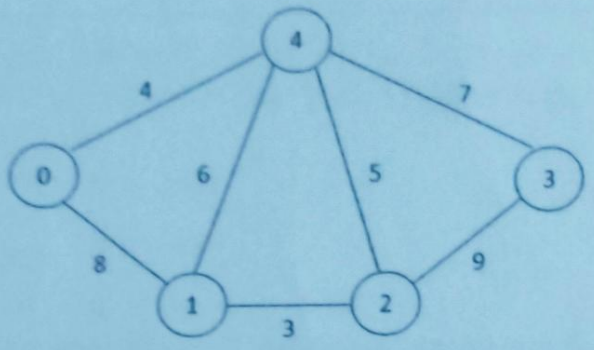
END SEMESTER ASSESSMENT (ESA) B.Tech. (CS&E) - May 2022

UE20CS251 - Design and Analysis of Algorithms

Time: 3 Hrs | Answer All Questions | Max Marks: 100

1.a)	Algorithm Foo(n) if(n=1) return 1 c ← Foo(n-1) c ← c+1 c ← c + Foo(n-1) return c Derive the asymptotic time efficiency of the above algorithm Foo in terms of Θ-notation.	6
1.b)	Using limits show that the order of growth of n^2 and n(n-1)/2 are same.	7
1.c)	Define O-notation, Ω-notation and θ–notation used in the asymptotic analysis of algorithms with appropriate graph sketches.	7
2.a)	Write a naive string matching algorithm to search for a pattern of length **m** in a text of length **n**. Mention the time complexity of the algorithm in Θ-notation.	6
2.b)	Design an algorithm to find the largest element in a non-empty array A[0..n-1] of numbers using the Decrease-and-Conquer approach. Derive its time complexity.	7
2.c)	Design a **Decrease-by-a-Constant-Factor** algorithm to search for a key in a sorted array A[0..n-1] of n real numbers. Derive its time complexity.	7
3.a)	Design an algorithm using each of the following design strategies for the problem of finding the largest element of a non-empty array A[0..n-1] of 'n' numbers. (i) Brute-Force: **Algorithm Largest_BruteForce(A[0..n-1])** (ii) Decrease-and-Conquer. **Algorithm Large_Dec_n_Conq(A[0..n-1])** (iii) Divide-and-Conquer: **Algorithm Large_Div_n_Conq(A[0..n-1])**	6
3.b)	Construct a heap for the following sequence of keys using the bottom-up approach of construction of heap that takes O(n) time. 2, 9, 7, 6, 5, 8, 10, 3, 6, 9	7
3.c)	Write the Merge Sort algorithm to sort an array A[0..n-1]. Write the algorithm in a	7

	pseudocode format.	
4.a)	(i) What is a spanning tree? (ii) What is a minimum spanning tree? (iii) Find the cost of a minimum spanning tree of the graph shown in the picture.  	6
4.b)	Construct the shift table used by the Horspool's algorithm for the pattern "**BARBER**" and explain searching the pattern in the following text. **JIM_SAW_ME_IN_A_BARBERSHOP**	7
4.c)	Find the shortest paths from the vertex **'a'** to all the other vertices using Dijkstra's single-source-shortest-paths algorithm for the graph below in the form of a weight matrix.	7
5.a)	Using dynamic programming strategy, design an algorithm to find the binomial coefficient using Pascal's identity given below (write the bottom-up version of the solution). $C(n,k) = C(n-1,k-1) + C(n-1,k)$ for $n > k > 0$ $C(n,0) = 1,\quad C(n,n) = 1$ for $n \geq 0$	6
5.b)	Define the classes of P, NP, and NP-complete problems.	7
5.c)	Write the O(n^3) Warshall's algorithm to find the transitive closure of a graph represented by an adjacency matrix A[1..n, 1..n], and find the transitive closure of the graph given below in the form of an adjacency matrix. a b c d a 0 1 0 0 b 0 0 0 1 c 0 0 0 0 d 1 0 1 0	7

3b.

Construct a heap for the following sequence of keys using the bottom-down approach of construction of heap that takes O(n) time.

2, 9, 7, 6, 5, 8, 10, 3, 6, 9.

4c.

Find the shortest paths from the vertex **'a'** to all the other vertices using Dijkstra's single-source-shortest-paths algorithm for the graph below in the form of a weight matrix.

	a	b	c	d	e
a	0	3	∞	7	∞
b	3	0	4	2	∞
c	∞	4	0	5	6
d	7	2	5	0	4
e	∞	∞	6	4	0