Hello Everyone!!

Class will start at 7:10 IST

Introduction

Sakshi Sachdeva

- 💼 Technical Lead at Qorix | Ex - ZF TCI | Ex - Bosch
- 📚 BTech from NIT Jalandhar.
- 👩‍🏫 Remote Instructor at Scaler
- 📍 Hail from Patiala,Punjab | Current residing in Hyderabad

Few terms that you shall see/hear throughout the course!

1. PSP (Problem Solving Percentage) - Solved Assignment Problems / Total Open Assignment Problems

- There are two types of section - Assignment and Additional. Assignment section consists of implementation of the problems done in class. PSP is calculated based on only Assignment Problems

- Additional Problems are slight modifications of assignment problems, they are not part of PSP but once you're done with assignment, we highly recommend to complete additional problems as well.

- Try to keep PSP least 80% no matter what. It shall really help you stay focused and we have seen in the past that people with ***>= 80%***, do well in interviews.

2. Attendance

- Try to maintain at least ***85% attendance*** either through classes or by watching recordings.

- Though I will recommend you to come to classes regularly because otherwise it may create backlogs.

- So, I expect all of you to attend live classes and if for any reasons you are unable to, then please send me a message stating the reason.

Schedule for next one month

- Introduction to Problem Solving
- Time Complexity
- Introduction to Arrays
- Prefix Sum
- Carry Forward & Subarrays
- Sliding Window & Contribution Technique
- Memory Management
- Sorting Basics
- 2D Matrices
- Bit Manipulation Basics
- Strings Basics
- Interview Problems
- Contest [covers full Intermediate DSA]

Contest will be organised after Intermediate Module

- It will be for 1.5 hours and will be conducted within class duration followed by Contest Discussion (Instructor shall be discussing contest problems).
- If for any reason you are unable to pass the contest, then we shall be having re-attempts as well, therefore nothing to worry. (We will provide more info on re-attempts and the passing criteria moving forward. Most likely, there will be 4 questions out of which 3 needs to be solved)
- Contests are critical to retain what you have learnt and measuring where you need improvement. Please take contests seriously.

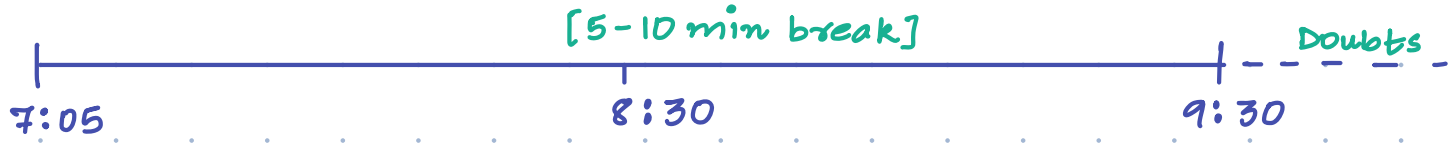
FAQs

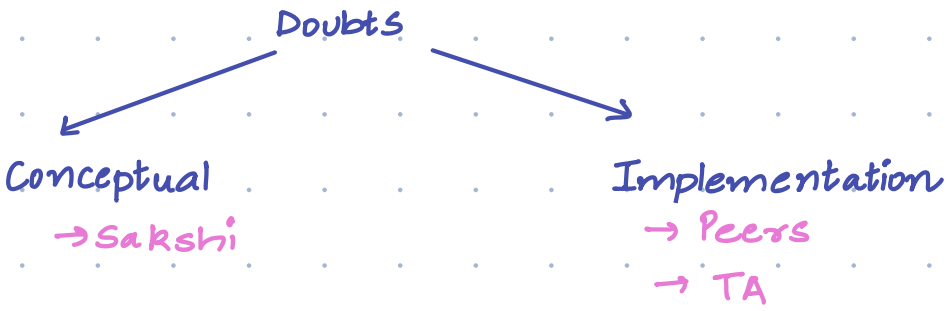
- Notes will be uploaded after the class.
- Assignment will be unlocked after the class ends.
- There is no deadline for assignments.

Lecture Days :

Monday
Wednesday
Friday

7:00 AM IST
We start at sharp 7:05





Mode OF Communication : Whatsapp
→ Announcement Group
→ Discussion Group
(Links on Notice Board)

Introduction to Problem Solving

TABLE OF CONTENTS

Factor

i is a factor of N → i will divide N completely.

N % i == 0

↓ Remainder

N = 100

i = 5 100/5 = 20

N/i

< *Question* > : Given N. Find the count of positive factors of N. (N > 0)

QUIZ

1. **N = 24**

Factors : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 24

ans : 8

QUIZ

2. **N = 10**

Factors : 1, 2, 5, 10 ans : 4

Observations :

Min Factor of a No : 1

Max " " " " : N

< / > Pseudo Code

```
int CountFactors(int N){
    int count = 0;
    for(i=1; i<=N; i++){        → N iterations
        if(N%i == 0){
            count++;
        }
    }
    return count;
}
```

N = 10

i : 1 – 10

1 2 3 4 5 6 7 8 9 10 11

Servers → 1 GHz

Assumption : 10^8 iterations are executed in 1s.

