Time & Space Complexity

Tuesday, 7 May 2024 1:46 PM

TC ! = Time Taken

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

Rules → O (No of operations)

1. Calculate worst case scenario
2. Avoid constants
3. Avoid lower values

Example 1

$= O(N^{10} + 4N^3 + 3)$

applying 2 & 3

$= O(N^{10})$

Example 2

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

---

outer loop runs N times

inner loop runs N times

code inside runs N X N times

⇒ $O(N^2)$

Example 3

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

[code]

}
}

Sol^n :

Iterations

for i = 0 [j = 0] → 1

for i = 1 [j = 0, 1] → 2

for i = 2 [j = 0, 1, 2] → 3

⋮

for i = N-1 [j = 0, 1, 2 ... N-1]
↳ N

Net Iterations :-

$1 + 2 + 3 + \dots N$

$= \frac{N \times (N-1)}{2}$ [exact complexity]

$= \frac{N^2}{2} - \frac{N}{2}$

$\approx O(N^2)$

Space Complexity

Auxiliary Space → space needed to solve the problem

\+

Input Space → space needed to store the input

```
int a, b;
c = a+b;
```
} O(3)

a, b → input space

c → auxiliary space

```
int a[n];
```
→ O(n)

Sum a+b

```
int a, b;
b = a+b;
```

✗ takes lesser space but AVOID MNIPULATING INPUT VAR.

take extra variable

CP

Servers take

1 sec $\approx 10^8$ operations

n sec $\rightarrow$ n x 10^8 operations

---

if in problem given

TL (Time Limit) = 1 sec

make code $\approx O(10^8)$