RV Educational Institutions ®
RV College of Engineering ®
Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi
Approved by AICTE, New Delhi
Go, change the world

Academic year 2021-2022 (Even Sem)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Date	29th June 2022	Maximum Marks	50
Course Code	18CS43	Duration	120 Min
Sem	IV Semester	Test-I	
DESIGN & ANALYSIS OF ALGORITHMS			

Sl. No.	**PART A**	M	BT	CO
1	In empirical analysis the quadratic efficiency class of the algorithm will have what type of graph	**1**	**2**	**2**
2	State the basic operation in the Tower of Hanoi Problem	**1**	**1**	**1**
3	For an input size of 15 elements, how many times the basic operation will be executed in selection sort.	**1**	**2**	**2**
4	Write the recurrence to denote the worst case of Quicksort	**1**	**2**	**2**
5	The algorithm like Quick sort does not require extra memory for carrying out the sorting procedure. This technique is called ___________	**1**	**1**	**1**
6	If $T(n) = 7T(n/3) + n^2$, then by master method $T(n) =$	**1**	**3**	**2**
7	Find the number of swaps done to sort the following elements in alphabetical order using Bubble Sort **E X A M P L E**	**2**	**3**	**3**

8	What is the time complexity of following code
	`void fun( )`
	`  {`
	`              for( i=1,i<=n;i++)`
	`              for(j=1;j<=i`^{2}`;j++)`
	`              for(k=1;k<=n/2;k++)`
	`                   sum=sum + a[i][j]*k;`
	` }`
		2	**3**	**3**

	PART B			
1a	Discuss with a neat flow chart the algorithm design and analysis process.	**6**	**2**	**1**
1b	For the algorithm to find the largest element in a list of n numbers, indicate (i) a natural size metric for its inputs (ii) its basic operation (iii) whether the basic operation count can be different for inputs of the same size:	**4**	**3**	**1**
2a	Write a recursive Tower of Hanoi and analyze its efficiency by writing the recurrence relation.	**6**	**3**	**2**
2b	Illustrate the general plan to analyze the efficiency of non-recursive algorithm.	**4**	**1**	**2**
3a	With an algorithm discuss the efficiency of selection sort.	**6**	**2**	**2**

3b	Compare the orders of growth using limits: i) ½ n(n-1) and n^2 ii) n! and 2^n	**4**	**3**	**3**
4a	Write the partition algorithm used in quicksort. Apply the same to sort the elements 5 3 1 9 8 2 4 7. Draw the recursive call tree.	**6**	**3**	**3**
4b	Discuss the efficiency of quicksort.	**4**	**2**	**2**
5a	Apply Strassen's algorithm to compute the matrix multiplication of A1 and B1 matrix. $\mathbf{A1} = \begin{bmatrix} 3 & 2 \\ 5 & 6 \end{bmatrix}$ $\mathbf{B1} = \begin{bmatrix} 5 & 6 \\ 1 & 3 \end{bmatrix}$	**6**	**4**	**4**
5b	Write a pseudocode for a divide-and-conquer algorithm for finding the position of the largest element in an array of n numbers.	**4**	**3**	**2**

Course Outcomes:

COURSE OUTCOMES

1. Understand and explore the asymptotic runtime complexity of algorithms by using mathematical relations.
2. Select and apply appropriate design techniques to solve real world problems.
3. Estimate the computational complexity of different algorithms.
4. Apply the efficient algorithm design approaches in a problem specific manner.

BT-Blooms Taxonomy, CO-Course Outcomes, M-Marks

Marks Distribution	Particulars		CO1	CO2	CO3	CO4	L1	L2	L3	L4	L5	L6
	Test	Max Marks	14	28	14	04	06	20	30	04	-	-

****

Academic Year 2022- 2023 (Even Sem)

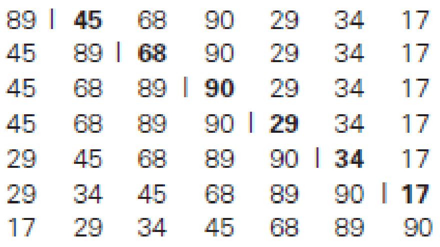
Department of Computer Science and Engineering

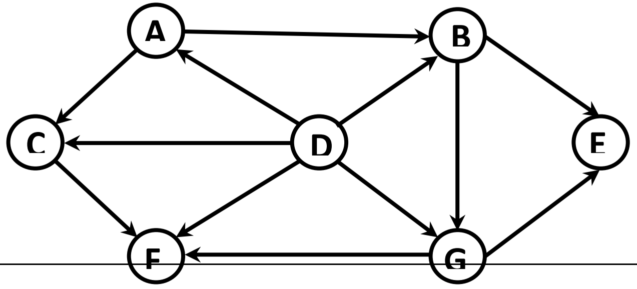
Date	10/8/2023	Maximum Marks	50
Course Code	21CS43	Duration	90 Min

Design and Analysis of Algorithms
CIE 2 Scheme and Solution

<table>
<tr><th>SL</th><th>PART A</th><th>M</th><th>BT</th><th>CO</th></tr>
<tr><td>1 a</td><td>You are given n coins (where n = 2k, for k is a Natural number) which look exactly alike but one is fake (Lighter than the rest). Write an efficient algorithm to determine the fake coin using a balance (Only balance is given and no weights). Find the complexity of the algorithm considering weighing as the basic operation. Comment on the design technique used.
Algorithm = 3 Marks
Time complexity = 1 Marks($\log_2 n$)
Design Technique = 1 Marks(Decrease and Conquer)</td><td>5</td><td>3</td><td>2</td></tr>
<tr><td>1b</td><td>Write an algorithm to perform insertion sort, and sort the given elements using same 89, 45, 68, 90, 29, 34, 17
Algorithm 2 marks

