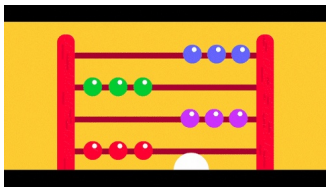
# Introduction to Problem Solving



Welcome Everyone

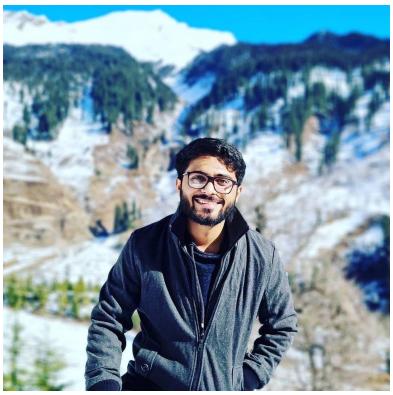
9:05

AGENDA:

- Counting factors
- Power of observations
- Square root
- About intermediate batch

# Instructor

## Tarun Luthra



- Software Engineer & Instructor @ Scaler
- Fullstack Developer @ LevelAI (USA based startup)
- Fullstack Developer @ Coding Minutes
- Instructor @ Coding Blocks

Dota 2, CS GO

Animes / Mangas

Fiction Books

Marvel / DC

# Factors & Prime Numbers

Q Given a no. , find the count of its factors

10 → 1, 2, 5, 10 → 4

25 → 1, 5, 25 → 3

N → [1, N] ← Range for factors of N

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

Iterations - N

Assumption

$10^8$ iterations → 1 sec

1 iteration → $\frac{1}{10^8}$ sec

$N = 10^9$

$10^9$ iterations → 10 sec

$N = 10^{18}$

$10^{18}$ iterations → $\frac{1}{10^8} \times 10^{18}$ sec

$= 10^{10}$ sec

≈ 317 years

You → Kids → Grandkids → $3^{rd}$ → $4^{th}$ → $5^{th}$

$6^{th}$ / $7^{th}$

---

## Optimize Code

Say,

$$a * b = N$$

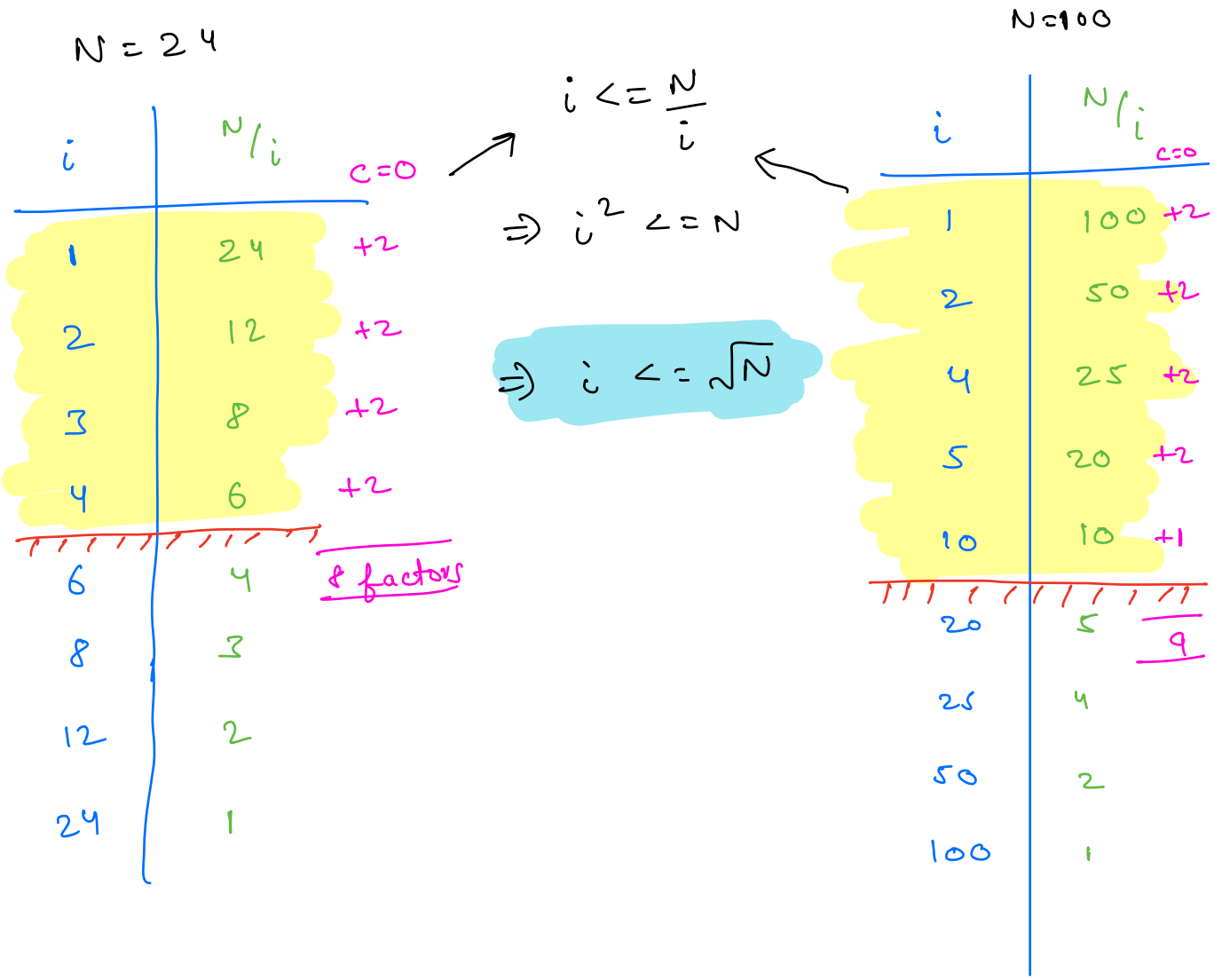
⇒ Both a and b are factor of N

