

Інформатика 7 клас

Урок №14

Складання та виконання алгоритмів з повторенням і розгалуженням у середовищі Scratch



Розділ 3. Алгоритми з повторенням і розгалуженням

У цьому розділі ви дізнаєтеся про:

- цикли (повторення) в природі, у повсякденному житті, під час вивчення навчальних предметів
- «цикл з лічильником»
- алгоритми з повторенням
- висловлювання, істинні та хибні висловлювання
- розгалуження
- алгоритми з розгалуженням
- «цикл з передумовою»

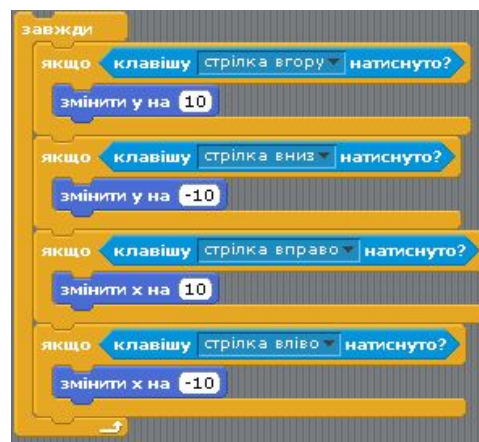


Сьогодні ви дізнаєтесь:

як створювати проекти,
що реалізують складні
алгоритми



як описати алгоритми
із вкладеними
розгалуженнями



як описати алгоритми
із вкладеними циклами





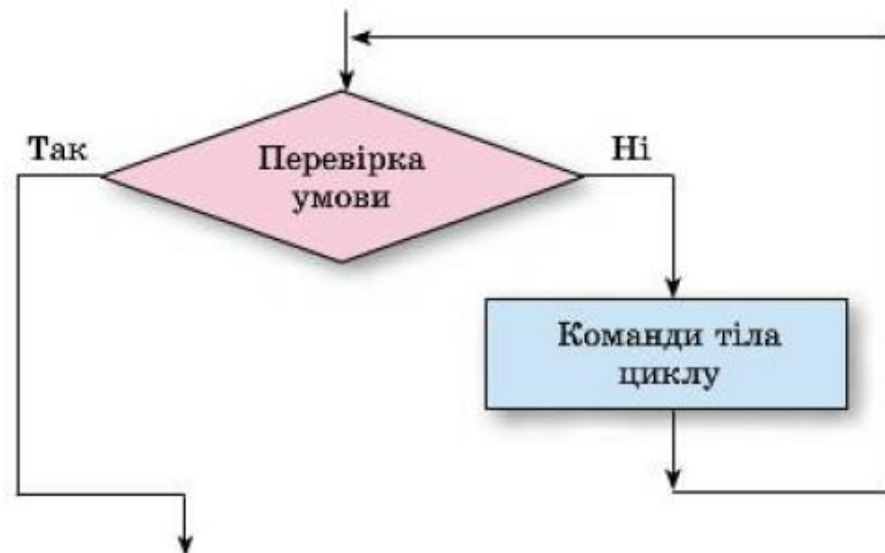
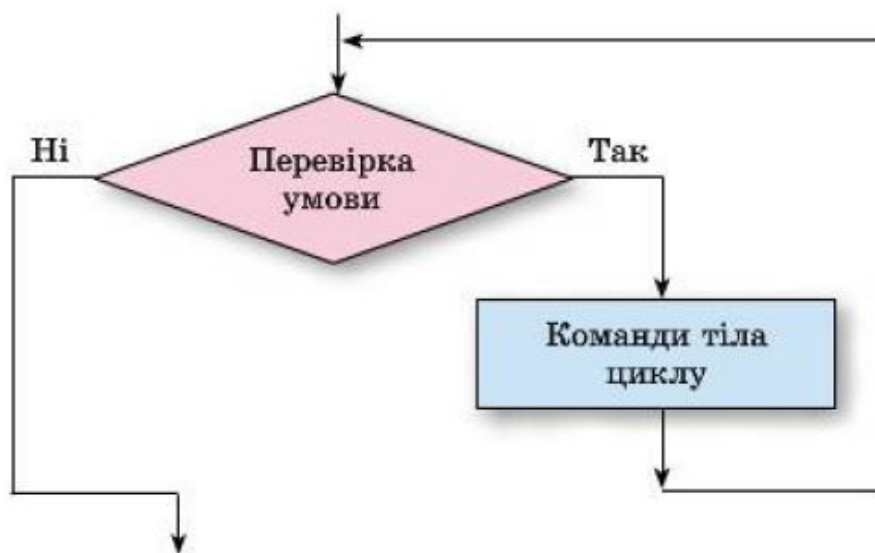
Запитання для повторення

