МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж

Лабораторна робота №4

з дисципліни

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

на тему

РОЗРОБКА ASCII ART ГЕНЕРАТОРА ДЛЯ ВІЗУАЛІЗАЦІЇ 2D-ФІГУР

Виконала:

ст. гр. ІТ-21сп

Кобленц Д.А.

Прийняв:

Щербак С.С.

Львів-2023

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

**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.

**Код програми:**  
# Task 10: User-friendly interface

import pyfiglet #Завдання 2: Бібліотека ASCII-арту

from colorama import Fore, Style

print(Fore.LIGHTMAGENTA\_EX + "Welcome to ASCII ART program" + Style.RESET\_ALL)

def get\_user\_input():

# Task 1: User input

user\_input = input(Fore.LIGHTYELLOW\_EX + "Enter a word or phrase to convert to ASCII art: ")

# Task 3: Choosing a font

print("Available font styles:\n" + Fore.LIGHTWHITE\_EX + "1. standard\n2. banner\n3. slant\n4. script"+ Fore.LIGHTYELLOW\_EX)

while True:

try:

font\_choice = int(input("Choose a font style number" + Fore.LIGHTWHITE\_EX + "(1-4):" + Fore.LIGHTYELLOW\_EX))

if 1 <= font\_choice <= 5:

break

else:

print("Please enter a number between 1 and 4.")

except ValueError:

print("Please enter the correct font style number.")

# Task 4: Text color

print("Available text colors:\n" + Fore.LIGHTRED\_EX + "1. Red\n" + Fore.LIGHTBLUE\_EX + "2. Blue\n" + Fore.LIGHTGREEN\_EX + "3. Green")

while True:

try:

color\_choice = int(input(Fore.LIGHTYELLOW\_EX + "Choose a text color number " + Fore.LIGHTWHITE\_EX + "(1-3):" + Fore.LIGHTYELLOW\_EX))

if 1 <= color\_choice <= 3:

break

else:

print("Please enter a number between 1 and 3.")

except ValueError:

print("Please enter the correct text color number.")

while True:

try:

width = int(input("Enter the width of the ASCII art " + Fore.LIGHTWHITE\_EX + "(number of characters in a line):" + Fore.LIGHTYELLOW\_EX))

if width > 0:

break

else:

print("Please enter a number greater than zero.")

except ValueError:

print("Please enter the correct width.")

while True:

try:

height = int(input("Enter the height of the ASCII art " + Fore.LIGHTWHITE\_EX + "(number of lines):" + Fore.LIGHTYELLOW\_EX))

if height > 0:

break

else:

print("Please enter a number greater than zero.")

except ValueError:

print("Please enter the correct height.")

# Task 8: Selection of symbols

use\_custom\_chars = input("Do you want to choose a special character to create ASCII art? " + Fore.LIGHTWHITE\_EX + "(y/n):" + Fore.LIGHTYELLOW\_EX).strip().lower()

if use\_custom\_chars == 'y':

char\_set = input("Enter the characters you want to use for ASCII art " + Fore.LIGHTWHITE\_EX + "(eg '@#\*'):" + Fore.LIGHTYELLOW\_EX)

else:

char\_set = None

preview\_enabled = input("Want to preview your ASCII art before saving?" + Fore.LIGHTWHITE\_EX + " (y/n):"+ Fore.LIGHTYELLOW\_EX).strip().lower()

return user\_input, font\_choice, color\_choice, width, height, char\_set, preview\_enabled

# Task 9: Preview function

def preview\_ascii\_art(ascii\_text, selected\_color):

print(selected\_color + ascii\_text + Style.RESET\_ALL)

def generate\_ascii\_art(text, font\_choice, color\_choice, width, height, char\_set=None):

fonts = ['standard', 'banner', 'slant', 'script']

selected\_font = fonts[font\_choice - 1]

ascii\_art = pyfiglet.Figlet(font=selected\_font)

colors = [Fore.LIGHTRED\_EX, Fore.LIGHTBLUE\_EX, Fore.LIGHTGREEN\_EX]

selected\_color = colors[color\_choice - 1]

ascii\_text = ascii\_art.renderText(text)

ascii\_lines = ascii\_text.split('\n')

