Data structures and Algorithms

~~Beginner~~ | Intermediate | Advanced

2 months | 4 - 4 1/2 months

Timings → Monday Wednesday Friday

9 pm – (11:00 – 11:30 pm)

2 - 2 1/2 hr session + Doubts → recorded

Break 8-10 min

Attendance not considered

- Lecture Notes will be uploaded and can be accessed using dashboard.

- PSP → Problem solving percentage

Assignment / Homework

Lecture –

Whatever we do in lecture

Slack

whatsapp

email → mohit.sharma@scaler.com

Any other doubts, please put it in Questions Tab, I'll take it at the end of session.

$N = 15$ factors → 1, 3, 5, 15

↓ which completely divides

N, x $\quad N \% x == 0$

↓ modulous

10 = {1, 2, 5, 10}

24 = {1, 2, 3, 4, 6, 8, 12, 24}

1 ⟶ N

```
int cnt = 0;

for( i = 1; i<=N; i++)
{
    if( N % i == 0)
        cnt++;
}
```

iteration → N

10^9 cc/sec

1 sec ≈ 10^8 iterations

N	Iteration = N	Time
10^8	10^8	1 sec
10^9	10^9	10 sec
10^{18}	10^{18} iterati	10^{10} sec ≡ 317 years

1 sec = 10^8 iter

$\frac{1}{10^8}$ sec = 1 iter

$\frac{10^9}{10^8}$ = 10^9 iter

10 sec

$1 \longrightarrow N$

$1 \longrightarrow N/2$

$N = 24$

i	N/i	
1	24	cnt = 2 (+2)
2	12	cnt = 4 (+2)
3	8	cnt = 6 (+2)
4	6	cnt = 8
?		
6	4	
8	3	
12	2	
24	1	

① Factors are occurring in pairs

$i * j = N$

$j = N/i$

② After a point, factors are repeating

$i <= N/i$

$i * i <= N$

$i^2 <= N$ or $i <= \sqrt{N}$

32

1	32
2	16
4	8
8	4
16	2
32	1

$N = 100$

1	100	cnt = 2
2	50	cnt = 4
4	25	cnt = 6
5	20	cnt = 8
10	10	cnt = 10
20	5	
25	4	
50	2	
100	1	

$i * i <= N$ ⇕ $i <= \sqrt{N}$

```
int cnt = 0;
for( i=1; i*i<=N; i++)
{
    if( N%i == 0)        // i, N/i
    {
        if( i != N/i) cnt += 2;
        else cnt += 1;
    }
}
```

iterations → $\sqrt{N}$

N	iteration	Time
10^8	10^4	0.0001 sec
10^{10}	10^5	0.001 sec
10^{18}	10^9	10 sec

1 it = $\frac{1}{10^8}$ sec

$10^4 \equiv \frac{10^4}{10^8}$ sec (10^{-4})

10 sec ↑ 317 years ☺

- Prime Number
  - exactly 2 factors
    - count no of factors == 2
  - 1 and itself

Advanced

7

1 * 7

Gauss ? 4th standard 1 → N

$S = 1 + 2 + 3 + 4 + \ldots\ldots + 99 + 100$

$S = 100 + 99 + 98 + 97 + \ldots\ldots + 3 + 2 + 1$

$2S = 101 + 101 + 101 \ldots\ldots 101 + 101$

$2S = 100 * 101$

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

Sum of $1 \rightarrow N$

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

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

+

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

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

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

- N - perfect square. Calculate the square root of the number.

i → 1 → N $i*i == N$ → square root

N = 100

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

N ~~times~~

$\sqrt{N}$

N is not a perfect square?

↓

int part of answer.

