28/08/2020 Program No 1 : Stack Using Array

Aim:

Write a C Program to implement stack using array.

Algorithm:

1. Start
2. Declare Top, stack[100] as global variables

PUSH(stack, max):

1. Declare a variable item which is to be inserted
2. check condition if top ≥ max
3. Print "Stack overflow, No element can be inserted".
4. Read the item to be inserted, item.
5. Increment Top by 1.
6. Stack [Top] = item

POP(stack):

1. check condition if top == -1
2. Print "Stack Underflow, no element to be deleted".
3. Declare and initialize item = Stack [Top]
4. Decrement Top by 1.
5. Print "Element has been deleted from stack".

Display (stack):

```
1. check condition if (top == -1)
2. Print "Stack is empty"
3. Declare and initialize i=top-
4. Print the stack elements
5. decrement i by 1.
6. Repeat steps 4-5 till i>=0
```

Main:

```
3. Assign top=-1
4. Declare the required variables Max, ch
5. Read the number of elements, n.
6. Read the choice from 1.PUSH. 2.POP, 3.DISPLAY, ch.
7. check condition if (ch == 1), if true → step 8, if false → step10
8. Execute PUSH (stack, Max)
9. Break
10. check condition if (ch == 2), if true → step 11, if false → step 13
11. Execute POP (stack, Max)
12. Break
13. check condition if (ch == 3), if true → step 14, if false → step 16.
14. Execute Display (stack, Max)
15. Break
16. Read the choices again : 1.PUSH      , ch.
                            2.POP
                            3. Display
                            4. Exit
17. Repeat steps 7-16 till (ch >= 1 && ch <= 3)
18. End.
```

Output :

```
Enter the maximum number of elements: 5
Enter choice:
1. PUSH
2. POP
3. DISPLAY

1
Enter item to be inserted:
100
Enter another choice from 1-3: 1. PUSH   2. POP
3. DISPLAY   4. Exit.
3
The elements in stack are:
100 ← Top
Enter another choice from 1-3: 1. PUSH   2. POP  3. DISPLAY
4. Exit.
2
Element 100 is deleted from stack
Enter another choice from 1-3: 1. PUSH  2. POP  3. DISPLAY
4. Exit
4.
```