415.101SC/ST 2000

Principles of Programming

Test: Monday 21st August 6.30pm – 7.45pm

Surname (Family Name):	
First Name(s):	
Login Name:	
ID Number:	
Lab Time:	

Note: Attempt **ALL** questions. Calculators are **NOT** permitted.
Write your answers in the spaces provided. There is space at the back for answers that overflow the allotted space. **Questions total 100 Marks**.

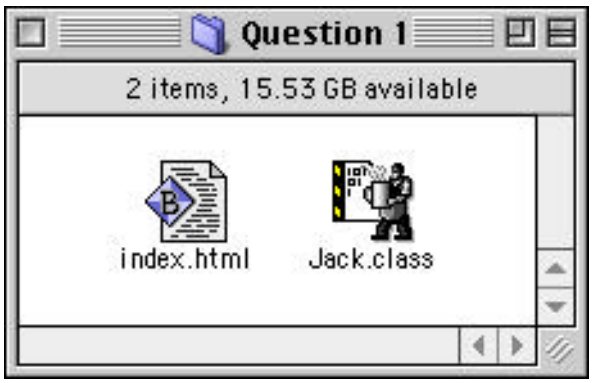
Section	Marks	Possible Marks
Q.1		10
Q.2		5
Q.3		10
Q.4		10
Q.5		6
Q.6		30
Q.7		4
Q.8		15
Q.9		10
Total		100

Question 1 (10 marks)

Here is the Java source code for a very simple applet:

```
import java.awt.*;
import java.applet.*;

public class Jack extends Applet {
  public void paint(Graphics g) {
    g.setColor(Color.black);
    g.drawLine(0, 0, 100, 100);
    g.drawLine(0, 100, 100, 0);
    g.drawLine(50, 0, 50, 100);
    g.drawLine(0, 50, 100, 50);
  }
}
```

This source code has been compiled into the byte code file, `Jack.class`. The folder in which this byte code file is stored also contains a file named `index.html`, as the following screen shot of the folder shows:

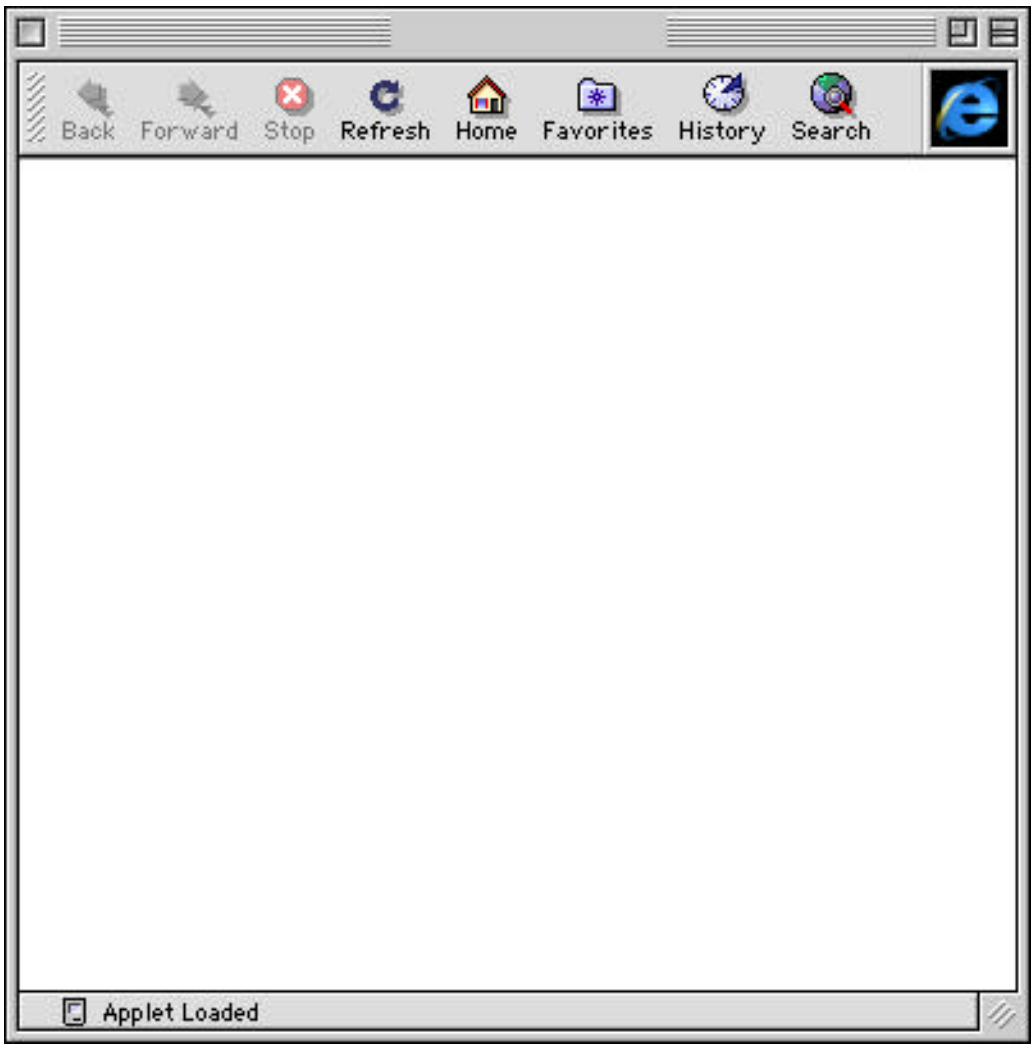


The file `index.html` contains the following HTML source for a simple web page:

```
<html><head><title>The Test</title></head>
<body><h1>Java</h1>
<ul>
<li>object oriented</li>
<li>platform independent</li>
</ul>
Here is an applet:
<hr>
```

```
<applet code = "Jack.class" width=100
height=100></applet>
<hr>
Go to the <a href = "http://www.cs.auckland.ac.nz">CS</a>
website.
</body>
</html>
```

