//Program to implement stack using LL – Visakh Bobby – S3R2-34

#include<stdio.h>

#include<malloc.h>

#include<stdlib.h>

//In stack follows LIFO - last in , first out. so we insert and deleted from the end itself

struct node {

        int info;

        struct node\* link;

      }\*first=NULL,\*top=NULL;

void PUSH(int uinfo)

{

  //Here top is the last element of LL

  struct node \* ptr= (struct node\*) malloc(sizeof(struct node)); //dynamic intialization of top

  ptr->info = uinfo;

  ptr->link = NULL;

  if(first==NULL)

  {

    first = ptr;

   top= ptr;

  }

  else

  {

    top->link = ptr;

    top=ptr;

  }

}

void POP()

{ if(top==NULL)

  {

    printf("Underflow ,No Element Present In Stack\n");

    return;

  }

  else if(first == top)

  {

    int deld = top->info;

    top=NULL;

    first=NULL;

    printf(" Element %d has been deleted\n ",deld);

  }

  else

  {

    struct node \* ptr = (struct node\*) malloc(sizeof(struct node));

    ptr = first;

    while(ptr->link!=top) //we transverse upto the node before ptr

      ptr=ptr->link;

    int deld = top->info; //top element being deleted

    ptr->link = NULL;

    top=ptr;

    ptr=NULL;

    free(ptr);

    printf(" Element %d has been deleted\n ",deld);

  }

}

void Display()

{

  if(first==NULL)

  {

    printf("Stack Underflow Condition, No Element Present\n");

    return;

  }

  else

  { printf("The Element Of The Stack Are :\n");

    struct node \* ptr = (struct node\*) malloc(sizeof(struct node));

    ptr = first;

    while(ptr!=NULL)

    {

      printf("%d\t",ptr->info);

      ptr=ptr->link;

    }

    printf("<--Top\n");

  }

}

void main()

{

  int choice,uinfo;

  while(1)

  {

    printf("Enter Choice : 1. PUSH , 2. POP , 3. Display , 4. Exit \n ");

    scanf("%d" , &choice);

  switch(choice)

  {

    case 1 : printf("Enter Element to Be Inserted\n");

         scanf("%d",&uinfo);

         PUSH(uinfo);

         break;

    case 2 : POP();

         break;

    case 3 : Display();

         break;

    case 4 : exit(0);

    default : printf("Invalid Choice , Try Again\n");

         break;

  }

  }

}

**Output:**

