Asymptotic Notations

Asymptotic notations give us an idea about how good a given algorithm is compared to some other algorithm
Let us see the mathematical definition of "order of" now.

Primarily there are three types of widely used asymptotic notations.

1> Big oh notation (O)
2> Big omega notation (Ω)
3> Big theta notation (Θ) ⟶ Widely used one!

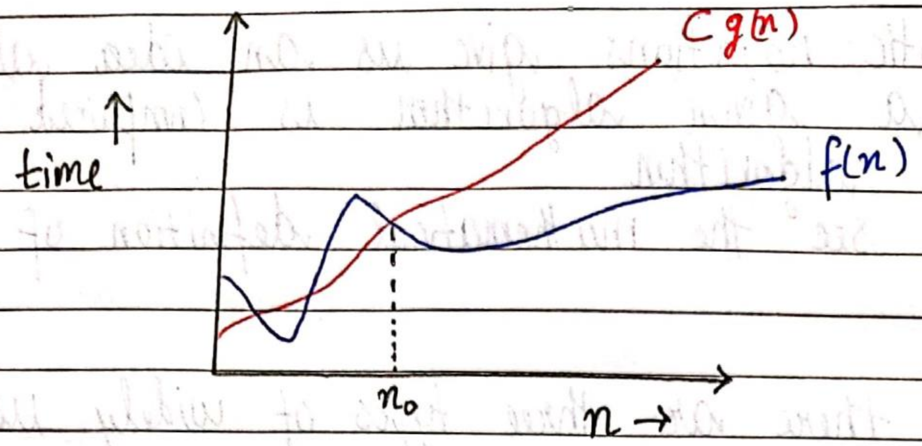
Big oh notation
Big oh notation is used to describe asymptotic upper bound
Mathematically, if $f(n)$ describes running time of an algorithm; $f(n)$ is $O(g(n))$ iff there exist positive constants c and n_0 such that

$$0 \leq f(n) \leq c\,g(n) \quad \text{for all } n \geq n_0$$

⇓ used to give upper bound on a function.

if a function is $O(n)$, it is automatically $O(n^2)$ as well!

Graphic example for Big oh (O)



Big Omega notation

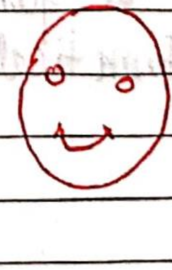
Just like O notation provides an asymptotic upper bound, Ω notation provides asymptotic lower bound. Let $f(n)$ define running time of an algorithm;

$f(n)$ is said to be $\Omega(g\,n)$ if there exists positive constants c and n_0 such that

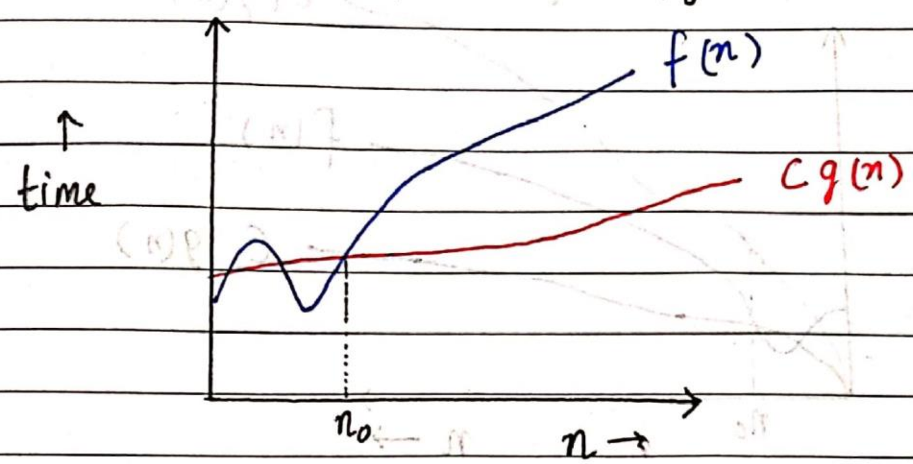
$$0 \leq c\,g(n) \leq f(n) \quad \text{for all } n \geq n_0$$

$\Downarrow$ used to give lower bound on a function

if a function is $O(n^2)$ it is automatically $O(n)$ as well



Graphic example for Big omega (Ω)



Big theta notation

Let f(n) define running time of an algorithm

f(n) is said to be $\Theta(gn)$ iff f(n) is $O(g(n))$ and f(n) is $\Omega(g(n))$

Mathematically,

$$0 \le f(n) \le C_1 g(n) \quad \forall n \ge n_0$$

$$0 \le C_2 g(n) \le f(n) \quad \forall n \ge n_0$$

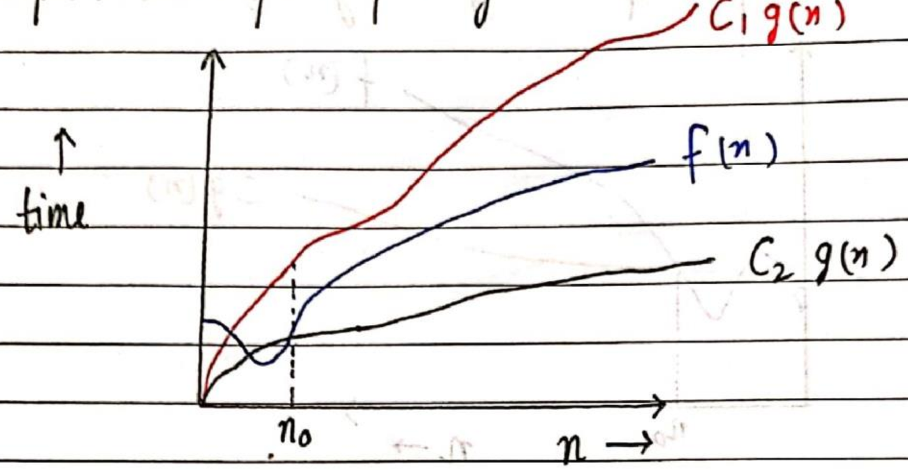
Sufficiently large value of n

Merging both the equations, we get:

$$0 \le C_2\, g(n) \le f(n) \le C_1\, g(n) \quad \forall n \ge n_0$$

The equation simply means there exist positive constants C_1 and C_2 such that f(n) is sandwiched between $C_2\, g(n)$ and $C_1\, g(n)$

Graphic example of Big theta



Which one of these to use ?

Since Big theta gives a better picture of runtime for a given algorithm, most of the interviewers expect you to provide an answer in terms of Big theta when they say "Order of".

Quick Quiz : Prove that n^2+n+1 is $O(n^3)$, $\Omega(n^2)$ and $\Theta(n^2)$ using respective definitions.

Increasing order of common runtimes

$$1 < \log n < n < n\log n < n^2 < n^3 < 2^n < n^n$$

Better

Worse

Common runtimes from better to worse