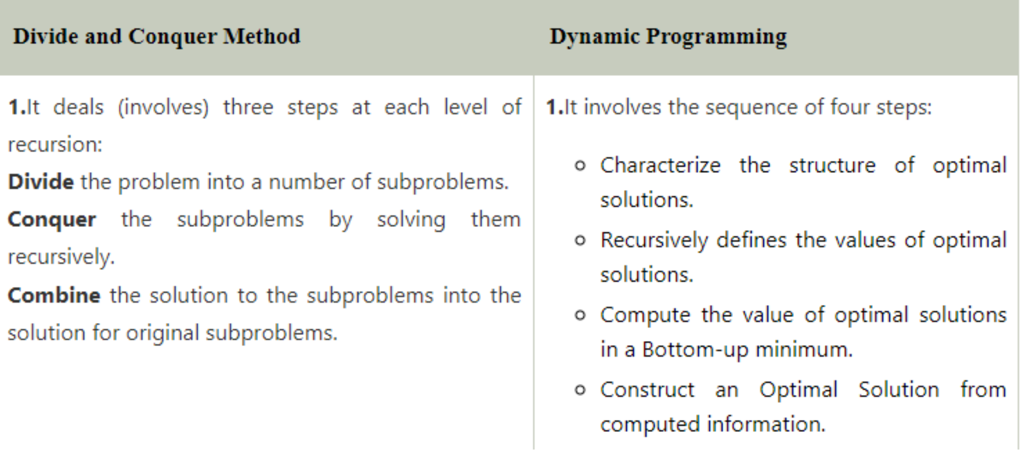
```
ALGORITHM InsertionSort(A[0..n − 1])
//Sorts a given array by insertion sort
//Input: An array A[0..n − 1] of n orderable elements
//Output: Array A[0..n − 1] sorted in nondecreasing order
for i ← 1 to n − 1 do
    v ← A[i]
    j ← i − 1
    while j ≥ 0 and A[j] > v do
        A[j + 1] ← A[j]
        j ← j − 1
    A[j + 1] ← v
```

Tracing


```
89 | 45   68   90   29   34   17
45   89 | 68   90   29   34   17
45   68   89 | 90   29   34   17
45   68   89   90 | 29   34   17
29   45   68   89   90 | 34   17
29   34   45   68   89   90 | 17
17   29   34   45   68   89   90
```
</td><td>5</td><td>2</td><td>2</td></tr>
<tr><td>2a</td><td>Along with DFS algorithm write the topological order of the given graph

</td><td>5</td><td>3</td><td>3</td></tr>
</table>

Topological Order : D, A , C, B, G, F, E

2b Illustrate with example how Dynamic programming is differ from Divide and Conquer — 5 | 2 | 4



Divide and Conquer Method	Dynamic Programming
1.It deals (involves) three steps at each level of recursion: **Divide** the problem into a number of subproblems. **Conquer** the subproblems by solving them recursively. **Combine** the solution to the subproblems into the solution for original subproblems.	**1.**It involves the sequence of four steps: ○ Characterize the structure of optimal solutions. ○ Recursively defines the values of optimal solutions. ○ Compute the value of optimal solutions in a Bottom-up minimum. ○ Construct an Optimal Solution from computed information.

Example 2 marks

3a Design Sort by counting algorithm and Sort the elements using same 20,25,21,23,67,22,23,28,26,21 — 5 | 3 | 3

```
ALGORITHM ComparisonCountingSort(A[0..n − 1])
//Sorts an array by comparison counting
//Input: An array A[0..n − 1] of orderable elements
//Output: Array S[0..n − 1] of A's elements sorted in nondecreasing order
for i ← 0 to n − 1 do Count[i] ← 0
for i ← 0 to n − 2 do
for j ← i + 1 to n − 1 do
if A[i] < A[j]
            Count[j] ← Count[j] + 1
else Count[i] ← Count[i] + 1
for i ← 0 to n − 1 do S[Count[i]] ← A[i]
return S
```

Algorithm Tracing 3 marks

3b Discuss the procedure used in Boyers Moor algorithm. in Apply the same to search the given pattern in the text — 5 | 3 | 4

Text: BESS_KNEW_ABOUT_BAOBAB.

Pattern: BAOBAB

B	A	O	B	A	B

Bad shift table is

B	A	O	_	Other
2	1	3	6	6

We need to find the value of d1
Where d1=t(c)-k
t(c) is the value of bad shift table
k is the number of matching character

K	Pattern	No Shifts d2
1	BAOBAB	2
2	BAOBAB	5
3	BAOBAB	5

B	E	S	S	_	K	N	E	W	_	A	B	O	U	T	_	B	A	O	B	A	B

B	A	O	B	A	B

4a | 5 | 1 | 3

Discuss Presort Element Uniqueness algorithm with its time complexity

```
ALGORITHM PresortElementUniqueness(A[0..n − 1])
//Solves the element uniqueness problem by sorting the array first
//Input: An array A[0..n − 1] of orderable elements
//Output: Returns “true” if A has no equal elements, “false” otherwise
sort the array A
for i←0 to n − 2 do
        if A[i]= A[i+ 1] return false
return true
```