# Task 7: The size of ART

scaled\_ascii\_lines = []

char\_set\_length = len(char\_set) if char\_set else 0

for line in ascii\_lines:

scaled\_line = ""

for char in line:

if char == ' ':

scaled\_line += ' '

else:

if char\_set:

scaled\_line += char\_set[hash(char) % char\_set\_length]

else:

scaled\_line += char

scaled\_ascii\_lines.append(scaled\_line.center(width)[:width])

scaled\_ascii\_text = '\n'.join(scaled\_ascii\_lines[:height])

colored\_text = selected\_color + scaled\_ascii\_text + Style.RESET\_ALL

return colored\_text

def main():

user\_input, font\_choice, color\_choice, width, height, char\_set, preview\_enabled = get\_user\_input()

ascii\_text = generate\_ascii\_art(user\_input, font\_choice, color\_choice, width, height, char\_set)

if preview\_enabled == 'y':

selected\_color = [Fore.LIGHTRED\_EX, Fore.LIGHTBLUE\_EX, Fore.LIGHTGREEN\_EX][color\_choice - 1]

preview\_ascii\_art(ascii\_text, selected\_color)

# Task 6: Saving to a file

save\_confirmation = input(Fore.LIGHTYELLOW\_EX + "Want to save ASCII art to a file?" + Fore.LIGHTWHITE\_EX + " (y/n):").strip().lower()

if save\_confirmation == 'y':

file\_name = input(Fore.LIGHTYELLOW\_EX + "Enter a file name to save the ASCII art " + Fore.LIGHTWHITE\_EX + "(without extension):")

with open(f"{file\_name}.txt", "w") as file:

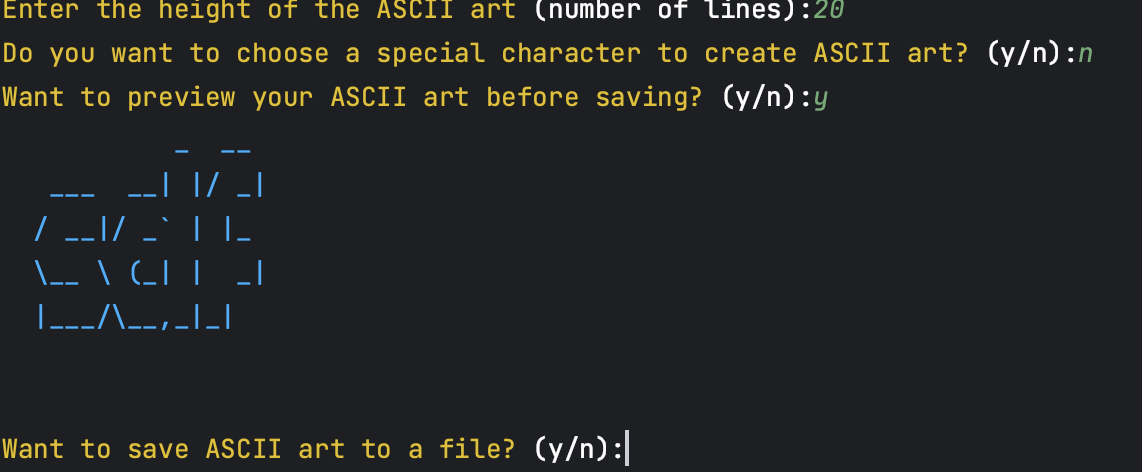
file.write(ascii\_text)

print(Fore.LIGHTYELLOW\_EX + f"ASCII art is saved in a file {file\_name}.txt\n" + Fore.LIGHTMAGENTA\_EX + "Thank you for using my program!")

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

main()

На рисунку 1 зображено результат виконання всіх 10 завдань.



**Рис. 1** Результат завдань

**Висновок:** Виконуючи ці завдання, я створив універсальний Генератор ASCII-арту, який дозволить користувачам налаштовувати свої творіння з різними шрифтами, кольорами, розмірами та символами. Проект надасть практичний досвід роботи з введенням користувача, зовнішніми бібліотеками, роботою з файлами та дизайном інтерфейсу користувача в Python.