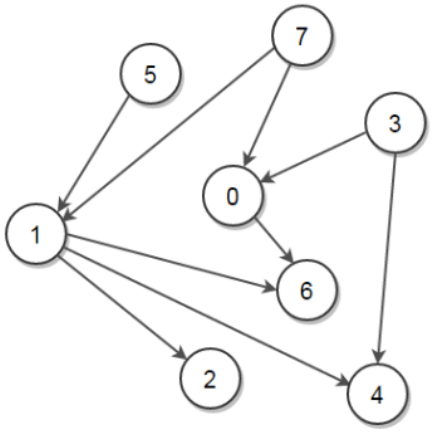
BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
Work Integrated Learning Programmes (WILP) Division
End-Semester Examination

Course No :
Course Title : Data Structure and Algorithm Design
Nature of Exam : Open Book
Weightage : 40
Duration :
Date of Exam :

No. of page = 2

Instructions :

1. Please read and follow all the instructions given on the cover page of the answer booklet.
2. Start each answer from a fresh page. All parts of a question should be answered consecutively.
3. Please ensure that your answers cover necessary technical details, avoiding unnecessary text and diagrams.

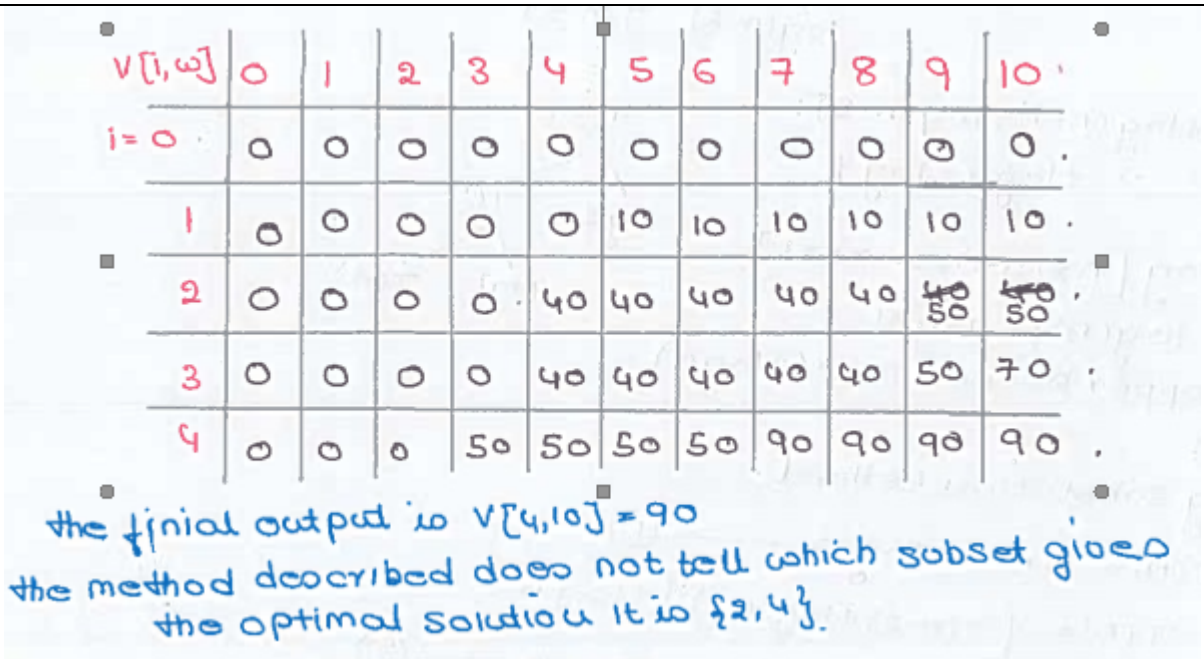
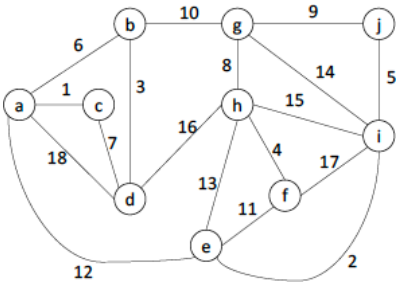
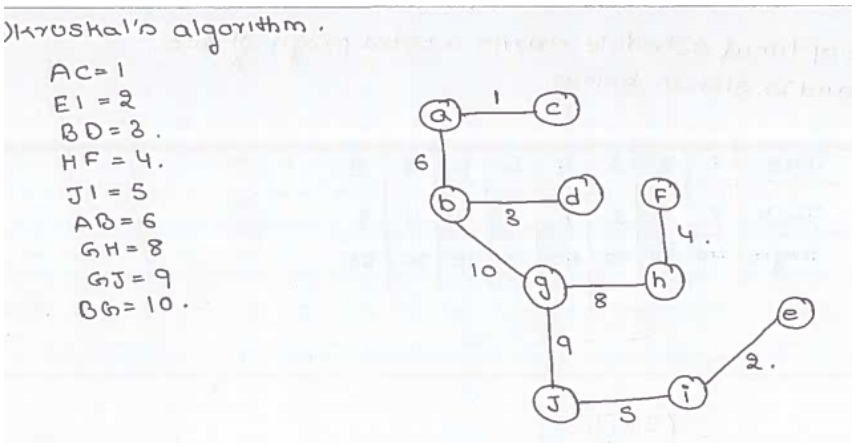
1	For each of the following recurrences, give an expression for the runtime T (n) if the recurrence can be solved with the Master Theorem. Otherwise, indicate that the Master Theorem does not apply. a. $T(n) = T(n/2) + 2^n$ b. $T(n) = 16T(n/4) + n$ c. $T(n) = 2^nT(n/2) + n^n$	4 M
	Solution. **2+2+2** **Can be considered for partial credits if final answer is not present/incorrect.** **Identifying which case of Master theorem applies-1M** **T(n)-1M** a)T (n) = Θ(2^n) (Case 3) b)T (n) = Θ(n^ 2) (Case 1) c) Does not apply (a is not constant)	
2	 Identify 4 different topological ordering for the below given graph	2 M
3	Given a schedule containing lecture sessions of different subjects for different courses in a college, a) **design an algorithm** (not program) to find minimum number of classrooms needed to conduct all these classes on time so that an optimal utilization of the classrooms can be made which may make maintenance work of classrooms easy. **b) Trace your algorithm** for the below given input c) **Calculate the minimum number of classrooms required for the same.** Input – Scheduled Lecture Sessions: Subject 1: 9:00 – 10:00 Subject 2: 9:30 – 10:30 Subject 3: 10:15 – 11:15 Subject 4: 9:45 – 10:45 Subject 5: 10:45 – 11:45	4+ 3+1+1 M