T (n) = Tsort(n) + Tscan(n) → (n log n) + (n) = (n log n)

4b | 5 | 2 | 2

Discuss how the problems are solved in Dynamic Programming? Find the value of $^4C_3$ using dynamic programming

$$C(n,k) = \begin{cases} 1 & k=0 \text{ or } k=n \\ 0 & k>n \\ C(n-1,\ k)+c(n-1,\ k-1) & k<n \ \&\ k!=0 \end{cases}$$

k →

n ↓

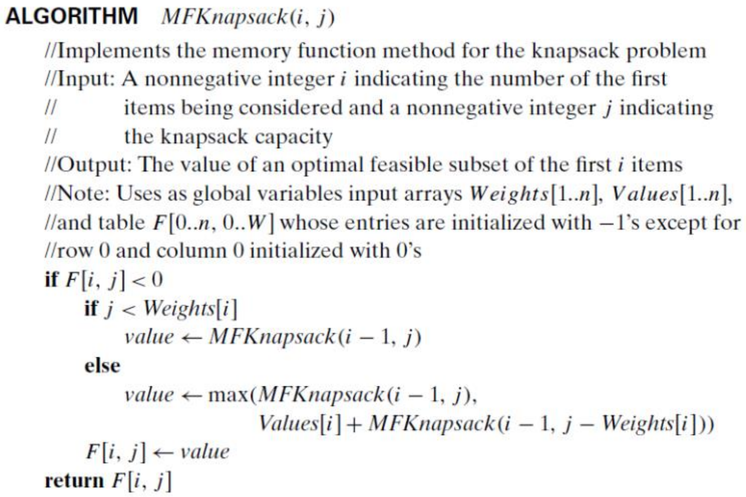
	0	1	2	3
0	1			
1	1	1		
2	1	2	1	
3	1	3	3	1
4	1	4	6	4

5a | 4 | 4 | 2

Apply Floyd’s algorithm for the weighted matrix given below

$$\begin{matrix} 0 & 2 & \infty & 1 & \infty \\ 6 & 0 & 3 & 2 & \infty \\ \infty & \infty & 0 & 4 & \infty \\ \infty & \infty & 2 & 0 & 3 \\ 3 & \infty & \infty & \infty & 0 \end{matrix}$$

D1	D2	D3	D4	Final Matrix

<table>
<tr><td></td><td>$\begin{matrix}0&2&\infty&1&\infty\\6&0&3&2&\infty\\\infty&\infty&0&4&\infty\\\infty&\infty&2&0&3\\3&\infty&\infty&\infty&0\end{matrix}$ $\begin{matrix}0&2&\infty&1&\infty\\6&0&3&2&\infty\\\infty&\infty&0&4&\infty\\\infty&\infty&2&0&3\\3&\infty&\infty&3&0\end{matrix}$ $\begin{matrix}0&2&\infty&1&\infty\\5&0&3&2&\infty\\9&\infty&0&4&\infty\\8&\infty&2&0&3\\3&\infty&\infty&3&0\end{matrix}$ $\begin{matrix}0&2&6&1&\infty\\5&0&3&2&\infty\\9&\infty&0&4&\infty\\8&\infty&2&0&3\\3&\infty&5&3&0\end{matrix}$ $\begin{matrix}0&2&6&1&\infty\\5&0&3&2&\infty\\9&11&0&4&\infty\\8&10&2&0&3\\3&5&9&3&0\end{matrix}$</td><td></td><td></td><td></td></tr>
<tr><td>5b</td><td>Write an algorithm to solve 0/1 Knapsack problem using Memory Functions, apply the same to find the maximum profit
Algorithm 3 Marks

<pre>
ALGORITHM MFKnapsack(i, j)
    //Implements the memory function method for the knapsack problem
    //Input: A nonnegative integer i indicating the number of the first
    //   items being considered and a nonnegative integer j indicating
    //   the knapsack capacity
    //Output: The value of an optimal feasible subset of the first i items
    //Note: Uses as global variables input arrays Weights[1..n], Values[1..n],
    //and table F[0..n, 0..W] whose entries are initialized with −1's except for
    //row 0 and column 0 initialized with 0's
    if F[i, j] < 0
        if j < Weights[i]
            value ← MFKnapsack(i − 1, j)
        else
            value ← max(MFKnapsack(i − 1, j),
                            Values[i] + MFKnapsack(i − 1, j − Weights[i]))
        F[i, j] ← value
    return F[i, j]
</pre>
W= 5
w_i 2 1 3 2
v_i 12 10 20 15

<b>Tracing of the input 3 marks</b>
<b>Maximum Profit: 37</b></td><td>6</td><td>3</td><td>4</td></tr>
</table>

Course Outcomes: After completing the course, the students will be able to:-	
CO1	Apply knowledge of computing and mathematics to algorithm analysis and design
CO2	Analyze a problem and identify the computing requirements appropriate for a solution
CO3	Apply mathematical foundations, algorithmic principles, and computer science theory to the modeling, and evaluation of computer-based solutions in a way that demonstrates comprehension of the trade-offs involved in design choices.
CO4	Investigate and apply optimal design, development principles, skills and tools in the construction of software solutions of varying complexity.
CO5	Demonstrate critical, innovative thinking, and display competence in oral, written, and visual communication.
CO6	Exhibits positive group communication exchanges in order to accomplish a common goal and engage in continuing professional development.

