



ФОРМИ ПОДАННЯ АЛГОРИТМІВ

Ви вже знаєте, що команди виконавцям можуть подаватися різними способами: спонукальними реченнями, звуковими або світловими сигналами, жестами, вибором команди меню або кнопки вікна та іншими.

Відповідно до цього алгоритми можуть бути подані в різних формах:

- словесній;
 - послідовністю спеціальних сигналів;
 - графічній
- та іншій.

Алгоритми приготування яєчні, отримання 1-го літра рідини в посудині та інші, розглянуті в попередньому пункті, були подані у **словесній формі**. У такій формі подання команди алгоритму записуються у вигляді послідовності спонукальних речень у певному порядку.

Прикладом алгоритму, поданому **послідовністю світлових сигналів**, є послідовність сигналів світлофора на переході:

1. Червоне світло (Стійте).
2. Жовте світло (Готуйтеся до переходу дороги).
3. Зелене світло (Переходьте дорогу).

Прикладом алгоритму, поданому **послідовністю звукових сигналів**, є послідовність дзвінків у театрі перед початком вистави:

1. Перший дзвінок (Готуйтеся зайти до глядацького залу).
2. Другий дзвінок (Заходьте до глядацького залу і займайте свої місця).
3. Третій дзвінок (Розпочати виставу).

Ще однією формою подання алгоритму є **графічна форма**, або подання алгоритму у вигляді **блок-схеми**.

У блок-схемі алгоритму кожна команда записується в геометричній фігурі (блоці) певного вигляду. Блоки з'єднуються між собою стрілками, що вказують, яку команду алгоритму потрібно виконати наступною. Наведемо деякі елементи (блоки) блок-схеми алгоритму (табл. 1.3).

Таблиця 1.3

Деякі елементи (блоки) блок-схеми алгоритму

Назва	Позначення	Призначення
Термінатор		Початок або кінець алгоритму
Процес		Виконання однієї або кількох команд
Дані		Уведення даних або виведення результатів

Ось як виглядає блок-схема алгоритму отримання в посудині 1-го літра рідини, розглянутого в пункті 1.3 (рис. 1.19):



Рис. 1.19. Блок-схема алгоритму отримання 1-го літра рідини

Складемо алгоритм обчислення значення виразу $73 + (37 - 12) \cdot 15$ для виконавця, до системи команд якого входять команди:

- 1) Виконувати арифметичні операції.
- 2) Запам'ятовувати результат обчислення значення арифметичних операцій.
- 3) Повідомляти запам'ятовані результати.

Подамо цей алгоритм у словесній і в графічній формах. У словесній формі цей алгоритм виглядатиме так:

1. Обчислити $37 - 12$ і запам'ятати результат.
2. Помножити запам'ятований результат на 15 і запам'ятати результат.
3. Додати останній запам'ятований результат до 73 і запам'ятати результат.
4. Повідомити останній запам'ятований результат.

Блок-схема цього алгоритму виглядатиме так (рис. 1.20):



Рис. 1.20. Блок-схема алгоритму обчислення значення виразу $73 + (37 - 12) \cdot 15$

РЕДАГУВАННЯ АЛГОРИТМІВ У ПРОГРАМІ SCRATCH

Розглянемо детальніше область скриптів вікна програми Scratch. У цій області розташовано три вкладки: **Скрипти**, **Образи** і **Звуки** (рис. 1.21). Щоб відкрити вміст вкладки, потрібно вибрати її ім'я.

Ви вже знаєте, що вкладка **Скрипти** призначена для розміщення в ній блоків з командами алгоритму для виконавця. Нагадаємо, що їх можна розмістити на цій вкладці, перетягнувши з палітри блоків.



Рис. 1.21. Вкладки області скриптів

Алгоритм, розміщений на вкладці **Скрипти**, можна редагувати, а саме видаляти блоки з командами, переміщувати їх, копіювати, додавати нові, змінювати значення в полях блоків та інше.

Щоб вилучити один з блоків з командою алгоритму, потрібно:

1. Вибрати кнопку **Вирізати**  на панелі інструментів.
2. Вибрати блок, який потрібно вилучити (рис. 1.22).

