

CS3391 OBJECT ORIENTED PROGRAMMING

Question Bank

UNIT V EVENT DRIVEN PROGRAMMING

PART – A

1. What is the need of graphics programming?

To provide a Graphical User Environment (GUI) to the java applications Graphics Programming plays a vital role.

2. List the reasons for using strings.

- It has a rich and convenient set of GUI elements.
- It has a small amount of dependencies on the underlying platform;
- It has less number of platform-specific bugs
- It gives a consistent user interface experience across platforms.

3. What is frame?

A frame is a top-level window. It consists of:

- A top part with a little, a little icon and buttons to minimize, maximize and close the window
- An optional menu area
- A content area where we can place buttons, text, drawings etc.

4. How is size a frame?

The pack() methods sizes the frame so that all its contents are at or above their preferred sizes. An alternative to pack is to establish a frame size explicitly by calling set size or set bounds. In general, using pack is preferable to calling set size, since pack leaves the frame layout manager in charge of the frame size, and layout managers are good at adjusting to platform dependencies and other factors that affect component size.

`frame.pack();`

5. How will you show a frame on the screen?

Calling set visible(true) makes the frame appear on screen. Sometimes we might see the show method used instead. The two usages are equivalent, but we use set visible(true) for consistency' sake.

`frame.setVisible(true);`

6. What is the use of valid set default close operation() frames?

If sets operation that will happen by default when the user initiates a close on this frame. Possible values are window constants .DO NOTHING ON CLOSE, window constants. HIDE ON CLOSE, Window constants. DISPOSE ON CLOSE and J frame. EXIT ON CLOSE. `FRAME.setdefaultcloseoperation(JFrame.EXIT-ON-CLOSE);`

7. What is the use of set size() method?

It requires the component so that has width and height.

`frame.set size(1000,700); //1000-width, 700-height`

8. What are components?

A component is an object having a graphical representation that can be displayed on the screen and that can interact with the user. Examples of components are the buttons, check boxes, and scrollbars of a typical graphical user interface.

9. List some graphics class line and shapes drawing methods.

```
void draw line(int x1,int y1,int x2, int y2) //Draws a line from( x1,y1) to (x2,y2)
```

```
void draw dashed line( int d,int x1, int y1, int x2, int y2) // Draws a dashed line from(x1,y1) to (x2,y2) with dash length d
```

```
void draw oval(int x, int y, int w, int h) // Draws an oval with top-left(x,y), width w and height h
```

```
void draw Rect(int x, int y,int w, int h) // Draws a rectangle withtop-left(x,y),width w and height h
```

```
void draw RoundRect(int x, int y,int w, int h, int aw, int ah) // Draws a rectangle with rounded corners with top-left(x,y), width w and height h. corners are arcs with width aw and height ah
```

```
void draw arc(int x, int y, int w, int h, int start, int angle)// Draws parts of an oval with top-left(x,y), width w and height h. Start at angle start and span angle. Angles are measured in degrees.
```

10. What are the constructors available for line2D class?

- public Line2D.Float()
- public Line2D.Float(float x1, float y1, float x2, float y2)
- public Line2D.Float(Point2D p1,Point p2)
- public Line2D.Double()
- public Line2D.Double(float x1, float y1, float x2, float y2)
- public Line2D.Double(Point2D p1, Point2D p2)

11. List the ellipse2D class constructors.

```
public Ellipse2D.Float(float x, float y, float w, float h)
```

```
public Ellipse2D.Double(double x, double y, Double w, double h)
```

12. Write the steps to draw 2D shapes.

1. Call the super class paint components() method with the graphics object passed in, then cast the graphics object to a Graphics2D object:

```
public void paint component(Graphics g)
{
    super.paint component(g);
    Graphics2D g2= (Graphics2D)g;
```

2. Set one or more of the necessary Graphics2D attributes:

```
g2.setPaint(...);
g2.setstroke(...);
g2.setcomposite(...);
g2.setfront(...);
g2.settransform(...);
g2.translate(...);
```

```
g2.rotate(...);  
g2.scale(...);  
g2.shear(...);
```

3. Draw an outlined or solid version of the shape (or both)

```
g2.draw(shape);  
g2.fill(shape);
```

13. What are the seven primary attributes of rendering engine?

- Paint
- Stroke
- Front
- Transformation
- Clipping space
- Rendering hints
- Composting rule

14. What are the logical font families defined by java 2D?

- Dialog
- Dialog input
- Mono spaced
- Serif
- Sans serif

15. How can we extract all the font families installed in the system?

Java software provides access to other fonts that are installed on our system. The names of all available font families can be found by calling the following:

```
Graphics Environment ge = Graphics Environment.getLocalGraphicsEnvironment();  
String [] font Families= ge.getAvailableFontFamilyNames();
```

16. How are the colors represented in Java?

Colors are represented by java. awt . color class. in java 1.0 and java 1.1 this class represents colors in the RGB color space. It has constructors that allow you to specify red ,green, and blue color coordinates as integers or as floating point values. It also defines a number of constants that represent colors by their common names, Such as color. black and color .white

