# Problem 1422: Maximum Score After Splitting a String

## Problem Information

**Difficulty:** Easy
**Acceptance Rate:** 65.11%
**Paid Only:** No
**Tags:** String, Prefix Sum

## Problem Description

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.

**Example 1:**

**Input:** s = "011101" **Output:** 5 **Explanation:** All possible ways of splitting s into two non-empty substrings are: left = "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.

## Code Snippets

**C++:**

```cpp
class Solution {
public:
int maxScore(string s) {

}
};
```

**Java:**

```java
class Solution {
public int maxScore(String s) {

}
}
```

**Python3:**

```python
class Solution:
def maxScore(self, s: str) -> int:
```