UCF Local Contest (Final Round) — August 31, 2024

Ready for Contest

filename: ready
Difficulty Level: Easy-Medium
Time Limit: 2 seconds

Whenever the UCF Programming Team Coaches prepare for the UCF Local Programming Contest, each problem requires valid judge solutions in C++ and Java. Python solutions may be written as well, but these are optional. Thus, the coaches consider a problem to be ready for the contest if it has at least one C++ solution and at least one Java solution.

Unfortunately, instead of writing solutions for the problems that need them, the coaches just pick their favorite problems and write solutions for those instead. Naturally, they use their favorite language as well instead of checking to see what solutions hasn't been written yet.

Although many solutions have been written, it may be the case that some problems aren't ready for the contest yet. Write a program to determine which problems are ready to go!

The Problem:

Given the number of problems that have been proposed, and a list of every solution written (which problem and which language), determine the problems that are contest ready (have both a C++ and a Java solution written).

The Input:

The first input line contains two integers: n ($1 \le n \le 10^5$), indicating the number of problems proposed and m ($2 \le m \le 3 \times 10^5$), indicating the number of solutions written. The problems are numbered 1 through n.

The information about each solution written are provided in the following m input lines, with each line specifying a single solution written. Each of these input lines contains two integers: p ($1 \le p \le n$), and L ($1 \le L \le 3$), indicating that a solution has been written for problem number p in language L. If L = 1, the language is C + +; if L = 2, the language is Java; if L = 3, the language is Python.

The Output:

Print a list of the problems that are ready for contest, sorted in numerical order, with one number outputted per line. It is guaranteed that at least one problem will be ready for contest so there will be some output.

(Sample Input/Output on the next page)

Sample Input Sample Output

3 11	1
3 11 1 1 1 1 3 1 3 1 3 1 2 2 1 2 2 1 3 1 2 3 2 2	1 2
1 1	
3 1	
3 1	
2 2	
1 2	
1 1 1 1 3 1 3 1 3 1 2 2 1 2 2 1 3 1 2 3 2 2	
3 1	
2 3	
2 2	
4 5	3
4 5 1 3 1 2 3 2 3 1 4 3	
1 2	
3 1	
4 3	
3 9	1 2 3
3 3	2
2 2	
3 2	
3 1	
2 1	
1 2	
3 9 3 3 1 3 2 2 3 2 3 1 2 1 2 3 1 2 1 1	