

## Лабораторна робота № 2

### ГРАФІЧНА МОВА ПРОГРАМУВАННЯ SCRATCH

#### Мета роботи:

Вивчити поняття: функцій, умов, булевих виразів, циклів.  
систем числення та навчитися переводити числа з однієї системи в іншу.

#### Обладнання та програмне забезпечення:

- IBM сумісна персональна обчислювальна машина;
- графічна мова програмування [Scratch](https://scratch.mit.edu/), доступна за посиланням <https://scratch.mit.edu/>.

#### Завдання до роботи:

Написати програму для ПЕОМ з використанням графічної мови Scratch, котра реалізує наступні елементи:

- функції;
- умови;
- логічні вирази;
- цикли.

### Теоретичні відомості Scratch

Scratch – це середовище та інтерпретована динамічна візуальна мова програмування, у якій код створюється шляхом маніпулювання графічними блоками. Середовище орієнтовано на початкове знайомство з основними концепціями та ідеями програмування. Завдяки динамічності, вона дає змогу змінювати код навіть під час виконання. Графічна мова програмування Scratch розроблена фахівцями Массачусетського технологічного інституту. Скретч отримав свою назву від техніки «скретчингу», яку використовують диск-жокеї, «міксуючи» музику для отримання різних звукових ефектів, Скретч дозволяє користувачам творчо «змішувати» різні медіа (включаючи графіку, звук та інші програми) під час створення проєктів. Логотип мови — рудий кіт Скретч. Онлайн версія програмного середовища доступна за адресою <https://scratch.mit.edu/>.

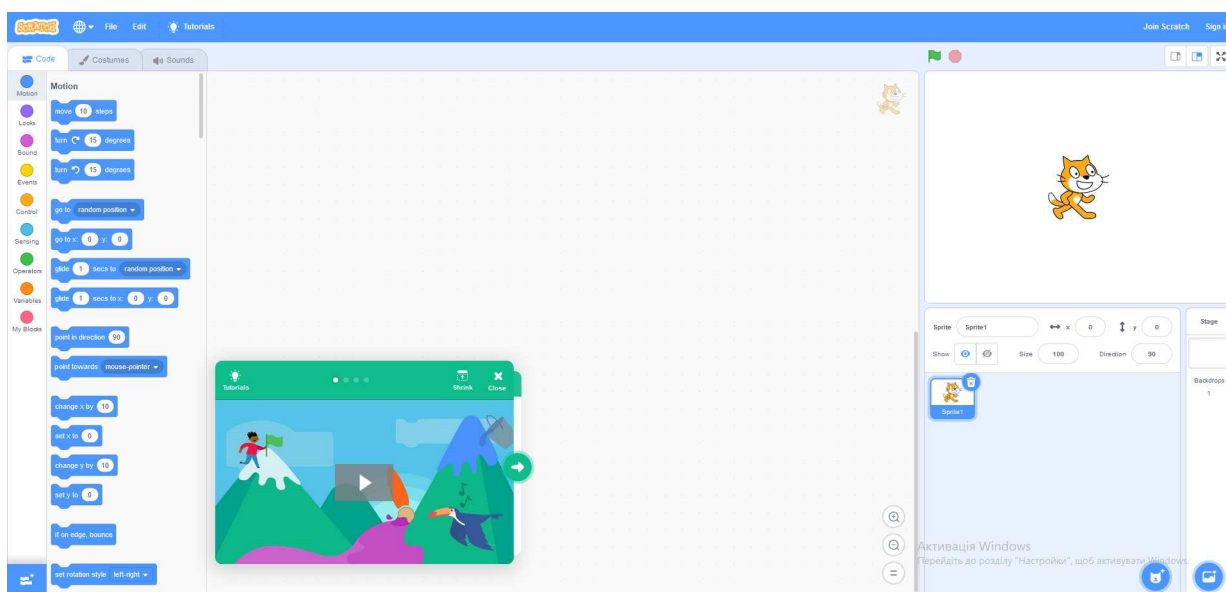


Рис. 1. Початковий вигляд вікна графічної мови програмування Scratch.

Створимо першу програму, котра видає повідомлення “Hello word”.

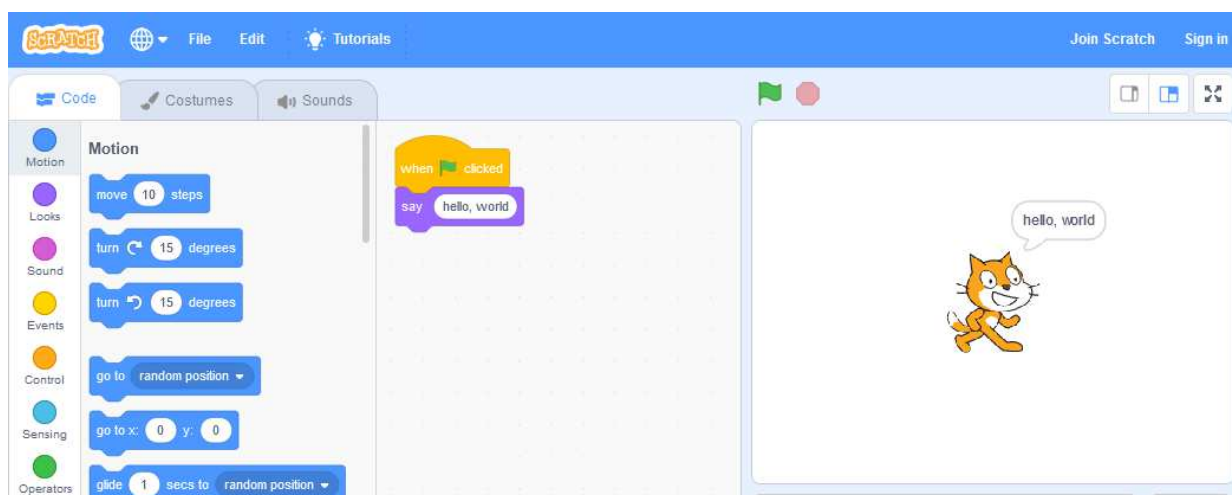


Рис. 2. Програма “Hello world” створена в графічному середовищі програмування Scratch.

