

NAME \rightarrow Vaibhav Sharma

Section \rightarrow CST-SPL-2 (wsp) 8 = n f

Roll No. \rightarrow 16

Data & Analysis of Algorithms

Tutorial - 1

1. Asymptotic notations are used to find the complexity of an algorithm when input is very large.

• Big O (O):

$$f(n) = O(g(n))$$

iff

$$f(n) \leq Cg(n) \quad \forall n > n_0$$

for some instant $C > 0$
 $g(n)$ is "tight upper bound" of $f(n)$

• Big Omega (Ω):

$$f(n) = \Omega(g(n))$$

iff

$$f(n) \geq Cg(n) \quad \forall n > n_0$$

for some constant $C > 0$
 $g(n)$ is "tight lower bound" of $f(n)$.

• ~~Any~~ Theta (Θ)

$$f(n) = \Theta(g(n))$$

$$c_1 g(n) \leq f(n) \leq c_2 g(n)$$

$$\forall n \geq \max(n_1, n_2)$$

for some constant $c_1 > 0$ & $c_2 > 0$
 $g(n)$ is both a tight upper bound & lower bound of $f(n)$.

2. (a) for $(i=1 \text{ to } n)$, $\{i = i * 2\}$

(1), 2, 4, 8, ... n

let K^{th} term $= n$

$$n = 1 \cdot (2^{K-1})$$

(a) \log_2 based upper bound Taking log on both sides

$$\log n = K-1 \log_2 2$$

$$K = 1 + \log n$$

$$O(1 + \log n)$$

$$O(\log n)$$

3

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

$$n = n-1 \text{ put (1)}$$

$$T(n-1) = 3T(n-2) \text{ --- (2)}$$

$$\text{put (2) in (1)}$$

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

$$n = n-2 \text{ put (1)}$$

$$T(n-2) = 3T(n-3) \text{ (3)}$$

$$T(n) = 27T(n-3)$$

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

$$n-k=0$$

$$n=k$$

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

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

$$= 3^n$$

$$O(3^n) \text{ Ans}$$

4.

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

$$(n \geq 1 \text{ } n-1 \geq 1 \text{ put (i)})$$

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

$$(2 \times T(n)) = 4T(n-2) \text{ --- (iii)}$$

$$n = n-2 \text{ put (i)}$$

$$T(n-2) = 2T(n-3) \text{ --- (iv)}$$

$$(2 \times 2 \times T(n)) = 8T(n-3)$$

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

$$n-k=0$$

$$k=n$$

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

$$= 2^n T(0)$$

$$= 2^n$$

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

6

Void function (int n)

```

{
    int i, count = 0;
    for (i = 1; i * i <= n; i++)
        count++;
}

```

$O(1 + \sqrt{n} + \sqrt{n} + \sqrt{n})$

$O(1 + 3\sqrt{n})$

$O(3\sqrt{n})$

~~$O(3\sqrt{n})$~~

$O(\sqrt{n})$

$O(n^{1/2})$ Ans

7

Void function (int n)

```

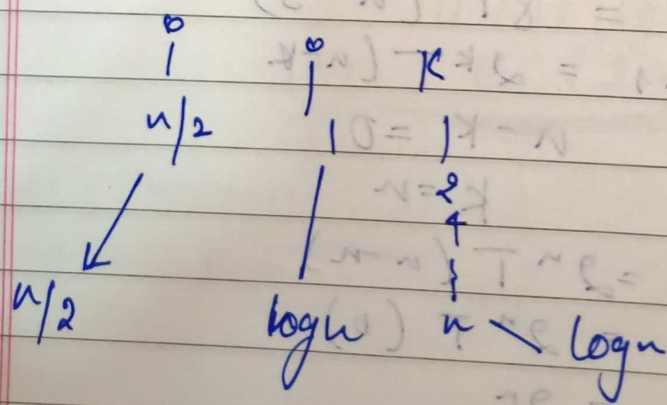
{
    int i, j, k, count = 0;
    for (i = n/2; i <= n; i++)
        for (j = 1; j <= n; j = j * 2)
            for (k = 1; k <= n; k = k * 2)
                count++;
}

```

$O(n/2 \times \log n \times \log n)$

$O(n (\log n)^2)$

Ans



9

Void function (Print n)

```

for (i=1 to n)
{
    for (j=1; j<=n; j=j+i)
        printf("%d", n);
}

```

$$O(n + n^2 + n^2 + n^2)$$

$$O(3n^2 + n)$$

$$\underline{O(n^2)} \text{ Ans}$$

10

As given $n^k \notin C^4$

relation b/w $n^k \notin C^4$ is

$$n^k = O(c^4)$$

$$n^k \leq a(c^4)$$

$$\forall n \geq n_0$$

• Constant, $a > 0$

$$\text{for } n_0 = 1$$

$$C = 2$$

$$\Rightarrow 1^k \leq a^{2.1}$$

$$\Rightarrow n_0 = 1 \text{ } \& C = 2.$$