(b) Assume that you are given a chain of matrices < A1 A2 A3 A4 >, with dimensions 2x3, 3x4, 4x6, and 6x7 respectively. Compute the optimal number of multiplications required to calculate the chain product and also indicate what the optimal order of multiplication should be using parentheses. [4]

6. (a) Give an algorithm to detect whether an undirected graph has a cycle or not. The algorithm must be an O(V) algorithm independent of the number of edges. [4]

(b) We are given 10 tasks T1, T2, T3, …., T9, T10. The execution of each task requires one unit of time. We can execute one task at a time. Each task Ti has a profit Pi and a deadline Di. Profit Pi is earned if the task is completed before the end of the Di unit of time. [4]

Task	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
Pi	20	21	34	25	34	12	28	16	26	30
Di	4	5	3	2	4	5	2	1	3	6

7. (a) Give an algorithm to compute MST in a graph where all edges have equal weights. Analyze its complexity. The algorithm should be more efficient than Prim's and Kruskal's. You may assume positive weights. [4]

(b) Find an optimal solution to the knapsack instance n=8, W=15. (v1, v2, v3, v4, v5, v6, v7, v8) = (10, 15, 15, 17, 16, 18, 13, 20) and (w1, w2, w3, w4, w5, w6, w7, w8) = (1, 2, 3, 5, 7, 1, 4, 5), where *n* is the number of items, W is the knapsack capacity that thief can carry, v_i stands for value or profit w_i stands for weight of the i^{th} element. [4]

8. Write short note (Any two) [4×2]

(a) Divine Conquer Vs. Dynamic Programming

(b) Union By Rank Dis-joint Set Data Structure

(c) P, NP and NPC class of problems

*****

AUTUMN END SEMESTER EXAMINATION-2018

5th Semester B.Tech & B.Tech Dual Degree

DESIGN AND ANALYSIS OF ALGORITHMS

CS-3001 / CS-502

[For 2015(L.E.), 2014 & Previous Admitted Batches]

Time: 3 Hours Full Marks: 60

Answer any Six questions including question No.1 which is compulsory.

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable and all parts of a question should be answered at one place only.

1. Answer the following questions: [2×10]

(a) Sort the functions in increasing order of asymptotic (big-O) complexity:

$f1(n)=n^{0.999999}\log n$, $f2(n)=10000000n$,

$f3(n)=1.000001^n$, $f4(n)=n^2$

(b) Consider the following C function.

```
int fun (int n)
{
  int i. j;
  for(i=1; i<=n; i++)
  {
     for(j=1; j<=n; j+=i)
     {
        print("%d ", i+j);
     }
  }
}
```

What is the time complexity of function fun in terms of Θ-notation?

(c) Construct a binary min heap containing the items 9, 2, 4, 3, 2, 1, 0, 8, 7. Give the final structure only. How many min heaps from the given inputs can possible?

(d) Your friend guesses an integer between 0 and n. You can ask questions like is the number less than 100? He will give YES NO answers. How many questions can your friend force you to ask, if you are a smart person?

(e) Write a recursive algorithm to compute the maximum element in an array of integers. You may assume the existence of a function "max(a, b)" that returns the maximum of two integers a and b.

(f) Assume that a merge sort algorithm in the worst case takes 30 seconds for an input of size 64. Which of the following most closely approximates the maximum input size of a problem that can be solved in 6 minutes?
p.256 q. 512 r.1024 s.20148

(g) What can be the best data structure to be used to find 10 maximum numbers from a big file containing billions of numbers? What is the worst case time complexity of this problem w.r.t. the data structure used?

(h) Define optimal storage on tapes problem.

(i) What is the need of randomized quick sort as compared to normal quick sort? Give a suitable example.

(j) Match the following:

(P) Prim's algorithm for minimum spanning tree	(i) Backtracking
(Q) Floyd-Warshall algorithm for all pairs shortest paths	(ii) Greedy method
(R) Merge sort	(iii) Dynamic programming
(S) Sum of Subset problem	(iv) Divide and conquer

2. (a) Write the pseudo code for insertion sort algorithm. Derive its best and worst case time complexity. [4]

(b) Solve the following recurrence relation without using Master's theorem: [4]

$$T(n) = 3T(n/2) + cn$$

3. (a) Let A be an n-element array. Design and analyze divide-and-conquer algorithm for finding the maximum value in an array A. [4]

(b) Use a dynamic programming algorithm to find the Longest Common Subsequence between the following two sequences: [4]
X = bababaaba
Y = baababaab

4. (a) What is an n-queens problem? Explain it for 4 queens. [4]

(b) Construct a Huffman code for the following data (show all the steps): [4]

Symbol	a	b	c	d
Frequency	0.1	0.2	0.3	0.4

How many bits are needed to encode a string containing 5 A's, 15 B's, 10 C's and 2 D's using this code. Compare this code with another code where each character is encoded with fixed two bits. Which code is better?

5. (a) You have to sort a sequence of n elements. The n elements have n/k subsequences of size k each. The subsequences have the following property: All elements of a subsequence are less than those of the preceding one and greater than those of the following subsequence. An example sequence of 6 elements with k = 2 is 2, 1, 5, 6, 21, 12. The subsequences here are 2, 1; 5, 6and 21, 12: Note that all elements of 2, 1 are less than those of 5, 6 and so on. You know the value of k. Give an O(nlogk) algorithm to sort the entire sequence. [4]