Marks Distribution	Particulars	CO1	CO2	CO3	CO4	CO5	CO6	L1	L2	L3	L4
	Max Marks	5	12	15	18	-	-	10	18	22	-

RV Educational Institutions ®
RV College of Engineering ®

Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi

Approved by AICTE, New Delhi

Go, change the world

Academic year 2022-2023 (Even Sem)

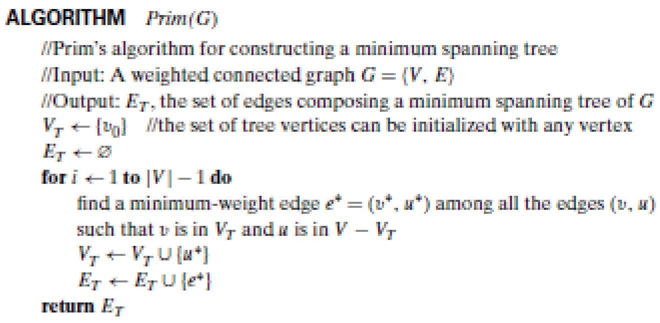
DEPARTMENT OF

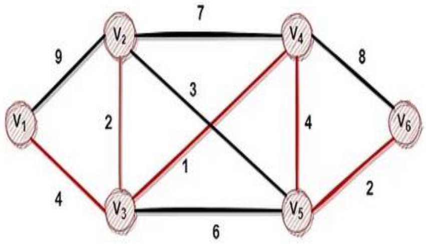
COMPUTER SCIENCE & ENGINEERING

Date	September 2023	Maximum Marks	50
Course Code	21CS43	Duration	90 Min
Sem	IV Semester	Improvement Test	

