

Midterm II - CSE11 – Fall 2013
CLOSED BOOK, CLOSED NOTES
50 minutes, 100 points Total.

Name: _____

ID: _____

Problem #1 (8 points)

Rewrite the following code segment using a for loop instead of a while loop (that is properly replace the while loop with a for loop so that running the rewritten code would yield the identical results as the original) Use the minimum number of statements possible.

```
int i = 100;
int k = 4;
while (i >= 0 )
{
    if ( (i % k) == 0 )
        System.out.println("Divisible");
    i = i - 3;
}
```

ans:

```
int k=4;
for ( int i=100; i >= 0; i-- )
{
    if ( (i % k) == 0 )
        System.out.println("Divisible");
}
```

Problem #2 (12 points)

Assume the file `ElectricalDevice.java` has only the following contents

```
public interface ElectricalDevice {
    public boolean powerOn();
    public boolean powerControl(int onOff);
    public static final int VOLTS = 120;
}
```

(Problem 2 cont'd) Further suppose that the complete source of Kettle.java is as follows

```
public class Kettle implements ElectricalDevice {  
  
    private int volts;  
    private int powerState = 0;  
    public Kettle() {  
        volts = ElectricalDevice.VOLTS;  
    }  
  
    public boolean powerOn() {  
        if (powerState != 0)  
            return true;  
        else  
            return false;  
    }  
  
}
```

(a) Kettle.java will not compile, What is the critical error?

Kettle does not implement the powerControl() method and therefore does not properly implement the Electrical device interface.

(b) Is the following a legal declaration? Why or why not?

```
ElectricalDevice [] myDevices = new ElectricalDevice[10];
```

Yes. This declares an array of references to objects that implement the ElectricalDevice interface.

(c) Is the following a legal declaration? Why or why not?

```
ElectricalDevice myDevice = new ElectricalDevice();
```

No. ElectricalDevice is an interface, not a class. The new operator can only be applied to classes.

Problem #3 (20 points) Read the following program. Answer questions on the following page

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

public class Q3 implements Runnable, ActionListener, ChangeListener {
    private JTextArea out;
    private JButton buttonReset;
    private JSlider mySlider;
    private static final int MAX_INT=12;

    private String squares(int b) {
        String rval="";
        for (int i = 1; i<= b; i++)
            rval += ".format("% 4d", i*i);
        return rval;
    }

    public void run() {
        JFrame f = new JFrame("Squares");
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        f.setLayout(new BorderLayout());

        String sq = squares(MAX_INT);
        out = new JTextArea(sq,1,4*(MAX_INT + 1));
        out.setFont(new Font(Font.MONOSPACED,Font.PLAIN,12));

        mySlider = new JSlider(1, MAX_INT, MAX_INT);
        buttonReset = new JButton("Reset");

        Container contentPane = f.getContentPane();
        contentPane.add( out, BorderLayout.CENTER );
        contentPane.add( mySlider, BorderLayout.NORTH);
        contentPane.add( buttonReset, BorderLayout.SOUTH);
        mySlider.addChangeListener(this);

        f.pack();
        f.setVisible(true);
    }

    public void stateChanged (ChangeEvent evt ) {
        int j = mySlider.getValue();
        out.setText(squares(j));
    }

    public void actionPerformed (ActionEvent evt) {
        out.setText(squares(MAX_INT));
        mySlider.setValue(MAX_INT);
    }

