

Hello Everyone 😊

We will start at 7:10

About Me:

- Harshal Garg.
- SDE at Amazon, Ex-Oracle
- B.Tech (CSE) from  
NIT Jalandhar, 2018
- Remote Instructor atScaler  
since last 1 yr. Experience  
of close to 100 lectures.
- Born in Agra, U.P.  
Currently residing in Hyderabad

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 problem, 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 85% no matter what. It shall really help you to stay focused and we have seen in the past that people with  $\geq 85\%$ , do well in Interviews.

2. **Attendance**

- Try to maintain at-least 75% attendance either through live classes or by watching recording.
- 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 reason you are unable to, then please send me a message stating the reason.

Content for Intermediate Module  
— To be covered over  
next 5 weeks.

- Introduction to Problem Solving
- Time Complexity
- Introduction to Arrays
- Prefix Sum
- Carry Forward
- Subarrays
- 2D Matrices
- Sorting Basics
- Hashing Basics
- Strings Basics
- Bit Manipulation Basics
- Interview Problems
- Contest [covers Full Intermediate DSA]

**Note:**

- 1. In Intermediate, we shall be learning the concepts around different topics and how to work with certain data structures.
  - This module is dedicated to make you comfortable with Programming.
- 2. Contest will be organised after Intermediate Module.
  - It'll will be for 1.5 hours and will be conducted within class duration followed by Contest Discussion (Instructor shall be discussing contest problems).
  - It'll consist of 3 questions and we expect you to solve  $\geq 2$  problems. If for any reason you are unable to solve, then we shall also be having re-attempts as well. (We'll provide more info on re-attempts moving forward)
  - Contests are critical to retaining what you have learnt and measuring where you need improvement. Please take contests seriously.
- 3. Be consistent in solving problems. If stuck, please post the issue in your WA/Slack group and let's make it a habit of helping each other as it will eventually help you to be better.

**FAQs :**

- Notes will be uploaded after the class.
- Assignments will be unlocked after the class ends.
- There is no deadline for assignments.
- If asking a question, ask in public chat.
- If answering a question, answer in private chat.

Lecture Days:-

Monday

Wednesday

Friday

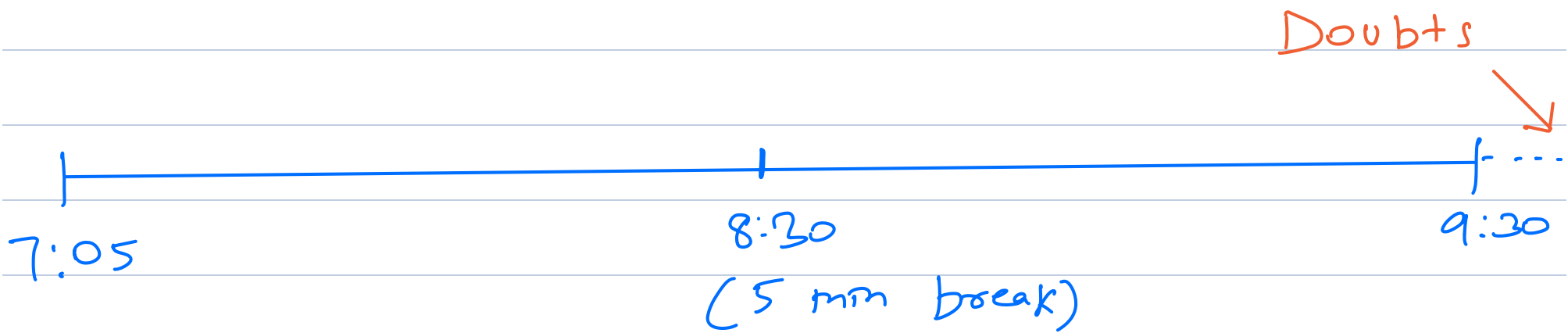
7:00 AM

1 ST.

We start at

7:05

sharp



## Doubts

Conceptual

Harshad Jay.

