Project Sprint #0

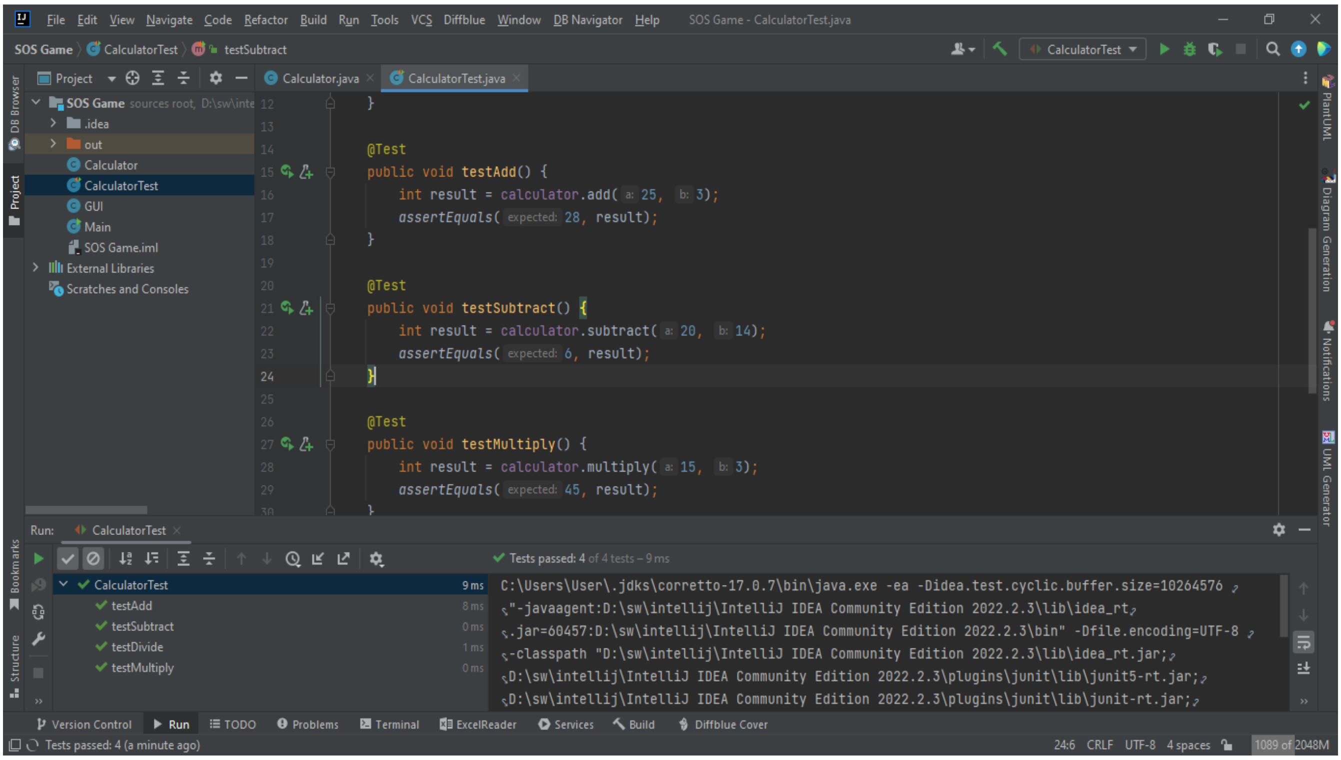
1. Key Decisions of the SOS Project

Object-oriented programming language	Java
GUI library (strongly encouraged)	Swing
IDE (Integrated Development Environment)	IntelliJ IDEA
xUnit framework (e.g., JUnit for Java)	JUnit
Programming style guide (must read it carefully)	Google Java Style Guide
Project hosting site	GitHub
Other decisions if applicable	-