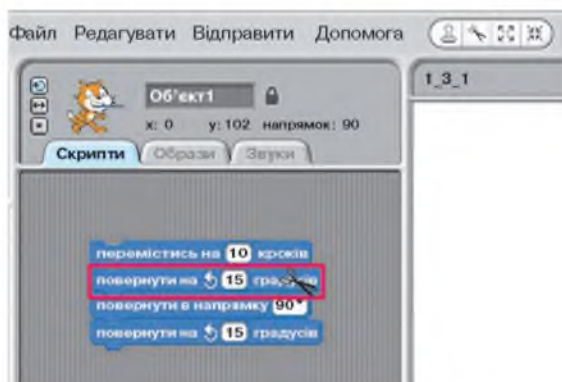


Рис. 1.22. Вилучення блока

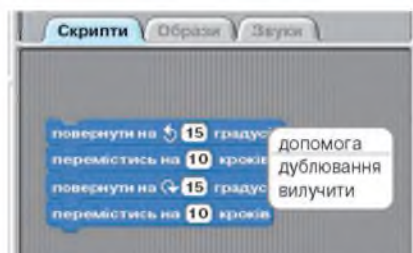


Рис. 1.23. Контекстне меню першого блока групи блоків

Щоб вилучити групу блоків, потрібно відкрити контекстне меню першого блока групи і вибрати команду **вилучити** (рис. 1.23).

Для вставлення блока до алгоритму потрібно перетягнути його з **палітри блоків** до **області скриптів** і розмістити в потрібному місці в алгоритмі. При цьому місце, до якого може бути вставлено блок, позначатиметься товстою білою лінією.

Блок або групу блоків можна копіювати. Для цього потрібно:

1. Відкрити контекстне меню блока або першого блока групи.
2. Вибрати команду **дублювати**.
3. Перетягнути копію блока або групи блоків у потрібне місце алгоритму (рис. 1.24).
4. Кладнути ліву кнопку миші.

Блок або групу блоків можна переміщувати. Для цього потрібно:

1. Установити вказівник на блок або на перший блок групи.
2. Натиснути і тримати натиснутою ліву кнопку миші.
3. Перемістити блок або групу блоків.
4. Відпустити ліву кнопку миші.

Переміщення блока або групи бло-

ків можна використати, наприклад, для того, щоб розділити групу блоків на кілька окремих груп.



Рис. 1.24. Копіювання групи блоків

ЗБЕРЕЖЕННЯ ТА ВІДКРИВАННЯ ПРОЕКТІВ

Створений проект можна зберегти на носії даних для подальшого використання. Для цього потрібно виконати **Файл** ⇒ **Зберегти як** або **Файл** ⇒ **Зберегти**. Ці способи збереження даних на носії ви вже використовували, наприклад під час вивчення редактора презентацій у 5-му класі.



Розділ 1

Під час виконання команди **Зберегти як** відкривається вікно **Зберегти проект** (рис. 1.25), у якому потрібно вибрати папку для запису проекту і ввести в поле **Нове ім'я файлу** ім'я файлу з проектом. Доцільно зберігати проекти або в папці **Мої проекти**, або на тому носії та в тій папці, яку вам укаже вчитель.

