Open in Visual Studio Code

stack_brackets

Task. Your friend is making a text editor for programmers. He is currently working on a feature that will
find errors in the usage of different types of brackets. Code can contain any brackets from the set
[]{}(), where the opening brackets are [,{, and (and the closing brackets corresponding to them
are],}, and).
For convenience, the text editor should not only inform the user that there is an error in the usage
of brackets, but also point to the exact place in the code with the problematic bracket. First priority
is to find the first unmatched closing bracket which either doesn't have an opening bracket before it,
like] in](), or closes the wrong opening bracket, like } in ()[} . If there are no such mistakes, then
it should find the first unmatched opening bracket without the corresponding closing bracket after it,
like (in {}([]. If there are no mistakes, text editor should inform the user that the usage of brackets
is correct.
Apart from the brackets, code can contain big and small latin letters, digits and punctuation marks.
More formally, all brackets in the code should be divided into pairs of matching brackets, such that in
each pair the opening bracket goes before the closing bracket, and for any two pairs of brackets either
one of them is nested inside another one as in (foo[bar]) or they are separate as in f(a,b)-g[c].
The bracket [corresponds to the bracket], { corresponds to }, and (corresponds to).
Input Format. Input contains one string S which consists of big and small latin letters, digits, punctuation
marks and brackets from the set []{}().
Constraints. The length of S is at least 1 and at most 105.
Output Format. If the code in S uses brackets correctly, output "Success" (without

the quotes). Otherwise,
output the 1-based index of the first unmatched closing bracket, and if there are no unmatched closing
brackets, output the 1-based index of the first unmatched opening bracket.

Use the test files

Use an input to choose files or input - F or I
If input I, wait for another input to input the brackets.

Check that you output the results.

In case of success, output Success