Final Exam
Computer Programming 230
Dr. St. John
Lehman College
City University of New York
Thursday, 20 May 2010

NAME (Printed) ________________________________
NAME (Signed) ________________________________
E-mail _______________________________________

Exam Rules

• Show all your work. Your grade will be based on the work shown.
• The exam is closed book and closed notes.
• When taking the exam, you may have with you pens or pencils, and an 8 1/2" x 11" piece of paper filled with notes, programs, etc.
• You may not use a computer or calculator.
• All books and bags must be left at the front of the classroom during this exam.
• Do not open this exams until instructed to do so.

<table>
<thead>
<tr>
<th>Question 1</th>
<th>Question 2</th>
<th>Question 3</th>
<th>Question 4</th>
<th>Question 5</th>
<th>Question 6</th>
<th>Question 7</th>
<th>Question 8</th>
<th>Question 9</th>
<th>Question 10</th>
<th>TOTAL</th>
</tr>
</thead>
</table>

1
1. True or False:

(a) ___ In Alice and Java, if statements can be nested inside other if statements.
(b) ___ In Alice, an event only occurs as a result of user action.
(c) ___ The index of an array always starts with 0.
(d) ___ All variables in Alice and Java are global.
(e) ___ Some methods in Alice and Java are called automatically.
(f) ___ In Java, only components have the keyboard focus.
(g) ___ In Java, the method `size()` returns the number of elements in an array.
(h) ___ In Java, arrays cannot be parameters (inputs) to a method.
(i) ___ In Java, you can only read in files, not print to files.
(j) ___ In Java, all exceptions must be handled by the method that generates them.

2. Write the Java code that declares

(a) a integer `i` that holds the number 1:

(b) a double `tax` which is 7.75:

(c) a string `myName` that holds your name:

(d) an object `upperLeft` of the class `Point`:

(e) an array `friends` of 10 `Person` objects:
3. What happens when the code is run?

(a) world.my first method ()

No variables

// Programming Project 3.5

Do together

While ( both ( ( frog distance in front of square asSeenBy = ground ) > 0.25 )
and ( ( frog2 distance in front of square asSeenBy = ground ) > 0.25 ) )

Do together

camera move amount = 1 meter toward target = stop style = abruptly duration
= 1.5 seconds

frog.hop ( random number minimum = 0.25 maximum = 2.25 )

frog2.hop ( random number minimum = 0.25 maximum = 2.25 )

(b) toyBall1.bounce ()

fallDistance = ( toyBall1 distance above ground )

fallDistance set value to ( ( toyBall1 distance above ground ) - ( ( subject =
toyBall1's height ) / 2 ) )

Do together

toyBall1 move down fallDistance meters duration = 0.5 seconds style = abruptly
asSeenBy = ground

toyBall1 move forward 0.5 meters duration = 0.5 seconds style = abruptly

toyBall1 turn forward 1 revolution duration = 0.5 seconds style = abruptly

While ( fallDistance > ( ( subject = toyBall1's height ) / 2 ) )

fallDistance set value to ( ( fallDistance - ( ( subject = toyBall1's height ) /
2 ) ) )

Do together

toyBall1 move up fallDistance meters duration = 0.5 seconds style = abruptly
asSeenBy = ground

toyBall1 move forward .25 meters duration = 0.5 seconds style = abruptly
asSeenBy = ground

toyBall1 turn forward 0.5 revolutions duration = 0.5 seconds style = abruptly

Do together

toyBall1 move down fallDistance meters duration = 0.5 seconds style = abruptly
asSeenBy = ground

toyBall1 move forward .25 meters duration = 0.5 seconds style = abruptly
asSeenBy = ground

toyBall1 turn forward 0.5 revolutions duration = 0.5 seconds style = abruptly
4. What is the output of the following code fragments:

(a) Output:
```
int numtimes = -1;
while ( numtimes <= 0 )
{
    System.out.print("Hi!");
    numtimes++;
}
System.out.print("Bye!");
```

(b) Output:
```
boolean done = false;
int total = 1;
while ( !done )
{
    if ( total > 4 )
    {
        done = true;
    }
    total = total*2;
}
System.out.println(total);
```

(c) Output:
```
int i, j;
for ( i = 0 ; i < 3 ; i++ )
{
    for ( j = 0 ; j < i ; j++ )
    {
        System.out.print("+");
    }
    System.out.println();
}
```