**Функція** — частина програми, яка реалізує певний алгоритм і дозволяє звернення до неї з різних частин загальної (головної) програми.

У прикладі на рис.2. функцією є блок фіолетового кольору котрий реалізовує промовляння фрази “hello, world”.

Висловлювання, яке може бути істинним (правильним) або хибним (неправильним) називається **умовою**.

**Проста умова** — це висловлювання, в якому два вирази з’єднані знаком операції відношення.

**Складена умова** — це висловлювання, в якому дві або більше простих умов з’єднані знаками логічних операцій.

**Висловлювання** — це деяке твердження, відносно якого можна сказати, що воно або істинне, або хибне. Наприклад: “ $5 > 2$ ” – “істинно”, а “ $2 + 2 = 5$ ” – “хибно”.

**Логічні вирази** — це вирази, що складаються з висловлювань, які можуть бути з’єднані логічними зв’язками. Ці вирази набувають логічного значення («хибне» або «істинне»). Логічні вирази можуть бути простими та складеними.

Таким чином, кожному висловлюванню можна приписати «0» (хибне) або «1» (істинне). Приклад: «5 — просте число» — істинне, « $2 = 3 + 5$ » — хибне висловлювання.



Рис. 3. Приклад програми котра реалізує рух до границь екрану.

Приклад на рис.3. використовує умову:



Ще один приклад програми котра використовує умову:



Рис. 4. Використання умови у програмі.

**Цикл** — різновид керівної конструкції, призначена для організації багаторазового виконання набору інструкцій (команд).



Рис. 5. Приклад використання циклу.

Послідовність інструкцій, призначена для багаторазового виконання, називається **тілом циклу**. Одноразове виконання тіла циклу називається **ітерацією**. Вираз, що визначає чи буде вчергове виконуватися ітерація, чи цикл завершиться, називається **умовою виходу** або умовою завершення циклу (або умовою продовження в залежності від того, як інтерпретується його істинність — як ознака необхідності завершення чи продовження циклу). Змінна, в якій зберігається номер поточної ітерації, називається **лічильником ітерацій** циклу або просто **лічильником циклу**. Цикл не обов'язково містить лічильник, також лічильник не зобов'язаний бути одним — умова виходу із циклу може залежати від декількох змінюваних в циклі змінних, а може визначатися зовнішніми умовами (наприклад, настанням певного часу), в останньому випадку лічильник взагалі не знадобиться.

### Різновиди циклів

**Безумовні цикли.** Іноді в програмах використовуються цикли, вихід з яких не передбачено логікою програми. Такі цикли називаються **безумовними** або **нескінченними**.

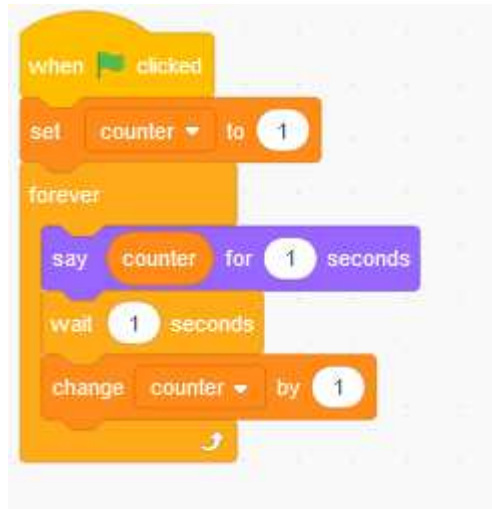


Рис. 6. Безумовний цикл.

**Цикл з передумовою** — цикл, що виконується доки істинна деяка умова, вказана перед його початком. Ця умова перевіряється до початку виконання тіла циклу, тому тіло може бути не виконане жодного разу (якщо умова з початку хибна).

**Цикл з післяумовою** — цикл, в якому умова перевіряється після виконання тіла циклу. Звідси випливає, що тіло циклу завжди виконується хоча б один раз.

**Цикл з виходом з середини** — найзагальніший тип умовного циклу. Синтаксично такий цикл оформляється за допомогою трьох інструкцій: початок циклу, кінець циклу та інструкції (команди) виходу з циклу. Інструкція початку позначає точку програми, з якої починається тіло циклу, інструкція кінця — точку, де тіло закінчується. Всередині тіла має бути присутня команда виходу з циклу, при виконанні якої цикл завершується і керування передається на оператор, наступний після інструкції кінця циклу. Природно, щоб цикл виконався більш ніж один раз, команда виходу має викликатися не безумовно, а тільки при виконанні умови виходу.

**Цикл з лічильником** — цикл, в якому деяка змінна змінює своє значення від заданого початкового значення до кінцевого значення з деяким кроком, і для кожного значення цієї змінної тіло циклу виконується один раз.

## ПОРЯДОК ВИКОНАННЯ РОБОТИ:

1. Опрацювати і засвоїти матеріал наведений в теоретичних відомостях.
2. Розглянути набір готових графічних програм, котрі додаються до роботи
3. Самостійно створити програми котрі реалізують: функції, умови та цикли.