DSA And CP 90 Days

Let us Do It

Resources, Notes, Questions, Solutions

27 Oct

2022

This document provides a complete guide to understanding data structures and algorithms in computer science. It contains a collection of information compiled by people who have successfully completed interviews at FAANG companies as well as a useful information, tips, graphics, etc. to help better understand the subject matter and be a better coder or to ace that interview.

As with all 30DaysCoding resources, this guide is provided to you with the mission of

making the world’s resources more accessible to all.

Enjoy!

**Table of Contents**

The document is divided as follows.

Table of Contents 2

[Starting 90 days plan for DSA 3](#_bookmark0)

[Recursion 3](#_bookmark3)

[Introduction 3](#_bookmark4)

[Examples and codes 4](#_bookmark5)

# Starting 90 Days Plan For Dsa

Resources, Notes, Questions, Solutions

27th Oct with Love Babber…

Starting the journey with full bucket of energy. Please almighty help me to complete me this course and get placed in a FAANG companies so that I can complete my dreams...

**Recursion**

## 

## Recursion

* What is Recursion?

When a function calls itself within its self-body it is called as recursion. Or When a big problem depends upon same type of small problems to solve can be solved by recursion.

E.g. –

If we take the example of 2n

2n = 2 \* 2n-1

So here 2n & 2n-1 are same type of sub problems & can be solved by recursion

Here the recurrence relation is

F(n) = 2 \* f(n-1)

E.g. –

If we take the factorial also it is a recurrence function

5! = 5 \* 4 \* 3 \* 2 \* 1

5! = 5 \* 4!

So, there is also recurrence applied

F(n) = n \* f(n-1)

We need a base case to solve a problem in recurrence problem base case decides when we must stop the recursion.

In the base case the return is compulsory.

Check the Below factorial code file

If there is not a base case, then there is a segmentation fault. Because of the infinity time running of the recursive function and till the memory stack of the program is not full. It works as a call stack and return the value to the parent function and chaining likely

// Factorial Func

int **Factorial** (int n){

    return n==0?1:n\***factorial**(n-1);

}

Q. Fibonacci Series with recursion –

Fibonacci is a infinite series that contains the past value of the current member.

1+2 = 3, 2+3 = 5, 3+4 = 7 …. etc

    // Fibonacci Series

    int **Fibonacci** (int n){

        // base case

        // int Ans = Fibonacci(n-1) + Fibonacci(n-2);

        return n==0? 0: n==1? 1: **Fibonacci**(n-1) + **Fibonacci**(n-2);

    }

Q. Return the digits in word form from a number

Given – 7638 Ans – seven six three eight

// Say the numbers in word

string **SayDig** (int num) {

    string arr[] = {"one","two","three","four","five","six","seven","eight","nine"};

    if(num == 0) return "";

    else {

**SayDig**(num/10);

        cout**<<**arr[(num%10) -1]**<<**" ";

    }

}