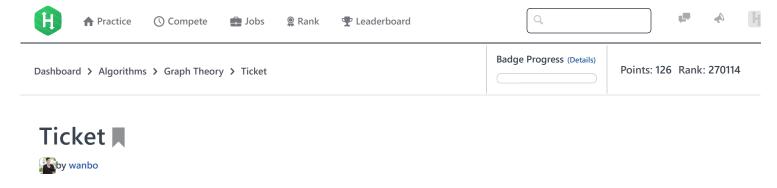
15/11/2017 HackerRank



There are n people at the railway station, and each one wants to buy a ticket to go to one of k different destinations. The n people are in a queue.

Discussions

Editorial A

There are m ticket windows from which tickets can be purchased. The n people will be distributed in the windows such that the order is maintained. In other words, suppose we number the people 1 to n from front to back. If person i and person j go to the same window and i < j, then person i should still be ahead of person j in the window.

Each ticketing window has an offer. If a person in the queue shares the same destination as the person immediately in front of him/her, a 20% reduction in the ticket price is offered to him/her.

For example, suppose there are $\bf 3$ people in the queue for a single ticket window, all with the same destination which costs $\bf 10$ bucks. Then the first person in the queue pays $\bf 10$ bucks, and the 2nd and 3rd persons get a discount of 20% on $\bf 10$ bucks, so they end up paying $\bf 8$ bucks each instead of $\bf 10$ bucks.

Try to distribute the n people across the m windows such that the total cost S paid by all n people is minimized.

Leaderboard

Input Format

Problem

Submissions

The first line contains 3 integers:

- **n** is the number of people
- **m** is the number of ticket windows
- $oldsymbol{k}$ is the number of destinations separated by a single space (in the same order)

Then k lines follow. The $i^{ ext{th}}$ line contains an alphanumeric string $ext{place}_i$ and an integer $ext{price}_i$:

- $place_i$ is the i^{th} destination
- **price**_i is the ticket price for **place**_i

Then n lines follow. The i^{th} line contains an alphanumeric string $destination_i$ which is the destination of the i^{th} person.

Constraints

- $1 \le n \le 500$
- $1 \le m \le 10$
- $1 \le k \le 100$
- ullet The $oldsymbol{k}$ available destinations have nonempty and distinct names.
- ullet Each person's destination appears in the list of $m{k}$ available destinations.
- $0 \le \operatorname{price}_i \le 100$

Output Format

Output n+1 lines. The first line contains S, the total cost that is to be minimized. In the i^{th} following line, print the ticket window which the i^{th} person goes to. The windows are indexed 1 to m. There may be multiple ways to distribute the people among the windows such that the total cost is

15/11/2017 HackerRank

minimized; any one will be accepted.

The answer S will be accepted if it is within an error of 10^{-3} of the true answer.

Sample Input

```
5 2 3
CALIFORNIA 10
HAWAII 8
NEWYORK 12
NEWYORK
NEWYORK
CALIFORNIA
NEWYORK
HAWAII
```

Sample Output

```
49.2
1
1
2
1
```

Explanation

At the beginning, all the people are in the same queue, and will go to the ticket windows one by one in the initial order.

- {1, 2, 4, 5} will buy ticket in the first window.
- {3} will buy ticket in the second window.

In the first ticket window, #1 will pay 12 bucks to go to NEWYORK, and #2 and #4 have the same destination with the person in front of them, so they will get 20% off, and will pay 9.6 bucks each. #5 has a different destination, so it will cost him 8 bucks to go to HAWAII.

In the second ticket window, #3 will pay ${f 10}$ bucks to go to CALIFORNIA .

```
f in Solved score: 100.00pts
Submissions:305
Max Score:100
Difficulty: Expert
Rate This Challenge:
☆☆☆☆☆
```

```
Current Buffer (saved locally, editable) & • •
                                                                                          Java 7
1 ▼ import java.io.*;
2 import java.util.*;
   import java.text.*;
3
   import java.math.*;
   import java.util.regex.*;
7 ▼ public class Solution {
8
9 ▼
        public static void main(String[] args) {
10 ▼
            /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11
   }
12
```

15/11/2017 HackerRank

Line: 1 Col: 1

1 Upload Code as File

Test against custom input

Run Code

Submit Code

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature