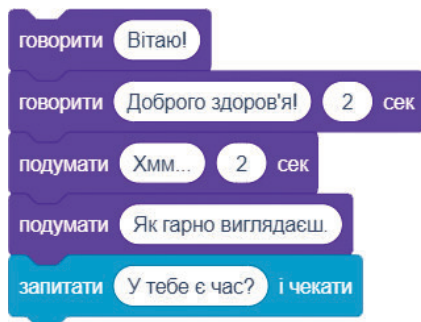


Рис. 3.23

Скретч дає змогу:

— об'єднувати рядки (це команда);

— дізнаватися про кількість символів у рядку (команда);

— визначати символ у рядку за вказаним номером (команда

);

— визначати, чи містить заданий текст деякий вказаний фрагмент тексту (команда).

Розглянемо використання числових та текстових даних, створюючи проєкт невеличкої історії спілкування двох дівчаток.

У Scratch для введення рядкових даних існує п'ять команд: чотири в категорії **Вигляд** і одна в категорії **Датчики**. Саме вони і представлені на рис. 3.23.

З допомогою них тут зазвичай передають різні повідомлення і репліки героїв проєктів, а також ставлять запитання до користувачів програми (якщо проєкт потребує їхніх відповідей).

Над рядками також можна виконувати операції, щоправда, не так багато, як над числами.



ВПРАВА 3



Завдання. Створити проєкт спілкування двох дівчаток, у якому ви зможете попрактикуватися у введенні текстових даних і поспостерігати за тим, як у Scratch здійснюється обчислення з числовими даними.

1. Додайте до проєкту два спрайти дівчаток з бібліотеки програми Скретч і розташуйте їх один проти одного. Дівчинку, розташовану ліворуч, умовно назовемо Оленка, а дівчинку, розташовану праворуч, — Світланка.
2. Складіть скрипт для Світланки.
 - а) Додайте команду **коли прапорець натиснуто**, що має запускати проєкт на виконання.
 - б) Нижче додайте скрипт:



Рис. 3.24

- в) Наступна команда має вивести повідомлення «Отже, в мене є 5 цукерок». Це буде скрипт:

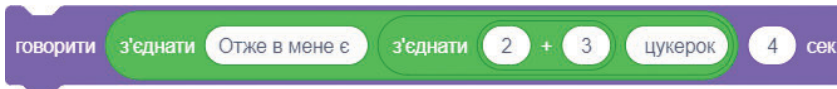


Рис. 3.25

- г) Завершіть висловлення першої дівчинки командою **оповістити Оленка** (рис. 3.26).



Рис. 3.26

3. Складіть скрипт для Оленки, дівчинки, що розташована ліворуч.

- а) Додайте перший скрипт **коли я отримую Оленка**, що дозволить спрайту із зображенням Оленки розпочати діалог (рис. 3.27).



Рис. 3.27

- б) Далі Оленка має запитати «Поділимося?» і повідомити кількість цукерок, у разі їхнього розподілу навпіл. Отже, програма поділить 5 на 2 і виведе повідомлення «Кожному виходить по 2.5 цукерки». Цей вислів виконайте з допомогою скрипта:

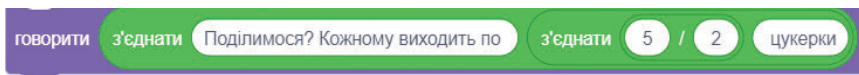


Рис. 3.28

- в) Завершіть висловлення та перейдіть до Світланки завдяки команді **оповістити Світланка** (рис. 3.29).

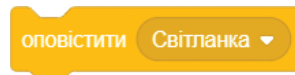


Рис. 3.29

4. Наступний скрипт створіть для Світланки.

- а) Додайте команду **коли я отримую Світланка**.
- б) Далі додайте команду, в якій спрайт виводить повідомлення «Ні, візьmemo цілі цукерки, а решту віддамо Сергію».

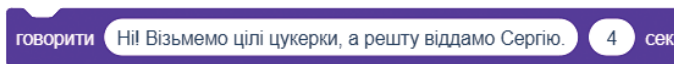


Рис. 3.30



- в) Наступною має бути виведена фраза «Сергій отримає 1 цукерку» (рис. 3.31). Сформуйте її з допомогою скрипта:

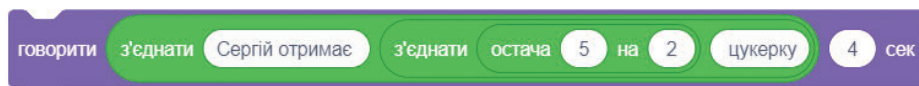


Рис. 3.31

- г) Завершіть виконання проєкту командою **зупинити все**.
- Запустіть проєкт та перевірте правильність його виконання.
 - Збережіть проєкт, надавши ім'я **Вправа_3**.

ВПРАВА 4



Завдання. Самостійно створити проєкт діалогу між однокласниками. Замість напису <формула> мають бути використані блоки для виконання відповідних обчислень.

Василь: Привіт!

Василь: Яку оцінку ти отримала сьогодні з математики?

Марійка: Спробуй порахувати сам.

Марійка: Поділи поточний рік на 100 та округли результат до цілого.

Марійка: Відними від одержаного результату 8.

Василь: Ти отримала <формула> балів.

Василь: А тепер спробуй визначити мою оцінку.

Василь: Знайди остачу від ділення кількості місяців року на кількість днів у тижні і помнож одержаний результат на 2.

Марійка: Твоя оцінка <формула> балів.

Збережіть файл з ім'ям **Вправа_4**.



Висновки

Типи даних — це одне з основних понять програмування, що визначає структуру даних, їхні можливі значення і те, як їх буде оброблено й подано в пам'яті комп'ютера.

В середовищі Скретч використовуються дані трьох типів: числа, текст, логічні дані. Числовими даними можуть бути цілі числа та десяткові дробі. Для встановлення розділового знака у дробових числах використовується «крапка». Рядковий тип даних — це тип даних, значеннями яких є будь-яка послідовність символів (літер, цифр, розділових знаків) довільної довжини.



Контрольні запитання та завдання

- Які типи даних використовуються в Scratch?
- Які види чисел можна вводити до числових полів?
- Наведіть приклади команд для обробки числових даних.
- Наведіть приклади команд для обробки текстових даних.



§23.

Поняття величин. Константи та змінні

23.1.

ПОНЯТТЯ ВЕЛИЧИН

Працюючи в середовищі Скретч, раніше ви надавали різні команди спрайтам, які дозволяли змінювати їхні властивості чи виконувати певні дії. Виявляється, Скретч також допомагає розв'язувати математичні задачі. Це можливе завдяки використанню величин різного типу.

Величина в інформатиці — це об'єкт, з яким пов'язана деяка множина значень. Величину можна порівняти з наповненою скринькою (рис. 3.32).

Під час створення Scratch- проєктів вам не раз доводилося вводити значення в поля блоків-команд або вибирати їх зі списків.

Роблячи це, ви, мабуть, помітили, що у Scratch ті величини, значення яких уводить користувач під час створення скриптів, залишаються незмінними протягом виконання проєкту.



Рис. 3.32

23.2.

ПОНЯТТЯ КОНСТАНТИ

Величини, значення яких залишаються сталими під час виконання програми, називають **константами**.

У Scratch константи задають самі користувачі під час створення проєктів, уводячи їхні значення у прямокутні й овальні поля блоків-команд.

Так, у скрипті наведено приклади (рис. 3.33) використання констант, які обрано зі списку, та констант, уведених безпосередньо користувачем. Константами можуть бути як числа, так і рядки символів. Зверніть увагу, що константи у Scratch не мають імен.

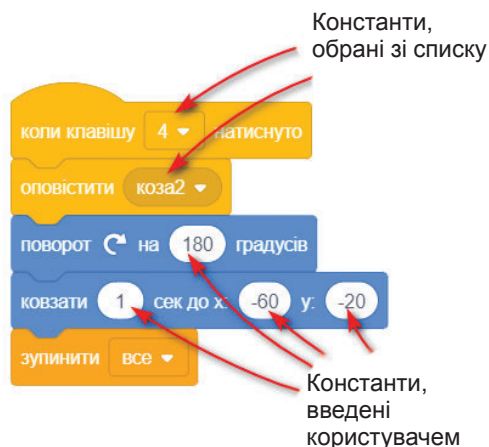


Рис. 3.33



У математичних рівняннях також використовують константи. Наприклад, у рівнянні $y = 3x + 7$ числа 3 і 7 є константами. Значення таких констант цілком залежить від волі людини, яка складає рівняння.

23.3.

ПОНЯТТЯ ЗМІННОЇ

На відміну від константи, **змінна** є величиною, яка під час виконання програми може набувати різних значень.

Змінна є складовою програми, яка має не лише значення, а й ім'я.

Змінну можна порівняти з коробкою, що має своє ім'я, а значення змінної — це вміст коробки (рис. 3.34).

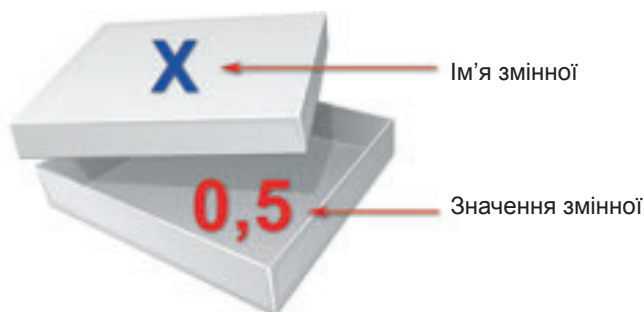


Рис. 3.34

Ви маєте знати, що поняття змінної використовують не лише в математиці та програмуванні, але й у природничих науках і техніці. Прикладами змінних можуть бути температура повітря, параметр функції (математичної, фізичної чи будь-якої іншої) тощо (рис. 3.35).

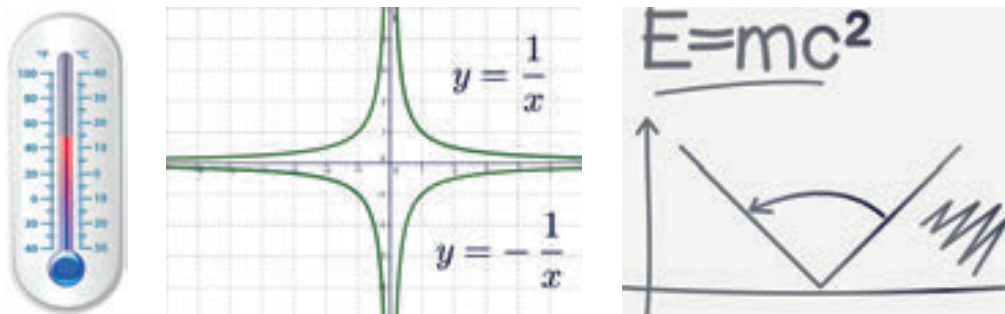


Рис. 3.35

Рівняння $y = 3x + 7$ також містить змінні. Змінними тут є поіменовані величини (x та y), що можуть набувати різних числових значень.



23.4.

ВИКОРИСТАННЯ ВБУДОВАНИХ ЗМІННИХ

У середовищі Скретч можна створити власні змінні або скористатися так званими *вбудованими змінними*.

До вбудованих змінних, їх частіше називають репортерами, належать блоки, що представлені у вигляді овалів, наприклад, **гучність**, **таймер**, **відповідь** (рис. 3.36).

Розглянемо використання одного із таких репортерів при виконанні наступної вправи.



Рис. 3.36

ВПРАВА 5



Завдання. У даній вправі ви маєте відтворити діалог двох спрайтів. Перший запитає другого: «Привіт! Коли ти звільнишся?». Другий має увести деякий час, наприклад, «12:15». Перший звернеться до другого «Гаразд, тоді до зустрічі о 12:15.». Звичайно, якщо другий персонаж укаже інший час, то в повідомленні буде той час, який укаже другий спрайт.

1. Розташуйте один навпроти одного два спрайти, обравши їх на власний розсуд.
2. Створіть скрипт для першого персонажа.

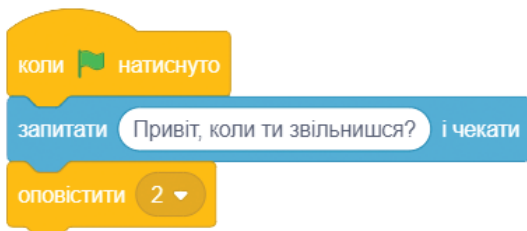


Рис. 3.37

3. Створіть скрипт для другого спрайта.

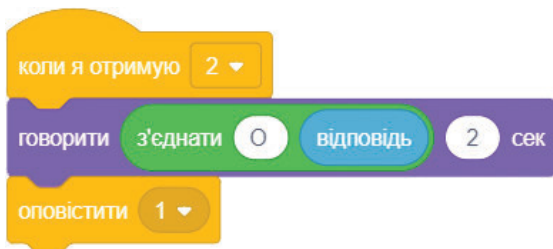


Рис. 3.38

4. Створіть ще один скрипт для першого спрайта.

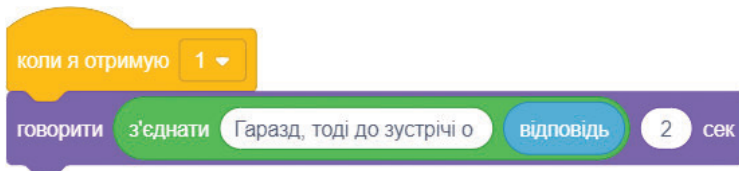


Рис. 3.39



Виконати вправи 6 та 7 ви зможете в електронному додатку.

- Запустіть проєкт на виконання, натиснувши зелений прапорець. У виносці першого спрайта має з'явитися повідомлення «Привіт! Коли ти звільнишся?». В нижній частині вікна проєкту з'явиться поле, в якому слід увести час (рис. 3.40).

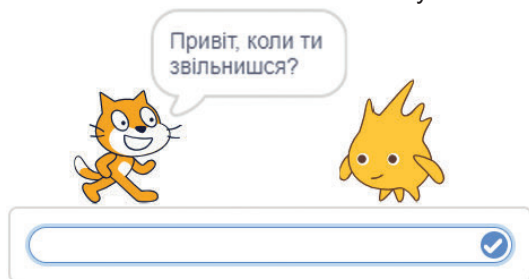


Рис. 3.40

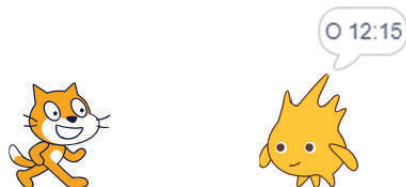


Рис. 3.41



Рис. 3.42

- Уведіть час і для підтвердження відповіді натисніть кнопку з галочкою. Також можна натиснути клавішу **Enter**. Після цього другий спрайт має повідомити час зустрічі (рис. 3.41).
- Після чого перший спрайт повідомляє час зустрічі (рис. 3.42).
- Збережіть файл з ім'ям **Вправа_5**.

ВПРАВА 8



Завдання. Створити проєкт, у якому кіт має накреслити відрізок, довжину якого вводить користувач. Після чого кіт має повідомити довжину побудованого відрізка.

- Для побудови відрізка та виведення повідомлення скористайтеся зазначеними нижче командами, самостійно визначивши їхню послідовність розташування.



- Самостійно додайте блоки, які при повторному запуску проєкту дозволять накреслити новий відрізок.
- Збережіть файл з ім'ям **Вправа_8**.



Висновки

Величина — це об'єкт, з яким пов'язана деяка множина значень.

Величини, значення яких залишаються сталими під час виконання програми, називають *константами*.

У Scratch значення констант уводять у прямокутні й овальні поля блоків-команд під час конструювання проєкту.

Величини, які під час виконання програми можуть набувати різних значень, називають *змінними*.



Контрольні запитання та завдання

1. Що таке величина?
2. Що таке константа?
3. Які види констант використовують у Scratch?
4. Що таке змінна?
5. Які вбудовані змінні використовуються у Scratch? Поясніть їхні призначення.

§24.

Присвоєння значень величинам

24.1.

СТВОРЕННЯ ЗМІННИХ

Репортери, які ми розглядали в попередньому параграфі, передають до проєкту значення зарезервованих у програмному середовищі Scratch змінних.

А що робити, коли серед вбудованих змінних немає потрібних?

У цьому випадку Scratch дає змогу створювати власні змінні. Для цього тут передбачена спеціальна категорія, що так і зветься, — **Змінні**.

Для створення нової змінної в даній категорії достатньо натиснути кнопку **Створити змінну**.

У вікні, що з'явилося, слід увести ім'я змінної, наприклад, **а**. Також слід обрати перемикач **Для усіх спрайтів** чи **Тільки для цього спрайта**.

