

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення
Дисципліна: Скриптові мови програмування (Python)

Лабораторна робота №1
Тема: «АРИФМЕТИЧНІ ВИРАЗИ, УПРАВЛЯЮЧІ КОНСТРУКЦІЇ
ТА МАСИВИ У PYTHON»

Виконав: ст. гр. КН-24
Куріщенко П. В.
Перевірив: ассистент
Ткаченко О.С.

Кропивницький 2024

Варіант - 16

Мета роботи - навчитися створювати найпростіші програми на Python, використовуючи оператори вибору і циклів, арифметичні вирази та масиви.

ЗАВДАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ

$$1) z = \sqrt{\frac{m+3}{m-3}}$$

Число m вводиться користувачем у консолі Python.

2) Дано натуральне число n , обчислити $y = 2 \cdot 4 \cdot 6 \dots (2 \cdot n)$.

3) Дано одномірний масив, що складається з N цілочисельних елементів.

- Знайти максимальний від'ємний елемент.
- Обчислити середнє арифметичне парних елементів масиву.
- Вивести ненульові елементи на екран у зворотному порядку.

Завдання 1

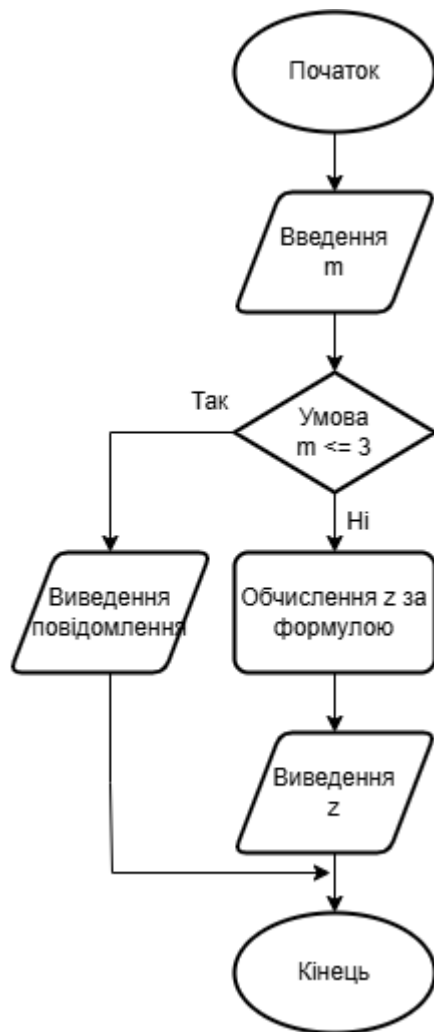
Опис програми та блок-схема:

1. Зчитування вхідних даних.
2. Перевірка умови:

А) Якщо $m \leq 3$, то програма виводить повідомлення і закінчує роботу.

Б) Інакше виконується наступний крок.

3. Обчислення виразу за формулою.
4. Виведення результату.



Лістинг програми:

```

1  import math
2
3  m = float(input("Enter a whole number > 3: "))
4
5  if m <= 3:
6      print("The number you entered is not greater than 3.")
7  else:
8      z = float(math.sqrt((m + 3) / (m - 3)))
9      print("The result 'z' is: ", z)

```

Результат виконання:

Запуск №1

```

Enter a whole number > 3: 4
The result 'z' is:  2.6457513110645907

```

Запуск №2

```
Enter a whole number > 3: 3
The number you entered is not greater than 3.
```

