Today's agenda

- No. of iteration
- Big O notations

Quiz

↳ How many numbers are in range [3,10] {corners included}

{3 4 5 6 7 8 9 10} → 8

[a,b] → b-a+1 (inc, inc) | [a,b) = b-a (inc, exc) | (a,b) = b-a-1 (exc, exc)

Quiz

↳ No. of steps for $N \to \frac{N}{2} \to \frac{N}{4} \to \dots\dots \to 1$

$x * 2 = 10 \Rightarrow x = 10/2$

$x + 2 = 10 \Rightarrow x = 10 - 2$

$x^2 = 10 \Rightarrow x = \sqrt{10}$

$x^2 = 10 \Rightarrow 2 = \log_x(10)$

$\log_b a$ ← number, result, base

$\Rightarrow \log_2 10 = ans \to 3.33\ldots$

$10 = 2^{ans}$

$ans = 2^{3.3\ldots}$

$\log_b a = ans \Rightarrow a = b^{ans}$

→ $\log_2 64 = ans$ → $64 = 2^{ans}$ → 6

→ $\log_2 32 = ans$ → $2^{ans} = 32$ → 5

→ $\log_2 33 = ans$ → $2^{ans} = 33$ → $5.\ldots$

Properties

(i) $\log_a a^n = n$

ex: $\log_5 5^{10} = 10$

(ii) $\log_c (a*b) = \log_c a + \log_c b$

ex: $\log_2 10 = \log_2 2 + \log_2 5$

Quiz

↳ No. of steps for $N \to \frac{N}{2} \to \frac{N}{4} \to \ldots\ldots \to 1$

$$\left(\left(N * \frac{1}{2}\right) * \frac{1}{2}\right) * \frac{1}{2} \ldots = 1$$

$$\frac{N}{2^{count}} = 1 \Rightarrow N = 2^{count}$$

$2^{count} = N$

$count = \log_2 N$

A.P → Arithmetic Progression

3 (+4) 7 (+4) 11 (+4) 15 (+4) 19 (+4) 23 (+4) 27 . . . .

first term = a = Starting number
Common diff = d = difference between Consecutive nos

1	2	3	4	nth term
(a+0*d)	(a+d)	(a+2*d)	(a+3*d)	. . . . . a+(N-1)*d

↳ Sum of first N terms : $\frac{N}{2}[2*a+(N-1)*d]$

G.P → geometric Progression

3 (*2) 6 (*2) 12 (*2) 24 (*2) 48 (*2) 96 . . .

50 (*½) 25 (*½) 12.5 (*½) 6.25

first term : a : Starting numbers.
Common ratio: r : multiple to get next no.

1	2	3	4	nth term
a	a*r	$a*r^2$	$a*r^3$	. . . . . $a*r^{n-1}$

↳ Sum of N terms in G.P : $a * \frac{r^N-1}{r-1}$

Quiz

```
int Sum=0;
for (int i=1; i<=N; i++){
    Sum = Sum+1;
}
```

$[1, N] \rightarrow N-1+1$

N^{th} iterations

$\Downarrow$

$O(N)$

Quiz

```
void func(int N, int M){
    for(int i=1; i<=N; i++){
        Print(i);
    }
    for (int i=1; i<=M; i++){
        Print(i);
    }
}
```

$\rightarrow [1, N] = N-1+1 = N$ iterations

$\rightarrow [1, M] = M-1+1 = M$ iteration

Total = $N+M$ iterations

$\Downarrow$

$O(N+M)$

- $N \geq M \rightarrow O(N)$
- $M \geq N \rightarrow O(M)$

Quiz

```
int fun (int N){
    int S=0;
    for (int i=0; i<=100; i++){
        S= S+ i2;
    }
    return S;
}
```

→ [0, 100] → 100 - 0 + 1 = 101 iterations

↓

O(1)

Quiz

```
void fun (int N){
    int S=0;
    for (int i=1; i*i<=N; i++){
        S= S+ i2;
    }
    return S;
}
```

→ [1, √N] ⇒ √N - 1 + 1 : √N iterations

↓

O(√N)

Quiz

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

N → N/2 → N/4 → ... → 1

→ logN

≠ O(logN)

Quiz

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

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

infinite

Quiz2

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

$1 \to 2 \to 4 \to 8 \dots \dots \dots \frac{N}{4} \to \frac{N}{2} \to N$

$\log N$ times (or $\log N$)

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

$N \to \frac{N}{2} \to \frac{N}{4} \to \frac{N}{8} \dots 2 \to 1$

$\log N$

1 2 3 4 5 $\Rightarrow$ 5

5 4 3 2 1 $\Rightarrow$ 5

Bolak till 9:30 Pm

Nested loops

Quiz

```
void fun (int n){
    int s=0;
    for (int i=1; i<=10; i++){
        for (int j=1; j<=n; j++){
            s = s+10;
        }
    }
}
```

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

10*N iterations

$O(N)$

Quiz:

```
void fun (int n){
    int s=0;
    for (int i=1; i<=n; i++){
        for (int j=1; j<=n; j++){
            s = s+10;
        }
    }
}
```

i	j	Count
1	[1, N]	N
2	[1, N]	N
3	[1, N]	N
⋮		
N	[1, N]	N

(N*N) iterations

$O(N^2)$

Qui2:

```
void fun (int N){
    int S=0;
    for (int i=1; i<=N ; i++){
        for (int j=1; j<=i ; j++){
            S = S+10;
        }
    }
}
```

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

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

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

$O(N^2)$

Qui2

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

$[1, 2^N]$ → 2^N iterations

$O(2^N)$

Quiz:

```
void fun (int N){
    int S=0;
    for (int i=1; i<=N; i++){
        for (int j=1; j<=2^i; j++){
            S = S+10;
        }
    }
}
```

i	j	Count
1	[1, 2^1]	2^1
2	[1, 2^2]	+ 2^2
3	[1, 2^3]	+ 2^3
⋮		+
N	[1, 2^N]	2^N

$2 + 2^2 + 2^3 + 2^4 \cdots\cdots 2^N =$ ($*2$, $*2$, $*2$)

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

iterations

$$a * \frac{r^N - 1}{r - 1}$$

$2*2^N - 2$

$O(2^N)$

Comparison of iteration

$1 < \log_2 N < \sqrt{N} < N < N * \log N < N\sqrt{N} < N^2 < 2^N \ldots$

Time Complexity

↳ Big O Notation → Approx iteration

(i) Calculate iteration
(ii) Neglect lower order terms
(iii) Neglect Constants

ex: $(2N^2 + 2N + 3N^0)$ iteration

⇓

$O(N^2)$

ex:

$8N^2 + 15N + 100$

↳ $O(N^2)$

ex: $4N + 8N\log N + 10$

↳ $O(N \log N)$

Ex: ~~10~~ $N \log N + 15\sqrt{N} + 1000$

is $O(N \log N)$