

## Lab Assignment -1

**Q1.** Implement stack data structure. Stack is linear data structure where operations are performed at one end. It follows LIFO (Last In First Out) structure. That means that the last element in is the first element out.

Write modules in your program such that following operations can be performed on the stack-

- a.) **Push** : Inserts an element in the stack. Take an element as an input from the user and push this element on the top of the stack.
- b.) **Pop** : Removes an element from the stack.
- c.) **Peek** : Returns the top element of the stack.
- d.) **isEmpty** : Returns true if the stack is empty, else false.
- e.) **Display** : Prints the elements of the stack.

**Q2.** Given an expression consisting only of braces. A brace is any of the following character { , } , ( , ) , [ , or ] . Two brackets are considered to be a *matched pair* if the an opening bracket (i.e., ( , [ , or { ) occurs to the left of a closing bracket (i.e., ) , ] , or } ) *of the exact same type*. There are three types of matched pairs of brackets: [], {}, and ().

A matching pair of brackets is *not balanced* if the set of brackets it encloses are not matched. For example, {[(())]} is not balanced because the contents in between { and } are not balanced. The pair of square brackets encloses a single, unbalanced opening bracket, ( , and the pair of parentheses encloses a single, unbalanced closing square bracket, ] .

By this logic, we say a sequence of brackets is considered to be *balanced* if the following conditions are met:

- It contains no unmatched brackets.
- The subset of brackets enclosed within the confines of a matched pair of brackets is also a matched pair of brackets.

Given strings of brackets, determine whether each sequence of brackets is balanced. If a string is balanced, print YES on a new line; otherwise, print NO on a new line.

### Input Format:

The first line takes as input the number of test cases **N**, denoting the number of strings.

Each line **i** of the subsequent N lines consists of single string **s**, denoting sequence of braces.

### Output Format :

For each string **s**, output YES if the expression is balanced, else NO.

### Sample Input :

2

[(){}[(()){}]  
[(())]

### Sample output :

YES  
NO