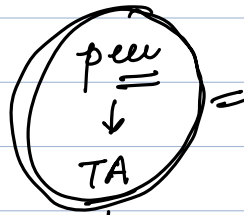


Advanced  
DSA  
↓  
4 1/2 months

Duration:- 2:15 - 2:30  
↓  
2:45 min



slack  
↗ whatsapp

slack (ping me)

①

N, count of factors of N.

N = 50

{1, 2, 5, 10, 25, 50}

cnt = 6

```

1 → 50
↑
cnt = 0; ①
for (int i = 1; i <= N; i++) ②
{
    if (N % i == 0) ③
    {
        cnt++; ④
    }
}
    
```

modulo → remainder

1 GHz =  $10^9$  cc/sec

1 cc = 1 instruction  
 $10^9$  instr/sec

1 iteration = 5 instr

$N = 10^9$

iterations  
 $10^9$

instruction

$5 \times 10^9$

Sec

$N = 10^{18}$

$10^{18}$

$5 \times 10^{18}$  instr

$5 \times 10^9$  sec

158 yrs

1 sec =  $10^9$  ins  
1 hr = 1 hr  
 $\frac{1}{10^9} \times 5 \times 10^{18} \times 10^9$

$$N = 24$$

$$N/i$$

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

factors are repeating

1) factors appear in pairs

$$i \leq N/i$$

$$i^2 \leq N$$

$$i=1 \rightarrow i \leq \sqrt{N}$$

for(  $i=1; i \leq \sqrt{N}; i++$  )

$$1 \rightarrow \sqrt{N}$$

if(  $N \% i == 0$  )

if(  $i == N/i$  )  $cnt++$ ;

else

$cnt += 2;$  //  $i, N/i$ ;

$$36$$

1	36	$t=2$
2	18	$t=2$
3	12	$t=2$
4	9	$t=2$
6	6	$t=2$
9	4	
12	3	
18	2	
36	1	

$$10^3 \times 10^{4.5} \times 10^{4.5}$$

$$N = 10^9$$

$$\text{iterations}(\sqrt{N})$$

$$10^{9/2}$$

$$\text{instrut} \approx 5 \times 10^5 \rightarrow$$

$$10^3 \times 10^3 \times 10^6$$

$$\frac{10^{9/2}}{10^9} = 10^{-9/2}$$

$$1 \text{ sec} = 10^9 \text{ instr}$$

$$1 \text{ year} = 10^7 \text{ sec}$$

$$10^7 \times 5 \times 10^5 = 5 \times 10^{12}$$

$$10^9 \times 5 \times 10^5 = 5 \times 10^{14}$$

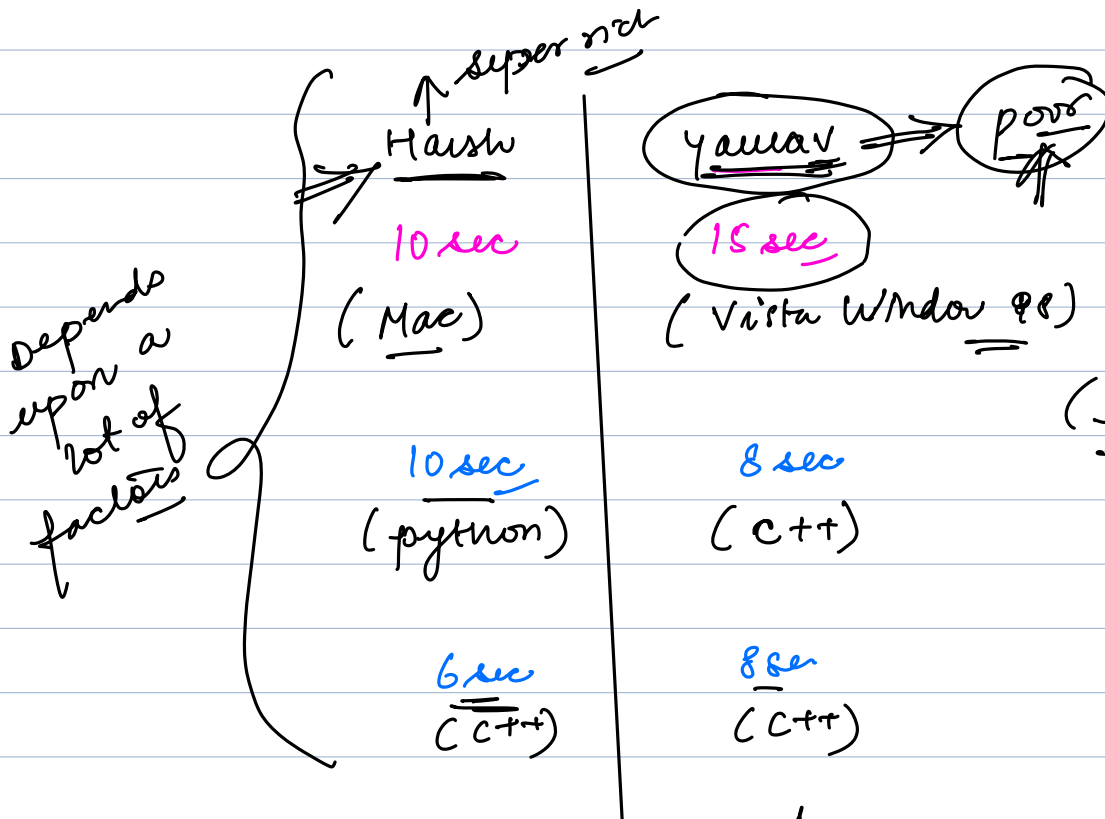
$$N = 10^{18} \xrightarrow{\text{iterations } (\sqrt{N})} 10^9$$

