# Growth Rates

Last modified: 2026-11-18 23:25:03 (PST)

*Remark.* The running time of an algorithm is a function of the size $n$ of its input. To compare running times, we ignore fixed factors and small inputs. This page defines the notation for that comparison.

## Upper bounds

**Definition 1** (Big-O)**.** Let $f$ and $g$ be functions from the natural numbers to the real numbers. Then $f(n)$ is **big-O** of $g(n)$ (Wikipedia contributors 2026), written $f(n) = O(g(n))$, when there exist a real number $c > 0$ and a natural number $N$ such that, for every natural number $n \geq N$,

$$|f(n)| \leq c\,|g(n)|.$$

In words, from some point $N$ on, the absolute value of $f(n)$ is at most a fixed multiple $c$ of the absolute value of $g(n)$. The "there exist" is an existential quantifier, and the "for every" is a universal quantifier.

**Example 1** (The function $3n + 5$ is big-O of $n$)**.** Let $f(n) = 3n + 5$ and $g(n) = n$. To show $f(n) = O(g(n))$ by Definition 1, choose $c = 4$ and $N = 5$. For every natural number $n \geq 5$,

$$\begin{aligned} |f(n)| &= 3n + 5 && (3n + 5 > 0) \\ &\leq 3n + n && (5 \leq n, \text{ since } n \geq 5) \\ &= 4n && (3n + n = 4n) \\ &= 4\,|g(n)|, && (n = |n|, \text{ since } n \geq 0) \end{aligned}$$

which is the inequality in Definition 1 with $c = 4$. Hence $3n + 5 = O(n)$.

## Lower bounds

**Definition 2** (Big-Omega)**.** Let $f$ and $g$ be functions from the natural numbers to the real numbers. Then $f(n)$ is **big-Omega** of $g(n)$ (Wikipedia contributors 2026), written $f(n) = \Omega(g(n))$, when there exist a real number $c > 0$ and a natural number $N$ such that, for every natural number $n \geq N$,

$$|f(n)| \geq c\,|g(n)|.$$

In words, from some point $N$ on, the absolute value of $f(n)$ is at least a fixed positive multiple $c$ of the absolute value of $g(n)$. Dividing by $c$ shows that $f(n) = \Omega(g(n))$ exactly when $g(n) = O(f(n))$ in the sense of Definition 1.

**Example 2** (The function $3n + 5$ is big-Omega of $n$)**.** Let $f(n) = 3n + 5$ and $g(n) = n$. To show $f(n) = \Omega(g(n))$ by Definition 2, choose $c = 3$ and $N = 1$. For every natural number $n \geq 1$,

$$\begin{aligned} |f(n)| &= 3n + 5 && (3n + 5 > 0) \\ &\geq 3n && (5 \geq 0) \\ &= 3\,|g(n)|, && (n = |n|, \text{ since } n \geq 0) \end{aligned}$$

which is the inequality in Definition 2 with $c = 3$. Hence $3n + 5 = \Omega(n)$.

## Matching bounds

**Definition 3** (Big-Theta)**.** Let $f$ and $g$ be functions from the natural numbers to the real numbers. Then $f(n)$ is **big-Theta** of $g(n)$ (Wikipedia contributors 2026), written $f(n) = \Theta(g(n))$, when both $f(n) = O(g(n))$ (big-O) and $f(n) = \Omega(g(n))$ (big-Omega). In words, $f(n)$ and $g(n)$ are within fixed multiples of each other, from some point on.

**Example 3** (The function $3n + 5$ is big-Theta of $n$)**.** Let $f(n) = 3n + 5$ and $g(n) = n$. By Example 1, $f(n) = O(n)$, and by Example 2, $f(n) = \Omega(n)$. Therefore $f(n) = \Theta(n)$ by Definition 3.

## References

Wikipedia contributors. 2026. *Big O Notation — Wikipedia, the Free Encyclopedia.* https://en.wikipedia.org/wiki/Big_O_notation.