# Partition Labels

You are given a string s. We want to partition the string into as many parts as possible so that each letter appears in at most one part. For example, the string "ababcc" can be partitioned into ["abab", "cc"], but partitions such as ["aba", "bcc"] or ["ab", "ab", "cc"] are invalid.

Note that the partition is done so that after concatenating all the parts in order, the resultant string should be s.

Return *a list of integers representing the size of these parts*.

**Example 1:**

**Input:** s = "ababcbacadefegdehijhklij"

**Output:** [9,7,8]

**Explanation:**

The partition is "ababcbaca", "defegde", "hijhklij".

This is a partition so that each letter appears in at most one part.

A partition like "ababcbacadefegde", "hijhklij" is incorrect, because it splits s into less parts.

**Example 2:**

**Input:** s = "eccbbbbdec"

**Output:** [10]

**Constraints:**

- 1 <= s.length <= 500
- s consists of lowercase English letters.