Rolling table

CODEIT.BG
Embrace The Challenge

SEASON 9 – FOURTH ROUND

You are given a table of lowercase latin letters with N rows and M columns.

You can take the upper i rows ($0 \le i < N$) and move them to the bottom and then take the leftmost j columns ($0 \le j < M$) and move them to the right (without changing the order). For example if i=2 and j=1 and you have the table

abc

def

ghi

you would move the top 2 rows to the bottom:

ghi

abc

def

and then the leftmost column to the right:

hi**g**

bc**a**

ef**d**

Your task is to find how many distinct pairs i and j would produce a table, identical to the one you started with (the letters at the corresponding positions should be the same)

Input

From the first line of the input file `rolling_table.in` N and M are entered.

On the next N lines M lowercase latin letters are entered - the contents of the table.

Output

In the output file `rolling_table.out` print the number of distinct pairs i and j, that would produce an identical table

Constraints

$1 \le N,\ M,\ N * M \le 100\ 000$

Time limit: 1.0 seconds

Memory limit: 256 MB

Example

Input (rolling_table.in)	**Output (rolling_table.out)**	**Explanation**
3 3 bba abb bab	3	The table would remain unchanged when (i=0, j=0) (i=1, j=1) (i=2, j=2)