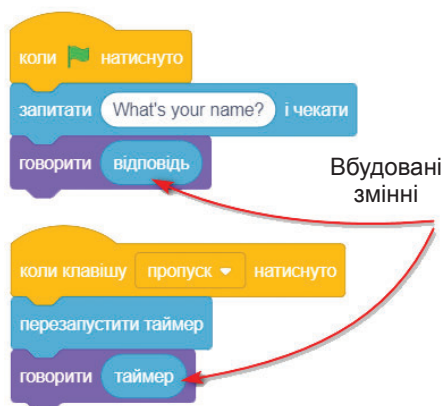


Рис. 3.43

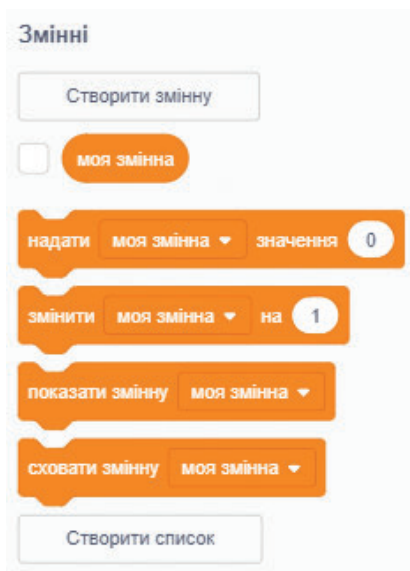


Рис. 3.44

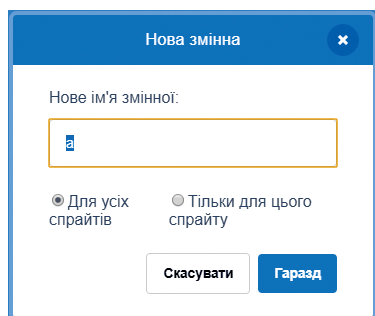


Рис. 3.45. Вікно створення нової змінної

Перший перемикач дає змогу створити **глобальну** змінну, таку, що її бачитимуть усі об'єкти проєкту, а другий — **локальну**, яку бачитиме лише поточний об'єкт.

Як наслідок, у палітрі команд з'явиться ця змінна, а на сцені проєкту ви побачите монітор змінної.

До речі, для відображення або приховування моніторів змінних або репортерів слід установлювати або знімати прапорці біля їхніх назв в області команд.

Створену змінну також можна легко видалити, клацнувши у контекстному меню змінної команду **Вилучити змінну**. Також з допомогою контекстного меню можна перейменувати змінну, обравши відповідну команду.

Зверніть увагу на правила надання імен змінним.

1. Ім'я змінної може складатися з будь-яких букв, цифр та символів.
2. Scratch розрізняє великі та маленькі букви при написанні імен змінних.

Використання деяких команд, які можна застосовувати до змінних, розглянемо, виконуючи наступну вправу.

ВПРАВА 9



Завдання. Створити проєкт знаходження площі прямокутної ділянки, довжину і ширину якої має ввести користувач. Для реалізації поставленої задачі створимо дві змінні a і b , які одержать значення довжин сторін прямокутника. Значення цих змінних має ввести користувач з клавіатури. Також створимо змінну s , яка буде результатом знаходження площі прямокутника.



Рис. 3.46

1. Створіть змінну a .
 - а) Перейдіть до категорії **Змінні**.
 - б) Натисніть кнопку **Створити змінну**.
 - в) Введіть її назву a .
 - г) Натисніть **Гаразд**.



2. Аналогічно створіть змінні **b** і **S**.
3. Створіть скрипт, з допомогою якого змінній **a** буде надано значення, введене користувачем з клавіатури.

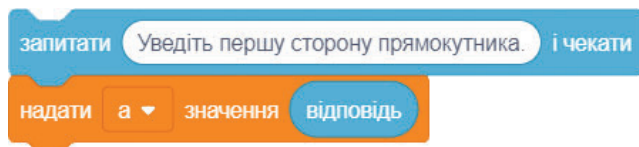


Рис. 3.47

Команда **Запитати ... і чекати** допоможе запам'ятати число, яке користувач уведе з клавіатури. Скрипт **надати a значення відповідь** дозволить змінній **a** надати значення відповіді, яку введе з клавіатури користувач.

4. Аналогічний вид буде мати скрипт для надання значення змінній **b**. Для його створення скопіюйте створений уже скрипт для змінної **a** та замініть напис-запрошення, а у списку команди **надати a значення відповідь** змінну **a** замініть на **b**.
5. Щоб знайти значення площі прямокутника за двома сторонами, необхідно значення довжин цих сторін перемножити: $S=a*b$. Сформуйте скрипт надання змінній **S** значення добутку **a** і **b** (рис. 3.48).
6. Створіть скрипт виведення результату обчислення (рис. 3.49).



Рис. 3.48



Рис. 3.49

7. Додайте до створеного скрипта команду **коли прапорець натиснуто**.
8. Перевірте роботу створеного проєкту. Натисніть прапорець та введіть перше число 5, друге 7. В результаті ви маєте одержати результат 35.
9. Поекспериментуйте з введенням інших чисел.
- 10*. Самостійно додайте команди побудови прямокутника із заданими користувачем значеннями довжини та ширини.
11. Збережіть проєкт з ім'ям **Вправа_9**.

Зі створеного скрипта програми (рис. 3.50) видно, що він фактично складається з трьох частин: блоку введення даних; блоку виконання обчислень; блоку виведення результату.

Як правило, така структура є типовою для створення проєктів на виконання обчислень.

Створюючи проєкти з використанням змінних, ви маєте пам'ятати, що:

- початкове значення створеної користувачем змінної завжди дорівнює 0;
- на кожному кроці виконання програми змінна може мати лише одне значення;
- після набуття змінною нового значення її попереднє значення втрачається.

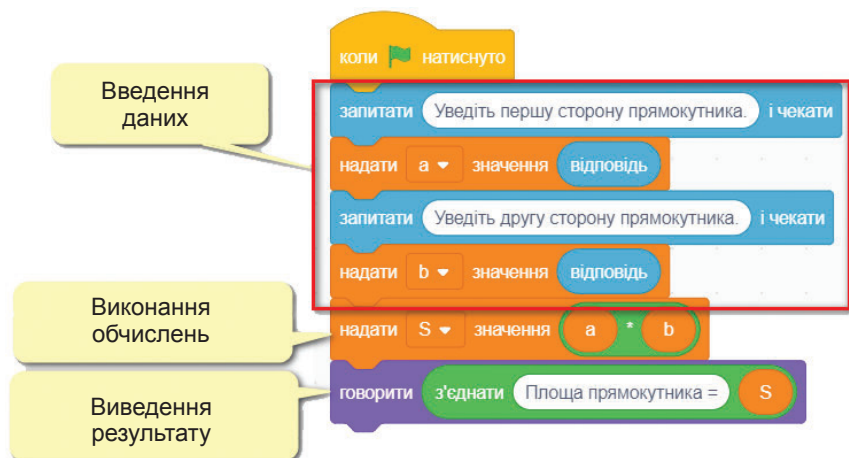


Рис. 3.50

24.2.

КОМАНДА ПРИСВОЄННЯ

Розглянемо детальніше команди, які можна використовувати для змінних.

Користувач може надавати змінним потрібні значення. Для цього у Скретчі є спеціальна команда **надати ... значення ...**.

У програмуванні надання змінній певного значення називають **операцією присвоювання**.

З її допомогою змінній можна присвоїти значення даних різних типів.

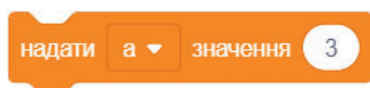


Рис. 3.51

Можна змінній надати значення деякого числа. Так, наприклад, для надання змінній **a** значення **3** необхідно скористатися командою (рис. 3.51).

Команду **надати <змінна> значення 0** часто використовують за потреби обнулити значення змінної (рис. 3.52).

Можна також змінній надати значення випадкового числа із заданого діапазону (рис. 3.53).



Рис. 3.52

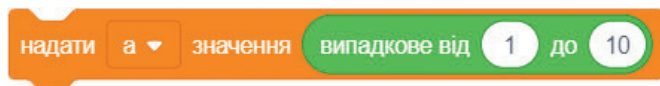


Рис. 3.53

Можна надати значення, що є результатом обчислення — наприклад, додавання (рис. 3.54).

Можна надати значення тексту (рис. 3.55).



Рис. 3.54



Рис. 3.55

Можна надати значення іншої змінної, створеної користувачем, — наприклад, надати змінній **a** значення змінної **b** (рис. 3.56).

Можна надати значення, введене користувачем з клавіатури (рис. 3.57).



Рис. 3.56



Рис. 3.57

Значення змінних можна змінювати, використовуючи команду **змінити на**. Це можна робити і під час створення проєкту, і в процесі його виконання (рис. 3.58).

Далі розглянемо вправу, в якій ви створите проєкт, який навчить kota надавати змінній значення результату обчислення виразів.

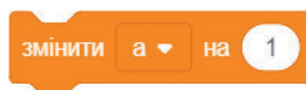


Рис. 3.58

ВПРАВА 10



Завдання. Створити проєкт обчислення значення виразу $y = \frac{7+2}{9-4} \cdot 5$.

1. Створіть змінну **y**.
2. Оскільки математичний вираз містить багато дій, розпочніть його складати з оператора, що виконуватиметься останнім, тобто з дії множення, поступово вкладаючи у нього решту операторів, аж до першого.
 - а) Спочатку вкажіть, що ви щось будете множити на 5 (рис. 3.59).
 - б) Додайте блок ділення на місце першого множника (рис. 3.60).
 - в) До блоку ділення на місце діленого додайте блок додавання, а на місце дільника — блок віднімання (рис. 3.61).
 - г) Введіть числа, що містить вираз (рис. 3.62).



Рис. 3.59



Рис. 3.60



Рис. 3.61



Рис. 3.62

3. Додайте одержаний вираз до команди **надати a значення** (рис. 3.63).



Рис. 3.63



4. Додайте команду **говорити** та додайте у її блок змінну **а**.
5. Зверху додайте команду **коли прапорець натиснуто**.
6. Запустіть проект. В результаті кіт має повідомити результат обчислення виразу.

Отже, при створенні скрипта, в якому необхідно виконати обчислення, слідкуйте за послідовністю додавання блоків обчислення. Керуйтеся правилом, що спочатку слід додати блок, дію якого потрібно виконати останньою, потім блок, дія якого буде передостанньою, і так просувайтесь з кінця в початок до першої дії.

ВПРАВА 11



Завдання. Створити проект побудови квадрата за різних значень довжини його сторони, значення якої має обрати користувач.

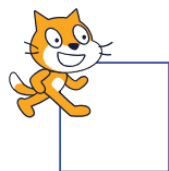


Рис. 3.64

1. Створіть змінну **а**.
2. Перетворіть її монітор на слайдер. Для цього клацніть на ньому правою кнопкою миші та оберіть значення слайдера.

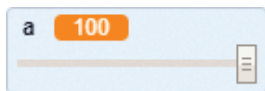


Рис. 3.65

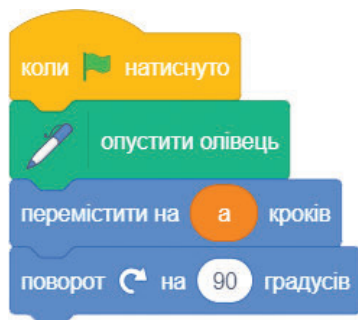


Рис. 3.66

3. Перетягніть повзунок слайдера у праве крайнє положення. В результаті змінна **а** має одержати значення 100 (рис. 3.65).
4. Створіть для кота скрипт побудови сторони квадрата та наступного повороту її на 90 градусів (рис. 3.66).
5. Запустіть створений скрипт подвійним клацанням миші. В результаті кіт має намалювати одну сторону і здійснити її поворот.
6. Скопіюйте останні дві команди ще 3 рази, після цього кіт має намалювати квадрат із довжиною сторони 100 точок.
7. Змініть значення змінної **а**. Для цього в контекстному меню слайдера оберіть команду **change slider range** і у вікні **Change slider range** введіть для параметра **Maximum value** значення **200**. Перетягніть повзунок у крайнє праве положення.
8. Запустіть проект на виконання. В результаті квадрат має стати більшим.
9. Поекспериментуйте з іншими значеннями змінної **а**.
- 10*. Змініть проект таким чином, аби значення змінної **а** користувач вводив з клавіатури.
11. Збережіть проєкт з ім'ям **Вправа_11**.



ВПРАВА 12



Завдання. Створити проекти розв'язування наступних задач. Надати проектам інтерактивності, використовуючи різні спрайти і налаштувавши взаємодію між ними. Збережіть проекти з іменами **Вправа_12_1**, **Вправа_12_2**, **Вправа_12_3** та **Вправа_12_4** відповідно.

Задача 1. Учень придбав N зошитів за ціною A грн за штуку і таку саму кількість обкладинок до них по B грн кожна. Визначте вартість усієї покупки. Переведіть отримане значення вартості у копійки.

Задача 2. Дівчинка назбирала K кошиків винограду по M кг N г у кожному кошику та A кошиків вагою B кг C г кожний. Скільки всього винограду назбирала дівчинка? (Аби задачу було легше розв'язувати, переведіть вагу винограду у грами.)

Задача 3. Робітник виготовив комплект деталей — A великих і B малих. Відомо, що велика деталь удвічі важча за малу, а весь комплект має масу M . Визначте маси малої та великої деталей.

Задача 4. Два автомобілі проїхали разом S км. Відомо, що один з них проїхав на 20 км більше. Знайдіть швидкість кожного автомобіля, якщо на весь шлях кожний витратив T год.



Висновки

У середовищі Scratch можна створити власні змінні. Це можна зробити в категорії **Змінні**, обравши команду **Створити змінну**.

У програмуванні надання змінній певного значення називають **операцією присвоювання**. У середовищі Scratch для цього використовують команду **надати ... значення ...**.

Змінним можна присвоїти числове або текстове значення, значення числового виразу, значення іншої змінної або значення, уведеного користувачем з клавіатури.

При створенні скрипта для виконання обчислень блоки обчислень слід додавати від останньої дії до першої.



Контрольні запитання та завдання

1. Опишіть послідовність дій для створення змінної.
2. Що таке локальна та глобальна змінні?
3. Укажіть правила створення змінних.
4. З допомогою якої команди присвоюють значення змінній?
5. З допомогою якої команди змінюють значення змінної?
6. Що таке монітор змінної? Опишіть його стани.
7. З допомогою якої команди змінній можна надати значення, введені користувачем?



§25.

Практична робота №5. «Розв'язування задач на обчислення»

Ви вже, мабуть, зрозуміли, що Scratch може стати вам у пригоді не тільки під час вивчення інформатики, а й на інших уроках, зокрема на уроці математики. Адже з його допомогою можна не лише створювати інтерактивні історії, мультики, ігри, презентації, а й організовувати тематичні опитування, розв'язувати різні задачі, обчислювати математичні рівняння тощо. Виконуючи завдання даної практичної роботи, ви розв'яжете різноманітні математичні задачі.

Завдання. Створити проекти розв'язання задач.

Задача 1. Двоє дівчаток збирали в лісі ягоди. Оленка принесла додому X кг ягід, а Тетянка на 2 кг більше. Скільки ягід дівчатка назбирали разом?

Аби проєкт не нагадував звичайну математичну задачу, зробіть його інтерактивним.

Проєкт має працювати таким чином.

Програма просить увести користувача вагу ягід, яких збрала Оленка.

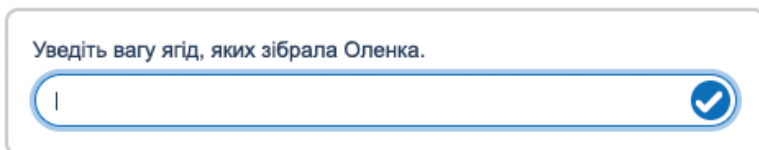


Рис. 3.67

Потім Оленка повідомляє, скільки вона збрала ягід, а Тетянка повідомляє, скільки ягід збрала вона. Сергій підраховує загальну кількість ягід та повідомляє її.

1. Додайте до проєкту три спрайти: дві дівчинки та хлопчик. Змініть їхні назви відповідно до імен дітей.
2. Оберіть для цих спрайтів сцену за власним бажанням.

3. Створіть три змінні x , y та z , які будуть набувати значень ваги ягід, яких збрала Оленка, вагу ягід, яких збрала Тетянка, та загальну вагу ягід, яких дівчата збрали разом.
4. Створіть для сцени скрипт, з допомогою якого буде введено вагу ягід, зібраних Оленкою. Останньою додайте команду переходу до об'єкта Оленка (рис. 3.68).