$$\text{not } 5 \times 10^9 \Rightarrow 5 \text{ sec}$$

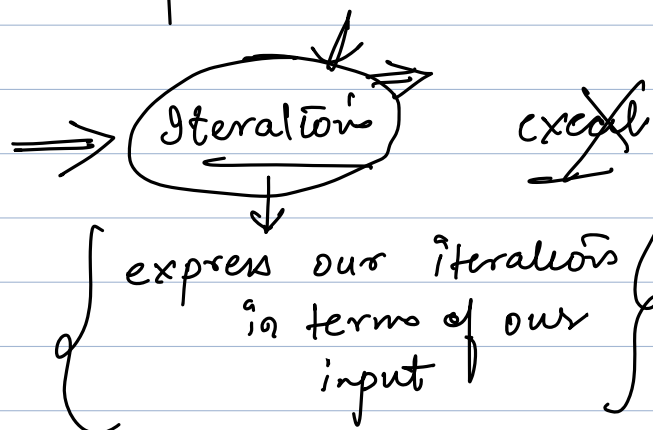
158 years

Why iterations?

Execution Time?



(python is slower C++)



① void func ( int N )

$$S = 0;$$

for (int i=0; i<n; i++)

3

$$S = S + i;$$

}

2

$N \Rightarrow \underline{O(N)}$

② void func ( int n )

$$S = 0;$$

```
for (int i=0; i<n; i+=2)
```

3

$$s + = i;$$

j

3

$$i^a = 1, \quad i^k = N, \quad i^a$$

Hand-drawn diagram illustrating a recursive process for  $N=7$ . The sequence of numbers 1, 3, 5, 7 is shown at the bottom, underlined with a pink line. Above this, a tree structure of circles is drawn. The root circle contains 7. An arrow points from 7 to a circle containing 2. From 2, an arrow points to a circle containing 3. Another arrow points from 2 to a circle containing 7, which has a checkmark next to it. A stick figure is drawn next to the circle containing 4, which is at the bottom right of the tree structure.

$0 \rightarrow \dots \rightarrow n$

$N/2$

$$= O(N)$$

void func (int n)

$$c^2 \leq N \rightarrow c \leq \sqrt{N}$$
$$S = 0;$$

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

3

$$s + = \lambda_j$$

3

3

1 →

$$\sqrt{N}$$
$$O(\sqrt{N})$$

```
void func (int n)
```

```
    i = N;
```

```
    while (i > 1)
```

```
    {
        i = i/2;
    }
```

```
}
```



$$N/2^k = 1$$

$$N = 2^k$$

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

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

$$k = \log_2 N$$

$$O(\log_2 N)$$

```
void func (int n)
```

```
    s = 0
```

```
    for (i = 0; i <= N; i += 2)
```

```
    {
```

```
        s = s + i;
```

```
    }
```

```
}
```

$$0 \times 2 = 0$$

$$2 \times 2 = 0$$

$$\log_2 N$$

infinite

```
void func (int n)
```

```
    s=0
```

```
    for (i=1; i <= N; i*=2)
```

