**Write a c program to implement two stacks in one array.**

**Program -**

// Write a c program to implement two stacks in one array.

#include <stdio.h>

#include<conio.h>

#include<stdlib.h>

#define MAX 10

struct stack{

int a[MAX];

int top,top2;

}s;

void push(int x);

void push2(int x);

void pop();

void pop2();

void display();

void display2();

int main()

{

int n, ch; // n= input number; ch= choice by the user

//clrscr();

s.top=-1; //stack is empty initially.

s.top2=MAX; //stack2 is empty.

printf("\n 1.Push in stack 1 \n 2.Pop from stack 1\n 3. Display stack 1\n 4.Push in stack 2 \n 5.Pop from stack 2\n 6. Display stack 2\n7.Exit\n");

while(1){

printf("\nPlease enter your choice - \n");

scanf("%d",&ch);

switch(ch)

{

case 1: if(s.top==(MAX/2)-1)

printf("!Stack 1 is full!\n");

else{

printf("\nPlease enter the number to be inserted - \n");

scanf("%d",&n);

push(n);

}

break;

case 2: if(s.top==-1)

printf("!Stack 1 is empty!\n");

else{

pop();

}

break;

case 3: display();

break;

case 4: if(s.top2==MAX/2)

printf("!Stack 2 is full!\n");

else{

printf("\nPlease enter the number to be inserted - \n");

scanf("%d",&n);

s.a[--s.top2]=n;

//printf("Top2 = %d",s.top);

//push2(n);

}

break;

case 5: if(s.top2==MAX)

printf("!Stack 2 is empty!\n");

else{

printf("The deleted element is - %d. \n", s.a[s.top2]);

s.top2++;

}

//pop2();

break;

case 6: if(s.top2==MAX){

printf("\nStack 2 is empty! No elements present.\n");

}

else{

printf("Stack 2 elements: ");

for(int i=s.top2; i<MAX; i++){

printf("%d\n",s.a[i]);

}

}

//display2();

break;

case 7: exit(0);

default: printf("\n\t !Invalid choice!");

}

}

return 0;

}

void push(int x){

s.a[++s.top]=x;

}

void pop(){

printf("The deleted element is - %d. \n", s.a[s.top]);

s.top--;

}

void display(){

if(s.top==-1){

printf("\nStack is empty! No elements present\n");

}

else{

printf("Stack 2 elements: ");

for(int i=s.top; i>=0; i--){

printf("%d\n",s.a[i]);

}

}

}

**Output –**

 

 