

**PES University, Bengaluru**

(Established under Karnataka Act 16 of 2013)

END SEMESTER ASSESSMENT (ESA) - DEC 2023**UE22CS252A - Data Structures and its Applications****Total Marks : 100.0**

1.a. Provide structure definition for node of a circular singly linked list and write a C function to perform ordered insertion in a circular singly linked list (2M+6M) M:Marks (8.0 Marks)

1.b. Write a C function to convert an infix expression to prefix expression using stack (Do not give code for stack functions) (8.0 Marks)

1.c. A polynomial consists of a set of terms, where each term consists of coefficient and exponent. Assume this polynomial is stored in a singly linked list (without header node) for which we maintain pointer to the first node, each node corresponds to a specific term of the polynomial (Assume terms are stored in descending order of exponents).
i) Write structure definition for a node of polynomial----- (2M)
ii) Assume two input polynomials are stored in the linked lists pointed by node pointers PolyA and PolyB, write a C function to add these two polynomials to generate linked list resPoly which represents resultant polynomial ----- (7M) (9.0 Marks)

2.a. A circular Queue is implemented using an array and is defined as follows

```
typedef struct Queue
{int front, rear, capacity;
int *array;
}Queue;
```

Provide C code for the following functions

int Isfull(Queue *PQ) // PQ is pointer to Queue type variable
(1M)

int isempty(Queue *PQ) (1M)

int Insert(Queue *PQ, int ele)// successful insertion return 1 and 0 otherwise (3M)

int Delete(Queue *PQ)// return the deleted element and -1 for unsuccessful deletion (3M) (8.0 Marks)

2.b. Provide structure definition for node of a binary tree represented using linked representation.-----2M

Write an iterative C function to traverse a Binary tree (Linked Representation) using Postorder traversal technique. Do not give function definitions for any auxiliary data structure required.-----6M (8.0 Marks)

2.c. Assume a binary search tree is created using linked representation and populated with the integer keys. Fill in the code for the following functions

1. bstnode* findmin(bstnode * root)// accepts the address of root node of binary search tree and returns the address of the node with the minimum value (3M)

2. This function should create a sorted singly linked list from the data present in binary search tree

sllnode* create_sll_from_BST(bstnode *root)// accepts the address of root node of binary search tree and returns pointer to first node of created linked list (6M)

(Note: provide code for additional functions required if any) (9.0 Marks)

3.a. Given a threaded binary tree where the NULL left and NULL right pointers of a node are made to point to the inorder predecessor and successor respectively, write an iterative function to traverse the tree in inorder. If this function is used to traverse a threaded binary search tree, what can you say about the traversal sequence (6M+2M)
(8.0 Marks)

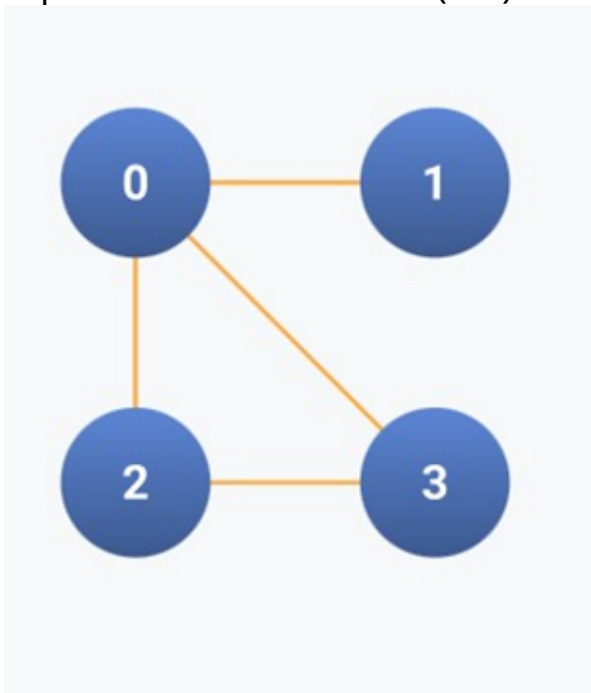
3.b. Construct an AVL Tree from the key values 12, 8, 9, 20, 10, 15, 3, 11, 5. List any one key which when inserted in the resultant tree will result in Left Left balance violation. (7M+1M)
(8.0 Marks)

3.c. Write a C function to create a min heap (array representation) from the given set of integer keys using bottom up heap construction. (6M)
Using this procedure construct a heap for the keys: 9, 3, 7, 4, 5, 6, 1, 2 (3M)
(9.0 Marks)

4.a. Draw the hash table that results after inserting the keys 33, 10, 9, 13, 12, 45, 26 under the following conditions: The size of the hash table is 12. Open addressing and double hashing is used to resolve collisions. The hash function used is $H(k) = k \bmod 12$ The second hash function is: $H_2(k) = 7 - (k \bmod 7)$ (6M)
What is the advantage of using double hashing over linear probing (2M)
(8.0 Marks)

4.b. Give structure definition for a node of a Trie Tree. (2M)
Write C function to print all the words of specific length stored in a Trie Tree (6M)
(8.0 Marks)

4.c. For the given graph, provide adjacency matrix and adjacency list representation. (3M)



Write a C function to determine the vertices reachable from a given source vertex in a graph (Adjacency List Representation) using BFS. Do not give code for auxiliary data structure used. (6M)
(9.0 Marks)