1. Що таке цикл в алгоритмі?

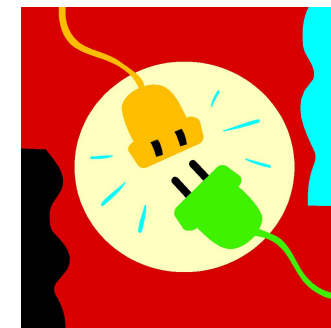
2. Як виконується стандартний цикл з передумовою?

3. Чим відрізняється виконання команди циклу з передумовою **повторювати поки в Scratch** порівняно зі стандартною командою циклу з передумовою?

4. Якими командами можна організувати цикл з передумовою в **Scratch**? Як виконуються ці команди?



Правила поведінки та безпеки в комп'ютерному класі



Як створювати проекти, що реалізують складні алгоритми?

У середовищі Scratch можна створювати складні проекти, наприклад, мультиплікації за мотивами відомих казок або власних історій, розробляти комп'ютерні ігри чи тренажери, моделювати процеси, що відбуваються у природі, або взаємодію частин механізмів.



Як створювати проекти, що реалізують складні алгоритми?

Перш ніж створювати проект у середовищі **Scratch** у тому числі і складний, слід:

1) визначити об'єкти проекту та їх зображення;

2) виділити події проекту та дібрати відповідні команди для їх реалізації;

3) задати метод розв'язування завдання, розділити завдання на підзадачі, у яких виділити базові алгоритмічні структури;

4) створити алгоритм і подати його словесно чи у вигляді схеми.



Як створювати проекти, що реалізують складні алгоритми?

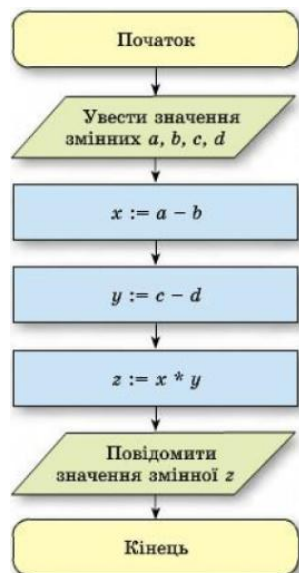
Далі алгоритм слід описати мовою середовища виконання алгоритмів, перевірити його правильність і зберегти разом з усіма об'єктами у файлі проекту.



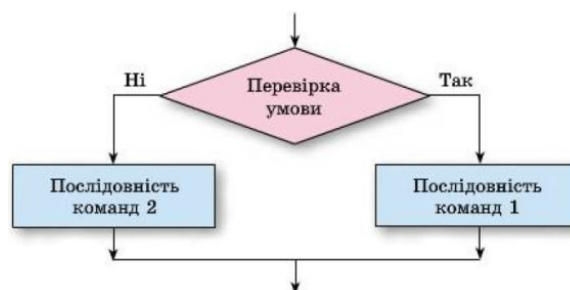
Як створювати проекти, що реалізують складні алгоритми?

