1. What is the output to the screen?

   public class Question1{
       public static void main (String[] args){
           System.out.println(f(4));
       }
       public static int f (int x){
           if (x==1 || x==0)
               return x;
           else
               return f(x-1) + f(x-2);
       }
   }

2. (a) Write a class PersonalData which has three String data fields representing the first name, last name, and social-security number. Make the PersonalData class implement the Comparable interface, based on comparing social-security numbers. Also write a constructor for the class to set the data.

   (b) Write a class SSNumberFormatException which extends the Exception class and sets the exception message to be "Invalid Social Security Number Format". Modify your PersonalData class so that the constructor throws a SSNumberFormatException if the social security number is not of the format xxx-xx-xxxx, where each x represents a digit from '0' to '9'.

3. Consider the following array of Strings:

   "happy" "home" "area" "asp" "xylophone" "ire" "ape"

   (a) Trace through the selection sort algorithm acting on the above String array.

   (b) Trace through the insertion sort algorithm acting on the above String array.
4. What is the output to the screen?

```java
public class Question3{
    public static void main (String[] args){
        String[] a = new String[3];
        a[0] = new String("House");
        a[2] = new String("Home");
        for (int i=0; i<3; i++){
            try{
                System.out.println(i/(i-2));
                System.out.println(a[i].length());
                System.out.println("Pass " + i);
            }catch (NullPointerException e){
                System.out.println("Properly initialize.");
            }
            catch (ArithmeticException e){
                System.out.println("Don’t divide by zero.");
            }
        }
    }
}
```

5. A class has an instance variable

```java
int[] a;
```

Write a class method

```java
void delete(int n)
```

which moves all elements of `a` with index `(n + 1)` or above one place to the left, and puts a 0 in the last element of `a`. (So if `a` contains the array elements 5 7 13 2 6 1, then after a method call `delete(2)`, `a` should contain 5 7 2 6 1 0.)

6. A function $f$ is defined for positive integers by

$$
f(n) = \begin{cases} 
    n^3 + f(n-1) & n > 0 \\
    0 & n = 0
\end{cases}
$$
Implement \( f \) as a \texttt{static} method.

7. Write a Java program which reads integers from a file \texttt{ints.dat} into an array (you may assume the file contains at most 1000 integers). Then use the \texttt{Sorting.selectionSort} method to sort the array. Print all the elements of the sorted array to the screen, separated by spaces. (You do not have to write the selection sort algorithm; just make an appropriate method call.)

8. Write a method

\[
\text{public void paintComponent(Graphics page)}
\]

which paints a bullseye picture consisting of 5 concentric black and white rings of width 25 pixels. The outermost circle should have radius 150 and center \((150, 150)\). Inside the innermost ring, draw a red disk (the bullseye) of radius 25. The picture is available at

\url{http://www.pegasus.rutgers.edu/~loftin/cs101/bullseye/bullseye.html}

9. Describe what happens when the up and down buttons are pushed in the following applet. In particular, what is the text in the label after the up button is pushed four times and then the down button is pushed once?

\[
\text{import java.awt.*;  
import java.awt.event.*;  
import javax.swing.*;  
public class App extends JApplet{  
   int num;  
   private JLabel label;  
   private JButton upButton, downButton;  
   public void init(){  
      Container content = getContentPane();  
      JPanel panel = new JPanel();  
      content.add(panel);  
      num = 0;  
      label = new JLabel("");  
   }
\]
upButton = new JButton("Up");
downButton = new JButton("Down");
ButtonListener listener = new ButtonListener();
upButton.addActionListener(listener);
downButton.addActionListener(listener);
panel.add(label);
panel.add(upButton);
panel.add(downButton);
}
private class ButtonListener implements ActionListener{
    public void actionPerformed(ActionEvent event){
        if (event.getSource() == upButton)
            num++;
        else if (event.getSource() == downButton)
            num--;
        String stars = "";
        for (int i=0; i<num; i++)
            stars += "*";
        label.setText(stars);
    }
}