USN | | | | | | | | | |

CMR INSTITUTE OF TECHNOLOGY, BENGALURU. ACCREDITED WITH A+ GRADE BY NAAC

Solution of Internal Assessment Test III – JULY. 2021

Sub:	Design & Analysis of Algorithms				Sub Code:	18CS42	Branch:	CSE
Date:	30/07/2021	Duration:	60 min's	Max Marks: 50	Sem/Sec:	4/A,B,C & D		OBE

		MARKS	CO	RBT
1a	0/1 Knapsack can be never be solved one of the following strategies: a) Dynamic programming b) Greedy approach c)Brute-force enumeration d)Backtracking	1	CO3	L2
1b	Bellman-Ford's algorithm to find shortest paths in a graph is considered dynamic programming because: a) It needs the distance to the same vertex multiple times b)It needs the distance to multiple vertices at the same time c)It uses recursion d)It does not use recursion	1	CO3	L2
1c	An algorithm to find the factorial of a number will NOT benefit from dynamic programming technique because: a)It does not involve recursion b)It involves recursion c) It does not involve repeated recursion with the same argument d)It has a terminating condition	1	CO3	L2
1d	If dynamic programming is not used, then the best case complexity of 0/1 Knapsack is: a) O(2^n) b)O(n) c)O(n^2) d)O(n^3)	1	CO3	L3
1e	Sub-tours must be avoided while searching for the best solution to a Traveling Salesman problem because: a)All sub-tours cannot be found b)Some sub-tours are not possible to cover c)Travelling Salesman problem considers only some sub-tours, not all d)Travelling Salesman problem considers only complete tours, not sub-tours	1	CO3	L2
1f	Consider a travelling salesman problem with 4 cities. The distance between the cities are as shown below: A B C D A 0 10 15 20 B 10 0 35 25 C 15 35 0 30 D 20 25 30 0 What is the cost of the least cost Hamiltonian Cycle through these cities? a)65 b)80 c)70 d)35	1	CO3	L3

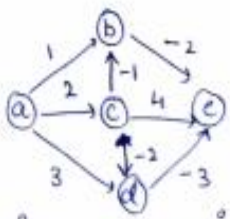
1g	What will be the complexity of Floyd's algorithm if the dijkstra's algorithm is performed again repeatedly for each vertex? a) O(V4) b)O(V) c)O(VE) d)O(VE+log V)	1	CO3	L2
1h	What happens in Dkij when the value of k is 1 in Floyd's algorithm. a)0 intermediate vertex b)N intermediate vertex c)N-1 intermediate vertex d)1 intermediate vertex	1	CO3	L2
1i	Choose the appropriate problem for the multistage graph from the list below. a)Travelling salesman b)0/1 Knapsack c)Resource allocation d)Assignment problem	1	CO3	L3
1j	How many solutions are there for 8 queens on an 8 x 8 board? a) 64 b) 91 c) 92 d) 93	1	CO3	L3
2a	11. True or False The worst case complexity of backtracking is the same as exhaustive search a) True b)False	1	CO2	L2
2b	Which one of the following is a correct option that provides an optimal solution for 4-queens problem? a) (3,1,4,2) b) (2,3,1,4) c) (4,3,2,1) d) (4,2,3,1)	1	CO3	L3
2c	The complexity of a recursive solution of the subset sum problem is: a) exponential b) linear c) logarithmic d) quadratic	1	CO2	L2
2d	One of the following squares need not be checked for obstruction, when we are looking to place a queen in the 3rd row, 2nd column of the 4-Queens problem: (1, 1) (2, 1) (1, 4) (4, 3)	1	CO3	L3

<table>
<tr><td>2e</td><td>The best-case complexity of Warshall's algorithm to find Transitive Closure of a graph is:

O(n3)
Theta(n3)
Omega(n3)
Any of the above</td><td>1</td><td>CO2</td><td>L3</td></tr>
<tr><td>2f</td><td>Consider the definition of a function F:
F(n) = F(n-1) + F(0) and
F(0) = 1.
Will the use of dynamic programming to implement the function F reduce the time complexity as compared to a recursive implementation? Explain.

Explanation with justification------------5M

Answer:
Though recursion is used, it is not repetitive. For instance, F(10) calls F(9) and F(9) calls F(8) and so on. However, F is called once per unique integer. Hence, even though dynamic programming can be used to implement the function, it does not significantly reduce the time complexity, since it is useful only when recursion is used multiple times with the same argument, as in the case of Fibonacci series. In this case, both iterative and recursive versions have O(n) as complexity.</td><td>5</td><td>CO3</td><td>L4</td></tr>
<tr><td>3a</td><td>Apply Bellman-Ford's algorithm for shortest paths on the graph below and explain how dynamic programming is useful in reducing the time taken, in comparison to a recursive implementation.
Problem solving using Bellman-Ford's algorithm----3M
Explanation ----------2M

