RANKA
DATE / /
PAGE

Assignment -1.

Ans. 1. Asymptotic notations describe the growth rate of Algorithms in terms of input size. They provide a way to describe time complexity of Algorithms.

- Big O(O) : Represent the upper bound of worst case scenario.
- Omega (Ω) : Represents the lower bound or best case scenario.
- Theta : Represent both upper and lower bounds, indicating tight bounds.

Ans. 2 O(log n)
∵ value of i will be doubled until it exceeds n.

Ans. 3 $3T(n-1) \quad n>0, T(0)=1.$

$T(n) = 3T(n-1)$
$T(0) = 1$

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

$T(n-1) = 3T(n-2) \quad | \quad T_n = 3T(n-2).3$
$T(n-2) = 3T(n-3) \quad | \quad T_n = T(n-3).3.3.3$

$T(n) = 3.3.3.T(0)$
n

$T_n = 3 \cdot 3 \cdot 3 \cdots \underset{n \text{ times}}{} \cdot T(0)$

$T_n = 3^n$

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

Q.3 $[2\,T(n-1)-1]$ for $n>0$, otherwise 1.

$T(0) = 1$

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

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

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

~~$T(n) = 2 \times 2 \times T$~~

$T(n) = 2 \times [2T(n-2)-1]-1$

$T(n) = 2 \times [2 \times (2T(n-3)-1)-1]-1$

$T(n) = 2 \times [8T(n-3)-2]-1$

$= 16T(n-3)-4-1$

$T(3) = 16T(n-3)-5$

$T(n) = 2+2\times2-\cdots\times2 \cdot (T(0)-n)$
$-2^2-2^1-2^0$

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

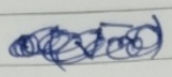
$T(n) = 2^n-2^n+1 = 1$

~~$O(2^n)$~~

T.C = $O(1)$

5.

Time complexity :-



$i = 1, 2, 3, 4, 5, \ldots$

$s = 1, 3, 6, 10, 15, \ldots$

$s = \frac{i.(i+1)}{2} \leq n$

$i^2 + i \leq 2n.$

$i \approx \sqrt{2n}$

So,

T.C $= O(\sqrt{n})$.

Q.6 $i * i < n$

$i^2 < n$

$i \approx \sqrt{n}.$

T.C :-

$O(\sqrt{n})$

7. Coor outer loop runs - n times
And.
T.C of nested loops - O(log n)

So.
T.C = O(n log n).

8. function(int n) {

if (n == 1) return;
for (i = 1 to n) { — O(n)
for (j = 1 to n) { — O(n)
print("*");
}
}
function(n - 3); — T(n/3)
}

~~T.C = O(n^3)~~.

T.C = $O(n^2) + T(n/3)$

(9) void function (int n) {

```
for (i = 1 to n) {        ——— (n)
    for (j = 1; j <= n; j = j + i)  ——— (n)
    {
        print ("*");
    }
}
```

$i = 1$, $j = 1, 2, 3, 4 \ldots$

$i = 2$, $j = 1, 3, 5, \ldots$

$i = 3$, $j = 1, 4, 7, \ldots$

$\therefore \quad T(n) = n + n/2 + n/3 + n/4 + \ldots$

$T.C \Rightarrow O(n \log n)$

(10) Asymptotic relationship between n^k and c^n is that c^n grows faster than n^k. In other words, c^n is an exponential growth function, while n^k is a polynomial growth function.