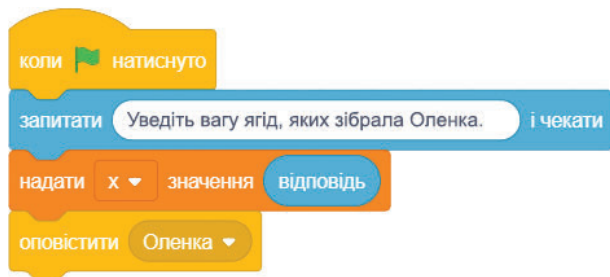


Рис. 3.68. Скрипт для сцени



5. Створіть скрипт для Оленки. Тут спрайт Оленки після одержання Оленка має повідомити зібрану нею вагу ягід та здійснити перехід до об'єкта Тетянка (рис. 3.69).

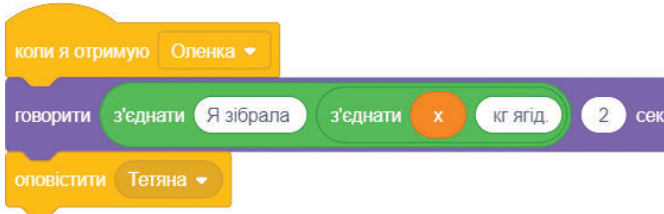


Рис. 3.69

6. Створіть скрипт для Тетянки. Вона має виконати обчислення, збільшуючи значення змінної x на 2, та присвоїти це значення змінній y . Після чого повідомити одержаний результат та здійснити перехід до спрайта Сергій (рис. 3.70).

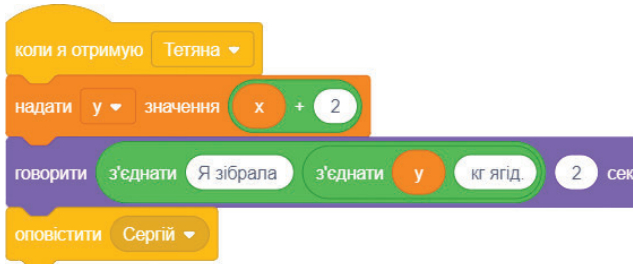


Рис. 3.70

7. Створіть скрипт для Сергія. Він має виконати додавання змінних x та y , присвоїти це значення змінній z та вивести повідомлення про загальну вагу ягід, зібраних дівчатками (рис. 3.71).

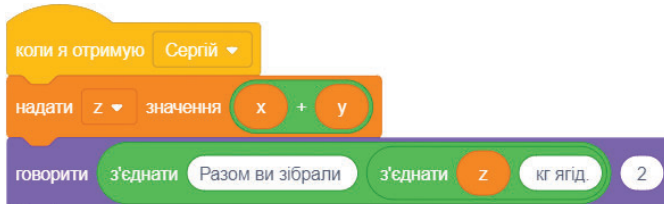


Рис. 3.71

8. Збережіть проєкт з ім'ям **ПР5_1**.

У математиці досить популярним є розв'язування геометричних задач. Наступний проєкт є розв'язанням однієї з таких задач.

Задача 2. Квадрат і прямокутник мають однакові периметри. Знайдіть сторону квадрата, якщо довжина та ширина прямокутника дорівнюють відповідно a м та b м (рис. 3.72).

1. Позначимо довжину сторони квадрата змінною x .
2. Для знаходження периметра прямокутника необхідно скористатися формулою: $P=(a+b)*2$, для знаходження периметра квадрата — формулою $P=4*x$ (рис. 3.73).

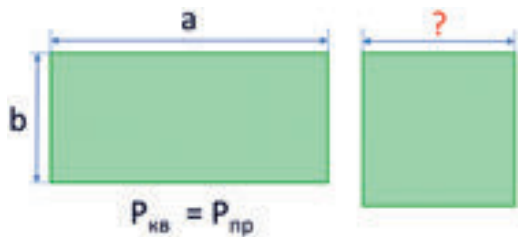


Рис. 3.72

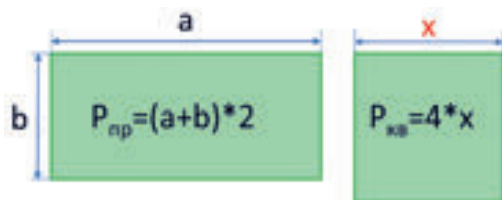


Рис. 3.73

$$P_{\text{пр}} = 4 * x \rightarrow x = P_{\text{пр}} / 4$$

Рис. 3.74

Звідси, оскільки периметри у даних фігур є рівними, то замість периметра квадрата підставимо значення периметра прямокутника, який ми знайдемо за двома відомими сторонами (рис. 3.74).

Використаємо виведені формули для створення проєкту.

1. Створіть змінні, необхідні для реалізації проєкту. Це будуть змінні **a**, **b** — довжини сторін прямокутника, **x** — довжина сторони квадрата, **P** — периметр.
2. Додайте команди введення значень сторін прямокутника.
3. Знайдіть периметр прямокутника за формулою $P = (a+b) * 2$.
4. Знайдіть сторону квадрата за формулою $x = P / 4$.
5. Виведіть результат обчислень.
6. Додайте команду **коли прапорець натиснуто**. В результаті ви маєте отримати такий скрипт, як на рис. 3.75.



Рис. 3.75

7. Для перевірки роботи проєкту введіть **a = 10**, **b = 20**. В результаті ви маєте одержати результат **15**.
8. Знайдіть та виведіть площу квадрата.
9. Збережіть проєкт з ім'ям **ПР5_2**.

**Виконати задачі**

3–7 ви зможете
в електронному
додатку.





§26.

Логічні величини. Вказівки розгалуження

26.1.

ПОНЯТТЯ ЛОГІЧНОГО ВИРАЗУ

У минулому році ви дізналися про існування таких конструкцій алгоритмів, як розгалуження та цикли, і про те, що саме вони дають змогу керувати порядком виконання програм. У цьому параграфі ми детальніше розглянемо використання розгалуження.

У реальному житті нам часто доводиться погоджуватися з тим чи іншим твердженням, подією або фактом чи заперечувати їх. Так, твердження «Сума чисел 3 і 5 більша за 7» ми визнаємо істинним, а «Сума чисел 3 і 5 менша від 7» — хибним. З точки зору логіки подібні висловлювання припускають лише два варіанти відповіді: «так» (ІСТИНА) та «ні» (ХИБНІСТЬ).

Подібне використовується у програмуванні: якщо результатом обчислення виразу може бути лише значення **істина** або **хибність**, то такий вираз називають *логічним*.

Логічні оператори використовують в усіх мовах програмування. Не є виключенням і Scratch. Для створення логічних виразів в Scratch використовують оператори порівняння:



Рис. 3.76

-  — лівий параметр більший за правий;
-  — лівий параметр менший від правого;
-  — параметри є рівними.

Логічні оператори у Scratch зазвичай використовують всередині С-блоків при створенні розгалужень і циклів — конструкцій алгоритмів, що керують порядком виконання програм і здатні формувати ілюзію розумної поведінки спрайтів. На рисунку 3.77 наведено приклади логічних блоків.



Рис. 3.77. Приклади логічних блоків

26.2.

ПОНЯТТЯ УМОВНОГО ОПЕРАТОРА

Ви вже знаєте, що оператор розгалуження застосовують у випадках, коли у програмі потрібно організувати вибір, тобто зробити так, аби певний набір команд у ній виконувався лише після виконання деякої умови. Через це такі оператори називають *умовними*.

Отже, **умовний оператор**, або як його ще називають *оператор розгалуження*, — це оператор, у якому здійснюється вибір однієї з двох дій в залежності від виконання або невиконання умови.

Слід зазначити, що оператори розгалуження використовуються майже в усіх мовах програмування. У більшості з них умовний оператор розпочинається ключовим словом **If** (якщо) і може бути поданий у двох формах — повній (умовний оператор з двома гілками) та скороченій (умовний оператор з однією гілкою).

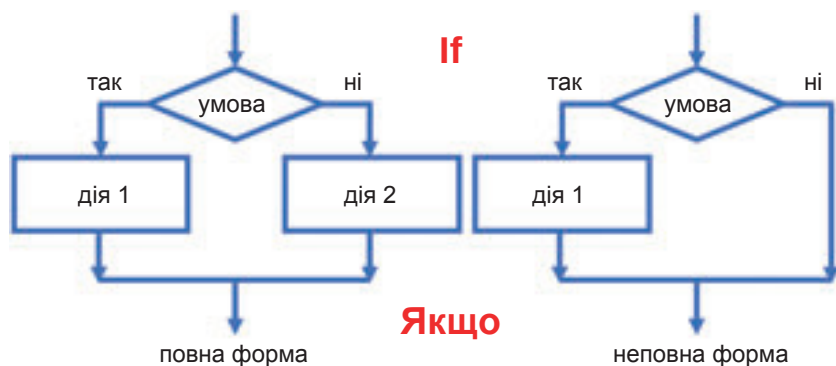


Рис. 3.78. Види операторів розгалуження

У мові програмування Scratch теж є такі оператори.



26.3.

НЕПОВНА ФОРМА РОЗГАЛУЖЕННЯ

На рис. 3.79 зображено блок-схему умовного оператора з однією гілкою та відповідний оператор мови Scratch. Під час виконання цього оператора перевіряється деякий логічний вираз (умова) і, коли він набуває значення **істина**, виконується команда (чи команди) всередині С-блока; інакше виконуються команди, що йдуть одразу за ним.

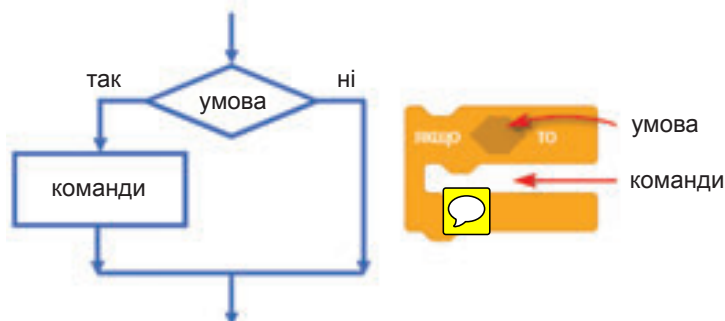


Рис. 3.79. Неповна форма розгалуження

ВПРАВА 13



Завдання. Виконавши це завдання, ви створите проєкт, у якому поведінкою об'єктів керуватиме блок умовного оператора з однією гілкою.

У цьому проєкті дівчинка після запитання про відвідування кінотеатру та отримання відповіді «так» на радіощах має підстрибнути та вигукнути «Ура!», при одержанні інших відповідей вона ніяких дій не виконує.

Розглянемо блок-схему алгоритму, відповідно до якого будуть відбуватися події в проєкті.

1. Спочатку ми одержимо відповідь на запитання дівчини «Ми йдемо сьогодні до театру?».
2. Якщо відповідь буде «так», то дівчинка підстрибне та вигукне «Ура!», при цьому зміниться її образ, далі здійсниться

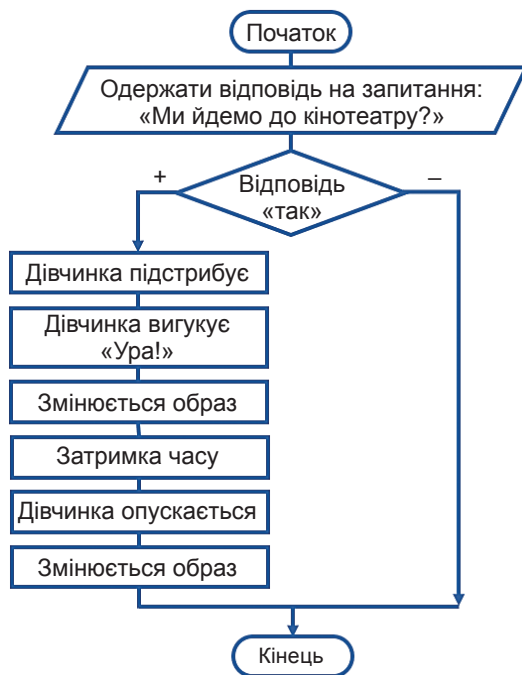


Рис. 3.80. Блок-схема проєкту



РОЗДІЛ 3. Алгоритми та програми

її затримка у повітрі (якщо цього не зробити, то рух вгору, а потім униз не буде відтворено). Після затримки часу дівчинка опуститься на землю і знову її образ зміниться на початковий.

3. Якщо ж відповідь буде якоюсь іншою, жодна подія в алгоритмі не буде виконуватися.

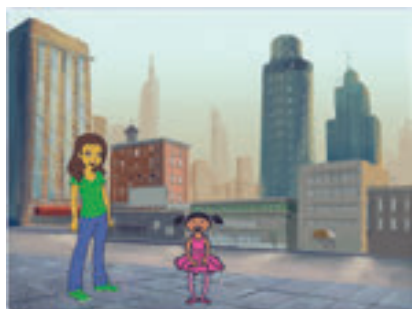


Рис. 3.81

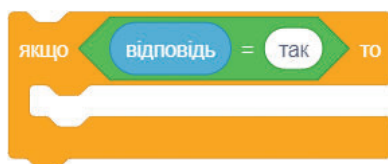


Рис. 3.82

Переходимо до реалізації самого проекту.

1. Розташуйте на сцені спрайт із зображенням дівчинки та її мами. Додайте фон сцени.
2. Залиште для спрайта дівчинки лише перші її два образи.
3. Створіть скрипт для спрайта дівчинки відповідно до блок-схеми.
 - а) Додайте команду **запитати ... і чекати**. Введіть до неї запитання «Ми йдемо до кінотеатру?».
 - б) Додайте команду **якщо...то....** Як умову сформулюйте блок **відповідь = так** (рис. 3.82).
 - в) Всередині С-блоку додайте команди, що будуть виконуватися, якщо вказана умова буде істинною:

- для підстрибування дівчинки застосуйте команду **змінити у на 100** (рис. 3.83);
- додайте команду **говорити «Ура!»** (рис. 3.84);
- нижче додайте команду **наступний образ** (рис. 3.85);



Рис. 3.83



Рис. 3.84



Рис. 3.85

- для затримки образу дівчинки додайте команду **чекати 1 секунд** (рис. 3.86);
- наступною додайте команду **змінити у на -100** (в результаті чого дівчинка має опуститися на землю) (рис. 3.87).
- останньою додайте команду **наступний образ** (рис. 3.88).



Рис. 3.86



Рис. 3.87



Рис. 3.88



4. Запустіть скрипт на виконання та введіть відповідь **так**. В результаті дівчинка має підстрибнути, змінюючи свій зовнішній вигляд.
5. Ще раз запустіть проєкт і введіть відповідь **авжеж**. Цього разу нічого не повинно відбуватися, адже відповідь має бути саме такою, як указано в команді програми.
6. Ще раз запустіть проєкт і введіть відповідь **Так** (з великої літери). І знову дівчинка не повинна виконувати ніяких дій.
7. Аби програма не реагувала на регістр уведеної відповіді, змініть умову програми наступним чином (рис. 3.89):



Рис. 3.89

8. Ще раз запустіть проєкт і введіть **Так** (з великої літери), на цей раз дівчинка має відреагувати на введену відповідь.
9. Збережіть проєкт з ім'ям **Вправа_13**.

26.4.

ПОВНА ФОРМА РОЗГАЛУЖЕННЯ

Функцію умовного оператора з двома гілками у Скретчі виконує команда **якщо...то...інакше...**. На рис. 3.90 ви бачите блок-схему такого оператора та відповідний Е-блок, що використовується в середовищі Скретч.

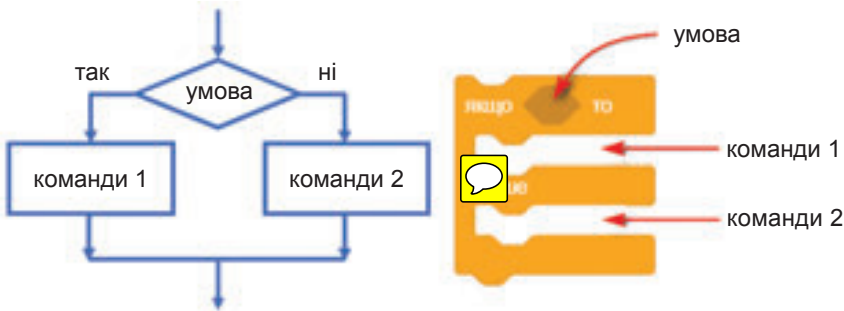


Рис. 3.90. Повна форма розгалуження

Під час виконання цього оператора перевіряється задана умова і, коли вона стає істинною, виконується блок **команди 1**, який розташований після гілки **то**. У протилежному випадку виконується блок **команди 2** після гілки **інакше**.

Далі розглянемо створення проєкту, в якому поведінкою об'єкта керує блок умовного оператора з двома гілками.



