# 5392. Maximum Score After Splitting a String

My Submissions (/contest/weekly-contest-186/problems/maximum-score-after-splitting-a-string/submissions/)

Back to Contest (/contest/weekly-contest-186/)

Given a string s of zeros and ones, return the maximum score after splitting the string into two **non-empty** substrings (i.e. **left** substring and **right** substring).

The score after splitting a string is the number of **zeros** in the **left** substring plus the number of **ones** in the **right** substring.

User Accepted:	5674
User Tried:	5942
Total Accepted:	5795
Total Submissions:	11166
Difficulty:	Easy

### Example 1:

```
Input: s = "011101"
Output: 5
Explanation:
All possible ways of splitting s into two non-empty subsleft = "0" and right = "11101", score = 1 + 4 = 5
left = "01" and right = "1101", score = 1 + 3 = 4
left = "011" and right = "101", score = 1 + 2 = 3
left = "0111" and right = "01", score = 1 + 1 = 2
left = "01110" and right = "1", score = 2 + 1 = 3
```

### Example 2:

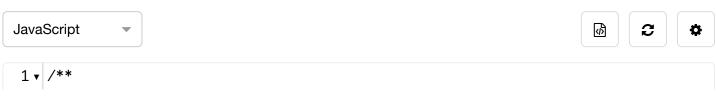
Input: s = "00111"
Output: 5
Explanation: When left = "00" and right = "111", we get the maximum score = 2 + 3 = 5

## Example 3:

Input: s = "1111"
Output: 3

#### **Constraints:**

- 2 <= s.length <= 500
- The string s consists of characters '0' and '1' only.



```
* @param {strina} s
 2
 3
     * @return {number}
 4
 5 v const maxScore = (s) => {
        let category = s.length - 1;
 6
 7
        let allScore = ☐;
 8
        let allCategory = new Map();
 9
10 ▼
        for (let i = 0; i < s.length - 1; i++) {
            allCategory.set(s.substring(0, i + 1), s.substring(i + 1, s.length));
11
12
        }
13
        let keys = allCategory.keys();
14
15
        for (const key of keys) {
16 ▼
            let eachScore = 0;
17
            for (const j of key) {
18 ▼
                 if (j == 0) {
19 ▼
20
                     eachScore++;
21
                 }
            }
22
23
            let value = allCategory.get(key);
24
25 ▼
            for (const j of value) {
                 if (i == 1) {
26 ▼
27
                     eachScore++;
28
29
30
            allScore.push(eachScore);
31
32
        return getMaxArr(allScore);
33
    };
34
35 v const getMaxArr = (arr) => {
36
        let max = arr[0];
        for (i = 1; i < arr.length; i++)
37
38
            if (arr[i] > max)
39
                max = arr[i];
40
        return max;
41
    };
```

Custom Testcase

Use Example Testcases

Run

**△** Submit

# Submission Result: Accepted (/submissions/detail/330205522/) ?

More Details ➤ (/submissions/detail/330205522/)

Share your acceptance!