**Program to implement STACK using Arrays.**

#include<stdio.h>

#define MAX 5

int STACK[MAX];

int top = -1;

int isEmpty();

int isFull();

void push(int element);

void pop();

void peek();

void display();

int main() {

int choice, element;

printf("====MENU====\n");

printf("1.PUSH\n");

printf("2.POP\n");

printf("3.PEEK\n");

printf("4.DISPLAY\n");

printf("5.Exit\n");

while (choice != 5) {

printf("\nEnter your choice:");

scanf("%d", &choice);

switch (choice)

{

case 1:

printf("Enter the element to be added in the stack:");

scanf("%d", &element);

push(element);

break;

case 2:

pop();

break;

case 3:

peek();

break;

case 4:

display();

break;

case 5: printf("Exiting....\n");

break;

default:printf("Wrong Choice\n");

break;

}

}

return 0;

}

int isEmpty() {

if (top == -1) {

return 1;

}

else {

return 0;

}

}

int isFull() {

if (top == MAX -1) {

return 1;

}

else {

return 0;

}

}

void push(int element) {

if (isFull()) {

printf("STACK OVERFLOW\n");

}

else {

top = top + 1;

STACK[top] = element;

printf("Element added succesfully\n");

}

}

void pop() {

int element;

if (isEmpty()) {

printf("STACK UNDERFLOW\n");

}

else {

element = STACK[top];

top = top - 1;

printf("The element %d is sucessfully deleted\n",element);

}

}

void peek() {

int element;

if (isEmpty()) {

printf("STACK UNDERFLOW\n");

}

else {

printf("The top most element of the STACK is %d\n",STACK[top]);

}

}

void display() {

int i, element;

if (isEmpty()) {

printf("STACK is empty. \n");

}

else {

printf("Stack Elements: ");

for (i = top;i >= 0;i--)

printf("%d\t", STACK[i]);

printf("\n");

}

}

**Output:**

