

# **BÀI TẬP THỰC HÀNH CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT**

## **LAB MANUAL**

**Academic Year : 2022 - 2023**

**Semester : III Semester**

**Prepared by**

**Ms. KhangVQH**

<b>S. No.</b>	<b>Experiment</b>
1	<b>SEARCHING TECHNIQUES</b>
2	<b>SORTING TECHNIQUES</b>
3	<b>SORTING TECHNIQUES</b>
4	<b>IMPLEMENTATION OF STACK AND QUEUE</b>
5	<b>APPLICATIONS OF STACK</b>
6	<b>IMPLEMENTATION OF SINGLE LINKED LIST</b>
7	<b>IMPLEMENTATION OF DOUBLE LINKED LIST</b>
8	<b>IMPLEMENTATION OF STACK USING LINKED LIST</b>
9	<b>IMPLEMENTATION OF QUEUE USING LINKED LIST</b>
10	<b>IMPLEMENTATION OF BINARY SEARCH TREE</b>

## WEEK-6

### IMPLEMENTATION OF SINGLE LINKED LIST

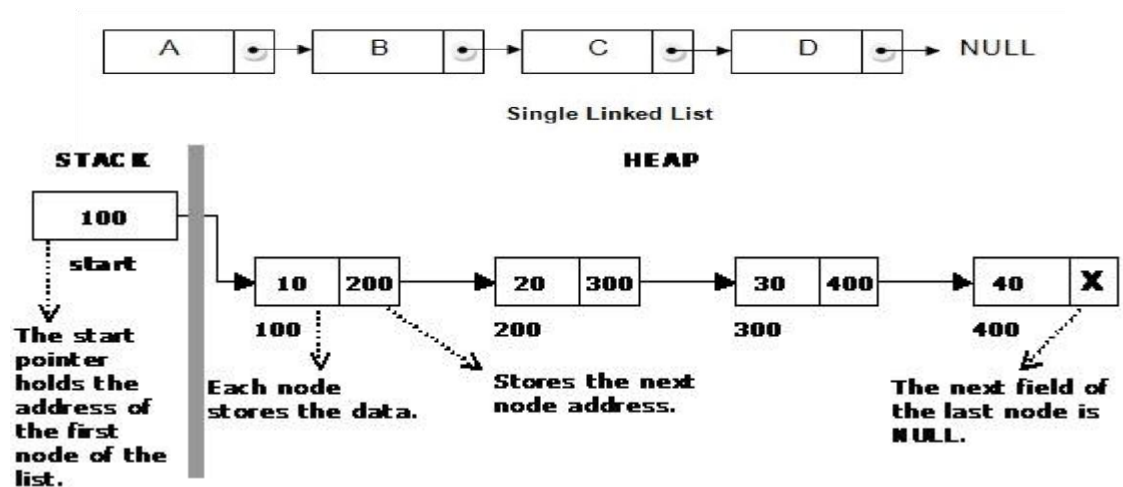
#### 6.1 OBJECTIVE:

- a. Write C program to perform the following operations on single linked list. (i)  
Creation (ii) insertion (iii) deletion (iv) traversal

#### 6.3 PROGRAM LOGIC:

##### Procedure for Single linked list

1. A singly linked list's node is divided into two parts. The first part holds or points to information about the node, and second part holds the address of next node. A singly linked list travels one way.
2. The beginning of the linked list is stored in a "start" pointer which points to the first node. The first node contains a pointer to the second node. The second node contains a pointer to the third node, ... and so on.
3. The last node in the list has its next field set to NULL to mark the end of the list.
4. The basic operations in a single linked list are: Creation, Insertion, Deletion, Traversing.



#### 6.3 IMPLEMENTATION:

## INPUT / OUTPUT:

```
*****SINGLE LINKED LIST*****
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:1
Enter data:10
Node created 10
Node inserted 10
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:3
Enter data:30
Node created 30
Node inserted 30
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:2
Enter data:20
Enter the position:1
Node created 20
Node inserted 20
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:7
```

```
traversal forward
10
20
30
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:8
Number of nodes 3
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:5
Enter position:2
Node deleted 20
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:4
Node deleted 10
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
```

```
Node deleted 30
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:7
traversal forward
None
1. Insert at beginning
2. Insert at middle
3. Insert at end
4. Delete at beginning
5. Delete at middle
6. Delete at end
7. Traversal forward
8. Count number of nodes
9. Exit
Enter choice:9
Exit
```

#### 6.4 LAB ASSIGNMENT:

1. Formulate a program to create a singly linked list and perform insertion, deletion and traversing operations on a singly linked list.
2. Write a program to merge two linked list?
3. Compose a program to print odd nodes of a linked list?
4. Write a program to divide the linked list into two parts into odd and even list?

#### 6.5 POST-LAB VIVA QUESTIONS:

1. What is the time complexity to insert a node at the beginning of linked list?
2. What is the time complexity to traverse a linked list?
3. How many modifications are required to delete a node at the beginning?
4. How many modifications are required to insert a node in the middle of the linked list?
5. What are the types of linked list?
6. What are the applications of a linked list?