

Lab Final Question (Object Oriented Programming)

Q1. Write a C++/Java program to create a function that returns a pointer to an array of integers.

C++

```
#include <iostream>
using namespace std;
int* createArray(int size) {
    int* arr = new int[size];
    for (int i = 0; i < size; i++) {
        arr[i] = i + 1;
    }
    return arr;
}
int main() {
    int size;
    cin >> size;
    int* arr = createArray(size);
    cout << "Array elements: ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    delete[] arr;
    return 0;
}
```

Q2. Write a C++/Java program that demonstrates the implementation of a parameterized constructor.

C++

```
#include <iostream>

using namespace std;

class Student {
    string name;
    int age;
public:
    Student(string n, int a) : name(n), age(a) {}
    void display() {
        cout << "Name: " << name << ", Age: " << age;
    }
};

int main() {
    Student s("Alice", 20);
    s.display();
    return 0;
}
```

3. Write a C++/Java program using a class and method to calculate the sum of three user-input numbers.

C++

```
#include <iostream>

using namespace std;

class SumCalculator {
```

public:

```
int calculateSum(int a, int b, int c) {  
    return a + b + c;  
}  
  
void getInputAndCalculate() {  
    int num1, num2, num3;  
    cout << "Enter three numbers: ";  
    cin >> num1 >> num2 >> num3;  
    int sum = calculateSum(num1, num2, num3);  
    cout << "Sum of the three numbers is: " << sum << endl;  
}
```

};

```
int main() {  
    SumCalculator calculator;  
    calculator.getInputAndCalculate();  
    return 0;  
}
```

4. Write a C++/Java program to calculate the area of the rectangular room using method overloading.

C++

```
#include <iostream>  
  
using namespace std;  
  
class Area {  
public:  
    int calculate(int length, int width) {
```

```

        return length * width;
    }
    int calculate(int side) {
        return side * side;
    }
};

int main() {
    Area a;
    cout << "Area (rect): " << a.calculate(4, 5) << endl;
    cout << "Area (square): " << a.calculate(5);
    return 0;
}

```

5. Write a C++ /Java program to calculate area and volume of any room using single inheritance.

C++

```

#include <iostream>
using namespace std;
class Area {
public:
    int calculateArea(int l, int w) {
        return l * w;
    }
};

class Volume : public Area {
public:

```

```

    int calculateVolume(int l, int w, int h) {
        return calculateArea(l, w) * h;
    }
};

int main() {
    Volume v;
    cout << "Volume: " << v.calculateVolume(2, 3, 4);
    return 0;
}

```

6. Write Java program to use method overriding.

```

class Animal {
    void sound() {
        System.out.println("Animal makes a sound");
    }
}

class Dog extends Animal {
    @Override
    void sound() {
        System.out.println("Dog barks");
    }

    public static void main(String[] args) {
        Animal a = new Dog();
        a.sound();
    }
}

```

```
}
```

7. Write a Java program for adding applet to HTML file.

Java Applet

```
import java.applet.Applet;  
import java.awt.Graphics;  
public class HelloApplet extends Applet {  
    public void paint(Graphics g) {  
        g.drawString("Hello, World!", 20, 20);  
    }  
}
```

HTML File

```
<applet code="HelloApplet.class" width="300" height="200"></applet>
```

8. Write a Java program that demonstrates the implementation of a interface.

```
interface Drawable {  
    void draw();  
}  
class Circle implements Drawable {  
    public void draw() {  
        System.out.println("Drawing a circle");  
    }  
    public static void main(String[] args) {  
        Drawable d = new Circle();  
        d.draw();  
    }  
}
```

```
}  
}
```

9. Write a C++/Java program to concatenate two strings entered by the user.

C++

```
#include <iostream>  
using namespace std;  
int main() {  
    string s1, s2;  
    cout << "Enter two strings: ";  
    cin >> s1 >> s2;  
    cout << "Result: " << s1 + s2;  
    return 0;  
}
```

10. Write a Java GUI program to convert Celsius temperature into Fahrenheit using a button. When the button is clicked, the program should convert the temperature to Fahrenheit and display the result in a label.

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
public class CelsiusToFahrenheitConverter extends JFrame {  
    private JTextField celsiusField;  
    private JLabel resultLabel;
```

```
public CelsiusToFahrenheitConverter() {
    setTitle("Celsius to Fahrenheit Converter");
    setSize(350, 200);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setLayout(new GridLayout(3, 2, 10, 10));
    JLabel celsiusLabel = new JLabel("Enter Celsius:");
    celsiusField = new JTextField();
    JButton convertButton = new JButton("Convert");
    resultLabel = new JLabel("Fahrenheit: ");
    convertButton.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            convertTemperature();
        }
    });
    add(celsiusLabel);
    add(celsiusField);
    add(convertButton);
    add(resultLabel);
    setLocationRelativeTo(null);
    setVisible(true);
}

private void convertTemperature() {
    try {
```



```

        double celsius = Double.parseDouble(celsiusField.getText());
        double fahrenheit = (celsius * 9 / 5) + 32;
        resultLabel.setText("Fahrenheit: " + fahrenheit);
    } catch (NumberFormatException e) {
        resultLabel.setText("Invalid Input!");
    }
}

public static void main(String[] args) {
    new CelsiusToFahrenheitConverter();
}
}

```

11. Write a Java swing GUI program to calculate the sum of two numbers entered by the user using text fields, a button, and a label to display the result.

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class SumCalculator extends JFrame {
    private JTextField num1Field, num2Field;
    private JLabel resultLabel;

    public SumCalculator() {
        setTitle("Sum Calculator");
        setSize(350, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}

