PES University, Bengaluru

(Established under Karnataka Act 16 of 2013)

END SEMESTER ASSESSMENT (ESA) - Dec 2024

UE23CS252A - Data Structures and its Applications

Total Marks : 100.0

1.a. Given a pointer to the first node of a doubly linked list. Write a function to delete and return the data of the node at the nth position **where n is greater than 1(n>1).** Use the following decalaration the structure node.

```
struct node
{
  int data;
  struct node *prev, *next;
}
```

function prtotype

int delete_pos(struct node* head, int pos); // head is pointer to the first node, pos is the position of the node to be deleted (8.0 Marks)

1.b. Given a pointer to the first node of a singly linked list of integer elements, write a recursive function to return the largest element in the list.

Use the following declaration of struct node.

```
struct node
{
 ind data;
 struct node *next;
};
```

function prototype

int maximum(struct node *head). // head is pointer to the first node of the list (6.0 Marks)

1.c. Write a function called merge that takes two sorted stacks A and B (min on top) and merge into a new stack C that is sorted (min on top). The stacks are implemented as a structure. You are allowed to use only the stack operations such as pop, push, empty and peep. You need to write only push function. Other functions are called in the function merge.

.

Use the following structure declaration of stack.

```
struct stack
{
  int s[100];
  int top;
}
void merge(struct stack *A, struct stack *B, struct stack *C);// A, B and C
are pointers to stack . A and B are merged to  C
```

(6.0 Marks)

1.d. Convert the following infix expression to postfix and prefix

A + (((B - C) * (D - E) + F) / G) $ (H - J) // $ is the exponentiation operator

(5.0 Marks)

2.a. Given a Circular Queue of size **SIZE**, having **n** integers, Write a function that finds the maximum element in the queue. The queue should remain intact after finding the maximum element. You should use only standard functions of queue (qinsert, qdelete). You need to also write the functions qinsert and qdelete. f and r contain the index of the first and last element of the queue.
Use the following declaration of the queue

```
struct queue
{
  int element[100];
  int f, int r;
  int size;
}
```

function prototypes

```
int max( struct queue *q);
int qinsert(struct queue*ptr_q, int data)
int qdelete(struct queue*ptr_q)
```

(8.0 Marks)

2.b. Consider a Double-Ended Queue (Deque) implemented using a Doubly Linked List. Write functions to perform the following operations:

1. InsertFront: Insert an element at the front of the deque.
2. DeleteRear: Delete an element from the rear of the deque.

consider the following structure declaration

```
struct node
{
   int data;
   struct node *next * prev
};
struct dequeue
{
   struct node *front;
  struct node *rear
};
```

front and rear are pointers to the first and last node of the dequeue

```
void qinsert_front(struct deque *p,int x); // p is pointer to  dequeue
int qdelete_rear(struct deque *p);  // p is pointer to  dequeue
```

(6.0 Marks)

2.c. Construct a binary search tree inserting the following numbers in the sequence given
30,40,20,28, 35,38, 50,25,22,24
Show the tree after deletion of 20 . (6.0 Marks)

2.d. Write a function to recursively compute the height of a binary tree. Use the following structure of the tree node.

```
struct tnode
{
  int data;
  struct tnode*left;
  struct tnode * right;
};
```

function prototype

```
int height(struct tnode *root);//root is pointer to the root of the tree
```

(5.0 Marks)

3.a. You are tasked with designing a flight reservation system to manage unique flight numbers using a Binary Search Tree (BST) implemented using an array. The BST uses a level-order representation with the rule that the root node is stored at index 0.

a) Write a function InsertFlight(arr, flightNo) to insert a new flight number into the BST while maintaining the BST property. Assume the array has a sufficient size to accommodate new entries and intialised with -1
b) Write a function to list all the flights in increasing order of thier numbers.
c) Write a function to search for a flight number using recursion; the function should return 1 if flight number is found. else return -1
(8.0 Marks)

3.b. Insert the following strings in the sequence given to construct an AVL Tree
BAG, ZOO, CAT, WOLF, DOG , YAK, XMAS (6.0 Marks)

3.c. Assume the following elements {2, 3, 8, 4, 7, 6, 5, 9, 10} represent the priorities of tasks in an operating system, where higher values indicate higher priority.

1. Construct a max heap of the elements using the bottom-up approach.
2. Recreate the heap after deleting the task with the highest priority

(6.0 Marks)

3.d. You are tasked with building a social network graph where users are represented as nodes and friendships between users are represented as edges in an undirected graph represented as adjacency matrix. To explore the connections in the network, you decide to implement a Breadth-First Search (BFS) traversal.

Write a function BFS (graph, startNode) that performs a Breadth-First Search (BFS) traversal of the graph starting from a given node (startNode). You need not write the queue functions. (5.0 Marks)

4.a. A hash table of size m=11 uses quadratic probing to resolve collisions. The hash function used is:
h(k)=k mod 11. The sequence for quadratic probing is given by:

$h_i(k)$ = (h(k) + i^2) where i=1,2,3...11.

Insert the following keys into the hash table:

27, 18, 29, 28, 39, 6

Show the placement of these keys in the hash table step by step. Also show the final Hash Table (8.0 Marks)

4.b. You are tasked with implementing a dictionary application using a Trie data structure. The application allows users to search for words efficiently.

Insert the given words into a Trie

"cat", "car", "care", "bat", "bar", "bare", "barn" .

Show the Trie structure after all the words have been inserted. How many nodes including the root nodes are present in the Trie.

Implement a function to search for a word stored in the trie. Use suitable structure for the trie node. (6.0 Marks)

4.c. Write a function to detect a cycle in the undirected graph represented as adjaceny matrix using DFS. (6.0 Marks)

4.d. Construct a suffix tree for the string xyzxyz$. How many leaf nodes are present in the constructed suffix tree ? (5.0 Marks)