In the window below, draw what would appear if the file `index.html` is opened in a browser:



Question 2 (5 marks)

The following Java application is supposed to simulate the tossing of a coin, that is, roughly half the time this application is run the output should be "A head was thrown" and the other half the output should be "A tail was thrown".

```
public glass WrongCoin {

  public static void main(String() args)

    double rand == Math.random();

    if (rand < 0.5);
      System.out.println("A head was thrown");
    else
      System.out.println("A tail was thrown");
  }
}
```

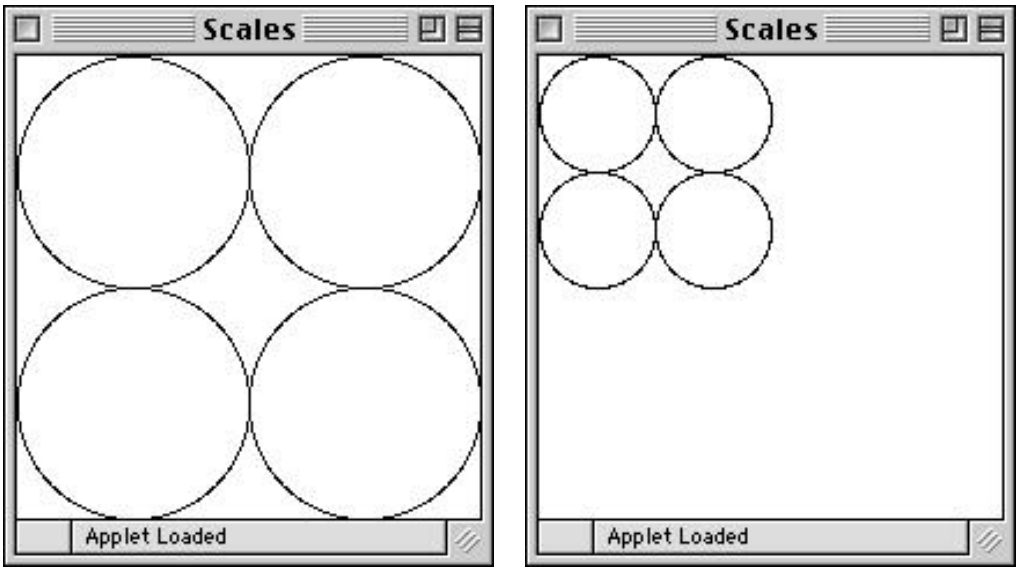
However, there are 5 syntax errors in the code above. Fix all the errors by writing the program out correctly and CIRCLE the changes. You will lose marks for making unneccessary changes.

Question 3 (10 marks)

You need to complete the `paint()` method of the following applet:

```
import java.awt.*;
import java.applet.*;

public class Scales extends Applet {

  public void paint(Graphics g) {
    final int SIZE = 100;




  }
}
```

All of the parameters to the drawing methods you call in the `paint()` method must be based on the constant `SIZE`. When the value of `SIZE` is `100` the applet should produce the drawing on the left below, where the applet window is exactly `200` pixels wide and `200` pixels high:

