H Roll No. .....

TCS/TIT–502

B. TECH. (CS/IT) (FIFTH SEMESTER)
MID SEMESTER EXAMINATION, 2018
DESIGN AND ANALYSIS OF ALGORITHMS

Time : 1:30 Hours

Maximum Marks : 50

Note : (i) This question paper contains two Sections.

(ii) Both Sections are compulsory.

Section—A

1. Fill in the blanks/True/False : (1×5=5 Marks)

(a) Time complexity of Heap sort is given by Θ (.....).

(b) If T (n) = 16* T (n/4) + n*n*lg n, then T (n) = Θ (.....).

(c) If T (n) = n^4 + ig n^n + $2^{lg\ n}$, then T (n) = Θ (.....).

(d) Quick sort is best algorithm when array is already sorted. (True/False)

(e) Selection sort is better then bubble sort in most of the cases. (True/False)

2. Attempt any *five* parts : (3×5=15 Marks)
   (a) Define time and space complexity.
   (b) Give the Recurrence relation for Merge sort and solve it using Master's method.
   (c) Differentiate between fractional knapsack and 0–1 knapsack problem.
   (d) Design algorithm for Brute force string matching.
   (e) Design algorithm for Activity selection problem.
   (f) Derive run time complexity of Max_ Heapify function used in Heap Sort.

Section—B

3. Attempt any *two* parts of choice from (a), (b) and (c). (5×2=10 Marks)
   (a) Solve T (n) = 3* T (n – 1) + c if n > 1 and T (1) = c where c is positive constant.
   (b) Find out run time complexity of the following code :

```
for (i = 1; i < = n; i = 2* i)
  { j = n;
    while (j > 0)
    { p = q + r;
      j - -;}
    }
```


   (c) Explain Master's method with the help of example.

4. Attempt any *two* parts of choice from (a), (b) and (c). (5×2=10 Marks)
   (a) Solve the given recurrence relation using Recursive tree method :
   T (n) = 4* T (n/2) + n assume T (1) = 1.
   (b) Design the algorithm of bubble sort for sorting numbers in the decreasing order and derive the Time Complexity for Worst Case.
   (c) Give solution for the following fractional-Knapsack problem (Knapsack Size = 16) :

Item	Cost	Weight
1	8	5
2	30	10
3	20	6
4	15	4

5. Attempt any *two* parts of choice from (a), (b) and (c). (5×2=10 Marks)
   (a) Apply Quick sort on the following sequence to sort and show intermediate steps :
   16, 9, 11, 4, 17, 5, 3, 12

(b) Derive the run time complexity for insertion sort for Best Case and Worst Case.

(c) Arrange the following function into increasing order of their asymptotic growth :

(i) $f(n) = 2^{[n*\lg n]} + n^2 + 6$

(ii) $g(n) = n^{10} + \lg(n!) + n$

(iii) $h(n) = \lg(n^n) + 8^{(\lg n)} + n$

F. No. : b-46