Kono Implementation in Lisp

Philip Glazman

Bug Report

This program does not have any known bugs.

Feature Report

All the features listed in the rubric are implemented as described with the exception of the computer being able to quit the round.

Description of Data Structures

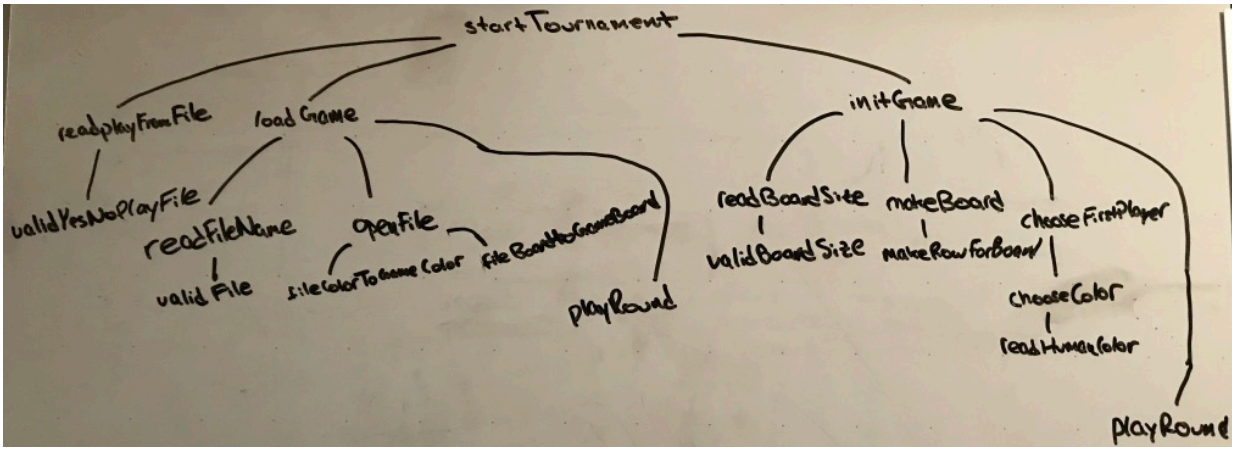
Data Structures

- players
    - List containing player-one, player-one's color, player-two, and player-two color. The order of the players is determined on how the game is started. If a new game is started, player-one is the first player that won the dice-roll. If the game is loaded from file, player-one is the computer. Going forward, this structure should be standardized in order to avoid redundant selector functions.
- board
    - List of lists representing the board. Each list is a row. Each row contains *n* number of strings that represent a game piece.
    - The board is sometimes un-nested for convenience. Using the *flatten* function, the each row inside the board is concatenated so the entire board is represented as a single list.
- scores
    - List containing computer, computer's color, human, human player's color, and round number.
- listOfPieces
    - List of coordinates representing game pieces for computer. Each coordinate contains a list where the first element is the row number and the second element is the column number.
- opponentCoordinates
    - List containing coordinates of closest opponent. The first element is the row number and the second element is the column number.

