МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж



Лабораторна робота №3

з дисципліни Спеціалізовані мови програмування

на тему

Розробка ASCII ART генератора для візуалізації текстових даних

Виконав:

студент групи РІ-21сп

Данило ДЯЧОК

Львів – 2024

**Мета виконання лабораторної роботи:** створення додатка Генератора ASCII-арту.

**План роботи**

**Завдання 1: Введення користувача**

Створіть Python-програму, яка приймає введення користувача для слова або фрази, яку треба перетворити в ASCII-арт.

**Завдання 2: Бібліотека ASCII-арту**

Інтегруйте бібліотеку ASCII-арту (наприклад, pyfiglet або art) у вашу програму для генерації ASCII-арту з введення користувача

**Завдання 3: Вибір шрифту**

Дозвольте користувачам вибирати різні стилі шрифтів для свого ASCII-арту. Надайте список доступних шрифтів та дозвольте їм вибрати один.

**Завдання 4: Колір тексту**

Реалізуйте опцію вибору користувачем кольору тексту для їхнього ASCII-арту. Підтримуйте основний вибір кольорів (наприклад, червоний, синій, зелений).

**Завдання 5: Форматування виводу**

Переконайтеся, що створений ASCII-арт правильно відформатований та вирівнюється на екрані для зручності читання.

**Завдання 6: Збереження у файл**

Додайте функціональність для збереження створеного ASCII-арту у текстовому файлі, щоб користувачі могли легко завантажувати та обмінюватися своїми творіннями.

**Завдання 7: Розмір ARTу**

Дозвольте користувачам вказувати розмір (ширина і висота) ASCII-арту, який вони хочуть створити. Масштабуйте текст відповідно.

**Завдання 8: Вибір символів**

Дозвольте користувачам вибирати символи, які вони хочуть використовувати для створення ASCII-арту (наприклад, '@', '#', '\*', тощо).

**Завдання 9: Функція попереднього перегляду**

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

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

Створіть зручний для користувача інтерфейс командного рядку для додатка, щоб зробити його інтуїтивно зрозумілим та легким у використанні.

**Текст програмної реалізації:**

**art\_generator.py:**import pyfiglet

from termcolor import colored

def generate\_ascii\_art(text, font='standard', color='white', width=0, symbols=''):

if width > 0:

ascii\_art = pyfiglet.figlet\_format(text, font=font, width=width)

else:

ascii\_art = pyfiglet.figlet\_format(text, font=font)

if symbols:

ascii\_art = replace\_symbols(ascii\_art, symbols)

colored\_ascii\_art = colored(ascii\_art, color)

return colored\_ascii\_art

def replace\_symbols(ascii\_art, symbols):

symbol\_map = {' ': symbols[0] if len(symbols) > 0 else ' ',

'\_': symbols[1] if len(symbols) > 1 else '\_',

'|': symbols[2] if len(symbols) > 2 else '|',

'/': symbols[3] if len(symbols) > 3 else '/',

'\\': symbols[4] if len(symbols) > 4 else '\\'}

return ''.join(symbol\_map.get(char, char) for char in ascii\_art)

def save\_ascii\_art(ascii\_art, filename):

with open(filename, 'w') as file:

file.write(ascii\_art)

**Результати тестування:**

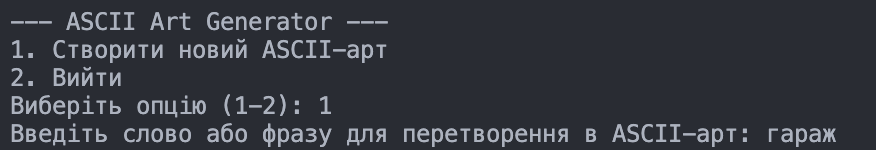


Рис. 1. Консольний інтерфейс

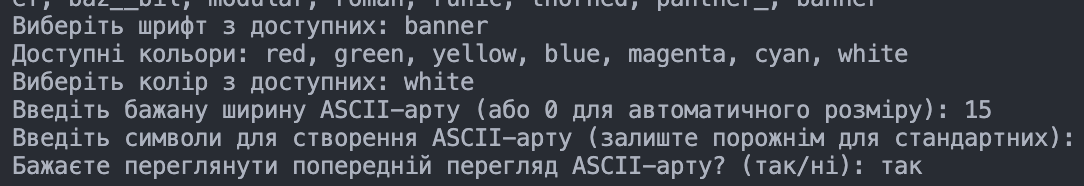


Рис. 2. Консольний інтерфейс



Рис. 3. Результат виведення

**Висновки:** На цій лабораторній роботі я створив додаток Генератора ASCII-арту.