Implementation

Peen's.

TA.

## Today's Agenda:-

- ① Count the Factors
- ② Optimisation for counting the factors.
- ③ Check if a number is prime.
- ④ Sum of  $N$  natural numbers.
- ⑤ Definition of A.P & G.P.
- ⑥ Finding no. of Iterations.
- ⑦ How To Compare algorithms.

Q-1. What is a factor?

$i$  is a factor of  $N$ .

$i$  completely divides  $N$ .

Q:- How to check it programmatically?

:-

$$N \% i == 0$$

↑  
factor.

Q:- Given  $N$ , count the factors  
 $N > 0$

Quiz 1  $N = 24$  - 8.

1, 2, 3, 4, 6, 8, 12, 24.

Quiz 2  $N = 10$

1, 2, 5, 10 - 4.

# Naive / Brute Force Approach

12..

1 2 3 4 5 6 7 8 9 10 11 12

Pseudocode.

```
count = 0;
for (i → 1 to N) {
    if (N % i == 0)
        count++;
}
return count;
```

→ Code runs on servers

→ It takes some time to execute / run your code.

→ We would like to estimate how much time a code might take to run.

Servers have the capacity of running  $\sim 10^8$  iterations per second

No. of iterations our program takes for a given input  $N$ ?

N	iterations	Execution Time
$10^8$	$10^8$	$10^8 \rightarrow 1 \text{ sec.}$ $1 \text{ sec.} \cdot 10^8$
$10^9$	$10^9$	$10 \text{ sec.} \cdot 10^8$ $1 \rightarrow \frac{1}{10^8} \text{ sec.}$
$10^{18}$	$10^{18}$	$10^{10} \text{ sec.}$ $\Downarrow$ $317 \text{ years.}$ $10^9 \rightarrow \frac{1 \times 10^9}{10^8}$ $\frac{1}{10^8} \times 10^{18} = 10^{10}$

Optimisation:-

N.  
 $a$  is a factor of  $N$ .  
 $N/a$  is also a factor of N.

24.

a	N/a
1	24
2	12
3	8
4	6
6	4
8	3
12	2
24	1

Obs 1:- Factors are repeating.

$$a < N/a.$$

# Factors exist in pair.

$$i < N/i; \quad i^2 < N \\ i < \sqrt{N}$$

# Code.

```
for (i = 1; i < sqrt(N); i++) {
    if (N % i == 0) {
        count += 2;
    }
}
return count;
```

$$N = 24.$$

$$\text{count} = 9 \cancel{7} 4 \cancel{8}$$

$$i = 1 \cancel{2} \cancel{3} 4 5$$

$$N = \underline{25}$$

$$1, 5, 25.$$

1	<del>2</del>	24.
2	<del>x</del>	12
3	<del>8</del>	8
4	<del>6</del>	6.

count = 0;

```
for (i = 1; i < sqrt(N); i++) {
```

```
    if (N % i == 0) {
```

```
        if (i == N/i) { count++; }
        else.
```

```
        count += 2;
```

```
    }
```

```
}
return count;
```

$$N = \underline{25};$$

$$i = 1 \cancel{2} \cancel{3} 4 \cancel{5} 6$$



$$\text{cont} = \phi \neq 3$$

$$1 \times 25$$

$$5$$

N	iterations	Execution Time	
$10^8$	$10^4$	$10^{-4}$ <u>sec.</u>	$1 \rightarrow \frac{1}{10^8}$
$10^{18}$	$10^9$	10 <u>sec.</u>	$10^4 \rightarrow \frac{10^4}{10^8}$

317 years

$\longrightarrow$  10 sec.

Q:- Given  $N$ , you need to check if it is a prime number or not. ?  
Exactly 2 factors.

:- Quiz 3.

10	11	23	2	25	27	31
4	2	2	2	3	4	2
	✓	✓	✓			✓

Ans = 4.

```
boolean checkPrime ( int N ) {
```

```
    if ( countFactors(N) == 2 )  
        return True;  
    else  
        return False;  
}
```

Break till 8:40.