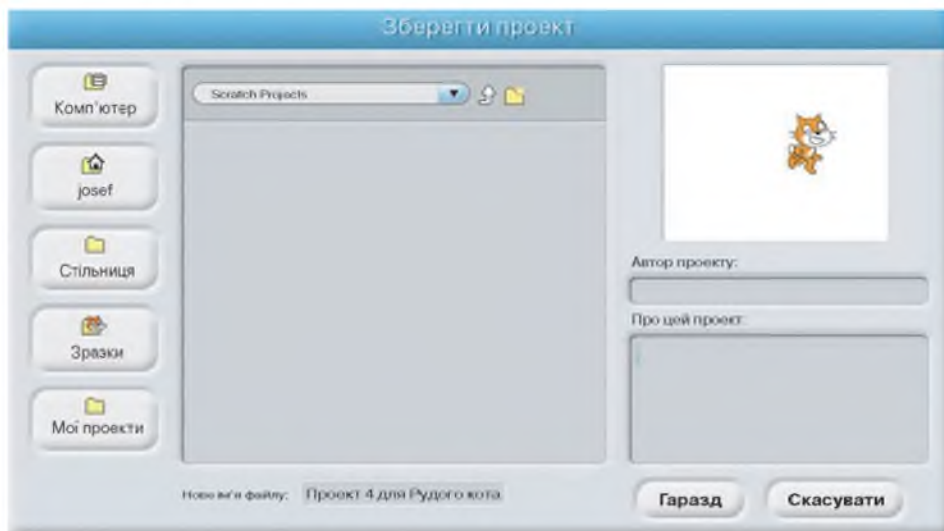


Рис. 1.25. Вікно **Зберегти проект**

Зробити поточною папку **Мої проекти** можна вибором кнопки **Мої проекти** на панелі в лівій частині вікна (рис. 1.25). А вибрати потрібний носій даних можна, відкривши список доступних носіїв даних вибором кнопки **Комп'ютер**. Вибір кнопки **Стільниця** робить поточною папку **Робочий стіл**.

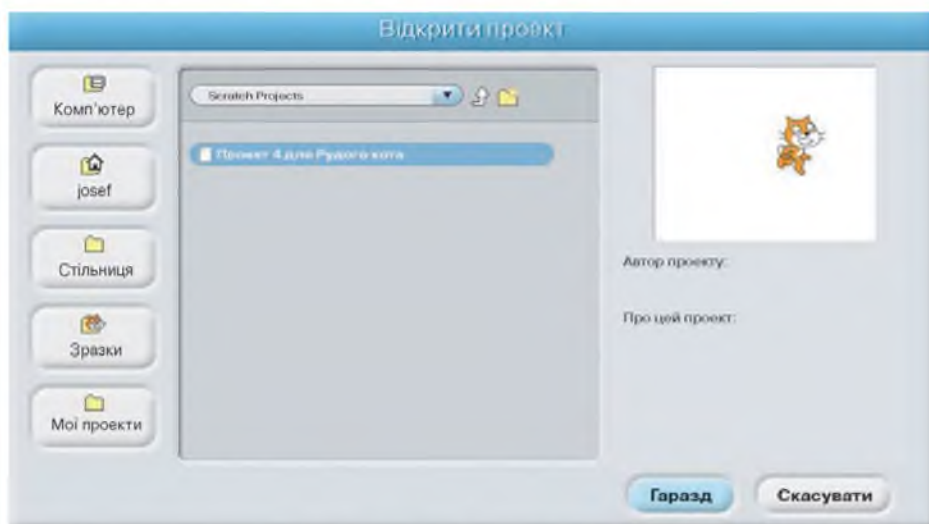
У цьому самому вікні можна в поле **Автор проекту** ввести своє прізвище, а в поле **Про цей проект** – короткий опис проекту.

Після введення даних у поля цього вікна потрібно вибрати кнопку **Гаразд**.

Збережений на носії даних файл з алгоритмом можна відкрити у вікні програми **Scratch**. Для цього потрібно виконати **Файл** ⇒ **Відкрити**. У результаті виконання цієї команди відкривається вікно **Відкрити проект** (рис. 1.26), у якому слід:

1. Відкрити вміст потрібної папки.
2. Вибрати файл із проектом.
3. Вибрати кнопку **Гаразд**.

Якщо під час відкриття збереженого на носії проекту в області **скриптів** розміщено алгоритм іншого проекту, то відкривається вікно, вибором кнопки якого можна зберегти поточний проект на носії (рис. 1.27). Таку саму ситуацію ви спостерігали, коли відкривали збережений на носії файл із зображенням у графічному редакторі **Paint**.

Рис. 1.26. Вікно **Відкрити проект**

Це саме вікно відкривається, якщо закривати програму з незбереженим проектом.

Якщо у вікні **Відкрити проект** вибрати кнопку **Зразки**, то відкривається список папок, у яких збережено файли з різноманітними зразками цікавих проектів (рис. 1.28).

