TABLE 2.1 Values (some approximate) of several functions important for analysis of algorithms

n	$\log_2 n$	n	$n \log_2 n$	n^2	n^3	2^n	$n!$
10	3.3	10^1	$3.3 \cdot 10^1$	10^2	10^3	10^3	$3.6 \cdot 10^6$
10^2	6.6	10^2	$6.6 \cdot 10^2$	10^4	10^6	$1.3 \cdot 10^{30}$	$9.3 \cdot 10^{157}$
10^3	10	10^3	$1.0 \cdot 10^4$	10^6	10^9		
10^4	13	10^4	$1.3 \cdot 10^5$	10^8	10^{12}		
10^5	17	10^5	$1.7 \cdot 10^6$	10^{10}	10^{15}		
10^6	20	10^6	$2.0 \cdot 10^7$	10^{12}	10^{18}		

Input size $\rightarrow 2n$

$$\log_2 n \longrightarrow \log_2 2n = \log_2 2 + \log_2 n$$

$$= 1 + \log_2 n \Rightarrow \text{increases by } 1$$

$$n \longrightarrow 2n \Rightarrow \text{doubled}$$

$$n \log n \rightarrow 2n \log 2n = 2n(\log_2 2 + \log_2 n)$$

$$= 2n(1 + \log_2 n) \Rightarrow \text{more than two-fold increase}$$

$$n^2 \longrightarrow (2n)^2 = 4n^2 \Rightarrow \text{four-fold increase}$$

$$n^3 \rightarrow (2n)^3 = 8n^3 \Rightarrow \text{eight-fold increase}$$

$$\left.\begin{array}{l} 2^n \\ n! \end{array}\right\} \text{Exponential Growth functions.}$$

$\rightarrow$ A computer that solves 10^{12} operations per second, to execute 2^{100} operations $\Rightarrow 4 \times 10^{10}$ years

$100! \longrightarrow$ takes much longer

ALGORITHM Sequential Search $(A[0 \ldots n-1], K)$

// searches for the search key, K in the array A by sequential search

// Input : Array $A[0 \ldots n-1]$ and a search key 'K'

```
i ← 0
while i < n and A[i] ≠ K do
        i ← i+1
if i < n return i
else return -1
```

Input size → n

Basic Operation → key comparison

Worst-Case ⇒ $C_{worst}(n) = n$

Best-Case ⇒ $C_{best}(n) = 1$

Average Case Efficiency

For a typical or random input,

$p \rightarrow$ probability of ~~finding the~~ .
a successful search

$1-p \rightarrow$ probability of an unsuccessful search.

For a successful search, the probability of the first match occuring at position 'i' is

$$\frac{p}{n} \text{ for every } i$$

$$C_{avg}(n) = \left[1 \cdot \frac{p}{n} + 2 \cdot \frac{p}{n} + 3 \cdot \frac{p}{n} + \cdots \cdots + i \cdot \frac{p}{n} + \cdots \right] + n(1-p)$$

$$= \frac{p}{n}\left[1+2+3+\cdots \cdots + n\right] + n(1-p)$$

$$= \frac{p}{\cancel{n}} \frac{\cancel{n}(n+1)}{2} + n(1-p)$$

$$\boxed{C_{avg}(n) = \frac{p(n+1)}{2} + n(1-p)}$$

For a successful search, $p = 1$

$$C_{avg}(n) = \frac{n+1}{2}$$

For an unsuccessful search, $p = 0$

$$C_{avg}(n) = n$$

ASYMPTOTIC NOTATIONS & BASIC EFFICIENCY CLASSES

① Big-oh Notation (or) O-notation :-

A function $t(n)$ is said to be in $O(g(n))$, if $t(n)$ is bounded above by some constant multiple of $g(n)$ for all large n i.e., there exists some positive constant 'c' and some non-negative integer, n_0 such that

