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

Question Paper Code : 50467

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

Second/Third Semester

Computer Science and Design

CD 3291 – DATA STRUCTURES AND ALGORITHMS

(Common to : Computer Science and Engineering (Artificial Intelligence and Machine Learning)/Computer Science and Engineering (Cyber Security)/ Computer and Communication Engineering/Information Technology)

(Regulations 2021)

Time : Three hours | Maximum : 100 marks

Answer ALL questions.

PART A — ($10 \times 2 = 20$ marks)

1. What are commonly used built in classes for python?

2. State the properties of the Big-Oh notation.

3. What are referential arrays? Give example.

4. State the application of the double ended queues.

5. Why study sorting algorithm?

6. What is goal of hash function?

7. State the properties of binary trees.

8. Define multiway search trees.

9. What are the different ways to represent a graph?

10. State the components of a dynamic programming solution.

PART B — ($5 \times 13 = 65$ marks)

11. (a) (i) Distinguish between shallow and deep copying. (7)

(ii) Write a short Python function that takes a positive integer n and returns the sum of the squares of all the positive integers smaller than n. (6)

Or

(b) (i) What is objects? Explain the procedure of creating and using objects with example. (7)

(ii) Write a Python function that finds the minimum and maximum values in a sequence without using any loops. (6)

12. (a) (i) Distinguish between compact arrays and dynamic array. (7)

(ii) Describe how the built-in sum function can be combined with Python's comprehension syntax to compute the sum of all numbers in an $n \times n$ data set, represented as a list of lists. (6)

Or

(b) (i) Distinguish between singly linked lists and doubly linked list with example. (7)

(ii) What values are returned during the following series of stack operations, if executed upon an initially empty stack? push (5), push (3), pop( ), push(2), push (8), pop( ), pop( ), push(9), push(1), pop( ), push(7), push(6), pop( ), pop( ), push(4), pop( ), pop( ). (6)

13. (a) (i) State and explain the array-based implementation of Merge-sort with example. (7)

(ii) Show that the best-case running time of quick-sort on a sequence of size n with distinct elements is $\Omega\ (n \log n)$. (6)

Or

(b) State and explain different collision-handling schemes with example. (13)

14. (a) (i) Explain the various tree traversal with example. (7)

(ii) Draw the binary tree representation of the following arithmetic expression : (6)

"$(((5 + 2) * (2 - 1))/((2 + 9) + ((7 - 2) - 1)) *8)$".

Or

(b) (i) Discuss about the operation on AVL tree with example. (7)

(ii) Implement the in-place heap-sort algorithm. Experimentally compare its running time with that of the standard heap-sort that is not in-place. (6)

15. (a) (i) State and explain the different graph traversals with suitable example. (7)

(ii) Explain topological ordering with example. (6)

Or

(b) Draw a simple undirected graph G that has 12 vertices, 18 edges and 3 connected components. Draw an adjacency matrix representation of the undirected graph. Draw an adjacency list representation of the undirected graph.

PART C — (1 × 15 = 15 marks)

16. (a) (i) What values are returned during the following sequence of queue operations, if executed on an initially empty queue? enqueue(5), enqueue(3), dequeue( ), enqueue(2), enqueue(8), dequeue( ), dequeue( ), enqueue(9), enqueue(1), dequeue( ), enqueue(7), enqueue(6), dequeue( ), dequeue( ), enqueue(4), dequeue( ), dequeue( ). (8)

(ii) Implement a function that reverses a list of elements by pushing them onto a stack in one order and writing them back to the list in reversed order. (7)

Or

(b) (i) Consider the sequence of keys (5, 16, 22, 45, 2, 10, 18, 30, 50, 12, 1). Draw the result of inserting entries with these keys (in the given order) into an initially empty red-black tree. (8)

(ii) Design an efficient algorithm for the matrix chain multiplication problem that outputs a fully parenthesized expression for how to multiply the matrices in the chain using the minimum number of operations. (7)

———————