21 Jun

Big o theta Omega notation :→

II Insertion sort :→

A.length = n = input array length

Best case time Complexity = $an + b \times 1$

Worst " " " = $a'n^2 + b'n + c \times 1$

Space Complexity = 3 variables

$n\uparrow$ $(an+b)\uparrow$

↙

increases

ignore the Constants :→

→ $an+b \to O(n)$

→ $n\uparrow$ $n\uparrow\uparrow$

↳ much more increases

$10\times\left[\begin{array}{ll} n=10, & n^2 = 100 \\ n=100 & n^2 = 10,000 \end{array}\right] 100\times$

$an^2 + bn + c \to O(n^2)$

Analysis of algorithms.

① space Complexity = 3 variable

$= 3 \times 1$

Notations → Big O :→

↳ order of n

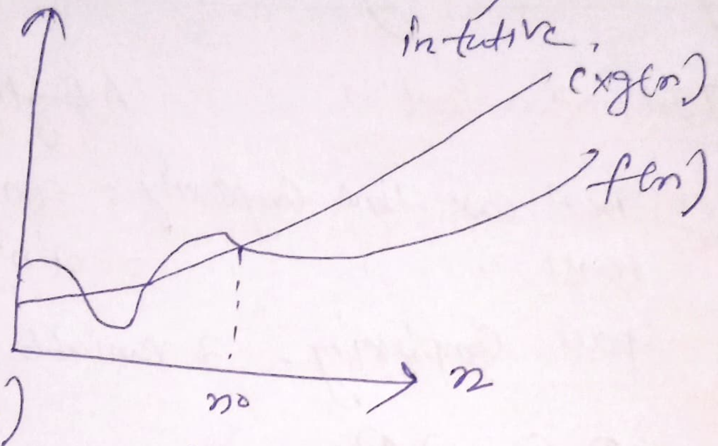
best

$f(n) = an + b + 1 \to O(n)$

$f(n) = a'n^2 + b'n + c' \to O(n^2)$

$f(n) = 3 = c \to O(1)$

intutive,



$n \geq n_0$

$f(n) = O(g(n))$

if and only if, there exists n_0 and c such that $0 \leq f(n) \leq c \times g(n)$ for all $n \geq n_0$

$f(n) = 2n^2 + 1 \times n + 3 \to O(n^2)$

$\uparrow a'$ $\quad \uparrow b'$ $\quad \uparrow c'$ $\quad \uparrow g(n)$

$2n^2 + 1 \times n + 3 \leq c \times n^2$ for all $n \geq n_0$

let, $c = 10$

$2n^2 + 1 \times n + 3 \leq 10n^2$ for all $n \geq n_0$

$n \geq 1$

$\Rightarrow 2n^2 + n + 3 \leq 10n^2$ for all $n \geq n_0$

$\Rightarrow 2 + \frac{1}{n} + \frac{3}{n^2} \leq 10$

$n = 1, \to 2 + \frac{1}{1} + \frac{3}{1} \leq 10$

$\Rightarrow 6 \leq 10$

$n=2,$

$2+\frac{1}{2}+\frac{3}{4} \le 10$ ✓

$n=3, \quad 3+\frac{1}{3}+\frac{3}{9} \le 10$

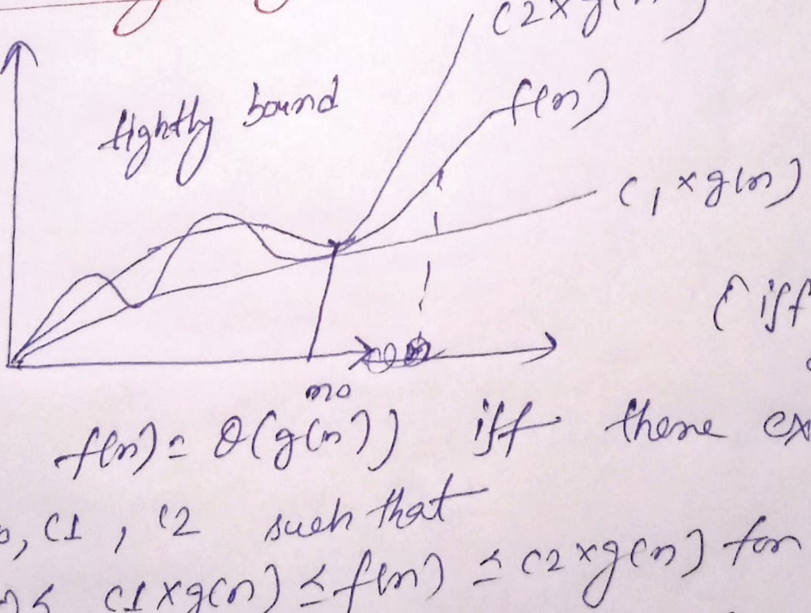
⋮

for all $n \ge 1$, $f(n) = 2n^2+n+3 \to O(n^2)$

$\Rightarrow f(n) \le 10 \times n^2$ for all $n \ge 1$

$\Rightarrow \boxed{f(n) = O(n^2)}$

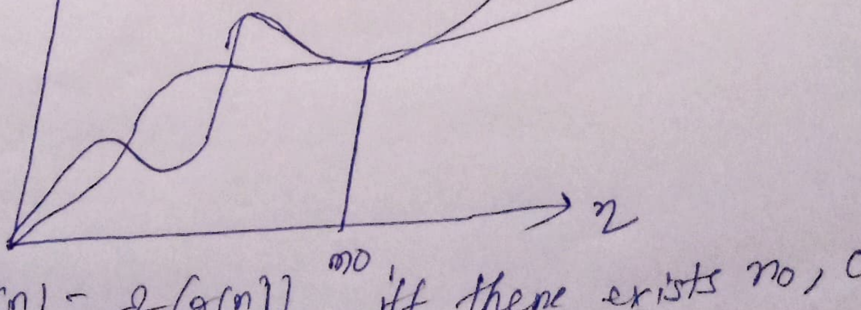
Notations → Big omega, Theta →



{ iff → if & only if }

$f(n) = \Theta(g(n))$ iff there exists n_0, c_1, c_2 such that $0 \le c_1 \times g(n) \le f(n) \le c_2 \times g(n)$ for all $n \ge n_0$.

Ω



$f(n) = \Omega(g(n))$ iff there exists n_0, c so that $0 \le c \times g(n) \le f(n)$ for all $n \ge n_0$