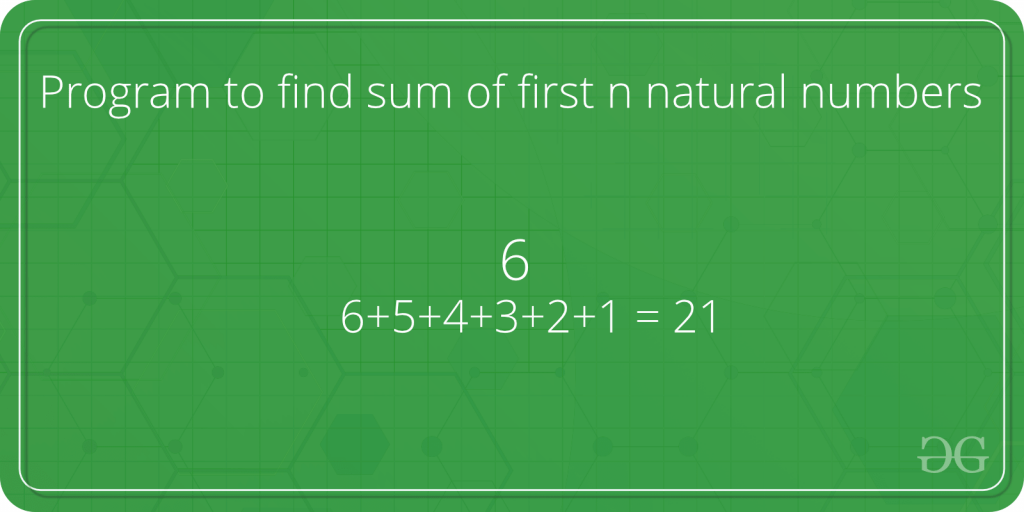
**Natural Number Sum using Recursion**



Given a number n, find sum of first *n* natural numbers. To calculate the sum, we will use a recursive function recur\_sum().  
**Examples :** 

Input : 3

Output : 6

Explanation : 1 + 2 + 3 = 6

Input : 5

Output : 15

Explanation : 1 + 2 + 3 + 4 + 5 = 15

Below is code to find the sum of natural numbers up to n using recursion : 

C++Java

// Java program to find the

// sum of natural numbers up

// to n using recursion

import java.util.\*;

import java.lang.\*;

class GFG

{

// Returns sum of first

// n natural numbers

public static int recurSum(int n)

{

if (n <= 1)

return n;

return n + recurSum(n - 1);

}

// Driver code

public static void main(String args[])

{

int n = 5;

System.out.println(recurSum(n));

}

}

**Output :**

15

**Time complexity :** O(n)

**Auxiliary space :** O(n)

To solve this question , **iterative approach** is the best approach because it takes constant or O(1) auxiliary space and the time complexity will be same O(n).