

127. Word Ladder

Hard  Topics  Companies

A **transformation sequence** from word `beginWord` to word `endWord` using a dictionary `wordList` is a sequence of words `beginWord ->`

- Every adjacent pair of words differs by a single letter.
- Every s_i for $1 \leq i \leq k$ is in `wordList`. Note that `beginWord` does not need to be in `wordList`.
- $s_k == endWord$

Given two words, `beginWord` and `endWord`, and a dictionary `wordList`, return *the number of words in the shortest transformation sequence*

Example 1:

Input: `beginWord = "hit", endWord = "cog", wordList = ["hot","dot","dog","lot","log","cog"]`
Output: 5
Explanation: One shortest transformation sequence is "hit" -> "hot" -> "dot" -> "dog" -> "cog", which

Example 2:

Input: `beginWord = "hit", endWord = "cog", wordList = ["hot","dot","dog","lot","log"]`
Output: 0
Explanation: The `endWord "cog"` is not in `wordList`, therefore there is no valid transformation sequence.

Constraints:

- $1 \leq beginWord.length \leq 10$
- $endWord.length == beginWord.length$
- $1 \leq wordList.length \leq 5000$
- $wordList[i].length == beginWord.length$
- `beginWord`, `endWord`, and `wordList[i]` consist of lowercase English letters.
- `beginWord != endWord`
- All the words in `wordList` are **unique**.

Seen this question in a real interview before? 1/4

Yes No

Accepted 1M Submissions 2.6M Acceptance Rate 38.7%

 Topics

 Companies

 Similar Questions

 Discussion (119)