2. Unit testing :

Find a tutorial on the unit test framework you have chosen and write at least two xUnit tests of a

program you have written or found elsewhere. Attach here (1) the screenshot of program execution:



```java
    }

    @Test
    public void testAdd() {
        int result = calculator.add(25, 3);
        assertEquals(28, result);
    }

    @Test
    public void testSubtract() {
        int result = calculator.subtract(20, 14);
        assertEquals(6, result);
    }

    @Test
    public void testMultiply() {
        int result = calculator.multiply(15, 3);
        assertEquals(45, result);
    }
```

```
✓ Tests passed: 4 of 4 tests – 9 ms

✓ CalculatorTest    9 ms
  ✓ testAdd         8 ms
  ✓ testSubtract    0 ms
  ✓ testDivide      1 ms
  ✓ testMultiply    0 ms

C:\Users\User\.jdks\corretto-17.0.7\bin\java.exe -ea -Didea.test.cyclic.buffer.size=10264576
"-javaagent:D:\sw\intellij\IntelliJ IDEA Community Edition 2022.2.3\lib\idea_rt
.jar=60457:D:\sw\intellij\IntelliJ IDEA Community Edition 2022.2.3\bin" -Dfile.encoding=UTF-8
-classpath "D:\sw\intellij\IntelliJ IDEA Community Edition 2022.2.3\lib\idea_rt.jar;
D:\sw\intellij\IntelliJ IDEA Community Edition 2022.2.3\plugins\junit\lib\junit5-rt.jar;
D:\sw\intellij\IntelliJ IDEA Community Edition 2022.2.3\plugins\junit\lib\junit-rt.jar;

Tests passed: 4 (a minute ago)
```

import org.junit.Before;

```
    }

import org.junit.Test;

import static org.junit.Assert.assertEquals;

public class CalculatorTest {
    private Calculator calculator;

    @Before
    public void setUp() {
        calculator = new Calculator();
    }

    @Test
    public void testAdd() {
        int result = calculator.add(25, 3);
        assertEquals(28, result);
    }

    @Test
    public void testSubtract() {
        int result = calculator.subtract(20, 14);
        assertEquals(6, result);
    }

    @Test
    public void testMultiply() {
        int result = calculator.multiply(15, 3);
        assertEquals(45, result);
    }

    @Test
    public void testDivide() {
        int result = calculator.divide(6, 3);
        assertEquals(2, result);
    }
}
```

*The source code of your program:

```
public class Calculator {
    public int add(int a, int b) {
        return a + b;
    }

    public int subtract(int a, int b) {
        return a - b;
    }

    public int multiply(int a, int b) {
        return a * b;
    }

    public int divide(int a, int b) {
        return a / b;
    }
}
```

3. GUI programming

GUI Program in Java Language

```
import javax.swing.*;
import javax.swing.border.EmptyBorder;
import java.awt.*;
```

```
public class GUI extends JFrame {


    JRadioButton redORadioButton;
    JRadioButton redSRadioButton;
    JRadioButton blueORadioButton;
    JRadioButton blueSRadioButton;
    JPanel bluePlayerEmptyPanel;
    JPanel bluePlayerRadioPanel;
    JPanel redPlayerRadioPanel;
    JPanel mainPanel;
    JPanel topPanel;

    JLabel sosLabel;
    JLabel boardSizeLabel;
    JTextField boardSizeTextField;
    JRadioButton simpleGameRadioButton;
    JRadioButton generalGameRadioButton;
    JPanel centerPanel;
    JPanel bluePlayerPanel;
    JLabel bluePlayerLabel;
    JPanel redPlayerPanel;
    JPanel redPlayerEmptyPanel;
    JLabel redPlayerLabel;
    JPanel boardPanel;

    int boardSize = 8;

    JPanel bottomPanel;
    JLabel currentTurnLabel;
```

```
JButton newGameButton;

public GUI() {

    setTitle("SOS Game");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setPreferredSize(new Dimension(700, 650));

    mainPanel = new JPanel(new BorderLayout());

    topPanel = new JPanel(new FlowLayout());
    sosLabel = new JLabel("SOS");
    simpleGameRadioButton = new JRadioButton("Simple Game");
    generalGameRadioButton = new JRadioButton("General Game");

    ButtonGroup gameTypeGroup = new ButtonGroup();
    gameTypeGroup.add(simpleGameRadioButton);
    gameTypeGroup.add(generalGameRadioButton);

    boardSizeLabel = new JLabel("Board Size:");
    boardSizeLabel.setBorder(new EmptyBorder(0, 100, 0, 0));
    boardSizeTextField = new JTextField(String.valueOf(boardSize));
    boardSizeTextField.setColumns(2);

    JSeparator line1 = new JSeparator(JSeparator.HORIZONTAL);
    line1.setPreferredSize(new Dimension(100, 2));
    topPanel.add(line1);

    topPanel.add(sosLabel);
    topPanel.add(simpleGameRadioButton);
    topPanel.add(generalGameRadioButton);
    topPanel.add(boardSizeLabel);
topPanel.add(boardSizeTextField);

JSeparator line2 = new JSeparator(JSeparator.HORIZONTAL);
line2.setPreferredSize(new Dimension(100, 2));
topPanel.add(line2);

topPanel.setBorder(new EmptyBorder(10, 10, 10, 10));


bluePlayerPanel = new JPanel(new BorderLayout());
bluePlayerLabel = new JLabel("Blue Player");
bluePlayerLabel.setBorder(new EmptyBorder(40, 20, 10, 20));
bluePlayerRadioPanel = new JPanel(new BorderLayout());
bluePlayerRadioPanel.setBorder(new EmptyBorder(40, 20, 10, 20));
bluePlayerPanel.add(bluePlayerLabel, BorderLayout.NORTH);

blueSRadioButton = new JRadioButton("S");
blueORadioButton = new JRadioButton("O");
bluePlayerRadioPanel.add(blueSRadioButton, BorderLayout.NORTH);
bluePlayerRadioPanel.add(blueORadioButton, BorderLayout.SOUTH);
ButtonGroup bluePlayerRadioGroup = new ButtonGroup();
bluePlayerRadioGroup.add(blueSRadioButton);
bluePlayerRadioGroup.add(blueORadioButton);
bluePlayerPanel.add(bluePlayerRadioPanel, BorderLayout.CENTER);

bluePlayerEmptyPanel = new JPanel();
bluePlayerEmptyPanel.setPreferredSize(new Dimension(50, 300));
bluePlayerPanel.add(bluePlayerEmptyPanel, BorderLayout.SOUTH);


redPlayerPanel = new JPanel(new BorderLayout());
redPlayerLabel = new JLabel("Red Player");
```

