A2. Death Stars (medium)

time limit per test: 2.5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The stardate is 1983, and Princess Heidi is getting better at detecting the Death Stars. This time, two Rebel spies have yet again given Heidi two maps with the possible locations of the Death Star. Since she got rid of all double agents last time, she knows that both maps are correct, and indeed show the map of the solar system that contains the Death Star. However, this time the Empire has hidden the Death Star very well, and Heidi needs to find a place that appears on both maps in order to detect the Death Star.

The first map is an $N \times M$ grid, each cell of which shows some type of cosmic object that is present in the corresponding quadrant of space. The second map is an $M \times N$ grid. Heidi needs to align those two maps in such a way that they overlap over some $M \times M$ section in which all cosmic objects are identical. Help Heidi by identifying where such an $M \times M$ section lies within both maps.

Input

The first line of the input contains two space-separated integers N and M ($1 \leq N \leq 2000$, $1 \leq M \leq 200$, $M \leq N$). The next N lines each contain M lower-case Latin characters (`a-z`), denoting the first map. Different characters correspond to different cosmic object types. The next M lines each contain N characters, describing the second map in the same format.

Output

The only line of the output should contain two space-separated integers i and j, denoting that the section of size $M \times M$ in the first map that starts at the i-th row is equal to the section of the second map that starts at the j-th column. Rows and columns are numbered starting from 1.

If there are several possible ways to align the maps, Heidi will be satisfied with any of those. It is guaranteed that a solution exists.

Example

input

```
10 5
somer
andom
noise
mayth
eforc
ebewi
thyou
hctwo
again
noise
somermayth
andomeforc
noiseebewi
againthyou
noisehctwo
```

output

```
4 6
```

Note

The 5-by-5 grid for the first test case looks like this:

mayth
eforc
ebewi
thyou
hctwo