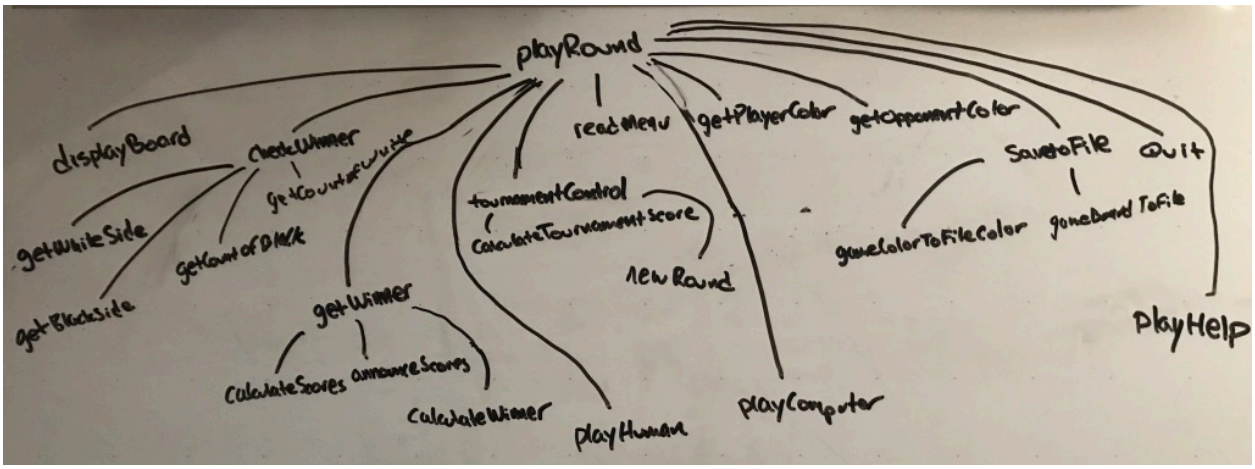
Top-Level Calling of Functions

Below are call charts for the entire program. In order to keep the chart organized, the charts are divided into separate parts.

