МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж

Лабораторна робота №5

з дисципліни

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

на тему

Розробка ASCII ART генератора для візуалізації 3D-фігур

Виконав:

студент групи РІ-21сп

Богдан Д.А.

Прийняв:

Шербак С.С.

Львів-2024

**Мета лабораторної роботи:** Cтворення додатка для малювання 3D-фігур у ASCII-арті на основі об’єктно - орієнтованого підходу та мови Python

**Завдання**

**Завдання 1: Проектування класів.**

Розробіть структуру класів для вашого генератора 3D ASCII-арту. Визначте основні компоненти, атрибути та методи, необхідні для програми.

**Завдання 2: Введення користувача.**

Створіть методи у межах класу для введення користувача та вказання 3D-фігури, яку вони хочуть намалювати, та її параметрів (наприклад, розмір, кольори).

**Завдання 3: Представлення фігури.**

Визначте структури даних у межах класу для представлення 3D-фігури. Це може включати використання списків, матриць або інших структур даних для зберігання форми фігури та її властивостей.

**Завдання 4: Проектування з 3D в 2D.**

Реалізуйте метод, який перетворює 3D-представлення фігури у 2D-представлення, придатне для ASCII-арту.

**Завдання 5: Відображення ASCII-арту.**

Напишіть метод у межах класу для відображення 2D-представлення 3D-фігури як ASCII-арту. Це може включати відображення кольорів і форми за допомогою символів ASCII.

**Завдання 6: Інтерфейс, зрозумілий для користувача.**

Створіть зручний для користувача командний рядок або графічний інтерфейс користувача (GUI) за допомогою об'єктно-орієнтованих принципів, щоб дозволити користувачам спілкуватися з програмою.

**Завдання 7: Маніпуляція фігурою.**

Реалізуйте методи для маніпулювання 3D-фігурою, такі масштабування або зміщення, щоб надавати користувачам контроль над її виглядом.

**Завдання 8: Варіанти кольорів.**

Дозвольте користувачам вибирати варіанти кольорів для їхніх 3D ASCII-арт-фігур. Реалізуйте методи для призначення кольорів різним частинам фігури.

**Завдання 9: Збереження та експорт.**

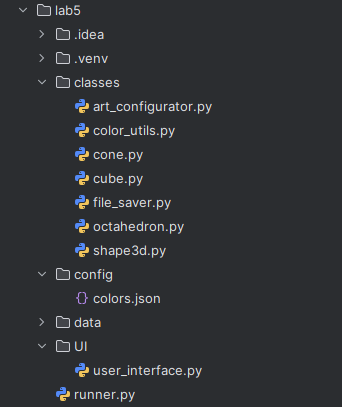
Додайте функціональність для зберігання згенерованого 3D ASCII-арту у текстовий файл

**Завдання 10: Розширені функції.**

Розгляньте можливість додавання розширених функцій, таких як тінь, освітлення та ефекти перспективи, для підвищення реалізму 3D ASCII-арту.

**Виконання роботи**

**Структура файлів:**



**Вміст файлу user\_interface.py:**

import os  
import sys  
  
lab5\_root = os.path.abspath(os.path.join(os.path.dirname(\_\_file\_\_), ".."))  
sys.path.append(lab5\_root)  
  
from lab5.classes.cube import Cube  
from lab5.classes.cone import Cone  
from lab5.classes.octahedron import Octahedron  
from lab5.classes.art\_configurator import ArtConfigurator  
  
def main():  
 art\_configurator = ArtConfigurator()  
  
 while True:  
 try:  
 shape\_type = input("Виберіть 3D форму (cube, cone, octahedron): ").strip().lower()  
 try:  
 size = int(input("Введіть розмір: "))  
 except ValueError:  
 raise ValueError("Розмір має бути цілим числом.")  
 match shape\_type:  
 case 'cube':  
 shape = Cube(size)  
 case 'cone':  
 shape = Cone(size)  
 case 'octahedron':  
 shape = Octahedron(size)  
 case \_:  
 raise ValueError("Невірний тип форми.")  
 break  
 except ValueError as e:  
 print(e)  
  