ВПРАВА 14



Рис. 3.91

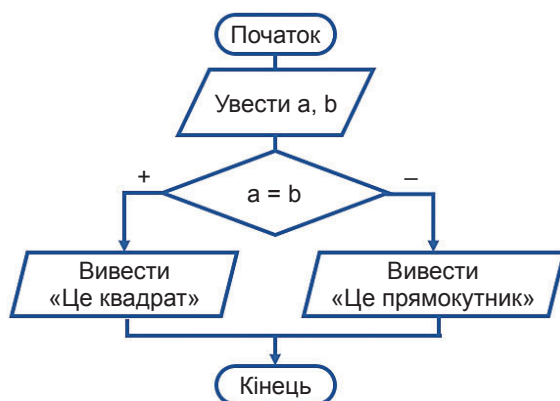


Рис. 3.92

1. У вікні нового проєкту створіть дві змінні **a** і **b**.
2. За допомогою команд **запитати ... і чекати** та **надати змінній ... значення відповідь** створіть скрипт присвоєння змінним **a** і **b** числових значень (рис. 3.93).
3. Додайте блок **якщо...то...інакше...**.
4. Уведіть умову **a=b** (рис. 3.94).



Рис. 3.93

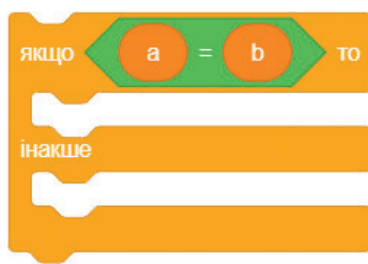


Рис. 3.94

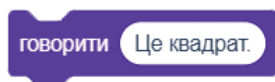


Рис. 3.95

5. Після команди **то** введіть дію, яка має виконатися у разі виконання умови, а саме, **говорити Це квадрат** (рис. 3.95).

Завдання. У програмі необхідно ввести два числа, що визначають довжини прилеглих сторін прямокутника. Якщо числа є рівними, то програма має повідомити, що ця фігура є квадрат, у протилежному випадку має вивести повідомлення «Це прямокутник».

Реалізацію проєкту в середовищі Скретч виконайте за наведеною нижче блок-схемою.

З блок-схеми видно, що спочатку слід ввести два числа, значення яких буде присвоєно змінним **a** і **b**. Потім необхідно перевірити, чи дорівнюють значення цих змінних. Якщо це так, то вивести повідомлення «**Це квадрат**», в протилежному випадку вивести «**Це прямокутник**».



6. Після гілки **інакше** додайте команду **говорити Це прямокутник** (рис. 3.96).

говорити Це прямокутник.

7. Додайте команду **коли прапорець натиснуто**.

Рис. 3.96

8. Запустіть проєкт на виконання.

9. Спробуйте ввести спочатку два однакових числа. В результаті кіт має повідомити, що це буде квадрат.

10. Ще раз запустіть проєкт і введіть два різних числа, тепер повідомлення буде «Це прямокутник».

11. Збережіть файл з ім'ям **Вправа_14**.

ВПРАВА 15



Завдання. Створити проєкт, у якому кіт дізнаватиметься в дітей про їхню вагу, порівнюватиме ці значення та повідомлятиме, хто з дітей важчий (рис. 3.97).

Для створення даного проєкту скористаємося алгоритмом знаходження більшого числа з-поміж двох заданих. На рис. 3.98 представлено блок-схему такого алгоритму.



Рис. 3.97

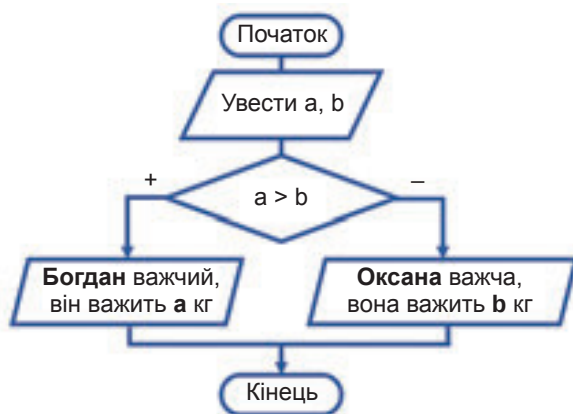


Рис. 3.98

Спочатку буде введено два числа: вагу дівчинки та вагу хлопчика, далі ці значення порівнюються у блоці умови, і в залежності від істинності умови буде виведено повідомлення з указуванням імені важчої дитини та її ваги. Переходимо до створення проєкту.

1. До проєкту додайте спрайти хлопчика і дівчинки з бібліотеки програми та розташуйте їх біля спрайта кота.
2. Після цього створіть змінні **a** і **b**.
3. Складіть скрипт для спрайта кота.

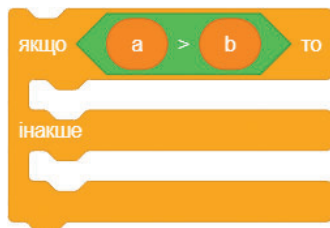


Рис. 3.99

- а) Як і в попередньому проєкті, введення значень змінних **a** і **b** виконайте з допомогою команд **запитати ... і чекати** та **надати змінній значення відповідь**.
- б) Далі додайте блок **якщо...то...інакше...**.
- в) Уведіть умову **a>b** (рис. 3.99).
- г) Після команди **то** введіть дію, яка має виконатися у разі істинної умови, а саме, **говорити Богдан важчий, він важить a кг** (рис. 3.100).

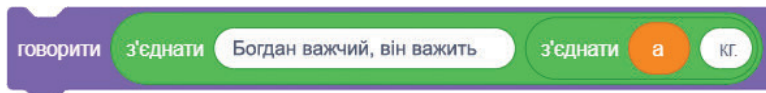


Рис. 3.100.

- г) Після гілки **інакше** додайте команду **говорити Оксанка важча, вона важить b кг** (рис. 3.101).

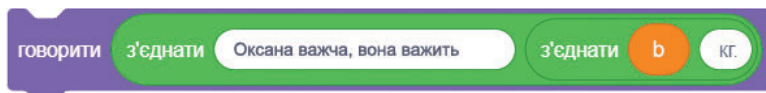


Рис. 3.101

- д) Додайте команду **коли прапорець натиснуто**.
4. Запустіть проєкт на виконання.
5. Уведіть спочатку перше число, яке буде більше ніж друге, наприклад, 45 і 40. В результаті кіт має повідомити: «Богдан важчий, він важить 45 кг».
6. Ще раз запустіть проєкт і введіть числа 35 і 40. Кіт має вивести повідомлення «Оксанка важча, вона важить 40 кг».
7. Самостійно додайте до цього скрипта команди, що змусять кота під час введення даних повертатися у бік того персонажа проєкту, до якого він звертається.
8. Збережіть файл з ім'ям **Вправа_15**.



Виконати вправи 16 та 17 ви зможете в електронному додатку.



Висновки

Вираз, результатом обчислення якого може бути лише значення *істина* або *хибність*, називають **логічним**.

Умовний оператор, або *оператор розгалуження*, — це оператор, у якому здійснюється вибір однієї з двох дій в залежності від виконання або невиконання умови.

Неповна форма умовного оператора в Scratch реалізована командою **якщо...то...**.

Функцію умовного оператора з двома гілками у Scratch виконує команда **якщо...то...інакше...**.



Контрольні запитання та завдання

1. Що таке логічний вираз? Наведіть приклади логічних виразів.
2. Що таке оператор розгалуження?
3. Які є форми оператора розгалуження?
4. Наведіть приклад алгоритму зі скороченою формою розгалуження.
5. Наведіть приклад алгоритму з повною формою розгалуження.
6. Які команди розгалуження використовують у середовищі Scratch?

§27.

Вкладені розгалуження

27.1.

ФОРМУВАННЯ ВКЛАДЕНИХ РОЗГАЛУЖЕНЬ

Іноді при розв'язуванні задач необхідно розглянути більше ніж два варіанти вибору — в цих випадках можна застосувати вкладення блоків розгалуження. Саме *вкладені розгалуження* ми і розглянемо в цьому уроці.

Якщо в точці розгалуження програми має бути більше ніж два варіанти вибору, додаткові умовні оператори можна вставити всередину секцій **якщо** або **інакше**.

Такі розгалуження у програмуванні використовують доволі часто, а додаткові оператори називають **вкладеними умовними операторами**. Скретч не обмежує кількість вкладених умовних операторів, тож ви можете реалізувати тут будь-які складні розгалуження.

Розглянемо проєкти з використанням вкладених розгалужень.

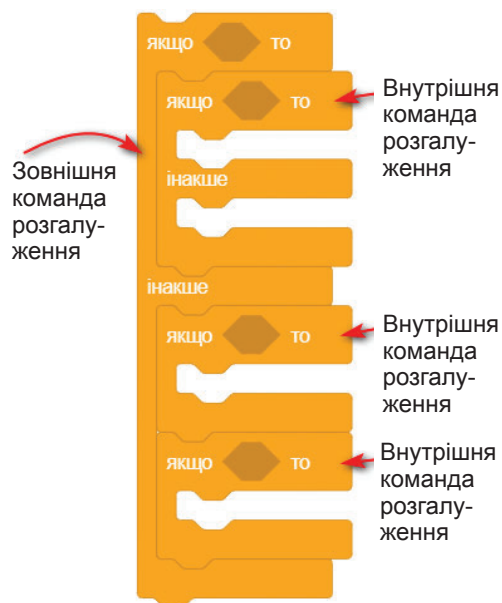


Рис. 3.102



ВПРАВА 18



Завдання. Створити проєкт, який за значенням температури виводив би назву стану, в якому перебуває вода за цієї температури: рідкий, твердий або газоподібний, тобто вода, лід або пара (рис. 3.103).

На рис. 3.104 представлено блок-схему розв'язання задачі.



Рис. 3.103

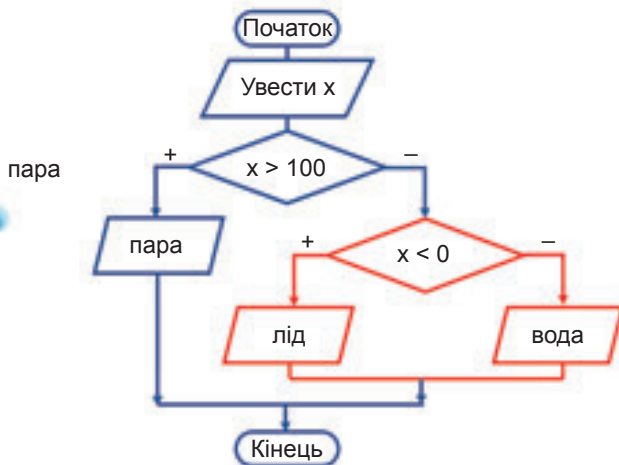


Рис. 3.104

1. Спочатку слід увести температуру води та надати її значення змінній x .
2. Далі перевірити умову $x > 100$.
3. Якщо умова виконується, то слід вивести повідомлення «пара».
4. У протилежному випадку розглянути умову $x < 0$.
5. Якщо ця умова виконується, то вивести «лід» і в протилежному випадку — «вода».

Зверніть увагу, що в даному алгоритмі на гілці **інакше** було застосовано вкладену вказівку розгалуження.

На блок-схемі її позначено червоним кольором.

Створіть проєкт за поданим алгоритмом.

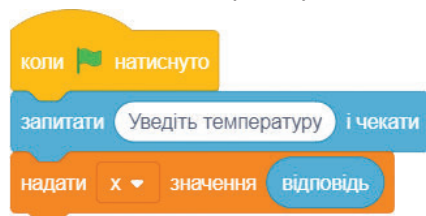


Рис. 3.105

1. Створіть змінну x для введення температури води.
2. Для введення значення цієї змінної скористайтеся командами **запитати Уведіть температуру і чекати** і **надати x значення відповідь** (рис. 3.105).
3. Для перевірки умови додайте команду **якщо... то...інакше...**.

4. Сформууйте умову $x > 100$.



5. Після гілки **то** додайте команду **говорити пара**.
6. Після гілки **інакше** додайте ще одну команду **якщо... то...інакше...**. Тут як умову введіть $x < 0$.
7. Якщо умова виконується, додайте команду **говорити лід**, інакше **говорити вода** (рис. 3.106).
8. Збережіть проєкт з ім'ям **Вправа_18**.

Якщо для розв'язання задачі необхідно врахувати декілька умов одночасно або враховуватись має одна із зазначених умов, використовують оператори логічних операцій **і** та **або**.

Так, розглянемо проєкт, в якому необхідно використати ці оператори.

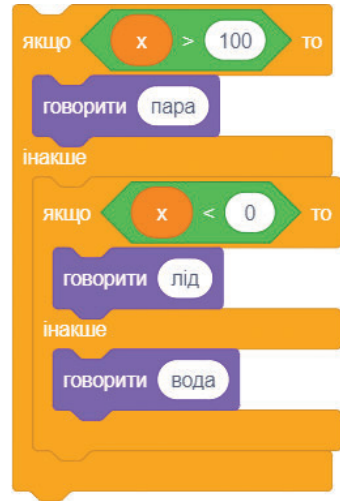


Рис. 3.106

ВПРАВА 19



Завдання. Створити проєкт, у якому за даними трьома сторонами трикутника необхідно визначити, яким він буде: рівностороннім, рівнобедреним чи різностороннім (рис. 3.107).

При створенні проєкту будемо керуватися блок-схемою, представленою на рис. 3.108.

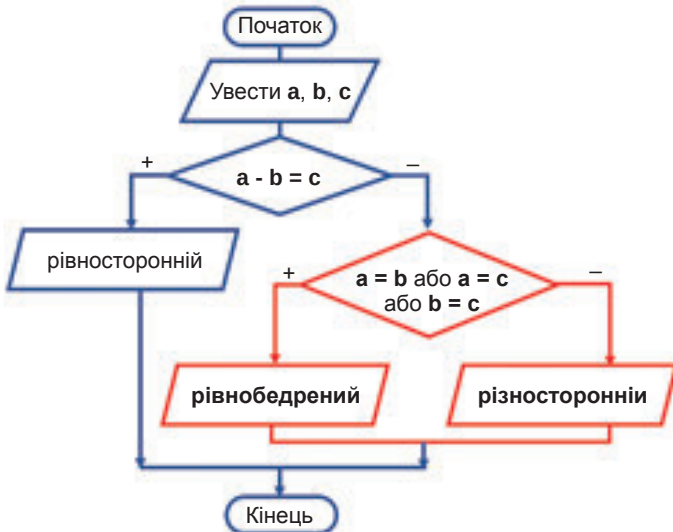
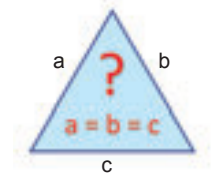
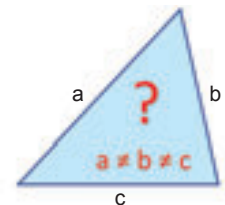


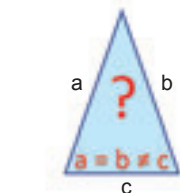
Рис. 3.108



рівносторонній



різносторонній



рівнобедрений

1. Спочатку введемо значення змінних **a**, **b** і **c**, що відповідають значенням довжин сторін трикутника.

Рис. 3.107



РОЗДІЛ 3. Алгоритми та програми

2. Наступний крок алгоритму — це перевірка рівності сторін між собою ($a=b=c$).
3. Якщо ця умова виконується, то буде виведено повідомлення, що трикутник є **рівностороннім**.
4. У протилежному випадку розглянемо умову рівності між собою лише двох сторін (це буде умова $a=b$ або $a=c$, або $b=c$).
5. Якщо принаймні одна з рівностей виконується, то буде виведено повідомлення про те, що трикутник є **рівнобедреним**, у протилежному випадку буде виведено повідомлення, що трикутник є **різностороннім**.

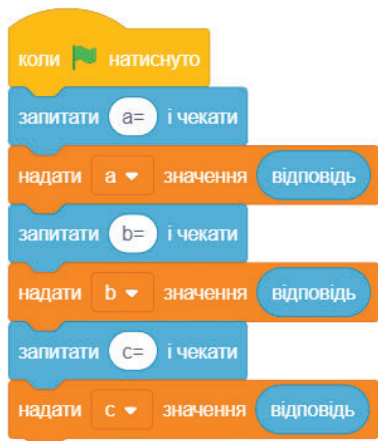


Рис. 3.109

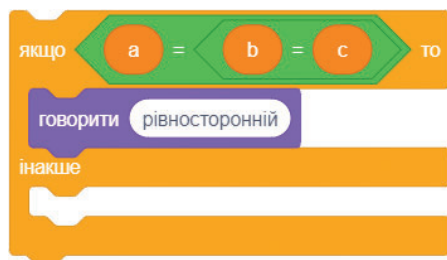


Рис. 3.110

Виведемо це повідомлення після гілки **то**, додавши команду **говорити рівнобедрений**.

9. До останньої гілки **інакше** додайте команду **говорити різносторонній**.

