

TUTORIAL SHEET → 1

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;  
}
```

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

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

$$= \sqrt{n}$$

$$i^2 < n$$

$$i < \sqrt{n}$$

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

Ans → 5 →

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

$$0 <= n \quad 1$$

$$1 <= n \quad 1$$

$$3 <= n$$

(0, 1, 3, 6, 10, 15, 21, ... n)
K terms.

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

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

$$K^2 + K = 2n$$

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

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

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

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

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

Ans-6 \Rightarrow void Recursion(int n) $\rightarrow T(n)$

{
if (n == 1) return;

recursion(n-1) $\rightarrow T(n-1)$

printf(n);

recursion(n-1); $\rightarrow T(n-1)$

}

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

$$T(n) = 2T(n-1) + 1 \quad \text{--- (1)}$$

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

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

$$T(n) = 4T(n-2) + (1+2) \quad \text{--- (2)}$$

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

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

$$T(n) = 8T(n-3) + (1+2+4) \quad \text{--- (3)}$$

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

$$T(n) = (6T(n-4) + (1+2+4+8) \quad \text{--- (4)}$$

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

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

$$k = n-1$$

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

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

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

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

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

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

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

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

Ans 7-7) It is a binary search algorithm. Ans.

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

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

by using masters method (can't be solved)

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

$$\text{so } a=1$$

$$b=2$$

$$f(n) = 1$$

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

$$\approx 1$$

$$n = f(n) = 1.$$

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

$$T(n) = O(\log_2 n) \quad \underline{\text{Ans.}}$$

$$\text{Ans} \rightarrow \textcircled{8} \rightarrow \textcircled{1} \quad T(1) = 1.$$

$$T(n) = T(n-1) + 1 \quad - \textcircled{1}$$

$$T(n) = T(n-2) + 2 \quad - \textcircled{2}$$

$$T(n) = T(n-3) + 3 \quad - \textcircled{3}$$

$$T(n) = T(n-k) + k \quad - \textcircled{4}$$

$$n-k = 1$$

$$k = n-1$$

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

$$T(n) = n$$

$$T(n) = O(n) \quad \underline{\text{Ans.}}$$

$$\textcircled{11} \quad T(n) = T(n-1) + n \quad - \textcircled{1}$$

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

$$T(n) = T(n-2) + (n + (n-1)) \quad - \textcircled{2}$$

$$T(n) = T(n-3) + (n + (n-1) + (n-2)) \quad - \textcircled{3}$$

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

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

$$n = k+1$$

$$k = n-1$$

$$T(n) = T(1) + (n + (n-1) + (n-2) + \dots + 1) \quad \textcircled{4}$$

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

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

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

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

Ans.

$$(ii) \quad T(n) = T(n/2) + 1 \quad - (1)$$

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

$$T(n) = T(n/4) + 2 \quad - (2)$$

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

$$T(n) = T(n/8) + 3 \quad - (3)$$

$$T(n) = T(n/2^k) + k \quad - (4)$$

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

$$2^k = n$$

$$k = \log_2(n)$$

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

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

Ans.

$$(iv) \quad T(n) = 2T(n/2) + 1$$

$$c=1$$

$$n^c = n$$

$$f(n) = 1$$

$$n^c \geq f(n)$$

$$T(n) = O(n)$$

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

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

Ans.

$$\textcircled{\text{vi}} \quad T(n) = 3T(n-1), \quad T(0) = 1.$$

$$T(n) = 3(T(n-1)) - \textcircled{1}$$

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

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

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

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

$$\text{For } n-k = 0$$

$$n = 1k$$

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

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

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

Ans.

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

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

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

$$T(n) = T(n^{\frac{1}{4}}) + 3 \quad -\textcircled{3}$$

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

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

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

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

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

$$2^k = \log n$$

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

$$T(n) = O(\log(\log n)) \quad \text{Ans.}$$

$$\textcircled{\text{viii}} \quad T(n) = T(\sqrt{n}) + n$$

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

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

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

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

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

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

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

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

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

$$T(n) = 1 + \left(\begin{array}{l} \text{G.P } a=n \\ r=\sqrt{n} \\ \text{no of terms } = k \end{array} \right)$$

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

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

$$T(n) = n \cdot \log \log(n) \quad \{ \text{by neglecting other values} \}$$

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

Ans 29) 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) $O(N * (N, N-1, \dots, 1))$

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

(4) $O(N * N)$

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

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

Ans 12) \times 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 $\Theta(n^{2.8})$

(a) $O(n^{2.8})$

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

$$\begin{aligned}\text{Ans} \rightarrow 16 \rightarrow f(n) &= 2^{2n} \\ \log f(n) &= 2n \log_2 2 \\ \log_{\log} f(n) &= 2n \\ f(n) &= 2^n \cdot 2^n \\ &= \Omega(2^n) \quad \text{Ans.}\end{aligned}$$

$$\begin{aligned}\text{Ans} \rightarrow 17 \rightarrow T(n) &= 2T\left(\frac{n}{2}\right) + n^2 \\ c &= 1 \\ n^c &= n \\ n^2 &> n \\ f(n) &> n^c \\ T(n) &= \Theta(n^2)\end{aligned}$$

$$\text{Ans} \rightarrow 18 \rightarrow O(\log N) \quad [\text{It's a G.C.D function where } n \text{ keeps } \div n \text{ decreasing by } n/2]$$

$$\begin{aligned}\text{Ans} \rightarrow 19 \rightarrow T(n) &= O[n^2 + n] \\ T(n) &= O(n^2) \quad \text{Ans.}\end{aligned}$$