Total No. of Pages:03

Roll No. DTU/2K15/CO/049

4 TH SEMESTER

[B.TECH. – CO/IT]

END SEM EXAM

(MAY- 2017)

CO/IT-208 ALGORITHMS DESIGN AND ANALYSIS

Time: 3 Hours **Max. Marks: 50**

Note: Attempt any 5 Questions Each Question Carry Equal Marks. Assume suitable missing data, if any.

Q1

a) For each of the following recurrence, solve them with the help of master theorem? (2X2=4)

(i) $T(n) = 0.5T\left(\frac{n}{2}\right) + n^2$

(ii) $T(n) = \sqrt{2}T\left(\frac{n}{2}\right) + logn$

b) Explain Big O, big Ω, big θ asymptotic notations. (6)

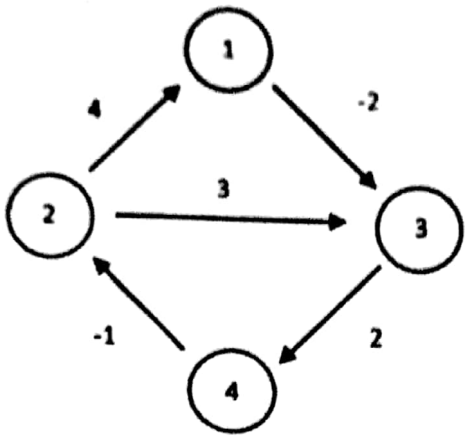
Q2

a) Write down the randomized quicksort algorithm. Also explain in detail what special incite does average-case gives to us? (2+3)

b) Suppose you were to drive from St. Louis to Denver along I-70. Your gas tank, when full, holds enough gas to travel m miles, and you have a map that gives distances between gas stations along the route. Let $d_1 < d_2 < \ldots\ldots < d_n$ be the locations of all the gas stations along the route where d_i is the distance from St. Louis to the gas station. You can assume that the distance between neighboring gas stations is at most m miles. Your goal is to make as few gas stops as possible along the way. Apply greedy approach to solve the problem by listing the algorithm, optimal substructure, greedy choice property. Also discuss your algorithm with the help of example and complexity analysis? (5)

Q3

a) Write down Floyd Warshall algorithm and its complexity. Show step by step implementation of the same algorithm on the given graph. (2+2)



b) Consider the below given recursive algorithm for calculating factorial of any integer n.

```
int fact( int n){
if(n==1)
        return 1;
else if(n==0) return 1;
else
    return n * fact(n-1);
}
```

First analyze the time complexity for the given algorithm. Then modify the same algorithm to calculate the series $1! + 2! + \cdots \ldots \ldots + n!$ when n is fed as the input without disturbing its divide and conquer nature. Now transform the modified algorithm to make it a bottom up dynamic programming algorithm. Compare the two algorithms in terms of complexity and overlapping sub problems.

(1+1+1+3).

Q4

(a) How is the dynamic programming solution of the travelling salesman problem, better than the naïve algorithm? Discuss with the help of example. Also derive and compare their complexity? (5)

(b) Consider a two team game where team A and B, play a series of games until one of the teams win 'n' games. Assume that the probability of team A winning a game is 'p', and losing it is q=(1-p). We also assume that there is no tie in a game. Let p(i,j) be the probability of team A winning the series if A needs 'i' more games to win the series and B needs 'j' more games to win the series. After setting a recurrence relation for p (i, j), write the pseudocode for solving this problem using dynamic programming. What is the complexity of the algorithm.

(2+2+1)

Q5

a) What is LC branch and bound? Apply LC search on famous 0/1 knapsack problem for the given instance of the problem. Capacity of knapsack is 12 kg? (2+3

Item	Weight	Value
1	4	$10
2	6	$15
3	3	$6
4	5	$8
5	2	$4

b) Define backtracking phenomenon? Cut the state space for 4- queen problem by applying back tracking? Also mention how you are going to take care of diagonal constraint in its implementation? (2+2+1)

Q6

(a) Explain Rabin-karp algorithm for string matching using an example? Analyze its run time complexity (3+2)

(b) Write down the pseudocode and obtain Huffman codes for the following instance and for the same and analyze its complexity?

Character	a	b	c	d	e	f	g	h
No of occurrences (in thousands)	1	1	2	3	5	8	13	21

(2+2+1)

Q7 (3x2=6)

a) Write short notes on:-

(i) P class

(ii) NP class

(iii) NP complete class

b) Write down an approximate solution for vertex cover problem? Also apply the same on the given graph and also explain its complexity? (1+2+1)