[ML] Identifying Spammers

Problem ID: spam

One issue that social platforms like Quora need to deal with is spam. A potential approach we could leverage to identify users producing spam (spammers) is to analyze the visit patterns. Normal users will tend to act differently from spammers, so this can help us identify and ban users producing this unwanted content. In this problem, your task is to determine which users are producing spam content given the list of `page_type:time` pairs from the users' visits to Quora. You will be given a training set of previous normal and spam user visit lists and labels of whether the user was a spammer. You need to predict the labels of users in a test set.

Notes

- The datasets for this problem are synthetically generated and **not** real user data. Therefore, they may not match intuitions on how spammers behave. In particular, we have modeled users as state machines with each `page_type` as a separate state. Normal users and spam users will have differing transition probability matrices, and the time each type of user spends on a page will be pulled from different distributions.
- To help test before submitting, a sample input is provided as an attachment to this problem.

Input

Your program will receive input from standard input.

The first row will be two space-separated integers n and m, representing the number of training and test samples.

Then, n lines will follow, each with up to v_{max} space-separated pairs of integers representing n users' visits to Quora. Let $p_{i,j} : t_{i,j}$ be the j-th pair of integers on the i-th of these n lines. This represents that user i's j-th visit to Quora was on page $p_{i,j}$ at time $t_{i,j}$.

Following that, there will be n lines, each with a single integer s_i representing whether the i-th training line was a spammer, where 1 represents that this user is a spammer, while 0 represents a normal user.

Finally, m lines of test data are provided, each with up to v_{max} space-separated pairs of integers, with the same format as the training data.

Output

Your program should write to standard output.

Print m lines representing the labels for each of the test users, where 1 represents that this user is a spammer, while 0 represents a normal user.

Constraints

- Train and test data are generated from the same distribution for each test case.
- $5 \cdot 10^3 \leq n \leq 3 \cdot 10^4$
- $10^3 \leq m \leq 5 \cdot 10^3$
- $0 \leq p_{i,j} \leq 12$
- $0 \leq t_{i,j} \leq 2^{31}$
- If $j_1 < j_2$, then $t_{i,j_1} < t_{i,j_2}$
- $v_{\text{max}} = 200$
- $0.05 \leq \frac{\sum_i s_i}{n} \leq 0.3$

Scoring

Score will be based on the F_1 score of predictions on the test set compared to the F_1 score of simply outputting all 1s.

$$F_1 = \frac{2}{\text{recall}^{-1} + \text{precision}^{-1}}$$

A 70% improvement over this baseline F_1 score will net full points, and scores in between will award points proportionally.

For example, if the dataset contains 30% positive labels, an F_1 score 0.4615 or below will award 0 points, while a score of 0.7846 or greater will score full points.

Sample Explanation

In this example, there are 11 training and 5 test users. There are two spam entries in the training set which are spammers, the 4th and 5th users. In the test set, a single user is a spammer, the 5th user. In this simple example, the spammers are easily identified as they tend land on a different page initially (page type 1).

Sample Input 1

```
11 5
2:0 2:14669 3:29642 1:41182 1:48298 3:53637
3:0
2:0 3:14874
1:0 0:0 1:5747
1:0 3:2049 2:12642 2:30991 2:49382 2:67816 0:86316 3:92663 0:102103 3:109224
2:0
2:0 3:14640
2:0 0:14974 1:17997 2:28943 0:43528 2:53765 1:68575 3:78493 1:89310 3:100206 3:110391
2:0 0:14995
2:0 1:14685 3:21289 1:32281 3:39379 2:52504 2:67318 1:82063 2:88576 1:103469
2:0 2:14872 2:29748 0:44401
0
0
0
1
1
0
0
0
0
0
0
2:0 1:18415 3:27095 2:37543 0:55935
3:0 2:7977 2:22721 0:37463 2:45641 2:60371 0:75187 2:82992 0:97862 1:102783 3:110944
3:0 2:11921 0:26812 2:29149
2:0
1:0 3:2794 1:13316 3:23960 0:33973 0:41125
```

Sample Output 1

```
0
0
0
0
1
```