Parenthesis Checker

Given an expression string **x**. Examine whether the pairs and the orders of “{“,”}”,”(“,”)”,”[“,”]” are correct in exp.
For example, the function should return 'true' for exp = “[()]{}{[()()]()}” and 'false' for exp = “[(])”.

Example 1:

Input:
{([])}
Output:
true
Explanation:
{ ([ ]) }. Same colored brackets can form
balaced pairs, with 0 number of
unbalanced bracket.

Example 2:

Input:
()
Output:
true
Explanation:
(). Same bracket can form balanced pairs,
and here only 1 type of bracket is
present and in balanced way.

Example 3:

Input:
([]
Output:
false
Explanation:
([]. Here square bracket is balanced but
the small bracket is not balanced and
Hence , the output will be unbalanced.

```
class Solution:

    #Function to check if brackets are balanced or not.
```

```
    def ispar(self,x):
        # code here

        stack = []
        if len(x)==1:
            return False
        for el in x:
            if el in ['(','{','[']:
                stack.append(el)
            elif el in [')','}',']']:
                if not stack:
                    return False
                temp = stack[-1]
                if (temp=='(' and el==')') or (temp=='[' and el==']') or
(temp=='{' and el=='}'):
                    stack.pop()
                else:
                    return False
        if len(stack):
            return False
        return True
```