

# Question Paper

Exam Date & Time: 10-Jan-2024 (09:30 AM - 12:30 PM)



## MANIPAL ACADEMY OF HIGHER EDUCATION

THIRD SEMESTER B.TECH. DEGREE EXAMINATIONS - JANUARY 2024  
SUBJECT: CSE 2122/CSE-2122- DATA STRUCTURES  
(MAKEUP)

Marks: 50

Duration: 180 mins.

Answer all the questions.

- 1A) Solve the following problem of sorting the elements in a ragged array using a C program. Write a function called Create that accepts an integer pointer to pointer and an integer 'n' and creates a ragged array consisting of 'n' rows using dynamic memory allocation functions. (4)

Write a recursive function called Sort which accepts an integer pointer and an integer 'm' to sort each row of the ragged array. Write the main function to test the above two functions and to display the ragged array before and after sorting.

- 1B) Define a structure *Employee* having following members (4)
- i) Name -a char array type
  - ii) Empcode-an int type
  - iii) Age- an int type( at the time of join)
  - iv) Joining date -a pointer to a structure Date which has three integer members for storing day, month and year.
  - v) Salary-a float type.

Write a C program to create an array of structure of type *Employee* to hold details of 'n' employees. Read the data into the structure array and display the same in main function. Write a function *Retire* which takes a pointer of the type structure *Employee* and compute the retirement year of each employee and the salary at that time considering 20% hike per year. (standard age of retirement is 60). Test this function from the main function.

- 1C) One of the applications of a stack is to convert postfix expressions to prefix expressions. List the steps involved in converting postfix expression to prefix expression demonstrating the use of stacks. (2)

- 2A) Illustrate the concept of dangling pointer and Write a program in C to print all the alphabets using pointer. (4)  
(1+3 = 4 marks)

- 2B) Prepare a user-defined function to reverse the elements of a given queue as shown below. (3)



- 2C) Prepare a complete C program consisting of a function *sortedInsert(. . .)* to insert an integer into a sorted doubly linked list, such that after insertion the list remains sorted. Also, include a function to display the contents of the list. (3)

- 3A) Write the functions *countNonleaf* to count the number of leaf nodes in a binary tree and *tree\_path* to (4)

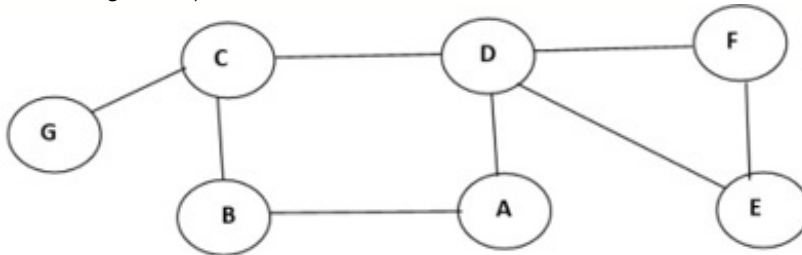
find the path from given node to the root.

- 3B) i) Construct a binary search tree for the list: 8,3,1,6,10,4,7,14,13. (4)  
ii) Search an element 7 in the tree and show all the steps.  
iii) Write the preorder and post-order traversals of the binary search tree.

- 3C) Write a function to insert an item at the end of Circular Singly Linked List. Use the following function (2)  
prototype:  
Nodeptr InsertLast(Nodeptr last, int item); where last points to the last node of linked list.

- 4A) Identify and Illustrate the way to deal with the problem of null pointers in the binary tree with a (4)  
diagram.  
Make the tree for the following elements 6, 4, 8, 2, and 5, and show all the steps.  
(2+2 = 4 marks)

- 4B) Distinguish between Depth-first search (DFS) and Breadth-first search (BFS). A road transportation (4)  
system between 5 cities is represented using the graph given below. Apply DFS to determine  
whether this graph is connected and also list the order in which vertices are visited (Consider 'G' as  
the starting vertex).



- 4C) Discover the suitable data structure for handling the Call in Customer care justify your answer and (2)  
implement the code for it.  
( $\frac{1}{2} + 1\frac{1}{2} = 2$  marks)

- 5A) Write functions to (4)  
i) Check whether binary trees are structurally identical or not.  
ii) Merge two binary trees by summing their values.

- 5B) For the following traversals of binary tree, construct the corresponding binary tree. Explain each (4)  
step of the construction process with figures. Finally write the level order traversal of the  
constructed tree.

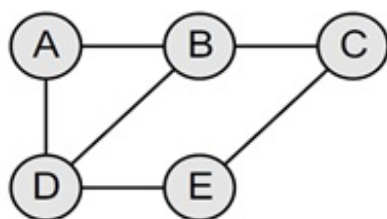
Preorder traversal:

39, 27, 18, 9, 10, 21, 19, 29, 28, 36, 45, 40, 54, 59, 65, 60

In-order traversal:

9, 10, 18, 19, 21, 27, 28, 29, 36, 39, 40, 45, 54, 59, 60, 65

- 5C) Adjacency list representation of a graph has the list of nodes which are adjacent to the (2)  
corresponding nodes of the graph. Apply the rule of generating adjacent list and write the adjacency  
list for the given graph.



-----End-----