 default\_color\_code = art\_configurator.reset\_color()  
  
 while True:  
 try:  
 print("\nВиберіть дію:")  
 print("1. Повернути форму")  
 print("2. Змінити розмір")  
 print("3. Вибрати колір")  
 print("4. Відобразити ASCII-арт")  
 print("5. Змінити форму")  
 print("6. Зберегти ASCII-арт в файл")  
 print("7. Вийти")  
  
 choice = input("Ваш вибір: ").strip()  
  
 match choice:  
 case '1':  
 art\_configurator.rotate\_shape(shape)  
  
 case '2':  
 art\_configurator.change\_shape\_size(shape)  
  
 case '3':  
 color\_code = art\_configurator.choose\_color()  
  
 case '4':  
 projection = shape.project\_to\_2d()  
 shape.render\_ascii(projection, color\_code if 'color\_code' in locals() else default\_color\_code)  
  
 case '5':  
 shape\_type = input("Виберіть нову 3D форму (cube, cone, octahedron): ").strip().lower()  
 try:  
 size = int(input("Введіть розмір для нової форми: "))  
 except ValueError as e:  
 raise ValueError("Розмір повинен бути цілим числом.")  
 match shape\_type:  
 case 'cube':  
 shape = Cube(size)  
 case 'cone':  
 shape = Cone(size)  
 case 'octahedron':  
 shape = Octahedron(size)  
 case \_:  
 raise ValueError("Невірний тип форми.")  
  
 case '6':  
 projection = shape.project\_to\_2d()  
 art\_configurator.save\_art(projection, shape)  
  
 case '7':  
 print("Вихід з програми.")  
 break  
  
 case \_:  
 print("Невірний вибір.")  
  
 except (ValueError, FileNotFoundError, NotImplementedError) as e:  
 print(f"Помилка: {e}")

Результатом виконання лабораторної роботи є генератор ASCII-артів для 3D фігур з можливістю міняти розмір фігури, обертати її навколо своєї осі, міняти колір фігури

На рисунку 1 зображено інтерфейс користувача.

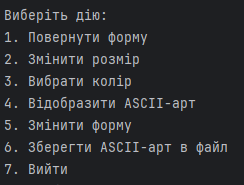
****

Рисунок 1 – Інтерфейс користувача

На рисунку 2 зображено згенерований кольоровий октаедр.

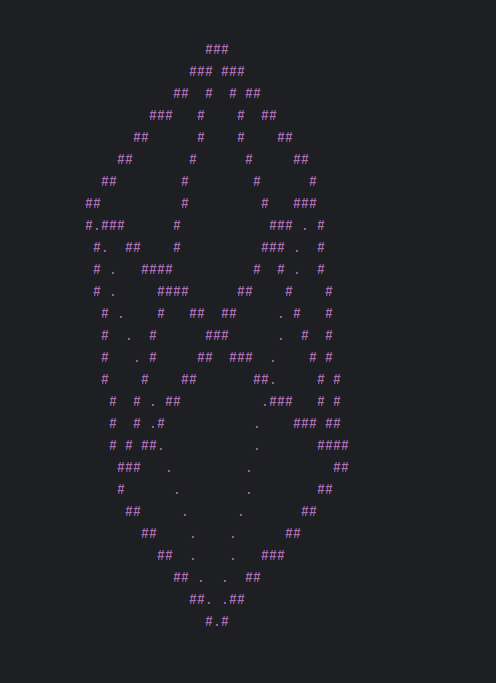
****

Рисунок 2 **–** Згенерованийкольоровий октаедр

**Висновок:** У ході виконання ЛР я створив високорівневий об'єктно-орієнтований генератор 3D ASCII-арту, який дозволяє користувачам проектувати, відображати та маніпулювати 3D-фігурами в ASCII-арті, а також міняти їх колір та зберігати їх у файлах.