В алгоритмах, які описують складні проекти, можуть використовуватися базові структури алгоритмів: слідування, розгалуження, повторення,

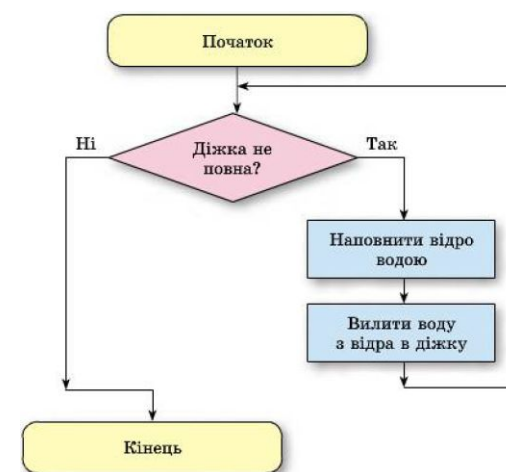
слідування



розгалуження



повторення



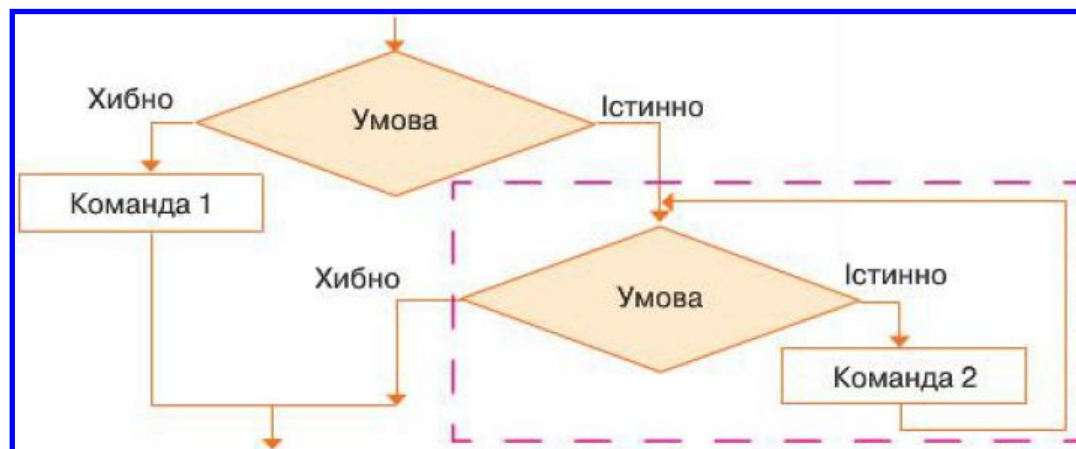
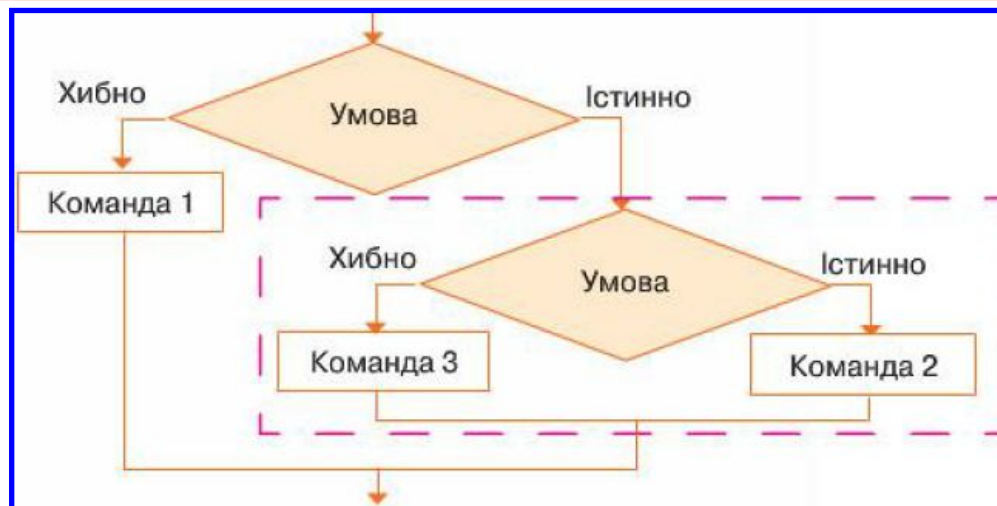
а тож різні їх комбінації.






Як описати алгоритми з вкладеними розгалуженнями?

Складні алгоритми часто поєднують декілька алгоритмічних структур, які можуть бути вкладеними одна в іншу.



Як описати алгоритми з вкладеними розгалуженнями?