(d) Output:
```
int i, j;
for ( i = 0 ; i < 6 ; i++ )
{
    for ( j = 0 ; j < 3 ; j++ )
    {
        if ( i%2 == 0 )
        {
            System.out.print("+");
        }
        else
        {
            System.out.print("-");
        }
    }
    System.out.println();
}
5. What is the output?

(a) if ( ( true ) && ( false ) )
    System.out.println("Yes");
else
    System.out.println("No");

Output: 

(b) boolean tobe = true;
    if ( tobe || !tobe )
        System.out.println("Yes");
else
    System.out.println("No");

Output: 

(c) int x = 1, y = 2, z = 3;
    if ( x+y*z < 10 )
        System.out.println("Yes");
else
    System.out.println("No");

Output: 

(d) int number = -6;
    boolean ispositive = ( number > 0 );
    boolean ismult3 = ( number % 3 == 0 );
    if ( ispositive || ismult3 )
        System.out.println("Yes");
else
    System.out.println("No");

Output: 

(e) int seconds = 120;
    if ( seconds%60 == 0 && seconds%3600 != 0 )
        System.out.println("Yes");
else
    System.out.println("No");

Output: 

6. Assume the following class definition:

```java
public class Mystery {
    public int number;
    public String message;
    public Mystery()
    { number = 3; message = "Hello"; }
    public String toString()
    { System.out.println(number+" "+message); }
    public void query()
    { int i;
        System.out.print(message);
        for ( i = 0 ; i < number ; i++ )
            System.out.print("!");
        System.out.println();
    }
}
```

and the following code has been executed:

```java
Mystery first = new Mystery();
Mystery second, third;
first.number = 2;
first.message = "Hi";
second = new Mystery();
second.number = 2*first.number;
third = first;
```

What is the output from the following statements?

(a) System.out.print(first);

Output: 

(b) first.query();

Output: 

(c) System.out.print(second);

Output: 

(d) second.query();

Output: 

(e) System.out.print(third);

Output: 

6
7. Examine the class below and answer the following:

(a) How many constructors does this class have?
(b) Does the panel have an associated action listener?
(c) Does the panel have an associated mouse listener?
(d) What does the `paintComponent()` method do, *in your own words*:

```java
public class DotsPanel extends JPanel {
    private final int SIZE = 6; // radius of each dot
    private ArrayList<Point> pointList;
    public DotsPanel() {
        pointList = new ArrayList<Point>();
        addMouseListener(new DotsListener());
        setBackground(Color.black);
        setPreferredSize(new Dimension(300, 200));
    }
    public void paintComponent(Graphics page) {
        super.paintComponent(page);
        page.setColor(Color.green);
        for (Point spot : pointList) {
            if (pointList.indexOf(spot) >= pointList.size()-10)
                page.fillOval((int)spot.getX()-SIZE, (int)spot.getY()-SIZE, SIZE*2, SIZE*2);
        }
        page.drawString("Count: " + pointList.size(), 5, 15);
    }
    private class DotsListener extends MouseAdapter {
        public void mousePressed (MouseEvent event) {
            pointList.add(event.getPoint());
            repaint();
        }
    }
}
```
8. (a) Write a \texttt{for}-loop that prints out the numbers from -5 to 0:
\begin{verbatim}
-5 -4 -3 -2 -1 0
\end{verbatim}

(b) Write a \texttt{while}-loop that reads characters from the \texttt{Scanner} object \texttt{line} while there are still characters on the line and prints out each character scanned on a separate line.

For example, if \texttt{hi mom} is entered, you should print:

\begin{verbatim}
 h
 i
 m
 o
 m
\end{verbatim}
You have just been accepted a job with a local bookstore. Your first assignment is to keep track of inventory of books at the store. Your predecessor, before quitting, began writing a Book class. Each of the methods of the class is proceeded by a comment that explains what the method should do. Fill in each method with the appropriate code:

```java
public class Book {
    public String title; // The title of the book
    public int numCopies; // Number of copies of the book
    public int numRequests; // Number of people requesting book
    public double price; // Price of book

    public Book(String t, int n, double p) {
        title = t; numCopies = n; price = p;
    }

    /* Prints all the information about the book: */
    public String toString() {
    }

    /* Calculates and returns the number of books available (ie the difference between numCopies and numRequests). */
    public int numAvailable() {
    }

    /* Returns true if there’s 1 or more books in stock, otherwise returns false*/
    public boolean inStock() {
    }
}
```
10. Create a new class called Rectangle that extends the abstract class BoundedShape below. Your Rectangle class should have two variables to store points and a constructor that takes two points and a color as input and stores them. You should also write a method draw() that draws a rectangle using the information stored in the class.

public abstract class Shape
{
    protected Color color;
    public void abstract draw(Graphics gc) { } }
public abstract class BoundedShape extends Shape
{
    protected Point upperLeft;
    protected int width, height;
    protected boolean filled;
    // Creates and returns a point representing the upper left corner of a bounding rectangle based on two points.
    protected Point determineUpperLeft(Point p1, Point p2)
    {
        int x = (int) Math.min(p1.getX(), p2.getX());
        int y = (int) Math.min(p1.getY(), p2.getY());
        return new Point(x, y);
    }
}