

Final Exam  
Computer Programming 230  
Dr. St. John  
Lehman College  
City University of New York  
Thursday, 17 December 2009

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.**

Question 1	
Question 2	
Question 3	
Question 4	
Question 5	
Question 6	
Question 7	
Question 8	
Question 9	
Question 10	
TOTAL	

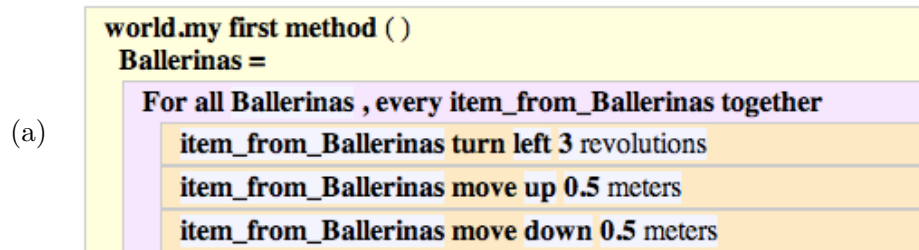
1. True or False:

- (a) \_\_\_ In Alice, the body of the loop can contain another loop, but only of the while type..
- (b) \_\_\_ In Alice, an event may occur only as a result of user action.
- (c) \_\_\_ In Alice, an index in the array always starts with 1.
- (d) \_\_\_ Some methods in Alice and Java are called automatically.
- (e) \_\_\_ In Java, to generate a keyboard event, a component must have the keyboard focus.
- (f) \_\_\_ In Java, Boolean isEmpty ( ) returns true if this list contains no elements.
- (g) \_\_\_ In Java, a variable can be used before it is declared.
- (h) \_\_\_ In Java, class hierarchies can contain no more than three levels.
- (i) \_\_\_ In Java, the content of the input file can only be read as text.
- (j) \_\_\_ In Java, all exceptions are considered checked exceptions, except objects of type RuntimeException.

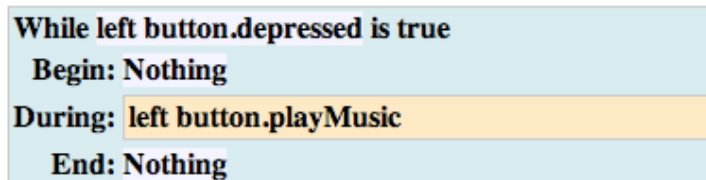
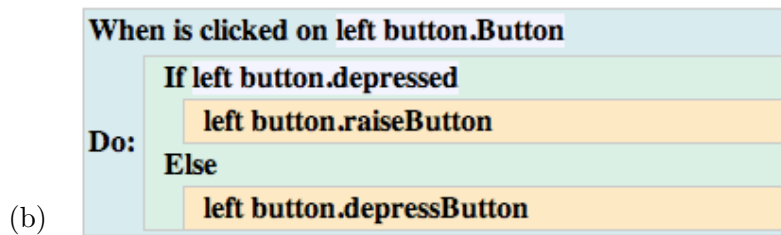
2. Write the Java code that declares

- (a) a variable `count` that holds the number 0:
- (b) a variable `score` which is 8.25:
- (c) a string variable `me` that holds your name:
- (d) an object `smiley` of the class `Circle`:
- (e) a list `ShapesToDraw` of 5 `Shape` objects:

3. What happens when the code is run?



## Events



4. What is the output of the following code fragments:

(a)

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

Output:

(b)

```
boolean done = false;
int total = 0;
while ( !done )
{
    if ( total > 3 )
    {
        done = true;
    }
    total = total + (total+1);
}
System.out.println(total);
```

Output:

(c)

```
int i, j;
for ( i = 0 ; i < 3 ; i++)
{
    for ( j = 0 ; j < 5 ; j++)
    {
        System.out.print("+");
    }
    System.out.println();
}
```

Output:

(d)

```
int i, j;
for ( i = 0 ; i < 3 ; i++)
{
    for ( j = 0 ; j < 5 ; j++)
    {
        if ( (i+j)%2 == 0 )
        {
            System.out.print("+");
        }
        else
        {
            System.out.print("-");
        }
    }
    System.out.println();
}
```

Output:

5. What is the output?

```
(a) if ( ( 1 <= 10) && ( 20 > 10 ) )  
    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 = 3, y = 3, z = 4;  
    if ( x+y*z > 15 )  
        System.out.println("Yes");  
    else  
        System.out.println("No");
```

**Output:**

```
(d) int number = 7;  
    boolean ispositive = ( number > 0 );  
    boolean iseven = ( number % 2 == 0 );  
    if ( ispositive || iseven )  
        System.out.println("Yes");  
    else  
        System.out.println("No");
```

**Output:**

```
(e) int year = 2004;  
    if (( year%4 == 0 && year%100 != 0 ) || ( year%400 == 0 ))  
        System.out.println("Yes");  
    else  
        System.out.println("No");
```

**Output:**

6. Assume the following class definition:

```
public class SampleClass {  
    public int number;  
    public String message;  
    public SampleClass()  
    { number = 0; message = "I love Java"; }  
    public void print()  
    { System.out.println(message); }  
    public void wonder()  
    {  
        int i;  
        for ( i = 0 ; i < number ; i++ )  
            System.out.print(message);  
    }  
}
```

and the following code has been executed:

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

What is the output from the following statements?

(a) `first.print();`

**Output:**

(b) `first.wonder();`

**Output:**

(c) `second.print();`

**Output:**

(d) `second.wonder();`

**Output:**

(e) `third.print();`

**Output:**

7. (a) Write a `for`-loop that prints out the even numbers from 10 to 0:  
10 8 6 4 2 0

- (b) Write a `while`-loop that reads characters from the `Scanner` object `line` while there are still characters on the line and prints out any digits from the `line`.

8. You have just been accepted a job with the Metropolitan Transit Authority (MTA). Your first assignment is to keep track of train departures. Your predecessor, before quitting, began writing a `Train` class. Each of the methods of the class is preceded by a comment that explains what the method should do. Fill in each method with the appropriate code:

```
public class Train
{
    public String start; //The starting location of the train
    public String end;   //The ending location of the train
    public int distance; //Distance travelled
    public int numCars;  //Number of cars on the train
    public int people;   //Number of people on the train

    public Train() {
        start = end = "Grand Central Station;
        distance = numCars = people = 0;
    }
    /* Prints all the information about the train: */
    public void print()
    {

    }

    /* Calculates and returns the number of passengers per car */
    public double carDensity()
    {

    }

    /* Calculates and returns the time of trip given the speed of train*/
    public double tripTime( double speed )
    {

    }

}
```



9. Create a new class called **Rectangle** that extends the abstract class **BoundedShape** below. Your **Rectangle** class should have a constructor that takes two points as input (and uses the **determineUpperLeft()** to store the upper left hand corner as well as the height and width. You should also write a method **draw()** that draws a rectangle to the screen with upper left point, **upperLeft**, and the given height and width.

```
import java.awt.*;
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);
    }
}
```

10. Write a **complete** Java program that asks the user for their full name, and then prints out their initials.

For example, if the user enters: `Herbert H Lehman`

Your program should print out: `The initials are HHL.`