TUTORIAL SHEET → 1

NAME → Jyoti Pandey
COURSE → B.Tech
SECTION → A
Roll.NO → 1961071

Ans → 1 → (3) O(N+N) time
O(1) space.

Ans → 2 → T(n) = O(n) , space O(1).

Ans → 3 → T(n) = O($\log_2 n$) , space O(1).

Ans → 4 →
```
int sum = 0, i;
for(i=0; i*i<n; i++)
{
  sum += i;
}
```

$= n + (n-1) + (n-2) + (n-3) + \cdots (n-k)$

$= n + (n*k) - (1^2 + 2^2 + 3^2 + \cdots k^2)$

$= \sqrt{n}$

$i^2 < n$

$i < \sqrt{n}$

T(n) = O($\sqrt{n}$) , space O(1)

Ans → 5 →
```
int j=1, i=0
while(i<=n)
{
  i = i+j;
  j++;
}
```

0 <= n · 1

1 <= n 1

3 <= n

(0, 1, 3, 6, 10, 15, 21, ---- n)

k terms.

K^{th} term $= \frac{(K*(K+1))}{2}$

$n = \frac{K^2+K}{2}$

$K^2 + K = 2n$

$K^2 + K - 2n = 0$

$K = \frac{-1 \pm \sqrt{1+8n}}{2}$

$K = \frac{\sqrt{8n+1}}{2}$

$K = \frac{\sqrt{8n}}{2} = \sqrt{n}$

$T(n) = \sqrt{n}$ space - O(1)

Ans-)6-) void Recursion(int n) — T(n)

```
{
  if(n==1) return;
  recursion(n-1)      ⟶ T(n-1)
  point(n);
  recursion(n-1);     ⟶ T(n-1)
}
```

$$T(n) = \begin{cases} 1 & n=1 \\ 2T(n-1)+1 & n>1. \end{cases}$$

$T(n) = 2T(n-1) + 1$ — ①

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

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

$T(n) = 4T(n-2) + (1+2)$ — ②

$T(n-2) = 2(T(n-3)) + 1.$

$T(n) = 4(2T(n-3) + 1) + (1+2)$

$T(n) = 8T(n-3) + (1+2+4)$ — ③

$T(n) = 8[2(T(n-4)+1) + (1+2+4)$

$T(n) = 16T(n-4) + (1+2+4+8)$ — (4)

$T(n) = 2^k T(n-k) + (1+2+4+8+\cdots)$
k times.

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

$k = n-1$

$T(n) = 2^{n-1} T(1) + (1+2+4+8+\cdots)$
$(n-1)$ times.

$T(n) = \frac{2^n}{2} + (1+2+4+8\cdots)$
$(n-1)$ times.

$S_n = \frac{a(r^n-1)}{r-1}$ $\quad a=1, r=2, n=n-1$

$T(n) = \frac{2^n}{2} + \left(\frac{2^{n-1}-1}{1}\right)$

$T(n) = \frac{2^n}{2} + \frac{2^n}{2} - 1$

$T(n) = 2\left(\frac{2^n}{2}\right) - 1$

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

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

Ans.

Ans-7) It is a binary search algorithm.

$T(n) = \log_2(n)$

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

by using masters method (can't be solved)

$T(n) = aT(\frac{n}{b}) + f(n)$

so $a = 1$

$b = 2$

$f(n) = 1$

$c = \log_b a = \log_2(1) = 0$.

$\simeq 1$

$n = f(n) = 1.$

$\therefore n^c = f(n)$

$T(n) = O(\log_2 n)$ Ans.

Ans → (8) → (1) $T(1) = 1.$

$T(n) = T(n-1) + 1$ — (1)

$T(n) = T(n-2) + 2$ — (2)

$T(n) = T(n-3) + 3$ — (3)

$T(n) = T(n-k) + k$ — (4)

$n - k = 1$

$k = n-1$

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

$T(n) = n$

$T(n) = O(n)$ Ans.

(11) $T(n) = T(n-1) + n$ — (1)

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

$T(n) = T(n-2) + (n + (n-1))$ — (2)

$T(n) = T(n-3) + (n + (n-1) + (n-2))$ — (3)

$T(n) = T(n-k) + (n + (n-1) + (n-2) + \cdots (n-k))$

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

$n = k + 1$

$k = n - 1$

$T(n) = T(1) + (n + (n-1) + (n-2) \cdots$ — (0)

$T(n) = 1 + (n + (n-1) + (n-2) + \cdots 1)$

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

$$T(n) = \frac{n^2+2}{2}$$

$$T(n) = O(n^2)$$ Ans.

(iii) $T(n) = T(n/2) + 1$ — (1)

$T(n/2) = T(n/4) + 1$

$T(n) = T(n/4) + 2$ — (2)

$T(n/4) = T(n/8) + 1$

$T(n) = T(n/8) + 3$ — (3)

$T(n) = T(n/2^k) + k$ — (4).

