# The Grid Search

by PRASHANTB1984

Problem | Submissions | Leaderboard | Discussions | Editorial

### Problem Statement

Given a 2D array of digits, try to find the location of a given 2D pattern of digits. For example, consider the following 2D matrix:

```
1234567890
0987654321
1111111111
1111111111
2222222222
```

Assume we need to look for the following 2D pattern:

```
876543
111111
111111
```

If we scan through the original array, we observe that the 2D pattern begins at the second row and the third column of the larger grid (the 8 in the second row and third column of the larger grid is the top-left corner of the pattern we are searching for).

So, a 2D pattern of $P$ digits is said to be present in a larger grid $G$, if the latter contains a contiguous, rectangular 2D grid of digits matching with the pattern $P$, similar to the example shown above.

### Input Format

The first line contains an integer, $T$, which is the number of test cases. $T$ test cases follow, each having a structure as described below:
The first line contains two space-separated integers, $R$ and $C$, indicating the number of rows and columns in the grid $G$, respectively.
This is followed by $R$ lines, each with a string of $C$ digits, which represent the grid $G$.
The following line contains two space-separated integers, $r$ and $c$, indicating the number of rows and columns in the pattern grid $P$.
This is followed by $r$ lines, each with a string of $c$ digits, which represent the pattern $P$.

### Constraints

$1 \leq T \leq 5$
$1 \leq R, r, C, c \leq 1000$
$1 \leq r \leq R$
$1 \leq c \leq C$

### Test Case Generation

Each individual test case has been generated by first specifying the size ($R$ and $C$) of the large 2D matrix, and then randomly generating the digits in it. A limited number of digits in the larger matrix may be changed by the problem setter

(no more than 5% of the total number of digits in the matrix). So the larger 2D matrix is almost-random. The pattern matrix has been manually-curated by the problem setter.

### Output Format

Display 'YES' or 'NO', depending on whether (or not) you find that the larger grid $G$ contains the rectangular pattern $P$. The evaluation will be case sensitive.

### Sample Input

```
2
10 10
7283455864
6731158619
8988242643
3830589324
2229505813
5633845374
6473530293
7053106601
0834282956
4607924137
3 4
9505
3845
3530
15 15
400453592126560
114213133098692
474386082879648
522356951189169
887109450487496
252802633388782
502771484966748
075975207693780
511799789562806
404007454272504
549043809916080
962410809534811
445893523733475
768705303214174
650629270887160
2 2
99
99
```

### Sample Output

```
YES
NO
```

### Explanation

The first test in the input file is:

```
10 10
7283455864
6731158619
8988242643
3830589324
2229505813
5633845374
6473530293
7053106601
0834282956
4607924137
3 4
9505
3845
3530
```

As one may see, the given 2D grid is indeed present in the larger grid, as marked in bold below.

```
7283455864
6731158619
8988242643
3830589324
2229505813
5633845374
6473530293
7053106601
0834282956
4607924137
```

The second test in the input file is:

```
15 15
400453592126560
114213133098692
474386082879648
522356951189169
887109450487496
252802633388782
502771484966748
075975207693780
511799789562806
404007454272504
549043809916080
962410809534811
445893523733475
768705303214174
650629270887160
2 2
99
99
```

The search pattern is:

```
99
99
```

This cannot be found in the larger grid.



**Submissions:** 10869

**Max Score:** 30

**Difficulty:** Moderate

More

**Current Buffer** (saved locally, editable) Java 8

```
1  import java.io.*;
2  import java.util.*;
3
4  public class Solution {
5
6      public static void main(String[] args) {
7          /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named
   Solution. */
8      }
9  }
```

Line: 1 Col: 1

Upload Code as File  Test against custom input  Run Code  Submit Code