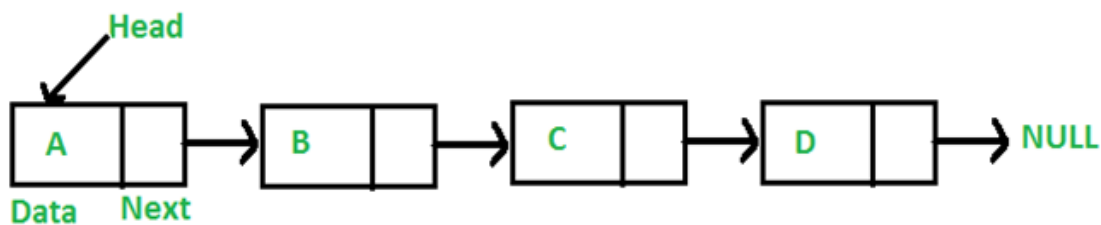


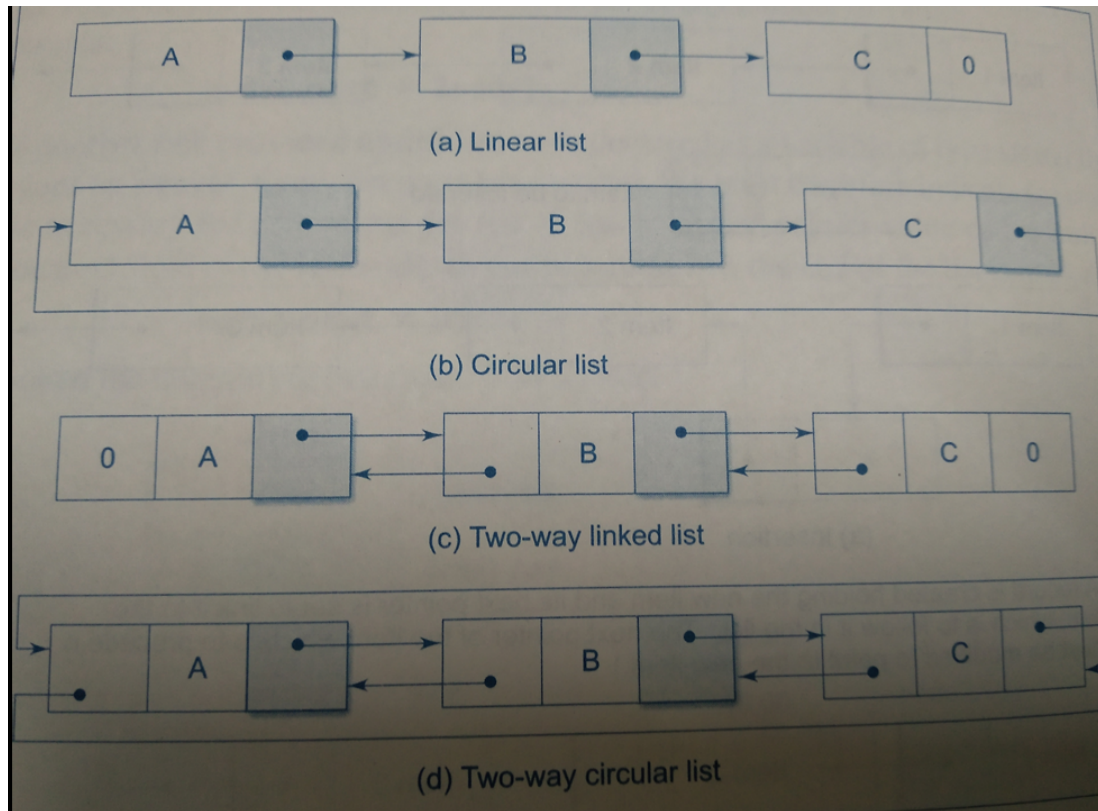
Date :

- A linked list, or one-way list, is a linear collection of data elements, called nodes, where the linear order is given by means of *pointers*
- That is, each node is divided into two parts : the first part contains the information of the element, and the second part, called the link field or next pointer field, contains the address of the next node in the list

- ☐ Traversal - To transverse is to move to and fro through the list. This is useful to access or search an element in the list
- ☐ Insertion - To place a new element in the list
- ☐ Deletion - To remove an element from the list



- **Singly linked list** - Each node contains only one link part
- **Circular linked list** -circular linked list has no beginning and no end. The last node points back to the first item.
- **Two way/doubly linked list**- this uses a set of pointers one pointing to next and other pointing to the proceeding item
- **Circular doubly linked list** -employs both the forward pointer and backward pointer in circular form



Exercise : Write a 'C' program to implement Linked List operation

```
#include<stdio.h>
#include<stdlib.h>
//Creation of Linked list node
struct node
{
    int data;
    struct node *next;
}*new,*head,*tail,*temp;
int main()
{
    int ch,key;
    int value,a;
    x:
    printf("1.Insert\n 2.Display\n 3.Search\n ");
    scanf("%d",&a);
    switch(a)
    {
        case 1:
```

```

{
do
{
printf("Enter element\n");
new=(struct node *)malloc(sizeof (struct node)); //dynamic memory allocation
scanf("%d",&value);
new->data=value;
new->next=NULL;
if(head==NULL)
{
head=new;
tail=new;
}
else
{
tail->next=new;
tail=new;
}
printf("1-To continue\t 2- Exit\n");
scanf("%d",&ch);
}while(ch==1);

goto x;
}
case 2:
{
printf("The elements in the list are\n");
temp=head;
while(temp!=NULL)
{
printf("%d\n",temp->data);
temp=temp->next;
}
goto x;
}
case 3:
printf("Enter a value to be searched");
scanf("%d",&key);
temp=head;
while(temp!=NULL)
{

```

```
        if(temp->data==key)
        {
            printf("Value found\n");
            goto x;
        }
        temp=temp->next;
    }
    printf("Value not found\n");
    goto x;
```