

```
#include <stdio.h>

#define MAX 10

int stack[MAX];

int top == -1;

void push(int element)

{

    if(top ==MAX - 1 )

    {

        printf("Stack Overflow \n");

    }

    else

    {

        stack[top++]= element;

        printf("%d pushed to stack \n", element);

    }

}

void pop(int value)

{

    if(top == -1)

    {

        printf("Stack Underflow");

    }

    else

    {

        stack(top--)= value;

        printf("%d popped from stack \n", value);

    }

}
```

```
}

void display()

{

    if (top == -1)

    {

        printf("Stack is empty \n");

    }

    else

    {

        printf("stack elements(top to bottom)");

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

        {

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

        }

        printf("\n");

    }

}
```

```
int main()

{

    int element,options;

    while(1)

    {

        printf("****MAIN MENU***");

        printf("\n 1.push");

        printf("\n 2.pop");


```

```
printf("\n 3.Display");
printf(***EXIT***);
printf("Enter your Choice:");
scanf("%d",&option);
switch(option)
{
    case 1:
        printf("Enter value to push: ");
        scanf("%d",&element);
        push(element);
        break;
    case 2:
        pop();
        break;
    case 3:
        display();
        break;
    case 4:
        printf("Exiting program \n");
        return 0;
    default:
        printf("Invalid Option \n")
}
}
```

OUTPUT:

The screenshot shows a Windows desktop environment with a taskbar at the bottom featuring various icons. A single open window is displayed, titled 'C:\Users\Admin\Desktop\dsa.exe'. The window contains the following text, which represents a stack operation:

```
C:\Users\Admin\Desktop\dsa.exe
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 1
Enter value to push: 5
5 pushed to stack.

Menu:
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 2
5 popped from stack.

Menu:
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 3
Stack is empty.
```