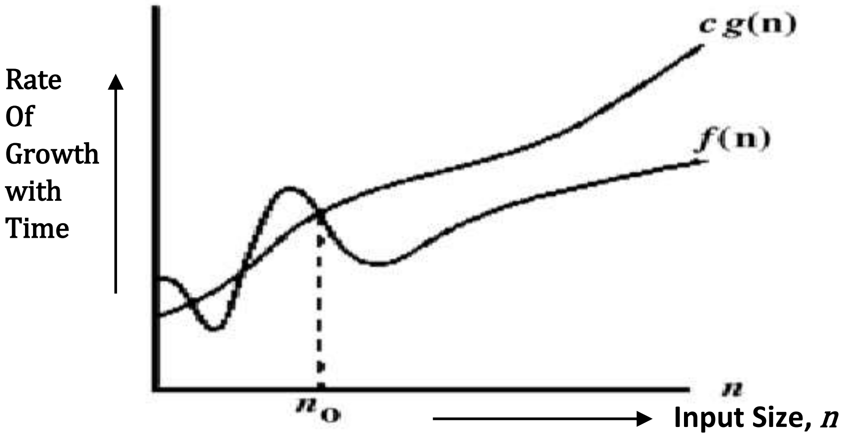
9.A. BIG -O NOTATION



DEFINITION: A function $\boldsymbol{f(n)}$ is said to be in $\boldsymbol{O(g(n))}$, denoted $\boldsymbol{f(n) \in O(g(n))}$, if $\boldsymbol{f(n)}$ *is bounded above by some constant multiple of* $\boldsymbol{g(n)}$ *for all large* $\boldsymbol{n}$, *i.e. if there exist some positive constant* $\boldsymbol{c}$ *and some nonnegative integer* $\boldsymbol{n_0}$ *such that*:

$$\boldsymbol{f(n) \leq cg(n) \text{ for all } n \geq n_0}$$

NOTE: THE BIG OH NOTATION FINDS OUT THE UPPER BOUND OF A POLYNOMIAL. AND $f(n)$ AND $g(n)$ ARE THEREFORE TWO POLYNOMIAL FUNCTIONS.

MORE DESCRIPTIVE WAY OF ABOVE DEFINITION:

*Let **f** and **g** be two functions that map a set of natural numbers to a set of positive real numbers , that is ,* $f: \mathbb{N} \rightarrow \mathbb{R}_{\geq 0}$.

*(It mean function **f** contains Set of Natural Numbers* $\mathbb{N}$ *which maps to real numbers* $\mathbb{R}$ *which are greater than and equal to* 0 *i.e.* ***Positive Real Numbers****)*.

THE THINGS MAY LOOK LIKE:

SAY, $f(n) = n^3 + n^2 + 3$ [POLYNOMIAL FUNCTION]

$if\ n = 0, f = 0^3 + 0^2 + 3 = 3$

$if\ n = 1, f = 1^3 + 1^2 + 3 = 2 + 3 = 5$

$if\ n = 2, f = 2^3 + 2^2 + 3 = 8 + 4 + 3 = 15$

THEN WE HAVE SET:

$\{(0,3), (1,5), (2,15)\}$

HENCE:

$Domain\ of\ Function\ (D_f)\ = \{0,1,2\}$

$Range\ of\ Function\ \left(R_f\right) = \{3,5,15\}$

Now, Let $O(g)$be the set of all functions with similar rate of growth.

i.e. if $f(n) = n^3 + n^2 + 3$, and $S(f(n)) = \{(0,3), (1,5), (2,15)\}$.

Then $O(g)$will have same type of set of pairs may not be equal .

Then , the relation $\boldsymbol{f(n) = O(g(n))}$holds true , if there exist two positive constants $\boldsymbol{c}$ and $\boldsymbol{n_0}$ such that $\boldsymbol{f(n) \le c \times g(n)}$.

- ***The function $f(n)$ is said to be in $O(g(n))$.***
- ***This is denoted as $f(n) \in O(g(n))$.***

i.e. $f(n)$ belongs to $O(g(n))$.

i.e. the elements in $f(n)$ contains in $O(g(n))$.

or simply as: $f(n) = O(g(n))$

- ***This implies that $f(n)$ never takes more than approximately $g(n)$ operations.***
- ***This implies $f(n)$is in the order of $g(n)$.***
- ***Which implies $f(n)$ having a growth rate that is less than or equal to that of $g(n)$.***
- ***This implies again that function f grows at a slower rate than a constant time $g(n)$ for all the values of a larger input of size $'n'$.***

MORE PRECISELY

We can distinguish $f\ (n)\ and\ g(n)\ like\ say$:

$f(n) = 4n^2 + 5n + 3\ and\ g(n) = 5n^2$, on a note we can see the rate of growth:

n	$4n^2 + 5n + 3$	$5n^2$
1	12	5
2	29	20
3	54	45
4	87	80
5	128	125
6	177	180

Comparison of $f(n)\ and\ g(n)$

Some Points on Big-Oh Notation:

1. $f(n) = O\big(g(n)\big)\ means\ at\ larger\ values\ of$ n , the $the\ upper\ bound\ of\ f(n)\ is\ g(n)$. $For\ example$, if $f(n) = n^4 + 100n^2 + 10n + 50\ is\ the\ given\ algorithm,$ $then\ n^4\ is\ g(n)$.

2. $g(n)\ gives\ the\ maximum\ rate\ of\ growth\ for\ f(n)\ at$ $at\ larger\ values\ of\ n$.

3. $O(g(n))$
   $= \{f :$ ***there exist positive constant*** c ***and*** n_0 ***such that*** $0 \leq f(n) \leq cg(n)$ ***for all*** $n \geq n_0\}$.

4. $g(n)$ ***is an asymptotic tight upper bound for*** $f(n)$.

5. n_0 ***is the point from which we need to consider the rate of growth for a given algorithm ,***
   hence n_0 ***is called threshold of the given function***.

6. Generally, we discard lower values of n . That means the rate of growth at lower values of n is not important.