$$t(n) \leq c\,g(n) \quad \text{for all } n \geq n_0$$

$$\Rightarrow \boxed{t(n) \in O(g(n))}$$

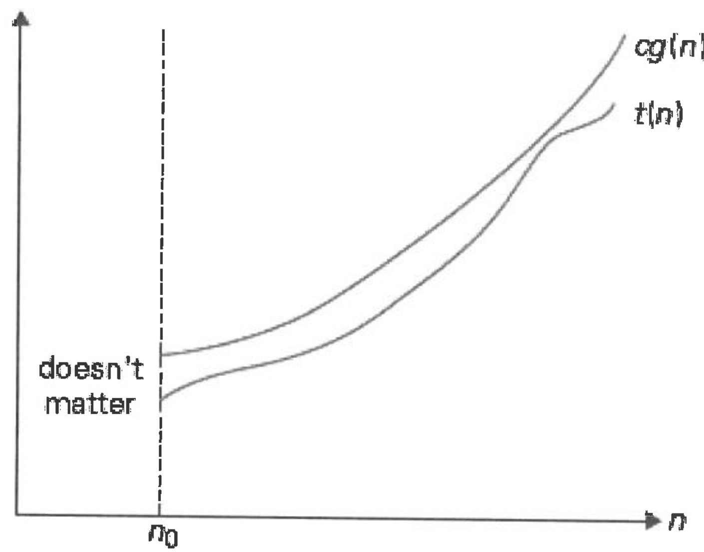

FIGURE 2.1 Big-oh notation: $t(n) \in O(g(n))$.

eg1 $t(n) = 100n + 5$

$g(n) = n^2$

Option-1:

$100n + 5 \leq 100n + n$ for all $n \geq 5$

$101n$

$101n \leq 101n^2$

$t(n) \leq c\,g(n)$ where $c = 101$

$n \geq 5$

$\Rightarrow \boxed{t(n) \in O(g(n))}$ where $c = 101$

$n_0 = 5$

Hence Proved by definition

~~Method-2~~

Option-2

$t(n) = 100n + 5$

$g(n) = n^2$

$100n + 5$

$100n + 5n$ for all $n \geq 1$

$105n$

$105n \leq 105n^2$

$\Rightarrow t(n) \leq c\,g(n)$

$\Rightarrow \boxed{t(n) \in O(g(n))}$ where $c = 105$, $n_0 = 1$

Hence Proved by definition

⟹ Flexibility to choose 'c' and 'n_0'

$$\boxed{\begin{array}{ccc} \text{Order of growth of } t(n) & \leq & \text{Order of growth of } g(n) \\ & \Rightarrow t(n) \in O(g(n)) & \end{array}}$$

② Big-Omega (or) Ω - notation

A function $t(n)$ is said to be in $\Omega(g(n))$, if $t(n)$ is bounded below by some constant multiple of $g(n)$ for all large 'n' i.e.; there exists some positive constant 'c' and some non-negative integer 'n_0' such that

$$t(n) \geq c\,g(n) \quad \text{for all } n \geq n_0$$

$$\Rightarrow \boxed{t(n) \in \Omega(g(n))}$$

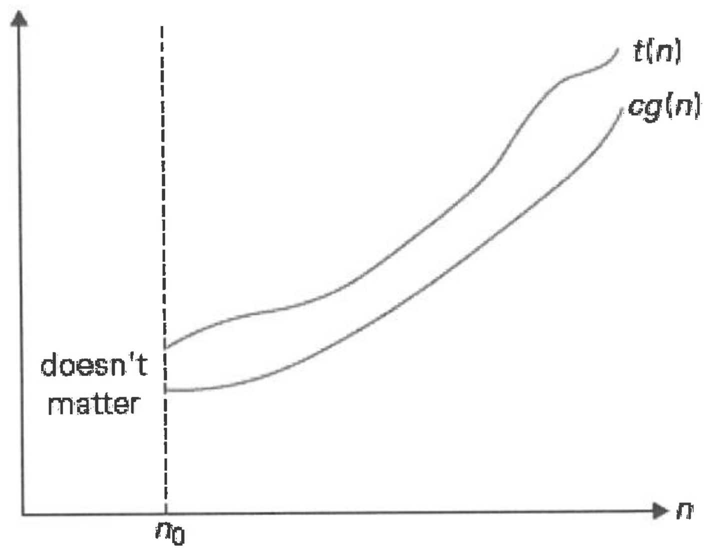

FIGURE 2.2 Big-omega notation: $t(n) \in \Omega(g(n))$.

(g) $t(n) = n^3$

$g(n) = n^2$

$$n^3 \geq n^2 \quad \text{for } n \geq 0$$

$$\Rightarrow \boxed{t(n) \in \Omega(g(n))} \quad \text{where } c = 1, \; n_0 = 0$$

$$n^3 \in \Omega(n^2)$$

Hence Proved by definition