S.No: 17	Exp. Name: *Write a C program to implement different Operations on Stack using Array representation*	Date:2023-05-12

Aim:

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

```
Sample Input and Output:
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 4
    Stack is empty.
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 2
    Stack is underflow.
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 3
    Stack is empty.
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 5
    Stack is underflow.
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 1
    Enter element : 25
    Successfully pushed.
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 1
    Enter element : 26
    Successfully pushed.
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 3
    Elements of the stack are : 26 25
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 2
    Popped value = 26
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 4
    Stack is not empty.
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 5
    Peek value = 25
    1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit
    Enter your option : 6
```

Source Code:

StackUsingArray.c

```c
#include<stdio.h>
#include<stdlib.h>
#define STACK_MAX_SIZE 10
int arr[STACK_MAX_SIZE];
int top=-1;
void push(int element)
{
```

```c
    if(top==STACK_MAX_SIZE-1)
    {
        printf("Stack is overflow.\n");
    }
    else
    {
        top=top+1;
        arr[top]=element;
        printf("Successfully pushed.\n");
    }
}
void display()
{
    if(top<0)
    {
        printf("Stack is empty.\n");
    }
    else
    {
        printf("Elements of the stack are : ");
        for(int i=top;i>=0;i--)
        {
            printf("%d ",arr[i]);
        }
        printf("\n");
    }
}
void pop()
{
    int x;
    if(top<0)
    {
        printf("Stack is underflow.\n");
    }
    else
    {
        x=arr[top];
        top=top-1;
        printf("Popped value = %d\n",x);
    }
}
void peek()
{
    int x;
    if(top<0)
    {
        printf("Stack is underflow.\n");
    }
    else
    {
        x=arr[top];
        printf("Peek value = %d\n",x);
    }
}
void isEmpty()
{
```

```c
        if(top<0)
        {
            printf("Stack is empty.\n");
        }
        else
        {
            printf("Stack is not empty.\n");
        }
    }
    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);
            }
        }
    }
```

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
```