

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
ІНСТИТУТ КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗВІТ
для лабораторної роботи № 3 з
дисципліни
«Спеціалізовані мови програмування»

Виконав:
студент гр. ІТ-32
Паньків Б. В.

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
import termcolor
from art import *
from termcolor import *
```

```

def user_input():

    try:
        fonts = art_param.XLARGE_WIZARD_FONT
        text = str(input("Enter text: "))
        print("Enter font from: ")
        print(fonts)
        font = (input())
        if font not in fonts:
            raise ValueError
        color = input("Enter color: ")
        if color not in termcolor.COLORS:
            raise ValueError
        size = int(input("Enter size modifier: "))
        char = input("Enter char(type ' ' to skip): ")
    except ValueError:
        font = 'dotmatrix'
        text = "test"
        color = 'red'
        size = 1
        char = ' '

    return text, font, color, size, char

```

```

def draw(text, font, color, size, char):
    art = text2art(text, font)
    with open("text.txt", 'w') as file:
        file.write(art)

    res = ""

    with open('text.txt', 'r') as f:
        for line in f:
            output = "".join([size * c for c in line.strip()])

            for _ in range(size):
                res = res + output + '\n'

    if (char != ' '):
        modif = ""
        for letter in res:
            if letter == ' ' or letter == '\n':

```

```
        modif += letter
    else:
        modif += char
    res = modif
```

```
return colored(res, color)
```

```
def __main__():
    text, font, color, size, char = user_input()
    res = draw(text, font, color, size, char)
    print(res)
    isSave = input("Save the art? (y/n)")
    if (isSave == 'y'):
        with open("text_out.txt", 'w') as f:
            f.write(res)
```

```
if __name__ == "__main__":
    __main__()
```

Посилання на GitHub-репозиторій: <https://github.com/BOHDAN1329/SMP>

Висновки: Виконавши ці завдання, я створив додаток генератора ASCII-арту.