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

Дмитрій Сас

Львів – 2024

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

**План роботи**

**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.

**Текст програмної реалізації:**

**art\_generator.py:**import pandas as pd

import matplotlib.pyplot as plt

import os

class CSVVisualizer:

    def \_\_init\_\_(self, filename):

        self.filename = filename

        self.data = None

        # Перевірка наявності файлу, створення, якщо не існує

        if not os.path.exists(self.filename):

            self.create\_sample\_csv()

        self.load\_data()

    def create\_sample\_csv(self):

        # Приклад базового CSV-файлу

        sample\_data = {

            "Date": ["2024-01-01", "2024-02-01", "2024-03-01", "2024-04-01"],

            "Value1": [10, 15, 20, 25],

            "Value2": [20, 25, 30, 35]

        }

        df = pd.DataFrame(sample\_data)

        df.to\_csv(self.filename, index=False)

        print(f"Створено новий CSV-файл із базовими даними: {self.filename}")

    def load\_data(self):

        try:

            self.data = pd.read\_csv(self.filename)

            print("Дані завантажено успішно.")

        except Exception as e:

            print(f"Помилка завантаження даних: {e}")

    def display\_data\_summary(self):

        if self.data is not None:

            print("Огляд даних:\n", self.data.describe())

        else:

            print("Дані ще не завантажені.")

    def plot\_data(self):

        if self.data is not None:

            plt.plot(self.data["Date"], self.data["Value1"], label="Value1")

            plt.plot(self.data["Date"], self.data["Value2"], label="Value2")

            plt.xlabel("Date")

            plt.ylabel("Values")

            plt.legend()

            plt.title("Basic Plot of Value1 and Value2 Over Time")

            plt.show()

        else:

            print("Дані ще не завантажені.")

    def plot\_multiple\_subplots(self):

        if self.data is not None and len(self.data.columns) > 2:

            fig, axs = plt.subplots(1, 2, figsize=(10, 5))

            axs[0].plot(self.data["Date"], self.data["Value1"], color='b')

            axs[0].set\_title("Value1 Over Time")

            axs[1].plot(self.data["Date"], self.data["Value2"], color='r')

            axs[1].set\_title("Value2 Over Time")

            plt.show()

        else:

            print("Необхідно більше стовпців для багатопанельної діаграми.")

# Використання класу

visualizer = CSVVisualizer("ваш\_файл.csv")

visualizer.display\_data\_summary()

visualizer.plot\_data()

visualizer.plot\_multiple\_subplots()

**Результати тестування:**

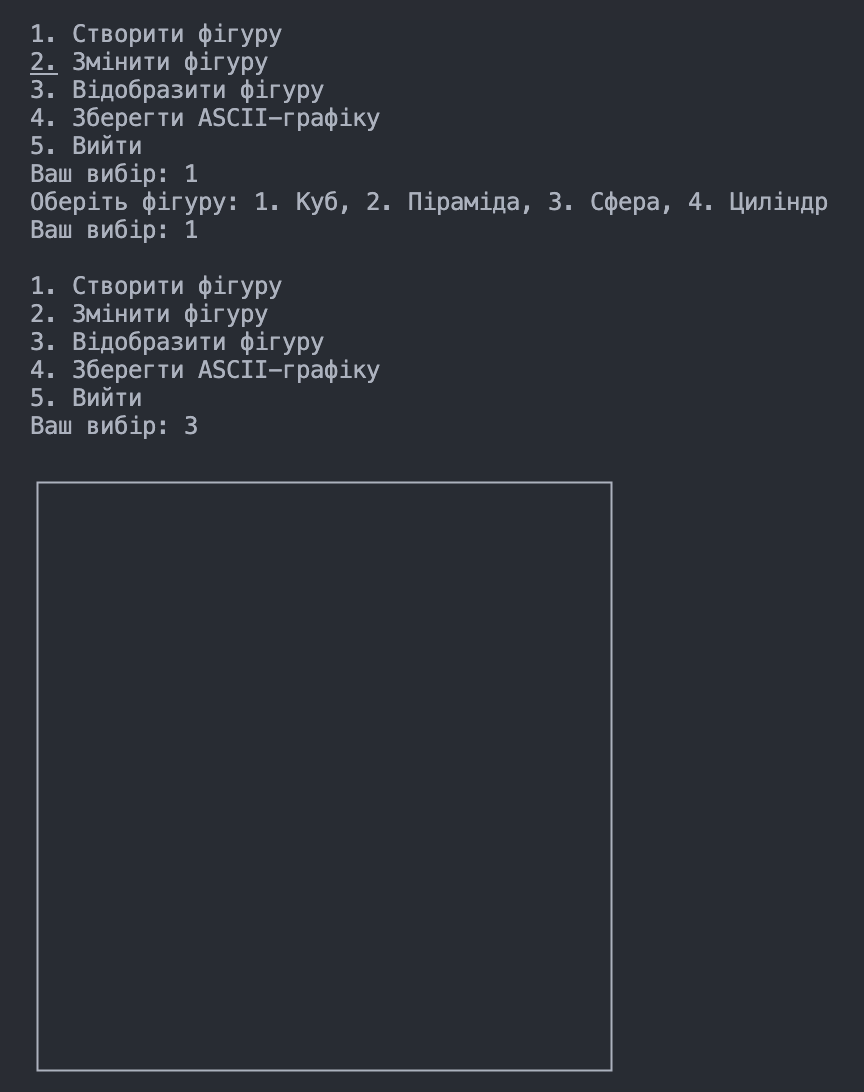


Рис. 1. Результат виведення кубу

**Висновки:** на цій лабораторній роботі був створений додаток для малювання 3D-фігур у ASCII-арті на основі об’єктно - орієнтованого підходу та мови Python.