Programme: **B. Tech**   Discipline: **Computer Science and Engineering**
Exam**: End Semester**   Year: **2024-2025**
Course Code: **CSD213**   Course Title: **Object Oriented Programming**
Date: **November 26, 2024**   Time: **12:00 PM – 02:00 PM**   Max. Marks: **70**

Name: _____   **Roll Number:** _____

Ques.1 Choose the correct answer(s):   **(50 Marks)**

1. **Which of the following is the root class of all exceptions in Java?**
    (a) Error   (b) Exception   (c) Throwable   (d) Runtime Exception

2. **Which of the following classes cannot be caught using a try-catch block in Java?**
    (a) IOException   (b) SQLException   (c) NullPointerException   (d) VirtualMachineError

3. **Which keyword is used to explicitly throw an exception in Java?**
    (a) Catch   (b) throw   (c) throws   (d) throwable

4. **What is the default exception handler for an uncaught exception in a Java program?**
    (a) The program terminates and displays an error message
    (b) The program continues to execute after the exception
    (c) The program is suspended and waits for the user to handle the exception
    (d) The exception is silently ignored

5. **Which of the following is a checked exception?**
    (a) NullPointerException   (c) ArrayIndexOutOfBoundsException
    (b) IOException   (d) ArithmeticException

6. **Which exception class is used for the unchecked exceptions in Java?**
    (a) IOException   (b) Error   (c) RuntimeException   (d) Throwable

7. **Which of the following statements about throw and throws is correct?**
    (a) throw is used to declare exceptions, and throws is used to throw exceptions.
    (b) throw is used to throw exceptions, and throws is used to declare exceptions.
    (c) Both throw and throws are used for throwing exceptions.
    (d) Neither throw nor throws are used to handle exceptions.

8. **Which of the following is the correct way to catch multiple exceptions in a single catch block in Java?**
    (a) catch(Exception | IOException | SQLException e) { // handle exception }
    (b) catch(Exception, IOException, SQLException e) { // handle exception }
    (c) catch(Exception e, IOException e) { // handle exception }
    (d) catch(Exception, IOException | SQLException e) { // handle exception }

9. **Which package is used to create Swing components in Java?**
    (a) java.awt   (b) javax.swing   (c) java.swing   (d) javax.awt

10. **What is the default layout manager for a JFrame?**
    (a) Flow Layout   (b) Border Layout   (c) Grid Layout   (d) None of the above

11. Which method is used to make a JFrame visible?
(a) setVisible(true)　　　　(b) show(true)　　　　(c) setVisible()　　　　(d) display()

12. Which of the following is the base class for all Swing components?
(a) Component　　　　(b) Container　　　　(c) JComponent　　　　(d) JPanel

13. What is the function of JButton in Java Swing?
(a) To display text
(b) To allow users to enter text
(c) To add buttons for actions
(d) To display images

14. Which method in Java Swing is used to set the title of a JFrame?
(a) setTitle()　　　　(b) setWindowTitle()　　　　(c) setHeader()　　　　(d) title()

15. Which Swing component is used for text input?
(a) JTextField　　　　(b) JButton　　　　(c) JLabel　　　　(d) JTextArea

16. Which method is used to add a component to a container in Java Swing?
(a) addComponent()　　　　(b) add()　　　　(c) insert()　　　　(d) put()

17. What is the default layout manager for a JPanel in Swing?
(a) BorderLayout　　　　(b) FlowLayout　　　　(c) GridLayout　　　　(d) BoxLayout

18. Which event listener is used to handle button click events in Swing?
(a) ActionListener　　　　(b) MouseListener　　　　(c) KeyListener　　　　(d) WindowListener

19. What does the JTextArea component in Swing allow a user to do?
(a) Enter a single line of text　　　(b) Enter multiple lines of text　　　(c) Display buttons　　　(d) Display images

20. What is the function of the repaint() method in Java Swing?
(a) To redraw a component
(b) To refresh the entire window
(c) To update the layout of components
(d) To hide a component

21. Which of the following is true about JList in Java Swing?
(a) It allows the user to select multiple items at once.
(b) It displays a list of items with checkboxes.
(c) It is used for displaying images.
(d) It is used to display text fields.