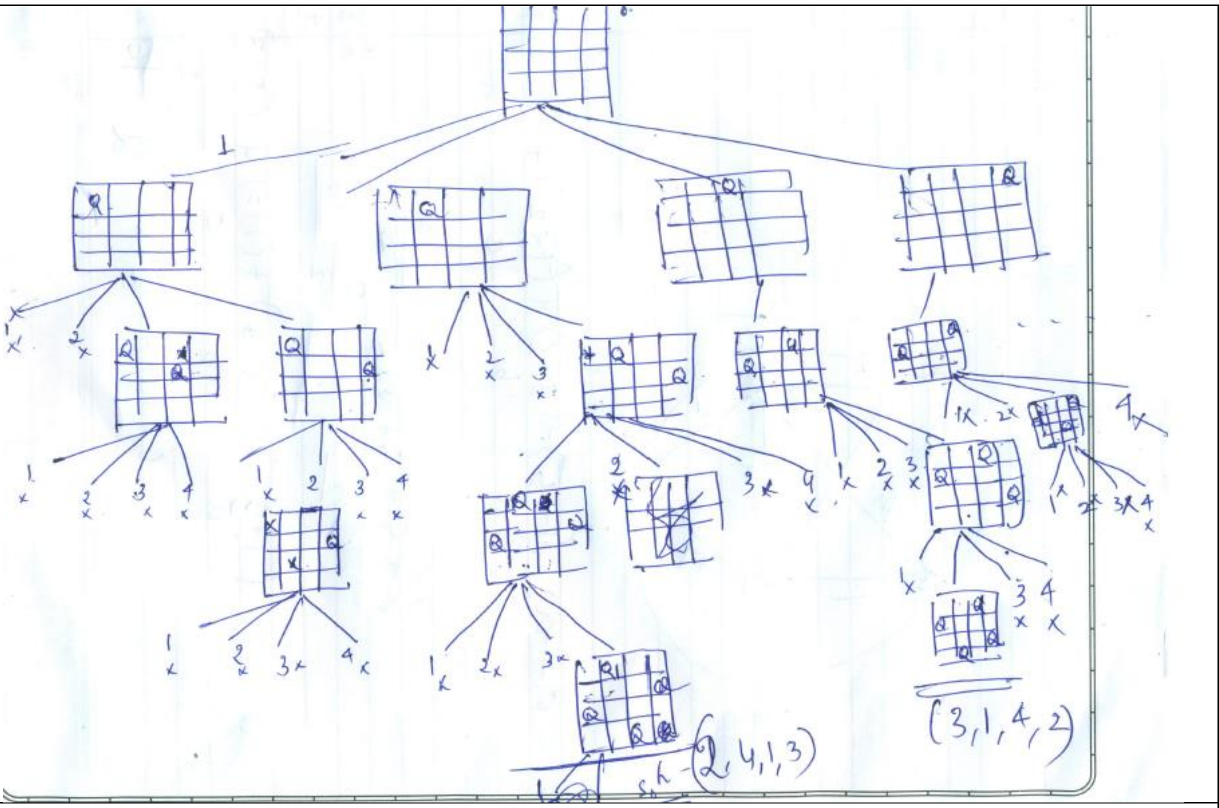
DESIGN AND ANALYSIS OF ALGORITHMS

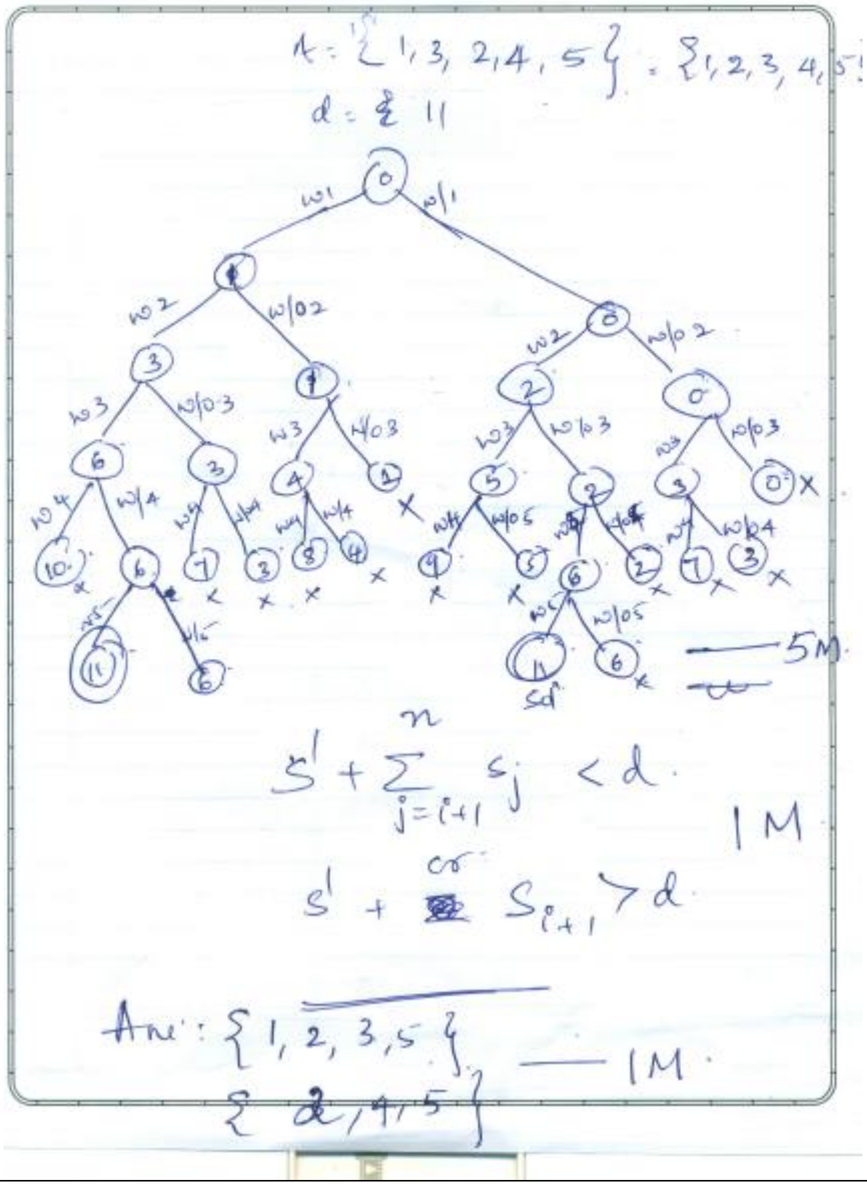
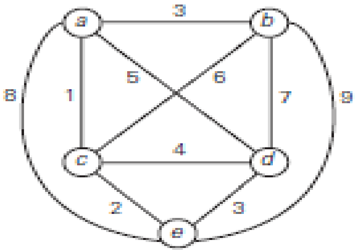
Sl. No.	Questions	M

1a.	Write the Algorithm to find minimum spanning tree using prim's algorithm-4M
	
	ALGORITHM *Prim(G)*
	//Prim's algorithm for constructing a minimum spanning tree
	//Input: A weighted connected graph $G = \langle V, E \rangle$
	//Output: E_T, the set of edges composing a minimum spanning tree of G
	$V_T \leftarrow \{v_0\}$ //the set of tree vertices can be initialized with any vertex
	$E_T \leftarrow \varnothing$
	for $i \leftarrow 1$ **to** $\lvert V \rvert - 1$ **do**
	  find a minimum-weight edge $e^* = (v^*, u^*)$ among all the edges (v, u)
	  such that v is in V_T and u is in $V - V_T$
	  $V_T \leftarrow V_T \cup \{u^*\}$
	  $E_T \leftarrow E_T \cup \{e^*\}$
	return E_T
		4

