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

Дмитрій Сас

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 colorama import Fore, Style, init

# Ініціалізуємо бібліотеку colorama для роботи з кольорами

init(autoreset=True)

class ASCIIArtGenerator:

    def \_\_init\_\_(self):

        self.text = ""

        self.font = "slant"

        self.color = Fore.WHITE

        self.symbol = "#"

    def get\_user\_input(self):

        # Введення користувача для тексту

        self.text = input("Vvedit' slovo abo frazu dlia ASCII-artu: ")

    def choose\_font(self):

        # Вибір шрифту

        fonts = pyfiglet.FigletFont.getFonts()

        print("Dostupni shrifti:")

        for i, font in enumerate(fonts[:10], 1):  # Вивести лише кілька прикладів шрифтів

            print(f"{i}. {font}")

        font\_choice = input("Vyberit' nomer shryftu abo vvedit' svoi: ")

        if font\_choice.isdigit() and int(font\_choice) <= len(fonts[:10]):

            self.font = fonts[int(font\_choice) - 1]

        else:

            self.font = font\_choice if font\_choice in fonts else "slant"

    def choose\_color(self):

        # Вибір кольору тексту

        colors = {

            "1": Fore.RED, "2": Fore.GREEN, "3": Fore.BLUE,

            "4": Fore.YELLOW, "5": Fore.MAGENTA, "6": Fore.CYAN, "7": Fore.WHITE

        }

        print("Dostupni kol'ory: 1. Chervonyi 2. Zelenyi 3. Synii 4. Zhouvtyi 5. Fioletovyi 6. Biryzovyi 7. Bilyi")

        color\_choice = input("Vyberit' kol'ir za nomerom: ")

        self.color = colors.get(color\_choice, Fore.WHITE)

    def generate\_ascii\_art(self):

        # Генерація ASCII-арту

        figlet = pyfiglet.Figlet(font=self.font)

        ascii\_art = figlet.renderText(self.text)

        return ascii\_art

    def preview\_art(self):

        # Попередній перегляд ASCII-арту

        ascii\_art = self.generate\_ascii\_art()

        print("\nPoperednij perehliad:")

        print(self.color + ascii\_art)

    def save\_to\_file(self, ascii\_art):

        # Збереження у файл

        with open("ascii\_art.txt", "w", encoding="utf-8") as file:

            file.write(ascii\_art)

        print("ASCII-art zberezheno u faili 'ascii\_art.txt'")

    def run(self):

        # Основний інтерфейс користувача

        print("Vitajemo u ASCII ART Generatori!")

        self.get\_user\_input()

        self.choose\_font()

        self.choose\_color()

        self.preview\_art()

        save\_choice = input("Bazhaete zberehty ASCII-art u fail? (tak/ni): ").strip().lower()

        if save\_choice == "tak":

            ascii\_art = self.generate\_ascii\_art()

            self.save\_to\_file(ascii\_art)

        print("Diakujemo za vykorystannia ASCII ART Generatora!")

# Запуск генератора

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

    generator = ASCIIArtGenerator()

    generator.run()

**Результати тестування:**

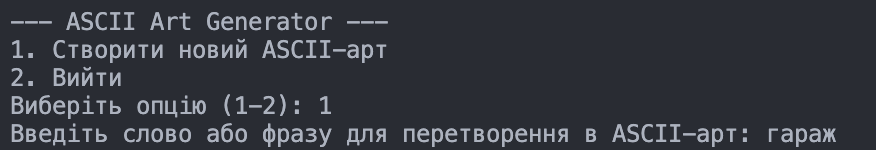


Рис. 1. Консольний інтерфейс

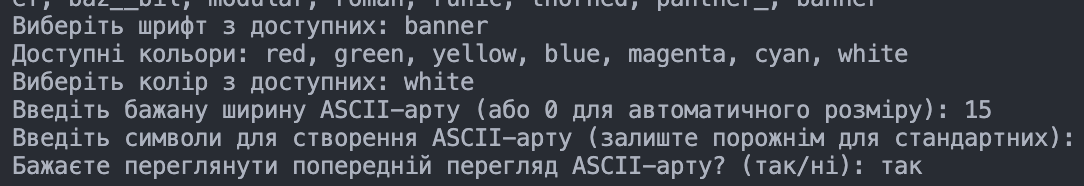


Рис. 2. Консольний інтерфейс



Рис. 3. Результат виведення

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