Розглянемо приклад. Нехай напрямок руху виконавця **Кіт** , який за замовчуванням переміщується вперед, можна змінити за допомогою відповідних клавіш клавіатури. Зазначені дії можна реалізувати за допомогою таких фрагментів двох програм у середовищі **Scratch**.

```
завжди
якщо клявішу стрілка вгору натиснуто?
  змінити у на 10
якщо клявішу стрілка вниз натиснуто?
  змінити у на -10
якщо клявішу стрілка вправо натиснуто?
  змінити x на 10
якщо клявішу стрілка вліво натиснуто?
  змінити x на -10
```

Приклад 1


```
завжди
якщо клявішу стрілка вгору натиснуто?
  змінити у на 10
інакше
  якщо клявішу стрілка вниз натиснуто?
    змінити у на -10
  інакше
    якщо клявішу стрілка вліво натиснуто?
      змінити x на -10
    інакше
      змінити x на 10
```

Приклад 2



Як описати алгоритми з вкладеними розгалуженнями?



Приклад 1 демонструє послідовне використання чотирьох команд неповного розгалуження. Рух виконавця в певному напрямі буде здійснено тільки за умови натиснення відповідної клавіші  .

```
завжди
якщо клавiшу стрiлка вгору натиснуто?
  змiнити у на 10
якщо клавiшу стрiлка вниз натиснуто?
  змiнити у на -10
якщо клавiшу стрiлка вправо натиснуто?
  змiнити x на 10
якщо клавiшу стрiлка влiво натиснуто?
  змiнити x на -10
```

Приклад 1

```
якщо клавiшу стрiлка вгору натиснуто?
  змiнити у на 10
```

```
якщо клавiшу стрiлка вниз натиснуто?
  змiнити у на -10
```

```
якщо клавiшу стрiлка вправо натиснуто?
  змiнити x на 10
```

```
якщо клавiшу стрiлка влiво натиснуто?
  змiнити x на -10
```



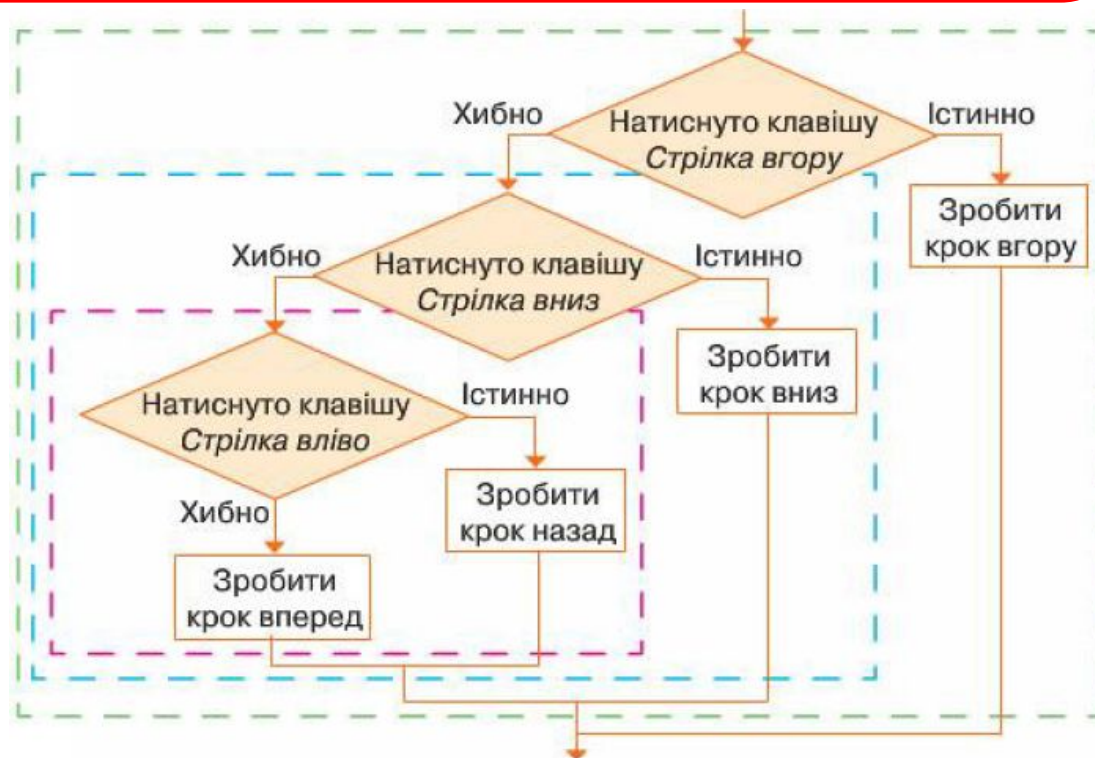


Як описати алгоритми з вкладеними розгалуженнями?

