Strings

Q. Split the Binary string into two substrings with equal 0's and 1's.

Given a binary string (s) of length N, the main task here is to find the max. count of consecutive substrings (s) can be divided into such that all the substrings are balanced i.e., they have equal number of 0s and 1s.

If it is not possible to split (s) satisfying the conditions the print -1

In the program;

① Initialize count0 and count 1 to keep track occurrences of 0's and 1's. ~~in the given~~ It traverse the string (s) character by character.

② When the count of 0's and 1's become equal increment the count.

③ If it is not possible to split string (s) then on that time count of 0's must not be equal to count of 1's then return -1.

④ If possible print the value of count after the traversal of the complete string.

The given input is:

binary_str = "0100110101"

Step-by-step tracking:

Index	Character	Count 0	Count 1	Balanced	Substring
0	0	1	0	No	—
1	1	1	1	Yes	"01"
2	0	1	0	No	—
3	0	2	0	No	—
4	1	2	1	No	—
5	1	2	2	Yes	"0011"
6	0	1	0	No	—
7	1	1	1	Yes	"01"
8	0	1	0	No	—
9	1	1	1	Yes	"01"

Time complexity is $O(n)$
Space complexity is $O(1)$