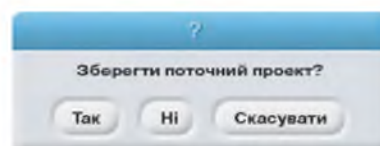
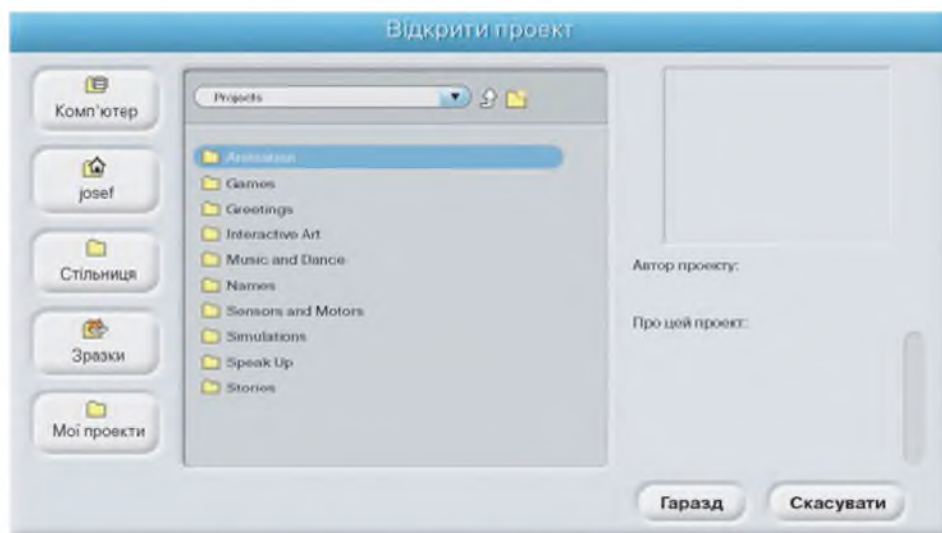
Рис. 1.27. Вікно **Зберегти поточний проект**

Рис. 1.28. Список папок з файлами зразків проектів



Розділ 1



Працюємо з комп'ютером

Увага! Під час роботи з комп'ютером дотримуйтеся правил безпеки та санітарно-гігієнічних норм.

1. Запустіть програму Scratch.
2. Відкрийте проект **вправа 1.4.1**, який міститься в папці **Розділ 1\Пункт 1.4** (рис. 1.29). Для цього:
 1. Виконайте **Файл ⇒ Відкрити**.
 2. Виберіть кнопку **Комп'ютер** у вікні, що відкрилося.
 3. Відкрийте послідовно вміст диска і папки **Розділ 1\Пункт 1.4**, у якій міститься файл з іменем **вправа 1.4.1**.
 4. Виберіть файл з іменем **вправа 1.4.1**.
 5. Виберіть кнопку **Гаразд**.

