HiQUE Rules
CS 405
Artificial Intelligence

The rules for HiQUE as scanned from the game instructions are displayed on the next page. Your job will be to write an AI program to play the game. For the class, we will make the following modifications to the rules:

1. A runner will automatically be removed after any move where it is possible to remove the runner. In the instructions, the opponent was given the option to remove the runner.
2. In the event that a game is stuck in a loop where both players repeat the same moves, the winner will be declared using the "Timing Option" declared in the rules.
3. There will be a time limit for each program to make a move. The time will be approximately 20 seconds, but may change depending upon the estimated number of moves to complete a game.
4. Your program must conform to the specification for moves given on page 3 of this document.

HIQUE™

Object of the game: To be the first player to take a Runner all the way across to the other side of the board or remove all of your opponents Runners.

The Runner:

The Runners are the smaller pieces. Each player has five runners at the beginning of the game. Runners can only move one space at a time. Forward or from side to side but never backward. They must stay in the light spaces and cannot jump any other pieces. You can win the game by taking a Runner all the way across the board.

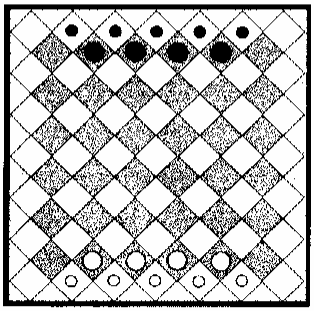
The Guard:

The Guards are the larger pieces. Each player has four Guards throughout the entire game. Guards can move forward, from side to side and backwards any number of spaces as long as they are not blocked by another guard. They must stay on the dark spaces and cannot jump other pieces. By placing two guards next to your opponents Runner (on any side) you can remove it from the game. (See Figure 2) It is possible to take up to four runners in one move. (See figure 3) If your opponent moves a Runner into a position where you have two of your Guards on any side of it you may remove it from the game. (This does not count as a turn) Guards are never taken out of the game.

To Begin Play:

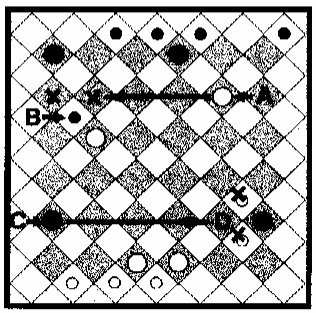
To set up pieces see figure 1. Black goes first. You may start with any piece you choose. The first player to take a Runner all the way across the board wins the game. You can also win by removing all your opponents runners.

Figure 1



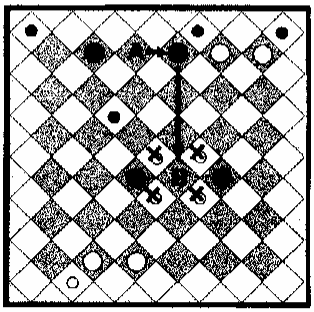
The Guards are the larger pieces. They always stay on the dark spaces. The Runners are the smaller pieces. They always stay on the light spaces.

Figure 2



If Guard A moves to either space marked here with an X it can remove runner B. Two Guards touching any side of a Runner can remove it. If Guard C moves to space D it can remove the two Runners marked with an X.

Figure 3



If Guard A moves to space B it can remove up to four runners here marked with an X. (Very unlikely, but it could happen)

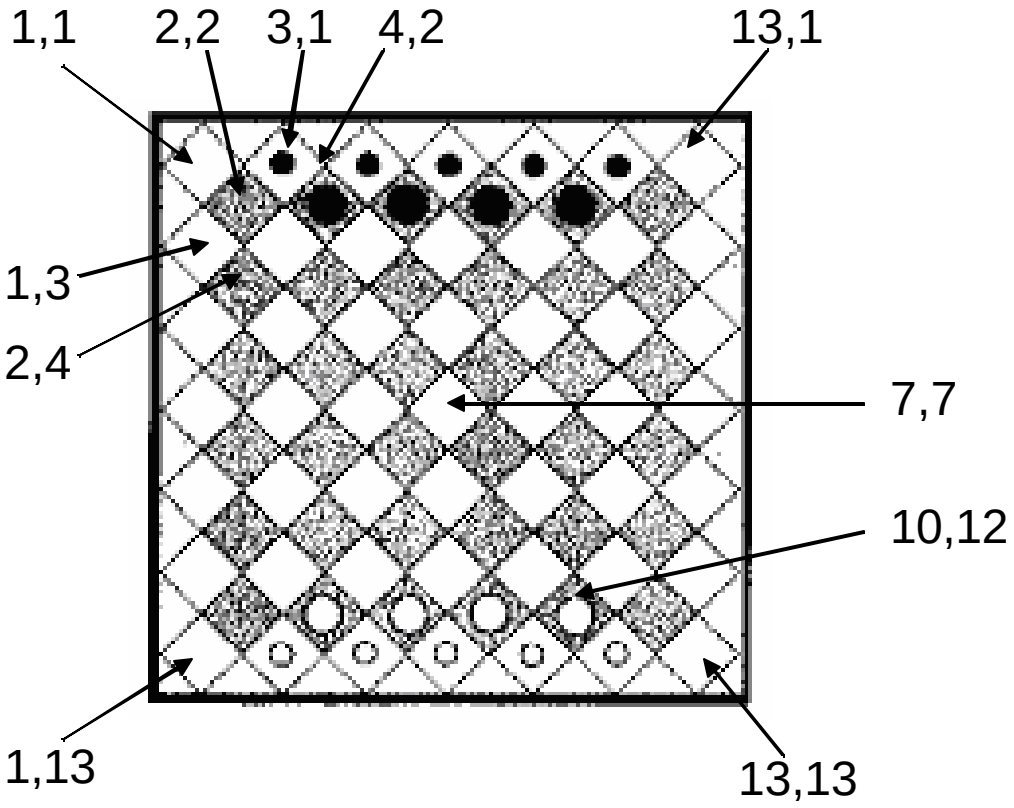
Timing Option: You may put a time limit on each game and win by having the most points. At the end of the game count one point for each Runner you have left plus and additional point for each space the Runner has advanced forward. (Not to the side) Total possible points per Runner is five. (If a players Runner has advanced six spaces forward to the other side, the player wins regardless of points)

Manufactured and distributed by D&L Company P.O. Box 7996, Porterville, Ca. 93258

Specification of Moves

Your program must be able to accept and generate moves using the coordinate system described below. Note that your program is free to use any internal representation for moves and the board that you choose, but there must be a way to map your representation to the one described below so that moves will be compatible with everyone else's programs.

The board is based on a 2D coordinate system, starting with coordinate 1,1 in the upper left corner and moving to coordinate 13,13 in the lower right corner. Guards will always occupy even coordinates and runners will always occupy odd coordinates.



Note that many possible coordinates using this system refer to invalid locations. For example, there is no cell referenced by 1,2.

Moves will be referenced by four numbers: X1 Y1 X2 Y2

Where X1 Y1 specify the source coordinates of the piece to move, and X2 Y2 specify the destination location to move the piece. For example some valid opening moves by black are:

3 1 3 3	(move leftmost runner down one square)
4 2 4 10	(move leftmost guard down to bottom)
4 2 2 2	(move leftmost guard over by one square)