First Name: __________________________

Last Name: __________________________

Student ID: __________________________

Section time (10am/11am) _________ TAs: __________________________

I pledge on my honor that I have not given or received any unauthorized assistance on this examination.

Your signature: __________________________

General Rules:

- This exam is closed-book and closed-notes.
- Check that the last page is page number 9.
- Write your answers in the space provided. If you need additional space, raise your hand.
- If you have a question, please raise your hand.
- Total point value is 100 points. Point values attached to individual problems are subject to possible (minor) changes.
- Good luck!
Problem 1 (45 pts)

1. (7 pts) Transform the following if-else statement into an equivalent switch statement

```java
int x = ...(initialization omitted) ...
if ( x == 0 ) foo1();
else if ( x == 1 ) foo2();
else if ( x >= 3 && x < 5 ) foo3();
else foo4();
```

2. (4 pts) Using the Java Random class and the nextInt(n) method write an code fragment that computes a random int r over the range 200 \leq r \leq 300. (Remember that nextInt(n) returns a random int in the range 0 to n-1.)
3. (6 pts) Consider the following initializations:

```java
int[ ] data = new int[8];
data = null;
```

Following this, give the result of each of the following expressions. If the expression results in an error (either compile-time or run-time error) write “ERROR”.

(a) `data.length` ________________________________

(b) `data == null` ________________________________

(c) `data[0] == null` ________________________________

4. (14 pts) Consider the following class. (The Control interface is defined elsewhere.)

```java
public class RemoteControl implements Control {
priivate int volume;
public int brightness;
public static boolean initialized;

    public void setVolume(int theVolume) { volume = theVolume; }
}
```

**You be the compiler.** The following main is in a different class RemoteControlDriver. Indicate whether each statement it is valid or invalid. (Consider each one in isolation; earlier errors do not affect later statements.)

```java
public static void main( String[] args ) {
    RemoteControl.brightness = 10;         VALID / INVALID
    RemoteControl r = new RemoteControl(); VALID / INVALID
    r.volume = 10;                         VALID / INVALID
    RemoteControl.initialized = false;    VALID / INVALID
    r.setVolume(20.0);                     VALID / INVALID
    r.setVolume(30);                       VALID / INVALID
    RemoteControl r2 = new Control();     VALID / INVALID
}  
```

3
5. (4 pts) A **privacy leak** occurs when: (Select one)
   a. the program fails to deallocate storage for private class objects.
   b. a class constructor does not initialize all of its private instance variables.
   c. a public class method returns a reference to a private instance variable.
   d. your roommate scans your diary and uploads it to the Wiki.

   Answer: ______________

6. (10 pts) Suppose that
   - Classes **Cat** and **Dog** are declared in package **animal**.
     - **Cat** is a **public class**
     - **Dog** is declared to be a **class** (without specifying public or private).
   - Public class **Mouse** is declared in the subpackage **animal.small**.
   - Public class **Bob** is declared in package **human**.
   - None of these import any packages, except **Bob**, which contains:

     ```java
     import animal.*;
     ```

   All of these classes define a public static method called `print()`. True or false:
   a. Cat methods can call **Dog.print()** without using import.   TRUE / FALSE
   b. Cat methods can call **Mouse.print()** without using import.   TRUE / FALSE
   c. Bob methods can call **Cat.print()**.                   TRUE / FALSE
   d. Bob methods can call **Dog.print()**.                   TRUE / FALSE
   e. Bob methods can call **Mouse.print()**.                   TRUE / FALSE
Problem 2 (10 pts)

Draw a memory map for the local variables \((n, s1, s2)\) of the main method and the contents of the heap at the point where the execution reaches “STOP HERE”.

```java
public class Subject {
    public String name;
    public int age;

    public Subject() { name = "NO_NAME"; age = 20; }

    public Subject(String n, int a) { name = n; age = a; }

    public void setName(String s) { name = s; }

    public static void main(String[] args) {
        String[] n = {"Lester", "Earl"};
        Subject s1 = new Subject(n[1], 43);
        Subject s2 = new Subject();
        s1.setName(s2.name);
        // STOP HERE
    }
}
```

Heap

- **n**:
- **s1**:
- **s2**:

```
Problem 3 (25 pts)

The goal of this problem is to implement a Stack data structure for String objects. For our purposes, a Stack is a list of String references in which insertions and removals occur from the same end, called the top.

Give a complete definition for a class Stack that stores a stack of String references in an array. Your array must grow by doubling its size whenever it runs out of capacity (but it never shrinks). Your class must support the following public methods:

- **Stack()**
  Builds an empty stack. The initial array should have capacity to store one String.

- **boolean empty()**
  Returns true if the stack is empty, and false otherwise.

- **String pop()**
  Removes the string reference at the top of the stack and returns its value. You may assume that the stack is nonempty.

- **void push(String s)**
  Inserts string s at the top of the stack. If the array is currently full then a new stack array of twice the current size must be created before inserting the new item.

Hints and restrictions:
- **Hint:** your instance variables might consist of a String array stack for the stack and an index top which stores the index of the top element. (Be careful in initializing top.)
- **Your stack array must grow by doubling, but need never shrink.**
- **Other than String, you may not use any of the Java class library objects or methods.**

**START YOUR SOLUTION ON THIS PAGE AND CONTINUE ON THE NEXT.**
FINISH YOUR SOLUTION TO PROBLEM 2 ON THIS PAGE.
Problem 4 (20 pts)

Given an array of numbers, a subvector is any sequence of consecutive elements. The maximum subvector is the subvector having the highest sum of values. For example, consider the following array:

```
6  -8  6  5  -10  11  20  -19  -5  16
```

The maximum subvector consists of the elements {6, 5, -10, 11, 20}, which sum to 32.

Give a method `maxSubvector(vec)`, whose parameter `vec` is an array of type int, and which returns the maximum subvector sum. (In the example above this would be 32.)

**Hint**: Don’t try to find a shortcut (because you will probably be wrong). Just generate all possible subvectors, compute their sums, and return the maximum sum.

**Write your answer here:**
USE THIS PAGE IF YOU NEED EXTRA SPACE.