BIRLA INSTITUTE OF TECHNOLOGY, MESRA, RANCHI
(END SEMESTER EXAMINATION)

CLASS: MCA & PRE-PHD — SEMESTER : II
BRANCH: CSE — SESSION : SP/2023

SUBJECT: CA419 ANALYSIS OF ALGORITHMS

TIME: 3 Hours — FULL MARKS: 50

INSTRUCTIONS:
1. The question paper contains 5 questions each of 10 marks and total 50 marks.
2. Attempt all questions.
3. The missing data, if any, may be assumed suitably.
4. Before attempting the question paper, be sure that you have got the correct question paper.
5. Tables/Data hand book/Graph paper etc. to be supplied to the candidates in the examination hall.

			CO	BL
Q.1(a)	Is it reasonable to measure an algorithm's efficiency as a function of a parameter indicating the size of the algorithm's input? What can be such other potential parameters.	[5]	CO1	BL4
Q.1(b)	Solve the recurrence $T(n) = 2T(n^{(1/2)}) + 1$. Your solution should be asymptotically tight. Clearly specify the base condition you choose for solving this recurrence relation.	[5]	CO1	BL3
Q.2(a)	The minimum spanning tree problem is the problem of finding a minimum spanning tree for a given weighted connected graph. Outline an algorithm that constructs a minimum spanning tree through a sequence of expanding subtrees. The initial subtree in such a sequence consists of a single vertex selected arbitrarily from the set V of the graph's vertices. On each iteration, expand the current tree in a greedy manner by simply attaching to it the nearest vertex not in that tree. This algorithm should stop after all the graph's vertices have been included in the tree being constructed. Discuss the time complexity of this method.	[5]	CO2	BL3
Q.2(b)	Using suitable example(s) discuss the limitations of Dijkstra's algorithm for finding the shortest paths (from a given source vertex to all the remaining vertices) in a directed and weighted graph.	[5]	CO2	BL2
Q.3(a)	Give recurrence relation for recursive *binary search* algorithm and hence derive its average case complexity.	[5]	CO3	BL3
Q.3(b)	Demonstrate the steps involved in sorting the following data set using standard merge sort algorithm- ['Q', 'U', 'I', 'C', 'K', 'S', 'O', 'R', 'T']. How many time the merging of two intermediary subsets takes place?	[5]	CO3	BL2
Q.4(a)	Computing a binomial coefficient is a standard example of applying dynamic programming to a non-optimization problem. In combinatorics, the binomial coefficient, denoted C(n, k) is the number of combinations (subsets) of k elements from an n-element set ($0 \le k \le n$). Suggest an algorithm for computing the binomial coefficient C(n, k) by the dynamic programming algorithm. What is the time efficiency of this algorithm?	[5]	CO4	BL2
Q.4(b)	Give a dynamic programming solution for *all pairs shortest paths* problem.	[5]	CO4	BL2
Q.5(a)	Give an iterative solution to traversing a tree in infix manner. Discuss time the complexity of this approach	[5]	CO5	BL2
Q.5(b)	Differentiate the classes P, NP, and NPC. Using a Venn diagram depict how these classes are supposedly related to each other.	[5]	CO5	BL2

::::::27/04/2023::::::E