Big Understanding of Little-o

Harshita Bhagat, Vaishali More,
TY (IT-2024),
Dr. Priyadarshan Dhabe,
Ph.D (IIT Bombay), Professor in IT, VIT, Pune

Little-o- A loose upper bound of a function

Big O represents all the upper bounds of a function f(n). It can be tight or loose bound.

Little 0 represents loose bounds (only) of a function f(n)

There can be multiple loose upper bounds of f(n)

If f(n)=9n then f(n)=o(n^2), f(n)=$o(n^3)$, ...

i.e f(n)=9n has loose upper bounds as n^2, n^3,.., But n can not be loose upper bound of f(n), in fact it is tight bound of f(n)

To prove that f(n)=o(g(n), we have to prove that

$$f(n) < c * g(n) \; \forall \; c > 0 \;\; \& \;\; n > \; n_0$$

That above in-equility hold for each value of c>0 and for some n_0

Q1. Prove that $f(n) = 9n$ then $f(n) = o(n^3)$

Method 1- prove that it works for some values of c=1,2,3

Definition: $f(n) = o(g(n))$ iff for $c > 0$ & $n_0 > 0$,

$$f(n) < c * g(n) \; \forall \; c > 0 \; \& \; n > n_0$$

In this case, $f(n) = 9n$

$$9n < c * g(n)$$ ...from definition of $o(n)$

$$9n < c * n^3$$

c	n	$\mathbf{9n < c * n3}$	$isValid?$
$c = 1$	$n = 1$	$9 * 1 < 1 * 1$	no
	$n = 2$	$9 * 2 < 1 * 8$	no
	$n = 3$	$9 * 3 < 1 * 27$	no
	$n = 4$	$9 * 4 < 1 * 64$	yes
	$n = 10$	$9 * 10 < 1 * 1000$	yes
	$n = 100$	$9 * 100 < 1 * 1000000$	yes
$c = 2$	$n = 1$	$9 * 1 < 2 * 1$	no
	$n = 2$	$9 * 2 < 2 * 8$	no
	$n = 3$	$9 * 3 < 2 * 27$	yes
	$n = 4$	$9 * 4 < 1 * 64$	yes
	$n = 10$	$9 * 10 < 1 * 1000$	yes
$c = 3$	$n = 1$	$9 * 1 < 3 * 1$	no
	$n = 2$	$9 * 2 < 3 * 8$	yes
	$n = 3$	$9 * 3 < 3 * 27$	yes
$c = 10$	$n = 1$	$9 * 1 < 10 * 1$	yes
	$n = 2$	$9 * 2 < 10 * 8$	yes

From above table, the inequality or little o

$9n < c * n^3$ *holds true for* *(1)*

$$c = 1 \quad n_0 = 4;$$
$$c = 2 \quad n_0 = 3;$$
$$c = 3 \quad n_0 = 2;$$
$$c = 10 \quad n_0 = 1 .$$

As *<u>for each value of c</u>* *there is a* $n > n_0$ *satisfying inequality in (1)*

Hence we can conclude that

$\boldsymbol{f}(\boldsymbol{n}) = \boldsymbol{o}(n^3)$ $for\ \forall c > 0\ \&\ n > n_0 \quad where\ n_0 = 4\ for\ c = 1.$

Q2. Prove that $f(n) = 2n + 3$ then $f(n) \neq o(n)$

Method 1

Definition: $f(n) = o(g(n))$ iff for $c > 0$ & $n_0 > 0$,

$$f(n) < c * g(n) \; \forall \, c > 0 \;\&\; n > n_0$$

In this case, $f(n) = 2n + 3$

$$2n + 3 < c * g(n)$$ …from definition of $o(n)$

$$2n + 3 < c * n$$

c	n	$\mathbf{2n + 3 < c * n}$	$isValid?$
$c = 1$	$n = 1$	$2 * 1 + 3 < 1 * 1$	no
	$n = 2$	$2 * 2 + 3 < 1 * 2$	no
	$n = 3$	$2 * 3 + 3 < 1 * 3$	no
	$n = 4$	$2 * 4 + 3 < 1 * 4$	no
	$n = 10$	$2 * 10 + 3 < 1 * 10$	no
	$n = 100$	$2 * 100 + 3 < 1 * 100$	no

From above table, for $c = 1$ no value of n satisfies the condition for little-o $f(n) < c * g(n)\ for\ \forall c > 0$ and $n > n_0$.

$Hence\ we\ can\ conclude\ that\ \boldsymbol{f(n) \neq o(n)}$.

lethod 2: Limit Theory

If $\lim_{n\to\infty}\left(\frac{f(n)}{g(n)}\right)$ then $f(n) = o(g(n))$

$$\lim_{n\to\infty}\left(\frac{2n+3}{n}\right) = \lim_{n\to\infty}\left(\frac{2*1+0}{1}\right) = 2$$

[by using L'Hospital's Rule, taking derivatives of f(n) and g(n)]

$$\lim_{n\to\infty}\left(\frac{2n+3}{n}\right) \neq 0$$

Hence, $\boldsymbol{f(n) \neq o(n)}$

Thank You