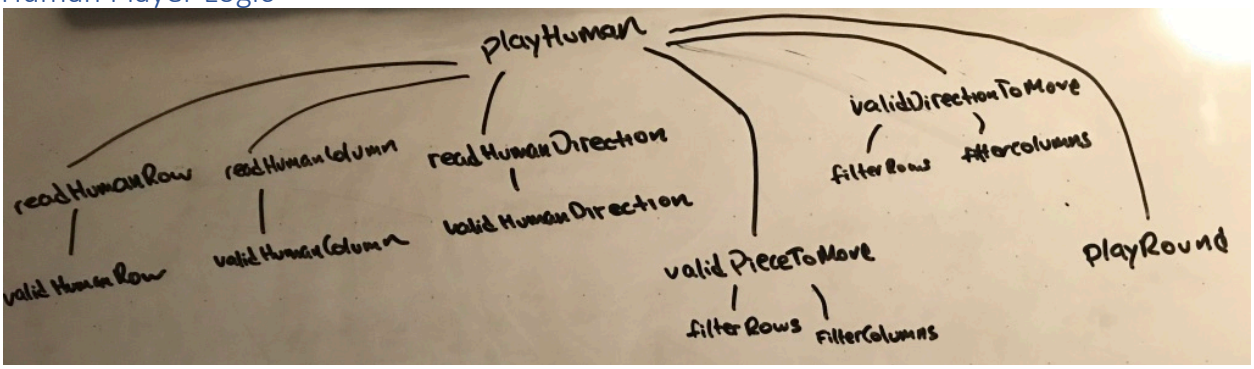
Overall Game



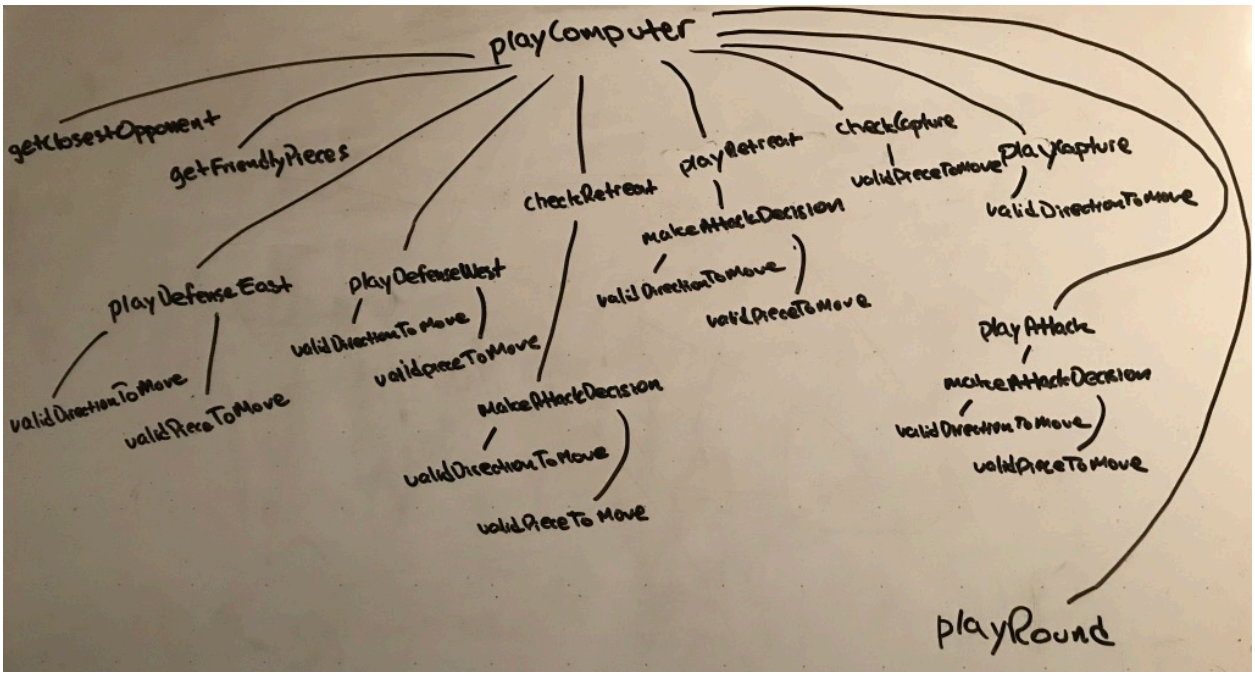
Round Logic



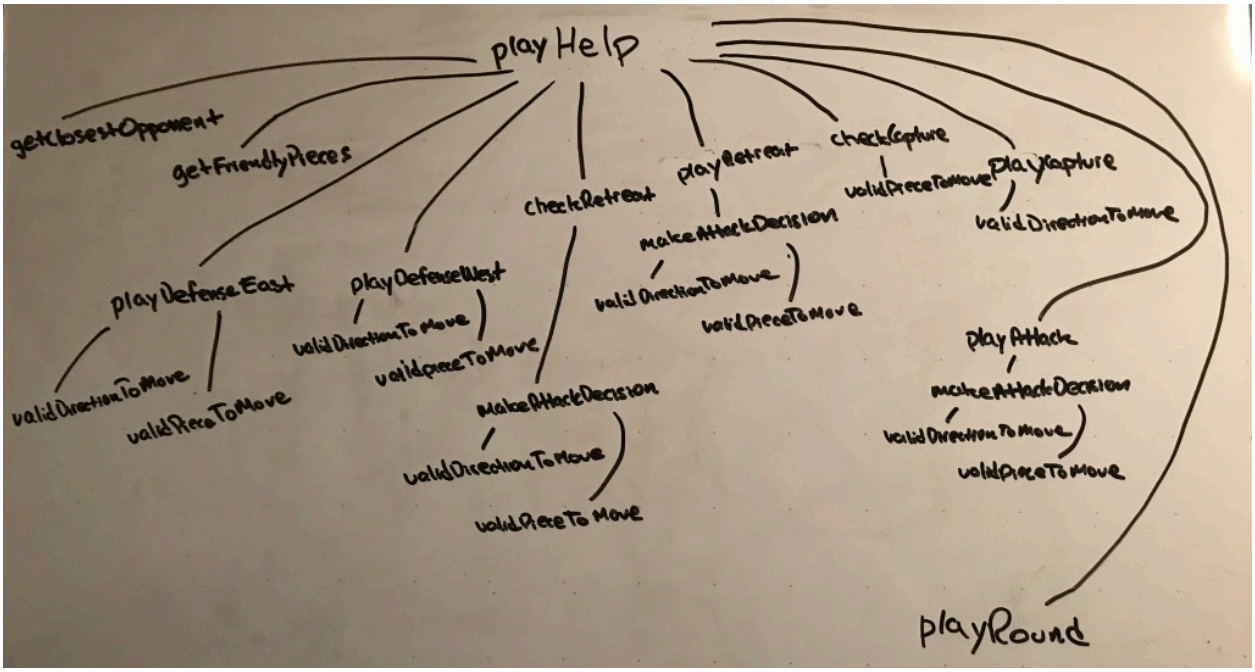
Human Player Logic



Computer Player Logic



Help Mode Logic



Log

Description	Timestamp	Duration
Added board generation, user input for board size, user input for loading game from file.	2/9/18 9:30pm	3 hr
Added board logic so that board is generated recursively.	2/12/18 12:08pm	1.5 hr
Added display function for board to be displayed.	2/12/18 12:45pm	.5 hr
Board rows and columns correctly numbered in display function. Home points correctly set up during board generation.	2/12/18 3:43pm	2 hr
Added menu for round logic. Players alternate during each round. Current player is properly displayed. Added color choice – human chooses color or computer randomly chooses color. Added random dice roll to determine who is first to play.	2/12/18 10:09pm	3 hr
Added serialization for loading existing game file.	2/13/18 12:40pm	2hr
Refactored user menu. Added user input for human player board movements. Added user input for selecting board pieces. Added input validation for selecting board pieces. Board updates on each player move. Players alternate. User is prompted with choose for piece movement based on direction.	2/16/18 4:54pm	5.5 hr