    public static void main(String[] args) {
        Q3 q3= new Q3();
        SwingUtilities.invokeLater(q3);
    }
}
```

Problem #3 (cont'd)

(a) When the program first runs, which of the following is the correct screen output? Explain your reasoning.



(a) is the correct initial screen

1. The slider is the `Border.NORTH` location, and is initialized to the maximum value (12)
2. the text area in the `Border.Center` Location an is intialized with the string "1 4 9 ... 144"
3. The reset button is placed in the `Border.SOUTH` location

(b) What happens when the method `ActionPerformed()` is invoked?

The slider is reset to it maximum value
The text is reset to the string "1 4 9 16 144"

(c) What will the program do when the Reset button is pressed by the user? Explain why.

Nothing. The `Q3` instance is not registered as an `ActionListener` with the Reset button. That is, `buttonReset.addActionListener(this)` does not appear in the code.

Problem #4 (20 points) Circle T (indicating True) or F (indicating False) for each one of the following.

T	<input checked="" type="checkbox"/>	F	A class can implement only one interface
T	<input checked="" type="checkbox"/>	F	<code>pause (100)</code> causes a program to wait exactly 100 ms before resuming
<input checked="" type="checkbox"/>	T	<input checked="" type="checkbox"/>	F The <code>onMouseClicked ()</code> method used in this class is only invoked when clicking on <code>objectdraw DrawingCanvas</code> object.
T	<input checked="" type="checkbox"/>	F	An <code>activeObject run ()</code> method must invoke <code>pause ()</code>
<input checked="" type="checkbox"/>	T	<input checked="" type="checkbox"/>	F In Program #4, the <code>record ()</code> method defined in a <code>CemeteryController</code> is termed a callback.
T	<input checked="" type="checkbox"/>	F	In Program #4, in order to use <code>Tombstone</code> objects in both <code>TestCemetery</code> and <code>Cemetery</code> the <code>Tombstone</code> must implement the <code>CemeteryController</code> interface.
<input checked="" type="checkbox"/>	T	<input checked="" type="checkbox"/>	F All for loops can be converted to while loops.
<input checked="" type="checkbox"/>	T	<input checked="" type="checkbox"/>	F A good programming habit is to declare instance variables with the <code>private</code> access modifier
T	<input checked="" type="checkbox"/>	F	A good approach to solving a programming problem is to write all the code first and then debug the result.
T	<input checked="" type="checkbox"/>	F	The <code>BorderLayout ()</code> layout used in Swing can be applied only to <code>JFrame</code> objects
T	<input checked="" type="checkbox"/>	F	The <code>FlowLayout</code> class defines five distinct areas in which a program can place graphical objects
T	<input checked="" type="checkbox"/>	F	<code>FilledOval ovals [10] ;</code> creates ten <code>FilledOval</code> objects named <code>oval0</code> , <code>oval1</code> , <code>oval2</code> , ..., <code>oval9</code> ;
<input checked="" type="checkbox"/>	T	<input checked="" type="checkbox"/>	F <code>int [] [] matrix = new int [10] [20] ;</code> creates an array of integers with 200 elements
T	<input checked="" type="checkbox"/>	F	If one declares <code>int [] [] matrix = new int [10] [20] ;</code> then <code>matrix.length</code> evaluates to 200.
T	<input checked="" type="checkbox"/>	F	All two-dimensional arrays in java are rectangular.
T	<input checked="" type="checkbox"/>	F	If a method returns a value, it can only return a primitive type.
T	<input checked="" type="checkbox"/>	F	The <code>continue</code> statement used inside of a loop body indicates that the program should immediately execute the first statement following the loop body.
<input checked="" type="checkbox"/>	T	<input checked="" type="checkbox"/>	F Array indices cannot be of <code>double</code> type
<input checked="" type="checkbox"/>	T	<input checked="" type="checkbox"/>	F <code>for (i = 0 ; i ++ < 10 ;) ;</code> is a valid for-loop.
T	<input checked="" type="checkbox"/>	F	The <code>this</code> keyword means “this class” when used in the context <code>addActionListener (this) ;</code>

Problem #5 (20 points) Write a public method called with the following signature

```
insertOval(FilledOval ovals[], int idx)
```

`insertOval` inserts an empty slot into the array `ovals` at index `idx` by properly moving any existing elements. It should only perform the operation if the last element of the `ovals` array is also empty. It should return `true` if the insert operation was successful, `false` otherwise. The method should not generate any exceptions for any combination of arguments.

```
import objectdraw.*;

public boolean insertOval(FilledOval ovals[], int idx)
{
    // check valid parameters
    if (ovals == null) return false;
    if (idx < 0 || idx >= ovals.length) return false;
    if (ovals[ovals.length - 1] != null) return false;

    for ( int i = ovals.length - 1; i > idx; i--)
        ovals[i] = ovals[i-1];
    ovals[idx] = null;
    return true;
}
```

Problem #6 (15 points) More fun with for loops

In each of the following, create a valid `for(; ;)` statement to achieve the desired results of the following two-line code segment:

```
for ( ; ; )  
    System.out.println(i);
```

example: print out integers from 1 to 10
ans: `for (i = 1; i <= 10; i++)`

Assume that `i` already declared as

```
int i;
```

- (a) print out k^2 for $k = 11, 13, 15, 17, 19, 21$
- (b) print out 3^k $k=1, 2, 3, 4, 5$
- (c) print out every ninth integer starting at 9 and ending at 108 (includes 108)
- (d) print out the numbers from 1 to 41 that are NOT evenly divisible by 3
- (e) print out the numbers 1, 4, 9, 16, 25, 36, 49, 64, 81, 100

- (a) `for (int j = i = 11; (i=j*j) <= 21*21; j+=2)`
- (b) `for (i = 3; i <= 3*3*3*3*3; i*=3)`
- (c) `for (i = 9 ; i <= 108 ; i+=9)`
- (d) `for (i = 1; i <= 41 ; i += i % 3)`
- (e) `for (int j = i = 1; (i=j*j) <= 100; j++)`
- (e) Alternative answer:
`for (int j = i = 1; (j+=2) <= 21; i += j)`

Problem #7 (5 points)

```
public class NotQuite
{
    private int firstArg;
    public static void main(String[] args)
    {
        NotQuite temp = new NotQuite();
        firstArg = Integer.parseInt(args[0]);
    }
}
```

when compiling the following error is given:

```
$ javac NotQuite.java
NotQuite.java:7: error: non-static variable firstArg cannot be referenced
from a static context
    firstArg = Integer.parseInt(args[0]);
    ^
1 error
```

Why is this error occurring? What is the simplest fix you can make to the code to address the error?

`firstArg` is instance variable and is only valid after a new `NotQuite` instance is created. `Main` is a static method and is invoked before any instance has been created. Hence `firstArg` is not available when the main method is invoked.

The simplest fix is to declare `firstArg` with the static modifier to make it a class variable.