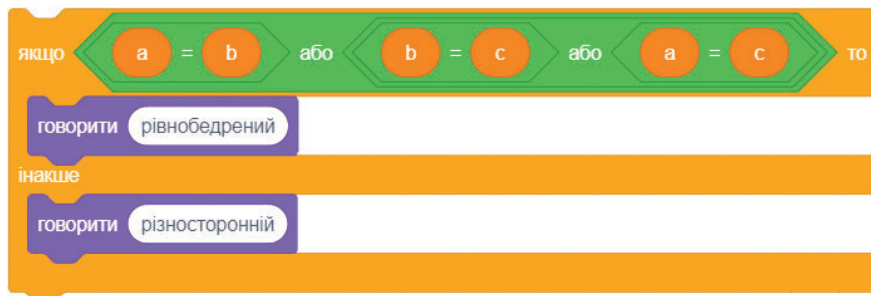


Рис. 3.111

Створіть проєкт, що відповідає розглянутій блок-схемі.

1. Створіть змінні **a**, **b** і **c**, що відповідають сторонам трикутника.
2. Додайте команди для введення значення цих змінних за допомогою блоків **запитати ... і чекати** та **надати змінній ... значення відповідь** (рис. 3.109).
3. Оскільки в алгоритмі представлено дві гілки розгалуження, додайте блок команди **якщо...то...інакше...**.
4. Сформуйте у ньому умову $a=b=c$.
5. У разі виконання відразу обох виразів порівняння буде виведено повідомлення, що трикутник є **рівностороннім**. Для виведення цього повідомлення після гілки **то** додайте команду **говорити рівносторонній**.
6. До гілки **інакше** знову додайте команду **якщо...то...інакше**, оскільки відповідно до блок-схеми ми маємо справу із вкладеним розгалуженням (рис. 3.110).
7. У цьому блоці сформуйте умову $a=b$ або $b=c$, або $a=c$.
8. В даному випадку у разі виконання принаймні одного з виразів порівняння буде виведено повідомлення, що трикутник є **рівнобедреним**.



10. Перевірте роботу програми. Запустіть проєкт. Уведіть числа 1, 1, 1. Ви маєте одержати повідомлення **рівносторонній**. При введенні 2, 2, 3 — **рівнобедрений**. Для значень 3, 4, 5 — **різносторонній**.
- 11*. Самостійно додайте умову: якщо сума двох сторін трикутника менша від третьої, то вивести повідомлення «такий трикутник не існує».
12. Збережіть файл з ім'ям **Вправа_19**.

27.2.

ЛОГІЧНІ ОПЕРАЦІЇ

У попередньому проєкті ми розглянули репортери, що дають змогу здійснювати логічні операції над діями порівняння. У Scratch усього їх три:

репортер **і** — умова є істинною, якщо обидві умови, об'єднані репортером, є істинними, цю операцію ще називають *логічне множення*;

репортер **або** — умова є істинною, якщо принаймні одна з умов, об'єднана репортером, є істинною, цю операцію називають *логічне додавання*;

репортер **не** — умова є істинною, якщо умова, розташована всередині цього репортера, є хибною, цю операцію називають *логічне заперечення*.

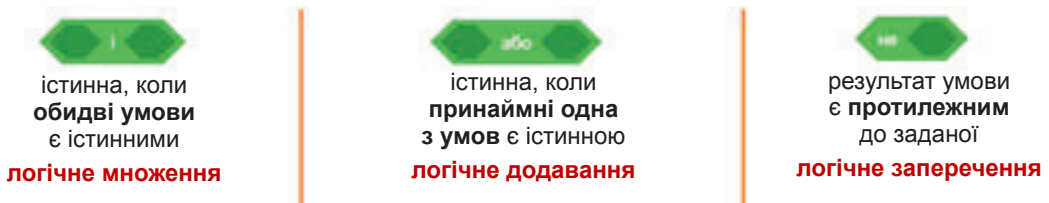


Рис. 3.112

Згадані щойно логічні операції **і**, **або**, **не** найзручніше описувати з допомогою *таблиць істинності*. У них наведено всі можливі значення істинності результатів обчислень складних висловлювань за різних значень простих вихідних висловлювань. Прості висловлювання тут позначено буквами **A** і **B**. Значення 0 свідчить про хибність висловлювання, 1 — про його істинність.

Логічне множення			Логічне додавання			Логічне заперечення	
A	B	A і B	A	B	A або B	A	не A
0	0	0	0	0	0	0	1
0	1	0	0	1	1	1	0
1	0	0	1	0	1		
1	1	1	1	1	1		

Рис. 3.113



27.3.

ВИКОРИСТАННЯ ЛОГІЧНИХ ОПЕРАЦІЙ

Логічні операції, як і арифметичні, мають свій пріоритет при виконанні, найвищий пріоритет має репортер **не**, після нього виконується репортер **і**, останнім репортер **або**, але, звичайно, важливість має і рівень вкладення одного репортеру до іншого.

Розглянемо приклади використання логічних операцій та визначимо їхню істинність.

Приклад 1.

Рис. 3.114

Цей вираз складається з двох простих умов $3 > 2$ та $2 < 1$, з'єднаних операцією **і**.

$3 > 2$ є істинною умовою;

$2 < 1$ є хибною умовою.

Відповідно до таблиці істинності весь вираз, об'єднаний сполучником «**і**», є **хибним**.

A	B	A і B
0	0	0
0	1	0
1	0	0
1	1	1



Рис. 3.115

Приклад 2.

Рис. 3.116

Цей вираз дуже схожий на попередній, але в ньому умови об'єднані оператором **або**.

З попереднього виразу ми вже визначили, що перша умова є істинною, а друга хибною.

Звертаємося до таблиці істинності операції **або**. Відповідно до неї весь вираз буде **істинним**.

A	B	A або B
0	0	0
0	1	1
1	0	1
1	1	1

Рис. 3.117





Приклад 3.



Рис. 3.118

Умова, що стоїть після частки **не** є істинною, отже, відповідно до таблиці істинності початковий вираз є хибним.



Рис. 3.119

Приклад 4.



Рис. 3.120

Для визначення істинності виразів, що мають декілька логічних операторів, слід починати із внутрішніх операцій. В даному прикладі це операція **і**.

Вона містить дві умови $4 > 3$ та $3 + 4 = 7$. Обидві ці умови є **істинними**. Відповідно до таблиць істинності і весь вираз, об'єднаний сполучником «і», є **істинним**.

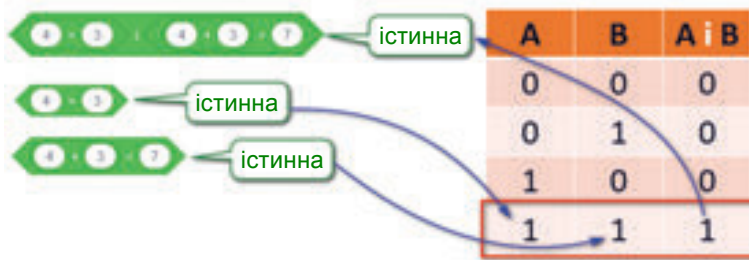


Рис. 3.121

Наступним є вираз, який з'єднує сполучником **або** розглянутий тільки що нами вираз та умову $2 * 3 = 5$.

Умова $2 * 3 = 5$ є хибною, а наступний вираз, ми вже визначили, є **істинним**.

Отже, відповідно до таблиці істинності для сполучника **або** і весь вираз буде **істинним**.

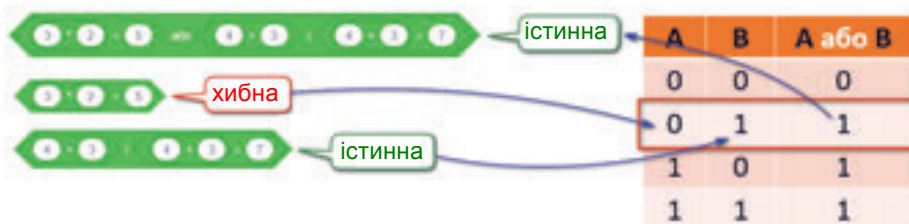


Рис. 3.122



І наостанок, оператор **не**, розташований зовні усього виразу, перетворює попередній вираз на **хибне** значення.



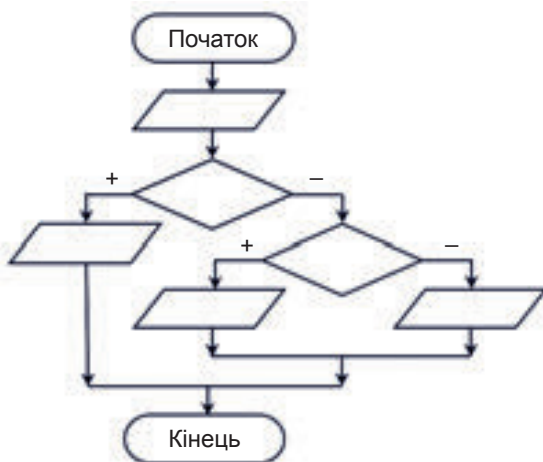
Рис. 3.123

ВПРАВА 20



Завдання. Довершити блок-схему та створити проєкт розв'язання задачі у середовищі Scratch.

Задача. Відомо вік людини — N років. Необхідно визначити її приналежність до однієї з таких вікових груп: дитина ($0 \leq N < 17$), робітник ($17 \leq N < 60$), пенсіонер ($N \geq 60$).



Виконати вправу

21 ви зможете в електронному

додатку.



Рис. 3.124



Висновки

Якщо в точці розгалуження програми має бути більше ніж два варіанти вибору, використовують вкладені умовні оператори.

Scratch не обмежує кількість вкладених умовних операторів.

Складні умови складаються з простих, що з'єднані логічними операціями **і**, **або**, **не**.

Використовуючи логічний оператор **і**, ми отримаємо істинне значення, якщо обидві умови, об'єднані ним, будуть істинними.

Операція **або** є істинною, коли принаймні одна з умов є істинною.

Оператор **не** робить істинною хибну умову і навпаки.



Контрольні запитання та завдання

1. У яких випадках використовують вкладені умовні оператори?
2. Скільки може бути вкладених операторів у Scratch?
3. Які оператори порівняння використовують у Scratch? Опишіть їхні призначення.

§28.

Практична робота №6. «Складання та виконання алгоритмів з розгалуженням»

Ми з вами вже познайомилися з операторами, що дозволяють використовувати різні форми вказівки розгалуження та конструювати умови різної складності.

Виконуючи завдання практичної роботи, ви створите проект з використанням різних форм розгалужень.

Завдання 1. Створити проект, у якому за заданим користувачем номером місяця дівчинка має повідомити відповідну пору року (зима, весна, літо чи осінь), після чого вона має змінити свій образ (одяг), а також відповідно до пори року має змінитися фон сцени.

Розглянемо блок-схему реалізації проекту.



Рис. 3.125

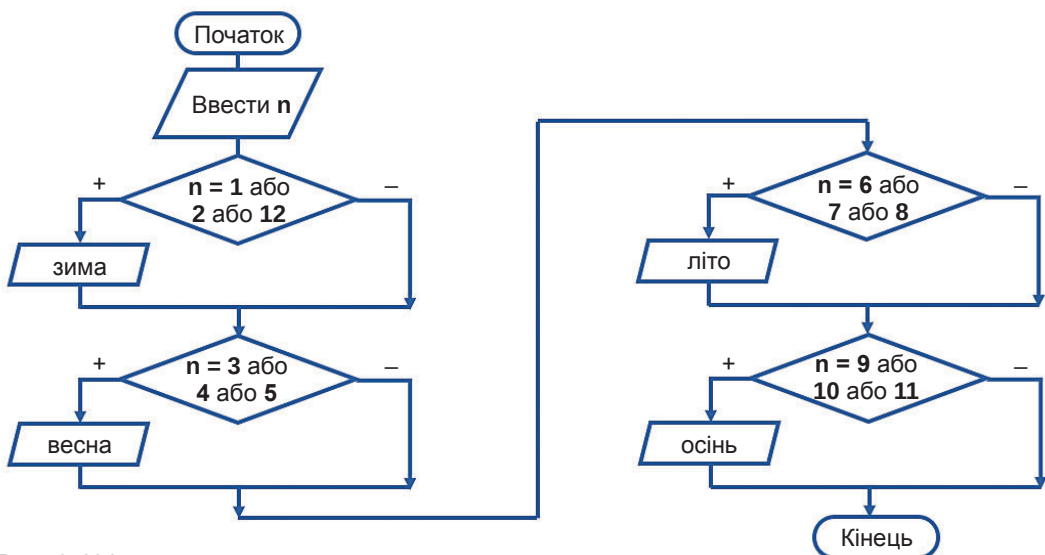


Рис. 3.126



1. Спочатку необхідно ввести значення змінної n , що відповідає номеру місяця.
2. Якщо значення $n=1$ або $n=2$, або $n=12$, буде виведено «зима».
3. Якщо $n=3$ або $n=4$, або $n=5$ — «весна».
4. Якщо $n=6$, $n=7$ або $n=8$ — «літо».
5. Якщо $n=9$, $n=10$ або $n=11$ — «осінь».

Створіть проєкт у середовищі Scratch.

1. До нового проєкту завантажте чотири фони для сцени з різними порами року, спрайт дівчинки та три її образи (файли проєкту розташовано в електронному додатку).
2. Змініть назви фонів та образів відповідно до пір року.
3. Створіть змінну n , якій будуть присвоюватися значення номерів місяців.
4. Створіть скрипт для спрайта дівчинки.

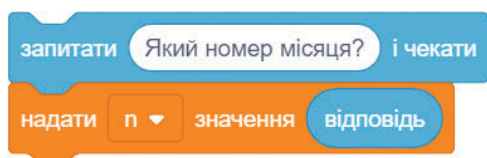


Рис. 3.127

- а) Додайте команди введення змінної n (рис. 3.127).
- б) Додайте команду **якщо...то...**, що відповідає неповній формі розгалуження.
- в) Сформууйте умову $n=1$ або $n=2$, або $n=12$.
- г) Під умовою додайте команду виведення повідомлення «зима».

- г) Наступною додайте команду зміни образу на **Зима**. З допомогою цієї команди дівчинка змінить свій одяг на зимовий.
 - д) Додайте команду **оповістити зима**. Ця команда звернеться до сцени, у якій буде виконано дію зміни фону (рис. 3.128).
5. Створіть скрипт для сцени, що дозволить змінювати її фон відповідно до пори року. Скрипт почне діяти, коли для спрайта дівчинки спрацює команда **оповістити зима** (рис. 3.129).



Рис. 3.128

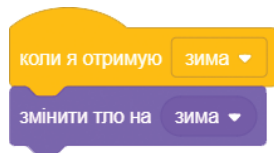


Рис. 3.129

6. Запустіть створений проєкт і введіть, наприклад, число **5**. В результаті не повинно нічого відбутися, оскільки для цього числа ви не вказували ніяких умов.



7. Ще раз запустіть проєкт і введіть число 2. Дівчинка має вивести повідомлення **Зима**. Також мають змінитися сцена і одяг дівчинки відповідно до зимової пори року.
8. Самостійно додайте команди для врахування інших номерів місяців. Для цього виділіть спрайт дівчинки. Скопіюйте утворену гілку **якщо** необхідну кількість разів та змініть указану в ній умову і напси в командах.
9. Також скопіюйте необхідну кількість разів уже створений скрипт для сцени, змінивши в ньому команди відповідно до пір року.
10. Додайте до скрипта, складеного для спрайта дівчинки, блок **коли натиснуто зелений прапорець** і запустіть проєкт на виконання. Перевірте працездатність проєкту для місяців різних пір року.
11. Збережіть файл з ім'ям **ПР6_1**.

Завдання 2. Створити проєкт представлення результатів змагань зі стрибків.

Відбудеться змагання зі стрибків у довжину трьох персонажів: слона, собаки та kota. Суддя слідкує за змаганням і має повідомляти кращий результат та того, хто цей результат показав.

По суті алгоритм реалізації проєкту зводиться до знаходження **більшого значення з трьох чисел**. Блок-схема цього алгоритму представлена на рис. 3.131.

1. Спочатку програма має одержати три числа і присвоїти їхні значення трьом змінним **a**, **b**, **c**. В нашому проєкті це будуть випадкові числа.
2. Далі в алгоритмі порівнюються значення перших двох змінних, більше значення присвоюється змінній **max**.
3. Потім здійснюється порівняння третьої змінної і значення, що одержала змінна **max**. Якщо значення третьої змінної більше — змінна **max** змінюється на значення змінної **c**, в протилежному випадку значення змінної **max** не змінюється.