Sum of first  $N$  numbers:-

Ans:

$$S = 1 + 2 + 3 + 4 + \dots + 100 = ?$$

$$S = 100 + 99 + 98 + \dots + 1$$

$$2S = \underbrace{101 + 101 + 101 + \dots + 101}_{100 \text{ terms}}$$

$$2S = 101 \times 100$$

$$S = \frac{101 \times 100}{2} = \underline{\underline{5050}}$$

$$S = 1 + 2 + 3 + \dots + N-2 + N-1 + N$$

$$S = N + N-1 + N-2 + \dots + 3 + 2 + 1$$

$$2S = \underbrace{(N+1) + (N+1) + (N+1) + \dots}_{N \text{ terms}}$$

$$2S = N \times (N+1)$$

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

# Sum of first  $N$  natural numbers.

Some more basic Maths:-

①  $[a, b] \rightarrow$   $a$  to  $b$  inclusive of  $a$  &  $b$ .

②  $(a, b) \rightarrow$   $a$  to  $b$  exclusive of  $a$  &  $b$ .

$(, )$   $\rightarrow$  Excluding  
 $[, ]$   $\rightarrow$  Including.

Ques 5

$[3, 10] - \underline{8}$ .

3, 4, 5, 6, 7, 8, 9, 10

$$\underline{10 - 3 + 1} = \underline{8}$$

Ques 6

$[a, b]$

$\begin{array}{ccccccc} & & & a+1 & a+2 & & \\ & & & | & | & & \\ - & - & - & a & - & - & b \\ & & & | & | & & \end{array}$

$$b - a + 1$$

$$\begin{aligned} a &= 2 \\ b &= 5 \end{aligned}$$

$$5 - 2 + 1 = 4.$$

$$(a, b) \rightarrow b - a - 1$$

Iterations :-

No. of times a loop runs.

Quiz 7

```
for (i = 1; i <= N; i++) {  
    if (i == N) break;  
}
```

$[1, N]$ .

$$N - 1 + 1 = \underline{N}.$$

## Quiz 8

```
for (int i = 0 ; i <= 100 ; i++) {  
    s = s + i + i^2 ;  
}
```

$$[0, 100] \rightarrow 100 - 0 + 1 \\ = \underline{\underline{101}}.$$

## Quiz 9

```
func () {
```

```
    for (int i = 1 ; i <= N ; i++) {
```

```
        if (i % 2 == 0) {  
            print(i);
```

```
        }
```

```
    }
```

```
    for (int j = 1 ; j <= m ; j++) {
```

```
        if (j % 2 == 0) {
```

```
            print(j);
```

```
        }
```

```
    }
```

```
}
```

$N + m$

# Geometric Progression (G.P.)

5      10      20      40      80

$$\frac{10}{5} = \frac{20}{10} = \frac{40}{20} = \frac{80}{40} = \underline{\underline{2}}$$

$$a = 5; \quad r = 2.$$

Generic Notation :

$$a, a \times r, a \times r^2, a \times r^3$$

$a \rightarrow$  first term

$r \rightarrow$  common ratio.

Sum of first  $N$  terms  
of a G.P.

$$= \frac{a(r^N - 1)}{r - 1}$$

$$r \neq 1$$

# Comparing Two Algorithms ?

Snit  
X  
15 sec  
(Windows XP)

⇓  
Macbook.  
7 sec.,  
C++

Awynth.  
Y.

10 Sec.  
(Macbook).

↓  
Python.  
⇓  
C++  
5 sec.,  
      

# By looking at the execution time, I cannot estimate which algo is better.

Factors involved in E.T.  
are  
O.S.  
Machine.  
Language.  
Temp.

# Heuristics criteria to compare two algorithms.



## Next Class :

→ Steps to calculate Big O, reasoning behind and also drawbacks.

→ Explore a Maths concept  
— Logarithm.

→ Space Complexity

→ TLE and Importance of Constraints.