#include <stdio.h>

#include <stdlib.h>

struct node

{

int info;

struct node \*ptr;

}\*top,\*top1,\*temp;

int topelement();

void push(int data);

void pop();

void display();

void destroy();

void stack\_count();

void create();

int count = 0;

void main()

{

int no, ch, e,g;

create();

do

{

printf("\n 1 - Push");

printf("\n 2 - Pop");

printf("\n 3 - Top");

printf("\n 4 - Exit");

printf("\n 5 - Display");

printf("\n Enter choice : ");

scanf("%d", &ch);

switch (ch)

{

case 1:printf("Enter data : ");

scanf("%d", &no);

push(no);

break;

case 2:pop();

break;

case 3:if (top == NULL)

printf("No elements in stack");

else

{

e = topelement();

printf("\n Top element : %d", e);

}

break;

case 4:exit(0);

case 5:display();

break;

default :printf(" Wrong choice, Please enter correct choice ");

break;

}

printf("Do you want to continue?Enter 1 to continue.");

scanf("%d",&g);

}while(g ==1);

}

void create()

{ top = NULL; }

void push(int data)

{

if (top == NULL)

{

top =(struct node \*)malloc(1\*sizeof(struct node));

top->ptr = NULL;

top->info = data;

}

else

{

temp =(struct node \*)malloc(1\*sizeof(struct node));

temp->ptr = top;

temp->info = data;

top = temp;

}

count++;

}

void display()

{

top1 = top;

if (top1 == NULL)

{

printf("Stack is empty");

return;

}

while (top1 != NULL)

{

printf("%d ", top1->info);

top1 = top1->ptr;

}

}

void pop()

{

top1 = top;

if (top1 == NULL)

{

printf("\n Error : Trying to pop from empty stack");

return;

}

else

top1 = top1->ptr;

printf("\n Popped value : %d", top->info);

free(top);

top = top1;

count--;

}

int topelement()

{

return(top->info);

}