

Coding Arena



Change Default Language ▼

Time Left

00 **17** **02**
 hr min sec

A B C D E F G H

Problem : Checking Moves

Background

A Chess board position is accurately captured by Forsyth-Edwards notation and is abbreviated as FEN. A FEN "record" defines a particular game position, all in one line of text and using only the ASCII character set. A FEN record consists of six fields. A complete description of the FEN format to represent Chess positions can be found [here](#)

For the purpose of this problem, only consider first of the six fields of FEN. Before we describe the problem, let us look at how FEN maps to a board position. The following 5 images show board positions and its corresponding FEN representation.

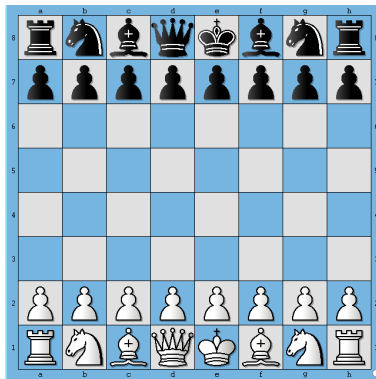
[Rules & Regulations](#)[Launch Code Editor](#)[Notifications](#)[Status messages](#)

Figure 1.

This board position depicts initial position before any side has made a move. In FEN format this board position is represented as

```
mbqkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR
```

Let's say, White plays e4. Then the board position looks like shown below

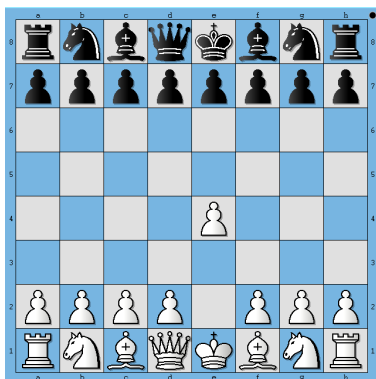


Figure 2.

This board position depicts the Chess board after White has played e4. In FEN format this board position is represented as

```
mbqkbnr/pppppppp/8/8/4P3/8/PPPP1PPP/RNBQKBNR
```

Similarly, 3 more half-moves are depicted in following diagrams

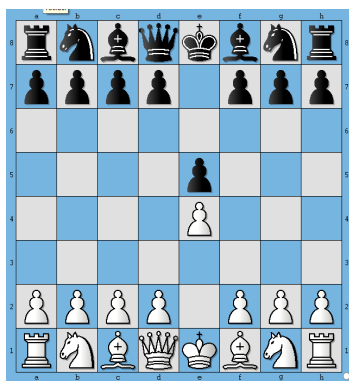


Figure 3.



Figure 4.

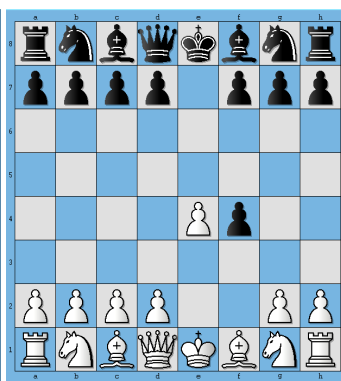


Figure 5.

The FENs corresponding to Figure 3, 4 and 5 are represented as

3. mbqkbnr/pppp1ppp/8/4p3/4P3/8/PPPP1PPP/RNBQKBNR
4. mbqkbnr/pppp1ppp/8/4p3/4PP2/8/PPPP2PP/RNBQKBNR
5. mbqkbnr/pppp1ppp/8/8/4Pp2/8/PPPP2PP/RNBQKBNR

Wikipedia describes first field of FEN format as follows

Piece placement (from white's perspective). Each rank is described, starting with rank 8 and ending with rank 1; within each rank, the contents of each square are described from file "a" through file "h". Following the [Standard Algebraic Notation](#) (SAN), each piece is identified by a single letter taken from the standard English names (pawn = "P", knight = "N", bishop = "B", rook = "R", queen = "Q" and king = "K").^[1] White pieces are designated using upper-case letters ("PNBRQK") while black pieces use lowercase ("pnbrqk"). Empty squares are noted using digits 1 through 8 (the number of empty squares), and "/" separates ranks

Statement

Given a board position in FEN format, your task is to find out all move(s) that white can make that lead to a check to the black king.

Input Format:

1. First line contains single FEN record, which corresponds to a particular board position
2. Second line contains -1 which indicates the end of input

Output Format:

1. The output must be printed as follows
 - a. A string "Checking moves " followed by "[<move format>]"
 - b. Where <move format> is move represented in format "[Piece][fromSquare]-[Piece][toSquare]"
 - c. If a Checking move involves a capture then represent it as "[Piece][fromSquare]-[Piece]x[toSquare]"
 - d. If the Piece inflicting a check is a pawn represent it only as [fromSquare]-[toSquare]
 - e. If there is more than one Checking move, then follow the rules given below
 - i. Moves must be sorted alphabetically according to their ascii values
 - ii. Two moves must be separated by a comma followed by a white space characters
 - iii. Between the last move and terminating "]" character there should be no space or comma
2. See Example section for better understanding of output format

Constraints:

1. The board position will always be White to move.
2. Since we focus on only first part of the FEN, we are essentially ignoring possibility of Castling being a checking move. **Hence our test cases don't contain FENs which give rise to such positions.**
3. There is no need to handle *En Passant* positions. There are no test cases involving *En Passant* moves.
4. No need to implement *pawn promotion* rules. Our test cases do not contain positions which will lead to a pawn getting promoted and inflicting a check.
5. There are no test cases in which capture by a white pawn leads to a check.

Sample Input and Output

SNo.	Input	Output
1	3k4/8/5p2/8/7B/R7/8/2K5 -1	Checking moves [Bh4-Bxf6, Ra3-Ra8, Ra3-Rd3]
2		

r2k4/8/4P3/2B5/8/8/8/4K3
-1

Checking moves [Bc5-Bb6, Bc5-Be7, e6-e7]

Explanation:

Board position for sample input 1:

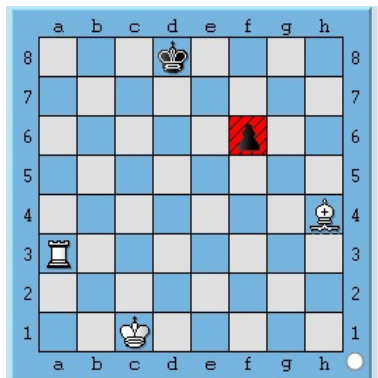


Figure 6.

Board position for sample input 2:

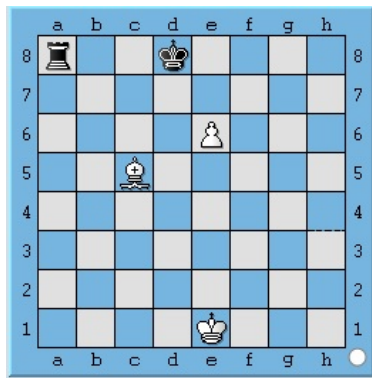


Figure 7.

Note:

Please do not use package and namespace in your code. For object oriented languages your code should be written in one class.

Note:

Participants submitting solutions in C language should not use functions from <conio.h> / <process.h> as these files do not exist in gcc

Note:

For C and C++, return type of main() function should be int.

© 2017 Tata Consultancy Services Limited. All Rights Reserved.

Submit Answer

- ☒ I, YOKESH confirm that the answer submitted is my own.
☐ I would like to provide attribution to the following sources.

Select Language ▾

Select File

Browse...

Submit