$\frac{n}{2^k} = 1$

$2^k = n$

$k = \log_2(n)$

$T(n) = T(1) + \log_2 n$

$T(n) = O(\log_2 n)$ Ans.

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

$c = 1$

$n^c = n$

$f(n) = 1$

$n^c > f(n)$

$T(n) = O(n)$

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

$T(n) = O(2^n)$ Ans.

(VI) $T(n) = 3T(n-1)$, $T(0) = 1$.

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

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

$T(n) = 9T(n-2)$

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

$T(n) = 3^k T(n-k)$

For $n-k = 0$

$n = k$

$T(n) = 3^n T(0)$

$T(n) = 3^n$

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

(VII) $T(n) = T(\sqrt{n}) + 1$ —(1)

$T(\sqrt{n}) = T(n^{\frac{1}{4}}) + 1$

$T(n) = T(n^{\frac{1}{4}}) + 2$ —(2)

$T(n) = T(n^{1/8}) + 3$ —(3)

$T(n) = T(n^{(\frac{1}{2^k})}) + k$ }.

For $T(n^{\frac{1}{k}}) = T(2)$

$n^{\frac{1}{2^k}} = 2$

$n^{\frac{1}{2^k}} = 2$.

$\frac{1}{2^k} \log n = 1$.

$2^k = \log n$

$k = \log_2(\log(n))$

$T(n) = O(\log(\log(n))$ Ans.

(VIII) $T(n) = T(\sqrt{n}) + n$

$T(\sqrt{n}) = T(n^{\frac{1}{4}}) + \sqrt{n}$

$T(n) = T(n^{\frac{1}{4}}) + (n+\sqrt{n})$

$T(n) = T(n^{\frac{1}{8}}) + (n+\sqrt{n}+n^{\frac{1}{4}})$

$T(n) = T(n^{\frac{1}{2^k}}) + (n+n^{\frac{1}{2}}+n^{\frac{1}{4}}+\text{----} \ k \text{ terms})$

for $n^{\frac{1}{2^k}} = 2$

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

$2^k = \log(n)$

$k = \log(\log(n))$

$T(n) = 1 + (n+\sqrt{n}+\sqrt{n}\sqrt{n}+\text{---})$

$T(n) = 1 + ($ G.P $a = n$, $r = \sqrt{n}$, no of terms $= k$ $)$

$T(n) = 1 + \left(n \frac{(\sqrt{n})^k - 1}{k-1}\right)$

$T(n) = 1 + n\left[\frac{(\sqrt{n})^{\log(\log n)} - 1}{\log(\log n) - 1}\right]$

$T(n) = n.\log\log(n)$ & by neglecting other values.

$T(n) = O(n.\log(\log(n)))$

Ans → 9 →

```
int sum=0, i ;
for i=0 ; i<n ; i++)
 {
  sum += i ;
 }
```

0, 1, 2, ---- n

so $T(n) = O(n)$,, , space $O(1)$

Ans)10) $O(N * (N, N-1, \dots\dots 1))$

$O\left(N * \frac{(N+1)}{2}\right)$

(4) $O(N * N)$

Ans) 11) $O\left(\frac{n}{2} * (\log_2 N)\right)$

$= O(n \log n)$

Ans

Ans) 12) x will always be a better choice for large input.

Ans) 13) $O(\log N)$

Ans) 14) $T(n) = 7\left(T\left(\frac{n}{2}\right)\right) + (3n^2 + 2)$

$f(n) = 3n^2 + 2$

$a = 7$

$b = 2$

$c = \log_b a = \log_2(7) = 2.807$

$n^c = n^{2.8} \approx n^{2.8}$

$f(n) = 3n^2 + 2$

so

$n^c > f(n)$

so $T(n) = O(n^{2.8})$ or (c) $\Theta(n^{2.8})$

(a) $O(n^{28})$

(a) $\Theta(n^3)$

Ans) 15) $f_1(n) = n^{\sqrt{n}}$

$f_2(n) = 2^n$

$f_3(n) = (1.000001)^n$

$f_4(n) = n^{(10 * 2^{\frac{n}{2}})}$

a) $f_2(n) > f_4(n) > f_3(n) > f_1(n)$

Ans 16 → $f(n) = 2^{2n}$

$\log f(n) = 2n \log_2 2$

$\log f(n) = 2n$

$f(n) = 2^n \cdot 2^n$

$\Omega(2^n)$ Ans.

Ans 17 → $T(n) = 2T(\frac{n}{2}) + n^2$

$c = 1$

$n^c = n$

$n^2 > n$

$f(n) > n^c$

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

Ans 18 → $O(\log N)$ [Its a G.C.D function when n keeps on decreasing by n/2]

Ans 19 → $T(n) = O[N^2 + N]$

$T(n) = O(N^2)$ Ans.