

6301 The Shortcut Eight-Puzzle

The shortcut eight-puzzle is a sliding puzzle that consists of a frame of 3×3 spaces where 8 numbered square tiles are placed in a random order with one tile missing. In each sliding move, a numbered square tile can be moved either UP, DOWN, LEFT, RIGHT, or UP-RIGHT to the empty space. (Note that a numbered square tile **cannot** be moved DOWN-LEFT to the empty space) The original object of the puzzle is to change the positions of the tiles from an initial state to a fixed goal state by making sliding moves that uses the empty space. But now, given any initial state and any goal state, your program is requested to output the minimal number of sliding moves that can change the positions of the tiles from the initial state to the goal state. If the goal state is unreachable from the initial state by making sliding moves that uses the empty space, just output '-1'.

For example, given the initial state in Figure 1 and the goal state in Figure 2, the output should be 3.

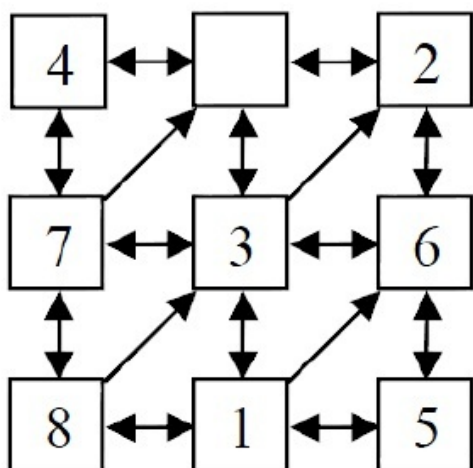


Figure 1: An initial state

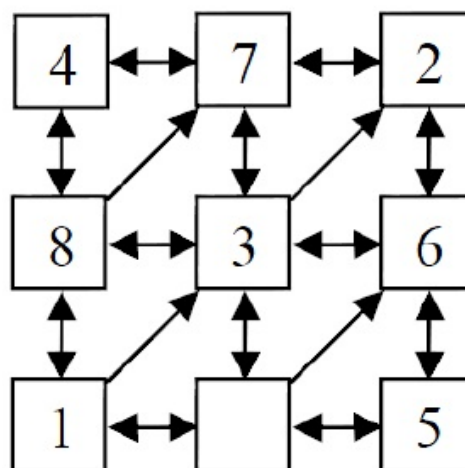


Figure 2: A goal state

Input

The number of test cases is given in the first line as an integer not greater than 5. For each test case, there are 2 more lines. The initial state is given in the first line in a row major order where 9 numbers representing the tiles and the empty space (denoted by '0') are separated by one space. The goal state is given in the second line in a similar way.

Output

For each test case, if the goal state is reachable from the initial state, please output the minimal number of moves necessary to reach the goal state in a separated line. If the goal state is unreachable from the initial state, please output '-1'.

Sample Input

```
2
4 0 2 7 3 6 8 1 5
```

```
4 7 2 8 3 6 1 0 5
2 6 4 1 3 7 0 5 8
8 1 5 7 3 6 4 0 2
```

Sample Output

```
3
27
```