



King Fahd University of Petroleum & Minerals
College of Computer Science and Engineering
Information and Computer Science Department
Second Semester 092 (2009/2010)

ICS 201 - Introduction to Computing II

Major Exam 2
Thursday, 13th May, 2010
Time: 120 minutes

Name:

ID#:

Please circle your section number below:

| | | | | |
|--------------|-----------------|-----------------|-----------------|----------------------|
| Section | 01 | 02 | 03 | 04 |
| Instructor | Sami | Tarek | Sukairi | Sukairi |
| Day and Time | SMW 7 - 7:50 | SMW 8 - 8:50 | SMW 9 - 9:50 | SMW 13:00 - 13:50 |

| Question # | Maximum Mark | Obtained Mark |
|--------------|--------------|---------------|
| 1 | 25 | |
| 2 | 10 | |
| 3 | 25 | |
| 4 | 40 | |
| Total | 100 | |

Question 1 [25 Points]:

a) [5 Points] What best describes the appearance of a frame with the following code?

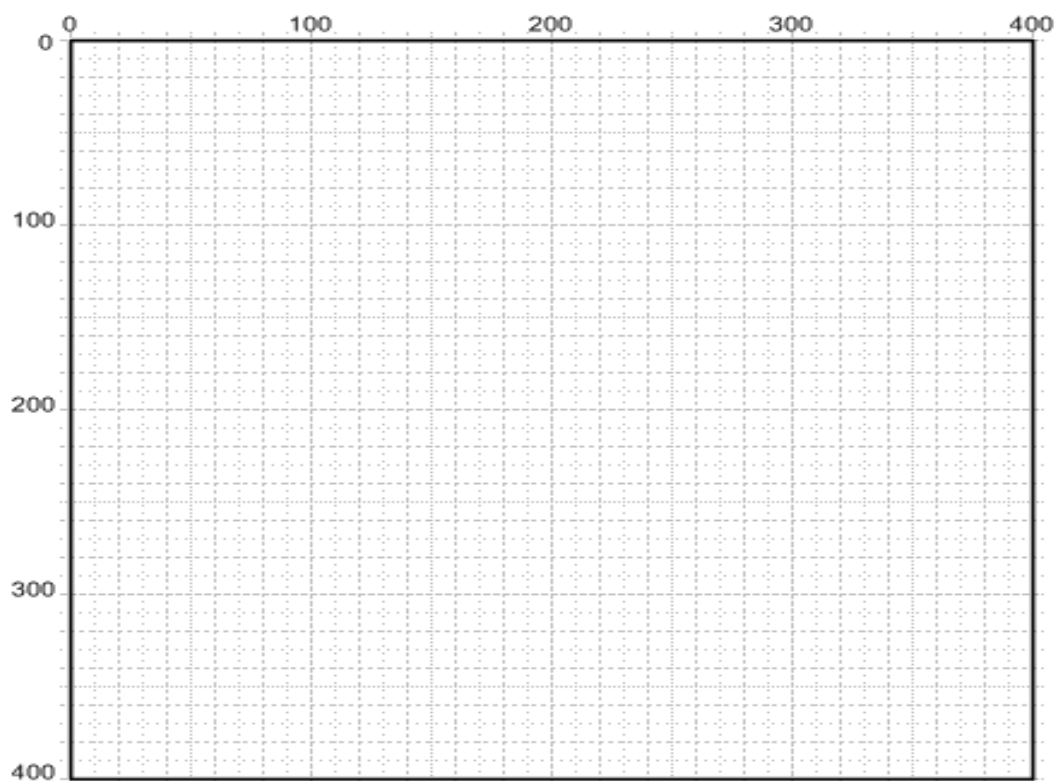
```
import javax.swing.*;
public class FlowAp extends JFrame{
public static void main(String argv[]){
    FlowAp fa=new FlowAp();
    fa.setSize(400,300);
    fa.setVisible(true);
}
FlowAp(){
    add(new JButton("One"));
    add(new JButton("Two"));
    add(new JButton("Three"));
    add(new JButton("Four"));
}
}
```

