**LAB # 09**

**Stack Implementation And Stack ADT Implementation**

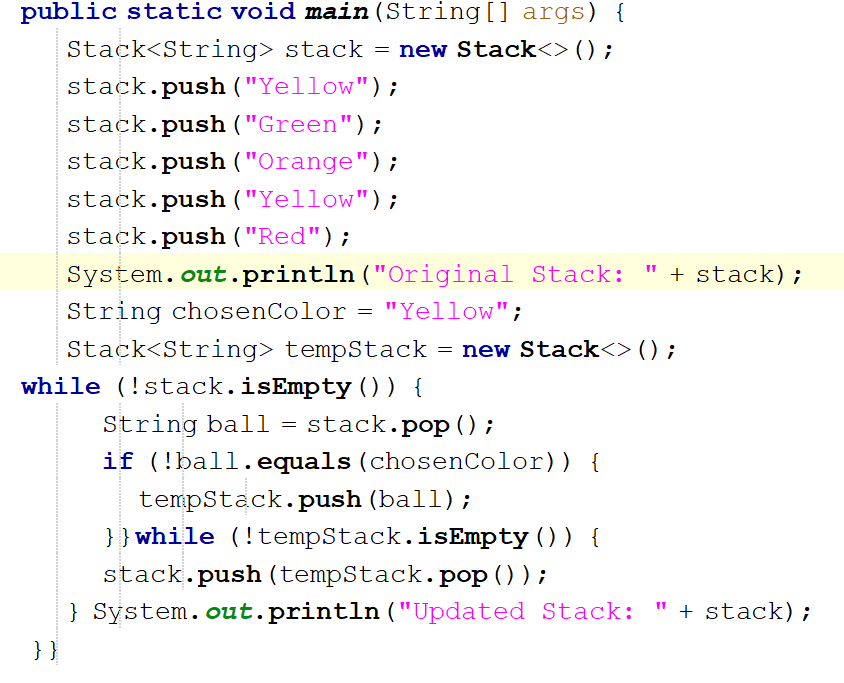
**OBJECTIVE :** To implement Stack as Arrays and linked list and Implement Stack by using Stack class.

**LAB TASK**

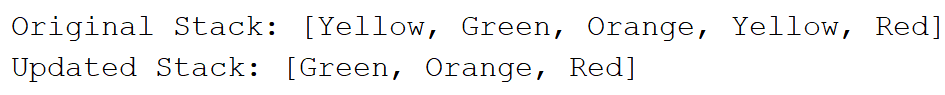
1. Write a program to create Stack (by using array) that can take input random color balls (i.e. red, yellow, orange, green) and store it in order. The user likes any one of the colored ball so he takes out all the balls, one by one, takes only the chosen color and keeps other in order so that he can return them to stack in exactly the same order as before minus the chosen ball. Print both the input and resultant stack. For example:

Original stack: Yellow->green->orange->yellow->red

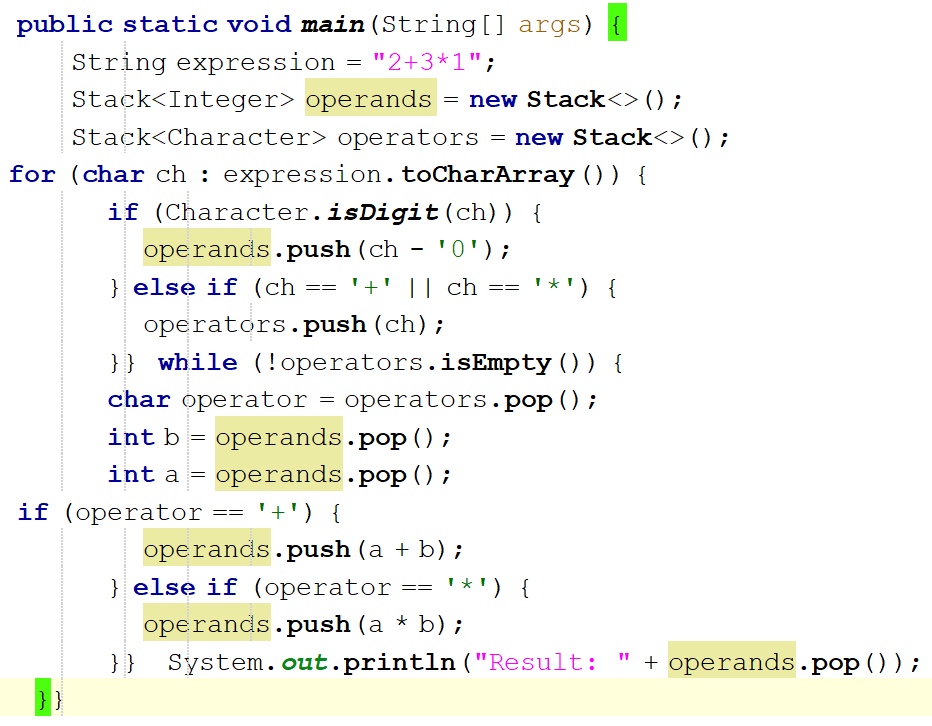
Updated stack: Green->orange->red i.e. yellow is removed