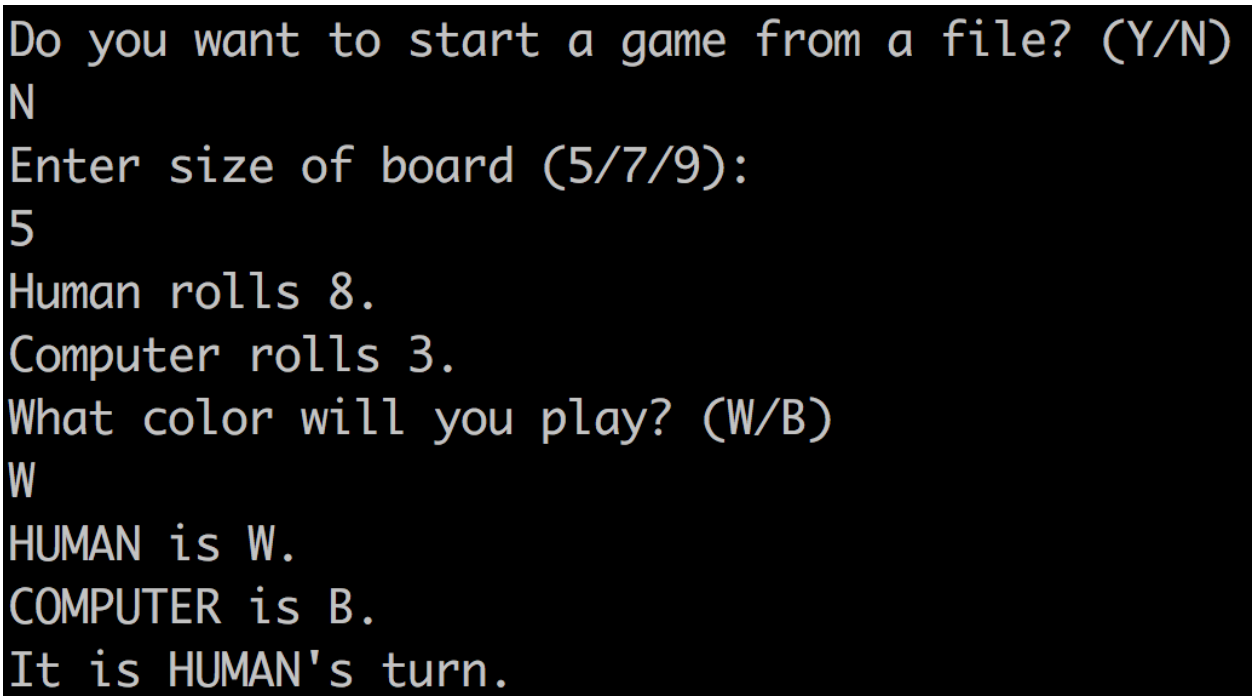
Added output for each player's color. Board now updates after each human move. Debugged issues where game board did not correctly load from file.	2/17/18 11:40pm	2.5 hr
Added more input validation for board movements. Refactored functions for loading game.	2/18/18 1:32pm	2 hr
Added utility functions for checking the number of remaining pieces for each player and checking if pieces occupy the opponent side. Round is now completed when human or computer moves all their available pieces to opposite side.	2/19/18 3:24pm	5 hr
Points correctly calculated for when pieces reach home points. Player scores announced. Winning player is announced and awarded difference in points in round scores to tournament scores. Added tournament control, user can play another round or exit game. Winner is first player in next round.	2/19/18 10:02pm	5 hr
Computer can identify closest opponent and attempt to block it. Began basic computer strategy outline.	2/20/18 1:07pm	2.5 hr
Completed computer strategy for blocking opponent piece from east side. Computer can now identify a list of its friendly pieces.	2/20/18 10:58pm	2 hr
Added serialization to save game to file. Finished computer blocking strategy. Added computer attack strategy and functionality for selecting random piece to move forward.	2/21/18 9:28pm	4 hr
Computer blocking strategy now updates board after block decision. Refactored offensive strategy. Added retreat strategy. Computer can identify its super pieces. Added to serialization so that super pieces can be loaded.	2/22/18 3:45pm	2.5 hr
Computer strategy for capturing pieces using its own super pieces is completed. Added input validation for all human user input. Added input validation for selecting pieces.	2/22/18 11:16pm	4 hr
Added comments for serialization process. Added comments to score counting. Fixed bug in serialization where saving a game would not correctly save game board.	2/23/18 10:24pm	2 hr
Fixed issue where human player could not move their super pieces. Fixed issue where capture strategy only worked for first identified super piece. Fixed issue when computer would not recognize opponent's super piece as opponent. Added additional comments for computer strategy. Added comments for all functions.	2/24/18 9:44pm	3.5 hr

