

The Knight's Travails

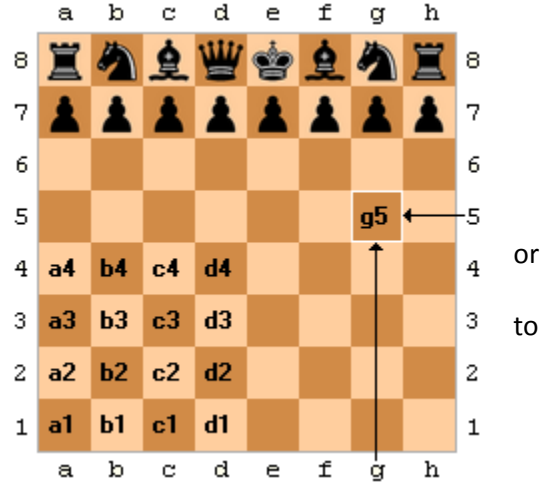
We have a challenge for you!

Given a standard 8x8 chessboard, design a C++ application that accepts two squares identified by algebraic chess notation. The first square is the starting position, and the second square is the ending position. Find the shortest sequence of valid moves to take a Knight piece from the starting position to the ending position. Each move must be a legal move by a Knight. For any two squares there may be more than one valid solution.

Vital information

Algebraic chess notation identifies each square with a letter from A to H and a number from 1 to 8. The columns are labeled with letters, and the rows are numbered. The lower left is A1.

A Knight moves two steps in a straight line from its starting position, and then one square to either the left or right. A Knight can jump over other pieces. In the diagram to the right the Knight at position B8 can move either A6 or C6, while the Knight at position G8 can move to F6 or H6.



Input

Must be two squares identified in algebraic chess notation representing the starting and ending positions of the Knight. The two squares are separated by a space.

Output

Must be a list of squares through which the Knight passes, in algebraic chess notation. This must include the ending position, but exclude the starting position.

Example

Test Input: A8 B7

Expected Output: C7 B5 D6 B7

Assignment brief:

In C++, write a program to find the shortest sequence of moves for a Knight between the starting and ending positions. Your program can accept the input and produce the output in whatever form you think is most effective. Include all code to run the program, and whatever instructions are necessary.