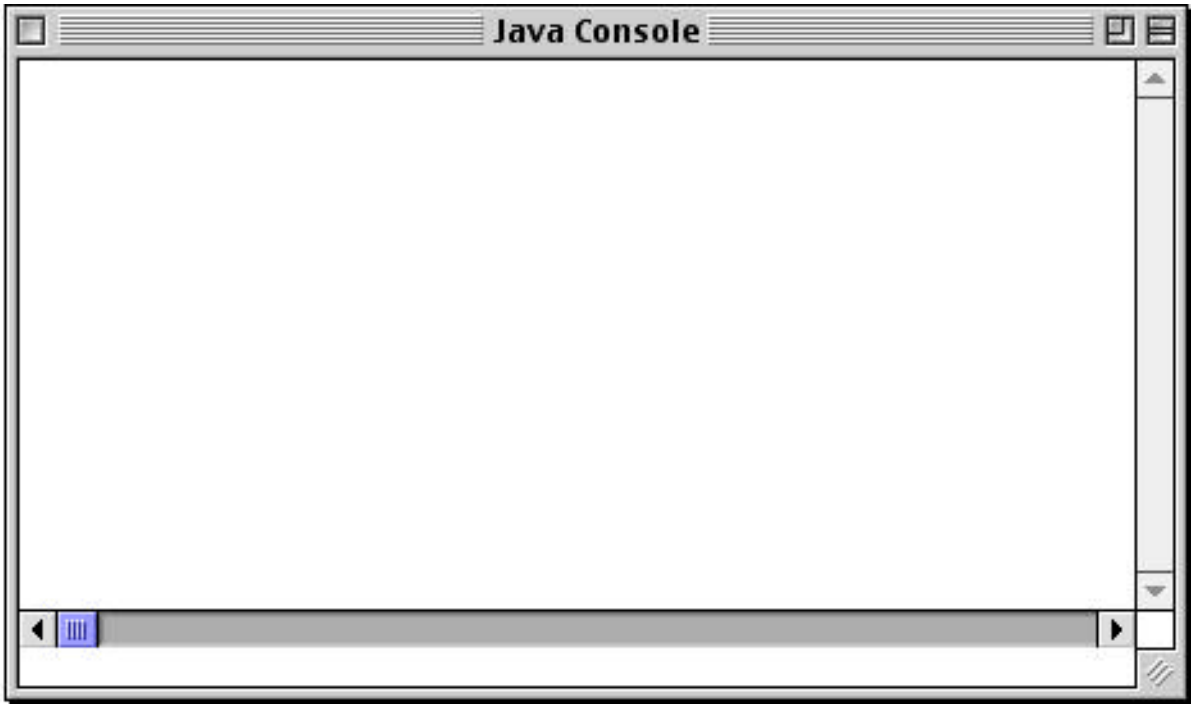


The constant `SIZE` can be used to scale the drawing. For example, if the value of `SIZE` is changed to `50`, the drawing on the right above should be produced.

Question 4 (10 marks)

What is the exact output of the following application (note that the "" in the `println()` statements represents the empty string):

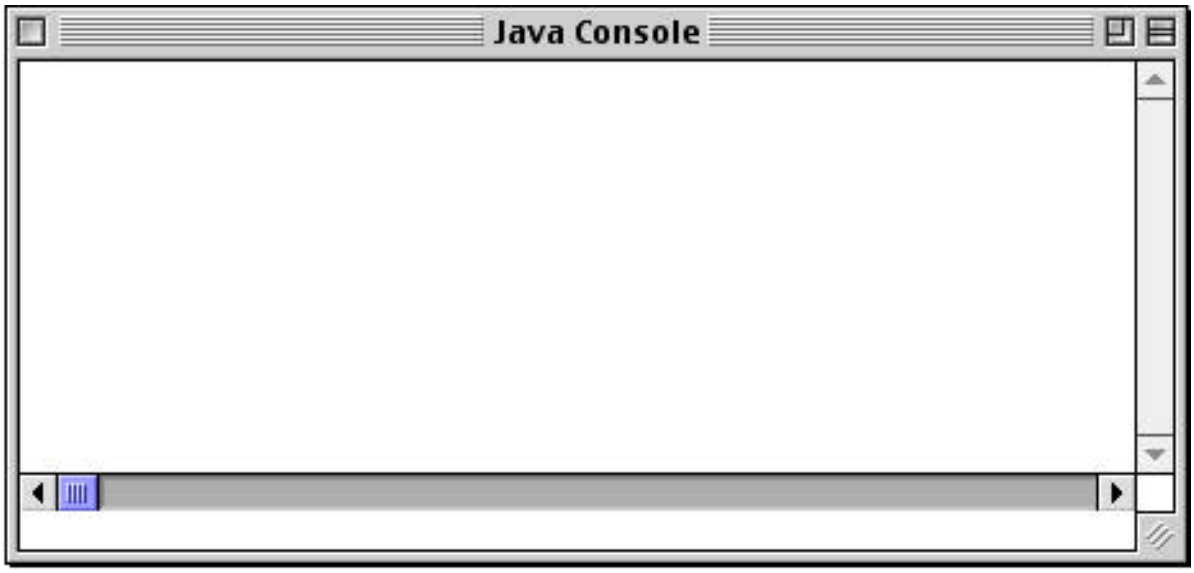
```
public class Q4 {
  public static void main(String[] args) {

    int i;
    double d;

    i = 5;
    d = 5.0;

    System.out.println(i + d);
    System.out.println("" + i + d);
    System.out.println(i + "" + d);
    System.out.println(i + d + "");

    System.out.println((int)((i * d) / (i + d))  - i);
    System.out.println((int)(i * d) % 3 - i);
  }
}
```



Question 5 (6 marks)

What is the output from the following program? You can assume that the `paint()` method is executed ONCE only.

```
Import java.awt.*;
import java.applet.*;

public class Q5 extends Applet{
 private int holdNum;

 public void init() {
  holdNum = 26;
  System.out.println("1. The number is: " + holdNum);

  int holdNum;
  holdNum = 15;
  holdNum = holdNum + 10;
  System.out.println("2. The number is: " + holdNum);
 }

 public void paint(Graphics g){
  holdNum = holdNum - 5;
  System.out.println("3. The number is: " + holdNum);
 }
}
```



