Theme: Time complexity

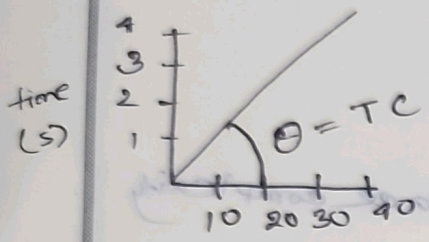
Date: / /

Sat Sun Mon Tue wed Thu Fri

* What is time complexity?

→ code ← 1s, 2s, 3s > Time taken != time complexity

→ Rate at which the time taken increases with respect to input size.



Every piece of code takes time in

TC → Big O notation

→ O (Time taken)

Example for (i=1; i<5; i++) { cout << "Ray"; }

steps:
1 - init
2 - condn
3 - incre
4 - Print
5
6
...

Total Num of steps → (5) x 3

∴ O(15)

But, i < N → O(3N)

* Three Rules
→ Worst Case Scenario
→ Avoid constants
→ Avoid lower values

Ex:

```
for (int i=0; i<n; i++)        3N + 3N
  {
    for (int j=0; j<n; j++) { single line }
  }
```

outer → start 0; run n times
inner → ~ . ~ n ~

i=0 [j = 0 ... i ...] 2
i=N-1 [j = ... 2]

N+N ....+N
= NXN

TC = O(N²)

Theme:

Date: / /
Sat Sun Mon Tue wed Thu Fri

*Space Complexity
→ memory space
→ Big O notation
→ auxiliary space + input space
↳ store the input

↳ space to take solving the problem

```
as
↓
C   =   a   +   b
```

=> int a[N] ⟶ Big O(N) ⟶ space complexity

Comp Programing:

code ⟶ server 1s ≈ 10^8 operations

TC = 1s হলে থাকলে, $O(10^8)$