**Recursive Functions (A function calls itself)**

**3 points**

**Divide a big problem into smaller subproblems**

**Output of prev step will be combined with the current step output**

**There has to be a base condition where we have to stop**

static int Fib(int n)

{

if(n<=1) return 1;

else

return Fib(n-1) + Fib(n-2);

}

Fib(10)

Fib(9) + Fib(8)

Fib(9)

Fib(8) + Fib(7)

---------------------------------------------------------------------------------------------------

10 to 1 1 to 10

function recurSum(n):

if n <= 1:

return n

return n + recurSum(n - 1)

end function

-------

input n

sum = recrSum(10)

print sum

---------------------------------------------------------------------------------------------------

public static void Main(string[] args)

{

Console.WriteLine (Sum(10));

}

static int Sum(int n)

{

if(n<=1) return 1;

else

return n + Sum(n-1);

}

---------------------------------------------------------------------------------------------------

Factorial Fac(5) 5 \* 4 \* 3 \* 2 \*1

static int Factorial(int n)

{

if(n==1)

return 1;

else

return n \* Factorial(n-1);

}