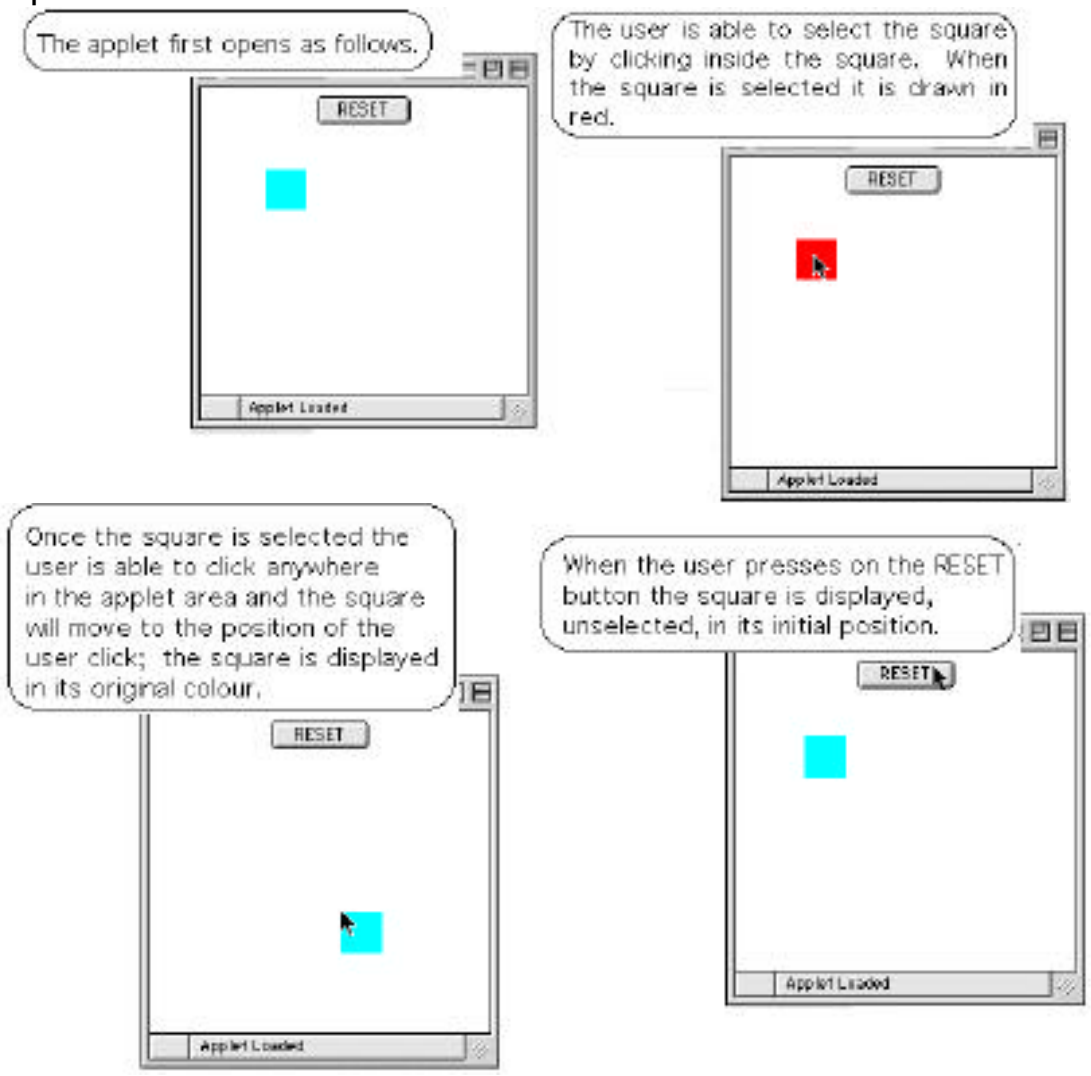
Question 6 (30 marks)

Complete the code for the following applet. The applet contains a reset button and displays a coloured square [the colour is your choice].

The user is able to select the square by pressing the mouse inside the square. When the square is selected it is drawn in red. [Note: if the square is not selected and the user presses outside the square nothing happens.]

Once the square is selected the user may press the mouse anywhere in the applet area and the square is moved to the position where the user pressed the mouse; once the square has been moved it is unselected and is displayed in its original colour.

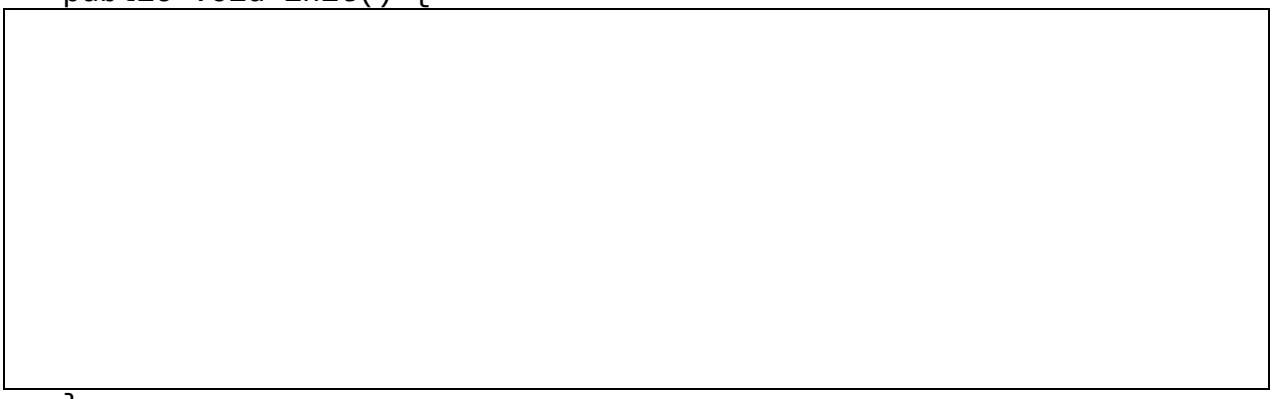
If the user presses the "RESET" button the square is displayed in its initial position and in its initial colour.