У **прикладі 2** використано три команди розгалуження повної форми, у яких тричі перевіряється, яка клавіша натиснута. **Рух** буде здійснюватись **без натиснення клавіші Стрілка вправо**. Алгоритм, застосований для складання фрагменту програми для прикладу 2, можна подати графічно.



Приклад 2

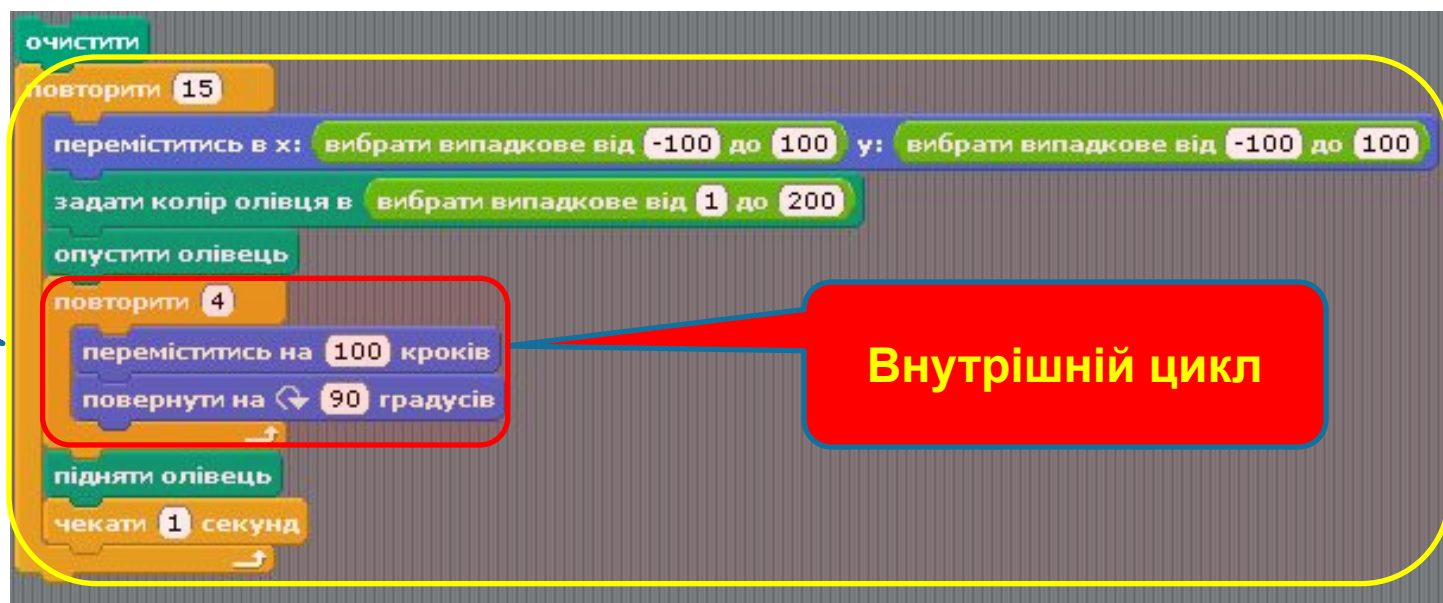




Як описувати алгоритми із вкладеними циклами?

При розв'язуванні задач часто необхідно деякі дії повторити кілька разів. Один із способів написання такої програми — включити повторення в набір команд, що повторюються всередині іншого циклу. Така структура, що складається з циклу в циклі, називається **вкладеними циклами**.

