

SAMPLE EXAM
Second Exam
Computer Programming 326
Dr. St. John
Lehman College
City University of New York
Thursday, 11 November 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.**

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) ___ Objects having the same interface can be used interchangeably.
- (b) ___ Only one class can implement any interface.
- (c) ___ A final class is always a base class.
- (d) ___ All methods are inherited by a derived class, including the private methods.
- (e) ___ You can create an object of an abstract class.
- (f) ___ Even if an exception occurs within a try block, the entire block is executed.
- (g) ___ A method that does not handle an exception must have a **throws** clause in a header.
- (h) ___ Objects of type implementable can be written to a binary file.
- (i) ___ You can write values of any primitive type to a binary file.
- (j) ___ A flow of data into or out of a program is called a stream.

2. (a) Define a Java interface and give an example.

(b) Circle all of the following that are included in an interface:

public	constructors	headings of	private instance
constants		public methods	variables

3. (a) Write the statement that will throw a new exception with the message “File Not Found”:

(b) Write a **try**-block that divides **x** by **y**:

(c) Write the first line of a java class called **Truck** than extends the class **Vehicle**:

(d) Write the first line of a java class called **Hexagon** than implements the interface **Measurable**:

4. Given the classes:

```
public class Mystery
{
    private int x;
    private String s;
    Mystery() { x = -10; s = "???"; }
    Mystery(int i) { x = i; s = "!!!"; }
    public String toString() { return(x+" "+s+" "+x); }
    public int get() { return (x); }
}

public class Enigma extends Mystery {
    private double code;
    Enigma() { super(); code = 3.1459; }
    Enigma(int i, double j){ super(i); code = j; }
    public String toString() { return(super.toString()+" "+code); }
}
```

Assume the following code has been executed:

```
Mystery first = new Mystery();
Enigma second = new Enigma();
Enigma third = new Enigma(5,10.01);
```

What is the output from the following statements?

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

Output:

(b) `System.out.print(second);`

Output:

(c) `System.out.print(third);`

Output:

(d) `System.out.print(first.get());`

Output:

(e) `System.out.print(second.get());`

Output:

5. Given the following program:

```
import java.util.*;
public class ModProgram {
    public static void main(String[] args) {
        System.out.println("Welcome to the modulus computer");
        Scanner reader = new Scanner(System.in);
        boolean done = false;
        while(!done){
            System.out.println("Enter two integer values");
            try{
                int first = reader.nextInt();
                int second = reader.nextInt();
                System.out.println(first+" mod "+second+" is "+(first % second));
            } catch (ArithmeticException e){
                System.out.println("Sorry, can not compute mod by 0");
            } catch (Exception e) {
                System.out.println("Sorry, you must enter two integer values");
                reader.nextLine();
            }
            System.out.println("Do another pair of values ? (y)");
            String response = reader.next().toLowerCase();
            if(!response.equals("y"))
                done = true;
        }
    }
}
```

(a) What is the output of the program if the user enters: 10 3?

(b) What is the output of the program if the user enters: 5 0?

6. Assuming `iStream` has been instantiated and linked to the file shown, what is the output of the following?

(a)	<u>File:</u>	<u>Output:</u>
<pre>while (iStream.hasNextLine()) { String line = iStream.nextLine(); line = line.substring(0,2); System.out.println(line); }</pre>	<pre>1 2 Buckle my shoe 3 4 Open the door</pre>	
(b)	<u>File:</u>	<u>Output:</u>
<pre>while (iStream.hasNextInt()) { int line = iStream.nextInt(); System.out.println(line); }</pre>	<pre>1 2 Buckle my shoe 3 4 Open the door</pre>	
(c)	<u>File:</u>	<u>Output:</u>
<pre>int count = 0; while (count < 3) { String line = iStream.nextLine(); count++; System.out.println(line); }</pre>	<pre>1 2 Buckle my shoe 3 4 Open the door</pre>	
(d)	<u>File:</u>	<u>Output:</u>
<pre>int count = 0; while (iStream.hasNextLine()) { String line = iStream.nextLine(); count= line.length(); System.out.println(count); }</pre>	<pre>1 2 Buckle my shoe 3 4 Open the door</pre>	

8. Create an abstract class `DiscountPolicy`. It should have a single abstract method `computeDiscount()` that will return the discount for the purchase of a given number of a single item. The method has two parameters, `count` and `itemCost`.

9. Derive a class `BulkDiscount` from `DiscountPolicy`, as described above. It should have a constructor that has two parameters, `minimum` and `percent`. It should define the method `computeDiscount()` so that if the quantity purchased of an item is more than `minimum`, the discount is `percent` percent.

10. Write a **complete** program that asks the user for the name of an input and output file and copies the contents of the input file to the output file.