

Time Complexity Solution

Date:

Ex-1

$$a = b;$$

Complexity: $O(1) \rightarrow$ constant time

Ex-2

$$\text{sum} = 0;$$

for ($i = 1; i \leq n; i++$)

$$\text{sum} += i$$

values of i

1

2

3

⋮

⋮

n

of time needed

1

1

1

⋮

⋮

+ 1

n

time complexity $O(n)$.

Ex-3 $sum = 0$ $for (j=1, j \leq n; j++)$ $for (i=1; i \leq j; i++)$
 $sum++;$ $for (k=0, k < n; k++)$ $A[k] = k;$ values of j# times inner loop runs

1

1

2

2

3

3

⋮

⋮

⋮

⋮

⋮

⋮

⋮

⋮

n

+ n

 $= 1+2+3+\dots+n$ $= \frac{n(n+1)}{2}$ $= \frac{n^2}{2} + \frac{n}{2}$ Time complexity = $O(n^2)$

Date:

Ex-4

```
sum = 0;
for (i = 1; i <= n; i++)
    for (j = 1; j <= n; j++)
        sum++;
```

values of i

1

2

3

⋮

⋮

n

of times inner loop runs

n

n

n

⋮

⋮

+

$n \times n$

$= n^2$

time complexity = $O(n^2)$

Ex-5

```
sum2 = 0;
for (i = 1, i <= n; i++)
    for (j = 1, j <= i; j++)
        sum2++;
```

values of i

1
2
3
⋮
n

of times inner loop runs

1

2

3

⋮

n

$$\begin{aligned}
 & \overbrace{1+2+3+\dots+n} \\
 &= \frac{n(n+1)}{2} \\
 &= \frac{n^2}{2} + \frac{n}{2}
 \end{aligned}$$

time complexity = $O(n^2)$

Date:

Ex-6

sum = 0

for (k = 1; k ≤ n; k = 2)

for (j = 1; j ≤ n; j++)

sum++

values of i

$$1 = 2^0$$

$$2 = 2^1$$

$$4 = 2^2$$

$$8 = 2^3$$

$$\vdots$$

$$n = 2^i$$

$$i = \log_2 n$$

of times inner loop runs

n

n

n

n

⋮

n

+

$$i \times n$$

$$= n \log n$$

$$O(n \log n)$$

Ex-7

sum = 0

for (k=1; k <= n; k *= 2)

for (j=1; j <= k; j++)

sum++

values of k

$$1 = 2^0$$

$$2 = 2^1$$

$$4 = 2^2$$

$$8 = 2^3$$

⋮

$$n = 2^i$$

$$i = \log_2 n$$

of times inner loop runs

$$1 = 2^0$$

$$2 = 2^1$$

$$4 = 2^2$$

$$8 = 2^3$$

⋮

$$n = 2^{\log_2 n}$$

$$= 2^0 + 2^1 + \dots + 2^{\log_2 n}$$

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

$$= \frac{2^{\log_2 n + 1} - 1}{2 - 1}$$

$$= 2^{\log_2 n + 1}$$

$$= 2 \cdot 2^{\log_2 n} = 2n$$

∴ complexity = $O(n)$