Programming Technology

Name: Murad Huseynov

Neptun ID: RX15MW

Second Assignment: TASK 10.

**Exercise Description**

**Rubic clock**

Create a game, which implements the Rubik clock. In this game there are 9 clocks. Each clock can show a time between 1 and 12 (hour only). Clocks are placed in a 3x3 grid, and initially they set randomly. Each four clocks on a corner has a button placed between them, so we have four buttons in total. Pressing a button increase the hour on the four adjacent clocks by one. The player wins, if all the clocks show 12.

Implement the game, and let the player restart it. The game should recognize if it is ended, and it has to show in a message box how much steps did it take to solve the game. After this, a new game should be started automatically.

**Class Diagram**

A screenshot of a computer

Description automatically generated

**Method Description**

1. getBoundingArea(): double 🡺 an abstract method declared in the abstract class ‘Shape’ used for calculating the bounding box area. For each shape, the method has been implemented in the relative class. The bounding box areas of each shape:

* Circle = length2
* Square = length2
* Regular Triangle = (length2)
* Regular Hexagon = (length2)

1. readFile(filename: String): ArrayList<Shape> 🡺 this static method was used for reading text files in order to test the success/error cases.
2. greatestArea(): String 🡺 this method finds and returns the type of the shape which has the greatest bounding box area.
3. toString(): String 🡺 Overridden method in order to display the properties of shapes.
4. getType(): String 🡺 returns the type of a shape.
5. getX(): double 🡺 returns the ‘x’ coordinate of a shape.
6. getY(): double 🡺 returns the ‘y’ coordinate of a shape.
7. getLength(): double 🡺 returns the side length/radius of a shape.