USA Computing Olympiad

OVERVIEW TRAINING CONTESTS HISTORY STAFF RESOURCES

USACO 2022 DECEMBER CONTEST, BRONZE
PROBLEM 3. REVERSE ENGINEERING

Return to Problem List

Contest has ended.

Submitted; Results below show the outcome for each judge test case

1	*	31.9mb	191ms
2	*	31.9mb	203ms
3	*	31.9mb	185ms
4	*	31.9mb	187ms
5	*	31.9mb	185ms
6	*	33.4mb	275ms
7	*	34.1mb	284ms
8	*	33.3mb	269ms
9	*	33.0mb	270ms
10	*	32.9mb	268ms
11	*	32.9mb	246ms
12	*	33.1mb	280ms

English (en)

Elsie has a program that takes as input an array of N ($1 \le N \le 100$) variables $b[0], \dots, b[N-1]$, each equal to zero or one, and returns the result of applying a sequence of if / else if / else statements on the input. Each statement examines the value of at most one input variable, and returns either zero or one. An example of such a program might be:

```
if (b[1] == 1) return 1;
else if (b[0] == 0) return 0;
else return 1;
```

For example, if the input to the program above is "10" (that is, $b[0] = 1$ and $b[1] = 0$), then the output should be 1.

Elsie has told Bessie the correct output for M ($1 \le M \le 100$) different inputs. Bessie is now trying to reverse engineer Elsie's program. Unfortunately, Elsie might have lied; it may be the case that no program of the form above is consistent with what Elsie said.

For each of T ($1 \le T \le 10$) test cases, determine whether Elsie must be lying or not.

INPUT FORMAT (input arrives from the terminal / stdin):

The first line contains T, the number of test cases.

Each test case starts with two integers N and M, followed by M lines, each containing a string of N zeros and ones representing an input (that is, the values of $b[0] \dots b[N-1]$) and an additional character (zero or one) representing the output. Consecutive test cases are separated by newlines.

OUTPUT FORMAT (print output to the terminal / stdout):

For each test case, output "OK" or "LIE" on a separate line.

SAMPLE INPUT:

```
4

1 3
0 0
0 0
1 1

2 4
00 0
01 1
10 1
11 1

1 2
0 1
0 0

2 4
00 0
01 1
10 1
11 0
```

SAMPLE OUTPUT:

```
OK
OK
LIE
LIE
```

Here's a valid program for the first test case:

```
if (b[0] == 0) return 0;
else return 1;
```

Another valid program for the first test case:

```
if (b[0] == 1) return 1;
else return 0;
```

A valid program for the second test case:

```
if (b[1] == 1) return 1;
else if (b[0] == 0) return 0;
else return 1;
```

Clearly, there is no valid program corresponding to the third test case, because Elsie's program must always produce the same output for the same input.

It may be shown that there is no valid program corresponding to the last test case.

SCORING:

- Inputs 2 and 3 have $N = 2$.
- Inputs 4 and 5 have $M = 2$.
- Inputs 6 through 12 have no additional constraints.

Problem credits: Benjamin Qi

Language: C

Source File: Choose File No file chosen

Submit Solution

Note: Many issues (e.g., uninitialized variables, out-of-bounds memory access) can cause a program to produce different output when run multiple times; if your program behaves in a manner inconsistent with the official contest results, you should probably look for one of these issues. Timing can also differ slightly from run to run, so it is possible for a program timing out in the official results to occasionally run just under the time limit in analysis mode, and vice versa. Note also that we have recently changed grading servers, and since our new servers run at different speeds from the servers used during older contests, timing results for older contest problems may be slightly off until we manage to re-calibrate everything properly.