

Міністерство освіти і науки України
Національний університет „Львівська політехніка”
Кафедра ЕОМ



Звіт

до лабораторної роботи № 7

з дисципліни: «Кросплатформні засоби програмування»

На тему: «Дослідження базових конструкцій мови Python»

Виконав:
Студент групи КІ-306
Баран В. М.
Прийняв:
Іванов Ю. С.

Львів 2023

Мета роботи: ознайомитися з базовими конструкціями мови Python

ЗАВДАННЯ

1. Написати та налагодити програму на мові Python згідно варіанту.

Програма має задовольняти наступним вимогам:

- програма має розміщуватися в окремому модулі;
- програма має генерувати зубчатий список, який міститиме лише заштриховані

області квадратної матриці згідно варіанту;

- розмір квадратної матриці і символ-заповнювач масиву вводяться з клавіатури;
- при не введенні або введенні кількох символів-заповнювачів відбувається коректне

переривання роботи програми;


- сформований масив вивести на екран;
- програма має містити коментарі.

2. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.
3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її

виконання та фрагменту згенерованої документації та завантажити його у ВНС.

4. Дати відповідь на контрольні запитання.

Варіант завдання:

№	
1	

Код програми:

```
# Введення користувача розміру квадратної матриці
nRows = int(input("Введіть розмір квадратної матриці: "))

# Ініціалізація пустого списку для матриці
arr = []

# Цикл для побудови квадратної матриці зі зменшенням кількості стовпців
for i in range(nRows):
    arr.append([' ' * (nRows - i)])

# Запит користувача ввести символ-заповнювач
filler = input("Введіть символ-заповнювач: ")

# Початковий пробіл для форматування виводу
```

```

spacing = 0

# Відкриття файлу для запису результату матриці
with open("MyFile.txt", "w") as fout:
    # Цикл для кожного рядка у матриці
    for i in range(nRows):
        # Додання пробілів ліворуч від кожного рядка для форматування
        for s in range(spacing):
            print(" ", end=' ')
            fout.write(" ")

        # Перевірка довжини символу-заповнювача
        if len(filler) == 1:
            # Цикл для заповнення матриці символом-заповнювачем
            for j in range(nRows - i):
                arr[i][j] = filler[0]
                print(arr[i][j], end=' ')
                fout.write(arr[i][j] + " ")

            # Обробка випадків, коли не введено символ або введено більше одного
СИМВОЛУ
            elif len(filler) == 0:
                print("\nСимвол-заповнювач не введено")
                break # Вихід із циклу
            else:
                print("\nЗанадто багато символів-заповнювачів")
                break # Вихід із циклу

        print()
        fout.write("\n")
        spacing += 2 # Збільшення пробілу для кожного рядка

fout.flush() # Збереження виводу в файл

```

Результати роботи програми:

```

Enter the size of the square matrix: 10
Enter a placeholder character: !
! ! ! ! ! ! ! ! ! !
  ! ! ! ! ! ! ! !
    ! ! ! ! ! ! !
      ! ! ! ! ! !
        ! ! ! ! !
          ! ! ! !
            ! ! !
              ! !
                !

```

Відповіді на контрольні запитання

1. Який вигляд має програма мовою Python?
- має вигляд послідовності інструкцій у текстовому файлі з розширенням ".py".
2. Як запустити на виконання програму мовою Python? - "python ім'я_файлу.py" у командному рядку.
3. Які коментарі підтримує Python?
- однорядкові коментарі з символом "#" та багаторядкові коментарі, обмежені потрійними лапками (" або """)
4. Які типи даних підтримує Python?
- числа (цілі, дійсні, комплексні), рядки, списки, кортежі, множини, словники та інші.
5. Як оголосити змінну?
- присвоївши їй значення, наприклад, "змінна = значення".
6. Які керуючі конструкції підтримує Python?
- умовні конструкції (if-elif-else), цикли (for, while), та інструкції для контролю потоку виконання програми.
7. Які операції підтримує Python?
- додавання, віднімання, множення, ділення, цілочисельного ділення, залишок від ділення, порівняння та інші.
8. Як здійснити ввід з консолі? - "input()".
9. Як здійснити вивід у консоль? - "print()".
10. Як здійснити приведення типів?
- Приведення типів можна виконати за допомогою функцій, наприклад, "int()", "float()", "str()".

Висновок

Під час вивчення базових конструкцій мови Python, було освоєно ключові аспекти програмування. Це включає в себе вигляд програми, запуск її виконання, роботу з коментарями та типами даних, оголошення змінних, керуючі конструкції, операції, а також навички вводу та виводу даних та приведення типів.