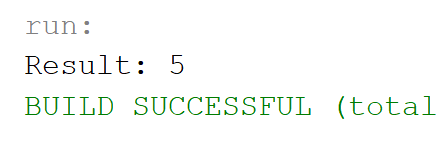
OUTPUT:



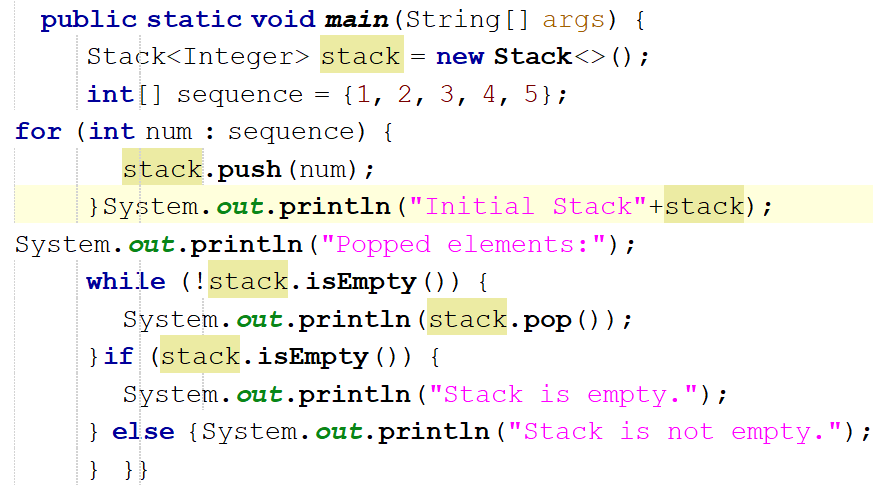
1. Write a program to implement stack (by using linked list) and insert and delete elements in a stack using its operations.



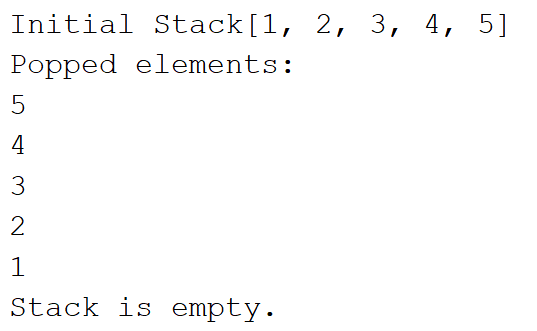
OUTPUT:



1. Write a program to create stack of 10 elements and perform five more operations on that stack.



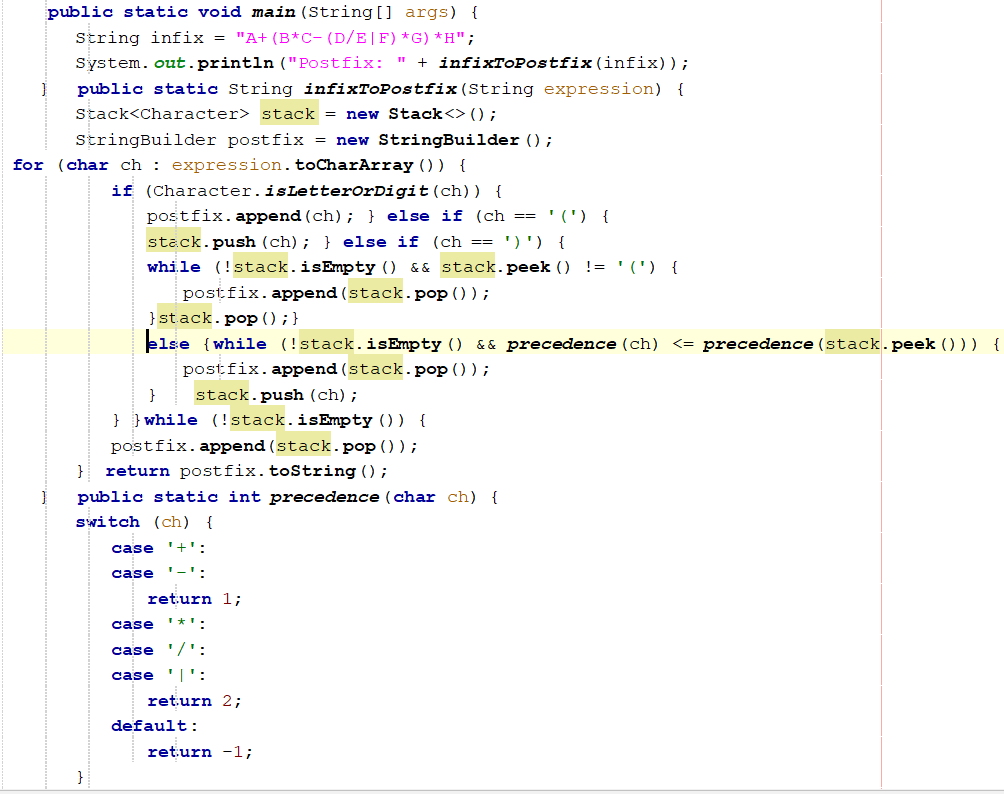
OUTPUT:



