Міністерство освіти і науки України

Національний університет «Львівська політехніка»



**ЗВІТ**

про виконання лабораторної роботи № 4

з курсу:

«Спеціалізовані мови програмування»

**Виконав:**

студент гр. ІТ-31

Роман ХРАПЧУН

**Прийняв:**

Сергій ЩЕРБАК

Львів 2023

**Тема:** Розробка ASCII ART генератора для візуалізації 2D-фігур

**Мета роботи:** Створення Генератора ASCII-арту без використання зовнішніх бібліотек

**Хід роботи**

***Завдання 1: Введення користувача***

Створіть програму Python, яка отримує введення користувача щодо слова або фрази, яку вони хочуть перетворити в ASCII-арт.

***Завдання 2: Набір символів***

Визначте набір символів (наприклад, '@', '#', '\*', тощо), які будуть використовуватися для створення ASCII-арту. Ці символи будуть відображати різні відтінки.

***Завдання 3: Розміри Art-у***

Запитайте у користувача розміри (ширина і висота) ASCII-арту, який вони хочуть створити. Переконайтеся, що розміри в межах керованого діапазону

***Завдання 4: Функція генерації Art-у***

Напишіть функцію, яка генерує ASCII-арт на основі введення користувача, набору символів та розмірів. Використовуйте введення користувача, щоб визначити, які символи використовувати для кожної позиції в Art-у.

***Завдання 5: Вирівнювання тексту***

Реалізуйте опції вирівнювання тексту (ліво, центр, право), щоб користувачі могли вибирати, як їх ASCII-арт розміщується на екрані.

***Завдання 6: Відображення мистецтва***

Відобразіть створений ASCII-арт на екрані за допомогою стандартних функцій друку Python.

***Завдання 7: Збереження у файл***

Додайте можливість зберігати створений ASCII-арт у текстовий файл, щоб користувачі могли легко завантажувати та обмінюватися своїми творіннями.

***Завдання 8: Варіанти кольорів***

Дозвольте користувачам вибирати опції кольорів (чорно-білий, відтінки сірого) для свого ASCII-арту.

***Завдання 9: Функція попереднього перегляду***

Реалізуйте функцію попереднього перегляду, яка показує користувачам попередній перегляд їх ASCII-арту перед остаточним збереженням

***Завдання 10: Інтерфейс, зрозумілий для користувача***

Створіть інтерфейс для користувача у командному рядку, щоб зробити програму легкою та інтуїтивно зрозумілою для використання.

**Реалізований код:**

import random

import os

characters = {

'a': [

" AA ",

" A A ",

" AAAAA",

" A A ",

" A A "

],

'b': [

" BBB ",

" B B ",

" BBBB ",

" B B ",

" BBB "

],

'c': [

" CCCC ",

" C ",

" C ",

" C ",

" CCCC "

],

'e': [

" EEEE ",

" E ",

" EEEE ",

" E ",

" EEEE "

],

'g': [

" GGGG ",

" G ",

" G GG ",

" G G ",

" GGGG "

],

'i': [

" III ",

" I ",

" I ",

" I ",

" III "

],

'r': [

" RRR ",

" R R ",

" RRR ",

" R R ",

" R R "

],

's': [

" SSSS ",

" S ",

" SSSS ",

" S ",

" SSSS "

],

'y': [

" Y Y ",

" YY ",

" Y ",

" Y ",

" Y "

]

}

def run\_lab4():

"""

Основна функція для виконання завдань лабораторної роботи 4.

"""

color\_option = input("Виберіть опцію кольорів (чорно-білий/відтінки сірого): ").lower()

input\_text = input("Введіть слово або фразу для ASCII-арту: ")

alignment = input("Виберіть вирівнювання (ліво, центр, право): ").lower()

if alignment not in ['ліво', 'центр', 'право']:

alignment = 'ліво'

width = int(input("Введіть ширину ASCII-арту: "))

height = int(input("Введіть висоту ASCII-арту: "))

def generate\_ascii\_art(input\_text, characters, alignment, width, height, color\_option):

"""

Функція для генерації ASCII-арту.

:param input\_text: Введений текст для ASCII-арту.

:param characters: Символи для генерації ASCII-арту.

:param alignment: Вирівнювання тексту ('ліво', 'центр', 'право').

:param width: Ширина ASCII-арту.

:param height: Висота ASCII-арту.

:param color\_option: Опція кольорів ('чорно-білий', 'відтінки сірого').

:return: Список рядків ASCII-арту.

"""

lines = []

if alignment == 'центр':

lines.append(input\_text.center(width))

elif alignment == 'право':

lines.append(input\_text.rjust(width))

else:

lines.append(input\_text.ljust(width))

for i in range(5):

line = ""

for char in input\_text:

if char.lower() in characters:

line += characters[char.lower()][i]

else:

line += " " \* 6

line += " "

if color\_option == 'чорно-білий':

line = "\033[30m" + line + "\033[0m"

elif color\_option == 'відтінки сірого':

line = "\033[90m" + line + "\033[0m"

lines.append(line)

return lines

def display\_ascii\_art(art):

"""

Функція для відображення ASCII-арту.

:param art: Список рядків ASCII-арту.

"""

for line in art:

print(line)

def save\_ascii\_art\_to\_file(ascii\_art):

"""

Функція для збереження ASCII-арту у файл у папці Data, яка знаходиться на одному рівні з папкою Core.

"""

file\_name = input("Введіть ім'я файлу для збереження (з розширенням .txt): ")

data\_directory = 'Data'

if not os.path.exists(data\_directory):

os.makedirs(data\_directory)

full\_file\_path = os.path.join(data\_directory, file\_name)

try:

with open(full\_file\_path, 'w') as file:

file.write(ascii\_art)

print(f"ASCII-арт збережено у файлі {full\_file\_path}")

except Exception as e:

print(f"Помилка при збереженні файлу: {e}")

def preview\_ascii\_art(art):

"""

Функція для попереднього перегляду ASCII-арту.

:param art: Список рядків ASCII-арту.

"""

print("Попередній перегляд ASCII-арту:")

for line in art:

print(line)

art = generate\_ascii\_art(input\_text, characters, alignment, width, height, color\_option)

display\_ascii\_art(art)

save\_option = input("Бажаєте зберегти ASCII-арт у файл? (так/ні): ").lower()

if save\_option == 'так':

save\_ascii\_art\_to\_file(art)

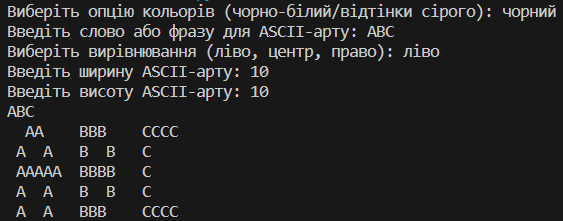
preview\_option = input("Бажаєте побачити попередній перегляд ASCII-арту? (так/ні): ").lower()

if preview\_option == 'так':

preview\_ascii\_art(art)

if \_\_name\_\_ == "\_\_main\_\_":

run\_lab4()



*Рис. 1 – Результат виконання програми*

**Висновок:** На даній лабораторній роботі я створив універсальний Генератор ASCII-арту, який дозволить користувачам налаштовувати свої творіння з різними шрифтами, кольорами, розмірами та символами.