Linked List

(One-way Linked List-Insertion and Deletion)

Insertion into a Linked List

1. Insertion at the beginning of the list
2. Insertion before a given node
3. Insertion of a node with a given location
4. Insertion into a sorted list.

Insertion at the beginning of the list

Insert_Begin(Info, Link, Start, Item)

```
1.  Set Info[New] := Item
2.  Set Link[New] := Start
3.  Set Start := New.
4.  Return
```

Insertion before a given node

Insert_Before_Node(Info, Link, Start, Item, Given_Item)

```
1.    Set Info[New] := Item
2.    If Info[Start] = Given_Item then
3.        Link[New] := Start  and  Start := New and Return.
4.    Set Save := Start  and PTR := Link[Start]
5.    While PTR ≠ NULL and Info[PTR] ≠ Given_Item do step 6
6.        Save := PTR  and  PTR := Link[PTR]
7.    If  PTR = NULL then Write: “Given Item is not in the List ” and Return.
8.    Link[New] := PTR  or  Link[Save] := New.
9.    Return
```

Insertion with a given location

Insert_In_LOC(Info, Link, Start, Item, LOC)

```
1.  Set Info[New] := Item
2.  If LOC=1 then Link[New] := Start and Start := New and Return.
3.  Set Save := Start  and PTR := Link[Start] and Count := 1.
4.  While PTR ≠ NULL and Count+1  ≠ LOC do step 5
5.      Save := PTR  and  PTR := Link[PTR] and Count := Count+1.
6.  Link[New] := PTR  and  Link[Save] := New.
7.  Return
```

Insertion into a Sorted List

Insert_Sorted_List_(Info, Link, Start, Item, LOC)

```
1.    Set Info[New] := Item
2.    If Item < Info[Start] then Link[New] := Start and Start := New and Return.
3.    Set Save := Start  and PTR := Link[Start].
4.    While PTR ≠ NULL do steps 5 and 6
5.        If Item > Info[PTR] then Save := PTR  and  PTR := Link[PTR].
6.        If Item < = Info[PTR] then Link[New] := PTR and Link[Save] := New and Return
7.    If PTR = NULL then Link[New] := PTR and Link[Save] := New.
8.    Return
```

Deletion From a One-Way Linked List

1. Deletion of the first node

1. Deletion of a node with a given location

2. Deleiton of a node with given value

Deletion of First Node

Suppose we want to delete the first node N from the one way list.

Delete_First(Info, Link, Start)

```
1. If Start = NULL then Return
2. Set Start := Link[Start]
3. Return
```

Deletion of a Node with Given Location

Suppose we want to delete a node N with location N from the list.

Delete_Node_LOC(Info, Link, Start, LOC)

```
1.  If  LOC = 1 then Start := Link[Start] and Return.
2.  Set Save := Start  and PTR := Link[Start] ans Count := 1
3.  While PTR!=NULL and Count + 1 ≠  LOC do step 4
4.      Save := PTR  and  PTR := Link[PTR]  and Count := Count + 1.
5.  If PTR = NULL then Write: "Location not found"  and Return.
6.  Set Save := Link[PTR]
7.  Return
```

Deletion of a Node with Given Value

Suppose we want to delete a node N with a given value from the list.

Delete_Node_Value(Info, Link, Start, Item)

```
1.  If Item = Info[Start]  then Start := Link[Start] and Return.
2.  Set Save := Start  and PTR := Link[Start]
3.  While PTR!=NULL and Info[PTR] ≠ Item do step 4
4.      Save := PTR  and  PTR := Link[PTR].
5.  If PTR = NULL then Write: “Item not in the List”  and Return.
6.  Set Link[Save] := Link[PTR]
7.  Return.
```

END