**Maximum difference of zeros and ones in binary string :-**

Given a binary string of 0s and 1s. The task is to find the maximum difference of the number of 0s and the number of 1s (number of 0s – number of 1s) in the substrings of a string.

**Note:** In the case of all 1s, the answer will be -1.

**Example 1:**

**Input** : S = "11000010001"

**Output** : 6

**Explanatio:** From index 2 to index 9,

there are 7 0s and 1 1s, so number

of 0s - number of 1s is 6.

**Example 2:**

**Input:** S = "111111"

**Output:** -1

**Explanation:** S contains 1s only

**Your task:**  
You do not need to read any input or print anything. The task is to complete the function **maxSubstring()**, which takes a string as input and returns an integer.

**Expected Time Complexity:** O(|S|)  
**Expected Auxiliary Space:** O(|S|)

**Constraints:**  
1 ≤ |S| ≤ 105  
S contains 0s and 1s only