

Лабораторна робота №3

Тема: Програмування умовних конструкцій

Мета роботи: Навчитись складати блок-схеми та програми до розгалужених типів алгоритмів на мові Python та виконувати їх на ПК.

1. Теоретичні відомості.

Множинне розгалуження

Множинне розгалуження – це спеціальна конструкція, в якій, в залежності від умови, виконується одна гілка, а інші пропускаються.

Заголовком називається строка, у якій відбувається перевірка умови (Наприклад: `if...:/elif...:/else:`)

Тіло оператора – це гілка, по якій виконується програма у разі, якщо перевірка умови поверне значення “True”.

Конструкція завжди починається з оператора *if*. При виконанні відбувається перевірка умови, яка записана після оператора.

`if a >= 0:` `if a == 1:` `if a < 0:`

Варто пам'ятати, що “=” – це оператор присвоєння, а “==” – порівняння.

У випадку, якщо оператор *if* повернув значення “True”, то виконується гілка, яка записана в тілі цього оператора. Тіло записується з 4 пробілами (одне натиснення `tab`) у кожній строці:

```
if a >= 0:
    a = a+3
    print(a)
elif a < -3:
    a += 1
    print(a)
else:
    print(a)
```

Для скорочення також можна замінити вираз `a = a+3` на `a += 3`

Якщо оператор *if* повернув значення “False”, то інтерпретатор пропустить тіло цього оператора і перейде до перевірки наступної умови, записаної після оператора *elif*.

Залишається останній оператор *else*. Він виконується тільки в тому випадку, якщо оператор *if* та всі оператори *elif* повернули “False”.

Існує кілька способів запису розгалудження:

- 1) Нормальна (повна) форма;
- 2) Коротка форма запису (*тернарний вираз*);
- 3) If-elif-else (еквівалент `switch case`).

Імпорт бібліотек (модулів), модуль MATH

Імпортувати модуль означає додати до основних функцій набір додаткових, які містить цей модуль. Це можна зробити чотирма способами:

1) Імпорт всієї бібліотеки.

При цьому визов функції потребує вказання назви модуля:

```
b = math.sin(a)
```

2) Імпорт всієї бібліотеки під іншою назвою.

Визов функції потребує вказання нової назви.

3) Імпорт окремої функції з бібліотеки:

Визов функції не потребує вказання назви модулю, але, якщо існувала функція з такою самою назвою, то вона перезаписується на нову.

4) Імпорт всіх функцій бібліотеки:

Імпортуються всі функції, при визові не треба вказувати назву модуля, але недолік той самий, як і при імпортуванні окремих функцій.

У модулі *math* є такі функції:

`ceil()` – округлення в більшу сторону,

`floor()` – округлення в меншу сторону,

`sin()` – синус радіану,

`cos()` – косінус радіану,

`tan()` – тангенс радіану,

`radians()` – перевод з градусів в радіани.

Крім того, у класу є два корисні поля: число π і число e .

2. Завдання

Написати програму, яка обчислює вхідні дані за формулою, відповідною за Вашим варіантом. Результат округлити до більшого значення та вивести на екран. Перевірити правильність виконання програми.

3. Приклад виконання

$$y = \begin{cases} -2.8x - 1 & -1 < x \\ 1 - |x| & |x| < 1 \\ 0 & x > 1 \end{cases}$$

Загальний вид програми:

```
from math import *

x = float(input('Введіть x '))
if x>1:
    print('y = 0')
elif abs(x)<1:
    print('y =', ceil(1-abs(x)))
else:
    print('y =', ceil((-2.8*x)-1))
input()
```

4. Хід виконання роботи

1. Скласти блок-схему алгоритму.
2. Скласти програму на мові Python, увести до ПК та виконати розрахунки.
3. Скласти звіт, який повинен містити:
 - найменування та мета роботи;
 - завдання;
 - текст програми;
 - результати розрахунку;
 - стислі відповіді на контрольні запитання.

5. Контрольні питання

1. Що виведеться в результаті виконання цієї програми?

```
from math import *
a = 82.6
if a == 82:
    print('1')
elif a > 82:
    print('2')
elif abs(a) < 81:
    print('3')
else:
    print('4')
```

2. Перечисліть способи імпорту модулів. Назвіть головний недолік імпорту окремих функцій з модуля.

6. Варіанти завдання

Номер варіанту	Формула	Умова
1	$y = \begin{cases} -0.5x^2 \ln(x) \\ 1 \\ e^{-0.5x} \cos 2x \end{cases}$	$\begin{aligned} 1 \leq x \leq 2 \\ x < 1 \\ x > 2 \end{aligned}$
2	$y = \begin{cases} x^2 - \frac{7}{x} \\ 1.5x^2 + \sqrt{x^2 + 1} \\ \log(x + 7\sqrt{x}) \end{cases}$	$\begin{aligned} x \leq 12 \\ 12 < x < 14 \\ x \geq 14 \end{aligned}$
3	$y = \begin{cases} \sin(x) * \ln(x) \\ \cos^2 x \end{cases}$	$\begin{aligned} 35 < x \\ x \leq 35 \end{aligned}$
4	$y = \begin{cases} 1.5x - \lg(1.5x) \\ 1 \\ 1.5x + \lg(1.5x) \end{cases}$	$\begin{aligned} 1.5x < 1 \\ 1 \leq 1.5x \leq 8 \\ 1.5x > 8 \end{aligned}$
5	$y = \begin{cases} \frac{2.7}{x} + 175x^2 - 0.89 \\ x \\ 2.75x + 175x^2 \end{cases}$	$\begin{aligned} x < 6 \\ 6 \leq x \leq 12 \\ x < 12 \end{aligned}$
6	$y = \begin{cases} 0.3 \sin\left(\frac{x^2 + 1}{10}\right) \\ \cos(x + 0.1) \end{cases}$	$\begin{aligned} \sin\left(\frac{x^2 + 1}{10}\right) > 0 \\ \sin\left(\frac{x^2 + 1}{10}\right) \leq 0 \end{aligned}$
7	$y = \begin{cases} -2.8x - 1 \\ 1 - x \\ 0 \end{cases}$	$\begin{aligned} x < 1 \\ 1 \leq x < 3 \\ x \geq 3 \end{aligned}$
8	$y = \begin{cases} 15 \cos x^2 \\ 4.14x + \sin(x^2 - 15) \\ (x - 2)^2 + 75 \end{cases}$	$\begin{aligned} x \leq 1 \\ 1 < x \leq 5 \\ x > 5 \end{aligned}$
9	$y = \begin{cases} \sin(x) + \sqrt[3]{ x } \\ 14 \cos(x) + 3x^2 \end{cases}$	$\begin{aligned} 1 < x \\ 1 \geq x \end{aligned}$
10	$y = \begin{cases} \lg(x + 3) \\ 1 + \sin^2 \sqrt{ 20.3x } \end{cases}$	$\begin{aligned} x < 3 \\ x \geq 3 \end{aligned}$