

At Jet.com, each customer is guaranteed quick service since there are always enough merchants available to fulfill a given order. To keep the number of available merchants high, Jet tries to minimize the number of merchants required to fulfill a given order.

Consider a list of items the customer has ordered, and a list of available merchants. For each merchant it is known which items they have in their inventory. Calculate which merchants should fulfill the order so that all items are accounted for and the number of merchants is kept as small as possible. Return the answer as a 0-based array of merchant indices. If there are several possible answers, return the **lexicographically smallest** one. If there is no answer, return `[-1]` instead.

### Example

```
For items = ["apples", "bananas", "kiwis"] and
merchants = [["apples", "pineapples"],
              ["apples", "kiwis"],
              ["kiwis", "papayas", "bananas"],
              ["bananas", "apples"]]
```

the output should be

```
merchantMinimization(items, merchants) = [0, 2].
```

In this case it's impossible to use a single available merchant to fulfill the entire order. However, there are several sets of 2 merchants that could work:

- merchants 1 and 2;
- merchants 1 and 3;
- merchants 2 and 3;
- merchants 0 and 2.

The last option forms the lexicographically smallest array, thus it is the correct answer.

### Input/Output

- **[time limit] 4000ms (py)**

- **[input] array.string items**

Array of items in the order. It is guaranteed that each item name is unique.

*Constraints:*

$1 \leq \text{items.length} \leq 12$ ,

$1 \leq \text{items}[i].\text{length} \leq 15$ .

- **[input] array.array.string merchants**

The list of available merchants. For each valid  $i$ , `merchants[i]` is an array of different strings representing items which the  $i^{\text{th}}$  merchant can fulfill.

*Constraints:*

$3 \leq \text{merchants.length} \leq 10$ ,  
 $1 \leq \text{merchants}[i].\text{length} \leq 7$ ,  
 $1 \leq \text{merchants}[i][j].\text{length} \leq 15$ .

- **[output] array.integer**

The indices of the group of merchants that should fulfill the order  
or `[-1]` if there is no answer.