CIS 5100:

Homework 10

Data Structures & Program

By: Bernice Templeman

10/26/2015

Btempleman2015@my.fit.edu

**18.7 What does the following code do?**

/\*

mystery is a public non-static method that receives two integers and returns an integer.

mystery is a recursive method that multiples two integers and returns the product as an integer.

because it is public , it gives access to clients from outside of the client … the method can be called.

because it is non-static, you can not call it without first creating an object of the class where the method is declared.

Please see additional comments added to the code:

Example output for: **int** c = *mystery*(3,4); **int** c = *mystery*(3,4);

12

1st time mystery is called: a = 3 b = 4 //returns 3 + mystery(3,4-1)= 3+9=12

2nd time mystery is called: a = 3 b = 3 //returns 3 + mystery(3,3-1)= 3+6= 9

3rd time mystery is called: a = 3 b = 2 //returns 3 + mystery(3,2-1)= 3+3= 6

4th time mystery is called: a = 3 b = 1 //returns 3

3+3+3+3 = 3 \* 4 = 12

\*/

// recursive declaration of the method mystery

**public** **int** mystery(**int** a, **int** b)

{

// test for base case

**if** (b == 1) // base case

**return** a;

**else** // recursion step

**return** a + *mystery*(a, b - 1);

}

**18.12 What does the following program do?**

**Output:**

**Result is: 55**

Description: Main method calls Recursive method to sum the integers in the array

Summary:

1st time: size =10,element[10-1]=10,returns 10+ 45 = 55

2nd time: size = 9, element[9-1]=9, returns 9 + 36 = 45

3rd time: size = 8, element[8-1]=8, returns 8 + 28 = 36

4th time: size = 7, element[7-1]=7, returns 7 + 21 = 28

5th time: size = 6, element[6-1]=6, returns 6 + 15 = 21

6th time: size = 5, element[5-1]=5, returns 5 + 10 = 15

7th time: size = 4, element[4-1]=4, returns 4 + 6 = 10

8th time: size = 3, element[3-1]=3, returns 3 + 3 =6

9th time: size = 2, element[2-1]=2, returns 2 +1 =3

10th time:Size = 1, element[0]=1, returns 1

**public** **class** CIS5100\_Ch18\_12 {

//Description: Recursive method sums the integers in the array

// mystery receives an array of integers and the size of the array (int)

**public** **static** **int** mystery(**int**[] array2, **int** size)

{

// test for base case

**if** (size == 1) //base case

// returns the first element

**return** array2[0];

**else //recursion step**

**return** array2[size - 1] + *mystery*(array2, size -1);

}

// main method that calls the recursive method

**public** **static** **void** main(String[] args)

{

// initializer list specifies the initial valuefor each element

**int**[] array = {1,2,3,4,5,6,7,8,9,10};

// declare result as an int and set it to sum of the array values

**int** result = *mystery*(array, array.length);

// output the result

System.***out***.printf("Result is: %d%n", result);

}

}