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

Виконав:

студент групи РІ-21сп

Владислав ДМИТРЕНКО

Львів – 2024

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

**План роботи**

**Завдання 1:** Створіть програму Python, яка отримує введення користувача щодо слова або фрази, яку вони хочуть перетворити в ASCII-арт.

**Завдання 2:** Визначте набір символів (наприклад, '@', '#', '\*', тощо), які будуть використовуватися для створення ASCII-арту. Ці символи будуть відображати різні відтінки.

**Завдання 3:** Запитайте у користувача розміри (ширина і висота) ASCII-арту, який вони хочуть створити. Переконайтеся, що розміри в межах керованого діапазону

**Завдання 4:** Напишіть функцію, яка генерує ASCII-арт на основі введення користувача, набору символів та розмірів. Використовуйте введення користувача, щоб визначити, які символи використовувати для кожної позиції в Art-у.

**Завдання 5:** Реалізуйте опції вирівнювання тексту (ліво, центр, право), щоб користувачі могли вибирати, як їх ASCII-арт розміщується на екрані.

**Завдання 6:** Відобразіть створений ASCII-арт на екрані за допомогою стандартних функцій друку Python.

**Завдання 7:** Додайте можливість зберігати створений ASCII-арт у текстовий файл, щоб користувачі могли легко завантажувати та обмінюватися своїми творіннями.

**Завдання 8:** Дозвольте користувачам вибирати опції кольорів (чорно-білий, відтінки сірого) для свого ASCII-арту.

**Завдання 9:** Реалізуйте функцію попереднього перегляду, яка показує користувачам попередній перегляд їх ASCII-арту перед остаточним збереженням

**Завдання 10:** Створіть інтерфейс для користувача у командному рядку, щоб зробити програму легкою та інтуїтивно зрозумілою для використання.

**Результати тестування:**

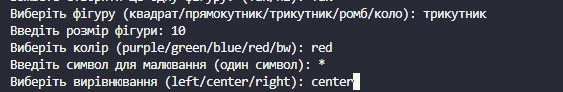
****

Рис. 1. Приклад користувацького інтерфейсу

  
Рис. 2. Результат генерації арту

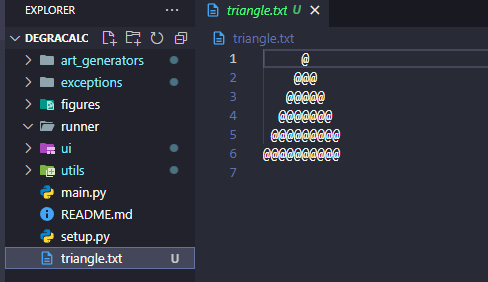


Рис. 3. Результат збереження

Текст обробника помилок

class ArtGenerationError(Exception):

def \_\_init\_\_(self, message="Помилка під час генерації фігури ASCII"):

self.message = message

super().\_\_init\_\_(self.message)

def \_\_str\_\_(self):

return f'ArtGenerationError: {self.message}'

Текст функції генерації кола

from exceptions.art\_errors import ArtGenerationError

def generate\_circle(radius):

"""

Генерує коло ASCII із заданим радіусом.

:param radius: Радіус кола.

:return: Список, що представляє коло (матриця 1 і 0).

"""

try:

if radius <= 0:

raise ArtGenerationError("Радіус повинен бути додатнім числом.")

return [[1 if (i - radius) \*\* 2 + (j - radius) \*\* 2 <= radius \*\* 2 else 0 for j in range(2\*radius+1)] for i in range(2\*radius+1)]

except Exception as e:

raise ArtGenerationError(f"Помилка під час генерації кола: {e}")

Текст інсталятора бібліотеки

from setuptools import setup, find\_packages

setup(

name="ascii\_art\_generator",

version="1.0.0",

description="Модуль для генерації ASCII-фігур",

author="Вlxxd",

packages=find\_packages(),

install\_requires=[],

)

**Висновки:** В ході виконання лабораторної роботи було створено генератор ASCII-арту з нуля та надано можливість налаштовувати символи розміри, вирівнювання, кольори. Також було створено інсталятор, для того, щоб можна було інсталювати бібліотеку та використовувати функції з неї.