ESE112– Programming with Java

Quiz 2 – November 9, 2006

Name: ____________________________________________

Student ID: _______________________________________
(from your PennCard)

Email ____________________________________________ @seas.upenn.edu

Instructions:

• You have 50 minutes to answer all of the questions. The quiz is worth 75 points. The point value of each question is given.

• Write your answers on the quiz pages. The back side of each page may be used as a scratch pad.

• Questions during the quiz should be about the wording only. If you have a question, raise your hand and we’ll come to you. (This is less disruptive for others than if you come to us.)

• DON’T PANIC! If you find a question that you cannot solve right away, consider moving on and returning to it after you finish the rest of the questions.

• Good luck!

<table>
<thead>
<tr>
<th>Score</th>
<th>Max</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13</td>
<td>Questions about references and arrays</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>Arrays of primitives</td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>Arrays of objects</td>
</tr>
<tr>
<td>Total</td>
<td>75</td>
<td></td>
</tr>
</tbody>
</table>
Questions about References and Arrays

1. (a) True or false: An array is a fixed-size collection of items that all have the same type. 1 points

(b) Briefly describe what kind of code generates an ArrayOutOfBoundsException 2 points

(c) Briefly describe what kind of code generates an NullPointerException 2 points

(d) True or false: If a reference to an object is passed to a method, the method can change the object. 2 points

(e) True or false: Assuming there is a class called Widget and two variables widget1 and widget2, both with type Widget, the expression “widget1 == widget2” is true if both variables are null or if both are pointing to the same object. 2 points

(f) True or false: ”An array is an object.” 2 points

(g) Given the following valid Java code, what is the range of valid indices/indexes for the array called ”data”? 2 points

```java
public void foo(int n){
    int[] data = new int[n];
    ...
}
```
Arrays of Primitives

2. Consider this Toolkit class:

```java
public class Toolkit {

    public void doit1(int[] data) {
        for (int i = 0; i < data.length; i = i + 2) {
            data[i] = data[i] * 2;  // data[i] *= 2;
        }
    }

    public void doit2(int[] data, int m) {
        for (int i = 0; i < data.length; i++) {
            if (data[i] > m)
                data[i] = m;
        }
    }

    public void doit3(int[] data) {
        for (int i = 0; i < data.length; i++) {
            int temp = data[i];
            data[i] = data[data.length - 1 - i];
            data[data.length - 1 - i] = temp;
        }
    }

}
```

(a) What does the doit1 method do? 4 points

(b) What does the doit2 method do? 4 points

(c) What does the doit3 method do? 4 points
Playlist (Arrays of Objects) REFERENCE SHEET

YOU MAY TEAR THIS PAGE OUT FOR REFERENCE.

We supply a Song class and ask you to complete a Playlist class. The code for the Song class and sample interactions appear below. Write the code for the Playlist class on the next page(s). Playlist should have an instance variable that is an array of Songs.

```java
public class Song{
    private String name;
    public Song(String name){
        this.name = name;
    }
    public String getName() { return name; }
}
```

Sample Interactions

```java
> Song song1 = new Song("happy birthday");
> Song song2 = new Song("the red and blue");
> Song song3 = new Song("ymca");
> Song song4 = new Song("swan song");
> Playlist pl = new Playlist(3); // The array of Songs should have size = 3
> pl.add(song1)
true
> pl.contains("happy birthday")
true
> pl.add(song1)
false  // song names must be unique
> pl.getTrackNumber("happy birthday") // the first track is #0. return -1 if song not found
0
> pl.contains("the red and blue") || pl.contains("ymca"))
false
> pl.add(song2);
> pl.add(song3);
> pl.getTrackNumber("ymca")
2
> pl.toString()
"[happy birthday, the red and blue, ymca]"
> pl.remove("the red and blue")
true
> pl.getTrackNumber("ymca")
1 // Note that when a track is removed, those with a higher number are "shifted left"
> pl.contains("the red and blue")
false
> pl.add(song4); // The Playlist is full, with 3 Songs. Additional Songs may be added by
> pl.add(song2); // effectively doubling the size of the array (in this case to size = 6)
> pl.toString()
"[happy birthday, ymca, swan song, the red and blue]"
```
Playlist (Arrays of Objects)

3. Write the code for Playlist below. Don’t bother to write a toString() method. TIP 1: Consider having some of your methods call the contains(...) method that you write. TIP 2: Save until last writing the code that doubles the array size as needed.
YOU MAY CONTINUE WRITING CODE ON THIS PAGE.