$$b = \frac{N}{a}$$

⇒ Both a and $\frac{N}{a}$ are factor of N

If a is a factor of N,

$\frac{N}{a}$ will also be a factor of N.




$N = 10,000$

$\frac{N}{2} = 5,000$

$\sqrt{N} = 100$

Range - $[1, \sqrt{N}]$

N * * (0.5)

math.sqrt(N)

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

i <= $\sqrt{N}$

Iterations - $\sqrt{N}$

Assumption

$10^8$ iterations → 1 sec

⇒ 1 iteration → $\frac{1}{10^8}$ sec

$N = 10^8$

Iterations $= \sqrt{10^8} = 10^4$ iterations

$10^4$ iterations $= \frac{1}{10^8} \times 10^4$

$= \frac{1}{10^4}$ sec

$= 0.0001$ sec

$N = 10^{18}$

Iterations $= \sqrt{10^{18}} = 10^9$

Time for 1 iteration

$10^9$ iterations $= \frac{1}{10^8} \times 10^9$

$= 10$ sec

317 yrs → 10 sec

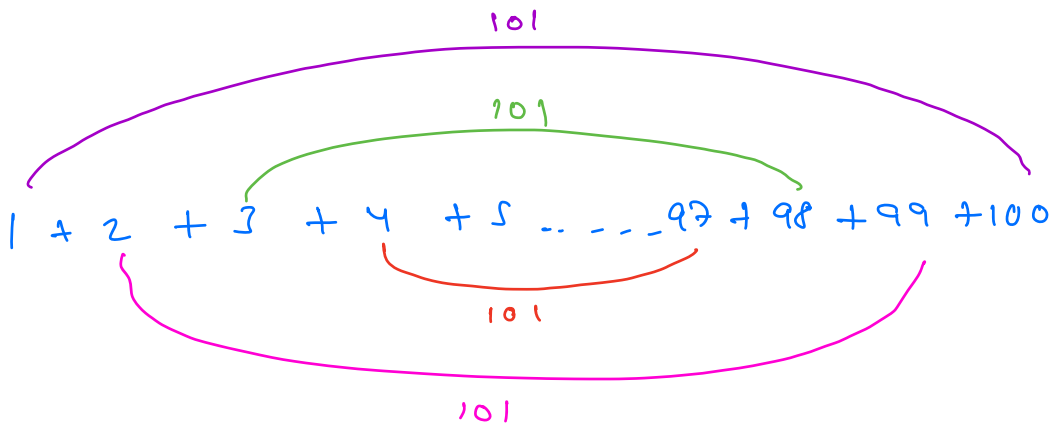
## Observations

2 months

---

Carl Freidrich Gauss

4th class




50 pairs

Sum of each pair = 101

Total Sum = 50 × 101 = 5050

$\frac{N(N+1)}{2}$

## How many times can you divide N by 2 till it reaches 1 ?

$\Rightarrow \log_2 N$

$2^6$ : 64 $\xrightarrow[/2]{}$ 32 $\xrightarrow[/2]{}$ 16 $\xrightarrow[/2]{}$ 8 $\xrightarrow[/2]{}$ 4 $\xrightarrow[/2]{}$ 2 $\xrightarrow[/2]{}$ 1

