Object-Oriented Software Design **Name:** ______________________

Fall 2024

Exam 2

October 14, 2024

Time Limit: 50 Minutes

This exam uses double sided pages with a total of 8 pages (including this cover page) and 11 questions. Total of points is 50.

Grade Table (for grading use only)

Question	Points	Score
1	2	
2	2	
3	4	
4	10	
5	8	
6	5	
7	2	
8	2	
9	2	
10	8	
11	5	
Total:	50	

This page is intentionally blank.

1. (2 points) Which of the following statements about Java interfaces is **FALSE**?

   □ A class can implement multiple interfaces

   □ Interface methods are public by default

   □ An interface can define constructors to initialize its constants.

   □ An Interface can extend another interface

2. (2 points) Define a Java interface called `Shape` that declares a `getArea()` method, which returns the area of the shape as a `double`.

3. (4 points) Define a `Rectangle` class that implements the `Shape` interface you defined in the previous question. Implement only the `Shape` interface in the `Rectangle` class, including the `getArea()` method, where the area is calculated as the product of the rectangle's width and length.

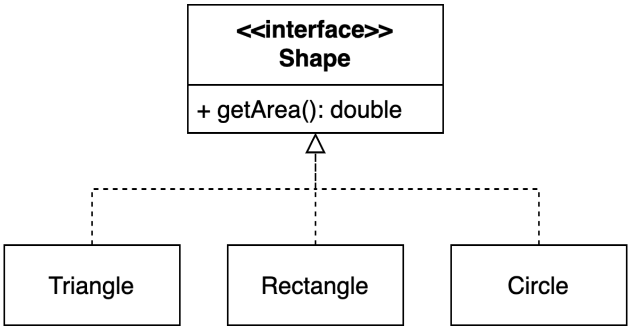

Figure 1: UML Class diagram of various shape classes

4. (10 points) Answer the following questions based on the UML class diagram shown in Figure 1.

(a) (2 points) Which statement demonstrates polymorphism through interfaces?

□ Circle circle = new Circle();
□ Shape shape = new Shape();
□ Shape shape = new Circle();
□ Circle shape = new Shape();

(b) (4 points) Write a static method that returns the amount of paint needed to paint any of the following shapes: `Rectangle, Circle, Triangle`. Assume that it takes 1.5 units of paint to paint 1 unit of area (total paint needed is the area times 1.5).

```
public static double getPaintEstimate(                                ) {







}
```

(c) (4 points) Suppose we defined a new class `Cylinder` as:

```
public class Cylinder {
   // assume necessary class details exist here





}
```

What adjustments do you need to make to the `Cylinder` class such that the `getPaintEstimate` method would work with a `Cylinder` object? Describe the adjustments needed within the `Cylinder` class above.

5. (8 points) The ShoppingCart class defined below supports three different payment methods: CreditCard, ApplePay, and GooglePay. Note the code duplication in the ShoppingCart class, associated with various payment methods. Redesign the solution to remove code duplication by introducing a PaymentMethod interface and refactor the ShoppingCart class to use it. Your redesigned solution should continue to support the existing three payment methods.

```
public class ShoppingCart {
    double totalAmount;

    public void payWithCreditCard(CreditCard paymentMethod) {
        paymentMethod.pay(totalAmount);
    }
    public void payWithApplePay(CreditCard paymentMethod) {
        paymentMethod.pay(totalAmount);
    }
    public void payWithGooglePay(GooglePay paymentMethod) {
        paymentMethod.pay(totalAmount);
    }





}

public class CreditCard {
    public void pay(double amount) {
        // details are irrelevant
    }
}
public class ApplePay {
    public void pay(double amount) {
        //details are irrelevant
    }
}
public class GooglePay {
    public void pay(double amount) {
        // details are irrelevant
    }
}
```

6. (5 points) Consider the following Java code:

```
Collection<String> fruits = new ArrayList<String>();
fruits.add("Apple");
fruits.add("Banana");
fruits.add("Cherry");
```

   (a) (2 points) What interface does `ArrayList` implement that allows it to be assigned to a `Collection` variable?

   (b) (3 points) How does this demonstrate polymorphism?

7. (2 points) Explain the concept of a prototype in the context of software development. How does it differ from a final product?

8. (2 points) What is the purpose of the JPanel class in the Java Swing API?

   □ To create a top-level window

   □ To serve as a container for organizing components

   □ To handle all user input events

   □ To manage the lifecycle and resources of Swing components

9. (2 points) Button click is one of the events that can be handled in Java Swing applications. Name two other types of events (besides a button click).

-
-

10. (8 points) Examine the following Java Swing code:

```
JFrame frame = new JFrame("My App");
JPanel panel = new JPanel();
JButton button = new JButton("Click me");
JLabel greeting = new JLabel("Hello, World!");

panel.add(button);
panel.add(greeting);
frame.add(panel);
frame.setVisible(true);
```

(a) (5 points) Provide a code snippet that demonstrates how to add an event listener (ActionListener) to the ”Click me” button in this GUI.

(b) (3 points) Draw a sketch of the GUI that this code would create. In your sketch, label each visual element (window, button, text) with the corresponding variable name from the code (`frame, panel, button, greeting`). Your sketch should clearly show how these components are nested within each other.

11. (5 points) Suppose you have a class, called `EventPanel`. You are using this class as part of you GUI.

(a) (2 points) The specific usage of that class is demonstrated in the `showEvent` method:

```
1 public void showEvent(Event e, JPanel container)
2 {
3     EventPanel p = new EventPanel(e);
4     container.add(p);
5 }
```

Fill in the blank in the `EventPanel` class definition below, based on how this class is used in the `showEvent` method.

```
public class EventPanel ______________________
{
    Event event;
    public EventPanel(Event e)
    {
       this.event = e;
    }
    // the rest of the class definition is irrelevant
}
```

(b) (3 points) Let's focus on the statement on line 4 of the `showEvent` method. The add method of container object has the following signature:

```
Component add(Component comp);
```

What allows `EventPanel` to be passed as an argument to the add method on line 4?