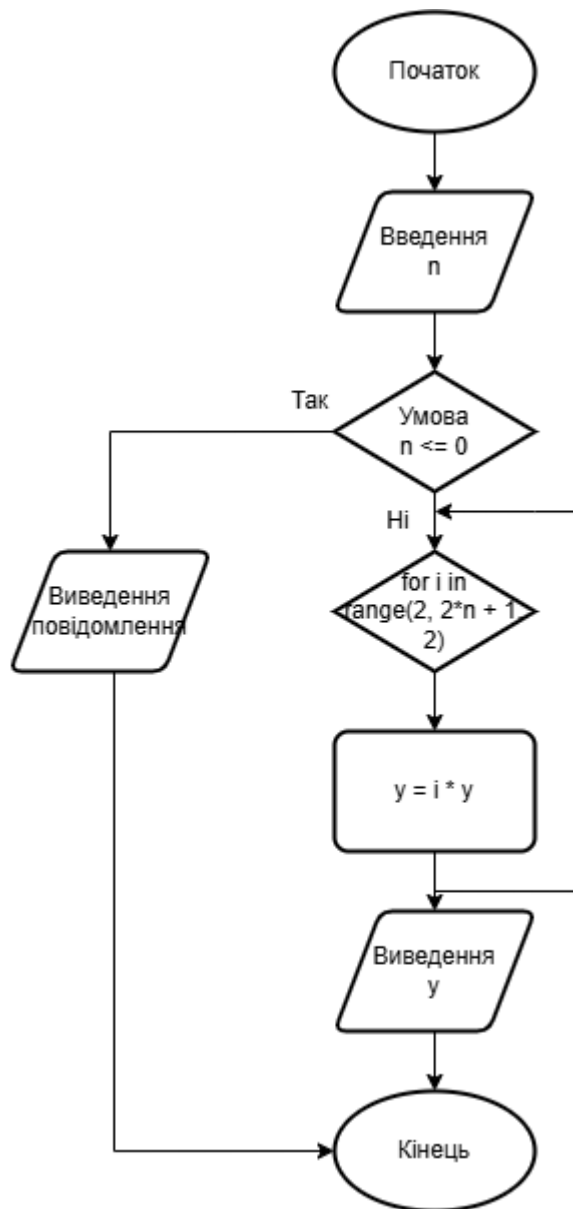
Запуск №3

```
Enter a whole number > 3: 11.56
The result 'z' is: 1.3041988266515394
```

Завдання 2

Опис програми та блок-схема:

1. Зчитування вхідних даних.
2. Перевірка умови:
 - А) Якщо $n \leq 0$, то програма виводить повідомлення і закінчує роботу.
 - Б) Інакше виконується наступний крок.
3. Обчислення добутку парних чисел.
 - 1) Використовується цикл `for`, який перебирає **парні числа** від 2 до $2*n$ включно (`range(2, 2*n + 1, 2)`).
 - 2) На кожній ітерації y (результат) множиться на поточне значення i .
4. Виведення результату.



Лістинг програми:

```

1  n = int(input("Enter an integer > 0: "))
2  y = int(1)
3
4  if n <= 0:
5      print("The number you entered is not greater than 0.")
6  else:
7      for i in range(2, 2*n + 1, 2):
8          y = int(i * y)
9      print("The result 'y' is: ", y)

```

Результат виконання:

Запуск №1

```
Enter an integer > 0: 3
The result 'y' is: 48
```

Запуск №2

```
Enter an integer > 0: 0
The number you entered is not greater than 0.
```

Запуск №3

```
Enter an integer > 0: 2.4
Traceback (most recent call last):
  File "c:\SLP-Kurishchenko-CS-24\Lab1\Tasks\Task2.py", line 1, in <module>
    n = int(input("Enter an integer > 0: "))
ValueError: invalid literal for int() with base 10: '2.4'
```

Завдання 3

Опис програми та блок-схема:

Ця програма виконує кілька операцій з масивом випадкових чисел. Вона складається з п'яти функцій: `create_array()`, `first_task()`, `second_task()`, `third_task()` і `main()`

1. Функція `create_array(n)` – створення масиву

- 1) Перевіряє, чи $n > 0$. Якщо ні – виводить повідомлення та завершує виконання.
- 2) Створює список `array` із n випадкових чисел у діапазоні від -100 до 100.
- 3) Виводить створений масив і повертає його.

2. Функція `first_task(array)` – пошук максимального від'ємного числа

- 1) Перебирає всі елементи масиву.
- 2) Шукає найбільше (найближче до нуля) від'ємне число.
- 3) Виводить знайдене значення або повідомлення, що від'ємних чисел немає.

3. Функція `second_task(array)` – середнє арифметичне парних елементів

1) Обчислює середнє арифметичне елементів, що стоять на парних позиціях (0, 2, 4, ...).

