Today's Content

↳ How to Calculate No. of iteration

↳ Big o notation

Time complexity != Time taken to Execute

↳ No. of iterations [Function of length of the input]

Rate of change of time as a function of input.

ASYMPTOTIC NOTATIONS :

- → Big O — O — Worst Case } Mostly used.
- → Omega — Ω — Best Case
- → Theta — Θ — Average Case

Quiz 1: Sum of N natural numbers:

↳ $\frac{n*(n+1)}{2}$

Quiz 2: $[3, 10] \rightarrow 8$

[→ inclusive (→ exclusive	$[a, b]$	$[a, b)$	(a, b)
	$b - a + 1$	$b - a$	$b - a - 1$

Quiz 3: $N \rightarrow \frac{N}{2} \rightarrow \frac{N}{4} \rightarrow \frac{N}{8} \dots\dots 1$

4 ($\log N$)

$\log_2 N + 1$

$$\left[\left[\left[\left[\left(N\right)\frac{1}{2}\right]\times\frac{1}{2}\right]\times\frac{1}{2}\right]\times\frac{1}{2}\right]\times\frac{1}{2} = 1$$

$$\frac{N}{2^x} = 1$$

8 ↓ 4 ↓ 2 ↓ 1 — 3

$N = 2^x \Rightarrow \log N = \log_2 2^x$

↳ $\log_a a^x = x$

$\log_2 N = x$

$\log_2 N$: How many times you are dividing

$\log_2 N + 1$: How many no.s will be there while dividing

A.P: Arithmetic Progression

Series:

4 7 10 13 16 19

+3 +3 +3 +3 +3

1st 2nd 3rd 4th N^{th} term

$a \quad (a+d) \quad (a+2d) \quad (a+3d) \quad \ldots\ldots \quad a+(n-1)d$

first term = a

Common difference = d

// Sum of N terms in A.P = $\frac{n}{2}[2a+(n-1)d]$

Geometric Progression:

3 6 12 24 48 ....

1st 2nd 3rd 4th N^{th} term

$ar^0 \quad ar^1 \quad ar^2 \quad ar^3 \quad \ldots \quad ar^{N-1}$

a: first term

r: Common ratio

// Sum of N terms in G.P = $a+\left(\frac{r^N-1}{r-1}\right)$

Qui2)

```
void func (int N){
    S=0
    for(int i=1; i<=N; i++){
        S=S+i;
    }
    return S;
}
```

[1, 2, 3, ---- N]

N-1+1 = N iter.

⇓

O(N)

Q)

```
void func (int N, int M) {
    for(int i=1; i<=N; i++) {        [1 ... N]
        if (i%2==0) {                N iteration
            Print(i);
        }
    }

    for(int i=1; i<=M; i++) {        [1 ... M]
        if (i%2==0) {                M iterations
            Print(i);
        }
    }
}
```

Total: (N+M) iterations

⇓

O(N+M) N≥M → O(N) M≥N → O(M)

Quiz 4:

```
int func (int N) {
    S=0;
    for(int i=0; i<=100; i++) {      [0 ... 100]
        S = S+i+i^2;                 100 - 0 + 1 = 101
    }
    return S;
}
```

⇓

101 iteration

Quiz 5)

```
void fun (N){
    for(int i=1; i*i<=N; i++){
        S = S + i^2;
    }
    return S;
}
```

$i^2 <= N$

$i <= \sqrt{N}$ [1, ... $\sqrt{N}$]

$\sqrt{N} - 1 + 1 \Rightarrow \sqrt{N}$ iteration

$O(\sqrt{N})$

Quiz 6)

```
void fun (N){
    int i = N;
    while (i!=1){
        i = i/2;
    }
}
```

$i = N$

$i = \frac{N}{2}$

$i = \frac{N}{4}$

$i = \frac{N}{8}$

$\Rightarrow \log_2 N$ iteration

$O(\log_2 N)$

$i = 1$

$N \rightarrow \frac{N}{2} \rightarrow \frac{N}{4} \rightarrow \cdots\cdots\cdots \rightarrow 1$

$\frac{N}{4} = \log_2 2^0$

Quiz 7)

```
void fun (int N){
    S=0;
    for(int i=0; i<=N; i=i*2){
        S= S+i;
    }
}
```

[0, 0, 0 .... 0]

⇓

infinite times

Quiz 8)

```
void fun (int N){
    int S=0;
    for (int i=1; i<N; i=i*2){
        S= S+i;
    }
}
```

1 ↓ 2 ↓ 4 ↓ 8 ⋮ ↓ N

⇒ $\log_2 N$

N → 1 ⇒ $\log_2 N$

1 → N ⇒ $\log_2 N$

Nested loops

Quiz 9)

```
void fun1 (int n){
    for (int i=1; i<=10; i++){
        for(int j=1;j<=N;j++){
            Print (10);
        }
    }
}
```

