→ Time Complexity / Space Complexity
→ O( )
→ Worst / Best / Avg
→ Why TLE?

Thursday's Session

→ Calculating Iterations

1Q) 21 Quizzes – Top 3 Diwali Scaler
Pen / Paper / Try it

1Q)

Given $\underbrace{N \rightarrow N/2 \rightarrow N/4 \rightarrow N/8 \text{ - - - - } 1}$: $\log_2 N$ { Recording Problem Solving }

2Q) [3 10] → 3 4 5 6 7 8 9 10 : 8 Numbers

→ 10 − 3 + 1 = 8

[a b] → All Elements from a to b Included

→ b − a + 1

[a b) → All Elements from a to b, b Excluded

→ b − a

(a b) → b − a − 1

Note: 9/2 → 4
11/2 → 5

3Q) Sum of N Elements in A.P → Common diff between 2 consecutive elements same.

$$\underbrace{a, \; a+d}_{d}, \; \underbrace{a+2d}_{d}, \; \underbrace{a+3d}_{d}, \; \underbrace{a+4d}_{d}, \; \ldots$$

a → first Term, d → Common Difference

$$S_N = \frac{N}{2}\left[2a + (n-1)d\right]$$

4Q) $\log_a a^N = N$, $\log_2 2^{10} = 10$, $\log_2 2^r = r$

Problems:

1Q
```
i = 1; i <= N; i++) {
    S = S + i
}
```

i = [1, ⟶ N]

start	end	Iterations
1	N	$N - 1 + 1 = N$

→ $O(N)$

2Q
```
i = 1; i <= N; i += 2)
    S = S + i
}
```

$i = 1, 3, 5, 7, 9, \ldots$

i value iterating in All Odd Number [1 N] → $[N+1]/2$ → $O(N)$

N = 8 : odd [1 - 8] = 4

N = 9 : odd [1 - 9] = 5

N = 10 : odd [1 - 10] = 5

N = 15 : odd [1 - 15] = 8

Given N, How Many odd [1 N]

ans → $N/2$: N Even

ans → $N/2 + 1$: N Odd

$9/2$

cnt: $(N+1)/2$

N = 8: $9/2 = 4$

N = 9: $10/2 = 5$

3Q) $i = 0;\ i <= 100;\ i{+}{+})\{$ $\quad i = [0, 100] \Rightarrow \{100 - 0 + 1\}$

$\Rightarrow \{101\}$

$\}$ $\quad \Rightarrow O(1)$

4Q) $i = 1;\ \underbrace{i * i <= N};\ i{+}{+})\{$

$\}$

$i * i = i^2, \quad i * i <= N \Rightarrow i^2 <= N \Rightarrow i <= \sqrt{N}$

$i = 1;\ i <= \sqrt{N};\ i{+}{+})\{$ $\quad i = [1, \sqrt{N}] \Rightarrow [\sqrt{N} - 1 + 1]$

$[\sqrt{N}]$

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

S3) Given $N > 0$

$i = N$

```
while(i >= 1){
    i = i/2
}
```

i Before	Total Iterations	i After $(i = i/2)$
$N \geq 1$	1	$N/2 : N/2^1$
$N/2 \geq 1$	2	$N/4 : N/2^2$
$N/4 \geq 1$	3	$N/8 : N/2^3$
$N/8 \geq 1$	4	$N/16 : N/2^4$

After k iterations $\rightarrow$ $N/2^k$

After x iteration $\rightarrow$ $N/2^x = 1$

$N/2^k = 1$, $2^k = N \rightarrow$

Apply $\log_2$ on both sides

$\log_2 2^k = \log_2 N$

$k = \log_2 N$ // Total no: of iterations are $\log_2 N$

$\Rightarrow O(\log_2 N)$

6Q)

```
i=0; i<=N; i=i*2){
|
}
```

i Before	Iteration	i After ($i = i*2$)
0	1	0
0	2	0
0	3	0
0	↓	
	∞	

7Q) ⟺ 5Q)

```
i=1; i<=N; i=i*2){
|
}
```

→ $\log_2^N$ iterations

i Before	Iterations	i After ($i=i*2$)
$1 <= N$	1	$2 : 2^1$
$2 <= N$	2	$4 : 2^2$
$4 <= N$	3	$8 : 2^3$
$8 <= N$	4	$16 : 2^4$

After k iterations $i = 2^k$

When will this loop stop

$i > N$

$x = \log_2^N$ → Total Iterations

$O(\log N)$

After x iterations we stop

$2^x = i > N$

$2^x > N$ $\{x > \log_2^N\}$

8Q)

```
i = 1; i <= 10; i++){
    j = 1; j <= N; j++){
    }
}
```

i	j	Total Iterations
1	[1 N]	N
2	[1 N]	N
3	[1 N]	N
4	[1 N]	N
.		
.		
10	[1 N]	N

$10 * N \rightarrow O(N)$

9Q

```
i = 0; i < N; i++){
    j = 0; j < N ; j++){
    }
}
```

i	j	Total Iterat
0	[0, N)	N-0 = N
1	[0, N)	N-0 = N
2	[0, N)	N-0 = N
3	[0, N)	N-0 = N
.		
N-1	[0, N)	N

