#include <stdio.h>

#include <stdlib.h>

void insrt(void);

struct node

{

int data;

struct node\* next;

};

typedef struct node position;

typedef struct node\* temp;

typedef struct node\* list;

struct node \*head = NULL;

int main(void)

{

int option,a,b;

checkpoint:

printf("\nChoose the option\n1.Insert\n2.Delete\n3.Display\n4.Exit\n");

scanf("%d",&option);

switch(option)

{

case 1:

{

int x, num;

position \*temp,\*ptr;

temp=malloc(sizeof(struct node));

printf("Enter the element to insert\n");

scanf("%d",&x);

temp -> data = x;

temp -> next = NULL;

if(head == NULL)

{

head=temp;

}

else

{

printf("Enter the position after which you want to insert\n");

scanf("%d", &num);

ptr=head;

while(ptr->data != num)

{

ptr=ptr->next ;

}

temp->next = ptr->next;

ptr->next = temp;

}

goto checkpoint;

}break;

case 2:

{

int x;

printf("\nEnter the element to delete\n");

scanf("%d",&x);

position \*temp,\*ptr;

ptr = head;

if(ptr->data == x)

{

head = ptr->next;

free(ptr);

}

else

{

while( ptr != NULL && ptr->next->data == x)

{

temp = ptr;

ptr=ptr->next ;

}

temp -> next = ptr -> next;

free(ptr);

}

printf("\nThe deleted element is %d\n",x);

goto checkpoint;

}break;

case 3:

{

position \*ptr;

if(head==NULL)

{

printf("\nList is empty:\n");

}

else

{

ptr=head;

printf("\nThe contents of the List are :\n");

while(ptr!=NULL)

{

printf("%d",ptr->data );

printf("->");

ptr=ptr->next ;

}

printf("NULL\n");

goto checkpoint;

}break;

}

default:

{

printf("Thank you for using my Singly Linked List");

exit(0);

}

}

return 0;

}