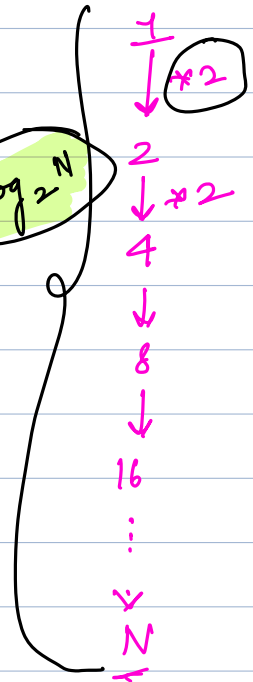
```
    {
```

```
        s = s + i;
```

```
    }
```

$O(\log_2 N)$

```
}
```



```
void func (int n)
```

```
    s=0
```

```
    for (j=1; j <= 10; j++)
```

```
    {
```

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

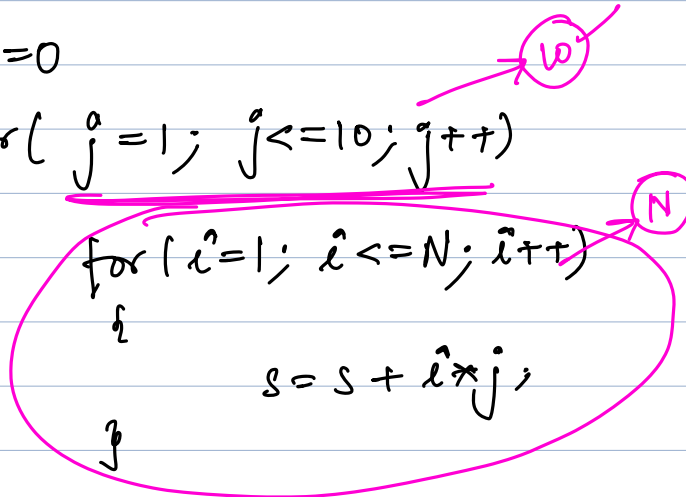
```
        {
```

```
            s = s + i*j;
```

```
        }
```

```
    }
```

```
}
```



$10 \times N$   
 $O(N)$

```
void func (int n)
```

s=0

```
for ( j=1; j<=N; j++)
```

```
{
```

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

```
{
```

s = s + i \* j;

```
}
```

```
}
```

```
}
```



$N^2$

$O(N^2)$

```
void func (int n)
```

s=0

```
for ( i=0; i<n; i++)
```

```
{
```

```
for ( j=0; j<=i; j++)
```

```
{
```

s = s + i \* j;

```
}
```

```
}
```

```
}
```

i	j	
0	0-0	①
1	0-1	②
2	0-2	③
⋮		
N-1	0-N-1	④

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

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

$\frac{N^2}{2} + \frac{N}{2}$

$= O(N^2)$

```
void func (int n)
```

s=0

```
for ( i=1, i<=n; i++)
```

```
{
```

$N$

$\log_2 N$

for ( $j=1$ ;  $j \leq n$ ;  $j=j*2$ )