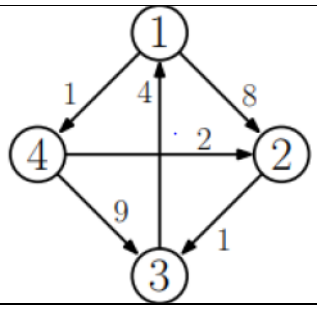
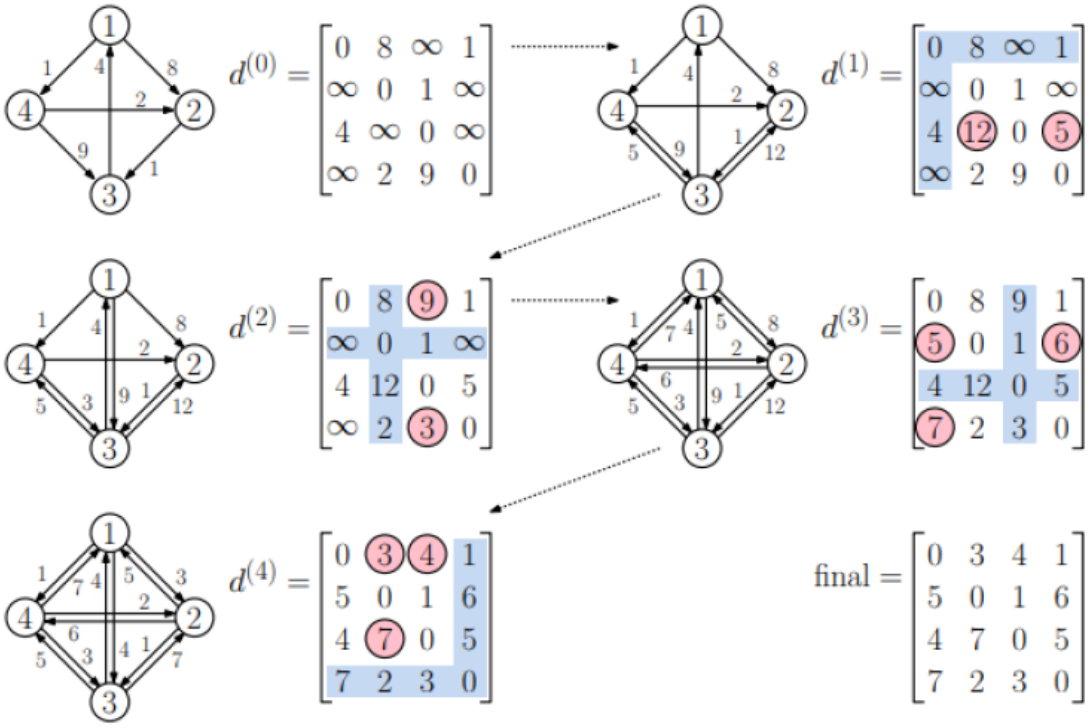
	Subject 6: 11:00 – 12:00 Subject 7: 11:15 – 12:15 Subject 8: 11:30 – 12:30 Subject 9: 11:30 – 12:30 What design strategy is used?	

