Omega Notation (Lower Bound on Order of Growth)

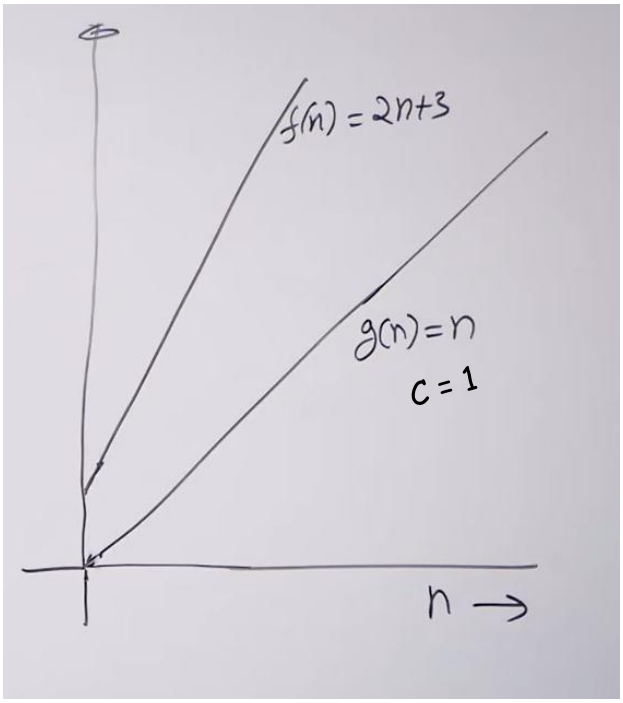
$f(n) = \Omega(g(n))$ if there exist positive constants c and n_0 such that $0 \leq cg(n) \leq f(n)$ for all $n \geq n_0$

Example: $f(n) = 2n + 3$ can written as $\Omega(n)$.

Place, $c = 1$

⇨ $cg(n) = n$

⇨ $n \leq 2n + 3$ $\quad n_0 = 0$



$f(n) = 2n^2 + 3n + 6$

$= \Omega(n^2)$

1. $\{n/4, n/2, 2n, 3n, 2n + 3, n^2, 2n^2, \ldots\ldots, n^n\} \in \Omega(n)$
2. If $f(n) = \Omega(g(n))$, then $g(n) = O(f(n))$
3. Omega notation is useful when we have lower bound on time complexity.