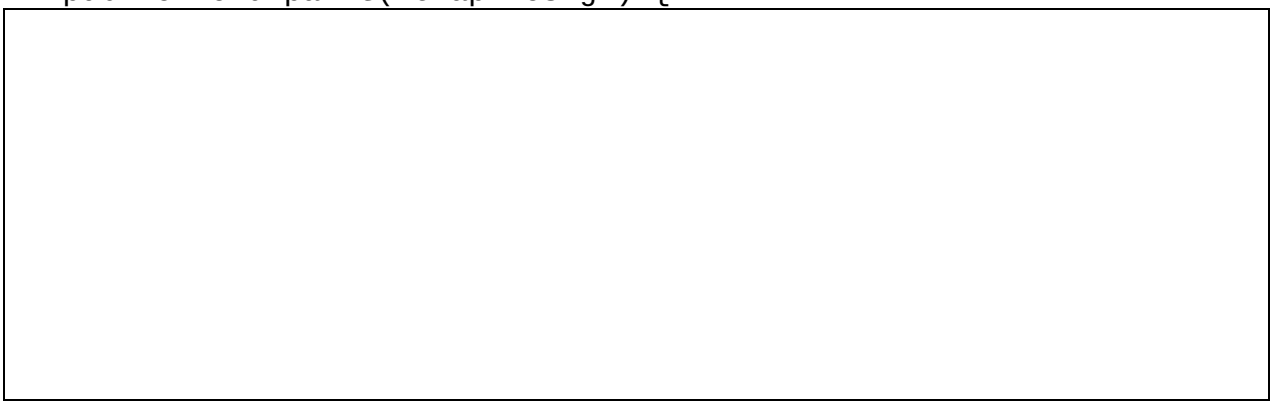
Complete the Applet below:.

```
Import java.awt.*;
import java.applet.*;
import java.awt.event.*;

public class Q07 extends Applet implements ActionListener,
                                                    MouseListener {

   private final int SIZE = 25;   /**  WIDTH and HEIGHT of the square */
   private final int START_X = 40;     /**  initial x position of the square */
   private final int START_Y = 50;     /**  initial y position of the square */
```

```
   /**  set up the Button, initialise instance variables etc. */
   public void init() {
```

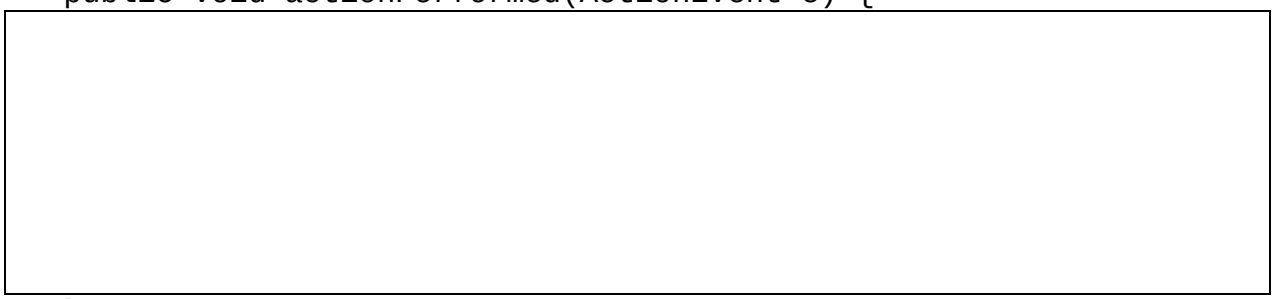


```
   }
   /**  display the square in its current position in the correct
     colour */
   public void paint( Graphics g ) {
```

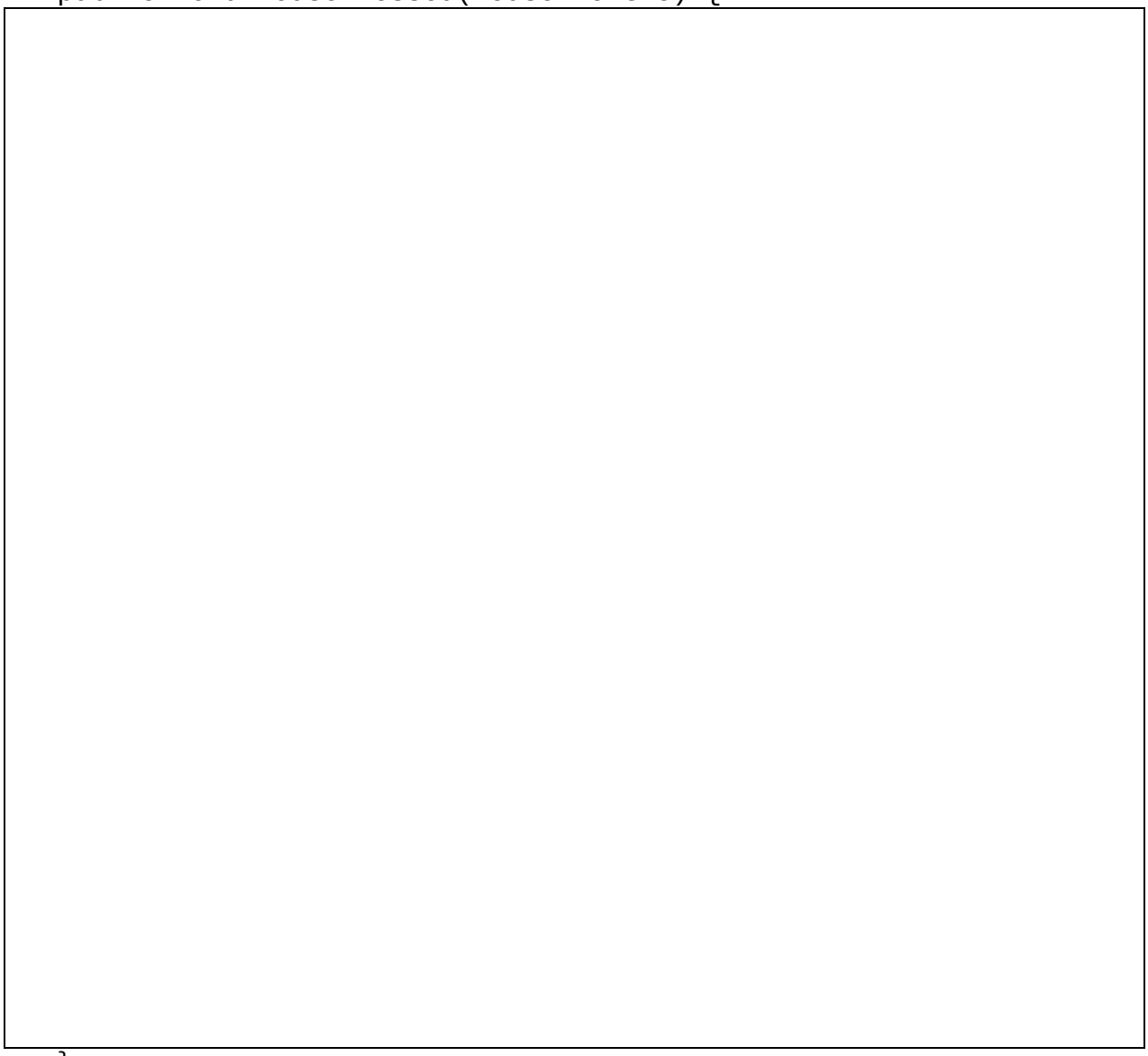


```
   }
```

```
    /**  called when the RESET button is clicked */
    public void actionPerformed(ActionEvent e) {
```



```
    }
    /**  called when the mouse is pressed */
    public void mousePressed(MouseEvent e) {
```



```
    }
    /**  these methods are not used */
    public void mouseEntered(MouseEvent e) {}
    public void mouseExited(MouseEvent e) {}
    public void mouseClicked(MouseEvent e) {}
    public void mouseReleased(MouseEvent e) {}
}
```