N	Iterations (N)	Execution Time
10^8	10^8	1s
10^9	10^9	10s
10^{18}	10^{18}	10^{10}s = 317 yrs

5 apples – 50Rs

1 apple – $\frac{50}{5}$ = 10

3 apples = 10*3 = 30

10^8 – 1s

1 – $\frac{1}{10^8}$

10^9 – $\frac{1*10^9}{10^8}$ = 10

10 – 1s

1 – $1/10^8$

10^{18} – $1 * 10^8 = 10^{10}$

10^8

Optimisation

$i * j = N \rightarrow i$ & j are factors of N

$j = \frac{N}{i} \rightarrow i$ & $\frac{N}{i}$ are factors of N.

N = 24

i		N/i
1	≤	24
2	≤	12
3	≤	8
4	≤	6
6		4
8		3
12		2
24		1

$i \leq N/i$

Part 1

Part 2

N = 100

i		N/i
1	≤	100
2	≤	50
4	≤	25
5	≤	20
10	≤	10
20		5
25		4
50		2
100		1

Observations :

1) Factors are Repeating / Factors exists in pairs.

2) Iterate till $i \leq \frac{N}{i}$

$\Rightarrow i^2 \leq N$

$\Rightarrow i \leq \sqrt{N}$

< / > Pseudo Code

```
int CountFactors(int N){
    count = 0;
    for(i = 1; i <= √N; i++){
        if(N%i == 0){
            if(i == N/i){
                count = count + 1;
            }
            else{
                count = count + 2;
            }
        }
    }
    return count;
}
```

1 : √N

√N iterations

N = 100

i : [1 : √100] = [1 : 10]

i = 1 2 3 4 5 6 7 8 9 10

count = 0 2 4 6 8 10

N	Iterations (√N)	Execution Time
10^{18}	$\sqrt{10^{18}} = 10^9$	$\frac{1}{10^8} \times 10^9 = 10\,s$

Servers → 1 GHz

Assumption : 10^8 iterations are executed in 1s.

Prime Numbers

Number having 2 factors (1, N)

QUIZ How many prime numbers are there?

10, 11, 23, 2, 25, 27, 31

ans : 4

< Question > : Given a number N. Check if it is prime or not.

< / > Pseudo Code

```
boolean CheckPrime(int N) {
    if(CountFactors(N) == 2)
        return true;
    else
        return false;
}
```

Range

[a , b] → This type of range means Both a & b INCLUDED.

(a , b) → " " " " " " " " EXCLUDED.

[a , b) → Include a & Exclude b

(a , b] → Exclude a & Include b.

QUIZ

How many numbers are in range?

[3 , 10] → 3, 4, 5, 6, 7, 8, 9, 10

ans : 8

QUIZ How many numbers are in range?

[a , b] → b - a + 1

[3, 10] 10 - 3 + 1 = 8

[a , b) → b - a + 1 - 1 = b - a

(a , b] → b - a + 1 - 1 = b - a

(a , b) → b - a - 1

Sum of first N numbers

Gauss

QUIZ

$1 + 2 + 3 + 4 + 5 - - - - - - - - - 98 + 99 + 100$

$S = 1 + 2 + 3 + 4 + 5 - - - - - - - - - 98 + 99 + 100$ —①

$S = 100 + 99 + 98 + 97 + 96 - - - - - - - 3 + 2 + 1$ —②

① + ②

$2S = \underbrace{101 + 101 + 101 + 101 - - - - - 101 + 101 + 101}_{100 \text{ times}}$

$2S = 101 * 100$

$S = \frac{101 * 100}{2} = 5050$

Sum of 1st N natural no's

$S = 1 + 2 + 3 - - - - - - (N-1) + N$

$S = N + (N-1) + (N-2) - - - - - - - - 2 + 1$

$2S = \underbrace{(N+1) + (N+1) + (N+1) - - - - (N+1) + (N+1)}_{N \text{ times}}$

$2S = N * (N+1)$

$S = \frac{N * (N+1)}{2}$

What is a iteration?

The no of times loop runs.

Quiz- 7

How many times will the below loop run ?

```
for(i -> 1 to N)
{
    if(i == N) break;
}
```

for (i = 1 ; i ≤ N ; i++)

i : [a , b] = [1 , n]

No of times loop will run = b - a + 1

= n - 1 + 1 = n

Quiz- 8

How many iterations will be there in this loop ?

```
for(i -> 0 to 100){
    s = s + i + i^2;
}
```

i : [a , b] = [0 , 100]

No of times loop will run = b - a + 1

= 100 - 0 + 1

= 101

Quiz- 9

How many iterations will be there in this loop?

```
func(){
    for(i -> 1 to N){
        if(i % 2 == 0){
            print(i);
        }
    }
    for(j -> 1 to M){
        if(j % 2 == 0){
            print(j);
        }
    }
}
```

i : [1, N]
N iterations

j : [1, M]
M iterations.

