# Topics to discuss

1. Big Omega Notation ($\Omega$-notation)

2. Examples

# Omega - $\Omega$ Notation

This notation gives the tighter lower bound of the given algorithm and we represent it as $f(n) = \Omega(g(n))$.
At larger values of $n$, the tighter lower bound of $f(n)$ is $g(n)$.

## Definition :

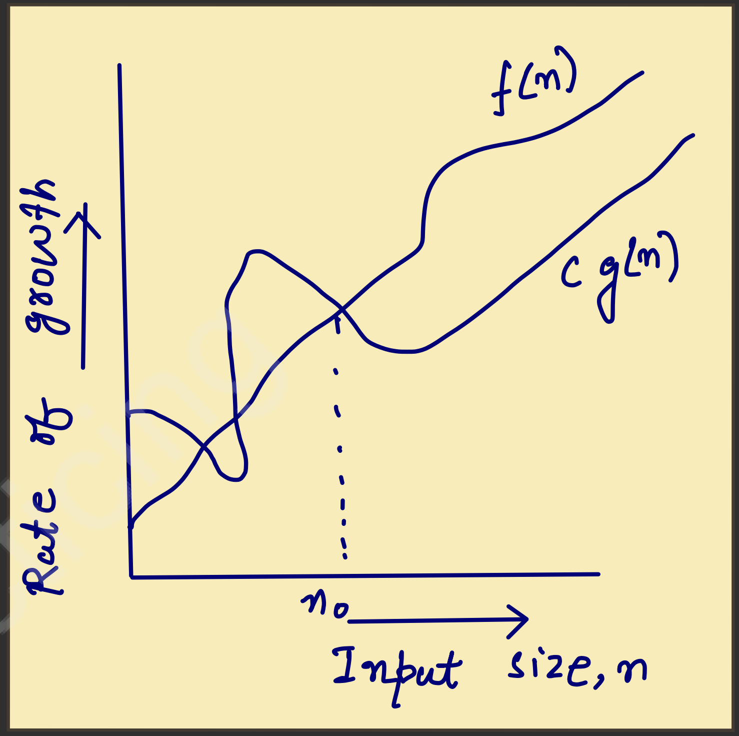
The $\Omega$ notation can be defined as $\Omega(g(n)) = \{ f(n) :$ there exist positive constants $c$ and $n_0$ such that $0 \le c\,g(n) \le f(n)$ for all $n \ge n_0 \}$.

$g(n)$ is an asymptotic tight lower bound for $f(n)$.

Our objective is to give the largest rate of growth $g(n)$ which is less than or equal to the given algorithm's rate of growth $f(n)$.

g(n) is an asymptotic tight lower bound for f(n).

Our objective is to give the largest rate of growth g(n) which is less than or equal to the given algorithm's rate of growth f(n).



1. Find the lower bound for $f(n) = 5n^2 - 1$

Solution: Given, $f(n) = 5n^2 - 1$

By definition, $0 \le c.g(n) \le f(n)$.

$$c.g(n) \le f(n)$$

$$4n^2 \le 5n^2 - 1$$

$c = 4$

$g(n) = n^2$

$$4n^2 - 5n^2 \le -1$$

$$-n^2 \le -1$$

$$n \ge 1$$

$f(n) = \Omega(g(n))$

$f(n) = \Omega(n^2)$

$c = 4, \ n \ge 1, \ n_0 = 1$

② Prove $f(n) = 100n + 5 \neq \Omega(n^2)$.

Solution :- $f(n) = 100n+5$

By definition, $0 \leq c \cdot g(n) \leq f(n)$

$c \cdot g(n) \leq f(n)$

$n^2 \leq 100n+5$ (Given, $g(n) = n^2$)

(False).

So, $f(n) = \Omega(n^2)$ is not true.

**Follow Now**

Start Practicing

i._am._arfin

Arfin Parween