Problem 937: Reorder Data in Log Files

Problem Information

Difficulty: Medium
Acceptance Rate: 56.87%
Paid Only: No
Tags: Array, String, Sorting

Problem Description

You are given an array of `logs`. Each log is a space-delimited string of words, where the first word is the **identifier**.

There are two types of logs:

* **Letter-logs** : All words (except the identifier) consist of lowercase English letters. * **Digit-logs** : All words (except the identifier) consist of digits.

Reorder these logs so that:

1. The **letter-logs** come before all **digit-logs**. 2. The **letter-logs** are sorted lexicographically by their contents. If their contents are the same, then sort them lexicographically by their identifiers. 3. The **digit-logs** maintain their relative ordering.

Return _the final order of the logs_.

Example 1:

Input: logs = ["dig1 8 1 5 1","let1 art can","dig2 3 6","let2 own kit dig","let3 art zero"]
Output: ["let1 art can","let3 art zero","let2 own kit dig","dig1 8 1 5 1","dig2 3 6"]
Explanation: The letter-log contents are all different, so their ordering is "art can", "art zero", "own kit dig". The digit-logs have a relative order of "dig1 8 1 5 1", "dig2 3 6".

Example 2:

Input: logs = ["a1 9 2 3 1","g1 act car","zo4 4 7","ab1 off key dog","a8 act zoo"] **Output:** ["g1 act car","a8 act zoo","ab1 off key dog","a1 9 2 3 1","zo4 4 7"]

Constraints:

* `1 <= logs.length <= 100` * `3 <= logs[i].length <= 100` * All the tokens of `logs[i]` are separated by a **single** space. * `logs[i]` is guaranteed to have an identifier and at least one word after the identifier.

Code Snippets

C++:

```
class Solution {
public:
vector<string> reorderLogFiles(vector<string>& logs) {

}
};
```

Java:

```
class Solution {
public String[] reorderLogFiles(String[] logs) {

}
}
```

Python3:

```
class Solution:
def reorderLogFiles(self, logs: List[str]) -> List[str]:
```