No. of Printed Pages : **3**

Sl. No.

C4-R4 : ADVANCED ALGORITHMS

NOTE :

1. Answer question 1 and any FOUR questions from 2 to 7.

2. Parts of the same question should be answered together and in the same sequence.

Time : 3 Hours **Total Marks : 100**

1. (a) Solve T(n) = T(n/4) + T(n/2) + cn^2; T(1) = c; T(0) = 0, where c is a positive constant.

(b) Does greedy algorithm give optimal solution for 0-1 Knapsack ? Justify your answer.

(c) What are the main principles (characteristics) of dynamic programming algorithms ?

(d) Compare Prim's and Kruskal's algorithms and what are they used for ?

(e) Heap sort and Merge sort have worst time complexity of **n* log n** but still Quick sort with worst case complexity as **n^2** is better. Explain why ?

(f) What are the classes of P and NP ? Is P = NP ?

(g) Prove that (i) $n! = {}_{o}(n^{n})$ (ii) $\lg(n!) = \theta(n\lg n)$. **(7x4)**

2. (a) Compute the prefix function π for pattern **abcaabbccab** when the alphabet is $\Sigma = \{a, b, c\}$.

(b) Is the sequence 23, 17, 14, 6, 13, 10, 1, 5, 7, 12 a max-heap ?

(c) (i) Find the running time of QUICK SORT when all elements of array A have the same value.

(ii) Show that the running time of QUICK SORT is Θ(n^2) when the array A contains distinct elements and is sorted in decreasing order. **(6+6+6)**

3. (a) A sequence of stack operations is performed on a stack whose size never exceeds k. After every k operations, a copy of the entire stack is made for backup purposes. Show that the cost of n stack operations, including copying the stack, is O(n) by assigning suitable amortized costs to the various stack operations.

(b) Suppose that we have a file (i.e., a long bit string) that is to be compressed. Suppose, further, that we decide to view the file as being composed of 3-bit blocks. (For convenience, we assume that the length of the file is evenly divisible by three.) We interpret each block as a single character. (Under this scenario, there are eight characters in the source alphabet, one for each of the distinct bit strings of length three.) The frequency with which each of the characters occurs in the file is given in the following table. (For convenience, we refer to the eight characters as a, b, . . . , h, as indicated in the table.)

Character	Frequency	Character	Frequency
a (000)	11.9%	e (100)	6.7%
b (001)	9.6%	f (101)	7.7%
c (010)	24.7%	g (110)	9.2%
d (011)	28.3%	h (111)	1.9%

(i) Construct a Huffman tree corresponding to these frequencies.

(ii) Show the binary encoding of each character (consistent with the tree constructed in (i)).

(iii) Compute the ratio between the lengths of the compressed file and the original file. **(9+9)**

4. (a) Suppose you were to drive from Lucknow to Delhi 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 $d_1 < d_2 <\ldots <d_n$ be the locations of all the gas stations along the route where d_i is the distance from Lucknow to the gas station. You can assume that the distance between neighbouring gas stations is at most m miles. Your goal is to make as few gas stops as possible along the way. Give the most efficient algorithm you can find to determine at which gas stations you should stop and prove that your strategy yields an optimal solution. Be sure to give the time complexity of your algorithm as a function of n.

(b) Consider the following algorithm.

```
Algorithm Mystery (n)//Input : A non negative integer n
S ← 0
for i ← 1 to n do
S ← S + i * i
return S
```

(i) What does this algorithm compute ?

(ii) What is its basic operation ?

(iii) How many times is the basic operation executed ?

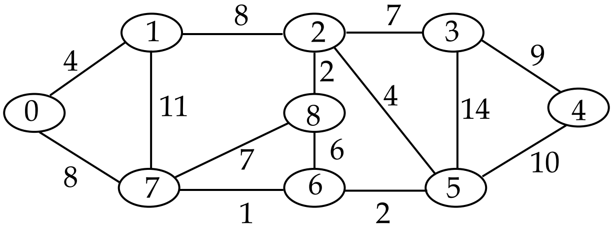
(iv) What is the efficiency class of this algorithm ?

(v) Suggest an improvement or a better algorithm altogether and indicate its efficiency class. If you cannot do it, try to prove that, in fact, it cannot be done. **(9+9)**

5. (a) Consider each of the following words as a set of letters: {arid, dash, drain, heard, lost, nose, shun, slate, snare, thread}. Show which set cover GREEDY-SET-COVER produces when ties are broken in favor of the word that appears first in the dictionary.

(b) Compute the values (d, *x*, *y*) that the call EXTENDED-EUCLID(899, 493) returns.

(c) Apply Dijkstra's Algorithm to find the shortest path.



(6+6+6)

6 . (a) (i) Define the 0-1 Knapsack problem and Express the 0-1 Knapsack Problem as a set.

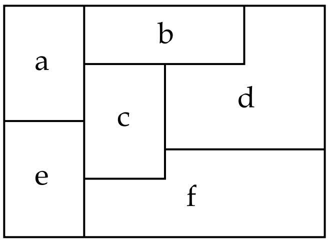
(ii) Show that there is a polynomial time transformation from the Partition Problem to the 0-1 Knapsack Problem.

(b) Define the Vertex cover problem of an undirected graph. The following is a proposed algorithm for the vertex cover problem. It uses the notion of the degree of a vertex, which is the number of edges that hit this vertex.

```
S = empty set.
While G has at least one edge
Find a vertex v of the highest possible degree in G
Add v to S.
Remove vertex v from G, along with all edges that hit v
Answer S.
```

A vertex cover needs to cover all of the edges, where an edge is covered by choosing a vertex from that edge. This algorithm is based on the principle that, by choosing a vertex with the most edges hitting it, we get covered the most edges. What kind of algorithm is this ? Is it guaranteed to find the smallest vertex cover ? If so, why ? If not, give a counter example. **(9+9)**

7. (a) Consider the following map :



Explain how we can use the graph-coloring problem to color the map so that no two neighboring regions are colored the same. Use your answer to above to color the map with the smallest number of colors.

(b) (i) What is the smallest number of divisions made by Euclid's algorithm among all inputs $1 \leq m, n \leq 10$?

(ii) What is the largest number of divisions made by Euclid's algorithm among all inputs $1 \leq m, n \leq 10$? **(9+9)**

- o O o -