**TITLE 34**

Write a C program to implement the STACK operation using array as a data structure. Users must be given the following choices to perform relevant tasks.

a. Push an element on to the STACK.

b. Pop and element from the STACK.

c. Peek the STACK.

d. Display the STACK.

e. Exit the program.

**OBJECTIVE:**

By the end of this program we will be able to implement STACK operation using an array.

**PROBLEM STATEMENT:**

In this problem we use STACK as an array. Input from user:

Enter the number of elements in the stack:

Once the input is entered and stored the output is printed.

**ALGORITHM:**

START

Define variables: i, j, choice, n, top

INPUT: Read from the keyboard

COMPUTATION: Compute stack as an array

DISPLAY: Displaying the choices the user wants to perform

STOP

**PROGRAM:**

#include <stdio.h>

int stack[100],i,j,choice=0,n,top=-1;

void push();

void pop();

void show();

void main ()

{

printf("Enter the number of elements in the stack:");

scanf("%d",&n);

printf("\*\*\*\*\*\*\*\*\*Stack operations using array\*\*\*\*\*\*\*\*\*");

printf("\n----------------------------------------------\n");

while(choice != 4)

{

printf("Choo

se one from the below options...\n");

printf("\n1.Push\n2.Pop\n3.Show\n4.Exit");

printf("\n Enter your choice \n");

scanf("%d",&choice);

switch(choice)

{

case 1:

{

push();

break;

}

case 2:

{

pop();

break;

}

case 3:

{

show();

break;

}

case 4:

{

printf("Exiting....");

break;

}

default:

{

printf("Please Enter valid choice ");

}

};

}

}

void push ()

{

int val;

if (top == n )

printf("\n Overflow");

else

{

printf("Enter the value:");

scanf("%d",&val);

top = top +1;

stack[top] = val;

}

}

void pop ()

{

if(top == -1)

printf("Underflow");

else

top = top -1;

}

void show()

{

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

{

printf("%d\n",stack[i]);

}

if(top == -1)

{

printf("Stack is empty");

}

}

**CONCLUSION:**

The simulation of the above C program helped me understand stacks in a better way and this will help me use stacks in other programs.

**OUTPUT:**

Enter the number of elements in the stack:3

Choose one from the below options…

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

1

Enter the value:5

Choose one from the below options…

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

1

Enter the value:6

Choose one from the below options…

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

1

Enter the value:7

Choose one from the below options…

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

3

5

6

7

Choose one from the below options…

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

4

Exiting….