N^2

$\rightarrow O(N^2)$

loops

```
for (i = 0; i < N; i++){
    for (j = 0; j <= i; j++){
        ...
    }
}
```

10:20 PM break

After break, strings

i	j → [0 i]	Total Iterations
0	[0 0]	1
1	[0, 1]	2
2	[0 2]	3
3	[0 3]	4
⋮		
N-1	[0, N-1]	N

Sum of N Natural Numbers → $\frac{(N)(N+1)}{2}$

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

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

11Q

```
i = 1; i <= N; i = i+2){
    j = 1; j <= i; j++){
        ====
    }
}
```

i	j [1 i]	Total Iterations
1	[1 1]	1
3	[1 3]	3
5	[1 5]	5
7	[1 7]	7
		9

i = i+2

i = i+2

i = i+2

Sum of all odd Numbers

[1, N]

Total Iterations

$1+3+5+7+9 \ldots$

$a = 1$

$d = 2$

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

Sum of N Terms in AP

$S_N = N/2\,[2a + (n-1)d]$

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

Sum of

1 odd = 1

2 odd = $4 = 2^2$

3 odd = $9 = 3^2$

4 odd = $16 = 4^2$

k odd = k^2

$S_N = \frac{(N+1)}{4}\left[2*1 + \left[\frac{N+1}{2} - 1\right]2\right]$

$S_N = \frac{(N+1)}{4}\left[2 + \frac{[N+1-2]}{\cancel{2}} \cancel{*2}\right]$

$S_N = \frac{(N+1)[\cancel{2} + N+1 - \cancel{2}]}{4}$

$S_N = \frac{(N+1)(N+1)}{4}$

$S_N = \frac{N^2 + 2N + 1}{4}$

$O(N^2)$

11Q)

```
i = 1; i <= N; i++){
    j = 1; j <= N; j = j * 2){
    }
}
```

i	j [1 N]	Total Iterations
1	[1 N]	$\log_2^N$
2	[1 N]	$\log_2^N$
3	[1 N]	$\log_2^N$
⋮		
N	[1 N]	$\log_2^N$

$O(N\log^N)$

12Q)

→ No overflow

```
i = 1; i <= (1<<N); i++){
}
```

$a << N = a * 2^N$

$1 << N = 1 * 2^N$

$1 << N = 2^N$

```
i = 1; i <= 2^N; i++){
}
```

→ $i \rightarrow [1, 2^N]$

$= 2^N$ iterations

$= O(2^N)$

138)

```
i = 1; i <= N; i++) {
    j = 1; j <= (1<<i); j++) {
        ...
    }
}
```

2^i

i	j : [1, 2^i]		
1	[1, 2^1]	2^1	Total Iterations
2	[1, 2^2]	2^2	
3	[1, 2^3]	2^3	
4	[1, 2^4]	2^4	
⋮			
N	[1, 2^N]	2^N	

Total Iterations =

$$2^1 + 2^2 + 2^3 + 2^4 + \dots 2^N \} \text{ G.P}$$

$$\left\{ a = 2,\ r = 2,\ N \right\}$$ Sum of N Terms in G.P = ($r > 1$)

$$\Rightarrow \frac{(a)(r^N - 1)}{r - 1} \Bigg\} \frac{(2)(2^N - 1)}{2 - 1} = (2) * (2^N) - 2$$

$$= 2 * 2^N - 2$$

$$= O(2^N)$$

Big (O) Notation

→ Iterations → N → Input size $N \geq 1$

→ Neglect lower order terms.
   lower degree

→ Neglect constant coefficients

Ex1: $100N^2 + 5N + 10^4 = O(N^2)$

Higher order: $N\log N > N$

Ex2: $5N + 6N\log N + 8\sqrt{N}$ → $O(N\log N)$

$\log_2 N, \; N > 1$

$\log_2 N > 1$

$N\log N > N$

Ex3: $N^2 + 6N\sqrt{N} + 10^4 N\log N = O(N^2)$

Ex4: $F(N) = 4N^2 + 3N + 10^6 : O(N^2)$

Ex5: If iteration are constant : $O(1)$

$\log(N) < \sqrt{N} < N < N\log N < N\sqrt{N} < N^2 \cdots 2^N < N!$

Doubts:

```
func (int N){
    i = 1; i <= 100; i++){
    }
}
```

→ 100

→ $O(1)$

2^N, $N!$

$N! > 2^N$

$N < 4$ | $N > 4$

$N! > 2^N$

```
i = 1; i <= N; i++){
    j = 1; j <= N; j++){
        k = 1; k <= j; k++){
        }
    }
}
```