мІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

нАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «лЬВІВСЬКА ПОЛІТЕХНІКА»

ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗВІТ

для лабораторної роботи № 3

з дисципліни

«Спеціалізовані мови програмування»

Виконав:

студент гр. ІТ-32

Сташкевич Владислав

Прийняв:

доц. каф. ІСМ

Щербак С.С.

Львів-2023

**Мета роботи:** Створення додатка Генератора ASCII-арту.

**Хід виконання:**

**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:**

import art

def choose\_font():

art.font\_list("test", "ascii")

return input("Choose the font to use: ")

def choose\_color() -> int:

return int(input("Choose the color to use: \n"

"1. Black \n"

"2. Red \n"

"3. Blue \n"

">> "))

def choose\_width() -> int:

return int(input("Enter width as a number of spaces: "))

def get\_art(font\_name: str = "z-pilot", color\_choice: int = 0, spaces: int = 0) -> str:

color\_text = '\033[%dm%s\033[0m'

text = input("Enter a text for your art: ")

color\_number = 0

if color\_choice == 2:

color\_number = 91

elif color\_choice == 3:

color\_number = 94

new\_art = art.text2art(text, font\_name, space = spaces)

return color\_text % (color\_number, new\_art)

def write\_into\_file(file\_path, content):

with open(file\_path, "a") as file:

file.write(content)

def read\_from\_file(file\_path):

with open(file\_path, "r") as file:

return file.read()

font = "z-pilot"

color = 0

width = 0

while True:

choice = int(input("Choose one: \n"

"1. Make an art\n"

"2. Change art font\n"

"3. Change art color\n"

"4. Change art width\n"

"5. See previous arts\n"

"6. Exit\n"

">> "))

if choice == 1:

art\_text = get\_art(font, color, width)

print(art\_text)

choice = int(input("Choose the next action: \n"

"1. Save an art\n"

"2. Return to the main menu\n"

"3. Exit\n"

">> "))

if choice == 1:

write\_into\_file("art.txt", art\_text)

elif choice == 3:

break

elif choice == 2:

font = choose\_font()

elif choice == 3:

color = choose\_color()

elif choice == 4:

width = choose\_width()

elif choice == 5:

print(read\_from\_file("art.txt"))

elif choice == 6:

break

else:

print("Wrong choice, try again :(")

**Посилання на GitHub-репозиторій:** <https://github.com/VladyslavStashkevych/SMP>

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