Total iterations : N+M.

Geometric Progression

- Series where the ratio of two consecutive terms remains same.

$a \quad ar \quad ar^2 \quad ar^3$

5 , 10, 20 , 40 , 80 , 160 , 320 , 640

$\frac{10}{5} = 2$

$\frac{20}{10} = 2$

$\frac{40}{20} = 2$

r : Common ratio $= 2$

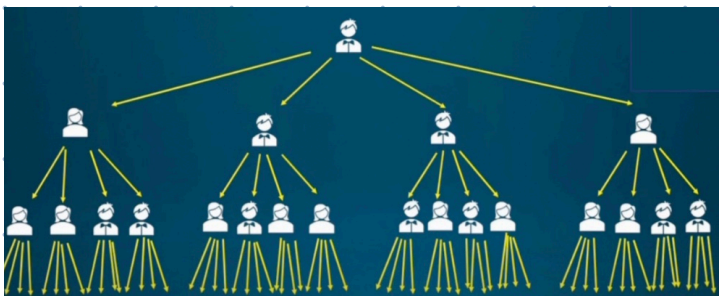
a : First term of GP $= 5$

$a, ar, ar^2, ar^3 - - - -$

Sum of first N terms of a GP : $\frac{a(r^N - 1)}{r - 1}$ $\quad r \neq 1$

N^{th} term of GP : ar^{N-1}

Real life example of GP



$1, 4, 16, \ldots$

$a = 1$

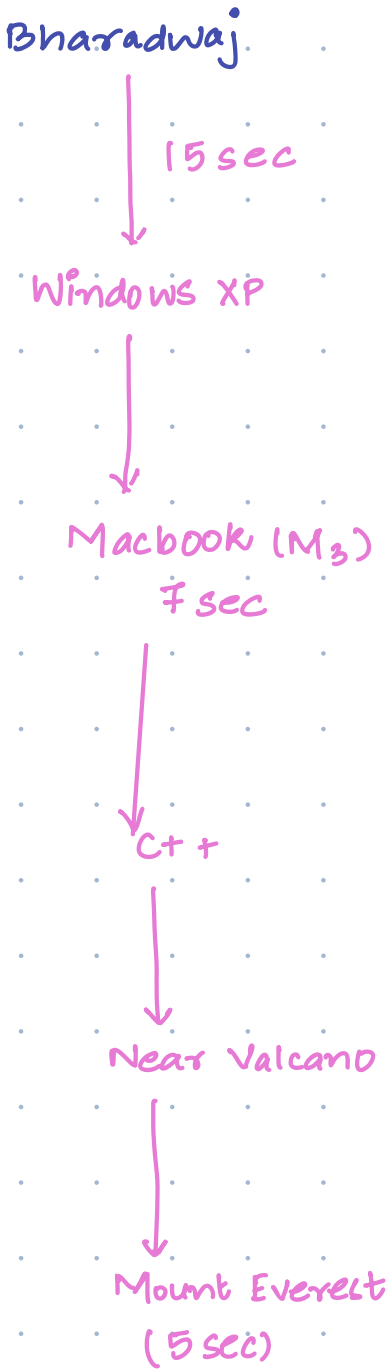
$r = 4/1 = 4$

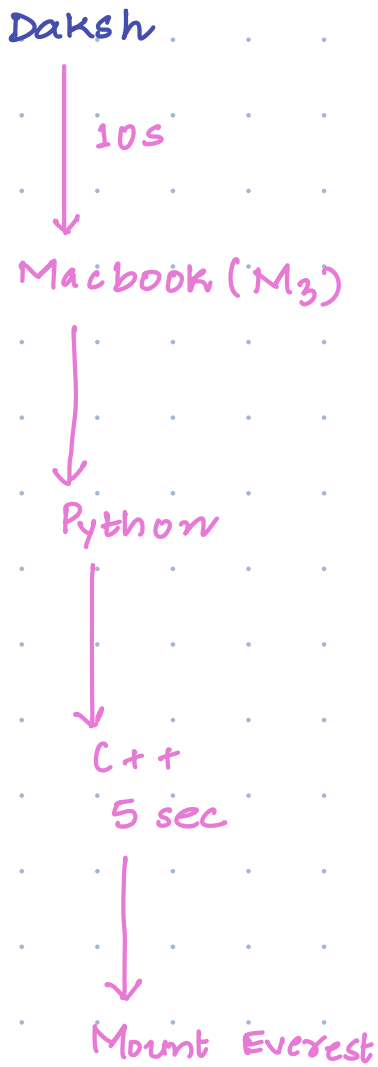
Number of people at 8th level: $ar^7 = 1 * 4^7 = 4^7$

Sum till 8th level: $\frac{a(r^N - 1)}{r - 1} = \frac{1(4^8 - 1)}{4 - 1} = \frac{4^8 - 1}{3}$

How to compare two algorithms?

N = 10^8. Given N elements, sort the elements in increasing order.





Execution time is not the good criteria to compare 2 algo since it depends on many external factors.

Criteria to compare 2 algo : No of Iterations