Jaypee University of Engineering and Technology

18B11CI311 – Data Structures
B.Tech -3rd Semester
Tutorial – 1 (Time Complexity)

1.Find the exact step counts (growth function) and time complexity for following algorithms -

i. 1.Input m and n 2.If m == n 3. Print "Same" 4. Sum=m+n 5. Print Sum 6.Else Print "Not same"	**ii.** 1. Initialize i =0 2. For i=0 to n-1 3. Read an integer in Arr[i] 4. For i=0 to n-1 5. Print Arr[i]	**iii.** Algorithm Add(a,b,c,m,n) 1. for i :=1to m do 2. for i :=1to n do 3. c[i,j]:=a[i, j]+b[i,j];
iv. 1.int i, j, k = 0; 2.for (i =1; i <= n*n; i++) 3. for (j = 1; j <= n; j = j +1) 4. k = k + n/2;	**v.** Algorithm Sum (a,n) 1 s:=0.0; 2 for i :=1to n do 3 s :=s+a[i]; 4 return s;	**vi.** 1. sum = 0 2. for i = 1 to n: 3. for j = i to n: 4. sum += a[i][j] 5. print(sum)
vii. 1. Set i =0 2. Repeat step 3-6 till i<=n 3. Set J=n 4. While J>0 5. J = J/2 6. i = i+1	**viii.** 1. Set I =n 2. While I > 1 3. J =1 4. While J < n 5. K=1 6. While K < n 7. K += 2 8. J *= 2 9. I /= 2	**ix.** void fun(int n, int arr[]) 1. { 2. int i = 0, j = 0; 3. for(; i < n; ++i) 4. while(j < n && arr[i] < arr[j]) 5. j++; 6. }

2. Show that following statements are correct:

i. 4n+100=O(n) ii. $500n^3 +6n+6=O(n^3)$

iii. $n^3 \neq O(n^2)$ iv. $5n^2-6n= O(n^2)$

v. $n! = O(n^n)$ vi. $2n^2 2^n +nlogn= O(n^2 2^n)$

vii. $3n^3 +4n^2 = \Omega(n^2)$ viii. $\sum_{i=0}^{n} i^2 = \theta(n^3)$

3. Compare the two functions n^2 and $2^n/4$ for various values of n. Determine when the second becomes larger than the first.

4. Which of the given options provides the increasing order of asymptotic complexity of functions f1, f2, f3 and f4?

f1(n) = 2^n, f2(n) = n^(3/2), f3(n) = nLogn, f4(n) = n^(Logn)