S.No: 18	Exp. Name: ***Write a C program to implement different Operations on Stack using Arrays.***	Date:2023-06-14

Aim:

Write a C program to implement stack operations using **arrays**.

Source Code:

StackUsingArray.c

```
#include <stdio.h>
#include <stdlib.h>
#define STACK_MAX_SIZE 10
#include "StackOperations.c"

int main() {
   int op, x;
   while(1) {
      printf("1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit\n");
      printf("Enter your option : ");
      scanf("%d", &op);
      switch(op) {
         case 1:
            printf("Enter element : ");
            scanf("%d", &x);
            push(x);
            break;
         case 2:
            pop();
            break;
         case 3:
            display();
            break;
         case 4:
            isEmpty();
            break;
         case 5:
            peek();
            break;
         case 6:
            exit(0);
  }
 }
}
```

StackOperations.c

```
int stack[STACK_MAX_SIZE],top=-1;
void push(int x)
{
   if(top==STACK_MAX_SIZE-1){
      printf("Stack is overflow.\n");
 }  else
```


ID: 22K61A05H2

```
   {
      top++;
      stack[top]=x;
      printf("Successfully pushed.\n");
 }
}
void pop(){
   if(top==-1){
      printf("Stack is underflow.\n");
 }  else{
      printf("Popped value = %d\n",stack[top]);
      top--;
 }
}
void peek(){
   if(top==-1){
      printf("Stack is underflow.\n");
 }  else{
      printf("Peek value = %d\n",stack[top]);
 }
}
void display(){
   if(top==-1)
   {
      printf("Stack is empty.\n");
 }  else
   {
      printf("Elements of the stack are : ");
      for(int i=top;i>=0;i--){
         printf("%d ",stack[i]);
  }
      printf("\n");
 }
}
void isEmpty(){
   if(top==-1)
   {
      printf("Stack is empty.\n");
 }  else{
      printf("Stack is not empty.\n");
 }
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option : 1
Enter element : 10
Successfully pushed. 1
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option : 1
Enter element : 20

```
Successfully pushed. 1
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option :  1
Enter element :  30
Successfully pushed. 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option :  3
Elements of the stack are : 30 20 10  5
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 5
Enter your option :  5
Peek value = 30 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option :  2
Popped value = 30 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option :  2
Popped value = 20 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option :  3
Elements of the stack are : 10  5
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 5
Enter your option :  5
Peek value = 10 4
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 4
Enter your option :  4
Stack is not empty. 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option :  2
Popped value = 10 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option :  3
Stack is empty. 4
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 4
Enter your option :  4
Stack is empty. 6
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 6
Enter your option :  6
```