Lab 11

Recursive Function

**Exercise 1**

Let’s assume you have to write a program which will print the first ten multiples of a given number. For example, if the number is 12, you have to print:

12

24

…

120

The task can be done by using a for loop, however, you want to try something new, Recursive function! Remember, the recursive function keeps calling itself until it reaches the base condition. For this exercise the base condition can be 1 or 10 depending on the initial multiplier value. If you call the function first time with 10, then the base condition is 1 or vice versa. A sample function can be:

void printMultiples(int givenNum, int multiplier) {

// check the base condition and return if it matches

// otherwise print the multiple and then call the function again.

}

*Hint: Call the function first time from main function with multiplier value 10 and inside the print function, call itself again with multiplier value decreased by 1, until it reaches 1.*

**Exercise 2**

Now try to write a recursive program to find the factorial of a given number. For example, if the given number is 10, then the factorial is:

1 \* 2\* 3\* 4\* ...\*10 = 3628800

While testing your program, don’t give big number as input, as the factorial value will be a fairly large number, chances of going out of memory is high. You may safely try it with numbers 1-10. Instead of *int* type, you may try *double* to store the factorial value, as its memory size is twice bigger than int.

**Sample Output:**

Once you are sure that the function is working fine, try to print it in following fashion:

10 \* 9 \* 8 \* 7 \* 6 \* 5 \* 4 \* 3 \* 2 \* 1 = 3628800

Then try to print in reverse order:

1 \* 2 \* 3 \* 4 \* 5 \* 6 \* 7 \* 8 \* 9 \* 10 = 3628800

**Big-O Notation**

Answer the following question and turn in a file:

1. What are the Big-O complexities of the exercise 1 and 2 of Lab 7? Briefly explain your answer.