17. List the constants that represents colors.

- Color. black
- Color. blue
- Color. cyan
- Color. dark gray
- Color. gray
- Color. green
- Color. light gray
- Color. magenta
- Color. orange
- Color. pink

- Color. red
- Color. white
- Color. Yellow

18. What is the difference between draw image() and draw scaled image?

- void draw image(Buffered image im ,int x,int y)
- void draw scaled image(Buffered image im, int x, int y, int w, int h)

19. Define event driven programming.

Event driven programming can be defined as an application architecture technique in which the application has a main loop which is clearly divided down to two sections; In embedded systems the same may be achieved using interrupts instead of a constantly running main loop; in that case the former portion of the architecture resides completely in hard ware.

20. What is an Action Object?

An Action is an action listener that provides not only action event handling, but also centralized handling of the state of action event firing components such as tool bar buttons, menu items, common buttons, and text fields. The state that an action can handle includes text, icon ,enabled, and selected status.

21. List the adapter classes for the different event listener interfaces provided by the AWT.

- Component Adapter
- Contains Adapter
- Focus Adapter
- key Adapter
- Mouse Adapter
- Mouse Motion Adapter
- Window Adapter

22. What are methods declared by mouse listener?

```
public abstract void mouse Clicked
public abstract void mouse Pressed
public abstract void mouse Released
public abstract void mouse Entered
public abstract void mouse Exited
```

23. What is a mouse event?

Mouse events notify when the user uses the mouse with a component. Mouse events occur when the cursor enters or exits a component's on the screen area when the user presses or releases one of the mouse buttons.

24. What are the uses of get Mouse button and get Mouse click count methods?

```
static int get Mouse button// Return information about which mouse button was pressed
static int get Mouse click count // Return the number of clicks on the mouse
```

25. Write the child classes of Input Event?

- Key Event
- Mouse Event

26. List the features of swing.

- Pluggable look
- Light weight components
- Do not depend on native peers
- Simplified graphics on native peers to render themselves.
- Similar behavior across all platforms.
- Portable look
- Only a few top level containers not light weight.
- New components- trees tables, sliders progress bars, text components.
- Arbitrary key board
- Debugging support

27. What are models?

Most Swing components have models. A button for example has a model that stores the button state what its keyboard mnemonic is whether it's enabled, selected or pressed. A list for example uses a list model to hold the list contents and a list Selection to track the list's current selection.

28. What is a view?

A very important part of the text package in the view class. as the name suggests it represents a view of the text model, or a piece of the text model. The view is not intended to be some completely new thing that one must learn, but rather is much like a light weight component. In fact the original view implementation was a light weight component. There were several reasons why the component implementation was abandoned in the favor of an alternative.

29. What are the methods available for layouts?

- get minimum span
- public abstract void mouse Clicked
- get preferred span
- get maximum span
- get alignment
- preference changed
- set size

30. What is controller?

The controller is the initial contact point for handling all requests in the system. This controller may delegate to a helper to complete authentication and authorization of a user or to initiate contact retrieval.

31. How will you set default button?

We set default button invoking the set default button method on a top level container's root pane. Here is the code that sets up the default button for the list dialog

Example

```
//In the constructor for a J dialog sub class:
```

```
get root pane() . set default button(set button)
```

32. What are check boxes?

Check boxes are similar to radio buttons but their selection model is different by convention. Any number of check boxes in a group-none, some, or all- can be selected.

33. What are layout managers?

A layout manager is an object that implements the layout manager interface and determines the size position of the components within container. Although components can provide size and alignment hints, a container's layout manager has the final say on the size and position of the components within the container.

PART – B

1. Explain classes under 2D shapes.

2. Explain event handling with examples.

3. Explain action event with an example.

4. What are swing components? Explain.

5. Describe the AWT event hierarchy.

6. Why creating a sub class of frame is preferred over creating an instance of frame when creating a window? Explain.

7. What are the methods supported by the following interfaces? Explain each of them

(a) Action listener interface

(b) Mouse motion listener interface

(c) Text listener interface

8. What are the functionality supported by java related to fonts? Explain how it improves the user interface.

9. What are the methods supported by key listener interface and mouse listener interface?

Explain each of them with examples

10. (a) How event driven programming is different from Procedure oriented programming?

(b) Discuss over view of java event handling mechanism

11. (a) Define graphics context . How do you obtain graphics context?

(b) Explain in brief different drawing functions of java

12. (a) Why creating a sub class of frame is preferred over creating an instance of frame when creating a window.

- (b) Explain the steps in creating a sub class of frame with the help of examples.
13. (a) What is delegation event model? Explain
(b) Describe with example event handling.
14. Explain about Model-View-Controller pattern.
15. Explain about adapter classes with examples.
16. Explain the color and font class in java.
17. Design a reservation system swing components.
18. Discuss in detail about layout design.