**Computer Science 2**   **Lab # 07**



**Dr. Hanh Pham**

**Student Last Name: Martinez Student First Name: Adriel**

**CS2 Section # 01**

**Due:** Problem A by the **end of the lab** and Problems B by the end of **Saturday** of the same week.

**TOPIC:**

**Project A:**

**Problem Description:**

1) Problem A: MyProgrammingLab # 71045 (chapter 18, Programming Projects)

**Analysis:**

(Describe the problem including input and output in your own words. Type your answer in the following with **BLUE font color**)

|  |
| --- |
| **INPUT: The user will enter in a positive integer for which that integer will go through a recursive sum series method.**  **OUTPUT: The output will be the individual printing of each series up to the integer that the user inputted. If the user enters 5, then the recursion method will be called 5 times and the program will print out the first 5 sum series in subsequent order. The output also will be formatted in a way that will only show the first 2 decimal places of the value.** |

**Design:**

(Describe the major steps for solving the problem. Type your answer in the following with **BLUE font color**)

|  |
| --- |
| **The major steps for solving this problem was determining how each series will be printed out all at one time, for instance if the user entered 20, all the series from 1 to 20 will be printed in order from least to greatest. To achieve this, I used a do-while loop and created a new variable, i, to use as a marker for each time I would go through the recursive method. So as the do-while loop continued to iterate, it would print out the corresponding series value based on i, and continue until i is less than or equal to the inputted value.** |

**Coding:** (Copy and Paste Source Code here. Type your answer in the following with **BLUE font color**)

|  |
| --- |
| **import java.util.Scanner;**  **public class SumSeries{**  **public static void main(String[] args){**  **Scanner input= new Scanner(System.in);**  **System.out.print("Enter the last positive integer you want m(i) to be calculated for:");**  **int n= input.nextInt();**  **int i=1;**  **do{**  **System.out.print("m("+i+") = ");**  **System.out.printf("%.2f",series(i));**  **System.out.println();**  **i++;**  **}while(i<=n);**  **}**  **public static double series(int x){**  **if(x==0){**  **return 0.0;**  **}**  **else**  **return (1.0/x+series(x-1));**  **}**  **}** |

**Testing:** (Describe how you test this program. Type your answer in the following with **BLUE font color**)

|  |
| --- |
| **RUN three times, using the same input as in the sample runs:**  **Test 1:**  **Enter the last positive integer you want m(i) to be calculated for:50**  **m(1) = 1.00**  **m(2) = 1.50**  **m(3) = 1.83**  **m(4) = 2.08**  **m(5) = 2.28**  **m(6) = 2.45**  **m(7) = 2.59**  **m(8) = 2.72**  **m(9) = 2.83**  **m(10) = 2.93**  **m(11) = 3.02**  **m(12) = 3.10**  **m(13) = 3.18**  **m(14) = 3.25**  **Test 2:**  **Enter the last positive integer you want m(i) to be calculated for:30**  **m(1) = 1.00**  **m(2) = 1.50**  **m(3) = 1.83**  **m(4) = 2.08**  **m(5) = 2.28**  **m(6) = 2.45**  **m(7) = 2.59**  **m(8) = 2.72**  **m(9) = 2.83**  **m(10) = 2.93**  **m(11) = 3.02**  **m(12) = 3.10**  **m(13) = 3.18**  **m(14) = 3.25**  **m(15) = 3.32**  **m(16) = 3.38**  **m(17) = 3.44**  **m(18) = 3.50**  **m(19) = 3.55**  **m(20) = 3.60**  **m(21) = 3.65**  **m(22) = 3.69**  **m(23) = 3.73**  **m(24) = 3.78**  **m(25) = 3.82**  **m(26) = 3.85**  **m(27) = 3.89**  **m(28) = 3.93**  **m(29) = 3.96**  **m(30) = 3.99**  **Test 3:**  **Enter the last positive integer you want m(i) to be calculated for:28**  **m(1) = 1.00**  **m(2) = 1.50**  **m(3) = 1.83**  **m(4) = 2.08**  **m(5) = 2.28**  **m(6) = 2.45**  **m(7) = 2.59**  **m(8) = 2.72**  **m(9) = 2.83**  **m(10) = 2.93**  **m(11) = 3.02**  **m(12) = 3.10**  **m(13) = 3.18**  **m(14) = 3.25**  **m(15) = 3.32**  **m(16) = 3.38**  **m(17) = 3.44**  **m(18) = 3.50**  **m(19) = 3.55**  **m(20) = 3.60**  **m(21) = 3.65**  **m(22) = 3.69**  **m(23) = 3.73**  **m(24) = 3.78**  **m(25) = 3.82**  **m(26) = 3.85**  **m(27) = 3.89**  **m(28) = 3.93** |