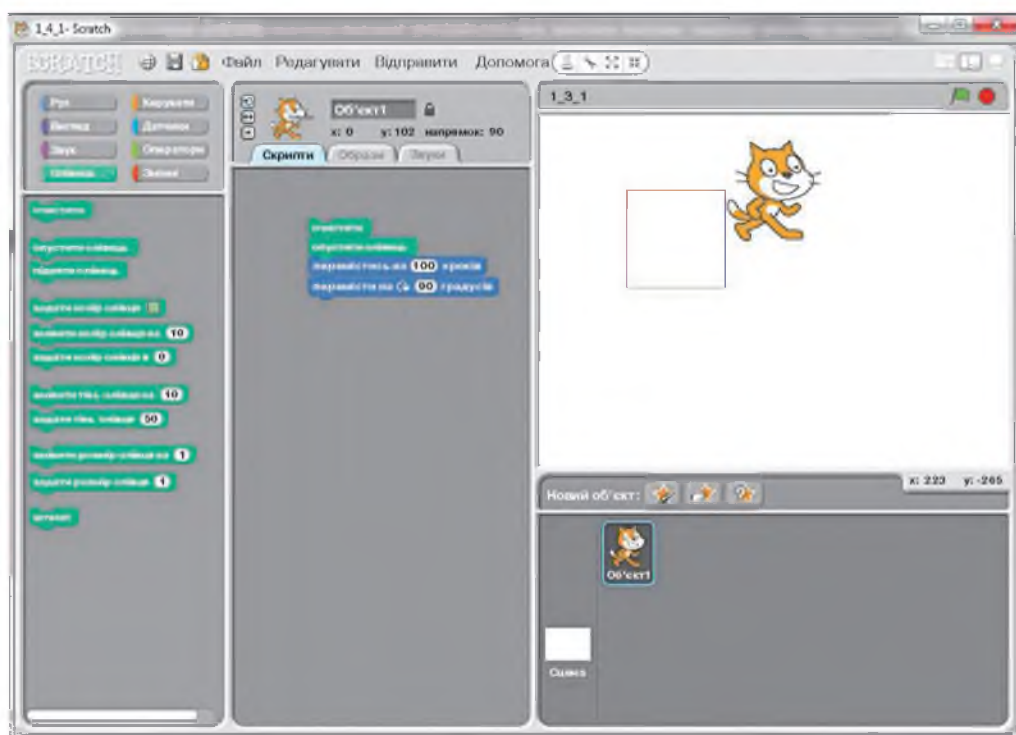







Рис. 1.29. Вікно з проектом **вправа 1.4.1**

3. Запустіть алгоритм на виконання. Що відбулося в результаті його виконання?
4. Скопіюйте в області скриптів групу з двох останніх блоків. Для цього:
 1. Відкрийте контекстне меню першої команди цієї групи блоків **переміститись на 100 кроків**.
 2. Виберіть команду **дублювати**.



3. Перемістіть нову групу команд і приєднайте її до останньої з наявних команд.
4. Клацніть ліву кнопку миші.
5. Скопіюйте групу із чотирьох останніх блоків.
6. Виконайте новий алгоритм. Що є результатом його виконання? Чим відрізняється друге виконання алгоритму від першого? Запишіть відповідь у зошит.
7. Відкрийте групу команд **Олівець**.
8. Вставте блок з командою **задати колір олівця** в алгоритм після команди **опустити олівець**. Для цього перетягніть блок із цією командою в область скриптів і розмістіть його після блока з командою **опустити олівець**.
9. Вставте блок з командою **задати колір олівця** в алгоритм після другої команди  **90 градусів**. Установіть у полі зелений колір.
10. Виконайте відредагований алгоритм. Як змінився результат виконання алгоритму? Запишіть відповідь у зошит.
11. Збережіть проект у своїй папці у файлі з іменем **вправа 1.4.1**. Для цього:
 1. Виконайте **Файл ⇒ Зберегти**.
 2. Відкрийте у вікні **Зберегти проект**, що відкрилося, уміст власної папки або папки, яку вкаже вчитель.
 3. Уведіть у поле **Нове ім'я файлу** ім'я **вправа 1.4.1**.
 4. Уведіть у поле **Автор проекту** своє прізвище.
 5. Виберіть кнопку **Гаразд**.
12. Видаліть усі блоки  **90 градусів**. Для цього для кожного такого блока:
 1. Виберіть кнопку **Вирізати**  на панелі інструментів.
 2. Виберіть блок  **90 градусів**.
13. Вставте в алгоритм після кожної команди **переміститись на 100 кроків** блок  **90 градусів**.
14. Виконайте відредагований алгоритм. Як змінився результат виконання алгоритму? Запишіть відповідь у зошит.
15. Збережіть проект у своїй папці у файлі з іменем **вправа 1.4.2**.



Найважливіше в цьому пункті

Алгоритми можуть бути подані в різних формах: словесній, графічній (блок-схема), послідовністю сигналів та інших.

У блок-схемі алгоритму кожна команда записується в геометричній фігурі (блоці) певного вигляду. Блоки з'єднуються між собою стрілками, що вказують, яку команду алгоритму потрібно виконати наступною.



Розділ 1

Алгоритми для виконавця у вікні програми **Scratch** можна редагувати:

- вилучати один з блоків з командою або групу блоків;
- копіювати один з блоків з командою або групу блоків;
- переміщувати один з блоків з командою або групу блоків;
- вставляти новий блок з командою або групу блоків.

Проект можна зберегти на носії даних для подальшого використання: **Файл** \Rightarrow **Зберегти як** або **Файл** \Rightarrow **Зберегти**.

Збережений на носії даних проект можна відкрити в середовищі **Scratch**: **Файл** \Rightarrow **Відкрити**.



