

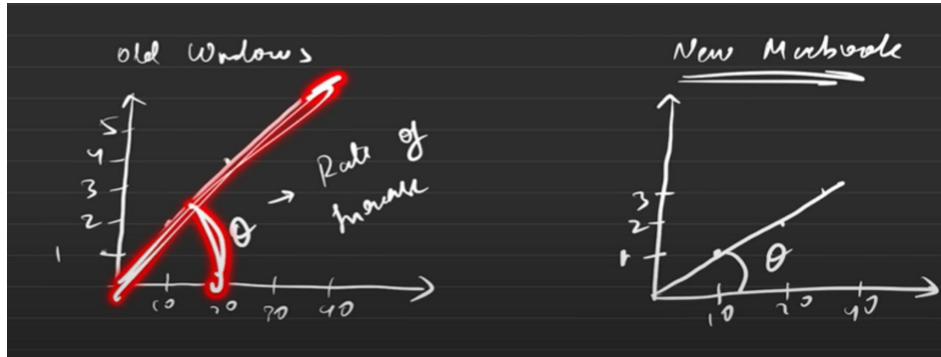
Time Complexity

12 July 2021 18:34

DEFINE

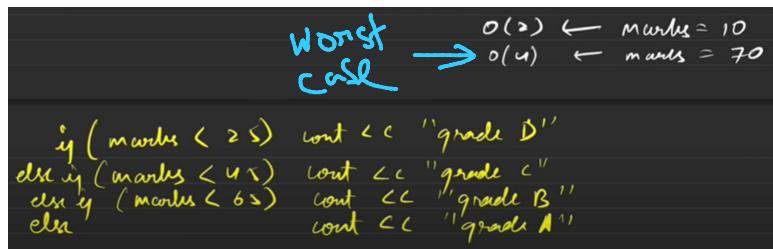
The rate at which time taken increases with respect to the input size is known as time complexity.

Representation of Time Complexity: Big-O-Notation [O(<No. Of Operations>)]

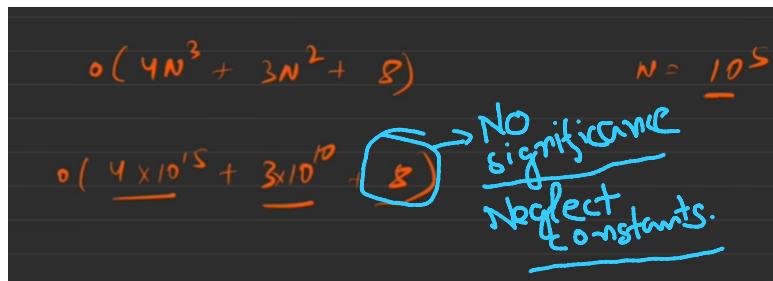


RULES

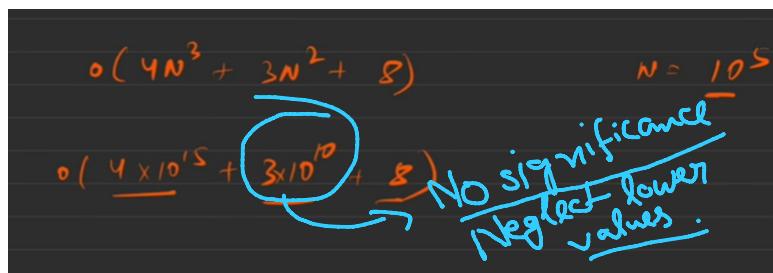
- Always compute TC considering worst case scenario.