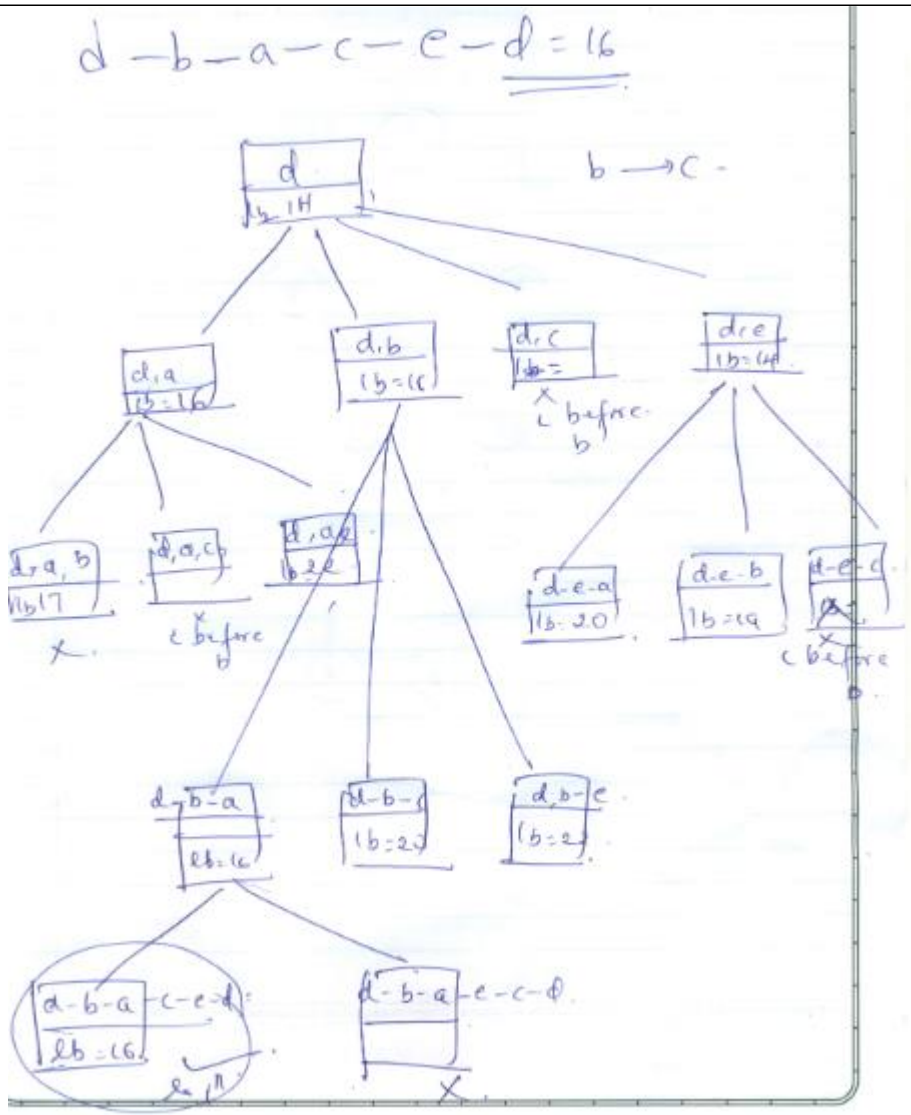
b	Solve the following instances of the single-source shortest-paths problem with vertex v_1 as the source:  V1 V2 V3 V4 V5 V6 9 7 8 3 2 4 1 4 2 6 Trace - 4M V1(-,-) V2(V1,9),V3(V1,4),V4(-,∞),V5(-,∞),V6(-,∞) V3(V1,4) V2(V3,6),V4(V3,5),V5(V3,10),V6(-,∞) V4(V3,5) V2(V3,6), V5(V4,9),V6(V4,13) V2(V3,6) V5(V4,9),V6(V4,13) V5(V4,9) V6(V5,11) V6(V5,11) Shortest Path & Cost- 1+1 V1-V3-V2=6 V1-V3=4 V1-V3-V4=5 V1-V3-V4-V5=9 V1-V3-V4-V5-V6=11	6
2.a	With the help of state space tree solve the 4-queens problem using Backtracking approach State space tree – 5M Solution -1M	6

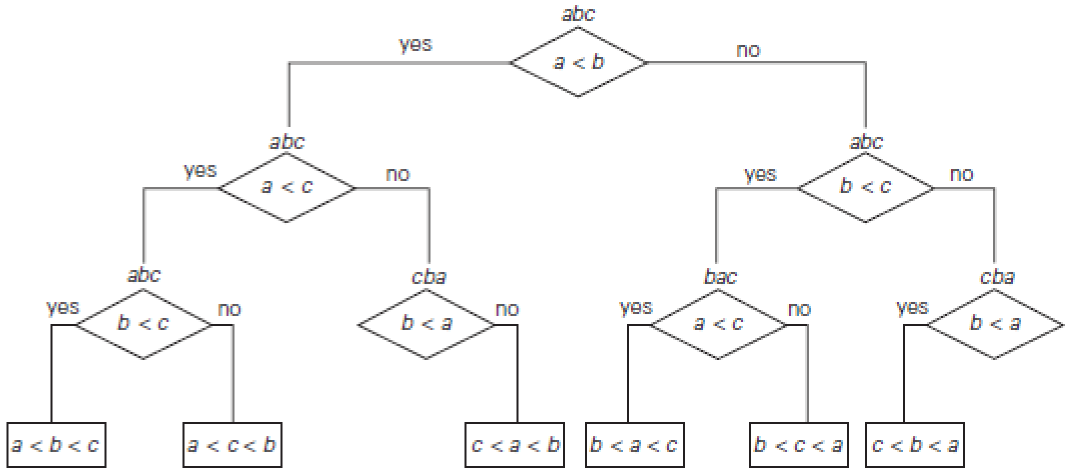
Academic year 2022-2023 (Even Sem)

		
b.	Differentiate Between the following: i) Optimal solution and Feasible solution : optimal solution is a feasible solution with the best value of the objective function a feasible solution is a point and Feasible solution in the problem's search space that satisfies all the problem's constraints ii) Promising node and Non-promising node Promising node corresponds to a partially constructed solution that may still lead to a complete solution otherwise, it is called non-promising 2+2	4
3 a.	General principle of backtracking design technique: The principal idea is to construct solutions one component at a time and evaluate such partially constructed candidates as follows. If a partially constructed solution can be developed further without violating the problem's constraints, it is done by taking the first remaining legitimate option for the next component. If there is no legitimate option for the next component, no alternatives for *any* remaining component need to be considered. In this case, the algorithm backtracks to replace the last component of the partially constructed solution with its next option.-3M	3
b	Apply backtracking to solve the following instance of the subset sum problem: $A = \{2,3,4,5\}$ and $d = 11$. Give the conditions used to terminate the node as non-promising. State space tree-5M Condition -2M	7

