

CS 100J Prelim 3

15 November 2005

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 beginning so that you can see what is expected. 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 (18 points).

(a) Suppose your friend wrote class `MyGUI`, below, but forgot to write complete specifications for the class and its methods.

```
public class MyGUI extends JFrame {
    /** The constructor: JFrame with title t
        and various components */
    public MyGUI(String t) {
        super(t);

        JPanel p= new JPanel();
        p.add(new JButton("yes?"));
        p.add(new JLabel("fun!!!"));
        p.add(new JButton("no?"));

        Box b= new Box(BoxLayout.Y_AXIS);
        b.add(new JTextField("Programming"));
        b.add(new JLabel("is"));
        b.add(p);

        Container cp= getContentPane();
        cp.add(b);

        pack();
        setVisible(true);
    }
}
```

0 _____ out of 02

1 _____ out of 18

2 _____ out of 15

3 _____ out of 24

4 _____ out of 20

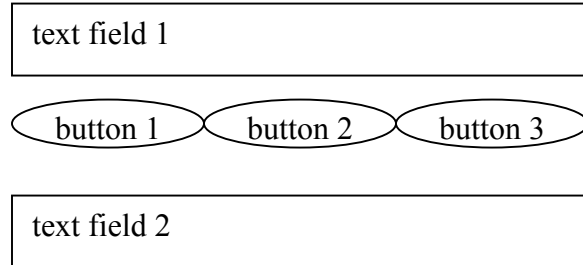
5 _____ out of 21

Total _____ out of 100

On the next page, draw an approximate representation of the GUI that is created by execution of this new-expression: `new MyGUI("wow!")`

If you think a written description of the GUI will help the grader, you can include it, but it is not necessary.

(b) Below, write a second the constructor, without parameters, for class `MyGUI` to produce the GUI shown on the right, using just the `BorderLayout` manager that comes with a `JFrame`. Write the constructor below. We realize that you may not remember the exact syntax for telling where to place a component; as long as the sense is right, you will receive credit.



```
/** Constructor: a JFrame as shown above and to the right */  
public MyGUI() {
```

```
}
```

Question 2 (15 points). Consider the following classes `Movie` and `Opinion`, which might be used by a movie rating website. Using the specifications, on the next page, write the body for method `getOverAllRating()` in the class `Movie`.

```
import java.util.*;
```

```
/** An instance is a movie with title and a
    list of opinions about the movie. */
public class Movie {
    // the title of the movie
    private String title;

    // opinions on the movie. Each
    element is of class Opinion
    private Vector opinions;

    /** Constructor: an instance of a
        movie with title t, no opinions */
    public Movie(String t) {
        title= t;
        opinions= new Vector();
    }

    /** = movie's title */
    public String getTitle() {
        return title;
    }

    /** = average of numerical ratings in
        Opinion instances in Vector
        opinions.
        (= -1 if opinions is empty).
        Precondition: opinions != null
        */
    public double getOverAllRating() {
```

```
/** An instance is an opinion about a movie. It
    consists of a rating in the range 1..10 and,
    possibly, a comment on the movie. */
public class Opinion {
    private int rating;
    private String comment;

    /** Constructor: an Opinion with rating r
        and no comment
        precondition: 1<= r <= 10 */
    public Opinion(int r) {
        rating= r;
        comment= "";
    }

    /** Constructor: an Opinion with rating r
        and comment c
        precondition: 1<= r <= 10 */
    public Opinion(int r, String c) {
        rating= r;
        comment= c;
    }

    /** = the rating */
    public int getRating() {
        return rating;
    }

    /** = the comment */
    public String getComment() {
        return comment;
    }
}
```

WRITE THIS METHOD
ON THE NEXT PAGE

```
    }
}
```

Cornell net id _____ Name _____

The following methods `Vector` might be useful (not all of them are), for `v` a `Vector` and `d` a `Double`:

