20. Short Encoding of Words

DescriptionHintsSubmissionsDiscussSolution

Given a list of words, we may encode it by writing a reference string `S` and a list of indexes `A`.

For example, if the list of words is `["time", "me", "bell"]`, we can write it as `S = "time#bell#"` and `indexes = [0, 2, 5]`.

Then for each index, we will recover the word by reading from the reference string from that index until we reach a "#" character.

What is the length of the shortest reference string S possible that encodes the given words?

Example:

```
Input: words = ["time", "me", "bell"]
Output: 10
Explanation: S = "time#bell#" and indexes = [0, 2, 5].
```

Note:

1. `1 <= words.length <= 2000.`
2. `1 <= words[i].length <= 7.`
3. Each word has only lowercase letters.

Seen this question in a real interview before?

- Subscribe to see which companies asked this question.

C++

Reverse the string first, Then sort the string vector by string length

check whether the current string is a substr of the rests.

If it is, the current string is skipped for length compute

finally, append ”#” for each word to concat a new string

```
class Solution {
public:
```

```
    int minimumLengthEncoding(vector<string>& words) {
        vector<string> w;
        for(auto word:words)
        {
            reverse(word.begin(),word.end());
           w.push_back(word);
        }
        sort(w.begin(),w.end());
        for(int i=0;i<words.size();++i)
        {
            for(int j=i+1;j<words.size();++j)
            {
                if(w[i]==w[j].substr(0,w[i].size()))
                {
                    w[i]="";
                    break;
                }
            }
        }
        int ret=0;
        for(string word:w)
        {
            if(word=="") continue;
            ret += word.size()+1;
        }
        return ret;
    }
};
```