Problem G
Integer Encryption
Time limit: 2 seconds

You've just received a secret message from one of your agents behind enemy lines, but you're a bit confused. The encryption method you use encrypts messages as a sequence of integers, where each integer has a specific meaning. But when you decrypt the message you've received it doesn't make any sense. After a moment's thought, you figure that your agent must have realized that the enemy can somehow decode your messages, so she applied a second level of encryption to the already encrypted message. In other words, this is what you think has happened:



Figure G.1:

Unfortunately, she didn't bother to inform you what method she used to perform this second encryption. After a second moment's thought, you figure that she probably was in a hurry, needed to act fast, so she simply picked a random value and added it to all of the values in ciphertext1. And after a third moment's thought, you realize that you can probably convert the string of integers in ciphertext2 to the actual encoded string of integers in ciphertext1 since you know of several strings of integers that would have had to be in ciphertext1. For example, if ciphertext2 was `10 893 -204 62` and you know that `883 -214` has to be in ciphertext1, then you figure that your agent simply added 10 to each value in ciphertext1, so the actual message to decode is `0 883 -214 52` (which ironically would further decrypt to "`had to use a second level of encryption`", but that's not important here).

Given ciphertext2 along with one or more strings of integers that should be in ciphertext1, you need to determine the value that your agent added to all the values in ciphertext1. If there is no such value, or if more than one such value exists, then you're going to need a fourth moment's thought, since obviously something else is going on that you haven't figured out yet.

Input

Input starts with a line containing two values n m, where n $(1 \leq n \leq 100\,000)$ is the number of values in ciphertext2, and m $(1 \leq m \leq 30)$ is the number of sequences of integers that should be in the message. Following this is a line with n integers w_1 w_2 ... w_p $(-1\,000 \leq w_i \leq 1\,000)$ specifying ciphertext2. After this are m lines, each specifying a sequence of integers that should be in ciphertext1. Each of these lines has the form p v_1 v_2 ... v_p $(p \geq 1, -1\,000 \leq v_i \leq 1\,000)$ specifying a sequence of p integers. The value of p can vary from line to line. The sum all the p's is $\leq 2n$.

Output

If there is a unique value k so that the m sequences appear in ciphertext2 after subtracting this integer from each value, output k. Otherwise, output `cannot decrypt`.

Sample Input 1

```
4 1
10 893 -204 62
2 883 -214
```

Sample Output 1

```
10
```

Sample Input 2

```
4 2
10 893 -204 62
2 883 -214
1 60
```

Sample Output 2

```
cannot decrypt
```

Sample Input 3

```
6 1
10 893 -204 62 894 -203
2 883 -214
```

Sample Output 3

```
cannot decrypt
```