Return	Method	Purpose
Object	<code>v.get(int k)</code>	= <code>v[k]</code>
int	<code>v.capacity()</code>	= the number of elements that are currently allocated for <code>v</code> 's list
int	<code>v.size()</code>	= the number of elements in <code>v</code> 's list
int	<code>v.indexOf(Object ob)</code>	= <code>i</code> , where <code>v[i]</code> is <code>ob</code>
Double	<code>d.doubleValue()</code>	= the value that <code>d</code> wraps
Double	<code>d.MAX_VALUE</code>	= the largest value of type <code>double</code>

```
/** = average of numerical ratings in the Opinion instances in Vector opinions.
```

```
    (= -1 if opinions is empty)
```

```
    Precondition: opinions != null */
```

```
public double getOverAllRating() {
```

```
    }  
}
```

Question 3 (24 points). This question uses classes Movie and Opinion of Question 2. Write the bodies of function findMax and procedure sortByRating() in class MovieDatabase, given below. Read the spec of sortByRating carefully! It gives additional requirement. on your solution.

/** An instance is a (very) simple database that contains Movie instances. */

```
public class MovieDatabase {
    private movie[] db;
```

/** Constructor: an instance of MovieDatabase, with a null database db */

```
public MovieDatabase() {
    db= null;
}
```

/** = the index of the movie in db[h..k] with the highest average rating */

```
private int findMax(int h, int k) {
```

```
}
```

/** Sort the movies in db by their average rating, as given by getOverAllRating().

The Movie with the highest average rating appears in db[0], followed by the Movie with the next highest average rating, etc.

You must: (1) write a selection sort algorithm.

(2) show the invariant of the only loop you write –as a picture, formula, or in English

(3) write a repetend that calls method findMax */

```
public void sortByRating() {
```

```
    }
}
```

Question 4 (20 points)

(a) Write a single statement that declares a **double** array `b` and initializes it to contain the five values 8, 5, 3, 2, 1, in that order.

(b) Write a while loop (with initialization) that is equivalent to the following for loop. Don't concern yourself with what the loop does.

```
for (k= b[i]; k > 0 && b[k] <32; k= k - 3) {  
    b[k-1]= k;  
}
```

(c) Rewrite the following statement so that it does not use an if-statement or conditional expression.

```
if (raining || snowing)  
    stayAtHome= false;  
else stayAtHome= ..true;
```

(d) Write down the steps in executing a call `m(5)` on the following method. Then, state when each of the variables `x`, `y`, and `z` are created during that call.

```
public static void m(int x) {  
    for (int y= 20; y != 0; y= y - 1) {  
        double z= k*k;  
        System.out.println(z + k);  
    }  
}
```

Question 5 (21 points). Gries is now in India and needed help converting from rupees to dollars and back. Someone wrote the classes shown at the bottom of the page –rather strange, because rupees are not usually thought of as a subclass of dollars, but that’s what we have to live with. Answer the following questions **You may not add any fields to the classes.**

There are 43.66 rupees in a US dollar.

- (a) Write the body of the constructor of class Dollar.
- (b) Write the body of the constructor of class Rupee.
- (c) Finish the return statement in function getRupee of class Rupee.
- (d) Draw an instance (folder) of class Rupee.
- (e) Suppose there are two variables f1 and f2, declared as follows: Dollar f1; Dollar f2; Indicate which of the following statements are legal:
 - a. f1= new Dollar(32);
 - b. f2= new Rupee(6);
 - c. String s1= f1.toString();
 - d. String s2= f2.toString();
 - e. double x= f2.getRupee();

```

/** an instance wraps a dollar value */
public class Dollar {
    private double f; // a value, in US dollars

    /** Constructor: an instance with
        dollar value f */
    public Dollar (double f) {

    }

    /** = this instance’s dollar value */
    public double getDollar() {
        return f;
    }
}

```

```

/** An instance wraps a rupee value */
public class Rupee extends Dollar {
    /** Constructor: an instance with rupee
        value c */
    public Rupee(double c) {

    }

    /** = this instance’s rupee value */
    public double getRupee() {

        return _____ ;
    }

    /** = string repr. of this instance */
    public String toString() {
        return "" + getRupee();
    }
}

```