```

```

setLayout(new GridLayout(4, 2, 10, 10));
JLabel num1Label = new JLabel("Enter first number:");
num1Field = new JTextField();
JLabel num2Label = new JLabel("Enter second number:");
num2Field = new JTextField();
JButton addButton = new JButton("Calculate Sum");
resultLabel = new JLabel("Result: ");
addButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
            double num1 =
Double.parseDouble(num1Field.getText().trim());
            double num2 =
Double.parseDouble(num2Field.getText().trim());
            double sum = num1 + num2;
            resultLabel.setText("Result: " + sum);
        } catch (NumberFormatException ex) {
            resultLabel.setText("Invalid Input!");
        }
    }
});
add(num1Label);
add(num1Field);
add(num2Label);

```

```

        add(num2Field);
        add(addButton);
        add(resultLabel);
        setLocationRelativeTo(null);
        setVisible(true);
    }

    public static void main(String[] args) {
        new SumCalculator();
    }
}

```

12. Write a Java swing GUI program to calculate the subtraction of two numbers entered by the user using text fields, a button, and a label to display the result.

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class SubtractionCalculator extends JFrame {
    private JTextField num1Field, num2Field;
    private JLabel resultLabel;

    public SubtractionCalculator() {
        setTitle("Subtraction Calculator");
        setSize(350, 200);
    }
}

```

```
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
setLayout(new GridLayout(4, 2, 10, 10));
```

```
JLabel num1Label = new JLabel("Enter first number:");  
num1Field = new JTextField();
```

```
JLabel num2Label = new JLabel("Enter second number:");  
num2Field = new JTextField();
```

```
JButton subtractButton = new JButton("Calculate Subtraction");  
resultLabel = new JLabel("Result: ");
```

```
subtractButton.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent e) {  
        try {  
            double num1 =  
Double.parseDouble(num1Field.getText().trim());  
            double num2 =  
Double.parseDouble(num2Field.getText().trim());  
            double subtraction = num1 - num2;  
            resultLabel.setText("Result: " + subtraction);  
        } catch (NumberFormatException ex) {  
            resultLabel.setText("Invalid Input!");  
        }  
    }  
});
```

```

        }
    });

    add(num1Label);
    add(num1Field);
    add(num2Label);
    add(num2Field);
    add(subtractButton);
    add(resultLabel);

    setLocationRelativeTo(null);
    setVisible(true);
}

public static void main(String[] args) {
    new SubtractionCalculator();
}
}

```

13. Write a Java swing GUI program to calculate the multiplication of two numbers entered by the user. Use text fields for input, a button to perform the calculation, and a label to display the result.

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

```

```
public class MultiplicationCalculator extends JFrame {  
    private JTextField num1Field, num2Field;  
    private JLabel resultLabel;  
  
    public MultiplicationCalculator() {  
        setTitle("Multiplication Calculator");  
        setSize(350, 200);  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        setLayout(new GridLayout(4, 2, 10, 10));  
  
        JLabel num1Label = new JLabel("Enter first number:");  
        num1Field = new JTextField();  
  
        JLabel num2Label = new JLabel("Enter second number:");  
        num2Field = new JTextField();  
  
        JButton multiplyButton = new JButton("Calculate Multiplication");  
        resultLabel = new JLabel("Result: ");  
  
        multiplyButton.addActionListener(new ActionListener() {  
            @Override  
            public void actionPerformed(ActionEvent e) {  
                try {
```

```
        double num1 =  
Double.parseDouble(num1Field.getText().trim());  
        double num2 =  
Double.parseDouble(num2Field.getText().trim());  
        double multiplication = num1 * num2;  
        resultLabel.setText("Result: " + multiplication);  
    } catch (NumberFormatException ex) {  
        resultLabel.setText("Invalid Input!");  
    }  
}  
});
```

```
add(num1Label);  
add(num1Field);  
add(num2Label);  
add(num2Field);  
add(multiplyButton);  
add(resultLabel);
```

```
setLocationRelativeTo(null);  
setVisible(true);  
}
```

```
public static void main(String[] args) {  
    new MultiplicationCalculator();  
}
```

```
}  
}
```

14. Write a Java swing GUI program to calculate the division of two numbers entered by the user. Use text fields for input, a button to perform the calculation, and a label to display the result.

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
  
public class DivisionCalculator extends JFrame {  
    private JTextField num1Field, num2Field;  
    private JLabel resultLabel;  
  
    public DivisionCalculator() {  
        setTitle("Division Calculator");  
        setSize(350, 200);  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        setLayout(new GridLayout(4, 2, 10, 10));  
  
        JLabel num1Label = new JLabel("Enter first number:");  
        num1Field = new JTextField();  
  
        JLabel num2Label = new JLabel("Enter second number:");  
        num2Field = new JTextField();
```



```

JButton divideButton = new JButton("Calculate Division");
resultLabel = new JLabel("Result: ");

divideButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
            double num1 =
Double.parseDouble(num1Field.getText().trim());
            double num2 =
Double.parseDouble(num2Field.getText().trim());

            if (num2 == 0) {
                resultLabel.setText("Error: Division by zero!");
            } else {
                double division = num1 / num2;
                resultLabel.setText("Result: " + division);
            }
        } catch (NumberFormatException ex) {
            resultLabel.setText("Invalid Input!");
        }
    }
});
```

```
add(num1Label);  
add(num1Field);  
add(num2Label);  
add(num2Field);  
add(divideButton);  
add(resultLabel);
```

```
setLocationRelativeTo(null);  
setVisible(true);  
}
```

```
public static void main(String[] args) {  
    new DivisionCalculator();  
}  
}
```