Problem 745: Prefix and Suffix Search

Problem Information

Difficulty: Hard
Acceptance Rate: 40.63%
Paid Only: No
Tags: Array, Hash Table, String, Design, Trie

Problem Description

Design a special dictionary that searches the words in it by a prefix and a suffix.

Implement the `WordFilter` class:

* `WordFilter(string[] words)` Initializes the object with the `words` in the dictionary. * `f(string pref, string suff)` Returns _the index of the word in the dictionary,_ which has the prefix `pref` and the suffix `suff`. If there is more than one valid index, return **the largest** of them. If there is no such word in the dictionary, return `-1`.

Example 1:

Input ["WordFilter", "f"] [[["apple"]], ["a", "e"]] **Output** [null, 0] **Explanation** WordFilter wordFilter = new WordFilter(["apple"]); wordFilter.f("a", "e"); // return 0, because the word at index 0 has prefix = "a" and suffix = "e".

Constraints:

* `1 <= words.length <= 104` * `1 <= words[i].length <= 7` * `1 <= pref.length, suff.length <= 7` * `words[i]`, `pref` and `suff` consist of lowercase English letters only. * At most `104` calls will be made to the function `f`.

Code Snippets

C++:

```
class WordFilter {
public:
WordFilter(vector<string>& words) {

}

int f(string pref, string suff) {

}
};

/**
* Your WordFilter object will be instantiated and called as such:
* WordFilter* obj = new WordFilter(words);
* int param_1 = obj->f(pref,suff);
*/
```

Java:

```
class WordFilter {

public WordFilter(String[] words) {

}

public int f(String pref, String suff) {

}
}

/**
* Your WordFilter object will be instantiated and called as such:
* WordFilter obj = new WordFilter(words);
* int param_1 = obj.f(pref,suff);
*/
```

Python3:

```
class WordFilter:

def __init__(self, words: List[str]):
```

```
def f(self, pref: str, suff: str) -> int:



# Your WordFilter object will be instantiated and called as such:
# obj = WordFilter(words)
# param_1 = obj.f(pref,suff)
```