

Date:

**Objectives:**  
 In this lab, student will be able to:

1. Understand concept of recursion
2. Write recursive programs

**Introduction**  
 A recursive function is a function that invokes/calls itself directly or indirectly.

### Steps to Design a Recursive Algorithm

- Base case:
  - for a small value of  $n$ , it can be solved directly
- Recursive case(s)
  - Smaller versions of the same problem
- Algorithmic steps:
  - Identify the base case and provide a solution to it
  - Reduce the problem to smaller versions of itself
  - Move towards the base case using smaller versions

### Solved exercise

Code snippet explaining concept of recursive functions

```
#include <iostream.h>
long factorial (long a) {
    if (a == 0) //base case
        return (1);
    return (a * factorial (a-1));
}
void main () {
    long number;
    cout << "Please type a number: ";
    cin >> number;
    cout << number << "! = " << factorial (number);
}
```

### Lab exercises

With the knowledge of modularization, function definition, function call etc.,

Write C++ programs as specified below:

### Recursive Functions

Write a recursive function, **GCD** to find the GCD of two numbers. Write a main program which reads 2 numbers and finds the GCD of the numbers using the specified function  
 Ex: GCD of 9,24 is 3.

- ✓ 2. Write a recursive function **FIB** to generate  $n^{\text{th}}$  Fibonacci term. Write a main program to print first N Fibonacci terms using function FIB. [Hint: Fibonacci series is 0, 1, 1, 2, 3, 5, 8, ...]

### Additional exercises

- ✓ 1. Write a recursive function **Print** to print 1 to N numbers using recursion. Do not use loops inside the function. Write a main function to test this.
- ✓ 2. Write a program to multiply two numbers using a recursive function.  
[Hint: Multiplication using repeated addition]