

[Total No. of Questions - 9] [Total No. of Printed Pages - 2]

May-24-0377

CS-301 (Data Structures) [CSE, IT]
B.Tech. 3rd (CBCS)

Time : 3 Hours

Max. Marks : 60

The candidates shall limit their answers precisely within the answer-book (40 pages) issued to them and no supplementary/continuation sheet will be issued.

Note : Each question carries 10 marks. Attempt one question each from section ABCD. Section E is compulsory and carries 20 marks.

SECTION - A

1. What is the difference between primitive, derived and abstract data types? Also give the difference between linear and nonlinear data structures with example. (10)
2. Why do you think that an array is classified as a data structure? Write two different algorithms to delete duplicate numbers from a linear array. In one of your algorithms, space complexity should be constant. What are the time complexities of both of your algorithms? (10)

SECTION - B

3. What do you mean by Stack? Write algorithms for PUSH and POP operations while implementing a stack using an array. How will you implement two Stacks using a single array (in most efficient way)? Write algorithms for defining PUSH1 and POP1 for stack1 and PUSH2 and POP2 for Stack2. (10)
4. Write algorithms to implement a LINEAR queue using both array and linked list. Don't forget to consider overflow and underflow conditions for both implementations. Do you need a circular queue while implementing it with linked list. If yes, then provide the application of such a queue, if no, then explain why? (10)

2

CS-301

SECTION - C

5. What is a binary tree? Write an algorithm for preorder traversal and convert the following algebraic expression " $5 * (6+2) - 12/4$ " into prefix notation using the same. (10)
6. Define Graph. What is adjacency matrix and discuss various adjacency matrix techniques to implement directed graph, undirected graph and a weighted graph? Which type of matrix will be created while implementing an undirected graph? Can an adjacency matrix be created for a weighted multigraph? (10)

SECTION - D

7. Write an algorithm for Quick Sort. What is its complexity for both worst and average case? Also sort C, O, M, P, U, T, E, R in increasing order using Quick Sort. (10)
8. Explain Hashing. Define some hash functions like folding method. How is collision resolved using quadratic probing? (10)

SECTION - E (Compulsory)

9. Write short note on following: (any five)
 - (i) When will you say an algorithm is efficient? Give the notations for time complexity.
 - (ii) What is balanced and unbalanced partitioning in Quick Sort?
 - (iii) What is a left and right skewed binary search tree? Why you do not prefer them?
 - (iv) Is it possible to apply binary search on a linked list?
 - (v) Write the role of a stack in 'function' call.
 - (vi) Radix Sort. (5*4=20)