CS 100J Prelim 3   17 April 2007

This 90-minute exam has 6 questions (numbered 0..5) worth a total of 100 points. Spend a few minutes looking at all questions before answering any. Budget your time wisely. Use the back of the pages, if you need more space. We have a stapler at the front of the room, so you can tear the pages apart.

Question 0 (2 points). Write your netid and your name, legibly, at the top of each page (Hint: do it now).

Question 1 (15 points). Recursion. In assignment A1, you implemented a class Rhino that had, among other things, the fields shown in the partially completed class Rhino shown below. Write recursive function numberNoFather, whose specification and heading appear in the class below. For r of class Rhino, if r is **null**, r's ancestral tree is empty; if r is not null, its ancestral tree consists of r together with the ancestral trees of its mother and father.

Don't write any other methods; just write the body of numberNoFather.

```
/** An instance describes a rhino.*/
public class Rhino {
    private String name;   // The name of the rhino.
    private Rhino father; // This rhino's father (null if unknown)
    private Rhino mother; // This rhino's mother (null if unknown)

    …

    /** = the number of rhinos in  r's  ancestral tree whose father is not known.
          Precondition: r may be null. */
    public static int numberNoFather(Rhino r) {




    }
}
```

0 _____	out of	02
1 _____	out of	15
2 _____	out of	18
3 _____	out of	20
4 _____	out of	25
5 _____	out of	20
Total _____	out of	100

Question 2 (18 points). GUIs. Consider class GUI on the right, which, for the purposes of this question, has no comments.

(a) What are the default layout managers of containers JFrame, JPanel, and Box, and how many components can be added to each of these containers?

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

public class GUI extends JFrame {

  public GUI(int n){
    super("prelim 3");
    Container cp= getContentPane();
    Box box= new
                Box(BoxLayout.Y_AXIS);
    for (int k= 0; k < n; k= k+1) {
      box.add(addButtons(k+1));
    }
    cp.add(box);
    pack();
    setVisible(true);
  }

  public Box addButtons(int n){
    Box box= new
                Box(BoxLayout.X_AXIS);
    for (int i= 0; i < n; i= i+1){
      box.add(new JButton("" + (n—i)));
    }
    return box;
  }
}
```

(b) In the evaluation of

    new GUI(4)

how many instances of class Box are created?

(c) Below, draw the JFrame that results from evaluating the following expression. Important is not the shape and shading of components but the placement of components and the labels on the buttons.

    new GUI(3);.

Question 3 (20 points). Arrays and methods.
Complete function `extract`, given below. We have given the postcondition of a for-loop that we expect you to write; the for-loop should process a range of integers. Please write the invariant of the for-loop, the initialization of local variable `result`, the for-loop itself, and the expression to be returned in the return statement.

```
/** =  a list of (some) characters of String s, as specified below. The characters in the list are to be
        separated by commas, and the list should be surrounded by '[' (at the beginning) and ']'
        (at the end of the list).
```

Each integer in `b` is the index of a character to put in the final list, as illustrated by this example: Suppose `s = "abcdef"` and suppose b is the array `{1,2,1,0,3}`. Then the string to be returned is: `"[b,c,b,a,d]"`.

Precondition: each b[i] is in the range 0..s.length()–1. */
```java
public static String extract(String s, int[] b) {

String result=                                              ;
```

```
       // postcondition: result = "["  catenated with the characters of s whose indices are
                                  given by  b[0..b.length()−1], separated by commas
```

```java
    return                                              ;

}
```

Question 4 (25 points) Classes. A *superellipse* is a curve whose points `(x, y)` satisfy the equation:

$$|x/a|^n + |y/b|^n = 1$$

where `a > 0, b > 0, n > 0`.

An *ellipse* is a curve whose points (x, y) satisfy the equation:

$$(x/a)^2 + (y/b)^2 = 1$$

where a > 0 and b > 0. So, an ellipse is a superellipse with n = 2. The area of an ellipse is

Math.PI * a * b.

A *circle* is a curve whose points (x, y) satisfy the equation:

$$(x/r)^2 + (y/r)^2 = 1$$

where r > 0. So, a circle is an ellipse with r = a = b.

(a) Complete the three classes given on the next page by completing the bodies of all methods. Except for the constructor of class `superEllipse`, all methods bodies should consist of one statement.

(b) State whether the following is true or false: In a constructor of a subclass, the first statement must be a call of a constructor of the superclass. Explain your answer.

(c) What are the four kinds of variables in Java, and where is each declared?

(d) Suppose the following method is placed in class `SuperEllipse`. During evaluation of a call `SuperEllipse.lSearch(v, null)`, when are local variables k and se created?

```
/** = index of s in v (v.size if not in) */
public static int lSearch( Vector<SuperEllipse> v, Ellipse s) {
   for (int k= 0; k < v.size(); k= k+1) {
      SuperEllipse se= v.get(k);
      if (se == s) return k;
   }
   return v.size();
}
```

```java
public class SuperEllipse {
  // The 3 fields describe a
     super ellipse
       |x/a|^n + |y/b|^n = 1
  private double a;
  private double b;
  private double n;

  /** Constr: an instance is the
     superellipse

       |x/a|^n + |y/b|^n = 1

    Precond: a > 0, b > 0, n > 0 */
  public SuperEllipse(double a,
                      double b,
                      double n) {



  }

  public double getA() {



  }

  public double getB() {



  }

  public double getN() {



  }
  /** = "Superellipse: a = <a>,
        b = <b>,
        n = <n>" */
  public String toString() {




  }
}
```

```java
public class Ellipse extends SuperEllipse{
  /** Constructor: an ellipse

        (x/a)^2 + (y/b)^2 = 1

     Precondition: a > 0, b > 0 */
  public Ellipse(double a, double b) {


  }
  /** = area of ellipse: PI * a * b */
  public double area() {



  }

  /** = "Ellipse: a = <a>, b = <b>,
         area = <area>" */
  public String toString() {




  }
}
public class Circle extends Ellipse {
  /** Constructor: an ellipse

        (x/r)^2 + (y/r)^2 = 1

     Precondition: r > 0 */
  public Circle(double r) {


  }

  public double getRadius() {


  }

  /** = "Circle: radius = <radius>,
         area = <area>" */
  public String toString() {



  }
}
```

Question 5 (20 points) Known algorithms. Write a specification and header for a binary search function that searches for the rightmost occurrence (if present) of x in sorted array segment b[p..q−1]. Your specification may be in terms of pictures, English, mathematical notation, or a mixture of all of them. But it must be a specification of binary search *as we have presented the algorithm in class*.

Then, develop the body of the function. When developing its body, write the invariant for the loop first. Then develop the loop and initialization using the four loopy questions. An answer that does not have a suitable loop invariant will receive less than half credit.

Please note carefully that the sorted array segment to be searched is b[p..q−1].