	 0 w1 w/1 1 0 w2 w/o2 3 1 2 0 w3 w/o3 6 3 4 1 5 2 3 0 w4 w/4 10 6 7 3 8 4 9 5 6 2 7 3 w5 w/5 11 6 11 6 x sd A: {1, 3, 2, 4, 5} . {1, 2, 3, 4, 5} d = {11 —— 5M $s' + \sum_{j=i+1}^{n} s_j < d$ 1M or $s' + S_{i+1} > d$ Ans: {1, 2, 3, 5} —— 1M {2, 4, 5}	
4	With the help of a state space tree. Solve the Travelling Salesman Problem for the following graph using branch and bound concept. Find the number of promising and non-promising nodes. Use vertex ‘d’ as starting node.  a b c d e 3 8 1 5 6 7 9 4 2 3 No of promising node -5 No ofnon- promising node -10 -----------2M Solution node-1 ------ Solution- d->b->a->c->e->d cost 16 1M State space tree -7M	10

Academic year 2022-2023 (Even Sem)

	 d – b – a – c – e – d = 16 d lb 14 b → c d, a lb=16 d, b lb=16 d, c lb= c before b d, e lb=14 d, a, b lb 17 d, a, c c before b d, a, e lb 22 d-e-a lb 20 d-e-b lb=19 d-e-c c before b d-b-a lb=16 d-b-c lb=22 d-b-e lb=2 d-b-a-c-e-d lb=16 d-b-a-e-c-d	
5 a.	Explain the following with an example. i) P Problem ii) NP Problem iii) NP-Complete Problem 1+2+2+M(Definition +Example)	5
b.	Give the Decision tree for the three-element selection sort 1+2+2=5M	5

	abc a < b yes no abc a < c yes no abc b < c yes no abc b < c yes no cba b < a no bac a < c yes no cba b < a yes a < b < c a < c < b c < a < b b < a < c b < c < a c < b < a	

Academic year 2022-2023 (Even Sem)

18CS43

USN

RV COLLEGE OF ENGINEERING®

(An Autonomous Institution Affiliated to VTU)

IVSemester B. E. Grade Improvement ExaminationsOctober 2021

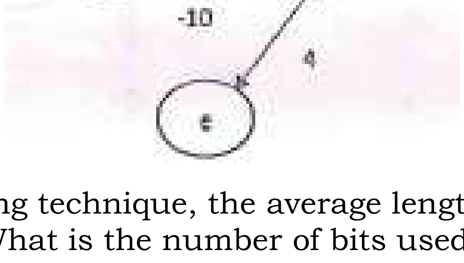
Common to CS / IS

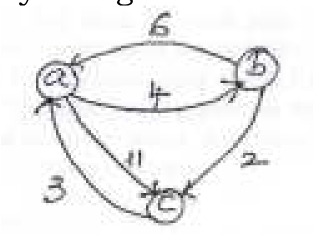
DESIGN AND ANALYSIS OF ALGORITHMS

Time: 03 Hours ***Maximum Marks: 100***

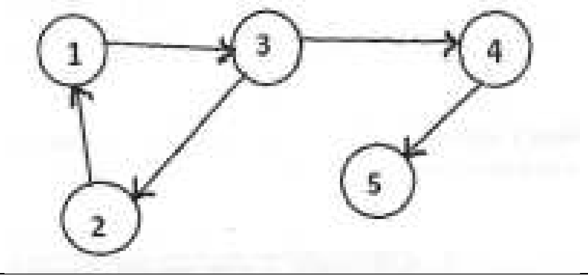
Instructions to candidates:

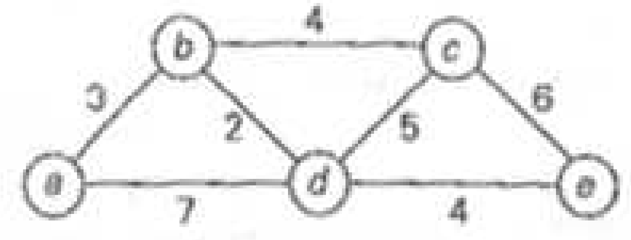
Answer any FIVE full questions out of TEN. Each carries 20 marks

