DATE / /
PAGE NO.

Tutorial-01

Q1 Asymptotic notations are mathematical tools to represent the time complexities of algorithm.

following are the notations that are used.

O — It defines an upper bound ~~of~~ ~~of~~ of an algo. it bounds a function only from above

Ω — Big omega notation provides the lower bound of function.

Θ - It represent both upper and lower bound function.

eg. for insertion sort
$O(n^2)$
$\Theta(n^2)$
$\Omega(n)$
linear time is best case and quadratic is worst case.

Q2 $o(\log n)$

Q3 as $T(n) = \begin{cases} 3T(n-1) & \text{if } n>0 \\ 1 & \text{otherwise.} \end{cases}$

$T(n) = 3T(n-1)$
$= 3(3T(n-2))$
$= 3^2(T(n-2))$
$= 3^3 T(n-3)$
$= 3^n T(n-n) = 3^n$

Ans4 here $T(n) = \begin{cases} 2T(n-1)-1, & \text{if } n \to 0 \\ 1, & \text{otherwise.} \end{cases}$

$T(n) = 2T(n-1)-1$

$= 2T(2T(n-2)-1)-1$

$= 2^2(T(n-2))-2-1$

$= 2^n - (2^n-1)$

$2^n - 2^n + 1 = 1 \Rightarrow T(n) = 1.$

Ans5 here, $s = s+i$, if K is total no. of iteration taken by the program then while loop terminates.

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

$K = O(\sqrt{n}).$

Ans6 $O(\sqrt{n})$

Ans8 Time Complexity : $O(n \log n)$.

Ans8 $O(n^2)$

Ans9 $O(n \log n)$.

Ans10 here n^K, a^n

$K = 1, \quad a > 1$

taking $K = a = 2$.

here $n^K = n^2$

and $a^n = 2^n$

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

$\therefore n^K = O(a^n).$