Course : B.Tech. **Branch :** ECE **Year/Semester :** I/II **Academic Year :** 2018-19

Subject : Data Structures **AdmittedBatch :** 2018

QUESTION BANK

UNIT-I: ARRAYS

1. A) Define data structure. Discuss different types of data structure their Implementations applications. [5 M]
   B) What is an array? Discuss different types of array with examples. [5 M]
2. A) Explain how to implement polynomial ADT using array. Discuss its Advantages And Disadvantages. [5 M]
   B) Explain polynomial addition using arrays [5 M]
3. A) Explain sparse matrix representation using array with an example. Discuss the Advantage and disadvantages of this method. [5 M]
   B) Discuss matrix multiplication with an example [5 M]
4. A) Explain in detail about transpose of matrix with example? [5 M]
   B) illustrate about polynomial representation along with ADT? [5 M]

UNIT-II: STACKS AND QUEUES

1. A) Write an algorithm to insert and delete a key from circular queue. [5 M]
   B) Explain the procedure to convert infix expression to postfix expression with the Following expression: ((A –(B+C) * D) / (E+F)) [5 M]
2. A) Explain the evaluation of prefix expression. Find the equivalent prefix of :8 6 3 + * 1 2 3 -/- [5 M]
   B) Explain basic operations of queue. List the steps to implement queue using stack. [5 M]
3. A) Explain the operations performed on simple queue with an example. [5 M]
   B) Convert following expression X+(Y * Z) – ((N * M +O) /P) in to post form. [5 M]
4. A) Write an algorithm for basic operations of stack. [5 M]
   B) Explain the procedure to evaluate postfix expression. Evaluate the following Postfix expression 7 3 4 + - 2 4 5 / + * 6 / 7 +? [5 M]

UNIT-III: LINKED LISTS

1. A) Write recursive algorithm for lists. [5 M]
   B) Explain the procedure to insert and delete element from sparse matrix. [5 M]
2. A) Write an algorithm to push and pop an element from linked stack. [5 M]
   B) Discuss sparse matrix representation using linked list. [5 M]
3. A) Write an algorithm to delete an element anywhere from doubly linked list. [5 M]
   B) Write applications of single linked list to represent polynomial expressions [5 M]
4. A) List various operations of linked list and explain how to insert a node anywhere in

The list. [5 M]

B) Show how to reverse a single linked list. [5 M]

UNIT-IV: TREES

1. A) Explain binary tree ADT. [4 M]
   B) Discuss representation of binary tree using arrays and linked list. [6 M]
2. A) What operations can be performed on binary trees? Discuss. [4 M]
   B) Write in-order, pre-order and post-order traversal of a binary tree. [6 M]
3. A) Construct max heap for the following: [5 M]
   140, 80 , 30 , 20 ,10 ,40 ,30 ,60 ,100 ,70 ,160 ,50 , 130, 110, 120
   B) Explain in-order traversal of threaded binary tree with an example. [5 M]
4. A) Define binary search tree. Show how to insert and delete an element from binary Search tree with an example? [6 M]
   B) Write algorithm to insert and delete an element from binary search tree. [4 M]

UNIT-V: GRAPHS

1. A) What is a graph? Explain the properties of graphs. [4 M]
   B) Write breadth first traversal algorithm. Explain with an example [6 M]
2. A) What are connected components of graph? Is there a method to find out all the Connected components of graph? Explain. [4 M]
   B) Explain Prim's algorithm with an example. [6 M]
3. A) Discuss kruskal's algorithm with an example. [6 M]
   B) Explain how to represent a graphs. [4 M]
4. A) Explain Warshall's algorithm to find transitive closure of a graph with a suitable Example. [5 M]
   B) Explain All pairs shortest path with example? [5 M]

UNIT-VI: SORTING

1. A) Write algorithm for merge sort. [5 M]
   B) Discuss how to sort elements using merge sort with suitable example. [5 M]
2. A) Rearrange following numbers using quick sort: [5 M]
   10, 6, 3, 7, 17, 26, 56, 32, 72
   B) Write a program to sort the elements using radix sort. [5 M]
3. A) State and explain insertion sort with example. [5 M]
   B) Differentiate between iterative merge sort and recursive merge sort [5 M]
4. A) State and explain heap sort with example. [6 M]
   B) Evaluate time complexity and space complexity of an algorithm. [4 M]