U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

December 2023 Supplementary Examinations

Programme: B.E.	**Semester: III**
Branch: Computer Science and Engineering	**Duration: 3 hrs.**
Course Code: 22CS3PCDST	**Max Marks: 100**
Course: Data Structures	

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

UNIT - I

1 a) Convert the following Infix expression into Postfix expression: **07**

((A + B) - C * (D / E)) + F

b) Write a program to implement queue (First-In First-out operations) using two stacks. **08**

c) Show the recursive function calls for computing the factorial(n) for n=5. Draw the recursion tree. **05**

UNIT - II

2 a) Consider the code given below: **05**

```
void fun(Queue *Q)
{
    Stack S;  // Say it creates an empty stack S

    // Run while Q is not empty
    while (!isEmpty(Q))
    {
       // deQueue an item from Q and push the dequeued item to S
       push(&S, deQueue(Q));
    }

    // Run while Stack S is not empty
    while (!isEmpty(&S))
    {
      // Pop an item from S and enqueue the poppped item to Q
      enQueue(Q, pop(&S));
    }
}
```

What does the above function do,explain with the suitable example.

b) Differentiate between Linear Queue and Circular Queue. **05**

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages.
Revealing of identification, appeal to evaluator will be treated as malpractice.

c) B.M.S. College of Engineering is planning to conduct a value added course on "Data Structures". The number of participants is restricted to 25. Participants are selected based on first come first serve basis. Also, if one of the participant wishes to not attend then the next one in the list can attend the same. Also, "Seats full!" message needs to be displayed when all 25 people have registered. Implement the above scenario using a suitable data structure. **10**

UNIT - III

3 a) There are two singly linked lists in a system. By some programming the end node of one of the linked list got linked to the second list, Write a program to print the point where two linked list were merged. **10**

b) Write a C program for deletion of a node from the given Single linked list using single pointer variable. **10**

OR

4 a) Write a function that moves the last element to the front in a given Singly linked list. For example, if the given linked list is 1->2->3->4->5, then the function should change the list to 5->1->2->3->4. **07**

b) Given a Singly linked list, write a function to swap elements pairwise. **08**

For Eg:
Input : 1->2->3->4->5->6->NULL
Output : 2->1->4->3->6->5->NULL

Input : 1->2->3->4->5->NULL
Output : 2->1->4->3->5->NULL

c) Consider the code given below: **05**

```
void fun2(struct Node* head)
{
  if(head== NULL)
    return;
  printf("%d  ", head->data);

  if(head->next != NULL )
    fun2(head->next->next);
  printf("%d  ", head->data);
}
```

What does the function do?
Also, write the output for the following two input values:

(a) 1->2->3->4->5
(b) 1->2->3->4->5->6

UNIT - IV

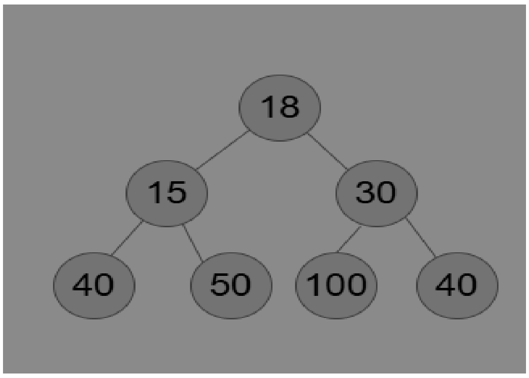
5 a) Construct a Binary Search Tree for the below given node values: **06**

13, 3, 4, 12, 14, 10, 5, 1, 8, 2, 7, 9, 11, 6, 18

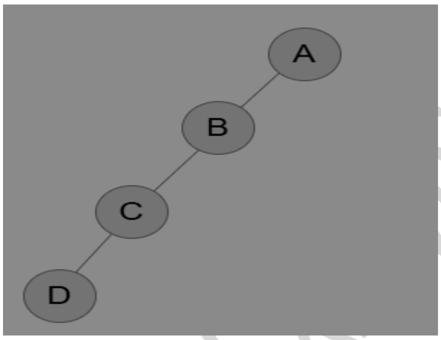
Write a Binary Search tree after deleting node 8 and 10.

b) Identify the type of Binary tree in the figures given below with proper justification. **08**

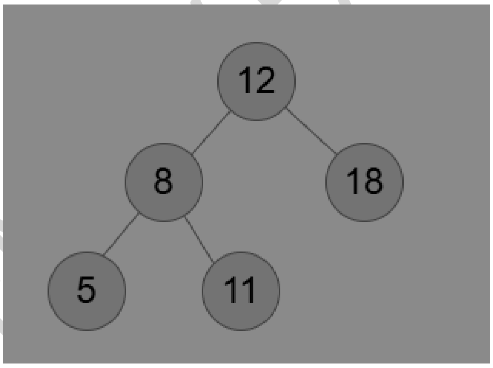
(a)



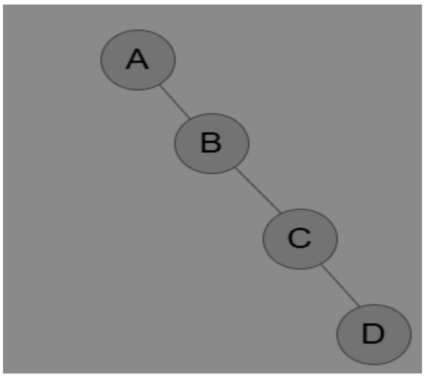
(b)



(c)



(d)



c) Given the following Preorder traversal of a Binary Search tree: **06**

20, 10, 5, 15,13, 35, 30, 42

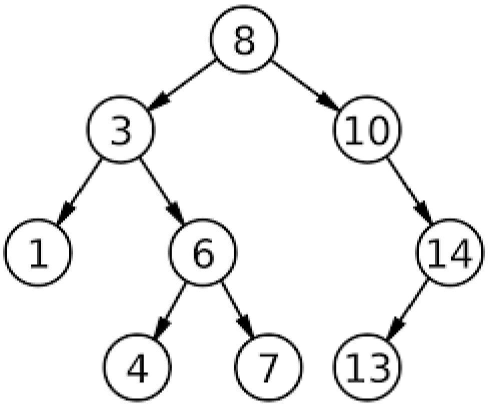
Construct the Binary tree. Show all the steps clearly.

OR

6 a) Write a program to find the minimum value in a Binary Search tree. **08**

b) For the below Binary Search tree, perform the following: **06**

(i) Inorder traversal
(ii) Preorder traversal
(iii) Postorder traversal



c) Write a program to remove all the leaf nodes from a Binary Search tree. **06**

UNIT - V

7 a) Construct a Hash table for the following numbers: **10**

5, 67, 91, 23, 45, 13, 19, 78, 56, 39

Show how collision is resolved using the following techniques with table size M=11

(1) Linear Probing
(2) Quadratic Probing
(3) Double Hashing with hash function H^1(K)= 7-(Key%7)

b) Construct a Hash table for the following numbers: **10**

18, 21, 45, 67, 11, 5, 61, 23, 77, 99, 19, 56, 61, 29, 68

Show how collision is resolved using Extendible Hashing with each step demonstrated clearly. Also, mention how directory expansion and bucket splitting happens. Consider bucket limit=3.

******