

**Міністерство освіти і науки України**  
**Національний університет "Львівська політехніка"**



**Звіт з лабораторної роботи №7**  
**з курсу “Кросплатформні засоби програмування”**  
**Дослідження базових конструкцій Python**

**Виконав: студент гр. КІ-306**

**Шаповал Віталій**

**Прийняв: к.т.н. Олексів М.В.**

**Львів 2024 р.**

## ТЕОРЕТИЧНІ ВІДОМОСТІ

Python програми – це набір \*.py файлів. Кожен файл – це окремий модуль. Модулі підключаються за допомогою ключового слова `import` після якого вказується назва файлу без розширення `py`. Щоб досягнути з одного модуля до іншого вони мають бути в одному каталозі. Якщо модуль лежить в іншому каталозі, то щоб до нього досягнути цей каталог має бути оформлений як пакет. Для цього достатньо у нього помістити порожній файл `__init__.py`. При підключенні пакету виконується вміст файлу `__init__.py`, який може містити визначення імен і інші необхідні для роботи з пакетом конфігураційні дії.

### Варіант № 28

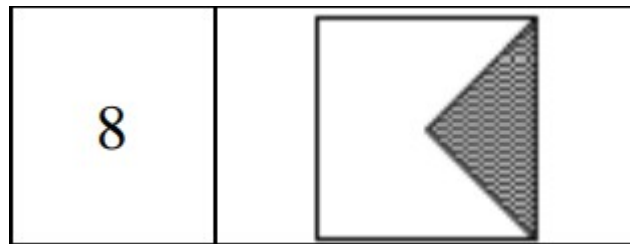


рис 1. варіант завдання

### 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.

## Код програми:

### Файл: *./lab7.py*:

```
# Import the necessary module
import sys

def main():
    # Asking user to input matrix size
    rows = int(input("Enter matrix size: "))

    if rows < 1:
        print("\nInvalid matrix size")
        return

    # Asking user to input the filler character
    filler = input("\nEnter filler character: ")

    if len(filler) == 0:
        print("\nNo filler character entered")
        return
    elif len(filler) > 1:
        print("\nToo many characters entered as filler")
        return

    # Creating matrix with specific triangular shape
    arr = []
    for i in range(rows // 2):
        arr.insert(i, [filler] * (i + 1))
        arr.insert(-i, [filler] * (i + 1))

    if rows % 2 == 1:
        arr.insert(rows // 2, [filler] * (rows // 2 + 1))

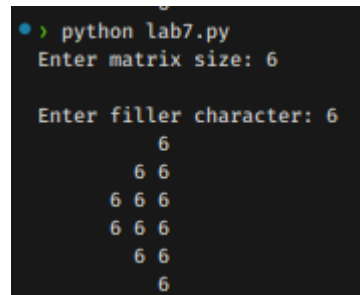
    # Displaying the matrix
    for i in range(len(arr)):
        # Printing leading spaces
        for j in range(rows - len(arr[i])):
            print(" ", end="")

        # Printing matrix elements
        for j in range(len(arr[i])):
            print(arr[i][j], end=" ")

        # Moving to the next line
        print()

if __name__ == "__main__":
    main()
```

### ***Виконання програми:***



```
> python lab7.py
Enter matrix size: 6

Enter filler character: 6
6
6 6
6 6 6
6 6 6
6 6
6
```

рис 2. результат виконання завдання в термінал

### **Висновок**

У цій лабораторній роботі я розробив програму на мові Python, що відповідає зазначеним вимогам. Програма була організована в окремому модулі та генерувала зубчастий список, який містив лише заштриховані області квадратної матриці відповідно до варіанту. Я реалізував механізм вводу, що дозволяє користувачу вводити розмір матриці та символ-заповнювач, при цьому програма коректно обробляла випадки, коли не було введено символ або було введено кілька символів.

Сформований масив виводився на екран, що дало можливість перевірити коректність виконання програми. Я також включив коментарі в код, що полегшило автоматичну генерацію документації. Завантаживши код на GitHub відповідно до методичних вказівок, я закріпив свої навички роботи з цією платформою.

Ця лабораторна робота дозволила мені глибше зрозуміти принципи роботи з зубчастими списками в Python, а також покращити навички документування коду.