American University of Science & Technology

Department of Computer Science

**CSI 311L – Java Programming Lab**

**Spring 2011/2012**

**Lab Work 7**

**Problem1:**

Trace the following java applications:

 a-

//Suppose that the generated numbers are: 14 25 46 45 39

public class Lab

{

 public static void main ( String a [] )

 {

 int x[] = new int [5];

 for( int i = 0; i < 5; i++ )

 x[i] = 1 + (int)(Math.random() \* 51);

 for( int i = 0; i < 5; i++ )

 System.out.println( x[i] );

 int c = 0;

 for( int i = 0; i < 5; i++ )

 if( x[i] % 2 == 0 )

 c++;

 System.out.println(

 "Percentage of Even Numbers: " + c \* 100 / x.length + "%" );

 System.out.println( "Percentage of Odd Numbers: " +

 ( x.length - c ) \* 100 / x.length + "%" );

 }

}

b-

class q8

{

 public static void main(String[] args)

 {

 int N = 1000 + 1;

 // Create Array

 Boolean p [] = new Boolean [ N ];

 // Initialize to TRUE

 for (int i=0;i<p.length ; i++)

 p[i] = true;

 for (int i=2;i<N ;i++)

 {

 // skip all FALSE elements

 if (p[i] == false)

 continue;

 // I removed all multiples of 'i'

 for (int j=i;j\*i<N ;j++)

 p[i\*j] = false;

 }

 // Print all prime numbers

 for (int i=1;i<N;i++)

 {

 if (p[i] == false)

 continue;// not prime

 System.out.println(i + " is prime!");

 }

 }

}

**Problem 2:**

write a java application to output the following shape:



**Problem 3:**

Write the class definitions and implementations for the following class specifications.

|  |
| --- |
| **Instructor** |
| **Attributes** | * id: an integer to identify the id of the instructor
* name: a String to identify the name of the instructor
* highestDegree: a String to identify the highest degree achieved by the instructor
* rate: a floating-point value to identify the rate per hour of the instructor
 |
| **Methods** | * Default constructor
* Constructor
* set methods
* get methods
* toString: to return information about an instructor
 |

|  |
| --- |
| **Course** |
| **Attributes** | * code: a String to identify the code of the course
* name: a String to identify the name of the course
* instructor: an object of class Instructor to identify the instructor of the course
* hours: an integer to identify the number of hours to be taught for this course
 |
| **Methods** | * Default constructor
* Constructor
* set methods
* get methods
* toString: to return information about a course
* compute: to compute the cost of the course
 |

1. Write a java application to do the following (Use JOptionPane methods):
* Create N objects of class Instructor and let the user initialize them
* Create N objects of class Course and let the user initialize them
* Compute and display the total cost of all courses
* Display information about courses taught by a specific instructor