

Тема. Оператор вибору. Оператор повторення.

Після цього заняття потрібно вміти:

-Розв'язувати задачі з використанням усіх базових алгоритмічних структур, змінних та констант.

Ознайомтеся з інформацією

Умовний оператор if...elif...else

Перевірку умови в гілці else можна записати скорочено, скориставшись ключовим словом **elif** (англ. *else if* — *інакше, якщо*).

Синтаксис оператора:

```
if <умова1>: <оператори 1>
elif <умова2>: <оператори 2>
...
elif <умова N>: <оператори N>
else: <оператори>
```

Інструкція if-elif-else припиняє перегляд наступних гілок, як тільки логічний вираз у поточній гілці буде True. Так, якщо вираз при if (перша гілка) буде True, то після виконання блоку <оператори 1> керування передається наступному за if-elif-else оператору. Якщо жодна з умов по гілках if-elif не є істинною, виконується блок операторів по гілці else.

Зверни увагу!

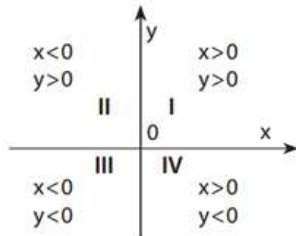
У гілці elif обов'язково повинен бути логічний вираз — умова, як у заголовку if . У кінці після всіх гілок elif може бути одна гілка else для обробки випадків, які не відповідають умовам гілки if і всіх elif .

Блок-схема множинного розгалуження	Приклад
<pre>graph TD U1{Умова 1} -- Так --> D1[Дії 1] U1 -- Ні --> U2{Умова 2} U2 -- Так --> D2[Дії 2] U2 -- Ні --> U3{Умова 3} U3 -- Так --> D3[Дії 3] U3 -- Ні --> D4[Дії 4] D1 --> Exit(()) D2 --> Exit D3 --> Exit D4 --> Exit</pre>	<pre>W = input('Введіть назву пори року: ') if W == 'весна': print('березень, квітень, травень') elif W == 'літо': print('червень, липень, серпень') elif W == 'осінь': print('вересень, жовтень, листопад') elif W == 'зима': print('грудень, січень, лютий') else: print('неправильні дані')</pre>

Приклад:

Визначити, у якій координатній чверті лежить точка з координатами x, y ($x \neq 0, y \neq 0$).

```
x = int(input('x = ?'))
y = int(input('y = ?'))
if x>0 and y>0: print('I чверть')
elif x<0 and y>0: print('II чверть')
elif x<0 and y<0: print('III чверть')
elif x>0 and y<0: print('IV чверть')
else: print('Точка лежить на осі')
```



Цикл із параметром

Цикл



Ви вже вивчали алгоритмічну структуру повторення в середовищі програмування Scratch.



Повторення (цикл) - це алгоритмічна структура, за допомогою якої та сама послідовність дій може виконатися кілька разів.

В мові програмування Python існує **два цикли**:

-**цикл із параметром** (кількість повторень тіла циклу заздалегідь відома);

-**цикл з умовою** (кількість повторень тіла циклу заздалегідь невідома).

Цикл із параметром (цикл for) використовують, якщо кількість повторень визначає певна послідовність значень (символи рядка, елементи списку тощо). Блок-схему циклу подано на рис.1.

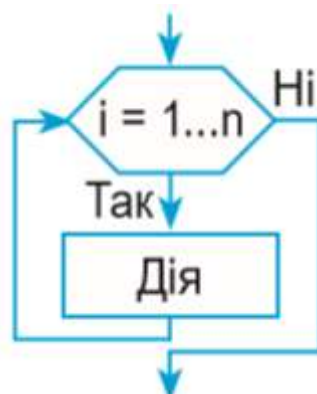


Рис.1

Цикл із параметром (цикл for)



Дії повторюються визначену кількість разів.

Синтаксис оператора циклу з параметром:

for i in range(n):

<тіло циклу>

Увага відступ!

i - параметр, що змінює своє значення з початкового (за замовчуванням дорівнює 0) до кінцевого **n**.

n - кінцеве значення (**число n не може бути дійсним, а лише цілим**)

in - наказує по чергово надавати змінній **i** усі цілі значення в діапазоні від **0 по n-1** (включно)

Функція range



Функція **range(n)** (перекладається як діапазон) з одним аргументом генерує діапазон чисел від **0 по n-1** (включно).

Приклад:

```
Options window help
for i in range(4):
    print(i)
```

Результат

```
=====
0
1
2
3
>>> |
```

Функція **range(n,m)** з двома аргументами дозволяє генерувати діапазон чисел від **n по m-1** (включно).

Приклад:

```
for i in range(1,6):
    print(i)
```

Результат

```
=====
1
2
3
4
5
>>> |
```

Перегляньте відео за посиланням:

<https://youtu.be/P7FAAsUyo4w>

Завдання

1. Перевірте, як виконуються програми-розв'язки запропонованих задач (використовуючи онлайн компілятор repl.it)

Задача 1. Рахуємо гроші. До введеного числа додати напис "гривень", "гривня", "гривні" згідно з правилами української мови

```

x=int(input('Введіть суму '))
a=x%10
b=x//10%10
if(a==1) and (b!=1):
    print (x, 'гривня')
elif ((a==2)or(a==3)or(a==4))and(b!=1):
    print (x, 'гривні')
else:
    print(x, 'гривень')

```

Задача 2. Дано список днів тижня. З'ясуємо, скільки з них припадають на вихідні (сб, нд).

```

D = ['пн', 'нд', 'чт', 'сб', 'пт', 'сб', 'вт', 'ср', 'нд', 'чт', 'чт', 'нд', 'вт', 'пн', 'ср', 'нд', 'сб']
K = 0
for day in D: # Перебір у змінній day елементів списку D
    if day in ['сб', 'нд']: K += 1
print(K)

```

Задача 3. На вході маємо числа (a і b). Задачею програми буде до числа a додавати одиницю b разів, а потім вивести кінцевий результат

```

a=int(input("введіть ціле число a:"))
b=int(input("введіть ціле число b:"))
for i in range(b):
    a=a+1
print(a)

```

2. Спробуйте написати та реалізувати код програми до задачі 4. Успіхів!

Задача 4. Програма має порахувати суму n чисел, перше з яких a, а кожне наступне на 3 більше від попереднього. (n і a – запитуються у користувача).

Виконані завдання надішліть вчителю на HUMAN або на електронну пошту balag.elizaveta@gmail.com