**Зовнішній
цикл**



Внутрішній цикл

У програмі, що містить вкладені цикли, обов'язково спочатку виконується **«внутрішній»** цикл, а потім — **«зовнішній»**.



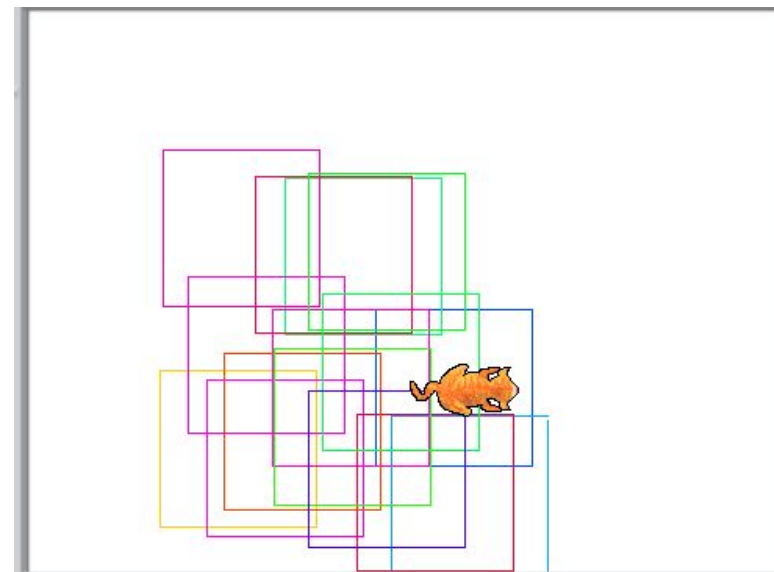
Як описувати алгоритми із вкладеними циклами?



Наприклад, щоб у проекті **Вітраж**, реалізованому в середовищі **Scratch**, отримати зображення з 15 різнокольорових квадратів, використовують два цикли.

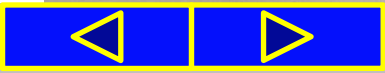
Зовнішній цикл: **15 разів** забезпечує побудову об'єкта різними кольорами

```
очистити
повторити 15
  переміститись в x: вибрати випадкове від -100 до 100 у: вибрати випадкове від -100 до 100
  задати колір олівця в вибрати випадкове від 1 до 200
  опустити олівець
  повторити 4
    переміститись на 100 кроків
    повернути на 90 градусів
  підняти олівець
  чекати 1 секунд
```



Внутрішній цикл: **Забезпечує** побудову сторін квадрата

Фізкультхвилинка





Працюємо з комп'ютером

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

Завдання. Створи проект, за яким виконавець **anjuli** складатиме і виконуватиме мелодію з випадкових п'яти нот, поки не зміняться всі його образи.

1. Завантаж середовище **Scratch**.
2. Визнач, скільки виконавців буде у твоєму проекті та якими вони будуть. Завантаж образ виконавця алгоритму з бібліотеки зображень середовища Scratch і додай ще 3-4 його образи, або зміни їх у графічному редакторі.
3. Передбач, що згідно з умовою задачі у проекті відбуватимуться такі події: звучатиме мелодія з п'яти нот, змінюватиметься образ виконавця.
4. Перевір свої міркування, на які підзадачі можна розділити отримане завдання:
 - а) запуск програми, метод — кнопка **Запуск**;
 - б) зміна образів виконавця, метод — цикл **Повторити K**, де значення *K* залежить від кількості образів виконавця;
 - в) звучання мелодії, метод — цикл **Повторити 5**;
 - г) зміна нот — випадкова величина, що задає величину номера ноти в нотному стані середовища **Scratch**.





Працюємо з комп'ютером

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

5. Склади алгоритм розв'язування завдання.
6. Із запропонованих команд склади алгоритм для реалізації завдання **Рис.1**.
7. Запусти програму на виконання. Переконайся, що завдання виконано правильно.
8. Збережи проект у своїй структурі папок з іменем **Композитор_Прізвище**.