- a. A Frame with buttons marked One to Four placed on each edge.
- b. A Frame with buttons Marked One to four running from the top to bottom
- c. A Frame with one large button marked Four in the Centre
- d. An Error at run time indicating you have not set a LayoutManager

b) [15 Points] What is approximately the expected outcome of the following java Applet? You can right the expected colour on the shape.

```
import java.applet.*;
import java.awt.*;
public class DrawingStuff extends Applet {
    int width, height;
    public void init() {
        width = getSize().width;
        height = getSize().height;
        setBackground( Color.black );
    }
}
```

```
public void paint( Graphics g ) {  
    g.setColor( Color.red );  
    g.drawRect( 10, 20, 100, 15 );  
    g.setColor( Color.pink );  
    g.fillRect( 240, 160, 40, 110 );  
    g.setColor( Color.blue );  
    g.drawOval( 50, 225, 100, 50 );  
    g.setColor( Color.orange );  
    g.fillOval( 225, 37, 50, 25 );  
    g.setColor( Color.yellow );  
    g.drawArc( 10, 110, 80, 80, 90, 180 );  
    g.setColor( Color.cyan );  
    g.fillArc( 140, 40, 120, 120, 90, 45 );  
    g.setColor( Color.magenta );  
    g.fillArc( 150, 150, 100, 100, 90, 90 );  
    g.setColor( Color.black );  
    g.fillArc( 160, 160, 80, 80, 90, 90 );  
    g.setColor( Color.green );  
    g.drawString( "ICS 201", 50, 150 );  
}}
```



c) [5 Points] Java Virtual Machine (JVM):

- Which part of the JVM holds the information about classes (methods, instance variables, etc.)?
- Where the JVM stores the objects created by a Java program?
- Which entity of the JVM is responsible of managing memory (freeing unused objects, etc.)?

Question 2 [10 Points]

- a. Write a Java statement to create a menu with the title "Cars".

- b. Write Java statements to create menu items named "Toyota", "GMC", and "BMW" and to add them to the menu created in (a).

- c. Write Java statements to
- create a menu bar
 - add the menu created in (b)
 - add the menu bar to the current container.

Question 3 [25 Points] Given the following simple Person class:

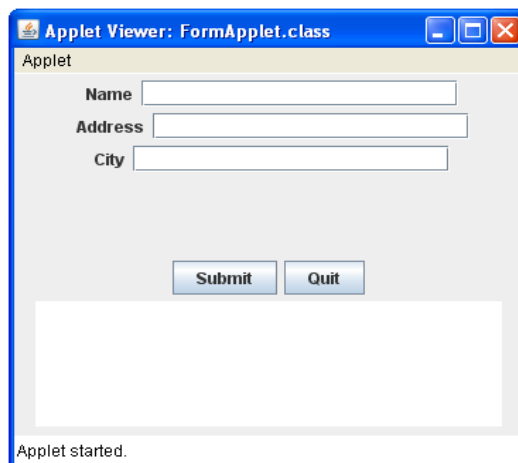
```
public class Person {
    private String name;
    private String address;
    private String cityName;

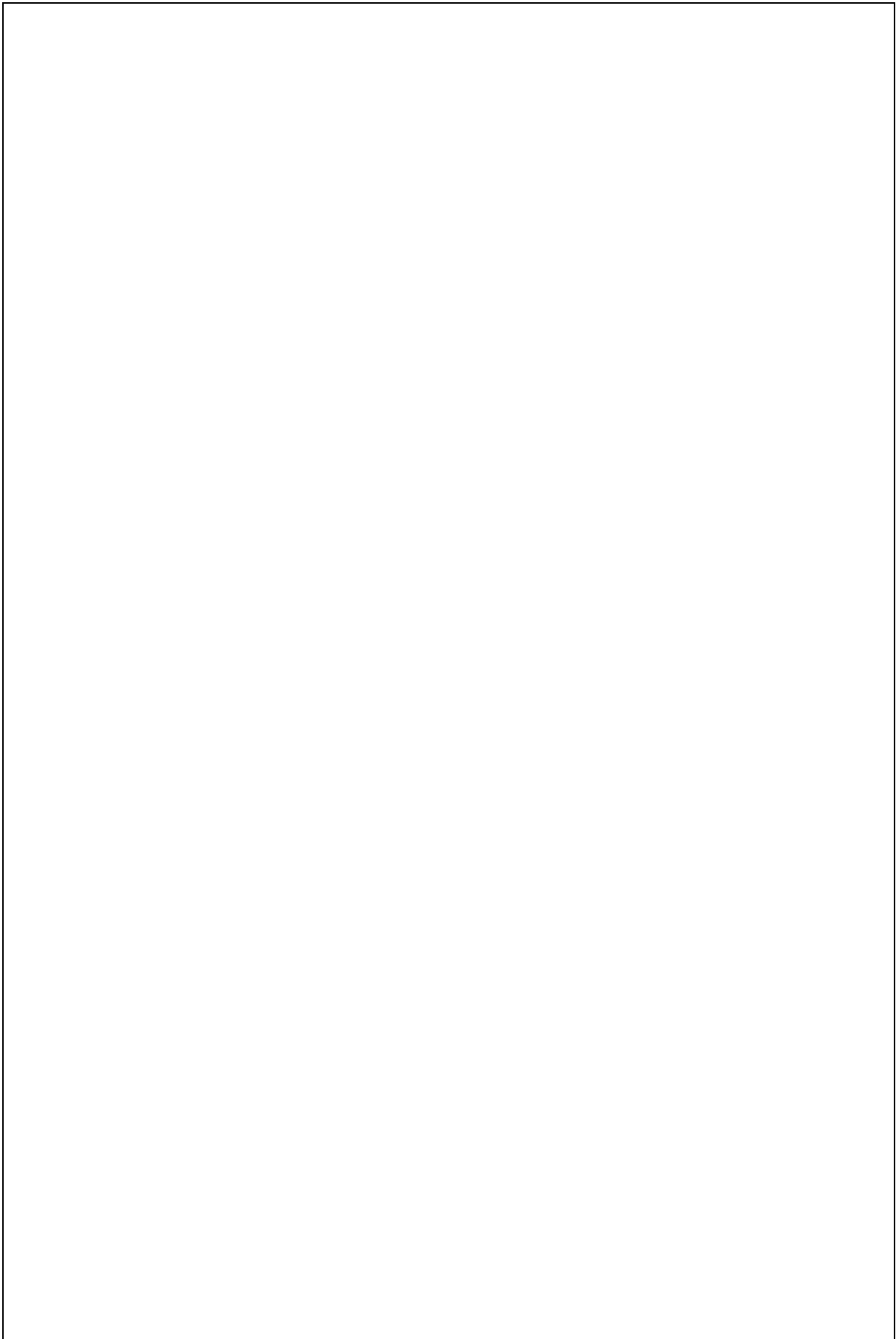
    public Person{
    }
    public Person(String n, String a, String c)
    {
        name = n;
        address = a;
        cityName = c;
    }
    // Accessor and Mutator methods
    public String toString()
    {
        String s = "Name\t : " + name + "\nAddress\t : " + address + "\nCity\t : "
            + cityName ;
        return s;
    }
}
```