Переходимо до створення проєкту. Розглянемо його сценарій. Після натискання прапорця усі спортсмени мають стриб-



Рис. 3.130

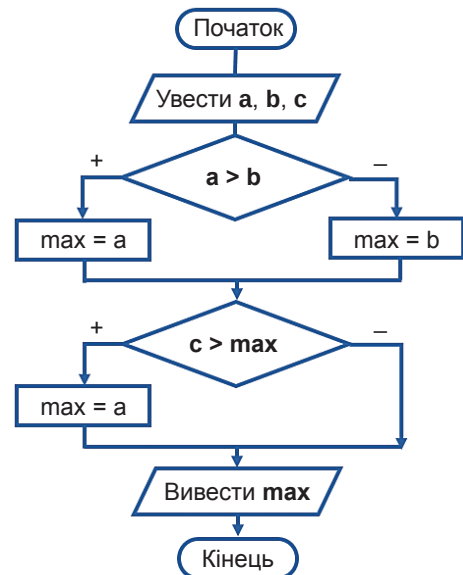


Рис. 3.131



нути на певну відстань — це буде випадкове число від 1 до 300. Значення цих відстаней буде зафіксовано змінними **a**, **b** та **c**. Після цього суддя порівняє одержані значення, відповідно до алгоритму, розглянутого в блок-схемі, та виведе повідомлення про кращого спортсмена і його результат.

1. Додайте до проєкту спрайти слона, собаки та судді.
2. Створіть змінні **a**, **b**, **c**, які дорівнюватимуть довжині стрибка слона, собаки та kota відповідно, а також змінну **max**, яка буде одержувати значення більшого з цих чисел.
3. Додайте змінну **персонаж**, вона буде набувати назви спортсмена, що має кращий результат.
4. Виділіть спрайт слона та утворіть для нього скрипт (зверніть увагу, що цей скрипт буде ідентичним для усіх спортсменів).
 - а) Першою додайте команду **коли прапорець натиснуто**. Після натискання прапорця усі спортсмени мають стрибнути.
 - б) Аби спортсмени кожного разу стрибали з одного й того ж місця додайте команду **задати x-180** (це приблизно лівий край сцени з урахуванням розміру самого спрайта).
 - в) Відстань, на яку має стрибнути спрайт, визначте з допомогою команди **надати a значення випадкове від 1 до 300**.
 - г) Аби стрибок відбувся на це значення, додайте команду **змінити x на a**.
 - г') Для подальшого опрацювання результатів стрибка передайте виконання наступного коду судді з допомогою команди **оповістити суддя** (рис. 3.132).

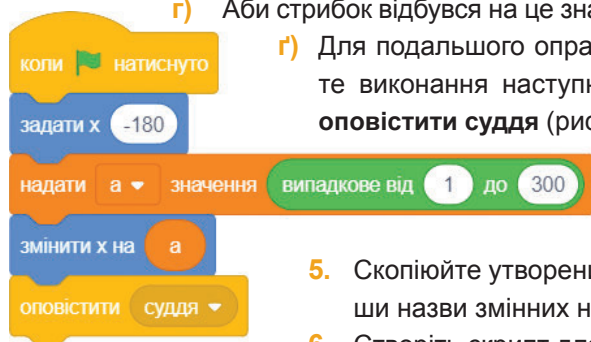


Рис. 3.132

5. Скопіюйте утворений скрипт на інших спортсменів, змінивши назви змінних на відповідні.
6. Створіть скрипт для спрайта судді.
 - а) Для початку опрацювання даних після стрибка спортсменів додайте команду **коли я отримую суддя**.
 - б) Далі, відповідно до блок-схеми алгоритму, додайте гілку з двома діями: команду **якщо...то...інакше...**.
 - в) Сформууйте в ній умову: **a>b**.
 - г) Після гілки **то** додайте команди **надати max значення a** та **надати персонаж значення слон**.
 - г') Гілку **інакше** заповніть самостійно, керуючись рисунком 3.133.
 - д) Нижче додайте команду **якщо...то...**, що відповідає неповній формі розгалуження.
 - е) Сформууйте у ній умову **c>max** (рис. 3.134).
 - є) Дії в цій гілці створіть самостійно, керуючись блок-схемою.

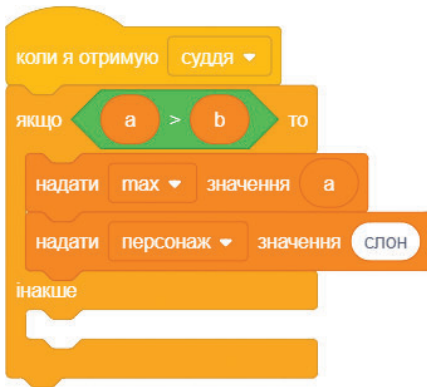


Рис. 3.133



Рис. 3.134

7. Додайте команду виведення повідомлення судді про результат найкращого спортсмена (рис. 3.135).

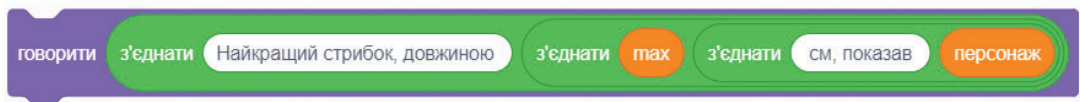


Рис. 3.135

8. Запустіть проект на виконання декілька разів та переконайтеся, що кожного разу ви будете отримувати різні результати. За значеннями, що висвічуються на моніторах, ви зможете відслідкувати правильність виконання проекту.
9. Збережіть файл з ім'ям **ПР6_2**.



Виконати завдання 3
ви зможете в електронному додатку.



§29.

Поняття циклу. Цикл з лічильником

29.1.

ПОНЯТТЯ ЦИКЛУ

Однією з найцінніших властивостей комп'ютера вважають його здатність багато разів невтомно виконувати одні й ті самі дії. Адже навіть дуже прості обчислення можуть завести людину в глухий кут, якщо деякі операції потрібно повторювати принаймні тисячі разів. У цьому параграфі ми розглянемо, як зробити багаторазові обчислення з використанням середовища Scratch.

З повторюваними обчисленнями програмісти стикаються повсякчас. Ви вже знаєте, що в програмуванні багаторазове повторення певної по-



слідовності називають **циклом**. Послідовність команд, які повторюються під час виконання оператора циклу, називають **тілом циклу**.

29.2.

БЕЗУМОВНИЙ ЦИКЛ

Найчастіше при створенні програм в Scratch використовують **безумовні цикли**, вихід з яких не передбачено логікою програми. Такі цикли ще називають **нескінченними**.

У Scratch, на відміну від інших мов програмування, для створення таких циклів є спеціальний оператор. На рис. 3.136 наведено блок-схему оператора безумовного циклу та відповідний Scratch-блок.



Рис. 3.136. Блок-схема (а) та відповідний Scratch-блок (б) безумовного циклу

Тіло такого циклу виконується доти, доки скрипт (частиною якого є цей цикл) активний. Зупинити виконання нескінченного циклу у Scratch можна з допомогою заглушки **зупинити цей скрипт** (що зупиняє виконання скрипта) та заглушки **зупинити все** (що зупиняє виконання всієї програми).

ВПРАВА 22



Завдання. Створити проєкт, у якому кіт, пересуваючись сценою, буде рахувати свої кроки.

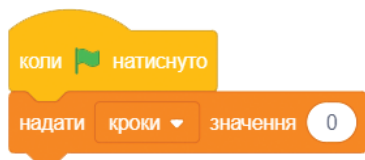


Рис. 3.137

1. Створіть змінну **кроки**.
2. Додайте команду **коли прапорець натиснуто**.
3. Додайте команду **надати кроки значення 0**. Поки кіт не зробив жодного кроку, це значення дорівнює 0 (рис. 3.137).
4. Додайте оператор безумовного циклу **завжди**.
5. Всередині цього оператора додайте команди:
 - **перемістити на 10 кроків** — для переміщення кота;



- **змінити кроки на 1** — кількість кроків збільшиться на 1. Тобто якщо спочатку це значення було 0, то при першому виконанні циклу значення 0 збільшиться на 1 і дорівнюватиме 1. На другому кроці значення 1 буде збільшено на 1 і дорівнюватиме 2 і так далі;
- **наступний образ** — для зміни вигляду кота;
- **подумати кроки** — команда виведе поточне значення кроків;
- аби кіт повертався, коли досягне межі, додайте команду **якщо на межі, відбити**;
- для затримки відображення образу додайте команду **чекати 1 секунд** (рис. 3.138).

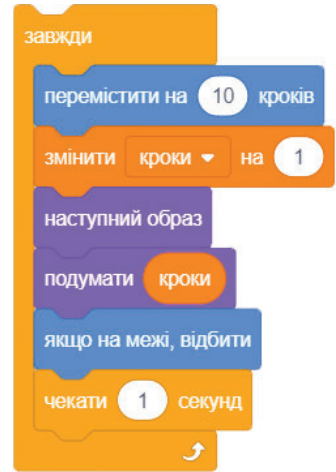


Рис. 3.138

6. Запустіть проект. Кіт має крокувати, рахуючи свої кроки.
7. Аби кіт завершив рахувати кроки через задану кількість кроків, наприклад, 7, додайте до блоку **завжди** команду **якщо...то....** Як умову додайте блок **кроки = 7**. Установіть всередині блоку команду **зупинити цей скрипт** (рис. 3.139).

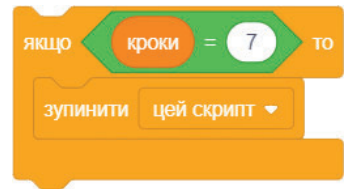


Рис. 3.139

8. Запустіть проект. На сьомому кроці кіт має завершити підрахунок.
9. Змініть умову зупинення скрипта на таку, що буде зупиняти скрипт після натиснення клавіші **пропуск**, для цього змініть умову на логічний блок **клавішу пропуск натиснуто?** (рис. 3.140).

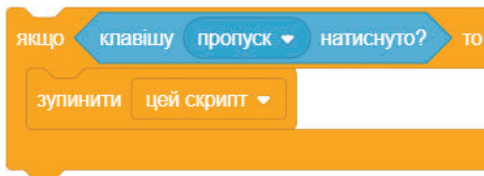


Рис. 3.140

10. Запустіть проект. Через деякий час натисніть клавішу **пропуск**. Після цього кіт має завершити підрахунок.
11. Змініть проект таким чином, аби кіт повідомляв лише парні кроки. Для цього замініть у команді **змінити кроки на значення 1** на 2 (рис. 3.141).
12. Запустіть проект та переконайтеся в правильності його виконання.
13. Збережіть файл з ім'ям **Вправа_22**.



Рис. 3.141



- 14*. Створіть копію файлу, змініть його скрипт таким чином, аби кіт повідомляв непарні кроки.
14. Збережіть файл з ім'ям **Вправа_22_1**.
- 15*. Створіть ще одну копію файлу та змініть скрипт кота таким чином, аби він рухався у зворотному порядку, рахуючи кроки від 10 до 0.
16. Збережіть файл з ім'ям **Вправа_22_2**.

29.3.

ЦИКЛ З ЛІЧИЛЬНИКОМ

Цикл з лічильником дає змогу організувати у програмі повторення певної послідовності дій задану кількість разів.

Такі цикли використовують у випадках, коли кількість виконуваних повторювань відома заздалегідь.

Scratch також має відповідний оператор, який працює за правилом «роби рівно стільки разів, скільки задано». На рис. 3.142 наведено блок-схему цього оператора. Перший її блок зображує лічильник, значення **N** (кількість повторювань) у якому щоразу після виконання тіла циклу зменшується на 1, аж доки не набуде значення **0**. Поряд представлено відповідний Scratch-оператор.

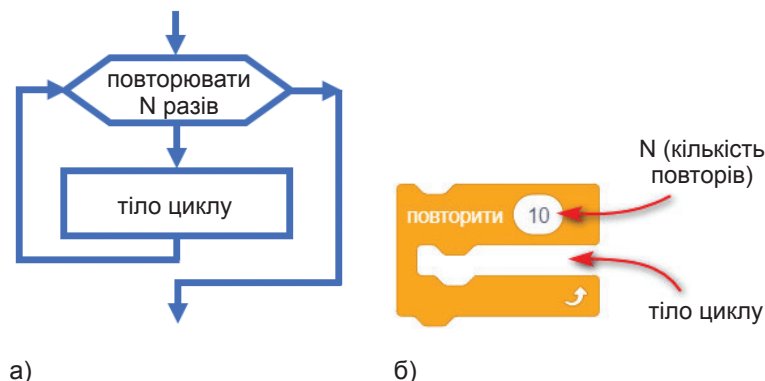


Рис. 3.142. Блок-схема (а) та відповідний Scratch-блок (б) циклу з лічильником

ВПРАВА 23



Завдання. Створити проєкт-історію про кота Скретча, в якій той збирав би камінці на березі моря та повідомляв їхню кількість. Зауважте: у перший день свого відпочинку він має додати до своєї колекції 1 камінець, у другий день — 2 камінці, у третій — 3 камінці і т. д. Допоможіть Скретчу дізнатися, скільки камінців міститиме його колекція наприкінці відпочинку. Кількість днів для відпочинку має ввести користувач (рис. 3.143).



Розглянемо блок-схему розв'язання задачі (рис. 3.144).



Рис. 3.143



Рис. 3.144

Відповідно до неї користувач спочатку має ввести значення n — кількість днів відпочинку кота.

1. Змінна i набуває значень поточного дня відпочинку, відповідно i кількість камінців, які кіт збере у цей день. Відповідно до алгоритму вона набуває початкове значення 1 . Тобто відлік починається з першого дня.
2. Змінна S — це загальна сума камінців. Звичайно, поки кіт нічого не зібрав, то її початкове значення дорівнює 0 .
3. Неважко здогадатися, що для знаходження S необхідно скористатися формулою

$$S=1+2+3+\dots+n.$$

4. Далі здійснюється перехід до виконання циклу, який буде повторюватися n разів, тобто стільки, скільки днів відпочиває кіт.
 - а) Тут до S додається значення кількості камінців на поточний день. Тобто у перший день це буде $0+1=1$ камінець.
 - б) Наступний крок $i=i+1$ збільшить значення змінної i на 1 , тобто номер дня збільшиться на 1 і стане вже дорівнювати 2 .
 - в) Далі знову здійснюється перехід до виконання тіла циклу, тобто команди $S=S+i$, і на другий день це вже буде $1+2$, тобто 3 камінці. Значення i знову збільшиться на 1 завдяки команді $i=i+1$ і стане дорівнювати 3 .
 - г) Так ці дії будуть виконуватися n разів.
5. Після виконання тіла циклу задану кількість разів буде виведено загальну кількість камінців, зібраних за n днів.



Створіть проєкт відповідно до наведеної блок-схеми.

1. Додайте до проєкту необхідну сцену, спрайт камінця та ще два його образи (аби кіт міг збирати різноманітні камінці). (Зображення камінців розташовано в електронному додатку.)
2. Створіть змінні i , n та S .
3. Створіть скрипт для спрайта кота.
 - а) Додайте команду **коли прапорець натиснуто**, що відповідає блоку **Початок** на блок-схемі.
 - б) Для введення n додайте команду **запитати Скільки днів я буду відпочивати? і чекати** та **надати n значення відповідь** (рис. 3.145).
 - в) Додайте команди надання змінній i значення **1** та надання змінній S значення **0** (рис. 3.146).

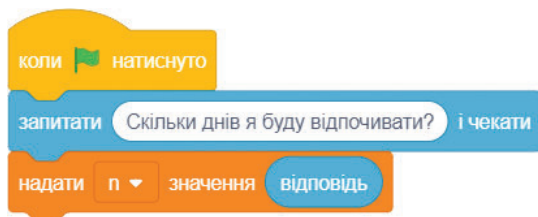


Рис. 3.145

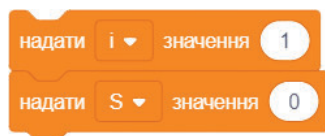


Рис. 3.146

- г) Переходимо до реалізації циклу з лічильником. Для цього додайте блок **повторити 10** і замість значення **10** установіть n .
- г) Усередині цього блоку сформууйте команди **змінити S на i** та **змінити i на 1** (рис. 3.147).
- д) Додайте команду виведення кількості камінців (рис. 3.148).

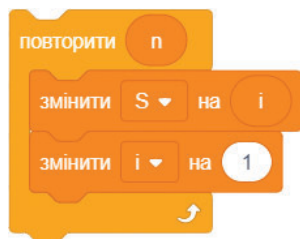


Рис. 3.147



Рис. 3.148

4. Запустіть проєкт на виконання. Введіть **5**. Ви маєте одержати **15**.

Удосконалимо проєкт таким чином, аби в кінці кожного дня кіт повідомляв, скільки в нього вже є камінців.

Нехай нам необхідно сформуувати фразу у форматі: у 1-й день — 1, у 2-й день — 3, у 3-й день — 6 і т. д.

Оскільки повідомлення буде змінюватися в залежності від номера дня, його слід розташувати у тілі циклу.