22. How do you set the default close operation for a JFrame to exit the application when closed?
(a) setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE)
(b) setCloseOperation(JFrame.EXIT)
(c) setWindowClose(JFrame.EXIT)
(d) setDefaultClose(JFrame.EXIT_ON_CLOSE)

23. What method in Java Swing is used to set the background color of a component?
(a) setBackgroundColor()　　　　(b) setColor()　　　　(c) setBackground()　　　　(d) changeBackground()

24. Which of the following classes is used for creating a dialog box in Swing?
(a) JOptionPane　　　　(b) JDialog　　　　(c) JOptionBox　　　　(d) DialogBox

25. Which of the following methods can be used to pause the execution of a thread in Java?
(a) sleep()　　　　(b) wait()　　　　(c) join()　　　　(d) All of the above

26. **What is the purpose of the synchronized keyword in Java?**
(a) To prevent thread starvation
(b) To ensure mutual exclusion while accessing shared resources
(c) To increase thread priority
(d) To release resources when a thread is done

27. **What happens if two threads attempt to access a shared resource simultaneously without synchronization?**
 (a) The program may produce incorrect results due to race conditions
 (b) One thread will be blocked automatically
 (c) The program will crash
 (d) Both threads will run without any problems

28. **Which interface must be implemented to handle an event in Java?**
(a) EventListener (b) ActionListener (c) EventHandler (d) Runnable

29. **What is the purpose of the ActionEvent class in Java?**
(a) To represent a key event
(b) To represent an action event (like button click)
(c) To represent a mouse event
(d) To handle window events

30. **Which method is used to register an event listener for a component in Java?**
(a) addActionListener()   (b) setEventListener()   (c) addListener()   (d) registerListener()

31. **Which of the following classes is used to handle mouse events?**
(a) MouseListener   (b) MouseEvent   (c) MouseAdapter   (d) All of the above

32. **Which of the following methods belongs to the MouseListener interface?**
(a) mousePressed()   (b) mouseClicked()   (c) mouseReleased()   (d) All of the above

33. **What is the function of ActionEvent in Java event handling?**
(a) To handle keyboard inputs
(b) To handle mouse movements
(c) To handle user interactions like button clicks
(d) To handle window resizing events

34. **What type of event does the WindowListener interface handle?**
(a) Keyboard events
(b) Mouse events
(c) Window events like opening, closing, etc.
(d) Action events

35. **Which method is called when a component is clicked by the mouse?**
(a) mousePressed()   (b) mouseClicked()   (c) mouseEntered()   (d) mouseReleased()

36. **What happens if an event listener is not registered for a component?**
(a) The event is processed automatically.
(b) The event will not be captured by the component.
(c) The application will crash.
(d) The event is passed to the parent component.

37. **Which interface is used to handle keyboard events in Java?**
(a) KeyListener   (b) KeyEvent   (c) ActionListener   (d) MouseListener

38. **Which method is invoked when a key is pressed on the keyboard?**
(a) keyPressed()   (b) keyReleased()   (c) keyTyped()   (d) keyEvent()

39. **What is the correct order of methods called for a mouse event?**
(a) mousePressed(), mouseReleased(), mouseClicked()
(b) mouseClicked(), mousePressed(), mouseReleased()
(c) mousePressed(), mouseClicked(), mouseReleased()
(d) mouseReleased(), mousePressed(), mouseClicked()

40. **What is the purpose of** MouseAdapter **in Java?**
(a) It is a default listener for mouse events.
(b) It is used to handle key events.
(c) It is a default adapter class for handling mouse events.
(d) It is used to handle focus events.

41. **Which of the following is a correct way to create an event handler for a button in Java?**
(a) Implement ActionListener and override actionPerformed()
(b) Implement ButtonListener and override actionPerformed()
(c) Implement MouseListener and override mouseClicked()
(d) Implement KeyListener and override keyPressed()

42. **In event handling, what is an** EventSource**?**
(a) The event that occurs   (c) The object that listens to the event
(b) The object that generates the event   (d) None of the above

43. **Which method in the** WindowListener **interface is called when a window is closed?**
(a) windowOpened()   (b) windowClosing()   (c) windowClosed()   (d) windowActivated()

44. **What is the purpose of the** FocusListener **interface?**
(a) To listen for mouse events   (c) To listen for focus changes in a component
(b) To listen for keyboard events   (d) To listen for action events

45. **What happens if you call** removeActionListener() **on a button?**
(a) The button will stop responding to clicks.
(b) The button will generate a warning.
(c) The action listener will be detached from the button.
(d) The button will become invisible.