d

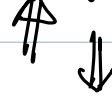
$s = s + *j$

j

}

}

$$N \log_2 N$$



$$O(N \log_2 N)$$

$$f(N) = N^2 + 10N$$

$$N=10$$

$$100 + 100$$

$$N=100$$

$$10^4 + 10^3$$



$$\frac{10^3}{10^3 + 10^4}$$

$$= 0.1 = 10\%$$

$$N=10^4$$

$$10^8 + 10^5$$

$$\frac{10^5}{10^8 + 10^5} = 10^{-3} = 0.001$$

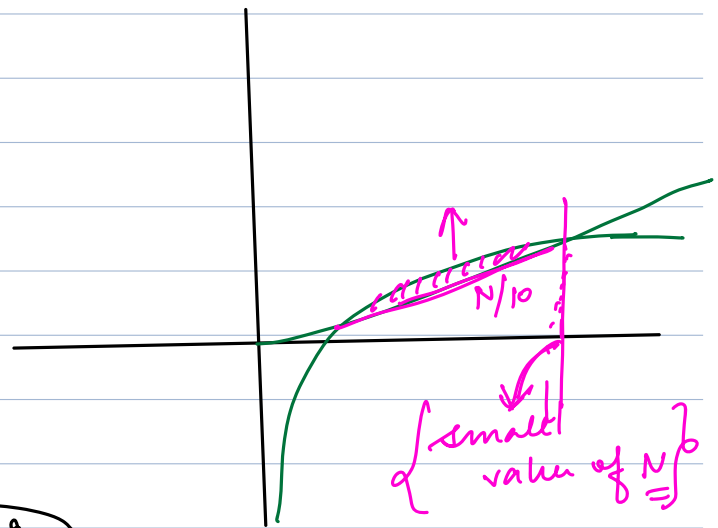
$$= 0.1\%$$

$N \rightarrow \text{large}$

$$O(1) \rightarrow \log_2 N \rightarrow \sqrt{N} \rightarrow N \rightarrow N \log_2 N < N^2 < (N^2 \log_2 N) < O(N^3) < O(2^N)$$







Bigo

- 1) calculate no of iterations
- 2) ignore lower order terms
- 3) ignore constant coeff =

$$\frac{O(C)}{=}$$

upper bound

•  $5N^2 + 10N + 6 \Rightarrow O(N^2)$  rate of growth

•  $4N + \log_2 N + 5 \Rightarrow O(N)$

space complexity → ↓

(Time & space)

```
{  
    (N)  
    int x, y;  
    cout << x << y << endl;  
}
```

4B

8 Bytes → O(1)

O(N)

~~O(N)~~  
↓  
O(1)

```
{  
    (N)  
    int x, y;  
    int arr[N];  
}
```

int x, y;

→ 8 Bytes

int arr[N];

→ 4N Bytes

4N + 8

↓  
O(N)

```
{  
    (N)  
    int x, y;  
    for (int i = 0; i < N; i++)  
    {  
        int arr[i];  
    }  
}
```

int x, y;

for (int i = 0; i < N; i++)

int arr[i];

⇒ O(N)

```
{  
    ()  
    int x, y;  
    int arr[N][N];  
}
```

int x, y;

8B

int arr[N][N];

4N<sup>2</sup> Bytes

4N<sup>2</sup> + 8

↓  
O(N<sup>2</sup>)

Array → { prefix sum  
 carry forward  
 sliding window  
 contribution tech }

12 min  
 5-10 min

Arrays - 1, 2, 3

Bit - 2

Maths - { Modulo  
 prime  
 GCD  
 combn }

Recursion → T.C/S.C

Sorting

Search BS {  
 4<sup>th</sup> x

Two pointers

Hash - 2

Step path - 2

Stack - 3 & 4

LL - 3 & 4

Trees - 5  
 - 2 Trees  
 - 2 Heaps

Greedy - 1

Backtr - 2

Dp - 6 → 8

Graph 4-5

DSA  
 ↓  
 Advanced

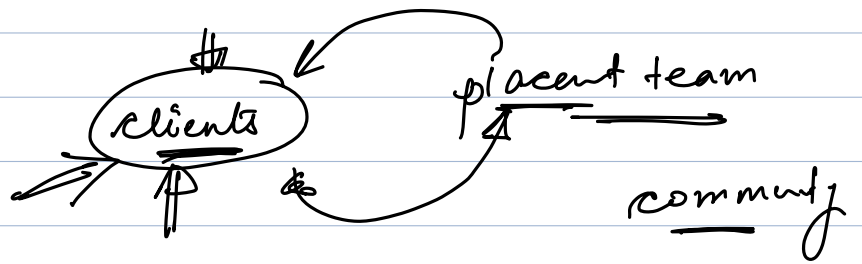
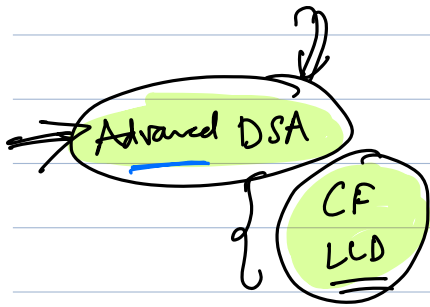
CF  
 Laps Module  
 LCD - 15-18 serv

LCD

{ Advance DSA } Mock interview → LCD  
 ↓ mock in

class  
 Assignment / Homework

Advanced DSA  
 Inter



problem solved  
probley given  
answ/HW