N = 50

```
int i=1;
while( i*i<=N)
{   i++;
}
return i-1;
```

i = 1	1*1 <= 50
= 2	2*2 <= 50
3	3*3 <= 50
4	4*4 <= 50
5	5*5 <= 50
6	6*6 <= 50
7	7*7 <= 50
8	8*8 > 50

square root

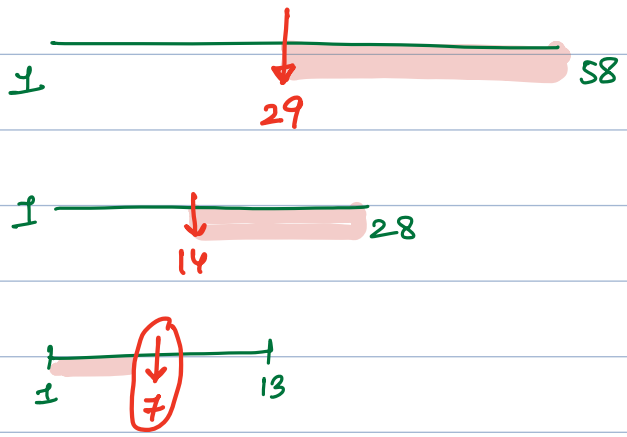
↓

Binary search → Advanced

Out of Topic: Advanced

58

$\log_2 N < \sqrt{N}$



29 * 29 > 58

14 * 14 > 58

7 * 7 <= 58

log Basics

$\log_b a = x$ $\quad b^? = a$

How many time base 'b' should be raised to get a value of a

$\log_2 64 = 6$

$\log_3 27 = 3$

$\log_5 625 = 4$

$\log_3 10 = 2.*$

$\log_2 2^{128} = 128$

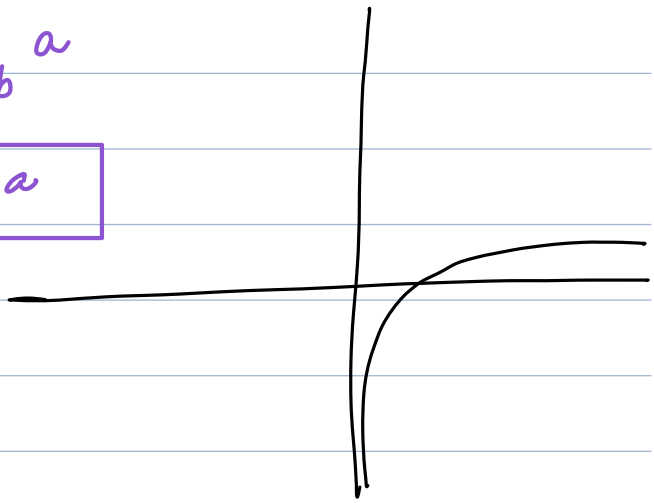
$\log_b b^N = N$

$k = ?$

$$b^k = a$$

$$\log_b b^k = \log_b a$$

$$\boxed{k = \log_b a}$$



$\boxed{\log_2 N}$

- How many times you should divide N by 2 to reach 1 ?

100 → /2 → 50 → /2 → 25 → /2 → 12 → /2 → 6 → /2 → 3 → /2 → 1

$N \rightarrow /2^1 \rightarrow N/2^1 \rightarrow /2^2 \rightarrow N/2^2 \rightarrow /2^3 \rightarrow N/2^3 \rightarrow \cdots \rightarrow /2^k \rightarrow N/2^k$ (k times)

$$N/2^k = 1$$

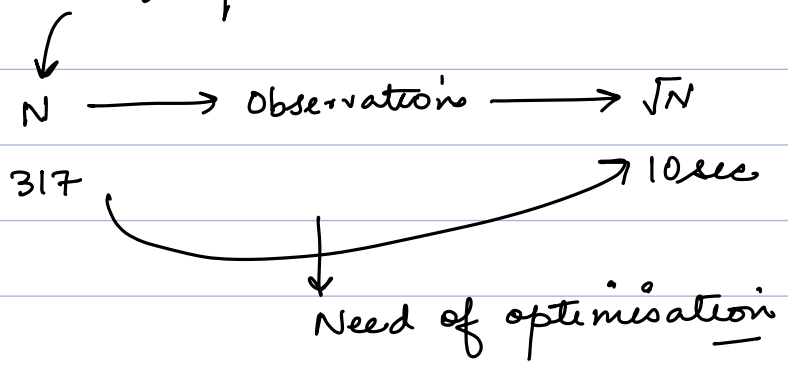
$$N = 2^k$$

$$\log_2 N = \log_2 2^k$$

$$\boxed{\log_2 N = k}$$

Recap

1) No of factors



2) Sum 1 → N $\frac{N*(N+1)}{2}$

3) Square root

4) $\log_b a$