46. **Which of the following is an example of an event in Java?**
(a) Clicking a button   (b) Pressing a key   (c) Moving the mouse   (d) All of the above

47. **What is the return value of "GRAPHIC".substring(0, 5)?**

48. What method do you use to get the mouse-point position for a mouse event?

49. Can an inner class be used in a class other than the class in which it nests?

50. Can the modifiers **public**, **protected**, **private**, and **static** be used for inner classes?

Ques.2 What will be the output of the following Java programs?   **(7 Marks)**

1.
```java
import java.io.*;
class GFG {
        public static void main (String[] args)
        {
                int a=5;
                int b=0;
                try{
                        System.out.println(a/b);
                }
                catch(ArithmeticException e){
                        e.printStackTrace();
                }
        }
}
```

2.
```java
import java.io.*;
class GFG1 {
        public static void main (String[] args)
        {
                int a=5;
                int b=0;
                try{
                        System.out.println(a/b);
                }
                catch(ArithmeticException e){
                System.out.println(e.toString());
                }
        }
}
```

3.
```java
import java.io.*;
class GFG1 {
        public static void main (String[] args)
        {
                int a=5;
                int b=0;
                try{
                System.out.println(a/b);
                }
        catch(ArithmeticException e){
                System.out.println(e.getMessage());
                }        }
        }
```

4.
```java
public class ExceptionHandling extends
endsemException {
void exp()throws Exception{
System.out.println("In Exception Handling");
}
```

```java
public static void main(String args[]){
try{
endsemException e = new ExceptionHandling(); }
catch(ArithmeticException e){
System.out.println(e); }
finally{
System.out.println("PASS"); } }}
class endsemException{
        void exp()throws
ArithmeticException{
System.out.println("In endsemException"); }
}
```

5.
```java
import java.util.*;
class demo2 {
public static void main(String[] args)
{
        Vector v = new Vector(20);
        v.addElement("Geeksforgeeks");
        v.insertElementAt("Java", 2);
        System.out.println(v.firstElement());
        }
}
```

6.
```java
import java.util.*;
class demo3 {
public static void main(String[] args)
{
        Vector v = new Vector(20);
v.addElement("ObjectOrientedProgramming");
        v.insertElementAt("Java", 0);
        System.out.println(v.firstElement());
}
}
```

7.
```java
import java.util.*;
class demo7 {
public static void main(String[] args)
{
        Vector v = new Vector(30);
        v.addElement("Geeksforgeeks");
        v.addElement("Java");
        v.addElement("C++");
        v.addElement("C");
        v.addElement("Geeksforgeeks");
System.out.println(v.indexOf("Geeksforgeeks", 2));
}
}
```

Ques.3 Below given are the commonly used methods of *JLabel* class of Swing. Write down the purpose of each of these methods in front of them: **(4 Marks)**

1. getIcon()
2. setIcon(Icon i)
3. getText()
4. setText(String s)

Ques.4 Below given are the commonly used methods of *JTextField* class of Swing. Write down the purpose of each of these methods in front of them: **(4 Marks)**

1. setColumns(int n)
2. setFont(Font f)
3. addActionListener(ActionListener l)
4. int getColumns()

Ques.5 Explain how to register a handler object and how to implement a handler interface. **(1 Mark)**

Ques.6 What is an event source object? What is an event object? Describe the relationship between an event source object and an event object. **(1 Mark)**

Ques.7 There are multiple errors in the following code. Rectify them and write the output for the same.
(3 Marks)

```
public class EndSemPolyAccess {
public static void main(String args[]){
SpaceHollywood.PrintSpace();
SpaceHollywood sh = new SpaceOdessey();
sh.PrintSpace(12.3f);
SpaceOdessey so = new SpaceOdessey();
so.PrintSpace();
SpaceOdessey so1 = new InterStellar();
so1.PrintSpace(12);
InterStellar I = new InterStellar();
I.PrintSpace(12.3f);
}}
class SpaceHollywood{
static void PrintSpace(){
System.out.println("In SpaceHollywood");
}}
class SpaceOdessey extends SpaceHollywood{
@Override
void PrintSpace(){
System.out.println("In SpaceOdessey");
}
void PrintSpace(Float l){
System.out.println("In SpaceOdessey");
}}
class InterStellar extends SpaceOdessey{
@Override
void PrintSpace(int i){        System.out.println("In InterStellar"+i);        }}
```

****