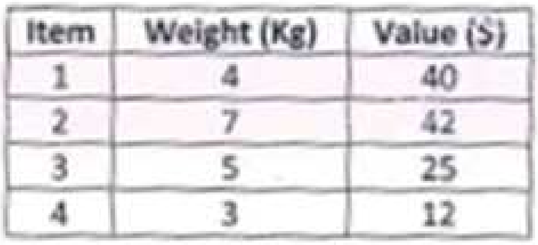
1	1.1	What type of problems can be solved using Dynamic Programming Strategy.	02
	1.2	How many times the basic operation gets executed while finding the value of binomial coefficient$C(6,5)$ using dynamic programming approach? Show tracing.	02
	1.3	Load shedding is necessary to maintain the system reliability in a decentralization grid system, which is vulnerable to power supply shortage. A proposed model considers the customer values and load characteristics to meet the electricity requirement and aims to maximize the customer values within given supply capacity. Which is the best-known algorithm and the algorithm design strategy to that can be used to solve this puzzle to find the best optimized solution capturing the customer values and discrete characteristics of load?	02
	1.4	X company has the largest decentralized communication protocol P2P for sharing files and data of large size over the Internet. P2P communication protocol has the world's highest Internet traffic each day. It offers clients to download files for windows , Mac and Android OS. It offers feature rich, safe programs for desktop, a browser based streaming and downloading product, and a mobile downloader for android is available. Which known algorithm and the design strategy best suits to solve the above puzzle?	02
	1.5	What is the worst and average time complexity of Boyer-Moore algorithm for a pattern of length m and a text of length n?	02
	1.6	A person wants to visit different places in India. He wants to visit from North to South part and list the places. But there are some places which he wants to visit before some other places. How can this problem be solved as an application of graph?	02
	1.7	How many swaps are made by insertion sort to sort(20,10,30,15,16,25) in ascending order? Show tracing.	02
	1.8	Given the graph find the shortest path using Dijkstra's algorithm from source'a'. 	02
	1.9	Using Huffman encoding technique, the average length of each character is found to be 2.25bits. What is the number of bits used to represent 50 characters? Show the calculations.	02

	1.10	Find the total number of changes(Change in element value) to graph given below after 2 iterations of Floyd's algorithm. Show the iterations.  	02

2	a	Analyze the algorithm for time complexity.
		ALGORITHM findme(A[0..n-1])
		//Input: Array a[0..n-1] of numbers
		temp←∞
		for i←0 to n-1 do
		    for j←0 to n-1 do
		        if i≠ j and $\|A[i] - A[j]\| < temp$
		            temp←$\|A[i] - A[j]\|$
		return temp
		can the efficiency be improved? If so, propose an algorithm and find its time complexity.
			12

	b	Apply bubble sort on the following input and count the number of comparisons done. Show complete passes and comparisons: 21,10,15,88,95,5.	08
3	a	Write a recursive function to solve the tower of Hanoi pruzzle. Analyze its running time efficiency. Show all the steps in analysis.	12
	b	Apply selection sort on the following input and count the number of comparison done A L G O R I T H M S	08
4	a	Write the algorithm to traverse a graph using depth-first search traversal. What is its time efficiency. ?	08
	b	A digraph is called strongly connected if for any part of two distinct vertices u and v there exists a directed path from u to v and a directed path from v to u . Design a DFS-based algorithm for identifying strongly connected components in a given graph. Apply the same to the following digraph to determine its strongly connected components.  	12
5	a	Write the pseudo code of merge sort algorithm(along with the merge) and analyses for worst case time efficiency.	10
	b	Compare the standard Brute force algorithm and the divide and conquer algorithm for matrix multiplication.	10

6	a b	Consider a university endowment that needs to invest $100 million. This sum must be split between three types of investments: stocks, bonds, and cash. The endowment managers expect an annual return of 10%,7% and 3% for their stock, bond and cash investments , respectively. Since stocks are riskier than bonds, the endowment rules require the amount invested in stocks to be no more than one third of the money is invested in bonds. In addition, at least 25% of the total amount invested in stocks and bonds must be invested in cash. Design a transform and conquer based solution to the above problem to find how the mangers can invest the money to maximize the return. Apply Horspool's algorithm to find the pattern "BARBER" in the string. "JIM_SAW_ME_IN_THE_BARBERSHOP"	10 10
7	a b	Jealous husbands: There are n(n≥ 2) married couples who need to cross river. They have a boat that can hold no more than two people at a time. To complicated matters , all the husbands are jealous and will not agree on any crossing procedure that would put a wife on the same bank of the river with another woman's husband without the wife's husband being there too, even if there are other people on the same bank. Can they cross the river under such constraints? Design transform and conquer based solution to the above problem to find solution when n=2. Draw the state space tree for the solution. Apply Boyer Moore's Algorithm to find the pattern "BAOBAB" in the string. "BESS_KNEW_ABOUT_BAOBABS".	10 10
8	a b	Write the algorithm for computing binomial coefficient C(n,k) using dynamic programming approach. Draw the binomial coefficient table for C(8,3). Apply Dijkstra's algorithm to find single source shortest paths from source vertex 'a'. 	10 10
9	a b	There are n houses build in a line, each of which contains some value in it. A theif is going to steal the maximal values of these houses, but he can't steal in two adjacent houses because the owner of the stolen houses will tell his two neighbours left and right side. Given the number of houses (n) and a list of n values, write a recurrence relation to sove the given problem. Write an Algorithm for the identifiedrecurrence relation to find the maximum stolen value. Trace your algorithm for input: val [ ]={6,7,1,3,8,2,4} What are the advantages of memory function? Write a memory function method to solve the knapsack problem.	12 08

<table>
<tr><td>10</td><td>a</td><td>What are the conditions in Branch and Bound technique to terminate a search path in state space tree? Apply Branch and Bound approach to solve the 0/1 Knapsack problem with the instance:
Knapsack capacity=10kg

<table>
<tr><th>Item</th><th>Weight (Kg)</th><th>Value ($)</th></tr>
<tr><td>1</td><td>4</td><td>40</td></tr>
<tr><td>2</td><td>7</td><td>42</td></tr>
<tr><td>3</td><td>5</td><td>25</td></tr>
<tr><td>4</td><td>3</td><td>12</td></tr>
</table></td><td>12</td></tr>
<tr><td></td><td>b</td><td>What are NP,P,NP-complete and NP-Hard problems? Give examples for each.</td><td>08</td></tr>
</table>