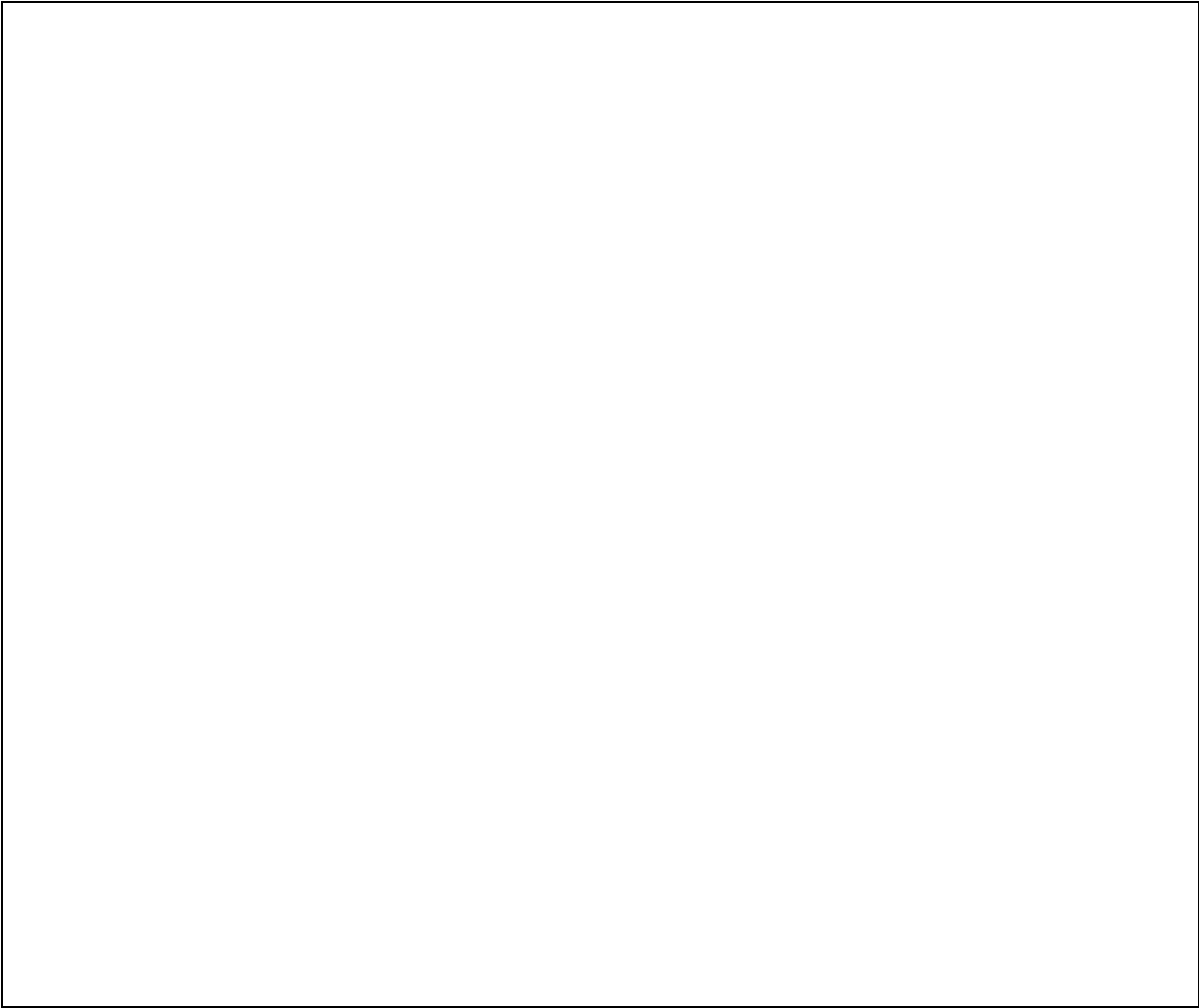
Using TextFields, write an applet that requests a name, address and a city name from a user. The applet should come also with:

