

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
    int data;
    struct node *next, *prev;
};

void insert Front(struct node **headptr, int value)
{
    struct node *newnode;
    newnode = (struct node*) malloc (sizeof(struct node));
    newnode->data = value;
    newnode->next = NULL;
    newnode->prev = NULL;
    if (*headptr == NULL)
        *headptr = newnode;
    else
    {
        newnode->next = *headptr;
        (*headptr)->prev = newnode;
        *headptr = newnode;
    }
}

void delFunc(struct node **headptr, int value)
{
    struct node *temp, *del;
    temp = *headptr;
    if (temp == NULL)
    {
        printf("The list is empty!!!\n");
        return;
    }
    else if (temp->next == NULL)
    {
        *headptr = NULL;
    }
}
```

}

else

{

if ((\*headpti) -> data == ~~value~~ value)

{

(\*headpti) = (\*headpti) -&gt; next;

(\*headpti) -&gt; prev = NULL;

printf("The value has been deleted!!!\n");

return;

}

while (temp -&gt; next != NULL)

{

if ((temp -&gt; next) -&gt; data == value)

{

del = temp -&gt; next;

temp -&gt; next = del -&gt; next;

(temp -&gt; next) -&gt; prev = temp;

printf("The value has been deleted!!!\n");

return;

}

temp = temp -&gt; next;

}

printf("The element was not found!!!\n");

}

}

void display(struct node \*temp)

{

if (temp == NULL)

{

printf("List is Empty!!!\n");

return;

}

else

{

while (temp != NULL)

```

    {
        printf("%d\t", temp->data);
        temp = temp->next;
    }
    printf("\n");
}

int main(int argc, char **argv)
{
    struct node * head = NULL;
    int choice, ele;
    while(choice != 4)
    {
        printf("Enter choice 1) Insert Element 2) Delete Element 3) Display\n");
        scanf("%d", &choice);
        switch(choice)
        {
            case 1: printf("Enter value to be inserted: "); scanf("%d", &ele); insert(head, ele);
            case 2: printf("Enter value to be deleted: "); scanf("%d", &ele); delete(head, ele);
            case 3: display(head); break;
            case 4: exit(0);
            default: exit(0);
        }
    }
    return 0;
}

```