ODA →

Siddhi Shankar

Assignment 20021517 CST-SPL2

1 Asymptotic notations are the mathematical notations used to describe the running time of an algorithm when input tend towards a particular value or a limiting value

Ex: bubble sort, when input are already sorted time taken by algo is Linear

Also tells complexity of algo when input is large

Mainly 3 types
→ Big - O Notation
→ Omega Notation
→ Theta Notation

For example
→ Big O Notations

$$f(n) = O(g(n))$$

iff

$$f(n) \leq c \cdot g(n)$$

2 for (i=1 to n) {i=i*2;}

Sol^n G $\sum_{i=1}^{n}$ 1+2+4 . . . n

$$n = 2^k$$

$$2n = 2^k$$

$$\log 2n = k \log 2$$

$$\log 2 + \log n = k \log 2$$

$$\log n + 1 = k$$

$$O(k) = O(1 + \log n)$$

$$= O(\log n)$$

3. $T(n) = \{3T(n-1)$ if $n>0$ otherwise 1

$$
\begin{aligned}
T(n) &= 3T(n-1) \\
&= 3(3T(n-2)) \\
&= 3^2 T(n-2) \\
&= 3^3 T(n-3) \\
&= \cdots \\
&= 3^n T(n-n) \\
&= 3^n T(0) \\
&= 3^n
\end{aligned}
$$

$T.C = O(3^n)$

4. $T(n) = \{2T(n-1)-1$ if $n>0$, otherwise $1\}$

$$
\begin{aligned}
T(n) &= 2T(n-1)-1 \\
&= 2(2T(n-2)-1)-1 \\
&= 2^2(T(n-2)) - 1 - 2 - 1 \\
&= 2^n T(n-n) - 2^{n-1} - 2^{n-2} - 2^{n-3} \\
&\quad - 2^2 - 2^1 - 2^0 \\
&= 2^n - 2^{n-1} - 2^{n-2} \ldots - 2^2 - 2^1 - 2^0 \\
&= 2^n - (2^n - 1)
\end{aligned}
$$

$T(n) = 1$

$T.C = O(1)$

5) int i=1 s=1;
while (s<=n)
{ i++, s=s+i;
Printf('#');
}

Soln $s_i = s_{i-1} + i$

→ At each Repetation value of i increases by 1 Therefor
→ i^{th} term of s contains sum of first 'i' Positive integers
→ K total no of times while loops Repeat

$$1+2+3 \cdots + k = [K(K+1)/2] > n$$

$$k = O(\sqrt{n})$$

$$T.C = O(\sqrt{n})$$

(6) Void func (int n)
{ int i, count = 0;
for (i=1; i*i<=n; i++)
count++
}

Soln

$i^2 <= n$ $i = 1, 2, 3 \cdots \sqrt{n}$

$i <= \sqrt{n}$ $\sum_{i=1}^{n} 1+2+3 \cdots + \sqrt{n}$

$$T(n) = \frac{\sqrt{n} \times (\sqrt{n}+1)}{2}$$

$$T(n) = \frac{n \times \sqrt{n}}{2}$$ $\boxed{T(n) = O(n)}$

PAGE No.
DATE

20021517 CST-3P2

7 void fun^c (int n) {
int j, i, k count=0;
for (i=n/2; i<=n; i++)
for (j=1; j<=n; j=j*2)
for (k=1; k<=n; k=k*2)
count++ }

Sol^n for $k = k^2$

$k = 1, 2, 4, 8 \cdots n$

GP → $a=1, r=2$

$$\frac{a(r^n-1)}{r-1}$$

$$= \frac{1(2^k-1)}{1}$$

$\log n = k$

$i = 1\ 2 \cdots n$; j – $\log n, \log n \cdots \log n$

$k = \log n * \log n \cdots \log n * \log n$

$\Rightarrow O(n * \log n * \log n)$

$= O(n \log^2 n)$

(8) function (int n)

{int (n==1)

return; // $O(1)$

for (i=1 to n) // $i=1, 2, 3 \cdots n \Rightarrow O(n)$

{ for (j=1 to n) // $j = 1, 2, 3, 4 \cdots n^2 \Rightarrow O(n^2)$

{ print ('*') }}

function(n-3); } $T(n/3)$

$\Rightarrow T(n) = T(n/3) + n^2$

$\Rightarrow a=1, b=3 \quad f(n) = n^2$

$c = \log_3 1 = 0$

$= n^0 = 1 > (f(n) = n^2)$

$T(n) = \Theta(n^2)$

(9) Void function (int n)
{ for (i=1 to n)
{ // O(n)
for (j=1; j<=n; j=j+i)
Print ("*") // O(n)
}}

Sol^n for i=1 => j = 1,2,3 ... n = n
for i=2 => j = 1,3,5 ... n = n/2
for i=3 => j = 1,4,7 ... n = n/3

for i=n => j=1 ...

$$\Rightarrow \sum_{j=n}^{1} = n + \frac{n}{2} + \frac{n}{3} + \frac{n}{4} \cdots + 1$$

$$\sum_{j=n}^{1} n\left[1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \cdots + \frac{1}{n}\right]$$

$$\sum_{j=n} n[\log n]$$

$$T(n) = [n \log n]$$
$$T(n) = O(n \log n)$$

(10) n^k and c^n what is the asymptomic relationship between these functions?
Assume k>=1 and c>1 are constant. value c and n0 for which Relation holds.

as given n^k and c^n
Relation b/w n^k and c^n is
$$n^k = O(c^n)$$
as $n^k \le c^n$
$\forall n \ge n_0$ Some constant $a > 0$

for ~~n~~ $n_0 = 1$

$c = 2$

$\Rightarrow 1^k \le a^1$

$\boxed{n_0 = 1 \ \& \ c = 2}$

x —— x —— x —— x —— x