Experiment 1
Date 21-09-2022

STACK USING ARRAY

Aim

To write a menu driven C Program to implement a Stack using arrays with the operations:
a. Pushing elements to the Stack.
b. Popping elements from the Stack
c. Display the contents of the Stack after each operation

1 Stack Algorithm

1.1 Algorithm

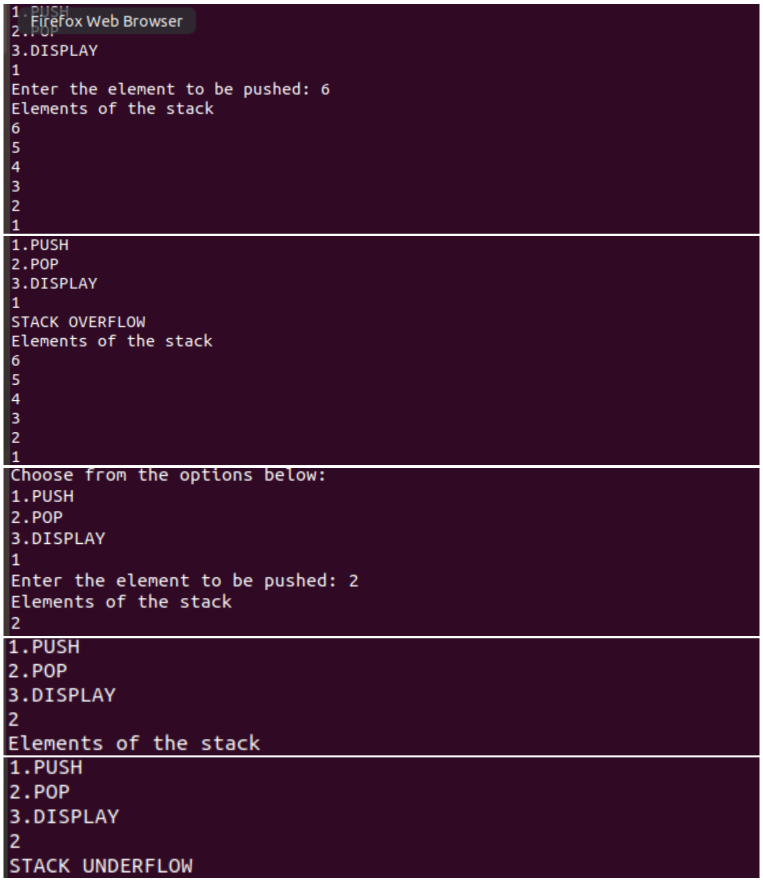
```
Step 1 : Start
Step 2 : Let top = -1
Step 3 : Input n
Step 4 : Print "Choose from the options below:  "
Step 5 : Print  "1. PUSH  2 POP 3. DISPLAY ’’
Step 6 : Input option
Step 7 : If option=1 then push () display()
Step 8 : If option = 2 then pop () display()
Step 9 : If option = 3 then display()
Step 10 : If op!= 1 or op!=2 or op!= 3 then print "Invalid Option’’ and go to step 4
Step 11 : Start of function push ()
Step 12   : Let int n
Step 13 : If top > =4 then print "STACK OVERFLOW" and go to Step 21
Step 14 : Input n
Step 15 : Let top <- top+1
Step 16 :  Let stack [top] = n and go to Step 21

Step 17 : Start of function int pop (stack)
Step 18 : If top = -1 then print "STACK UNDERFLOW" and go to Step 21
Step 19 : return stack[pop]
Step 20 : Let top <- top-1

Step 21 : Start of function display(stack, n)
Step 22 : If top = -1 then paint "Empty stack"  and go to step 4
Step 23: Let pos <-top
Step 24 : If pos = top then go to step 4
Step 25: Print stack[pos]
Step 26: Let pos <- pos+1  and go to Step 24
Step 27: Stop
```

1.2 Program

```
22-09-2022

#include<stdio.h>
#include<string.h>

int static stack[5],top=-1,pos=-1;
  void push()
  {
   int n;
   if(top<=4)
{printf("Enter the element to be pushed: ");
scanf("%d",&n);
top++;
stack[top]=n;
}
   else
printf("STACK OVERFLOW\n");


  }
  int pop()
  {
   if(top==-1)
printf("STACK UNDERFLOW\n");
  else
     {  top--;
return stack[top+1];
      }
   }
  void display()
{
```

```
pos=top;
printf("Elements of the stack\n");
  while(pos>=-0)
{
printf("%d\n",stack[pos]);
pos--;
}
}


void main()
{
int option;
char choice;
do{
printf("Choose from the options below: \n");
printf("1.PUSH\n2.POP\n3.DISPLAY\n");
scanf("%d",&option);
switch(option)
{
case 1:push();
       display();
       break;
case 2:pop();
       display();
       break;
case 3:display();
   break;
default:printf("Invalid Option\n");

}
printf("Do you want to continue:(y/n)");
scanf("%c",&choice);
}while(choice=='Y'||'y');



}
```

1.3 Sample Input and Output



```
1.PUSH
2.POP
3.DISPLAY
1
Enter the element to be pushed: 6
Elements of the stack
6
5
4
3
2
1
```

```
1.PUSH
2.POP
3.DISPLAY
1
STACK OVERFLOW
Elements of the stack
6
5
4
3
2
1
```

```
Choose from the options below:
1.PUSH
2.POP
3.DISPLAY
1
Enter the element to be pushed: 2
Elements of the stack
2
```

```
1.PUSH
2.POP
3.DISPLAY
2
Elements of the stack
```

```
1.PUSH
2.POP
3.DISPLAY
2
STACK UNDERFLOW
```

Figure 1: Output

1.4 Result

Implemented Stack using Array