Приклад алгоритму

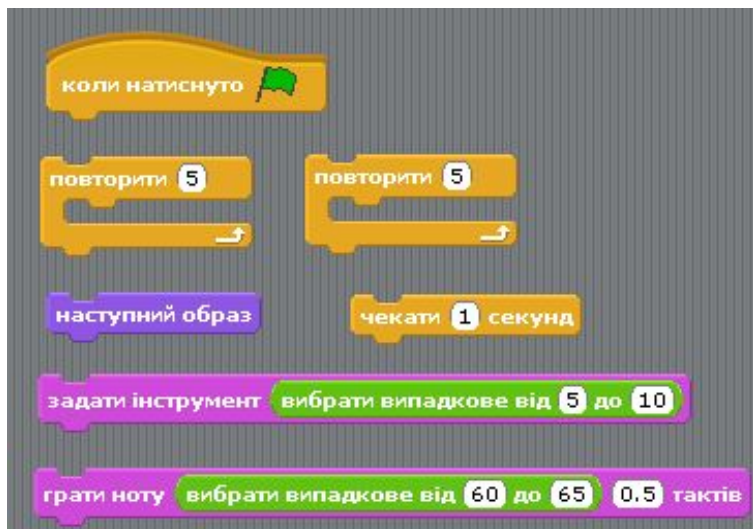


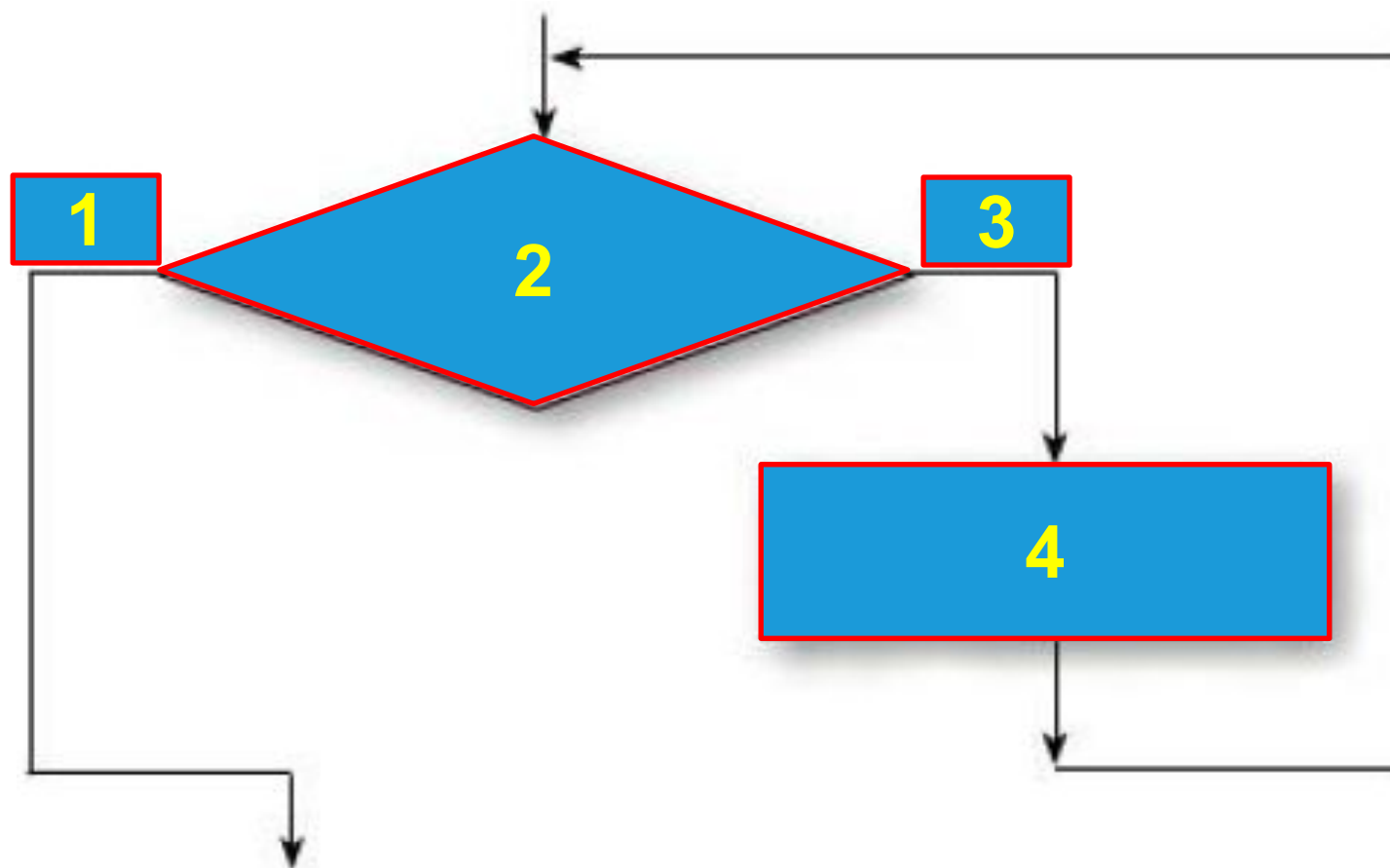
Рис.1



Актуалізація зошит завдання 1 с.40



1. Заповніть наведену блок-схему циклу з передумовою.

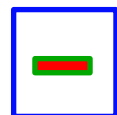


Робота із зошитом

завдання 1 (с.38)



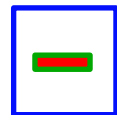
Позначте правильні продовження речення: **Команди тіла циклу з передумовою...**



1) завжди виконуються більше ніж один раз



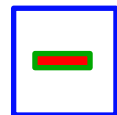
2) можуть виконуватися більше ніж один раз



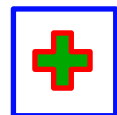
3) завжди виконуються тільки один раз



4) можуть виконуватися один раз



5) не виконуються жодного разу



6) можуть не виконуватися жодного разу



Підсумок

Виконання стандартного **циклу з передумовою** відбувається так: виконавець виконує **команду перевірки умови**; якщо результат виконання цієї команди **істина**, то виконавець виконує **команди тіла циклу**, після чого знову виконує **команду перевірки умови**; якщо ж результат виконання команди перевірки умови **хиба**, то виконання команд тіла циклу не відбувається і виконавець переходить до виконання першої команди наступного фрагмента алгоритму.

Команди тіла циклу з передумовою саме **«можуть бути виконані більше ніж один раз»**, а не **«обов'язково виконуються більше ніж один раз»**. Адже вони **«можуть виконатися тільки один раз»** або **«не виконуватимуться жодного разу»**.

