мІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

нАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «лЬВІВСЬКА ПОЛІТЕХНІКА»

ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНОРМАЦІЙНИХ ТЕХНОЛОГІЙ

|  |
| --- |
|  |



ЗВІТ

Про виконання лабораторної роботи № 5

З дисципліни

«Спеціалізовані мови програмування»

на тему «Розробка ASCII ART генератора для візуалізації 3D-фігур»

Виконала:

студентка гр. ІТ-31

Проців Р.В.

Прийняв:

доц. каф. ІСМ

Щербак С.С.

Львів – 2023

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

**Хід 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.

**Програмний код**:

from figure import \*

def get\_character\_input():

while True:

character = input("Enter a character to represent in the shape: ")

if Figure3D.is\_appropriate\_character(character) is False:

print("You should have entered one character!")

else:

return character

def get\_color\_position\_input():

while True:

try:

color = int(input("Enter a number of color: "))

if color not in range(len(colors)):

print("You should have entered a color option which is available!")

else:

return color

except ValueError:

print("You should have entered an integer number!")

def get\_length\_input():

while True:

try:

length = int(input("Enter a length: "))

if length <= 0:

print("You should have entered a length greater than 0!")

else:

return length

except ValueError:

print("You should have entered an integer number!")

def get\_scale\_input():

while True:

try:

scale = float(input("Enter a scale for figure: "))

if scale <= 0:

print("You should have entered a scale greater than 0!")

else:

return scale

except ValueError:

print("You should have entered a float number!")

representation\_3d\_file = "cube.txt"

def main():

is\_figure\_available: bool = False

is\_3d\_representation\_available = False

while True:

print("1 - Create a cube")

print("2 - Display 2D")

print("3 - Save 3D")

print("0 - Exit")

option = str(input("Enter an option: "))

match option:

case "1":

character = get\_character\_input()

print("There are such colors available:")

display\_colors()

color\_position = get\_color\_position\_input()

length = get\_length\_input()

scale = get\_scale\_input()

try:

figure = Cube(length, character, color\_position)

is\_figure\_available = True

representation\_3d = figure.get\_3d\_representation(scale=scale)

print(representation\_3d)

is\_3d\_representation\_available = True

except ValueError as e:

print(e)

is\_figure\_available = False

case "2":

if is\_figure\_available is True:

representation\_2d = figure.get\_2d\_representation()

[print(item) for item in representation\_2d]

else:

print("There is no figure available!")

case "3":

if is\_3d\_representation\_available is True:

try:

with open(representation\_3d\_file, "w") as file:

file.write(representation\_3d)

except PermissionError:

print("You do not have permission to write to the file!")

except FileNotFoundError:

print("The file does not exist!")

else:

print("There is no figure available!")

case "0":

break

case \_:

print("Invalid option!")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**GitHub Repository:** <https://github.com/fxxwol/python/tree/main/lab_5>

**Висновок:**  в результаті даної лабораторної роботи було створено високорівневий об'єктно-орієнтований генератор 3D ASCII-арту, який дозволить користувачам проектувати, відображати та маніпулювати 3D-фігурами в ASCII-арті.