Question 7 (4 marks)

Do the following boolean expressions evaluate to true or false?

```
Int a = 34;
int b = 34;
int c = 62;
```

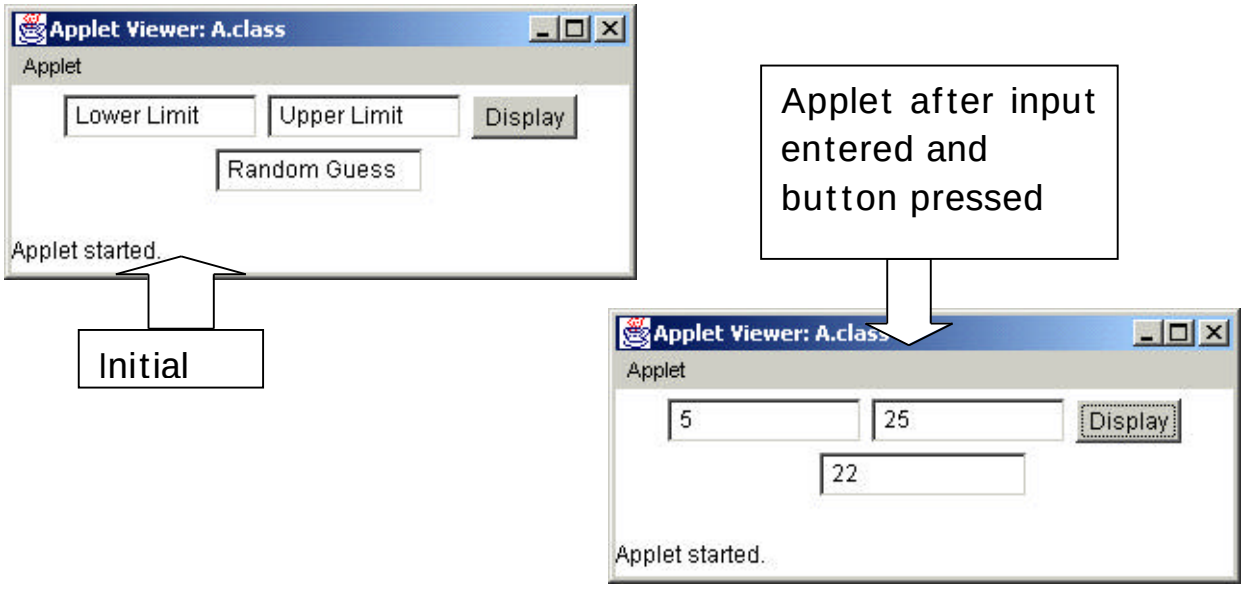
```
(!(a < b)) && (b >= c)
```

```
(a > b || a <= c) && (c != b)
```

QUESTION 8 next page

Question 8 (15 marks)

The applet shown here takes two numbers as input (a lower and upper limit), and displays a random number between the lower and upper limit (inclusive). The pictures below show the initial applet, and the applet after input has been entered and the button pressed.



The formula for converting a random number between 0 and 1 to a number between *lower* and *upper* is:

*number = Math.random() * (upper - lower) + lower*

Note that the *number* must be rounded to the nearest integer number.

