HW3 : Due Friday

Use a stack to implement symbol matching.

The user of your program will enter a sequence of parenthesis and square brackets ([ ]).

The user will denote the end of the sequence with a .

Your program should push open brackets onto the stack. When a closed bracket is read, your program should first make sure the stack is not empty. If it is not, it should then make sure that the top of the stack matches the type of paren or bracket that is being closed. If it does not (or the stack was empty) then the sequence is not matched. After matching a closed bracket, the opening bracket is popped off the stack.

When the last input is read (a '.'), then the program should make sure the stack is not empty. If it is not then all symbols have not been matched. If it is then all symbols have been matched and the parens/brackets are balanced.