**Assignment 2**

# Student Name:

Olesia Mykhailyshyn

# Group:

ОПД 2

# Task:

# This project is a console-based application for drawing various geometric shapes on a grid-based blackboard. The main goal is to implement a system that uses polymorphism and inheritance to support drawing multiple types of shapes while providing an interactive command-line interface (CLI) to manipulate these shapes. The application supports creating, listing, and managing shapes such as triangles, rectangles, circles, and lines.

This task leverages inheritance and polymorphism to create a flexible and extensible system where new shapes can be easily added by extending the Figure class. The CLI provides a simple interface for users to interact with the application.

# System Model:

# Зображення, що містить текст, почерк, біла дошка, малюнок Автоматично згенерований опис

# GitHub Link:

# <https://github.com/olesia-mykhailyshyn/Assignment_2_OOP.git>

# Solution Description:

The implemented software is a simple drawing application designed for drawing geometric shapes on a grid-based board using ASCII characters. It allows users to add, manipulate, and display shapes such as triangles, rectangles, circles, and lines through a command-line interface (CLI).

**Figure Class Hierarchy**: Defines a base class Figure and derived classes for each shape type (Triangle, Rectangle, Circle, and Line).

**Board Class**: Represents a grid of characters (Board) where shapes are drawn and displayed.

**CLI Class**: Provides a user interface for interacting with the board and shapes through text commands.

**Command Handling and User Interaction**:

The main application (main.cpp) initializes a CLI object and enters an infinite loop to continuously read and execute commands from the user.

Depending on the command, the corresponding method in CLI is called to perform the required operation.

For example, when a user enters the add command, the system prompts for additional parameters such as shape type, coordinates, and dimensions. Based on the shape type, a corresponding shape object is created and added to the board​(main).

**File Operations for Saving and Loading**:

The system supports saving the current board state (along with all shapes) to a text file and loading it back. Each shape has a getSaveFormat() method that returns a string in a specific format to represent its state.

The CLI::save and CLI::load methods handle file I/O, ensuring that the board can be persisted between sessions​(CLI).

# Testing:

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить знімок екрана, чорний, дизайн

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить знімок екрана, чорний, дизайн

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить знімок екрана, текст, дизайн

Автоматично згенерований опис

Зображення, що містить текст, Шрифт, знімок екрана

Автоматично згенерований опис

Зображення, що містить знімок екрана, Мультимедійне програмне забезпечення, текст, програмне забезпечення

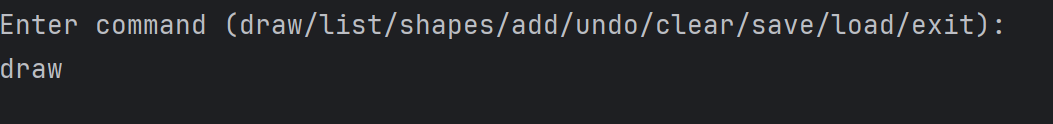
Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, програмне забезпечення, Мультимедійне програмне забезпечення

Автоматично згенерований опис



Зображення, що містить текст, знімок екрана

Автоматично згенерований опис



Зображення, що містить текст, знімок екрана, програмне забезпечення, Мультимедійне програмне забезпечення

Автоматично згенерований опис

Зображення, що містить знімок екрана, темрява, чорний, феєрверк

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

# Conclusions:

 **What was implemented**:

* Developed a console-based drawing application to render various shapes (triangles, rectangles, circles, and lines) on a grid board.
* Implemented OOP principles using inheritance and polymorphism.
* Utilized shared\_ptr for safe memory management of figures.
* Used friendship between classes to facilitate direct access to internal properties, gaining a deeper understanding of practical friend class usage.

 **Differences from the proposed model**:

* Implemented some custom validations for shape parameters that were not defined in the original model.
* Adjusted file loading and error-handling mechanisms based on real usage scenarios.

 **Additional Notes**:

* The most challenging part was implementing the drawing algorithms for different shapes, as precise logic was required.
* Relied on external resources like YouTube tutorials and algorithm websites to understand and implement shape rendering.
* Friendship between classes simplified access to private members, enhancing the flexibility of the solution.

# Appendices:

<https://www.youtube.com/watch?time_continue=423&v=S_isDjezoz8&embeds_referring_euri=https%3A%2F%2Fwww.google.com%2Fsearch%3Fq%3Dhow%2Bto%2Bdraw%2Brectangle%2Bc%252B%252B%2Bby%2Bcoordinates%26oq%3Dhow%2Bto%2Bdraw%2Brectangle%2Bc%252B%252B%2Bby%2Bcoordinates%26gs_&source_ve_path=MTM5MTE3LDEzOTExNywyMzg1MQ> --- drawing the rectangle

<https://www.geeksforgeeks.org/program-print-circle-pattern/> --- circle

<https://www.youtube.com/watch?v=HaZh4SVCXyg> – circle

<https://qna.habr.com/q/574775> -- line