Linked Lists

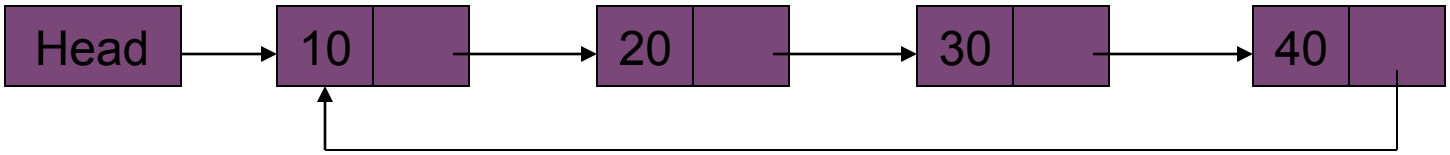
bitCode technologies

learn.innovate.share

Singly Circular Linked List

- A small change is made to the linear list, so that the next field in the last node contains a pointer back to the first node rather than the NULL pointer.
- Such a linked list is called as the circular linked list.
- From any point in such a list it is possible to reach any other point in the list.
- A NULL pointer represents an empty circular linked list.

Structure



• In This case as the list is circular there is no first or last node of the list. However some convention should be used so that a certain node is last node and the node following that is first node.

•The address of the first node is placed in the head.

bitCode technologies

Singly Circular Linked List

- Advantages

  - Every node is accessible from a given node.

  - In singly linked list to do every operation we have to go sequentially starting from the head. This is not a case with circular linked list.

- Disadvantages :

  - Without some care in programming it is possible to get into an infinite loop (identify the head node).

Class “CLinkedList”

```
class CLinkedList {
  Node * head;

public:
  CLinkedList();
  void insert( int data );
  void insertAtBeg( int data );
  void delByPos( int pos );
  void display();
  ~CLinkedList();
};
```

Doubly Linked List

- Singly and Circular are Linked lists in which we can traverse in only one direction. Sometimes it is required that we should be able to traverse the list in both directions.

For example to delete a node from the list we ought to have a address of previous node in the list.

- For a linked list to have such a property implies that each node must contain two link fields instead of one. The links are used to denote the predecessor and successor of a node.

- A linked list containing this kind of nodes is called Doubly linked list.

Doubly Linked List - Structure

- Here a node is having three parts
  - data part
  - address part to store address of previous node
  - address part to store the address of next node.

bitCode technologies

Doubly Linked List

- The advantage here is that the list could be traversed in both the directions. Which removes the drawbacks of Singly and Singly circular linked list.
- But still in some situations it may be costly to put two address fields into a Node.

Implementation

```
class DLinkedList {

      Node * head;

public:

  DLinkedList();
  void insert( int data );
  void insert( int data, int pos );
  void deleteByVal( int data );
  void deleteByPos( int pos );
  void display();
  ~DLinkedList();
};
```