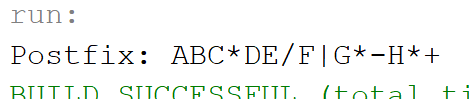
**HOME TASK**

1. Write a program to convert Infix expression into postfix expression by stack using linked list.

**A + ( B \* C - ( D / E | F ) \* G ) \* H**

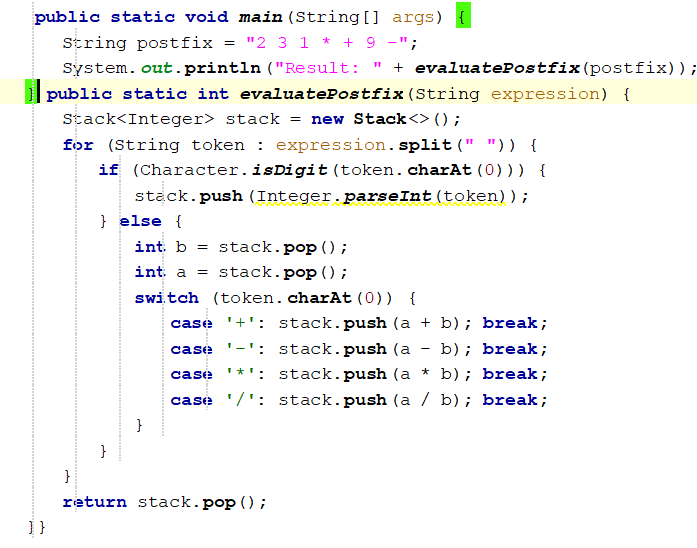
****

**OUTPUT:**

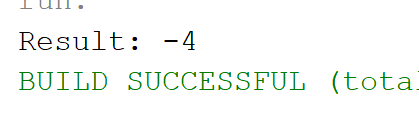
****

1. Write a program that evaluate following postfix expression by stack using array.

1. **3 1 \* + 9 –**

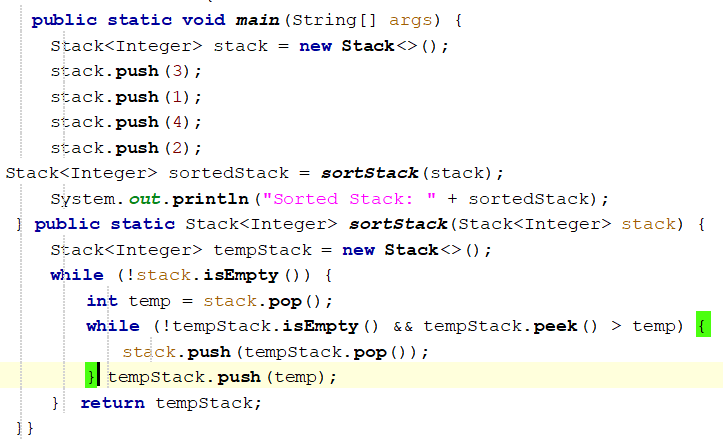
****

**OUTPUT:**

****

1. You are tasked with writing a program to check if a given string of brackets is balanced. A string is said to be balanced if:
2. Every opening bracket has a corresponding closing bracket.
3. Brackets close in the correct order.

The brackets used are: {, }, [, ], (, ).



**OUPUT:** 