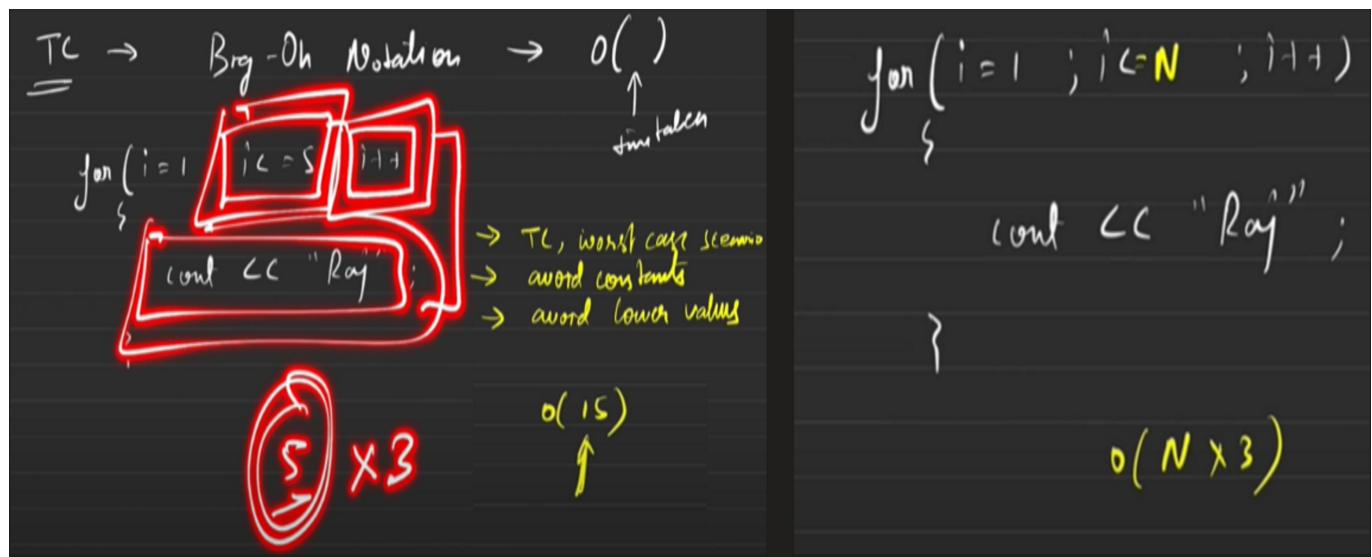
- Avoid constants.



- Avoid lower values.



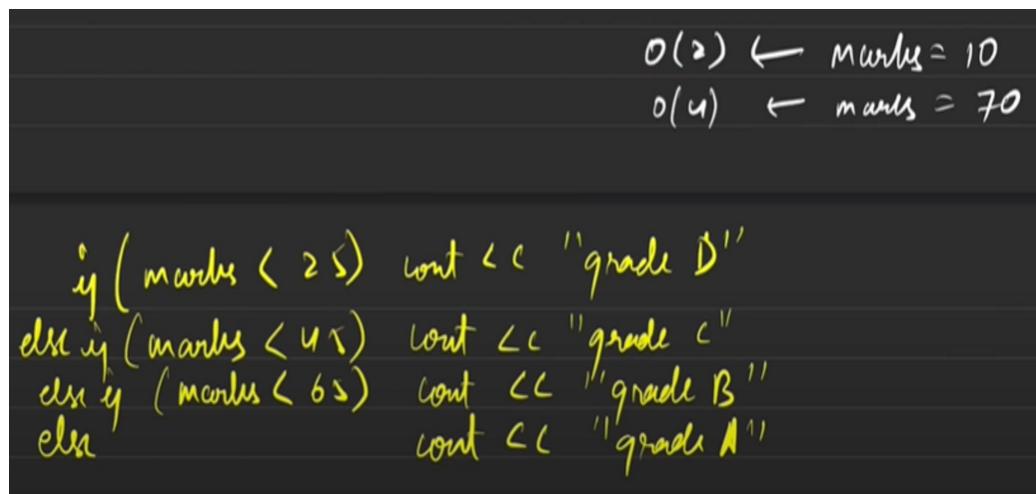
EXAMPLE



ASYMPTOTIC ANALYSIS

- Best case ($\Omega[\Omega]$ - Lower Bound).
- Average case ($\Theta[\Theta]$ - Average).
- Worst case (O - Upper Bound).

➤ *Best and worst case representation*



➤ *Average case representation*

It is defined as the sum of best and worst case divided by two.

Best + Worst

QUESTIONS

Q1:

```

for (int i = 0 ; i < N ; i++)
{
    for (int j = 0 ; j < N ; j++)
        {
            // Block of code
            // Constant Time.
        }
}

```

$N + N + N + \dots + N$

$N \times N$

(N^2)

$\begin{cases} i=0 & \{j=0, 1, 2, 3, \dots, N\} \\ i=1 & \{j=0, 1, 2, 3, \dots, N\} \\ i=2 & \{j=0, 1, 2, \dots, N\} \\ \vdots & \vdots \\ i=N-1 & \{j=0, \dots, N\} \end{cases}$ TUF

Q2:

```

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

```

$i = 0 \quad \{j = 0\}$

$i = 1 \quad \{j = 0, 1\}$

$i = 2 \quad \{j = 0, 1, 2\}$

\vdots

$i = n-1 \quad \{j = 0, 1, 2, \dots, n-1\}$

$(1 + 2 + 3 + 4 + \dots + n)$

$\frac{N \times (N+1)}{2} = \frac{N^2}{2} + \left\lceil \frac{N}{2} \right\rceil$
 $= O\left(\frac{N^2}{2}\right) \approx O(N^2)$