Problem 1417: Reformat The String

Problem Information

Difficulty: Easy
Acceptance Rate: 52.11%
Paid Only: No
Tags: String

Problem Description

You are given an alphanumeric string `s`. (**Alphanumeric string** is a string consisting of lowercase English letters and digits).

You have to find a permutation of the string where no letter is followed by another letter and no digit is followed by another digit. That is, no two adjacent characters have the same type.

Return _the reformatted string_ or return **an empty string** if it is impossible to reformat the string.

Example 1:

Input: s = "a0b1c2" **Output:** "0a1b2c" **Explanation:** No two adjacent characters have the same type in "0a1b2c". "a0b1c2", "0a1b2c", "0c2a1b" are also valid permutations.

Example 2:

Input: s = "leetcode" **Output:** "" **Explanation:** "leetcode" has only characters so we cannot separate them by digits.

Example 3:

Input: s = "1229857369" **Output:** "" **Explanation:** "1229857369" has only digits so we cannot separate them by characters.

Constraints:

* `1 <= s.length <= 500` * `s` consists of only lowercase English letters and/or digits.

Code Snippets

C++:

```cpp
class Solution {
public:
string reformat(string s) {

}
};
```

Java:

```java
class Solution {
public String reformat(String s) {

}
}
```

Python3:

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