Num of iteration = How many times you are running the statement

i	j	
1	[1, N]	N
2	[1, N]	N
3	[1, N]	N
⋮		+
10	[1, N]	N

10 * N iteration

Quiz 10)

```
void fun2 (int n) {
    for(int i=1; i<=N; i++){
        for(int j=1;j<=N;j++){
            Print (i*j);
        }
    }
}
```

i	j	
1	[1, N]	N
2	[1, N]	N
⋮		+
N	[1, N]	N

N * N iteration

Quiz 11:

```
void fun3 (int N){
    for(int i=0; i<N; i++){
        for(int j=0; j<=i; j++){
            Print(i*j);
        }
    }
}
```

i	j	
0	[0,0]	1
1	[0,1]	2
2	[0,2]	3
⋮	⋮	⋮
N-1	[0,N-1]	N

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

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

break: 10:30 Pm

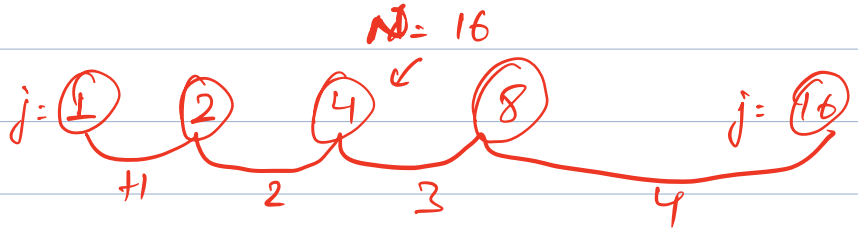
Quiz 12:

```
void fun4 (int N){
    for(int i=1; i<=N; i++){
        for(int j=1; j<=N; j=j*2){
            Print(i*j);
        }
    }
}
```

i	j	
1	[1,N] *2	$\log_2 N + 1$ +
2	[1,N]	$\log_2 N + 1$ +
⋮	⋮	⋮
N	[1,N]	1 + $\log_2 N + 1$

$N*(\log_2 N + 1)$ iterations

$= N\log_2 N + N = O(N \log_2 N)$



$\log_2 16 = 4$

$4 + 1 = 5$ times

Quiz 13)

```
void fun5(int N){
    for(int i=1; i<=2^N; i++){
        Print(i);
    }
}
```

$[1, 2, 3 \ldots 2^N]$

$2^N - 1 + 1 = 2^N$ iterations

Quiz 14:

```
void fun6(int N){
    for(int i=1; i<=N; i++){
        for(int j=1; j<=2^i; j++){
            Print(i*j);
        }
    }
}
```

i	j	
1	$[1, 2^1]$	2^1
2	$[1, 2^2]$	2^2
3	$[1, 2^3]$	2^3
⋮		
N	$[1, 2^N]$	2^N

GP sum: $a * \left(\frac{r^N - 1}{r - 1}\right)$

$2^1 + 2^2 + 2^3 + \ldots\ldots 2^N = 2 * \left(\frac{2^N - 1}{2 - 1}\right)$

(GP)

$= 2 * (2^N - 1)$

Quiz 15)

```
for(int i=N; i>0; i=i/2){
    for(int j=1; j<=i; j++){
        Print(i*j);
    }
}
```

i	j	
N	[1, N]	N
N/2	[1, N/2]	N/2
N/4	[1, N/4]	N/4
⋮	⋮	⋮
1	[1, 1]	1

$\Rightarrow 2+2^N-2 \Rightarrow O(2^N)$

$N + \frac{N}{2} + \frac{N}{4} + \cdots + 1$

no. of terms $= \log_2 N + 1$

(I) GP sum: $a + \left(\frac{r^{\text{No. of terms}} - 1}{r-1}\right) = N + \frac{\left(\left(\frac{1}{2}\right)^{\log_2 N + 1} - 1\right)}{\frac{1}{2} - 1}$

(II) $= \frac{N}{-1/2} + \left[\left(\frac{1}{2}\right)^{\log_2 N + 1} - 1\right]$

$2^{\log_2 N} = N$

$2^{\log_2 N} = x$

$\log_2 N = \log_2 x$

$N = x$

(III) $= -2N + \left[\frac{1}{2^{\log_2 N} \cdot 2} - 1\right]$

(IV) $= -2N \left[\frac{1}{2N} - 1\right]$

(V) $= -\frac{2N}{2N} - (-2N) \Rightarrow 2N - 1$

// Comparison

↳ for large value of N

$1 < \log\log N < \sqrt{N} < N < N\log N < N\sqrt{N} < N^2 < 2^N$

$1 < \log 10 < \sqrt{10} < 10 < 10\log 10 < 10\sqrt{10} < 10^2 < 2^{10}$

// How to write Big O → what? why?

(i) Calculate iterations based on loop

(ii) Neglect lower order terms.

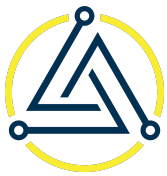
or

keep only the highest order term

(iii) Neglect Constant Co-efficient

eg: $10N^2 + 100N + 10^4$ ∴ total no. of iteration

$O(N^2)$



Quiz 1) $4N^2 + 3N + 10^6$

$\downarrow O(N^2)$

Quiz 2) $4N + 3N\log N + 10^6$

$\downarrow O(N\log N)$

Quiz 3) $4N\log N + 3N\sqrt{N} + 10^6$

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

$\frac{1}{2^n} = 1 \Rightarrow n = 0$

$N \rightarrow \frac{N}{2} \rightarrow \frac{N}{4} \rightarrow \frac{N}{8} \quad \cdots\cdots \quad (1)$

$\Downarrow$

TODO

N^{th} term ar^{N-1}

(1) $\quad 1 = N * \left(\frac{1}{2}\right)^{n-1}$

$a^{b+c} = a^b * a^c$

$2^{10} = 2^6 * 2^4$

↳ you didn't come this far, only to come this far