$\Rightarrow$ 6 times

5 $\xrightarrow[/2]{}$ 2 $\xrightarrow[/2]{}$ 1 $\Rightarrow$ 2 times

$\frac{5}{2} = 2.5$

27 $\rightarrow$ 13 $\rightarrow$ 6 $\rightarrow$ 3 $\rightarrow$ 1 $\rightarrow$ 4 times

---

$2^0$ : 1 — 0 times

$2^1$ : 2 — 1 times

$2^2$ : 4 — 2 times

$2^3$ : 8 — 3 times

10 — 3 times

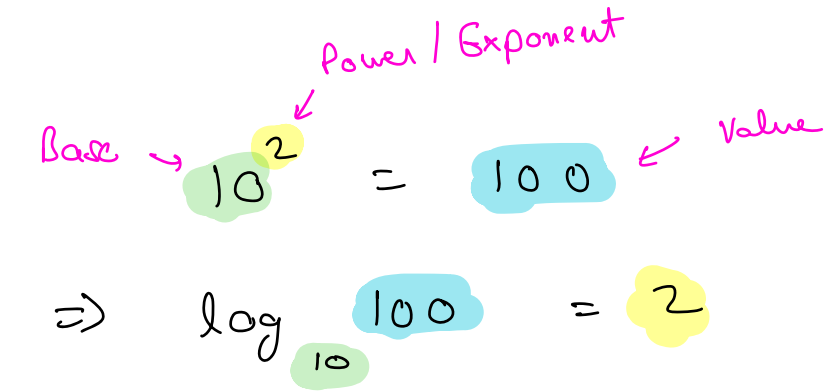
$2^4$ : 16 — 4 times

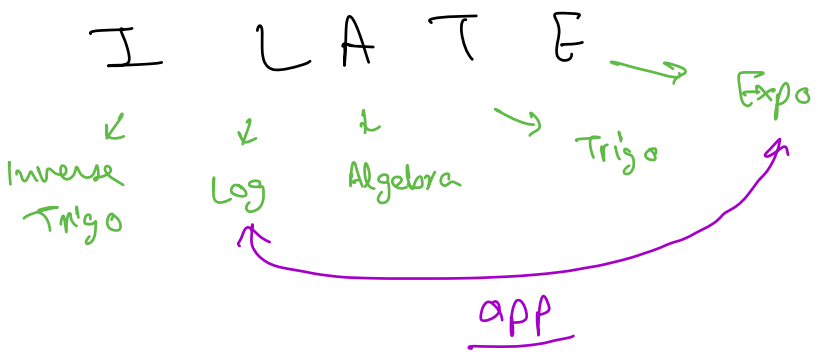
$2^s$ $\longrightarrow$ s times

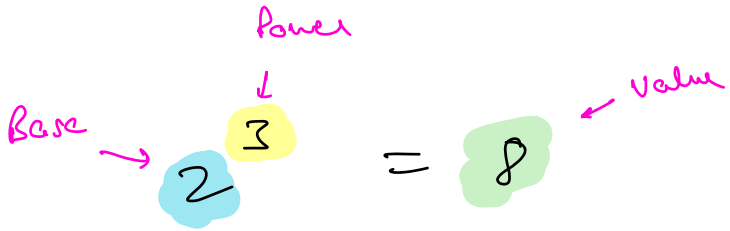
int $(\log_2 N)$

$2^6 : 64 \longrightarrow$ 6 times

---






$$\log_2 8 = 3$$

$$2^x = 8$$

---

$$\log_4 64 = 3$$

$$4^x = 64$$

$$\Rightarrow 4^x = 4^3$$

$$\Rightarrow x = 3$$

---

$$\log_{11} 121 = 2$$

$$11^{x} = 121$$

$$\Rightarrow 11^{x} = 11^{2}$$

$$\Rightarrow x = 2$$

Please revise

- log
- AP
- GP

Q Given a perfect square, find its square root.

35, 24, 49, 50

↑

$7^2$

---

N = 100 → 10

N = 25 → 5

N = 36 → 6

N = 20 → Invalid Input

---

Range - [1, N]

```
sqrt (int N) {
    for (i=1; i<=N; i++) {
        if (i*i == N)
            return i
    }
}
```

N = 100

i → 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

N = 25

i → 1, 2, 3, 4, 5

$\sqrt{N}$ iterations

Optimize ??

