LAB 3 And Exam Review

Problem 1: (Nested Loops)

Write a Java Console applications that asks the user to enter an integer between 1 and 20. (check by using an if statement if out of the range). The program should displays a triangle that contains rows of " *". It displays 1 asterisk on the first row, 2 on the second, 3 on the third row and so on...up to the number entered by the user. Then it goes in the reverse order by decreasing the number of asterisks by one per line back down to 1.

A Sample Run for n =5 is as follow:

```
*
**
***
****
*****
****
***
**
*
```

Solution:

```java
import java.util.*;

class triangle{
    public static int errorCheck(int num) {
        Scanner S = new Scanner(System.in);
        if(num <= 0 || num > 20) {
            System.out.print("Please input a valid number: ");
            num = S.nextInt();
            errorCheck(num);
        }
        return num;
    }

    public static int printTriangle(int num) {
        num = errorCheck(num);

        // print top half of isoceles triangle
        for(int i = 1; i <= num; i++) {
            for(int j = 1; j <= i; j++) {
                System.out.print("*");
            }
            System.out.println();
        }

        // print bottom half of isoceles triangle
```

```
        // which is a mirror of the top half minus one
        for(int i = num-1; i > 0; i--) {
            for(int j = 1; j <= i; j++) {
                System.out.print("*");
            }
            System.out.println();
        }
        return 0;
    }

    public static void main(String[] args) {
        Scanner S = new Scanner(System.in);
        System.out.print("Enter an integer from 1 to 20: ");
        int num = 0;
        // get user input for number between 1 and 50

        try{
            num = S.nextInt();
            printTriangle(num);
        }
        catch (InputMismatchException e) {
            num = errorCheck(num);
            printTriangle(num);
        }
    }
}
```

Problem 2: (Strings)

Write a full Java program that asks the user to type in 5 words, prompting the user for each word with a number. The program then reports the longest and the shortest word the user typed in.

Solution:

```
import java.util.*;
import java.lang.String;

class string {
    public static void main(String[] args) {
        Scanner S = new Scanner(System.in);
        String[] stringArray = new String[5];

        // user enters 5 words
        for(int i = 0; i < 5; i++) {
            int temp = i + 1;
            System.out.println("Input a word for the word #" + temp);
```

```
            stringArray[i] = S.next();
        }

        // find the longest word
        int indexForLong = 0;
        int elementLength = stringArray[0].length();
        for(int i = 1; i < 5; i++) {
            if(stringArray[i].length() > elementLength) {
                indexForLong = i;
                elementLength = stringArray[i].length();
            }
        }
        System.out.println("The longest word you entered is: " +
stringArray[indexForLong]);

        // find the shortest word
        int indexForShort = 0;
        elementLength = stringArray[0].length();
        for(int i = 1; i < 5; i++) {
            if(stringArray[i].length() < elementLength) {
                indexForShort = i;
                elementLength = stringArray[i].length();
            }
        }
        System.out.println("The longest word you entered is: " +
stringArray[indexForShort]);
    }
}
```

Problem 3: (Arrays)

Write a Java GUI Program that will store student ID, names and gpa in arrays, Then ask the user to enter a student ID, then search and display the student Info if it is found in the arrays, display an error otherwise

Solution:

```
import java.util.*;
import javax.swing.JOptionPane;

class arrays {
    public static void main(String[] args) {
        // declare and initialize the arrays
        int[] studentID = new int[] {1,2,3,4,5};
        String[] names = new String[] {"Bob","Karry","Karen","Sid","Megan"};
        double[] studentGpa = new double[] {3.4,4.0,1.4,2.0,4.0};

        int userStudentID = 0;
```

```
        // ask the users to enter a student ID
        userStudentID = Integer.parseInt(JOptionPane.showInputDialog(null, "Input a
student ID"));
        // conditional to see if the studentID is found in array
        for(int i = 0; i < studentID.length; i++) {
            if(userStudentID == studentID[i]) {
                JOptionPane.showMessageDialog(null, "The student info is as follows: "
+ names[i]);
                JOptionPane.showMessageDialog(null, "Name: " + names[i]);
                JOptionPane.showMessageDialog(null, "GPA: " + studentGpa[i]);
                JOptionPane.showMessageDialog(null, "ID: " + studentID[i]);
            }
            else if(studentID[i] == studentID.length - 1) {
                JOptionPane.showMessageDialog(null, "Error: Unable to find student
ID");
            }
        }
    }
}
```

Problem 4: (Bubble Sort Arrays)

Write a Java Program that ask the user to enter the size of an integer array then all the elements of the array, then apply the Bubble sort algorithm, the Program should display the array before and after the sort.

Solution:

```
import java.util.*;

class bubbleSort {
    public static void main(String[] args) {
        Scanner S = new Scanner(System.in);

        // Get user input for the size of the array
        int sizeOfArray = 0;
        System.out.print("What is the size of the array going to be: ");
        sizeOfArray = S.nextInt();

        // declare the array of size of user input
        int[] bubbleArray = new int[sizeOfArray];
        // run through the array for user input
        for(int i = 0; i < sizeOfArray; i++) {
            System.out.println("Enter an integer number: ");
            bubbleArray[i] = S.nextInt();
```

```
        }
        // display the array by running through it
        for(int i = 0; i < sizeOfArray; i++) {
            if(i == sizeOfArray - 1) {
                System.out.print(bubbleArray[i] + " ");
            }
            else{
                System.out.print(bubbleArray[i] + ", ");
            }
        }
        // run through the bubble sort algorithm
        int temp = 0;
        for (int i = 0; i < sizeOfArray; i++) {
            for (int j = 1; j < (sizeOfArray - i); j++) {
                if (bubbleArray[j - 1] > bubbleArray[j]) {
                    temp = bubbleArray[j - 1];
                    bubbleArray[j - 1] = bubbleArray[j];
                    bubbleArray[j] = temp;
                }
            }
        }
        System.out.println();
        // display the new sorted algorithm
        for(int i = 0; i < sizeOfArray; i++) {
            if(i == sizeOfArray - 1) {
                System.out.print(bubbleArray[i] + " ");
            }
            else{
                System.out.print(bubbleArray[i] + ", ");
            }
        }
        System.out.println();
    }
}
```