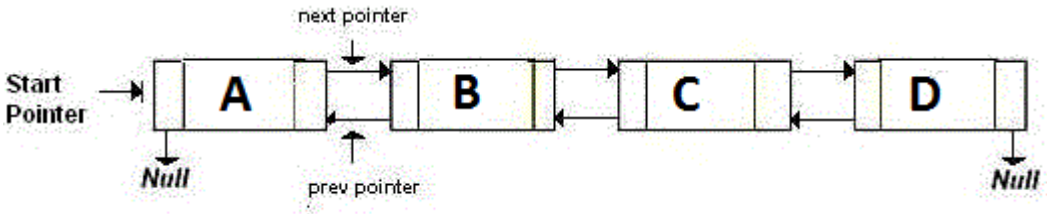
Design, Develop and implement C program for the following operations on doubly linked list.

a. Create doubly linked list of N nodes with integer data by adding each node at the front.

b. Delete the node of a given data if it is found, otherwise display appropriate message.

c. insert a node to the left of the node whose key value is read as input.

d. Display the contents of the list.




```
#include <stdio.h>
#include <stdlib.h>
struct abb
{
   int info;
   struct abb *p, *n;
 };
typedef struct abb *node;
node header=NULL;
node getnode();
void ins();
void insl();
void del();
void disp();
int main()
{
    int ch;
    while(1)
    {
          printf("\nChoices:");
          printf("\n\t1-Insert");
          printf("\n\t2-Insert left");
          printf("\n\t3-Delete node");
```

```
            printf("\n\t4-Display");
            printf("\n\t5-Exit");
            printf("\nEnter your choice: ");
            scanf("%d",&ch);
            switch(ch)
            {
                  case 1: ins();
                        break;
                  case 2: insl();
                        break;
                  case 3: del();
                        break;
                  case 4: disp();
                        break;
                  default:return 0;
            }
      }
}
 node getnode()
{
      node x;
      x=(node) malloc(sizeof(struct abb));
      return x;
}
void ins()
{
      node temp;
      int x;
      temp=getnode();
      printf("\nEnter the element to be inserted: ");
      scanf("%d",&x);
      temp->info=x;
      temp->p=NULL;
      temp->n=NULL;
      if(header==NULL)
            header=temp;
      else
      {
            temp->n=header;
            header->p=temp;
```

```
            header=temp;
        }
    }
    void insl()
    {
        node temp,ele;
        int x, y;
        if(header==NULL)
        {
            printf("\nEmpty list\n");
            return;
        }
        printf("\nEnter the element to be inserted: ");
        scanf("%d",&y);
        printf("\nTo left of which element should %d be inserted? Enter:
    ",y);
        scanf("%d",&x);
        temp=getnode();
        ele=header;
        temp->info=y;
        temp->p=NULL;
        temp->n=NULL;
        if(header->info==x)
        {
            temp->n=header;
            header->p=temp;
            header=temp;
        }
        else
        {
            while(ele!=NULL)
            {
                if(ele->info==x)
                    break;
                ele=ele->n;
            }
            if(ele!=NULL)
            {
                temp->p=ele->p;
                (ele->p)->n=temp;
```

```
                temp->n=ele;
                ele->p=temp;
        }
        else
                printf("\nNo element found");

    }
}
void del()
{
    node temp;
    int x;
    temp=header;
    if(header==NULL)
    {
        printf("\nNo element deleted.\n");
        return;
    }
    printf("Enter the node to be deleted: ");
    scanf("%d",&x);
    if(header->info==x)
    {
        temp=header;
        header=header->n;
        header->p=NULL;
        free(temp);
    }
    else
    {
        while(temp!=NULL)
        {
            if(temp->info==x)
                break;
            temp=temp->n;
        }
        if(temp!=NULL)
        {
            if(temp->n!=NULL)
            {
                (temp->n)->p=temp->p;
```

```
                    (temp->p)->n=temp->n;
                    free(temp);
                }
                else
                {
                    (temp->p)->n=NULL;
                    free(temp);
                }

            }
            else
            {
                printf("Element not found");
            }
        }
    }
    void disp()
    {
        node temp;
        if(header==NULL)
            printf("Absent");
        else
        {
            for(temp=header;temp!=NULL;temp=temp->n)
                printf("%d ",temp->info);
        }
    }
```

Output:

```
Choices:
    1-Insert
    2-Insert left
    3-Delete node
    4-Display
    5-Exit
Enter your choice: 1

Enter the element to be inserted: 10
```

```
Choices:
    1-Insert
    2-Insert left
    3-Delete node
    4-Display
    5-Exit
Enter your choice: 4
20 10
```