Reg. No. : | E | N | G | G | T | R | E | E | . | C | O | M |

Question Paper Code : 50008

B.E./B.Tech. DEGREE EXAMINATIONS, APRIL/MAY 2024.

Third Semester

Artificial Intelligence and Data Science

AD 3351 — DESIGN AND ANALYSIS OF ALGORITHMS

(Common to Computer Science and Business Systems)

(Regulations 2021)

Time : Three hours | Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. What is abstraction? Why do we need Abstract Data Type?

2. What is empirical analysis? How it is done?

3. Write an algorithmic steps for string-matching problem using brute-force technique.

4. For a digraph with n vertices, what is the largest number of distinct solutions, the topological sorting problem can have?

5. State the principle of optimality of dynamic programming.

6. Why the time efficiency class of Warshall's algorithm is inferior to that of the traversal-based algorithm for sparse graphs represented by their adjacency lists?

7. Why do we need slack and surplus variables in the LPP for solving using simplex method?

8. State the mathematical formulation of maximum flow problem.

9. How does a branch and bound algorithm work?

10. What do you mean by polynomial time approximation algorithm?

PART B — (5 × 13 = 65 marks)

11. (a) (i) Spotify, a digital music service platform employs algorithms to present you with music choices that will peak your interest and keep you on the platform. The general rule of thumb for determining if you like a song or not all depends on the first 30 seconds. If a user skips an unsaved song before the 30-second mark, Spotify counts that as a thumbs down. Identify the Finiteness, Definiteness, Generality and Effectiveness characteristics of an algorithm for this scenario. (6)

(ii) Obtain the time complexity of following recurrence relation in big-oh (O) notation $T(n)=2T(n/2)+n$ with $T(1)=1$ (7).

Or

(b) (i) Exemplify in detail about asymptotic notations and basic efficiency classes of algorithm. (7)

(ii) Design a recursive algorithm for computing 2^n for any nonnegative integer n that is based on the formula $2^n=2^{n-1}+2^{n-1}$. Set up a recurrence relation for the number of additions made by the algorithm and solve it. (6)

12. (a) (i) Consider the problem of finding the distance between the two closest numbers in an array of n numbers. (The distance between two numbers x and y is computed as $|x-y|$). Design a presorting-based algorithm for solving this problem and determine its efficiency class. (7)

(ii) Outline an exhaustive-search algorithm for the Hamiltonian circuit problem. (6)

Or

(b) (i) Illustrate in detail about heap Construction and heap sort algorithm. (7)

(ii) Construct and demonstrate the divide-and-conquer closest-pair algorithm. (6)

13. (a) (i) Write a pseudo code of the bottom-up-dynamic programming algorithm for the knapsack problem. (6)

(ii) Illustrate in detail about Huffman trees and codes. (7)

Or

(b) (i) Outline the Dijkstra's algorithm for solving single-source shortest-path problem for directed weight graphs. (6)

(ii) Write down the binary search algorithm and trace it with a key value of $k=57$ in the data set

$L=\{5,23,28,31,35,38,42,44,49,53,55,57,61,65,77,88,90,97,99\}$. (7)

14. (a) (i) Design a simple algorithm for checking whether a given marriage matching is stable and determine its time efficiency class. (7)

(ii) Explain how the maximum-flow problem for a network with several sources and sinks can be transformed into the same problem for a network with a single source and a single sink. (6)

Or

(b) (i) Prove that a matching M is a maximum matching if and only if there exists no augmenting path with respect to M. (7)

(ii) Solve the following linear programming problem geometrically (Graphically) (6)

Maximize $x+2y$

Subject to

$4x \geq y$

$y \leq 3+x$

$x \geq 0,\ y \geq 0$

15. (a) Solve the following instance of the Knapsack problem by the branch-and bound algorithm

Item	Weight	Value	
1	10	\$100	
2	7	\$63	W = 16
3	8	\$56	
4	4	\$12	

Or

(b) Compare and contrast between P, NP, NP-complete and NP hard problems with examples.

PART C — (1 × 15 = 15 marks)

16. (a) Let $A=\{a_1,...,a_n\}$ and $B=\{b_1,....,b_m\}$ be two sets of numbers. Consider the problem of finding their intersection, i.e., the set C of all the numbers that are in both A and B.

(i) Design a brute-force algorithm for solving this problem and determine its efficiency class (5)

(ii) Design a presorting-based algorithm for solving this problem and determine its efficiency class. (5)

(iii) Compare the efficiency of both the algorithms. (5)

Or

(b) Solve the all-pairs shortest-path problem for the digraph with the following weight matrix :

$$\begin{bmatrix} 0 & 2 & \infty & 1 & 8 \\ 6 & 0 & 3 & 2 & \infty \\ \infty & \infty & 0 & 4 & \infty \\ \infty & \infty & 2 & 0 & 3 \\ 3 & \infty & \infty & \infty & 0 \end{bmatrix}$$