Tutorial1 ⇒ Ayank Gupta

Section B

Ans1 ⇒ Asymptotic notations are mathematical tools to represent the time complexities of algorithms.

⇒ Following are the notations that are used ⇒

① O ⇒ It defines an upper bound as an algo. it bounds a function only from above.

② Ω notation ⇒ Big omega notation provides the lower bound of function

③ θ notation ⇒ It represents both upper and lower bounds function.

⊙ Eg. ⇒ lets take eg. of Insertion Sort ⇒

⇒ It takes linear time in best case and quadratic in worst case ⇒

$O(n^2)$

$\theta(n^2)$

$\Omega(n)$

Ans2 ⇒ Complexity ⇒ $O(\log n)$

$\boxed{\text{Ans3}}$ $\Rightarrow$ here $T(n) = \begin{cases} 3T(n-1) & \text{if } n>0 \\ 1 & \text{otherwise} \end{cases}$

$$T(n) = 3T(n-1)$$
$$= 3(3T(n-2))$$
$$= 3^2\,T(n-2)$$
$$= 3^3\,T(n-3)$$
$$\vdots \qquad \vdots$$
$$3^n\,T(n-n) = 3^n$$

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

$$T(n) = 2T(n-1)-1$$
$$= 2T(2T(n-2)-1)-1$$
$$= 2^2(T(n-2)) - 2 - 1$$
$$= 2^2(2T(n-3)-1) - 2 - 1$$
$$= 2^3\,T(n-3) - 2^2 - 2^1 - 2^0$$
$$= 2^n - (2^n - 1)$$
$$\Rightarrow 2^n - 2^n + 1 = 1 \Rightarrow T(n) = 1$$

$\boxed{\text{Ans5}}$ $\Rightarrow$ here $S = S + i$

if k is total no. of iteration taken by the program, then while loop terminates.

$1+2+3+\ldots k \Rightarrow k[(k+1)/2] > n$

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

$\boxed{\text{Ans6}} \Rightarrow O(\sqrt{n})$

$\boxed{\text{Ans7}} \Rightarrow$ Time complexity $\Rightarrow O(n\log n)$

$\boxed{\text{Ans 8}} \Rightarrow O(n^2)$

$\boxed{\text{Ans9}} \Rightarrow O(n\log n)$

$\boxed{\text{Ans10}} \Rightarrow$ Here n^k, a^n

$k>=1, a>1$

taking $k=a=2$

$n^2 \quad 2^n$

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

$\therefore n^k = O(a^n)$