Problem 1542: Find Longest Awesome Substring

Problem Information

Difficulty: Hard
Acceptance Rate: 46.01%
Paid Only: No
Tags: Hash Table, String, Bit Manipulation

Problem Description

You are given a string `s`. An **awesome** substring is a non-empty substring of `s` such that we can make any number of swaps in order to make it a palindrome.

Return _the length of the maximum length**awesome substring** of_ `s`.

Example 1:

Input: s = "3242415" **Output:** 5 **Explanation:** "24241" is the longest awesome substring, we can form the palindrome "24142" with some swaps.

Example 2:

Input: s = "12345678" **Output:** 1

Example 3:

Input: s = "213123" **Output:** 6 **Explanation:** "213123" is the longest awesome substring, we can form the palindrome "231132" with some swaps.

Constraints:

* `1 <= s.length <= 105` * `s` consists only of digits.

Code Snippets

C++:

```
class Solution {
public:
int longestAwesome(string s) {

}
};
```

Java:

```
class Solution {
public int longestAwesome(String s) {

}
}
```

Python3:

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