**Data Structures & Algorithms Note**

* **Time Complexity**

What is Best Case Scenario in Time Complexity?

* A program having the least number of operations to execute is referred to as best case scenario in Time Complexity.

Time Complexity is represented by **three** terms:

* Big O Notation (O) [Worst Case scenario, most used]
* Theta Notation (θ) [Average Case scenario]
* Omega Notation (Ω) [Best Case scenario]

Ex: p = 0;

for(int i=1; p<=n; i++) { p = p+i; }

* i p

1. 1
2. 1+2
3. 1+2+3
4. 1+2+3+4

. .

. .

K 1+2+3+…+K [k(k+1)/2]

So, the number of operations will be k(k+1)/2

Assume that p>n

Now k(k+1)/2 > n

k(k+1) > n/2

k2 > n

So the Complexity is O(n2).

Ex: for(int i=1; i<n; i\*=2) { Statement }

* i

1 = 20

1\*2 =2 = 21

1\*2\*2 = 4 = 22

1\*2\*2\*2 = 8 = 23

.

.

1\*2\*2\*2\*…k = 2k

Now assume that, i>=n

2k >= n

2k = n

k =

So the Time Complexity is O().

Ex: for(int i=1; i\*i < n; i++) { statement; }

* i

1

2

3

.

.

i\*i

Now, i2 >n

i2 = n

i =

So, the Time Complexity is O().

|  |  |
| --- | --- |
| Loop | Big O() |
| for(int i=0; i<n; i++) | n = O(n) |
| for(int i=0; i<n; i+=2) | n/2 = O(n) |
| for(int i=n; i>1; i--) | O(n) |
| for(int i=0; i<n; i\*=2) | O() |
| for(int i=0; i<n; i\*=3) | O() |
| for(int i=0; i<n; i/=2) | O() |
| for(int i=1; i\*i < n; i++) | O() |