This exercise is a variation of the exercise: "". The only difference in the rules of the game is that now the jumps do not make disappear the stones that have been jumped.

Write a program such that, given two configurations of a solitaire, prints if is possible to go from one to the other one.

**Input**

Input consists of a natural $n \geq 3$, followed by the descriptions of the two configurations, each one with $n$ rows with $n$ characters each one. A ‘X’ indicates a stone. The empty positions are indicated with a dot.

**Output**

Your program must print "1" if you can go from one configuration to the other one, or "0" if it is not possible.

**Sample input 1**

```
3
.XX
.X.
.XX
.X.
.X.
.XX
```

**Sample output 1**

```
1
```

**Sample input 2**

```
3
XXX
...
XXX
XXX
...
XXX
```

**Sample output 2**

```
1
```

**Sample input 3**

```
4
XX...
.XX.
..X.
XXX.
XXX.
...X
XX..
..XX
```

**Sample output 3**

```
0
```