Most of the applet code has been included below. Write the method `getGuess` to complete the applet given below.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

public class Q8 extends Applet implements ActionListener{

  TextField first;
  TextField second;
  TextField guess;
  Button display;
```

```
  public void init(){
    first = new TextField("Lower Limit");
    second = new TextField("Upper Limit");
    guess = new TextField("Random Guess");
    display = new Button("Display");

    display.addActionListener(this);

    add(first);
    add(second);
    add(display);
    add(guess);
  }

  public void actionPerformed(ActionEvent e){
    String s;
    s = getGuess(first.getText(), second.getText());
    guess.setText(s);
  }

  //Write your getGuess method here
```

```
}
```

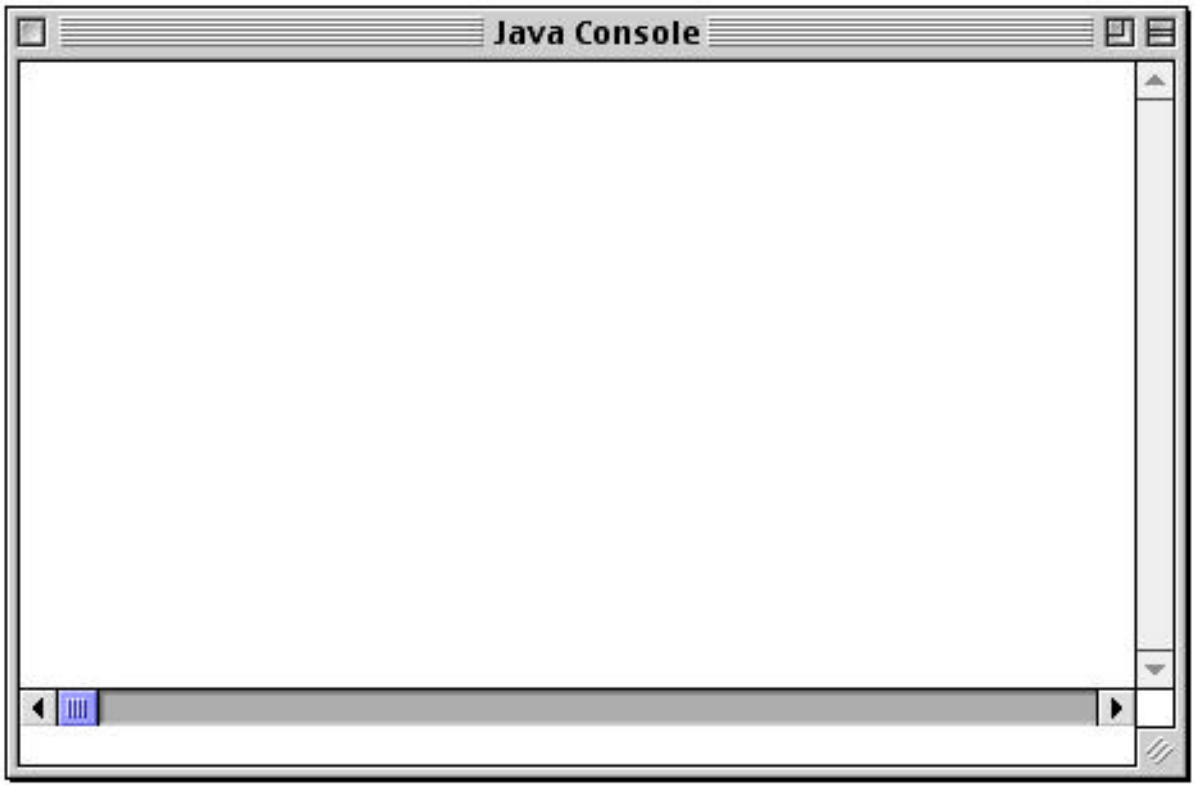
Question 9 (10 marks)

What is the output of the following applet?

```
Import java.awt.*;
import java.awt.event.*;
import java.applet.*;

public class A extends Applet{

  int a, b, c;

  public void init(){
    a = 0;
    b = 0;
    c = 0;

    reset();
    print(a,b,c);

    test1(a,b);
    print(a,b,c);

    reset();
    print(test2(a),b,c);
  }

  private void test1(int a, int b){
    a = 10;
    b = 30;
    c = 50;
  }

  private int test2(int x){
    int c;
    c = x+b;
    return c;
  }

  private void reset(){
    a = 1;
    b = 3;
    c = 5;
  }
```

CONTINUED

```
  private void print(int a, int b, int c){
    System.out.println("a = "+a+" b = "+b+" c = "+c);
  }

}
```



Overflow Sheet

Please indicate the number of the question you are answering.

Overflow Sheet

Please indicate the number of the question you are answering.

Methods you may find useful

Graphics class

```
void setColor(Color colour);
void fillOval(int x, int y, int width, int height);
void drawOval(int x, int y, int width, int height);
void drawLine(int x1, int y1, int x2, int y2);
```

Integer class

