

## Тема. Підпрограми. Практична робота №3

### Очікувані результати заняття

Після цього заняття потрібно вміти складати проекти з підпрограмами.

### Повторюємо

- Які типи даних опрацьовуються в Python?
- Як позначаються цілі числа, дробові числа?
- Як виглядає блок-схема алгоритму з розгалуженням?
- Які види розгалужень вам відомі? Як записати кожний вид мовою Python?

### Виконайте вправи

- <https://wordwall.net/uk/resource/64286326>
- <https://wordwall.net/resource/72823015>

### Ознайомтеся з інформацією

**Підпрограма** – це логічно незалежна спеціальним чином оформлена частина програми для розв'язування певної задачі. До підпрограм можна багаторазово звертатися з інших частин програми. Таке звернення називають **викликом підпрограм**.

У Python усі підпрограми називають **функціями**. Кожна функція це «державка в державі», мініпрограма в програмі. В ній діють всі закони, що притаманні будь-якій програмі. Функція являє собою набір команд, як і сама програма, але вона розв'язує менш складне завдання.

Іноколи функції групуються у **модулі**. Ми вже розглядали модуль підключення графічних можливостей **turtle**. В модулях зберігається опис його функції, які можна підключати і використовувати в будь-якій частині проекту або навіть в інших проектах. Існують вбудовані функції і **функції, створені користувачем**. **Вбудовані функції** є частиною мови і можуть викликатися за іменем без попереднього оголошення. Наприклад, **forward(30)** це ні що інше, як виклик вбудованої функції, яка прописана в стандартному модулі **turtle** і промальовує відрізок довжиною у 30 пікселів у напрямку руху.

Будь-яка функція має такий загальний вигляд:

```
def <назва> ([параметри]):  
    <блок команд>  
    [return <результат>]
```

**Параметр** (також називають аргументом) – це змінна, яка отримує конкретне значення під час звернення до функції. Параметри вказувати не обов'язково, але при цьому круглі дужки опускати не можна.

Інструкція **return** повертає результат виконання функції в основну програму.



## Поміркуйте

Розгляньте приклад коду з підпрограмою та поясніть результат і доцільність використання функції:

```
def заощадження(кишенькові, зароблені, витрачені):  
    return кишенькові + зароблені - витрачені  
print(заощадження(50, 100, 20))
```

## Додаткова інформація

[Магія двох бітів](#)

## Практична робота



Елементи [сніжинки](#) називаються фракталами.

**Фрактал** — нерегулярна, самоподібна структура. В широкому розумінні фрактал означає фігуру, малі частини якої в довільному збільшенні є подібними до неї самої.

Як бачимо на малюнку, **сніжинка Коха** — це рівносторонній трикутник. Його можна вважати сніжинкою Коха нульового порядку. Сніжинка більш високого порядку народжується з попередньої більш низького порядку шляхом заміни кожного ребра на чотири, з'єднаних ніби складені долоньки ( $\_/\_$ ). Тобто сторона замінюється на чотири долоньки втричі менші, а кожна сторона долоньки також замінюється 4-ма долоньками втричі меншими і т.д.

## Завдання 1 (комп'ютер)

Побудуємо сніжинку Коха. Тут кожна насупна сторона в 3 рази менша за попередню. А також є два кути повороту: 60 та 120 градусів.

Цю програму можна реалізувати різними способами, в прикладі розглянемо один з них. Змінна *size* — це довжина сторони, а змінна *n* — це деякий параметр, що показує рівень фігури.

```

from turtle import *
t=Pen()
size=300
pensize(2)
n=2
pencolor("red")
def koh(size,n):
    if n==0:
        forward(size)
    else:
        koh(size/3,n-1)
        left(60)
        koh(size/3,n-1)
        right(120)
        koh(size/3,n-1)
        left(60)
        koh(size/3,n-1)

def fraktal(size,n):
    for i in range(3):
        koh(size,n)
        right(120)
fraktal(size,n)

```

- Інсталюйте на комп'ютер [одне з середовищ програмування](#).
- Наберіть та реалізуйте поданий код.
- Зробіть фото екрану з результатом виконання програми та надішліть на HUMAN або на електронну пошту [nataliartemiuk.55@gmail.com](mailto:nataliartemiuk.55@gmail.com)

## Завдання 2 (смартфон)

- Перейдіть на сайт із середовищем програмування, наприклад <https://www.online-python.com/>
- Введіть та виконайте код до [задачі](#) одним із поданих способів.
- Зробіть фото екрану з результатом виконання програми та надішліть на HUMAN або на електронну пошту [nataliartemiuk.55@gmail.com](mailto:nataliartemiuk.55@gmail.com)

## Домашнє завдання

Опрацювати підручник с.90-94

## Джерела

- [Магія двох бітів](#)
- [Python – просто!](#)
- Інформатика : підруч. для 7 кл. загальноосвіт. навч. закл. / Й.Я.Ривкінд [та ін.]. — Київ: Генеза, 2024.