

Approach

The first thing we should do is find out where each character appears for the last time. This helps us determine the boundaries of a partition dynamically while iterating through the string.

We start by scanning the string to record the last occurrence of each character in an index array. This helps us determine how far we must extend a partition to fully include any character we encounter.

Now, we use two pointers:

- One pointer (partitionEnd) keeps track of the farthest point we need to reach for the current partition.
- The other pointer (partitionStart) marks where the current partition begins.

As we iterate through the string, we keep extending partitionEnd to the maximum last occurrence of any character encountered. Once we reach partitionEnd, we finalize the partition and store its size. Then, we update partitionStart for the next partition.

Once we reach the end of this boundary, we record the partition size and move on to the next segment. By the end, we obtain the possible valid partitions, ensuring that no character appears in more than one.

This visualization demonstrates the step-by-step execution of the partitionLabels algorithm with the input string: **"bobhaspepper"**

Step 1: Compute Last Occurrences

The first loop stores the last index of each character in the string:

Character	'b'	'o'	'h'	'a'	's'	'p'	'e'	'r'
Last Index	2	1	3	4	5	9	10	11

Step 2: Find Partitions

Now, we iterate through the string, maintaining partitionEnd and partitionStart.

i	s[i]	Last Index	partitionEnd	Is Partition End?	Partition Size	New Start
0	'b'	2	2	No	-	-
1	'o'	1	2	No	-	-
2	'b'	2	2	Yes	3	3
3	'h'	3	3	Yes	1	4
4	'a'	4	4	Yes	1	5
5	's'	5	5	Yes	1	6
6	'p'	9	9	No	-	-
7	'e'	10	10	No	-	-
8	'p'	9	10	No	-	-
9	'p'	9	10	No	-	-
10	'e'	10	10	Yes	5	11
11	'r'	11	11	Yes	1	12

Visualization of Partitions

b	o	b	h	a	s	p	e	p	p	e	r
0	1	2	3	4	5	6	7	8	9	10	11

Final Output

[3, 1, 1, 1, 5, 1]

This means the partitions of "bobhaspepper" are:

- "bob" → Length = 3
- "h" → Length = 1
- "a" → Length = 1
- "s" → Length = 1
- "peppe" → Length = 5
- "r" → Length = 1