5. Додайте команду виведення фрази у форматі: у 1-й день — 1, у 2-й день — 3, у 3-й день — 6 і т. д.
6. Аби повідомлення було видно, додайте блок затримки часу **чекати 1 секунд**.
7. Самостійно визначте, куди необхідно додати створені команди, аби проєкт працював коректно (рис. 3.149).
8. Запустіть проєкт. Уведіть 4. Кіт має виводити інформацію про кількість камінців, яку він має наприкінці кожного дня, 4 рази. В останній день це має бути число $1 + 2 + 3 + 4 = 10$.
9. Звичайно, проєкт став би цікавішим, якби ми також бачили процес появи камінців у кота наприкінці кожного дня, для цього створіть скрипт для камінців, попередньо додавши команду коту **оповістити камінці** (рис. 3.150).

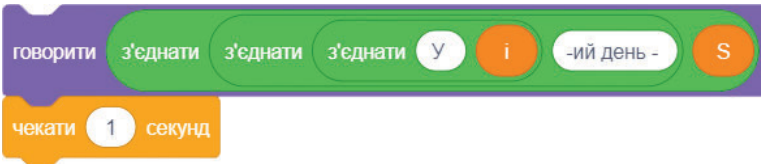


Рис. 3.149



Рис. 3.150

10. Для спрайта камінця додайте команду **коли я отримую камінці**.
11. Наступною додайте команду **повторити 10**. Кількість повторень установіть **і** (що відповідає кількості камінців, які буде додано в поточний день).
12. В середині цього блоку додайте команду **показати** для відображення камінця, наступна команда **змінити x на 10**, для розташування наступного камінця правіше від попереднього.
13. Далі додайте команду **штамп** (саме вона і створить копію спрайта).
14. Для зміни виду камінця додайте команду **змінити образ на випадкове від 1 до 3** (ця команда буде обирати один із наявних у нас трьох образів камінців) (рис. 3.151).
15. Запустіть проєкт. Уведіть значення 5. Камінці послідовно мають з'явитися на сцені.
Якщо проєкт запустити ще раз, то камінці будуть продовжувати додаватися до тих, що вже були до цього.
16. Аби цього не відбувалося, створіть для камінця скрипт, у якому при запуску проєкту камінці відображатимуться із

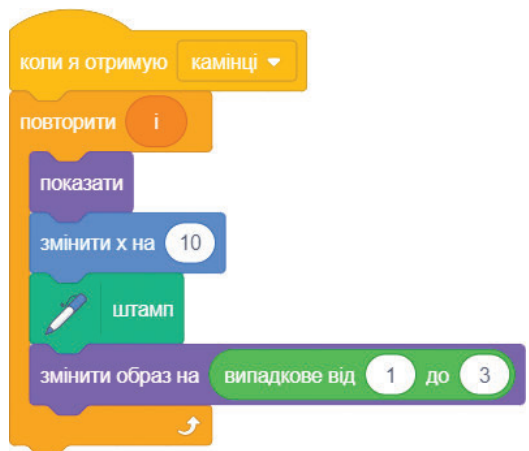


Рис. 3.151



самого початку. Для цього додайте команди **коли прапорець натиснуто**, **очистити все** — для вилучення усіх копій камінця, **сховати** — для приховування самого спрайта, та **перемістити в $x=-200$ $y=-150$** (рис. 3.152).

17. Запустіть проєкт ще раз та введіть число 8.

Тепер збирання камінців має відбуватися спочатку, але вони не усі помістяться на сцені. Отже, проєкт вимагає ще деякого вдосконалення.

18. Додайте команди, при виконанні яких, якщо камінець досягає межі, то він переміщується в початкову точку по координаті x і на 20 одиниць угору по осі y .

а) Всередину циклу додайте блок **якщо...то...**. В якості умови додайте блок **торкається межа**.

б) Після гілки **то** додайте команди: **змінити y на 20**, **задати x -200** (рис. 3.153).



Рис. 3.152

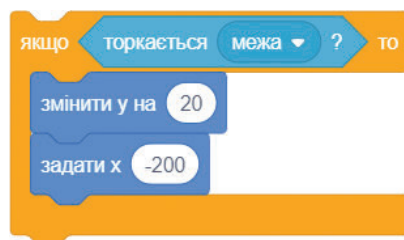


Рис. 3.153



Виконати вправи 24 та 25 ви зможете в електронному додатку.



19. Запустіть проєкт. Уведіть число 10. Тепер камінці мають відображатися у декілька рядів.

20. Самостійно вдоскональте цей проєкт, зробивши так, аби кіт, повідомляючи результат, належним чином відмінював слово «камінець» (наприклад: 1 камінець, 3 камінці, 6 камінців тощо) (рис. 3.154).

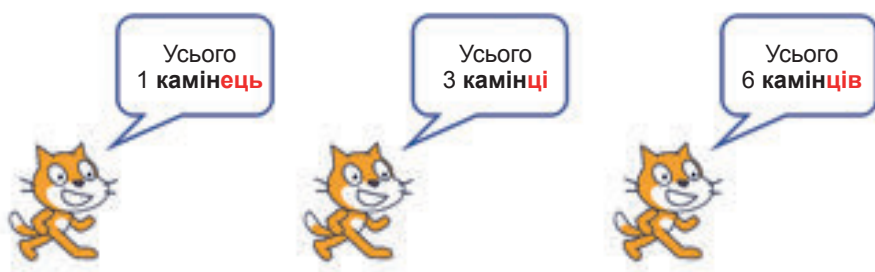


Рис. 3.154

21. Збережіть файл з ім'ям **Вправа_23**.

**Висновки**

У програмуванні багаторазове повторення певної послідовності називають **циклом**.

Послідовність команд, які повторюються під час виконання оператора циклу, називають **тілом циклу**.

Найчастіше при створенні програм у Scratch використовують безумовні цикли, для яких використовується команда **завжди**.

Цикл з лічильником дає змогу організувати у програмі повторення певної послідовності дій задану кількість разів. Такі цикли використовують у випадках, коли кількість виконуваних повторювань відома заздалегідь. Для реалізації цього виду циклів в Scratch використовують команду **повторити**.

**Контрольні запитання та завдання**

1. Що таке цикл у програмуванні?
2. Що таке тіло циклу?
3. У чому полягає принцип виконання дій у безумовному циклі?
4. Яким чином можна зупинити виконання безумовного циклу?
5. У чому полягає принцип виконання дій у циклі з лічильником?

§30.**Цикли з умовами**

Програмісти часто стикаються із ситуаціями, коли кількість повторювань наперед не відома, а команди у програмі мають повторюватися за виконання певних умов. Саме в таких випадках стають у нагоді *цикли з умовами*.

Цикл з умовою в Scratch представлений з допомогою оператора **повторити до...**

Блок-схема та відповідний Scratch-блок зображені на рис. 3.155.

У цьому циклі тіло циклу буде виконуватися доти, доки не виконається умова. При виконанні умови цикл завершується. Якщо умова виконається відразу, то тіло циклу не виконається жодного разу.



Рис. 3.155. Блок-схема (а) та відповідний Scratch-блок (б) циклу завжди якщо

Розглянемо використання циклу з умовою, виконуючи наступну вправу.

ВПРАВА 26



Завдання. Створити проєкт реалізації наступних подій.



Рис. 3.156

Хлопчик підкидає гральну кістку-кубик, на шести гранях якого нанесено комбінації крапок, що символізують числа 1, 2, 3, 4, 5, 6. Необхідно визначити, яка знадобиться кількість підкидань кубика до досягнення заданої користувачем загальної суми крапок. Оскільки при підкиданні кубика немає гарантії, що загальна сума крапок буде рівно такою, яку ввів користувач, то будемо вважати, що як тільки сума стане дорівнювати вказаному значенню або перевищить його, підрахунок кількості підкидань закінчується.

I. Визначимо вхідні та вихідні дані.

Відповідно до умови дано число, яке необхідно одержати, підкидаючи кубик. Позначимо його змінною x .

Також в якості даних є значення, яке ми будемо отримувати при кожному підкиданні кубика, його позначимо змінною a .

Результат виконання проєкту, кількість підкидань до досягнення заданого числа надамо змінній k .

Також нам знадобиться проміжна змінна S , яка буде підраховувати суму набраних очок після кожного підкидання кубика.

II. Розглянемо блок-схему реалізації даного проєкту.

1. Уведення змінної x , що приймає значення, до якого слід підкидати кубик.
2. Установлення початкових значень для змінних S та k .
3. Здійснення перевірки умови, що значення суми S , одержаної в результаті підкидань кубика, дорівнює або більше за уведене користувачем значення x .
4. Якщо умова є хибною, то здійснюється введення значення змінної a , що є випадковим числом від 1 до 6, одержаного в результаті одного підкидання кубика.



5. Надання змінній S суми попереднього значення S та значення a .
6. Збільшення значення змінної k на 1.
7. Виведення поточного значення змінної k .
8. Після цього знову здійснюється перехід до перевірки умови $S \geq x$.
9. Тіло циклу буде виконуватися до тих пір, поки умова не стане істинною.
10. Як тільки умова стане істинною, буде здійснено вихід із циклу і виконання проєкту буде завершено.

III. Створіть проєкт реалізації даного алгоритму.

1. Додайте до проєкту спрайти хлопчика та кубика. Для спрайта кубика додайте ще п'ять його образів (файли із зображеннями кубика ви можете взяти в електронному додатку до підручника).
2. Створіть скрипт для спрайта хлопчика.

- а) Додайте команду **коли прапорець натиснуто**.
- б) Уведіть значення числа, додавши команди **запитати Уведіть число** і **чекати**.
- в) Надайте змінним S і k значення 0 з використанням команд **надати значення**. Ці змінні будуть мати початкове значення 0 , оскільки до підкидання кубика кількість підкидань та сума набраних очок при підкиданні дорівнюють нулю (рис. 3.159).

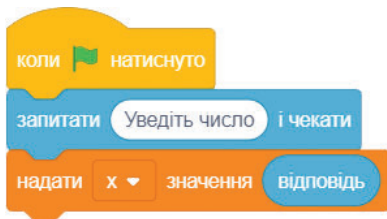


Рис. 3.158

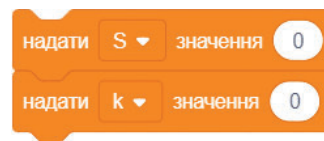


Рис. 3.159

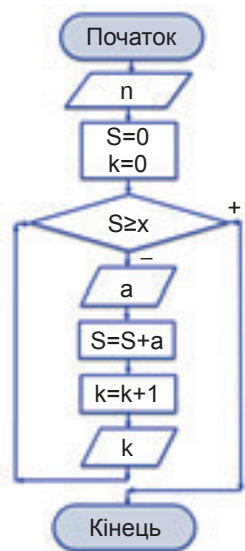


Рис. 3.157

- г) Додайте блок **повторити до ...**. Створіть у ньому умову, в якій необхідно перевірити, що набрана сума очок при підкиданні дорівнює або більша, ніж уведене користувачем число. Отже, математичною мовою це буде команда порівняння $S \geq x$ (рис. 3.160).



Рис. 3.160



Наступні команди розташуйте всередині блоку **завжди якщо**.

- г) Надайте значення змінній **a** з допомогою команди **надати a значення випадкове від 1 до 6**.
 - д) Аби це значення додавалося до змінної **s**, застосуйте команду **змінити s на a**.
 - е) Також додайте команду **змінити k на 1** для підрахунку кількості підкидань кубика.
 - є) Додайте команду **говорити k**, що буде повідомляти останнє значення кількості підкидань.
 - ж) Додайте команду **чекати 1 секунд**, що дозволить демонструвати підрахунок кількості підкидань з інтервалом 1 секунда (рис. 3.161).
3. Цікаво було б побачити в проєкті зміну граней на кубуку, відповідно до значення, яке на ньому випадає. Для цього всередину блоку **повторити до...** додайте команду **оповістити кубик** і перейдіть до створення скрипта кубика (рис. 3.162).
- а) Спочатку додайте команду **коли я отримую кубик**.
 - б) Наступною додайте команду **змінити образ на a**. Тут **a** — значення випадкового числа від 1 до 6, яке буде одержано при виконанні відповідної команди для спрайта хлопчика. Оскільки образів кубика також шість і кожен його номер відповідає кількості зображених на кубуку крапок, то ви побачите на кубуку кількість крапок, що дорівнює значенню змінної **a** (рис. 3.163).

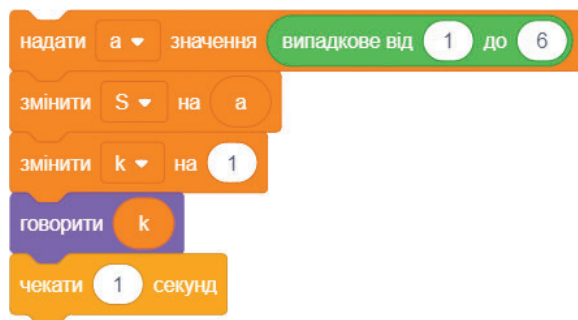


Рис. 3.161



Рис. 3.162

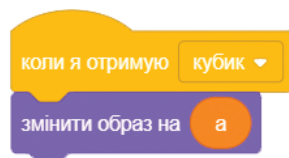


Рис. 3.163



Виконати вправи 27, 28 та 29 ви зможете в електронному додатку.



4. Запустіть проєкт на виконання. Введіть число 15. Ви маєте побачити, що значення кубика змінюється, а хлопчик повідомляє кількість підкидань. Як тільки значення **S** стане рівним або більше за введене вами значення **x**, тіло циклу перестане виконуватися.
5. Збережіть файл з ім'ям **Вправа_26**.



Висновки

Цикли, тіло яких виконується за певних умов, називають **циклами з умовами**.

Цикл, тіло якого виконується доти, доки не буде істинною умова, реалізується з допомогою блоку **повторити до...**



Контрольні запитання та завдання

1. Що таке цикл з умовою?
2. У чому полягає принцип виконання дій у циклі **повторити до...**?
3. У чому відмінність у циклі з лічильником та у циклі з умовою?

§31.

Вкладені цикли

31.2.

ВКЛАДЕНІ ЦИКЛИ

Іноді для створення більш складних проєктів цикли вкладають один в одного. Цикли, так само як і умовні оператори, можуть бути вкладеними. У цьому випадку розрізняють зовнішній і внутрішній цикли (рис. 3.164).

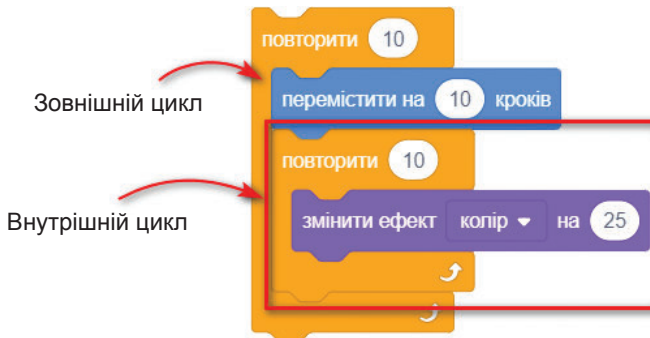


Рис. 3.164

Розглянемо вкладені цикли, створюючи проєкти різноманітних зображень.

ВПРАВА 30



Завдання. Створити проєкт побудови квадрата з використанням оператора циклу (рис. 3.165).

1. Створіть скрипт для побудови сторони і наступного повороту спрайта кота на 90°. Для цього додайте команди **перемістити на 100 кроків** та **поворот на 90 градусів** (рис. 3.166).



Рис. 3.165



РОЗДІЛ 3. Алгоритми та програми

- Оскільки усі сторони квадрата рівні, то попередні команди необхідно виконати підряд 4 рази. Для цього додайте блок **повторити 4** (рис. 3.167).

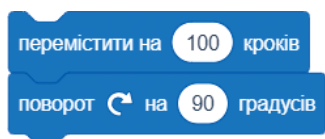


Рис. 3.166

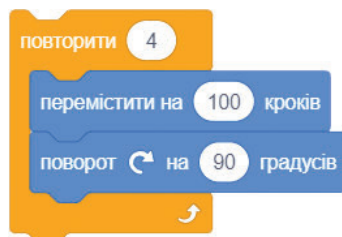


Рис. 3.167

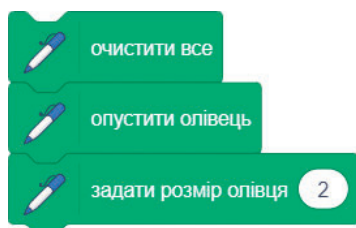


Рис. 3.168

- Перед блоком циклу додайте команди **очистити все** та **опустити олівець**.
- Щоб квадрат було краще видно, додайте команду **задати розмір олівця 2** (рис. 3.168).
- Додайте команду **коли натиснуто прапорець** та запустіть проект. У результаті має бути побудовано квадрат.
- Збережіть файл з ім'ям **Вправа_30**.

