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

Глухенький Д. Ю.

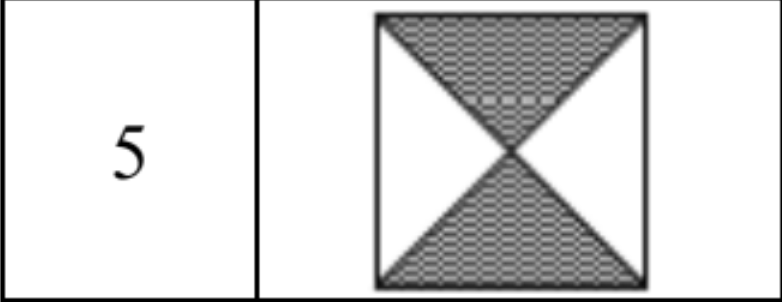
**Прийняв:**

доцент кафедри ЕОМ Іванов Ю. С.

Львів – 2023

**Мета роботи:** ознайомитися з базовими конструкціями мови Python.

# Завдання (варіант № 5)

****

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 введені або введенні кількох символів-заповнювачів відбувається коректне переривання роботи програми; • сформований масив вивести на екран;
* програма має містити коментарі.

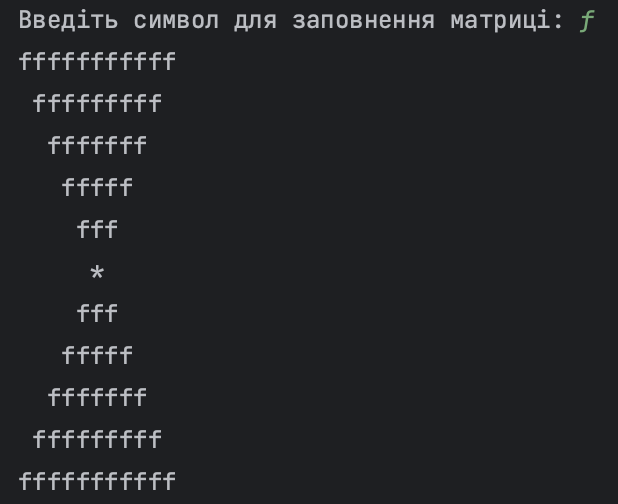
1. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.
2. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації та завантажити його у ВНС.
3. Дати відповідь на контрольні запитання.

# Вихідний код програми

**Файл lab7.py**

import os  
import struct  
import math  
import sys  
  
def size\_input():  
 while True:  
 try:  
 size = int(input("Введіть розмір квадратної матриці: "))  
 if size <= 0:  
 print("Розмір повинен бути більшим за 0")  
 continue  
 return size  
 except Exception as ex:  
 print(f"{ex} Введіть ціле число")  
  
  
def filler\_input():  
 while True:  
 try:  
 filler = input("Введіть символ для заповнення матриці: ")  
 if len(filler) != 1:  
 print("Введіть 1 символ")  
 continue  
 return filler  
 except Exception as ex:  
 print(ex)  
  
  
# def generate\_array(size, filler):  
# arr = []  
# for i in range(size):  
# row = [' '] \* size  
# if i <= size // 2:  
# row[i:size - i] = [filler] \* (size - 2 \* i)  
# else:  
# row[size - i - 1:i + 1] = [filler] \* (2 \* (i - size // 2) + 1)  
# arr.append(row)  
# return arr  
  
def generate\_array(size, filler):  
 arr = []  
 for i in range(size):  
 row = [' '] \* size  
 if i == size // 2:  
 row[i:size - i] = ['\*'] \* (size - 2 \* i)  
 elif i < size // 2:  
 row[i:size - i] = [filler] \* (size - 2 \* i)  
 else:  
 row[size - i - 1:i + 1] = [filler] \* (2 \* (i - size // 2) + 1)  
 arr.append(row)  
 return arr  
  
def print\_array(arr):  
 for row in arr:  
 print(''.join(row))  
  
def write\_middle\_rows\_to\_file(arr, filename):  
 middle\_index = len(arr) // 2  
 with open(filename, 'w') as f:  
 for i in range(middle\_index - 1, middle\_index + 2):  
 f.write(''.join(arr[i]) + '\n')  
  
  
size = size\_input()  
filler = filler\_input()  
arr = generate\_array(size, filler)  
print\_array(arr)  
  
write\_middle\_rows\_to\_file(arr, 'middle\_row.txt')

# Результат виконання програми

****

# Відповіді на контрольні запитання

1. Який вигляд має програма мовою Python?
   * має вигляд послідовності інструкцій у текстовому файлі з розширенням ".py".

1. Як запустити на виконання програму мовою Python?
   * "python ім'я\_файлу.py" у командному рядку.

1. Які коментарі підтримує Python?
   * однорядкові коментарі з символом "#" та багаторядкові коментарі, обмежені потрійними лапками (''' або """).

1. Які типи даних підтримує Python?
   * числа (цілі, дійсні, комплексні), рядки, списки, кортежі, множини, словники та інші.

1. Як оголосити змінну?
   * присвоївши їй значення, наприклад, "змінна = значення".

1. Які керуючі конструкції підтримує Python?
   * умовні конструкції (if-elif-else), цикли (for, while), та інструкції для контролю потоку виконання програми.

1. Які операції підтримує Python?
   * додавання, віднімання, множення, ділення, цілочисельного ділення, залишок від ділення, порівняння та інші.

1. Як здійснити ввід з консолі?
   * "input()".

1. Як здійснити вивід у консоль?
   * "print()".

1. 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 ввіду та виводу даних та приведення типів.