Snap Dragons

Input File: *snapin.txt*
Output File: *snapout.txt*

Time Limit: 1 second

Have you ever heard of Melodramia, little one? It is a faraway kingdom: a land of daring knights and reclusive wizards, of terrifying gryphons and majestic phoenixes, of ice queens and talking lions. It is a place where every day is a fantastic adventure.

Once upon a time in Melodramia, there lived two dragons named Rose and Scarlet. They were the best of friends, never stealing each other's gold or kidnapped princes, and whenever they had an argument they always settled it with a game of *Dragon Snap*. Dragon Snap was a game they had invented, and it worked like this:

- Rose would take R blank cards and write a secret number on each one. These numbers were in no particular order, and sometimes she would write the same number more than once.
- Scarlet would take S blank cards and do the same.
- At the same time, both dragons flipped over one of their cards at random. If these had the same number written on them, the cards were a *Snap pair*, and whichever dragon shouted "Snap!" first was the winner. Otherwise, they flipped the cards back over and tried again.

After many years of playing, the dragons noticed that having more possible Snap pairs would often lead to a faster game. For example, if Rose chose 6,4,2,4 as her cards and Scarlet chose 9,5,2 as her cards, then only 1 out of the 12 possible pairs would be a Snap pair (Rose's *2* paired with Scarlet's *2*), and the game could go many rounds without finishing.

On the other hand, if Rose chose 8,9,9,9,9 as her cards and Scarlet chose 9,9,8,9,9,9 as her cards, then 21 out of the 30 possible pairs would be Snap pairs (1 pair of *8*s and 20 pairs of *9*s), and the game would end very quickly.

Curious, Rose and Scarlet called upon a powerful magician (i.e. you) to write a program that would read in the list of cards they had each chosen and calculate how many Snap pairs could be made.

(continued over . . .)

Constraints

To evaluate your solution, the judges will run your program against several different input files. All of these files will adhere to the following bounds:

- $0 \leq R, S \leq 100\,000$, where R and S are the number of cards chosen by Rose and Scarlet respectively.
- All card numbers are integers between 1 and 10 000 inclusive.

As some of the test cases will be quite large, you may need to think about how well your solution scales for larger input values. However, not all the cases will be large. Specifically:

- For 50% of the available marks, $1 \leq R, S \leq 1000$.

Input

Your program should read from the file `snapin.txt`. The first line of this file will consist of the integers R and S separated by a space. The following R lines will list the numbers on Rose's cards, one per line. A further S lines will follow, listing the numbers on Scarlet's cards, one per line.

Output

Your program should write to the file `snapout.txt`. Your output file should consist of a single integer: the number of Snap pairs that can be made from the given cards.

Sample Input 1

```
4 3
6
4
2
4
9
5
2
```

Sample Output 1

```
1
```

Sample Input 2

```
5 6
8
9
9
9
9
9
9
8
9
9
9
```

Sample Output 2

```
21
```

Scoring

The score for each input scenario will be 100% if the correct answer is written to the output file, and 0% otherwise.