

Problem N. Premove Checkmate

Input file: *standard input*
 Output file: *standard output*
 Time limit: 2 seconds
 Memory limit: 512 mebibytes

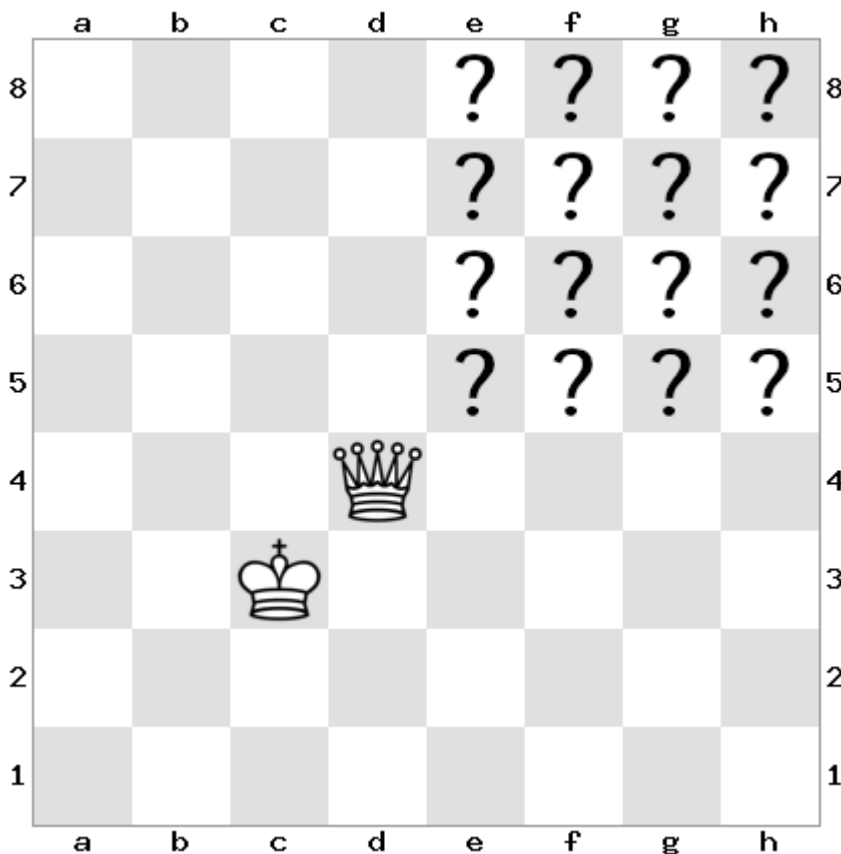
Be careful, the premove description in this problem is not the same as used in a popular chess websites.

You play chess in the internet. You got a position with a king and a queen versus alone opponent's king. The problem is, you don't have enough time on the clock not only to checkmate the opponent, but to make even one move.

Luckily, there is a premove function. The premove mechanic works as follows. You enter a sequence of moves on the board (these can be arbitrary moves, described just by starting and ending squares, it's not necessary that your piece should stay on the starting square). These moves are stored in a queue. You can add moves to the queue only when it's your opponent move.

Then, after your opponent makes a move, if your premove queue is not empty, the moves are popped out one by one. If the popped move is invalid, it's just skipped, and once a popped move is valid, it's done on the board immediately, and your clock time doesn't get decreased.

In the current position, your king is on c3, your queen is on d4, and the opponent's king is on some square in the top-right quarter of the board (e5, e6, e7, e8, f5, f6, f7, f8, g5, g6, g7, g8, h5, h6, h7 or h8). You can't remember where exactly because it's too little time remaining on your clock and you are too stressed!



It's your opponent's move, and, as you don't have clock time remaining, you have to enter a premove queue leading straight to the checkmate. According to chess rules, the checkmate must be made in 50 moves (only your moves count for this rule). The premove queue can store more, up to 500, moves — some of them can be skipped as invalid and therefore don't count.