Fixed issue where tournament scores would not correctly print if there was a tie. Fixed an issue when points would not be deduced for quitting game. Added round number to serialization.	2/24/18 10:54pm	1 hr

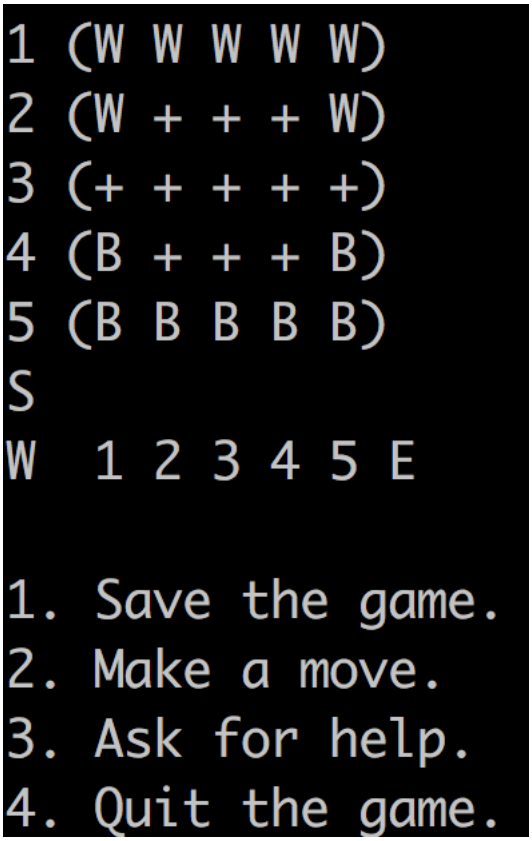
How to Run the Program

To run from command line:
sbcl --non-interactive --load program.lsp

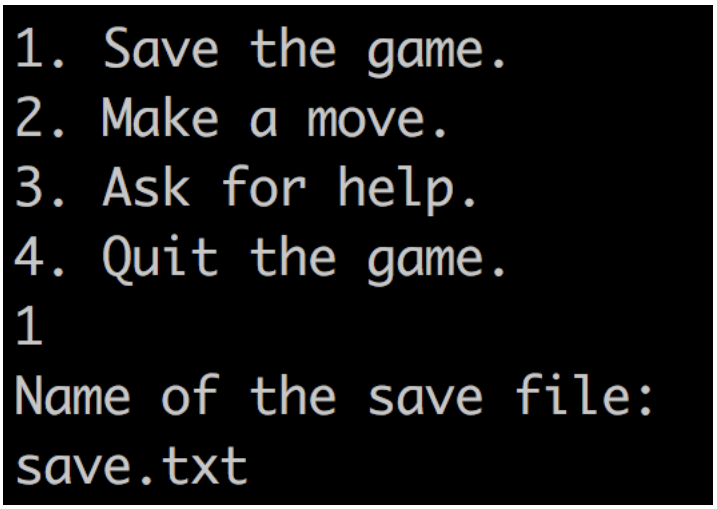
Starting a new game.



Player Menu.



Saving Game.



Loading Game from File.

```
Do you want to start a game from a file? (Y/N)
Y
Name of game file:
save.txt
It is HUMAN's turn.
N
1 (W W W W W)
2 (W + + + W)
3 (+ + + + +)
4 (B + + + B)
5 (B B B B B)
S
W  1 2 3 4 5 E

1. Save the game.
2. Make a move.
3. Ask for help.
4. Quit the game.
```

Making a Move.

```
It is HUMAN's turn.
N
1 (W W W W W)
2 (W + + + W)
3 (+ + + + +)
4 (B + + + B)
5 (B B B B B)
S
W  1 2 3 4 5 E

1. Save the game.
2. Make a move.
3. Ask for help.
4. Quit the game.
2
Enter row of piece to move:
2
Enter column of piece to move:
1
Enter direction to move (NW/NE/SE/SW):
SE
```

Computer Making Move.

```
1. Save the game.
2. Make a move.
4. Quit the game.
2
The computer moved the piece at (5,4) northwest.
It wanted to block the human piece by moving the piece to (4,3).
It is HUMAN's turn.
N
1 (W W W W W)
2 (+ + + + W)
3 (+ W + + +)
4 (B + B + B)
5 (B B B + B)
S
W  1 2 3 4 5 E
```

Asking for Help.

```
1. Save the game.
2. Make a move.
3. Ask for help.
4. Quit the game.
3
It is suggested to move the piece at (1,3) southeast.
This will advance the piece to (2,4).
```

Quiting Game.

```
1. Save the game.
2. Make a move.
3. Ask for help.
4. Quit the game.
4
Quiting game. Deducting 5 points for quiting.
HUMAN has -5 points.
COMPUTER has 0 points.
```

Overall Game.

```
Do you want to start a game from a file? (Y/N)
N
Enter size of board (5/7/9):
5
Human rolls 12.
Computer rolls 10.
What color will you play? (W/B)
W
HUMAN is W.
COMPUTER is B.
It is HUMAN's turn.
N
1 (W W W W W)
2 (W + + + W)
3 (+ + + + +)
4 (B + + + B)
5 (B B B B B)
S
W  1 2 3 4 5 E

1. Save the game.
2. Make a move.
3. Ask for help.
4. Quit the game.
2
Enter row of piece to move:
2
Enter column of piece to move:
1
Enter direction to move (NW/NE/SE/SW):
SE
It is COMPUTER's turn.
N
1 (W W W W W)
2 (+ + + + W)
3 (+ W + + +)
4 (B + + + B)
5 (B B B B B)
S
W  1 2 3 4 5 E

1. Save the game.
2. Make a move.
4. Quit the game.
2
The computer moved the piece at (5,4) northwest.
It wanted to block the human piece by moving the piece to (4,3).
```