2) Виводить кількість таких елементів(для зручності) та їх середнє значення.

4. Функція `third_task(array)` – реверс усіх елементів, окрім першого

1) Змінює порядок усіх елементів у списку, крім першого (`array[0]`).

2) Виводить оновлений масив.

5. Функція `main()` – запуск програми

1) Зчитує `n` – кількість чисел у масиві.

2) Викликає `create_array(n)`, щоб створити масив.

3) Виконує три завдання (`first_task()`, `second_task()`, `third_task()`).



Лістинг програми:

```

1     import random
2
3     def create_array(n):
4         if n <= 0:
5             print("The number you entered is not greater than
6             0.")
7             exit()
8
9         array = [random.randint(-50, 50) for i in range(n)]
10        print("The array is: ", array)
11        return array
12
13    def first_task(array): #Find the max negative number in the
14        array

```



```

13         max_negative = None
14         for i in array:
15             if i < 0 and (max_negative is None or i >
max_negative):
16                 max_negative = i
17         if max_negative is None:
18             print("There are no negative numbers in the array.")
19         else:
20             print("The max negative number is: ", max_negative)
21
22     def second_task(array): #Calculate the arithmetic mean of
paired elements in the array
23         sum = 0
24         count = 0
25
26         for i in range(0, len(array), 2):
27             sum += array[i]
28             count += 1
29         print("The count of paired elements in the array is: ",
count)
30         print("The arithmetic mean of paired elements in the
array is: ", sum / count)
31
32     def third_task(array): #Move all non-zero elements
33         non_zero = iter([i for i in array if i != 0][::-1])
34         for index in range(len(array)):
35             if array[index] != 0:
36                 array[index] = next(non_zero)
37         print("The array with reversed non-zero elements is: ",
array)
38
39     def main():
40         n = int(input("Enter a count of numbers in the array: "))
41         array = create_array(n)
42         first_task(array)
43         second_task(array)
44         third_task(array)
45
46     main()

```

Результат виконання:

Запуск №1

```

Enter a count of numbers in the array: 20
The array is: [16, 42, 32, 21, 16, 13, 28, 7, -37, -18, 35, 25, 0, -31, -22, 16, -44, -18, -41, -5]
The max negative number is: -5
The count of paired elements in the array is: 10
The arithmetic mean of paired elements in the array is: -1.7
The array with reversed non-zero elements is: [-5, -41, -18, -44, 16, -22, -31, 25, 35, -18, -37, 7, 0, 28, 13, 16, 21, 32, 42, 16]

```

Запуск №2

```

Enter a count of numbers in the array: 0
The number you entered is not greater than 0.

```

Запуск №3

```
Enter a count of numbers in the array: 1
The array is: [6]
There are no negative numbers in the array.
The count of paired elements in the array is: 1
The arithmetic mean of paired elements in the array is: 6.0
The array with reversed non-zero elements is: [6]
```

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які особливості та переваги мови Python Ви знаєте?

- **Динамічна типізація** – змінні не потребують явного оголошення типу.
- **Велика стандартна бібліотека та розвинене співтовариство** – численні модулі та пакети для різних завдань.
- **Мультипарадигмальність** – підтримка процедурного, об'єктно-орієнтованого та функціонального програмування.

2. Назвіть основні принципи синтаксису мови Python.

- Використання **відступів** (indentation) для визначення блоків коду (без фігурних дужок).
- Лаконічність і простота запису команд.

3. Як здійснюється введення/виведення даних у мові Python?

- **Введення** – функція `input()` дозволяє зчитувати дані з консолі.
- **Виведення** – функція `print()` виводить дані на екран.

4. Який синтаксис циклу `for` у мові Python?

```
1 for element in iterable:
2     # тіло циклу
```

- Цикл перебирає елементи будь-якої послідовності (список, рядок, кортеж, словник тощо).

- Часто використовується функція `range()` для генерації послідовності чисел.

5. Назвіть принципи роботи зі списками у мові Python.

- Списки є **впорядкованими** та **змінними** колекціями.
- Доступ до елементів здійснюється за індексом (починаючи з 0).
- Підтримують **слайсинг** для отримання підсписків.