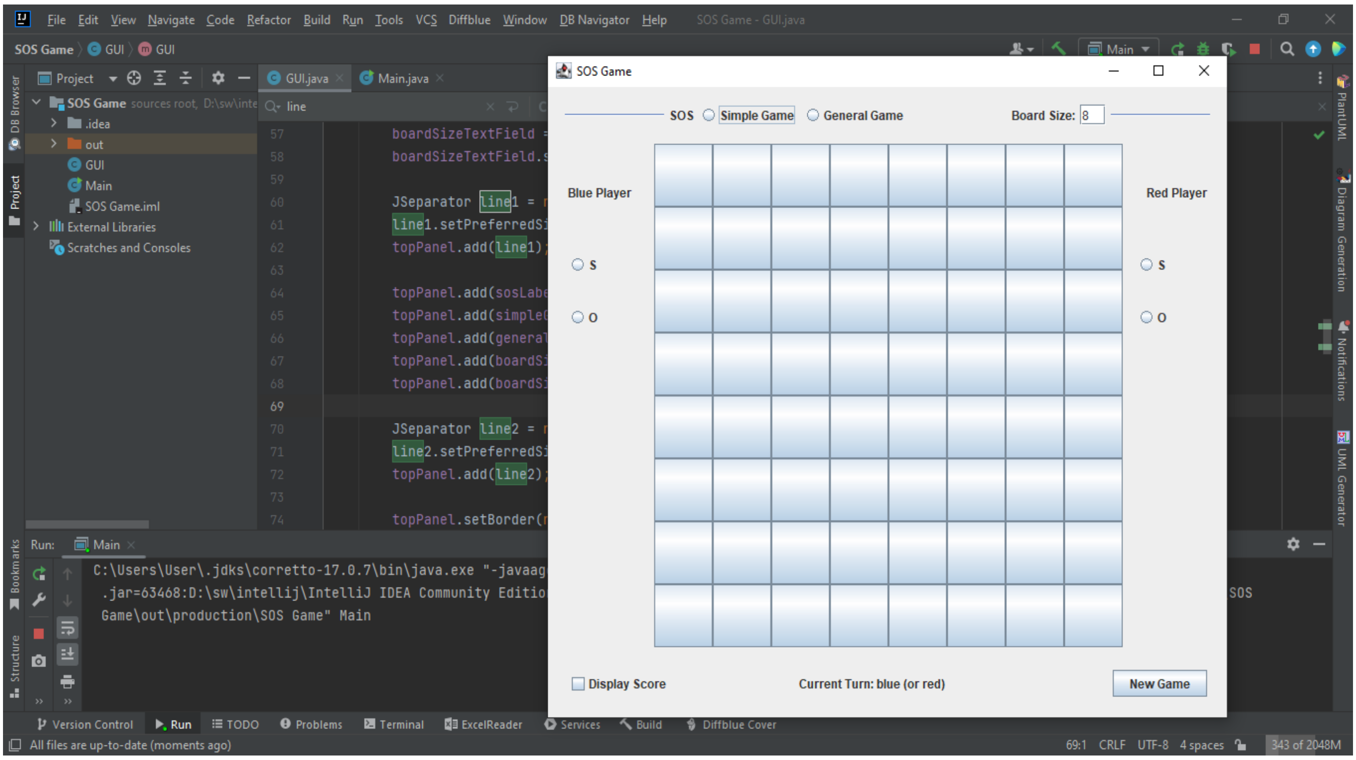
```
redPlayerLabel.setBorder(new EmptyBorder(40, 20, 10, 20));
redPlayerPanel.add(redPlayerLabel, BorderLayout.NORTH);

redPlayerRadioPanel = new JPanel(new BorderLayout());
redPlayerRadioPanel.setBorder(new EmptyBorder(40, 10, 10, 20));
redSRadioButton = new JRadioButton("S");
redORadioButton = new JRadioButton("O");
ButtonGroup redPlayerRadioGroup = new ButtonGroup();
redPlayerRadioGroup.add(redSRadioButton);
redPlayerRadioGroup.add(redORadioButton);
redPlayerRadioPanel.add(redSRadioButton, BorderLayout.NORTH);
redPlayerRadioPanel.add(redORadioButton, BorderLayout.SOUTH);
redPlayerPanel.add(redPlayerRadioPanel, BorderLayout.CENTER);

redPlayerEmptyPanel = new JPanel();
redPlayerEmptyPanel.setPreferredSize(new Dimension(50, 300));
redPlayerPanel.add(redPlayerEmptyPanel, BorderLayout.SOUTH);

boardPanel = new JPanel(new GridLayout(boardSize, boardSize));

for (int i = 0; i < (boardSize * boardSize); i++) {
   JButton button = new JButton("");
   boardPanel.add(button);
}

centerPanel = new JPanel(new BorderLayout());
centerPanel.add(bluePlayerPanel, BorderLayout.WEST);
centerPanel.add(boardPanel, BorderLayout.CENTER);
centerPanel.add(redPlayerPanel, BorderLayout.EAST);



bottomPanel = new JPanel(new BorderLayout());
```

```
    JCheckBox checkBox = new JCheckBox("Display Score");
    bottomPanel.add(checkBox, BorderLayout.WEST);

    currentTurnLabel = new JLabel("Current Turn: blue (or red)");
