CSCI 1112 Algorithms and Data Structures

Lab 4 – Recursion

Part 1: Recursion vs. loops (4 points)

a) Write a non-recursive method called sumover that takes one argument n, which is a non-negative integer. The method returns a double value, which is the sum of the reciprocals of the first n positive integers: 1/1+1/2+1/3+……+1/n

For example, sumover(1) returns 1.0 (1/1); sumover(2) returns 1.5 (1/1 +1/2); sumover(3) returns approximately 1.833 (1/1 +1/2 + 1/3).

b) Write a method called recursiveSumover which performs the same operation as sumover using recursion. Define recursiveSumover(0) to be zero, which is the base case for recursion.

Part 2: More recursion (8 points)

a) Write a recursive method called *descending(int n)* that prints out the numbers from 1 to n in decending order. For example *descending(5)* prints out the following:

```
5
4
3
2
1
```

b) Write a recursive method called *triangle(int n)* that prints out a pattern of n lines of asterisks. The first line contains n asterisks, the second line contains (n-1) asterisks, and so on, up to the nth line, which contains 1 asterisk. For example, *traingle(5)* prints the following pattern:

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

Part 3: Recursive Graphics (8 points)

a) Download StdDraw.java and Fractals.java. Run the main method in Fractals.java and observe the pattern as you vary the value of N.

b) Write a method called DrawRecursiveTriangle(….) that produces the following pattern. Use the provided **drawTriangle(double x, double y, double size)** method which draws a single triangle of the specified size centered at x and y.