Команда циклу з передумовою **повторювати поки** в **Scratch** виконується так: виконавець виконує **команду перевірки умови**; якщо результат виконання цієї команди **хиба**, то виконавець виконує **команди тіла циклу**, після чого знову виконує **команду перевірки умови**; якщо ж результат виконання команди перевірки умови **істина**, то виконання команд тіла циклу не відбувається і виконавець переходить до виконання першої команди наступного фрагмента алгоритму.

Підсумок



1. Що таке цикл в алгоритмі?
2. Як виконується стандартний цикл з передумовою?
3. Чи можуть команди тіла циклу не виконуватися жодного разу?
Поясніть свою відповідь, проілюструйте пояснення прикладами.
4. Чи може виконання циклу ніколи не закінчитися? Поясніть свою відповідь, проілюструйте пояснення прикладами.
5. Чим відрізняється виконання команди циклу з передумовою **повторювати поки** в **Scratch** порівняно зі стандартною командою циклу з передумовою?
6. Якими командами можна організувати цикл з передумовою в **Scratch**? Як виконуються ці команди?



Вправи для очей



Наші очі трішки втомились і ми зараз відпочинемо. Виконуємо вправи за командою:

- 1. Швидко поморгати, закрити очі і посидіти спокійно, повільно рахуючи до 5. Повторити 4-5 разів.**
- 2. Повільно поводити очима зліва направо і справа наліво, вгору-вниз і навпаки по 3 рази.**

Розгадай ребус



Відповідь: **ПІСЛЯУМОВА**



Домашнє завдання:

Опрацювати:



Підручник Повторити § 3.4. (с. 86 – 92);

Виконати у підручнику
завд. 4 (с. 91)



Виконати у зошиті
завд. 3, 4 (с. 40 - 41)



Спасибі за увагу !