```
Int Integer.parseInt(String str);
```

String class

```
String String.valueOf(int x);
```

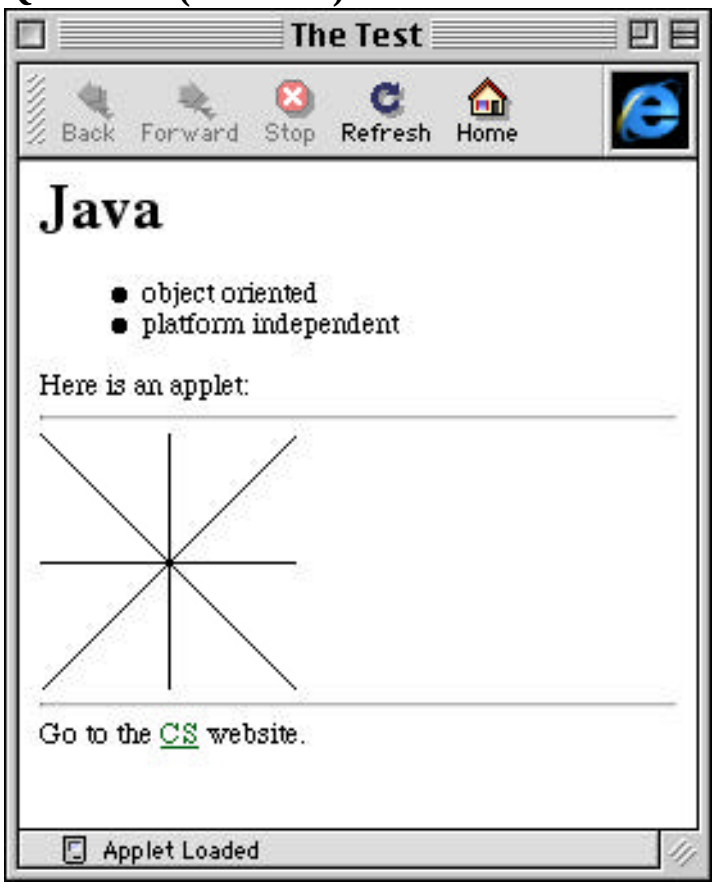
TextField class

```
String getText();
void setText(String str);
```

Rough Working

This sheet will **NOT** be marked

Question 1 (10 marks)



Question 2 (5 marks)

1 `()` after `String`
2 `==` after `rand`
3 ; after `if`
4 `glass`
5 missing { after `main`

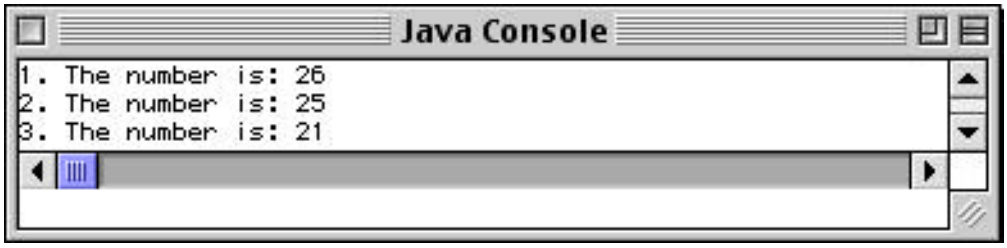
Question 3 (10 marks)

```
  public void paint(Graphics g) {

    final int SIZE = 50;

    g.drawOval(0, 0, SIZE, SIZE);
    g.drawOval(0, SIZE, SIZE, SIZE);
    g.drawOval(SIZE, 0, SIZE, SIZE);
    g.drawOval(SIZE, SIZE, SIZE, SIZE);

  }
```

Question 4 (10 marks)

```
10.0
55.0
55.0
10.0
-3
-4
```

Question 5 (6 marks)



Question 6 (30 marks)

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;

public class question07 extends Applet implements ActionListener, MouseListener
{

   private final int SIZE = 25;   /**  WIDTH and HEIGHT of the square */
   private final int START_X = 40;   /**  initial x position of the square */
   private final int START_Y = 50;   /**  initial y position of the square */
```

```
/**  current x, y of the square */
  private int squareX, squareY;
  /** reset Button */
  private Button resetB;
  /** boolean to store whether square area is cleared */
  private boolean isSelected;
```

```
   /**  set up the TextFields, Button */
   public void init() {
```

```
     isSelected = false;
     squareX = START_X;
     squareY = START_Y;

     addMouseListener(this);
```

```
    resetB = new Button("RESET");
    resetB.addActionListener(this);
    add(resetB);

  }
  /**  display the square in its current position */
  public void paint( Graphics g ) {
    if (isSelected)
      g.setColor(Color.red);
    else
      g.setColor(Color.cyan);

    g.fillRect(squareX, squareY, SIZE, SIZE);

  }
  /**  called when the RESET button is clicked */
  public void actionPerformed(ActionEvent e) {
    isSelected = false;
    squareX = START_X;
    squareY = START_Y;

    repaint();
  }

  /**  called when one of the buttons is pressed */
  public void mousePressed(MouseEvent e) {
    int  xPos, yPos;
        //get the x, y value of the user click
    xPos = e.getX();
    yPos = e.getY();
    if (isSelected){
      squareX = xPos;
      squareY = yPos;
      isSelected = false;
    }
    else   if   (xPos>squareX   &&   xPos<squareX+SIZE   &&
yPos>squareY
                                    && yPos<squareY+SIZE){
      isSelected = true;
    }

    repaint();
  }
  /**  these methods are not used */
  public void mouseEntered(MouseEvent e) {}
  public void mouseExited(MouseEvent e) {}
  public void mouseClicked(MouseEvent e) {}
  public void mouseReleased(MouseEvent e) {}
}
```

Question 7 (4 marks)

```
(!(a < b)) && (b >= c): false
(a > b || a <= c) && (c != b)): true
```

Question 8 (15 marks)

```
  //Write your getGuess method here
  public String getGuess(String lowerS, String upperS){
    int guess;
    int lower, upper;
    lower = Integer.parseInt(lowerS);
    upper = Integer.parseInt(upperS);
    guess = (int) Math.round(Math.random()*(upper -
                                            lower) + lower);
    return String.valueOf(guess);
  }
```

Question 9 (10 marks)