USA COMPUTING OLYMPIAD

Transformations

A square pattern of size N x N (1 <= N <= 10) black and white square tiles is transformed into another square pattern. Write a program that will recognize the minimum transformation that has been applied to the original pattern given the following list of possible transformations:

- #1: 90 Degree Rotation: The pattern was rotated clockwise 90 degrees.
- #2: 180 Degree Rotation: The pattern was rotated clockwise 180 degrees.
- #3: 270 Degree Rotation: The pattern was rotated clockwise 270 degrees.
- #4: Reflection: The pattern was reflected horizontally (turned into a mirror image of itself by reflecting around a vertical line in the middle of the image).
- #5: Combination: The pattern was reflected horizontally and then subjected to one of the rotations (#1-#3).
- #6: No Change: The original pattern was not changed.
- #7: Invalid Transformation: The new pattern was not obtained by any of the above methods.

In the case that more than one transform could have been used, choose the one with the minimum number above.

PROGRAM NAME: transform

INPUT FORMAT

Line 1:	A single integer, N
Line 2..N+1:	N lines of N characters (each either \`@' or \`-'); this is the square before transformation
Line N+2..2*N+1:	N lines of N characters (each either \`@' or \`-'); this is the square after transformation

SAMPLE INPUT (file transform.in)

```
3
@-@
---
@@-
@-@
@--
--@
```

OUTPUT FORMAT

A single line containing the the number from 1 through 7 (described above) that categorizes the transformation required to change from the \`before' representation to the \`after' representation.

SAMPLE OUTPUT (file transform.out)

```
1
```

Submission file Name: 选取文件 未选择文件 Send it in!

USACO Gateway | Comment or Question