```
#include<stdio.h>
#include<stdlib.h>

struct node
{
        int value;
        struct node *link;
};
typedef struct node node1;
void in_beg();
void in_end();
void in_bw();
void del_beg();
void del_end();
void del_bw();
void display();
node1 *create();
node1 *nptr;
node1 *start=NULL;
void main()
{
        int ch,wish;

        do
        {
        printf("\n\n\tLinked list implementation\n\t*****************************\n\n\t1.insert at
        beginning\n\t2.insert at end\n\t3.insert in between\n\t4.delete from beginning\n\t5.delete
        from end\n\t6.delete in between\n\t7.display\n\t8.exit\n\n\tenter your choice : ");
        scanf("%d",&ch);
          switch(ch)
           {
            case 1:in_beg();
                    break;
            case 2:in_end();
                    break;
            case 3:in_bw();
                    break;
            case 4:del_beg();
                    break;
            case 5:del_end();
                    break;
            case 6: del_bw();
```

```
                        break;
              case 7:display();
                       break;
              case 8:exit(0);
                       break;
               default:printf("\ninvalid choice");
            }
          printf("\nDo you wish to continue ?(1/0)\n");
         scanf("%d",&wish);
      }
    while(wish==1);
}
node1 *create()
{
          node1 *nptr=(node1*)malloc(sizeof(node1));
          if(nptr==NULL)
          {
           printf("memory overflow");
            return 0;
          }
          else
            return nptr;
}

void in_beg()
 {
   int val;
   node1 *nptr=create();
   printf("enter element");
   scanf("%d",&val);
   nptr->value=val;
   if(start==NULL)
     {
              start=nptr;
               nptr->link=NULL;
     }
   else
     {
               nptr->link=start;
               start=nptr;
     }
  }
```

```
void in_end()
   {
     node1 *temp,*nptr=create();
     int val;
     printf("enter element");
     scanf("%d",&val);
     nptr->value=val;
     nptr->link=NULL;
     temp=start;
     while(temp->link!=NULL)
       {
                   temp=temp->link;
       }
      temp->link=nptr;
}

void in_bw()
    {
     node1 *temp,*nptr=create();
     int val,pos,i;
     printf("enter element and position to be inserted ");
     scanf("%d %d",&val,&pos);
     nptr->value=val;
     nptr->link=NULL;
     temp=start;
     if(pos==1)
       {
                 nptr->link=start;
                 start=nptr;
       }
     else
       {
                 for(i=1;i<pos-1;i++)
                  {
                      temp=temp->link;
                  }
                 nptr->link=temp->link;
                 temp->link=nptr;
       }
    }
```

```
void display()
   {
    node1 *temp;
    if(start==NULL)
          printf("LIST EMPTY!!\n");
    temp=start;
    while(temp!=NULL)
     {
             printf("%d ",temp->value);
             temp=temp->link;
     }
   }

void del_beg()
  {
   node1 *temp;
   if(start==NULL)
             printf("LIST EMPTY\n");
   else
     {
              temp=start;
              start=start->link;
              free(temp);
     }
   }

void del_end()
   {
    node1 *temp,*prev;
    temp=start;
   while(temp->link!=NULL)
           {
             prev=temp;
             temp=temp->link;
    }
   prev->link=NULL;
   free(temp);
  }

void del_bw()
  {
   node1 *temp,*prev;
   int i,pos;
   printf("enter position of the node to be deleted");
   scanf("%d",&pos);
   temp=start;
   if(pos==1)
     start=start->link;
   for(i=1;i<pos;i++)
    {
      prev=temp;
      temp=temp->link;
    }
    prev->link=temp->link;
   free(temp);
   }
```

OUTPUT:

```
ubuntu@ubuntu-H81M-S:~/sneha43$ gcc singly.c
ubuntu@ubuntu-H81M-S:~/sneha43$ ./a.out
        Linked list implementation
        *****************************


        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 1
enter element1
Do you wish to continue ?(1/0)
1

        Linked list implementation
        *****************************
        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 1
enter element2

Do you wish to continue ?(1/0)
1

        Linked list implementation
        *****************************

        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 2
enter element6
Do you wish to continue ?(1/0)
1

        Linked list implementation
        *****************************
        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 3
enter element and position to be inserted 7
2
Do you wish to continue ?(1/0)
1

        Linked list implementation
        *****************************

        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 7
2 7 1 6
Do you wish to continue ?(1/0)
1
        Linked list implementation
        *****************************

        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 4
Do you wish to continue ?(1/0)
1

        Linked list implementation
        *****************************

        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 7
7 1 6
Do you wish to continue ?(1/0)
1
        Linked list implementation
        *****************************
        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
```

```
          6.delete in between
          7.display
          8.exit
          enter your choice : 5
Do you wish to continue ?(1/0)
1

          Linked list implementation
          *****************************

          1.insert at   beginning
          2.insert at end
          3.insert in between
          4.delete from beginning
          5.delete from end
          6.delete in between
          7.display
          8.exit
          enter your choice : 7
7 1
Do you wish to continue ?(1/0)
1
          Linked list implementation
          *****************************

          1.insert at   beginning
          2.insert at end
          3.insert in between
          4.delete from beginning
          5.delete from end
          6.delete in between
          7.display
          8.exit
```

```
        enter your choice : 1
enter element9
Do you wish to continue ?(1/0)
1
        Linked list implementation
        *****************************

        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 6
enter position of the node to be deleted2
Do you wish to continue ?(1/0)
1
        Linked list implementation
        *****************************

        1.insert at   beginning
        2.insert at end
        3.insert in between
        4.delete from beginning
        5.delete from end
        6.delete in between
        7.display
        8.exit
        enter your choice : 7
9 1
Do you wish to continue ?(1/0)
```

0