Programming Construct: Intro to Recursion and Problem Solving on Recursion (Pre-read)

Relevel

by Unacademy

Introduction

Let me introduce you to recursion, one of the powerful algorithmic techniques in which the function calls itself. This allows the function to be repeated several times because it calls itself during its execution.

Here the function calls itself until the specified condition is met.

Every recursion function must have at least two cases: the recursive and base cases.

Why we use recursion?

We use recursion to solve problems that can be broken down into smaller repetitive problems. It is good to work on things with many possible branches and are too complex for an iterative approach.

One good example of this would be searching through the file system.

PMI (Principle of Mathematical Induction)

Principle of mathematical induction is a method using which we can prove a lot of mathematical formulas. For example let's try to prove that the sum of first N natural numbers is N*(N+1)/2 .

To prove this: we will be doing 3 things:

1. We will figure out the smallest value of N for which we already know the answer.
2. Then we will assume that the formula works for some value of N = k
3. Then we will prove that formula works for some value of N = K+1 or N = K-1

Now we can see that for N = 1, we already know the answer, i.e. 1 and we can see the formula works here i.e. (1)*(1+1)/2 = 1 . This is the smallest value of N for which we already know the answer. This is called the base case.

Now we will assume that for N = k, the formula works correctly, this is going to be an assumption that we will take

Now we will try to prove this for N = k + 1,

If the formula works fine for N = k, then sum is (k)*(k+1)/2

Then, to calculate the sum of first k+1 natural numbers, we can say that it will be equal to the sum of first n natural numbers + (k+1).

I.e. (k)*(k+1)/2 + (k+1) = ((k+1)*(k+2)) / 2 , viz equal to the answer using the formula for N = k+1

What we will do in the class?

1. Recursion
2. Types of recursion
3. Problem on each type
4. Array sorting using recursion
5. Fibonacci series using recursion
6. Factorial using recursion
7. Friends pairing using recursion
8. Printing number in increasing order.

Upcoming Teaser

- Introduction to searching
- Linear search
- Binary search
- Linear search vs binary search
- Variants of binary search