- a Submit button that collects all the data from the text fields, stores them in an array of Person objects and clears the textfields in order to enter the data for another person,
- a Quit button ends the session and displays all the information within a text area of the applet.

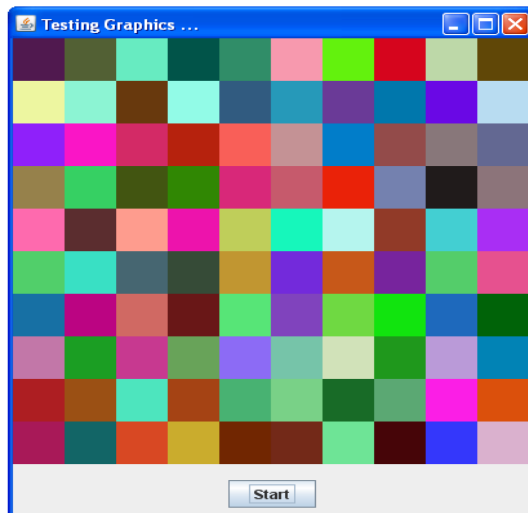
Note: you have to declare and manipulate the array of Person objects inside your class. The applet should look like this:





**Question 4 [40 Points]**

- a) **[25 Points]** Write a graphical application which creates a grid of 10 ´ 10 boxes with each having a random color, as shown below:



Clicking Start should change the color of each box in a random manner every 1 sec. This requires that each box is controlled by a separate thread.

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;

public class Q4 extends JFrame {
    private JPanel boxPanel;
    private int x;
    private int y;

    public Q4() {
        super("Testing Graphics ...");
        setSize(410, 480);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        boxPanel = new JPanel();
        add(boxPanel, BorderLayout.CENTER);

        JPanel startPanel = new JPanel();
        startPanel.setLayout(new FlowLayout());
        add(startPanel, BorderLayout.SOUTH);

        JButton start = new JButton(" Start ");
        start.addActionListener(new List1());
        startPanel.add(start);
    }

    public void paint(Graphics gf) {
        super.paint(gf);
        Graphics g = boxPanel.getGraphics();
        for (y = 0; y < 400; y += 40)
            for (x = 0; x < 400; x += 40) {
                Random ran = new Random();
                Color c = new Color(ran.nextInt(256), ran.nextInt(256),
                    ran.nextInt(256));
                g.setColor(c);
                g.fillRect(x, y, 40, 40);
            }
    }

    public static void main(String[] args) {
        Q4 f = new Q4();
        f.setVisible(true);
    }
}
```



```
private class List1 implements ActionListener {  
    public void actionPerformed(ActionEvent e) {  
        // complete this part
```



```
}
```

```
private class Filler extends Thread {  
    int x, y, s;  
    public Filler(int xVal, int yVal, int sVal) {  
        x = xVal;  
        y = yVal;  
        s = sVal;  
    }  
    public void run() {  
        // and this part
```



```
}
```

```
}
```

- b) **[15 Points]** Apply the steps of converting an Applet into an Application on the following code, you may add, and/or delete, re-write on the existing code to make it run as a standalone application.

```
import java.applet.*;
import java.awt.*;
public class DrawingLines extends Applet {
    int width, height;
    public void init() {
        width = getSize().width;
        height = getSize().height;
        setBackground( Color.black );
    }

    public void paint( Graphics g ) {
        g.setColor( Color.green );
        for ( int i = 0; i < 10; ++i ) {
            g.drawLine( width, height, i * width / 10, 0 );
        }
    }
}
```