ВПРАВА 31



Завдання. Вдосконалимо попередній проект, зробивши так, аби було у рядок побудовано стільки квадратів, скільки цього забажає користувач (рис. 3.169).



Рис. 3.169

Саме тут стануть у нагоді вкладені цикли.

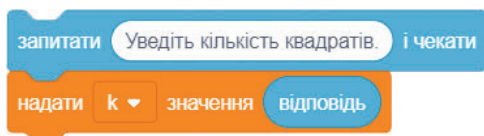


Рис. 3.170

- Створіть змінну **k**, яка буде визначати кількість квадратів.
- Для надання змінній **k** значення числа, введеного користувачем, додайте команди **запитати Уведіть кількість квадратів і чекати** і **надати k значення відповідь** (рис. 3.170).
- Нижче додайте блок **повторити**, кількість кроків установіть — **k** (кількість квадратів установлена користувачем).
- Усередині цього блоку розташуйте цикл побудови квадрата, створений у попередньому проєкті.
- Змініть у ньому значення сторони квадрата на **50**, щоб можна було розташувати декілька квадратів у рядок (рис. 3.171).



Рис. 3.171

6. Аби після побудови поточного квадрата кіт переміщався на певну відстань, додайте команди **підняти олівець** та **змінити x на 60**. Тут необхідно встановити число 60, яке буде дещо більше довжини сторони квадрата (довжина сторони квадрата 50), оскільки після побудови квадрата кіт має перестрибнути через нього, а потім ще зробити невеличкий відступ (рис. 3.172).
7. Аби кожного разу не повертати kota за хвіст у початкове положення, додайте перед побудовою команду **задати x-200** (рис. 3.173).
8. Запустіть проєкт. Уведіть кількість квадратів 5.

У результаті кіт буде малювати лише один квадрат та переміщатися далі, не залишаючи сліду.

9. Для вирішення цієї проблеми перемістіть команду **опустити олівець** до тіла зовнішнього циклу.
10. Запустіть проєкт ще раз та встановіть кількість квадратів 5. Тепер кіт має намалювати вказані 5 квадратів.
11. Збережіть файл з ім'ям **Вправа_31**.

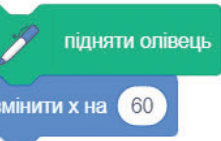


Рис. 3.172



Рис. 3.173

ВПРАВА 32



Завдання. Ще додамо зміни у попередній проєкт. Будемо малювати не тільки квадрати, а й правильні багатокутники, причому кількість сторін багатокутника буде вводити користувач (рис. 3.174 та рис. 3.175).

1. Змініть у команді **запитати і чекати** фразу «Уведіть кількість квадратів» на «Уведіть кількість багатокутників».
2. Додайте змінну **n**, яка має визначати кількість сторін багатокутника.



Рис. 3.174



Рис. 3.175



3. Продублюйте попередні дві команди, змінивши фрази запитів та імена змінних (рис. 3.176).

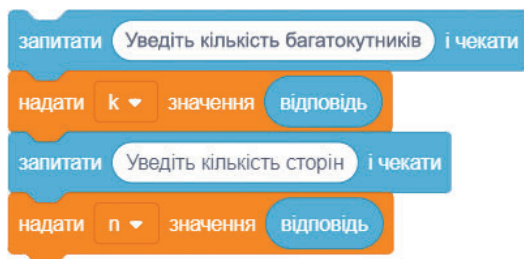


Рис. 3.176

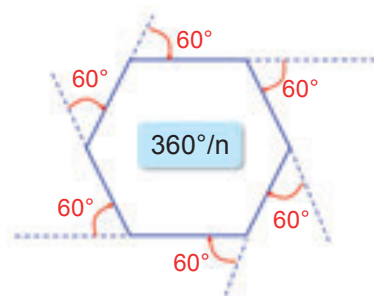


Рис. 3.177

Оскільки в даному проєкті ми хочемо мати справу не з квадратом, а з n -кутником, доцільно змінити кут обертання.

Отже, яким він має бути.

Будуючи багатокутник, по суті, кіт усього робить загальний кут оберту 360° .

При кожній вершині багатокутника це буде однакове значення. Отже, якщо таких кутів 6, то кут буде 60° , якщо кутів 4, то — 90° . Неважко здогадатися, що для правильного n -кутника такий кут повороту дорівнюватиме $360/n$ градусів (рис. 3.177).



Рис. 3.178

4. Отже, в команді **поворот на 90 градусів** змініть значення **90** на **$360/n$** (рис. 3.178).
5. Відповідно кількість повторень внутрішнього циклу змініть з **4** на **n** .
6. Для того, аби багатокутники були зображені різними кольорами, додайте в тіло зовнішнього циклу команду **змінити колір олівця на 20**.
7. Запустіть проєкт. Уведіть кількість багатокутників, наприклад **5**, а кількість сторін **6**. Кіт має намалювати 5 шестикутників.
8. Поекспериментуйте з введенням інших значень n і k . Спробуйте також змінити значення в командах **перемістити на ... кроків** та **змінити x на ...**



Виконати вправу 33 ви зможете в електронному

додатку.



9. Збережіть файл з ім'ям **Вправа_32**.

8*. Самостійно змініть проєкт таким чином, аби користувач також міг вводити довжину сторони багатокутника та відступ між багатокутниками.

9*. Збережіть файл з ім'ям **Вправа_32_1**.



ВПРАВА 34



Завдання. Створити зображення, використовуючи вкладені цикли.

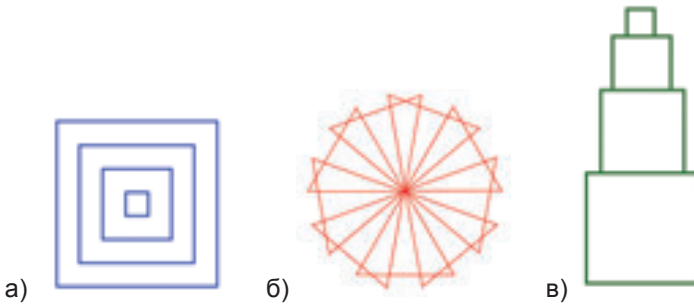


Рис. 3.179

Збережіть файли з іменами **Вправа_34_а**, **Вправа_34_б**, **Вправа 34_в**.



Висновки

Цикли, так само як і умовні оператори, можуть бути вкладеними. Розрізняють зовнішній і внутрішній цикли.

З допомогою вкладених циклів можна значно спростити побудову складних зображень.



Контрольні запитання та завдання

1. Які команди використовують у Scratch для виконання різних видів циклів?
2. Опишіть та порівняйте принципи виконання різних видів циклів.
3. Як ви думаєте, чому всі сучасні мови програмування використовують команди реалізації циклів?

§32.

Опис моделі у середовищі програмування

Вивчаючи електронні таблиці, ви навчилися створювати комп'ютерні моделі з використанням засобу MS Excel. Як уже зазначалося раніше, створення однієї й тієї ж моделі можливе засобами різних програмних додатків. Розглянемо створення комп'ютерних моделей з використанням середовища Scratch.



ВПРАВА 35

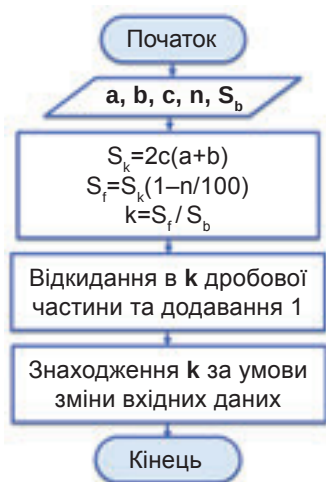


Рис. 3.180

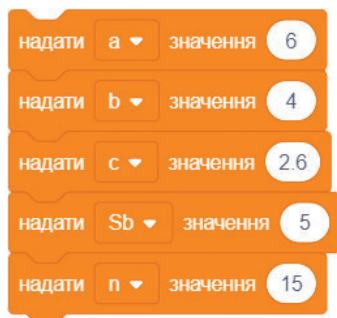


Рис. 3.181



Завдання. Створити комп'ютерну модель розв'язання задачі в середовищі Scratch.

Задача. Необхідно пофарбувати стіни прямокутної кімнати. Скільки необхідно придбати банок фарби, якщо відомо, що розміри кімнати (600x400x260) см; 15% площі кімнати — вікна та двері; одна банка фарби призначена для фарбування 5 м² стін.

Перші 3 етапи побудови моделі ми вже виконали вище (див. §20). Отже, відразу перейдемо до етапу побудови комп'ютерної моделі в середовищі Scratch, скориставшись блок-схемою розв'язання задачі.

4 етап. Створення комп'ютерної моделі.

1. Створіть змінні a , b , c , n , S_b , S_k , S_f , k .
2. Додайте команду **коли натиснуто прапорець**.
3. Аби при створенні та тестуванні проєкту не витратили кожного разу час на введення даних, надайте вхідним даним сталі значення у скрипті проєкту (рис.3.181).
4. Додайте скрипт обчислення площі S_k стін (рис.3.182).



Рис. 3.182

5. Додайте скрипт обчислення площі S_f стін, які необхідно пофарбувати (рис.3.183).



Рис. 3.183

6. Додайте скрипт обчислення кількості k банок фарби (рис.3.184).

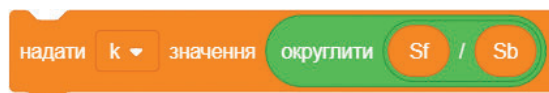


Рис. 3.184

7. Додайте скрипт виведення кількості k банок фарби (рис.3.185).

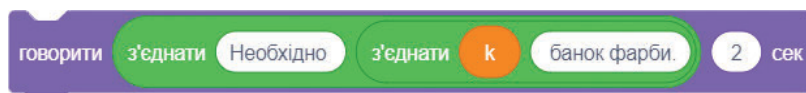


Рис. 3.185



8. Запустіть проєкт на виконання.

В результаті виконання обчислень ви маєте одержати значення **8,8**, отже, для фарбування стін кімнати із заданими параметрами необхідно придбати **9** банок фарби.

Аби кіт виводив відразу ціле значення кількості банок удосконалимо скрипт обчислення змінної **k**.

Логіка виконання цієї дії буде такою:

- **якщо** дробова частина одержаного результату (S_f / S_k) більша за 0,5, **то** округлити дане число до цілого,
- **інакше** округлити дане число до цілого та додати 1.

Для округлення виразу до цілого числа в Scratch використовується команда **округлити**. А як же перевірити значення дробової частини? Для знаходження дробової частини від числа слід скористатися наступною конструкцією (рис.3.186):



Рис. 3.186

9. Змініть скрипт знаходження кількості **k** банок.

- Додайте блок **якщо...то...інакше...**.
- Сформуйте у ньому умову: якщо дробова частина від ділення S_f на S_k більша за 0,5 (рис.3.187);



Рис. 3.187

- Після гілки **то** додайте скрипт округлення дії S_f / S_k (рис.3.188);

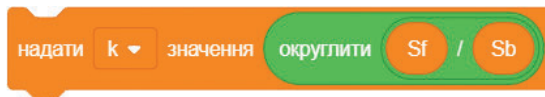


Рис. 3.188

- Після гілки **інакше** додайте скрипт додавання округлення дії S_f / S_k до 1 (рис.3.189);



Рис. 3.189



РОЗДІЛ 3. Алгоритми та програми

10. Запустіть проєкт на виконання. В результаті виконання обчислень ви маєте одержати значення 9.
11. Збережіть файл з ім'ям **Вправа_35**.

5 етап. Проведення комп'ютерного експерименту.

1. Визначте, скільки банок знадобиться, якщо фарби вистачить для фарбування 3 м^2 , 4 м^2 , $7,5 \text{ м}^2$ стін.
Для цього змініть команди надання значень змінним **a**, **b** і **c**.
2. Визначте, скільки фарби необхідно придбати на другу кімнату, що має розміри (450x350x260) см, а вікна та двері займають 10 % площі кімнати. Значення витрат фарби 5 м^2 .
Для цього змініть команди надання значень змінним **a**, **b** і **c**, увівши **a=4,5**, **b=3,5**, **c=2,6**, значення площі, що не потребує фарбування, **n** увести 10.
3. Збережіть файл.

ВПРАВА 36



Завдання. У середовищі Scratch створити комп'ютерну модель задачі із вправи 23 розділу 2.

Збережіть файл з ім'ям **Вправа_36.xlsx**.



Висновки

Побудова комп'ютерної моделі складається з п'яти етапів:

- 1) постановка задачі та її аналіз;
- 2) побудова інформаційної моделі;
- 3) розробка алгоритму дослідження моделі;
- 4) розробка комп'ютерної моделі;
- 5) проведення комп'ютерного експерименту.



Контрольні запитання та завдання

1. Що таке комп'ютерне моделювання?
2. Що таке комп'ютерна модель?
3. Опишіть суть етапів комп'ютерного моделювання.
4. Вкажіть відомі вам програмні засоби, що дозволяють створити комп'ютерну модель.



Питання для роздумів

1. Як ви думаєте, чи можна для створення однієї комп'ютерної моделі використовувати різні програмні засоби?
2. Як ви думаєте, чому для розв'язання однієї й тієї ж задачі іноді використовують різні комп'ютерні моделі?

ЗМІСТ

РОЗДІЛ 1. Служби Інтернету	5
§1. Поштові служби Інтернету	6
§2. Створення електронної скриньки. Робота з електронною поштою	12
§3. Адресна книга. Етикет електронного листування. Правила безпечного користування електронною скринькою	21
§4. Практична робота №1. Електронне листування з використанням вебінтерфейсу. Вкладені файли	26
§5. Хмарні технології. Онлайнкові перекладачі	29
§6. Інтернет речей	37
РОЗДІЛ 2. Опрацювання табличних даних	43
§7. Ознайомлення з електронними таблицями	44
§8. Редагування електронних таблиць	50
§9. Форматування електронних таблиць	55
§10. Формати даних	60
§11. Автозаповнення та автозавершення	65
§12. Практична робота №2. «Уведення, редагування та форматування даних у середовищі табличного процесора»	72
§13. Виконання обчислень	74
§14. Практична робота №3. «Виконання обчислень»	79
§15. Вбудовані функції	81
§16. Використання вбудованих функцій	88
§17. Помилки у формулах	93
§18. Практична робота №4. «Виконання обчислень за даними електронної таблиці»	95
§19. Моделі. Етапи побудови моделей	97
§20. Реалізація математичних моделей	105
РОЗДІЛ 3. Алгоритми та програми	109
§21. Повторення роботи в середовищі Scratch	110
§22. Типи даних	116
§23. Поняття величин. Константи та змінні	121
§24. Присвоєння значень величинам	125
§25. Практична робота №5. «Розв'язування задач на обчислення»	132
§26. Логічні величини. Вказівки розгалуження	135
§27. Вкладені розгалуження	143
§28. Практична робота №6. «Складання та виконання алгоритмів з розгалуженням»	151
§29. Поняття циклу. Цикл з лічильником	155
§30. Цикли з умовами	163
§31. Вкладені цикли	167
§32. Опис моделі у середовищі програмування	171

Відомості про стан підручника

№	Прізвище та ім'я учня/учениці	Навчальний рік	Стан підручника		Оцінка
			на початку року	в кінці року	
1					
2					
3					
4					
5					

Навчальне видання

КАЗАНЦЕВА Ольга Павлівна
СТЕЦЕНКО Ірина Володимирівна

ІНФОРМАТИКА

Підручник для 7 класу
закладів загальної середньої освіти

Рекомендовано Міністерством освіти і науки України

Видано за рахунок державних коштів. Продаж заборонено

Головний редактор  *Богдан Бодний*

Редактор *Володимир Дячун*

Обкладинка *Ростислава Крамара*

Комп'ютерна верстка *Андрія Кравчука*

Художній редактор *Ростислав Крамар*

Технічний редактор *Неля Домарецька*

Підписано до друку 21.  20. Формат 70×100/16. Папір офсетний. Гарнітура CentSchbook Win95BT.
Друк офсетний. Умовн. друк арк. 14,30. Умовн. фарбо-відб. 57,20. Обл.-вид. арк. 9,41.
Тираж 17 613. Зам. 343/06.

Видавництво «Навчальна книга–Богдан»

Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру видавців, виготівників
і розповсюджувачів видавничої продукції ДК №4221 від 07.12.2011р.

Навчальна книга – Богдан, просп. С. Бандери, 34а, м. Тернопіль, 46002

Навчальна книга – Богдан, а/с 529, м. Тернопіль, 46008

тел./факс (0352) 52-06-07, 52-05-48

office@bohdan-books.com

www.bohdan-books.com

Надруковано на ПП «Юнісофт», вул. Морозова, 136, м. Харків, 61036

Свідоцтво серія ДК №3461 від 14.04.2009 р.