

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра «Електронних обчислювальних машин»



Звіт  
з лабораторної роботи № 7  
з дисципліни: «Кросплатформенні засоби програмування»  
на тему: «Дослідження базових конструкцій мови Python»

**Виконав:**

студент групи КІ-306

Ширий Б. І.

**Прийняв:**

доцент кафедри ЕОМ

Іванов Ю. С.

## МЕТОДИЧНІ ВІДОМОСТІ РОБОТИ

### МЕТА

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

Для розробленої програми згенерувати документацію

---

#### №3

Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.

---

#### №4

Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації та завантажити його у ВНС.

---

#### №5

Дати відповідь на контрольні запитання.

# ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

## ВИХІДНИЙ КОД

Написав програму, яка генерує зубчатий список, який містить лише заштриховані області квадратної матриці згідно варіанту. Код програми наведено у лістингу 2.1.

*Лістинг 2.1. Код основної програми.*

```
import sys

# Функція для закриття файлу та виходу з програми
def close_program(fout):
    fout.flush() # Забезпечуємо запис в файл перед закриттям
    fout.close() # Закриваємо файл
    sys.exit(0) # Завершуємо програму з кодом виходу 0 (без помилок)

# Головна функція програми
def main():
    filler = input("Введіть символ-заповнювач: ")
    data_file = open("Matrix.txt", "w") # Відкриваємо файл для запису

    data_file.write("Символ-заповнювач: " + filler) # Записуємо символ-заповнювач у файл

    if len(filler) > 1: # Перевіряємо чи не було введено забагато символів
        print("\nВи ввели більше ніж один символ!\nЗавершення програми.")
        data_file.write("\nВи ввели більше ніж один символ!\nЗавершення програми.")
        close_program(data_file) # Викликаємо функцію для закриття файлу та виходу

    if not filler: # Перевіряємо чи був введений символ
        print("\nВи не ввели символ-заповнювач!\nЗавершення програми.")
        data_file.write("\nВи не ввели символ-заповнювач!\nЗавершення програми.")
        close_program(data_file) # Викликаємо функцію для закриття файлу та виходу

    print("\n")
    data_file.write("\n")

    n = int(input("Введіть розмір квадратної матриці: "))
    lst = [[' ' for _ in range(n)] for _ in range(n)]

    # Генеруємо зубчастий список
    for i in range(n):
        for j in range(n):
            if (i + 1) * 2 < n:
                lst[i] = [' ' for _ in range((i + 1) * 2)]
            elif (i + 1) * 2 > n + 2:
                lst[i] = [' ' for _ in range((n - i) * 2)]

    print("\n")

    # Виводимо зубчастий масив
```

```

for i in range(n):
    for j in range(n):
        num = 0
        if (len(lst[i]) // 2 > j) or (n // 2 == i) or ((len(lst[i]) // 2 + j >= n)):
            lst[i][num] = filler[0]
            print(lst[i][num] + " ", end="")
            data_file.write(lst[i][num] + " ")
            num += 1
        else:
            print(" ", end="")
            data_file.write(" ")
    print("\n")
    data_file.write("\n")

data_file.close() # Закриваємо файл після завершення запису

if __name__ == "__main__":
    main()

```

## РЕЗУЛЬТАТИ ВИКОНАННЯ

Вивід програми в консоль наведено на рисунку 2.1, а запис у файл у лістингу 2.2.

The screenshot shows the Visual Studio Code interface. The editor displays the Python script 'CPPT\_labN7\_Shyryl.py' with the following code:

```

1 import sys
2
3 # Функція для закриття файлу та виходу з програми
4 def close_program(fout):
5     fout.flush() # Забезпечуємо запис в файл перед закриттям
6     fout.close() # Закриваємо файл
7     sys.exit(0) # Завершуємо програму з кодом виходу 0 (без помилок)
8
9 # Головна функція програми

```

The console output shows the command to run the script and the resulting output:

```

PS A:\MULP\КЗП\лабораторні\роботи\lab 7> & C:/Users/bohda/AppData/Local/Programs/Python/Python311/python.exe "a:\MULP\КЗП\лабораторні\роботи\lab 7\CPPT_labN7_Shyryl.py"
Введіть символ-заповнювач: *

```

The output is a 15x15 matrix of asterisks, with the first row being all asterisks, and subsequent rows having a space at the beginning and end, and a space between the first and last asterisk.

Рисунок 2.1. Вивід в консоль.



Рисунок 2.2. Запис у файл.

## ВІДПОВІДІ НА КОНТРОЛЬНІ ПИТАННЯ

### ЯКИЙ ВИГЛЯД МАЄ ПРОГРАМА МОВОЮ PYTHON?

Програма мовою Python має вигляд послідовності інструкцій у текстовому форматі.

### ЯК ЗАПУСТИТИ НА ВИКОНАННЯ ПРОГРАМУ МОВОЮ PYTHON?

Для запуску програми мовою Python, використовуйте команду `python` назва\_файлу.py в командному рядку (терміналі).

### ЯКІ КОМЕНТАРІ ПІДТРИМУЄ PYTHON?

Python підтримує однорядкові коментарі, починаючи з символу `#`.

### ЯКІ ТИПИ ДАНИХ ПІДТРИМУЄ PYTHON?

Python підтримує різні типи даних, включаючи цілі числа, дійсні числа, рядки, списки, кортежі, словники та інші.

---

## ЯК ОГОЛОСИТИ ЗМІННУ?

Змінну можна оголосити, просто вказавши її ім'я і призначивши їй значення. Наприклад, `змінна = 10`.

---

## ЯКІ КЕРУЮЧІ КОНСТРУКЦІЇ ПІДТРИМУЄ PYTHON?

Python підтримує керуючі конструкції, такі як умовні оператори (`if`, `else`, `elif`), цикли (`for`, `while`), та інші.

---

## ЯКІ ОПЕРАЦІЇ ПІДТРИМУЄ PYTHON?

Python підтримує операції додавання, віднімання, множення, ділення, порівняння, та багато інших.

---

## ЯК ЗДІЙСНИТИ ВВІД З КОНСОЛІ?

Для введення даних з консолі використовується функція `input()`.

---

## ЯК ЗДІЙСНИТИ ВИВІД У КОНСОЛЬ?

Для виводу даних у консоль використовується функція `print()`.

---

## ЯК ЗДІЙСНИТИ ПРИВЕДЕННЯ ТИПІВ?

Для приведення типів використовується функція `int()`, `float()`, `str()`, тощо.

---

## ВИСНОВОК

Було створено програму мовою Python, яка генерує зубчатий список, містить лише заштриховані області квадратної матриці, розмір і символ-заповнювач яких вводяться з клавіатури. Програма також має коректні перевірки введених даних та виводить сформований масив на екран. Код програми супроводжується коментарями для зрозумілості.