Problem 418: Sentence Screen Fitting

Problem Information

Difficulty: Medium
Acceptance Rate: 36.36%
Paid Only: Yes
Tags: Array, String, Dynamic Programming

Problem Description

Given a `rows x cols` screen and a `sentence` represented as a list of strings, return _the number of times the given sentence can be fitted on the screen_.

The order of words in the sentence must remain unchanged, and a word cannot be split into two lines. A single space must separate two consecutive words in a line.

Example 1:

Input: sentence = ["hello","world"], rows = 2, cols = 8 **Output:** 1 **Explanation:** hello--- world--- The character '-' signifies an empty space on the screen.

Example 2:

Input: sentence = ["a", "bcd", "e"], rows = 3, cols = 6 **Output:** 2 **Explanation:** a-bcd- e-a--- bcd-e- The character '-' signifies an empty space on the screen.

Example 3:

Input: sentence = ["i","had","apple","pie"], rows = 4, cols = 5 **Output:** 1 **Explanation:** i-had apple pie-i had-- The character '-' signifies an empty space on the screen.

Constraints:

* `1 <= sentence.length <= 100` * `1 <= sentence[i].length <= 10` * `sentence[i]` consists of lowercase English letters. * `1 <= rows, cols <= 2 * 104`

Code Snippets

C++:

```
class Solution {
public:
int wordsTyping(vector<string>& sentence, int rows, int cols) {

}
};
```

Java:

```
class Solution {
public int wordsTyping(String[] sentence, int rows, int cols) {

}
}
```

Python3:

```
class Solution:
def wordsTyping(self, sentence: List[str], rows: int, cols: int) -> int:
```