



Welcome

TIME COMPLEXITY
&
SPACE COMPLEXITY

INTRODUCTION



Mahjabin Nibir
ID: 221-15-5057



MD.Habibur
Rahaman Habib
ID: 221-15-5174



Mehedi Hasan Saim
ID: 221-15-4844



Sohanur Rahman
ID: 221-15-4845



Ishrat Zahan
Easha
ID : 221-15-5212



❖ What is time complexity?

Time complexity is relation between size and running time(Operation).

❖ Why time complexity matters?

Efficiency: Helps identify algorithms that execute faster for larger inputs.


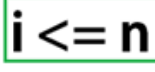
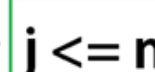
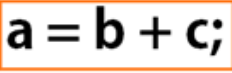
Scalability: Predicts performance when dealing with larger datasets.

Optimize Resources : Choose the best algorithm to save time and computational resources .

❑ 3 Ways to find Time Complexity

Best Case	Minimum number of steps
Average Case	Average number of steps
Worst Case	Maximum number of steps

Time Complexity

```
main ()  
{  
    x = y + z;   $O(1)$   
    for ( i = 1;  i <= n; i++)  $O(n)$   
    {  
        for(j = 1;  j <= n; j++)  $O(n)$   
        {  
             a = b + c;  $O(1)$   
        }  
    }  
}
```

So the Time Complexity:

$$T(n) = O(n^2)$$



What is space complexity?

The space Complexity of an algorithm is the total space taken by the algorithm with respect to the input size.

```
#include <stdio.h>

int sum(int a , int b) // Inter a and b
{
    return a + b;      // returning sum is integer too
}

int main() {
    // your code goes here
    int x = 10;
    int y = 20;
    printf("Sum of a & b =", sum(x,y));

    return 0;
}
```

There are five variables allocated in the memory:

The integer variables in the main function **x , y**.

The integer variables in the sum function **a , b**..

The third integer, returning variable **sum**.

A single variable uses 4 bytes
the total memory for this program is
20 bytes (5*4=20 bytes).

THANK
YOU

