

# Final review 2013

## Short answer

1. For this question you are to extend the code shown in the program shown below. When the program first runs this program the GUI will look like the image in Figure 1. You are to write the code that will change the text of the button. When the button label says "On" it will toggle to "Off" when the button is pressed and when the button label says "Off" it will toggle to "On" when the button is pressed. You can assume that all the necessary import statements have already been included.

```
public class Driver
{
    public static void main (String [] args)
    {
        JFrame myFrame = new JFrame ("Plain window");
        JButton myButton =new JButton("On");
        myFrame.add(myButton);
        myFrame.setBounds(100,100,300,200);
        myFrame.setVisible(true);
        // Make any changes to the Driver class here

    }
}
```

<< **Make any changes to the rest of the program here** >>

```
public class MyButtonListener implements ActionListener
{
    public void actionPerformed (ActionEvent e)
    {
        JButton myButton = (JButton) e.getSource ();
        if (myButton.getText().equals("On"))
            myButton.setText("Off");
        else if (myButton.getText().equals("Off"))
            myButton.setText("On");
    }
}
```

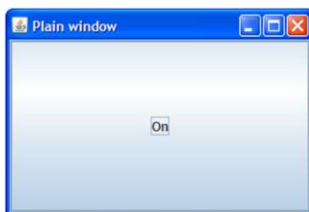


Figure 1

2. For this question you are to refer to the following Star Trek™ game. The base type of vessel is a ‘starship’ which has a number of basic attributes and abilities, some of which are shown below:

```
public class StarShip {
    private int hullValue;
    private int shieldValue;

    public int calculateDamage () {
        // Generate and return the damage inflicted on ship
        // Assume it is working correctly
    }

    public void absorbDamage (StarShip attacker) {
        << Write your answer here >>
    }
}
```

```
        << End of answer space >>
    }
}
```

The calculateDamage method is used by an attacker to determine the amount of damage that will be inflicted on a defending ship.

The absorbDamage() method is used by the defending ship in order to determine the effects of attack damage. This method takes one parameter: a reference to the attacking ship. Normally attack damage will be deducted from the shields until they are reduced to zero (shields are ‘down’) at which point the remaining attack damage is deducted from the hull.

A JemHadarShip is a starship that employs a weapon that can bypass the shields so that damage is always deducted directly from the hull. Only the JemHadar ships will have this ability in the game.

Write the code so that: (1) JemHadar starships have the ability to ‘penetrate’ the shields of the defender during an attack (2) Other attacking starships will first damage the shields and then the hull of the defender. You can only implement this ability by modifying the body of the absorbDamage method. You cannot change the rest of the program nor can you change the signature of this method.

3. What is the output of the following program?

```
public class TraceDriver
{
    public static void main (String [] args)
    {
        A a = new A ();
        B b = new B ();
        a.fun1();
        a.moreFun();
        b.fun1();
        b.fun2();
        b.fun3();
        b.soFun();
        a.moreFun();
    }
}
class A
{
    public int x;
    public int y;
    public A () { x = 1; y = 2; }
    public void fun1 ()
    {
        System.out.println(x + " " + y);
    }
    public void moreFun ()
    {
        System.out.println("This is so much fun!");
    }
}
class B extends A
{
    public int y;
    public int z;
    public B () { y = 20; z = 30; }

    public void fun1 ()
    {
        System.out.println(y + " " + z);
    }
    public void fun2 ()
    {
        System.out.println(x + " " + super.x);
    }
    public void fun3 ()
    {
        super.fun1();
        System.out.println(y + " " + z);
    }
    public void soFun ()
    {
        System.out.println(x + " " + y + " " + z);
    }
}
```

4. Critique the following implementation of the Model-View-Controller design pattern.

```
public class Manager
{
    private Node head = null;
    private char menuSelection;

    public void addElement (Node temp) {
        : : :
    }

    public void removeElement (int index) {
        : : :
    }

    public void displayList () {
        : : :
    }

    public void displayMenu () {
        : : :
    }

    public void getMenuSelection () {
        : : :
    }
}
```

### **Multiple choice**

- 1) Which of the following statements are true of Java programs?
  - a. A class can implement more than one interface.
  - b. A class can extend more than one class.
  - c. Multiple inheritance is a built-in part of the language.
  - d. (b) & (c)
  - e. None of the above statements are true.
  
- 2) Which of the following statements are true about Java programs?
  - a. Overriding refers to methods with the same name & parameter list but have separate definitions in the parent and child class.
  - b. Overloading refers to methods that have the same name but are distinguished by their parameter lists.
  - c. Preceding an attribute with the word private in a class definition (along with implementing the mandatory accompanying methods) is an example of information hiding.
  - d. Java programs are both compiled and interpreted.
  - e. All of the above are true

3) What is the output of the following code segment?

```
int[] intArray = new int[4];
try
{
    int x = intArray[5]/intArray[0];
    System.out.println(intArray[5]);
}
catch (ArrayIndexOutOfBoundsException exp)
{
    System.out.println("Error: array index out of bounds");
}
catch (Exception exp)
{
    System.out.println("Error: Exception");
}
```

- a. Error: Exception  
Error: array index out of bounds
  - b. Error: Exception
  - c. Error: array index out of bounds
  - d. Error: array index out of bounds  
Error: Exception
  - e. The program will crash, no output
- 4) Which of the following statements is the equivalent to stating that class Y inherits from class X?
- a. Class X is the parent, Class Y is the child
  - b. Class Y is the generalization, Class X is the specialization
  - c. Class X is the superclass, Class Y is the subclass
  - d. (a) & (c)
  - e. All of the above are equivalent

JT: Good luck with your finals.