Range - [1, N]

Binary search

N = 100

[1, 100]

1 2 3 4 ..... 49 50 51 52 ..... 98 99 100

Size = 100

$50^2 == 100$ ✗

$50^2 > 100$

[1, 49]

1 2 3 4 5 ..... 24 25 26 ..... 48 49

Size = 50

$25^2 == 100$ ✗

$25^2 > 100$

[1, 24]

1 2 3 4 .... 11 12 13 ---- 23 24

Size = 24

$12^2 == 100$ ✗

$12^2 > 100$

[1, 11]

1 2 3 4 5 6 7 8 9 10 11

Size = 12

$6^2 == 100$ ✗

$6^2 < 100$

[7, 11]

7 8 9 10 11

Size = 6

$9^2 == 100$ ✗

$9^2 < 100$

[10, 11]

Size = 2

10 11

$10^2 = = 100$ ✔

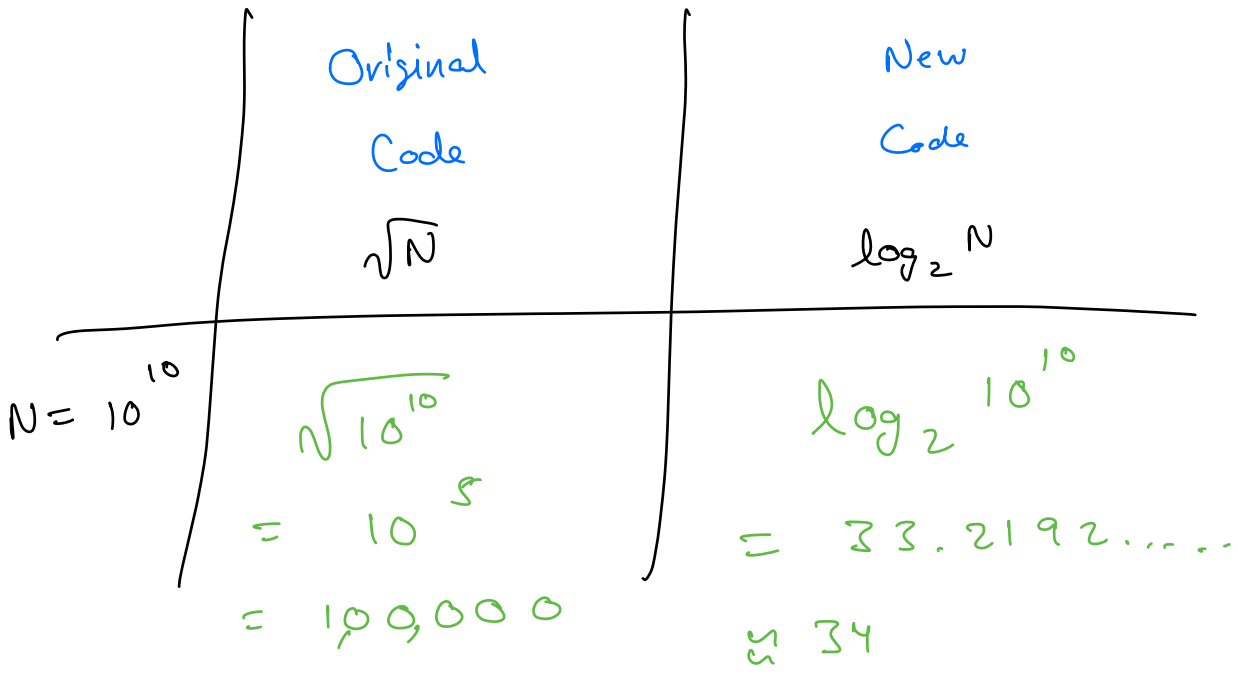
Square root of 100 is 10

Size

100 → 50 → 25 → 12 → ..... 1

N → $\frac{N}{2}$ → $\frac{N}{4}$ → $\frac{N}{8}$ → $\frac{N}{16}$ ..... 1

Iterations = $\log_2 N$




Optimised ??

Trailer

## Intermediate Content → 2 Months

Time & Space Complexity – 2

Arrays → 6/7

Bit Manipulation → 3

Maths → Modular – 1
→ Interesting – 1
} 2

String – 1

Sorting – 1

Hashing – 2

Recursion – 3

Data Structures – Linked list – 1
Stack / Queues – 1
Trees – 2

Subsets & Subsequences – 1

## Expectations

1) Attend session

2) Revise Notes

3) Asgn / HW

↳ Assignments — Directly covered in the class

↳ HW — Extra Problems that are slightly trickier version

4. Doubts

↓

Classroom Doubts

→ Live

→ Doubt Session

→

Problem based doubts

→ Debug on your own — 30 mins

→ TA help

→ Ask doubts in Doubt session

→ Slack

→ Problem Solving Sessions

After every 2/3 weeks

Pick least solved problems

---

→ Please join on time

9:05

---

If you miss

→ Go through recording

→ Atleast go through notes

---

# Doubts

Good Night

Thank You

Friday