Answer:
<table>
<tr><td>k\n</td><td>a</td><td>b</td><td>c</td><td>d</td><td>e</td></tr>
<tr><td>0</td><td>0</td><td>∞</td><td>∞</td><td>∞</td><td>∞</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>2</td><td>3</td><td>∞</td></tr>
<tr><td>2</td><td>0</td><td>1</td><td>2 [illegible]</td><td>0 [illegible]</td><td>-1</td></tr>
<tr><td>3</td><td>0</td><td>1</td><td>2</td><td>0</td><td>-1</td></tr>
<tr><td>4</td><td>0</td><td>1</td><td>2</td><td>0</td><td>-1</td></tr>
</table>


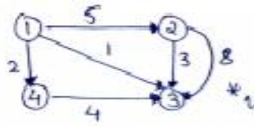
Dynamic programming for Bell-man Ford's.

Dynamic programming uses a bottom up approach by solving the subproblems to obtain an optimal solution for overlapping subproblems. It uses a technique called memoization to store the values for solved subproblem which then don't have to be re-computed, whereas in recursion we solve similar subproblem each time we encounter them, which increase the time complexity for the recursion.</td><td>5</td><td>CO3</td><td>L3</td></tr>
</table>

3b	Apply All Pair Shortest Paths algorithm to find the shortest paths for the graph below.	5	CO3	L3

Finding the optimal tour for a Travelling Salesperson for the given graph------5M

Answer:



all pairs shortest paths

*using undirected graph

$$d_{ij}^{k} = \min\{d_{ij}^{k-1}, d_{ik}^{k-1} + d_{kj}^{k-1}\}$$

$$D^0 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & 1 & 2 \\ 2 & 5 & 0 & 3 & \infty \\ 3 & 1 & 3 & 0 & 4 \\ 4 & 2 & \infty & 4 & 0 \end{array} \xrightarrow{\text{selecting (1) as intermediary vertex}} D^1 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & 1 & 2 \\ 2 & 5 & 0 & 3 & 7 \\ 3 & 1 & 3 & 0 & 3 \\ 4 & 2 & 7 & 3 & 0 \end{array}$$

selecting (2) as intermediary vertex

$$D^2 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & 1 & 2 \\ 2 & 5 & 0 & 3 & 7 \\ 3 & 1 & 3 & 0 & 3 \\ 4 & 2 & 7 & 3 & 0 \end{array} \xrightarrow{\text{selecting (3) as intermediary vertex}} D^3 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 4 & 1 & 2 \\ 2 & 4 & 0 & 3 & 6 \\ 3 & 1 & 3 & 0 & 3 \\ 4 & 2 & 6 & 3 & 0 \end{array}$$

selecting (4) as intermediary vertex

$$D^4 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 4 & 1 & 2 \\ 2 & 4 & 0 & 3 & 6 \\ 3 & 1 & 3 & 0 & 3 \\ 4 & 2 & 6 & 3 & 0 \end{array}$$

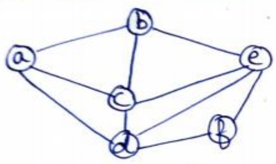
is the matrix of all pairs shortest path using Floyd's algorithm.

4a	Consider a graph G = (V, E) shown in fig. Find a Hamiltonian circuit using the Backtracking method.	5	CO3	L3

Finding a Hamiltonian circuit using the Backtracking method for the given graph-------5M

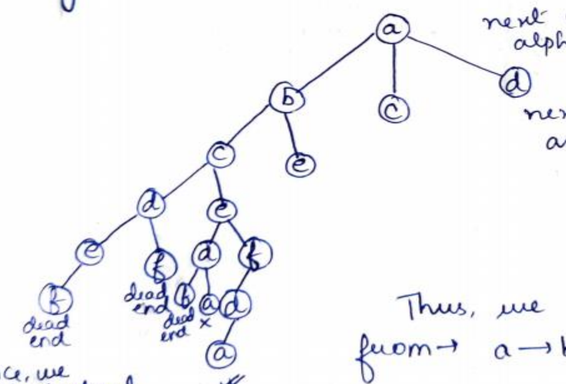
Answer:

1] Hamiltonian Circuit



G = (V, E)

Taking node (a) as starting point



next adding neighbours of 'a' alphabetically

next add neighbours of 'b' and so on, alphabetically

Thus, we find a Hamiltonian cycle from → a → b → c → e → f → d → a

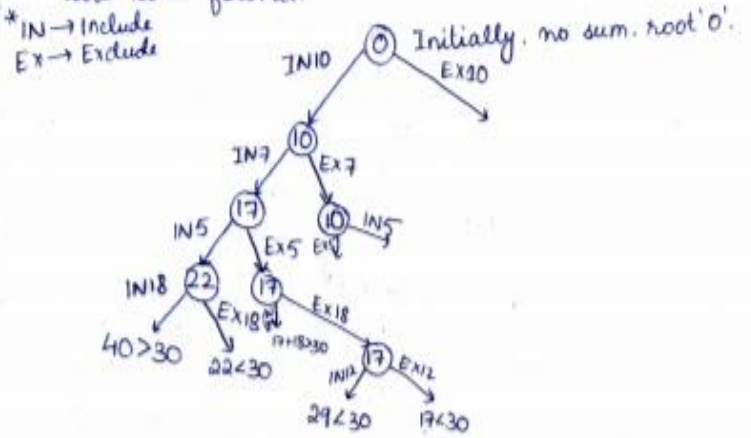
Since, we reach deadend, we go back to previous decision & make new decision.

