Note Title

Principles of Algorithms - Part 2.

Time Complexity.

(Specification / Description of Algorithm)

operation count.

positive integer!

Ex: Sorting, 'comparison'.

Running Time

execution of the code that is using the algorithm.

Worst case complexity: (WCC)

input I, $IC(I)$

$$WCC(n) = \max_{I \in I_n} (IC(I))$$

I_n = { set of all inputs of Size n }.

Complexity function.

(WCC function)

WCC (n)
associates an input size to the Max Instruction count.

f (n)

Efficiency of an algorithm is judged by its complexity function.

A — $7n^2 + 9n + 10$ — f (n)

B — $10n + 23$ — g (n)

How do we compare f (n), g (n) values ~~for~~ for large values of n.

f (n), $n \to \infty$

g (n), $n \to \infty$.

~~Asyp~~ Asymptotics of complexity functions are helpful in determining growth rate of complexity functions.

n	n^2	n^3
10	100	1000
100	10^4	10^6
1000	10^6	10^9.

	A	B
	$10000\, n^2$,	$\frac{1}{100}\, n^3$
$n = 10$	10^6	10
$n = 1000$	10^{10}	10^7.
$n = 10^6$	10^{16}	10^{16}
$n = 10^8$	10^{20}	10^{22}

$f(n)$ is $O(g(n))$

if $f(n) \leq c\, g(n)$

for some $c > 0$, $\forall\, n \geq n_0$.

$O \rightarrow$ upper bound.

$7n^2 + 9n + 10$ is $O(n^2)$.

$7n^2 + 9n + 10 \leq 7n^2 + 9n^2 + 10n^2$

$= 26 n^2$.

$(7n^2 + 9n + 10) \leq 26 n^2, \quad \forall\, n \geq 1,$

$7n^2 + 9n + 10$ is $O(n^2)$

$7n^2 + 9n + 10$ is $O(n^4)$

$7n^2 + 9n + 10 \le 7n^4 + 9n^4 + 10n^4$

$7n^2 + 9n + 10 \le 26n^4$; $n \ge 1$

Θ notation

$f(n)$ is $\Theta(g(n))$ if

$\rightarrow$ $c_1 g(n) \le f(n) \le c_2 g(n)$.

for $c_1, c_2 > 0$. for $\forall n \ge n_0$

$3n^2 \le 7n^2 + 9n + 10 \le 26n^2$

$7n^2 + 9n + 10$ is $\Theta(n^2)$

$f(n)$ is $\Theta(g(n))$

the growth rates of g & f are same.

$\Theta(\log n)$

$\Theta(n)$

$\Theta(n^2)$

$\vdots$

$\Theta(2^n)$

$\Theta(n \log n)$ is better than $\Theta(n^2)$ algorithm

$$\text{if } \lim_{n \to \infty} \frac{f(n)}{g(n)} = c, \quad c > 0$$

Then $f(n)$ is $\theta(g(n))$.

~~if~~ $\cancel{\lim_{n \to \infty} \frac{g(n)}{f(n)}}$ ~~is~~ $\cancel{c \geq 0,}$ $\cancel{(\infty)}$.

$$\text{if } \lim_{n \to \infty} \frac{f(n)}{g(n)} = c, \quad c \geq 0$$

$f(n)$ is $O(g(n))$.