Assignment- 07

Discussing the Fundamental Concepts of Recursion [CO 5]

| Deadline:

Instructions:
*For every task you need to show **tracing/simulation**, codes and final output. If tracing/simulation is missing, half of the marks will be deducted. Try to maintain sequence. Write name, student id, assignment number and date of submission clearly*.

Task 1

Simulate, **write** code and **show** final output for Binary Searching

I. Using Iterative method

II. Using Recursive method

[Take an array of your choosing for showing simulations and final output]

Task 2

Write a recursive program which sums the first ten positive numbers and **simulate** your code for any input.

Task 3

Trace the following Code showing each step. What will be the **output**?

```
public class fibonacci {

        static int fib(int n) {
                if (n <= 1){
                return n;
                }
                else{
                return fib(n-1) + fib(n-2);
                }
        }

        public static void main (String args[]) {
                int n = X; // X={(last 4 digit of your
                id%3)+ (last 3 digit of your id%4)+2}
                System.out.println(fib(n));
        }
}
```

Task 4

Trace the following Code showing each step. What will be the **output**?

```
// Java program to find factorial of given number

public class Test
{
// method to find factorial of given number
        static int factorial(int n) {
                if (n == 0) {
                return 1;
                }else{
                return n*factorial(n-1);
        }
}
```

```
        // Driver method
        public static void main(String[] args) {
                int num = X; // X={(last 4 digit of your
                id%3)+ (last 3 digit of your id%4)+2}

                System.out.println("Factorial of "+
                num + " is " + factorial(num));
        }
}
```

Task 5

Trace the following Code showing each step. What will be the **output**?

```
public class Test{

/* Function to calculate x raised to the power y */
        static int power(int x, int y) {
                if (y == 0){
                return 1;
                }else if (y % 2 == 0){
                return power(x, y / 2) * power(x, y / 2);
                }else{
                return x * power(x, y/2) * power(x, y/2);
                }
        }
```

```
/* Program to test function power */

        public static void main(String[] args) {
                int x = {last 4 digit of your
                id%5)+2};

                int y = {last 3 digit of your
                id%3)+1};

                System.out.printf("%d",
                power(x, y));
        }
}
```

Task 6

```
public static int mystery3(int n) {
        if (n < 0){
                return -mystery3(-n);
        }else if (n < 10){
                return n;
        }else{
                return mystery3(n/10 + n % 10);
        }
}
```

For each call below, **indicate** what value is returned with showing the steps:

mystery3(6) _____

mystery3(17) _____

mystery3(259) _____

mystery3(-479) _____

Task 7

```
public static void mystery4(String s) {
        if (s.length() > 0) {
                System.out.print(s.charAt(0));
                if (s.length() % 2 == 0){
                mystery4(s.substring(0, s.length() -1));
                }
                else{
                System.out.println("**");
                System.out.println("#"+ s.length()+ s.length()+"*");
                mystery4(s.substring(1, s.length()));
                System.out.print(s.charAt(s.length() - 1));
                }
        }
}
```

For each call below, **indicate** what output is printed with showing the steps:

mystery4("") _____

mystery4("a")_____

mystery4("ab") _____

mystery4("bc") _____

mystery4("abcd") _____