DSA

DATA STRUCTURE AND ALGORITHM

Asymptotic Analysis (Asymptotic Notations)

- The idea is to measure order of growth.
- Does not depend on machine or programming language.
- No need to implement algo, we can just analyze.
- All operations are taken to be constant irrespective of the input.

Order of growth

The function $f(n)$ is said to be growing faster than $g(n)$ if -

$$\lim_{n \to \infty} \frac{g(n)}{f(n)} = 0$$

OR

$$\lim_{n \to \infty} \frac{f(n)}{g(n)} = \infty$$

$f(n)$ and $g(n)$ represent time taken.

Way to find and compare growth

- Ignore lower order terms
- Ignore leading term constant

eg - $f(n) = 2n^2 + n + 6$ ↳ ignore, ignore

OOG - n^2 (quadratic).

$g(n) = 100n + 3$
$OOG - n$ (linear)

COMPARISON -
Order of growth :
$c < \log\log n < \log n < n^{1/3} < n^{1/2} < n < n^2 < n^3 < n^4 < 2^n < n^n$

(1) $f(n) = C_1 \log n + C_2$
$g(n) = C_3 n + C_4 \log\log n + C_5$

$f(n) \quad C_1 \log n + C_2$

$f(n) = \log n$

$g(n) = n$

As we can $g(n)$ grows faster, according to comparison.
Hence $g(n)$ is a bad algo.

(2) $f(n) = C_1 n^2 + C_2 n + C_3$

$f(n) = n^2$

$g(n) = C_4 n \log n + C_5 n + C_6$

$g(n) = n \log n$

$f(n) = \cancel{n}^2$
$g(n) = \cancel{n} \log n$
$f(n) = n$
$g(n) = \log n$

$f(n)$ OOG $> g(n)$ OOG

Best, Average and Worst Cases:

Best Case: Minimum order of growth for an algo.
Average Case: Based on the conditions.
Worst Case: Maximum OOG and size for an algo.

Asymptotic Notations

Big O : Exact or upper
Theta : Exact
Omega : Exact or lower

eg Linear search.

```
int search (int ar[], int n, int x)
{
  for (int i=0; i<n; i++)
  {
     if (ar[i] == x)
        return i;
  return -1;
  }
}
```

~~Big O~~ → O(n)

Big O Notation

Direct way - Ignore lower order terms
Ignore leading term constant

Q) $3n^2 + 5n + 6$

n^2

ooq → n^2

$O(n^2)$

(Q) $3n + 10n\log n + 3$

$O(n\log n)$

(Q) $10n^3 + 40n + 10$

$O(n^3)$

(Q) $\{100, \log 2000, (10)^4 \ldots \ldots\}$

→ $O(1)$

because any constant values can be written as big $O(1)$.

(union) $\cup \{\frac{n}{4}, 2n+3, \frac{n}{100} + \log n, n + 10000 + \log n + 10 \ldots\}$

$\in O(n)$

$\cup (n^2+n, 2n^2, n^2+1000n, n^2+n\log n, \frac{n^2}{10000} \ldots\}$

$\in O(n^2)$

Big O works for multiple variables also -

- $100n^2 + 1000m + \cancel{n}$: $(n^2 + m)$
- $100m^2 + 200mn + 30m + 20n$: $(m^2 + mn)$