Ans	Algortihm: Classroom_schdl()
	1. n = number of subjects.
	2. counter = 0, max = 0.
	3. Label all the start time of the session with a suffix 's'.
	4. sort all the timings irrespective of whether it is a start or end time in increasing order.
	5. If an end time of a session and start time of another session are same then put end time before the start time in the sorted list.
	6. go through all the sorted timings and update the counter and max variables -
	for (i = 0; i < 2n; i ++)
	If the time indicates start of the session
	{
	counter ++
	if (max < counter) then max ++
	}
	else
	counter - -
	7. return (max) Algorithm / pseudo-code _____ (6 Marks)
	
	Trace of the algorithm for the given Input:
	n = 9, counter = 0.
	Suffixing 's' for the start of sessions.
	9:00s – 10:00
	9:30s – 10:30
	10:15s – 11:15
	9:45s – 10:45
	10:45s – 11:45
	11:00s – 12:00
	11:15s – 12:15
	11:30s – 12:30
	11:30s – 12:30
	Sorting the timings irrespective of whether it's a start or end of session.
	{ 9:00s , 9:30s , 9:45s , 10:00 , 10:15s , 10:30 , 10:45 , 10:45s , 11:00s , 11:15 , 11:15s , 11:30s , 11:30s , 11:45 , 12:00 , 12:15 , 12:30 , 12:30 }
	counter = 1 , max = 1
	counter = 2 , max = 2
	counter = 3 , max = 3
	counter = 2 , max = 3
	counter = 3 , max = 3
	counter = 2 , max = 3
	counter = 1 , max = 3
	counter = 2 , max = 3
	counter = 3 , max = 3
	counter = 2 , max = 3
	counter = 3 , max = 3
	counter = 4 , max = 4
	counter = 5 , max = 5
	counter = 4 , max = 5
	counter = 3 , max = 5
	counter = 2 , max = 5
	counter = 1 , max = 5
	counter = 0 , max = 5
	return (5).
	i.,e minimum 5 class rooms required to conduct sessions in time. trace _____ (3 Marks)
	
	It's a greedy approach. (1 Marks)	

4	Fill out the bottom-up dynamic programming table for the 0-1 knapsack program. given a 10 kg sack and the following items Identify the selected items.	4 +1 M

I	1	2	3	4
V_i	10	40	30	50
W_i	5	4	6	3

		
5	Use Kruskal's algorithm, to find a minimum spanning tree of the given graph below. Draw the resulting spanning tree and list the edge in the order they are picked by the Kruskal's algorithm. 	4+1 M
		
7	Use an optimal algorithm to compute shortest paths between all pairs of vertices in the given graph, clearly showing all intermediate steps.	6 M

		
		
8	We want to store values between 0 to 9999 in a hash table of size 10.the hash function operates as follows: given a value x, add the four digits of x and take the last (right -most) digit. *a.* Insert, in the given order the values 3836,7209,2373,9412,6950,471,5569,9703 handle collusion using linear probing. Then, remove in the given order 3836,9412 and 5569. After every value deletion you have to rehash the existing values	5M

a)

0	3836
1	6950
2	471
3	
4	
5	2373
6	9412
7	5569
8	7209
9	9703.

b) Remove element final answer

0	6950
1	
2	471
3	
4	
5	2373
6	5569
7	
8	7209
9.	9703.