<table>
<tr>
<td>4b</td>
<td>Apply backtracking technique to solve the below instance of the subset sum problem S= {10, 7, 5, 18, 12, 20, 15}, d= 30. Show the solution by drawing a state space tree.

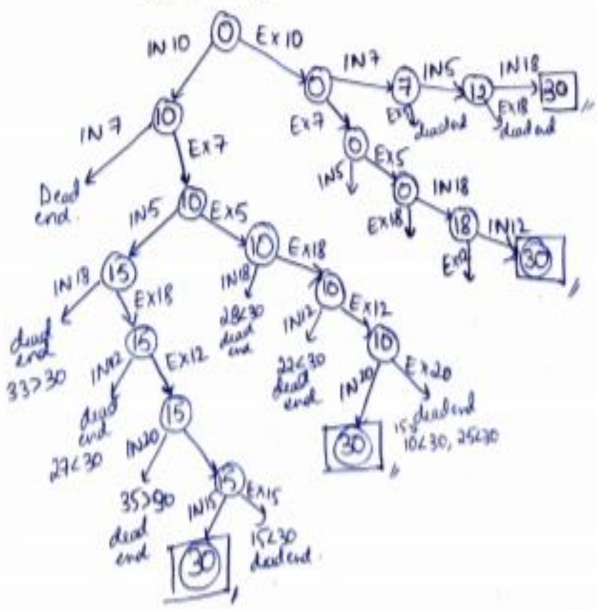
Solving problem----------3M
Drawing a state space tree--------2M

Answer:

$S = \{10, 7, 5, 18, 12, 20, 15\}$
$d = 30$ $\{10, 20\}$ $\{18, 12\}$ $\{7, 5, 18\}$ $\{10, 5, 15\}$
using state space tree, we encounter deadend in 2 cases
1] when sum of subset $< d$
2] when sum of subset $> d$
Thus, if our partial solution element's sum is equal to d at that time search terminates, or continues if all solutions are to be found.



Continuing, removing (17) node for space.

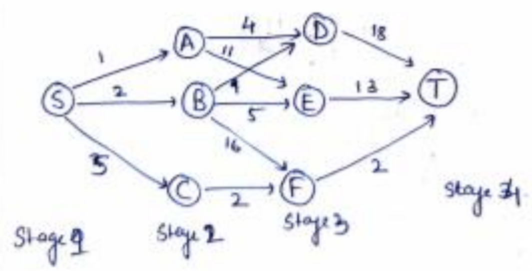
</td>
<td>5</td>
<td>CO3</td>
<td>L3</td>
</tr>
</table>

5

Determine the minimum cost path from source (S) to sink (T) for the graph using forward approach. Show all your work.

Solving problem to find minimum cost path for the given graph--------5M

Answer:



	S	A	B	C	D	E	F	T
Cost	9	22	18	4	18	13	2	0
Destination	C	D	E/F	F	T	T	T	T

$cost(i,j) = min(cost(j,l) + cost(i+1,l))$

stages, vertex, middle vertex

Stage 3 :- vertices → D, E, F

$cost(3,D) = min(cost(D,T) + cost(4,T))$
$= 18$

$cost(3,E) = 13$ } direct edges.

$cost(3,F) = 2$

Stage 2 :- vertices → A, B, C

$cost(2,A) = min\{[cost(A,D) + cost(3,D)], [cost(A,E) + cost(3,E)]\}$
$= min\{[4+18], [11+13]\} = \underline{\underline{22}}$

B has minimum to E or F as 18

$cost(2,B) = min\{[cost(B,D) + cost(3,D)][cost(B,E) + cost(3,E)]\}$
$= min\{[9+18], [5+13]\} = \underline{18}$

$cost(2,B) = cost[B,F) + cost(3,F)]$
$= 16 + 2 = \underline{18}$

$cost(2,C) = cost[(C,F) + cost(3,F)]$
$= 2 + 2 = 4$

Stage 1 :-

$cost(1,S) = min\{[cost(S,A) + cost(2,A)], [cost(S,B) + cost(2,B)], [cost(S,C) + cost(2,C)]\}$
$= min\{[1+22], [2+18], [5+4]\}$
$= 9$

hence, the path of minimum cost from source to sink is of $\underline{9}$ & is S → C → F → T.

10

CO3

L3