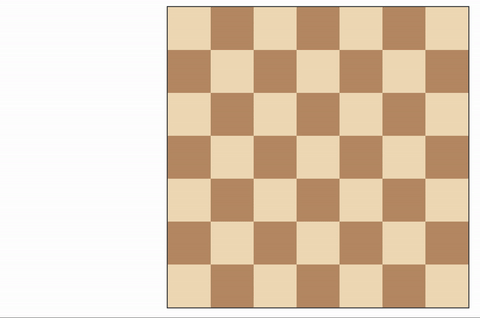
**Knights Isolation: Build an Adversarial Game Playing Agent**



**Synopsis**

In this assignment, you will experiment with adversarial search techniques by building an agent to play knights Isolation. The players control tokens that move like chess queens, this version of Isolation gives each agent control over a single token that moves in L-shaped movements--like a knight in chess.

**Isolation**

In the game Isolation, two players each control their own single token and alternate taking turns moving the token from one cell to another on a rectangular grid. Whenever a token occupies a cell, that cell becomes blocked for the remainder of the game. An open cell available for a token to move into is called a "liberty". **The first player with no remaining liberties for their token loses the game, and their opponent is declared the winner**.

In knights Isolation, tokens can move to any open cell that is rows-2 and column-1 or columns-2 and row-1 away from their current position on the board. On a blank board, this means that tokens have at most eight liberties surrounding their current location. Token movement is blocked at the edges of the board (the board does not wrap around the edges), however, tokens can "jump" blocked or occupied spaces (just like a knight in chess).

**Instructions**

You must implement an agent in the CustomPlayer class. The interface definition for game agents only requires you to implement the get\_action() method, but you can add any other methods to the class that you deem necessary. You need to build a basic agent by combining minimax search with alpha-beta pruning and iterative deepening.