Дайте відповіді на запитання

- 1°. У яких формах може бути подано алгоритм?
- 2°. У чому полягає словесна форма подання алгоритму?
- 3°. У чому полягає графічна форма подання алгоритму?
- 4°. У чому полягає подання алгоритму звуковими сигналами?
- 5°. Як вилучити блок з алгоритму?
- 6°. Як вилучити групу блоків з алгоритму?
- 7°. Як скопіювати групу блоків в алгоритмі?
- 8°. Як зберегти проект на носії даних?
- 9°. Як відкрити збережений на носії даних проект?



Виконайте завдання



- 1°. Наведіть приклади подання алгоритму в словесній формі.
- 2°. Наведіть приклади подання алгоритму послідовністю сигналів.
- 3°. Наведіть приклади алгоритмів, які ви вивчали на уроках української мови.
- 4°. Наведіть приклади алгоритмів, які ви вивчали на уроках математики.
- 5°. Складіть блок-схему наведеного алгоритму. Виконайте його.
 1. Знайдіть суму чисел 1 і 3.
 2. Додайте до одержаної суми число 5.
 3. Додайте до останньої одержаної суми число 7.
 4. Додайте до останньої одержаної суми число 9.
 5. Запишіть результат у зошит.
- 6°. Є повна посудина місткістю 8 літрів і дві порожні посудини місткістю 3 літри і 5 літрів. Складіть блок-схему алгоритму одержання в одній з посудин 2-х літрів рідини.
- 7°. Є координатний промінь з позначеними на ньому числом нуль і натуральними числами. На цьому промені мешкає виконавець **Коник**, який вміє переміщуватися по ньому, виконуючи команди: 1) стрибни на 4 одиниці праворуч; 2) стрибни на 3 одиниці ліворуч. Початкове положення **Коника** – точка 0. Складіть блок-схему алгоритму, за яким **Коник** за найменшу кількість стрибків опиниться в точці: а) 13; б) 22; в) 23.

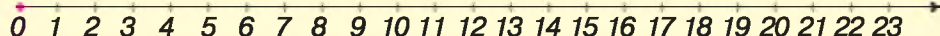


Рис. 1.30. Координатний промінь

- 8*. Складіть алгоритм обчислення значення виразу $(23 + 35) : (94 - 92)$. Подайте його у словесній і графічній формах. Виконайте складений алгоритм.
- 9*. Відкрийте один з проектів, що міститься в папці **Зразки**, і запустіть його на виконання. Що відбувається під час виконання цього проекту? Що є результатом виконання цього проекту? Запишіть результати своїх спостережень у зошит.

ПРАКТИЧНА РОБОТА № 1

«Складання алгоритмів для виконавців у словесній і графічній формах. Виконання алгоритмів»

- Виконайте алгоритм:
 - Візьміть карту України.
 - Виміряйте по карті та запишіть у зошит відстань у сантиметрах між Києвом і Донецьком.
 - Помножте виміряну відстань на масштаб, позначений на карті. Запишіть результат у зошит.
 - Поділіть отриманий добуток на 100 000, щоб отримати відстань у кілометрах.
 - Запишіть у зошит отриману відстань.
- Складіть алгоритм обчислення значення виразу $22 + 34 : (72 - 55)$. Запишіть алгоритм у словесній формі. Виконайте алгоритм.
- Складіть блок-схему алгоритму знаходження середнього арифметичного чотирьох чисел. Виконайте алгоритм для чисел 2,5; 1; 6; 0,3.



1.5. АЛГОРИТМИ В НАШОМУ ЖИТТІ. ПЛАНУВАННЯ ДІЯЛЬНОСТІ



- Що таке алгоритм?
- Пригадайте план створення комп'ютерної презентації.
- Наведіть план святкування вашого дня народження.

АЛГОРИТМИ В НАШОМУ ЖИТТІ

Алгоритми постійно присутні в нашому житті.

Кожного ранку, коли потрібно йти до школи, ви встаєте у визначений час (наприклад, о 7 годині), робите зарядку, вмиваєтеся, снідаєте, вдягаєте шкільну форму, берете шкільну сумку, яку склали ввечері, виходите з дому, йдете або їдете до школи.