    currentTurnLabel.setBorder(new EmptyBorder(0, 130, 0, 0));
    newGameButton = new JButton("New Game");

    bottomPanel.add(currentTurnLabel, BorderLayout.CENTER);
    bottomPanel.add(newGameButton, BorderLayout.EAST);
    bottomPanel.setBorder(new EmptyBorder(20, 20, 20, 20));

    mainPanel.add(topPanel, BorderLayout.NORTH);
    mainPanel.add(centerPanel, BorderLayout.CENTER);
    mainPanel.add(bottomPanel, BorderLayout.SOUTH);


    getContentPane().add(mainPanel);
    pack();
    setLocationRelativeTo(null);

  }
}
```

Main:

```
public class Main {
  public static void main(String[] args) {

    GUI gui = new GUI();
    gui.setVisible(true);
    }
}
```

The screenshot of program execution:



```
        boardSizeTextField =
        boardSizeTextField.s

        JSeparator line1 =
        line1.setPreferredSi
        topPanel.add(line1);

        topPanel.add(sosLabe
        topPanel.add(simpleG
        topPanel.add(general
        topPanel.add(boardSi
        topPanel.add(boardSi

        JSeparator line2 =
        line2.setPreferredSi
        topPanel.add(line2);

        topPanel.setBorder(
```

```
C:\Users\User\.jdks\corretto-17.0.7\bin\java.exe "-javaag
 .jar=63468:D:\sw\intellij\IntelliJ IDEA Community Editio                    SOS
 Game\out\production\SOS Game" Main
```

SOS ○ Simple Game ○ General Game Board Size: 8

Blue Player ○ S ○ O

Red Player ○ S ○ O

□ Display Score

Current Turn: blue (or red)