

EECS 3401 Winter 2017 Assignment 4

Available March 27. Due April 5 at 23:55.

Files for Assignment 4

1. [PDF handout](#).
2. [coverpage.txt](#): a cover page to put on top of what you hand in.
3. [othello.swi](#): starter code for your Othello implementation.
4. [play.swi](#): interactive game tree search engine (using plain minimax depth-first search).
5. [testboards.swi](#) and [testboards.pdf](#): The first file contains three test boards in PROLOG format. You are to run some evaluations on these as stated in `testboard.pdf`. First, run your minimax implementation on the test boards and fill in the first table. Secondly, if you have implemented alpha-beta search, run your alpha-beta implementation on the test boards and fill in the second table. In this case, also write one or two paragraphs on page 2, discussing your results in the two tables.
6. [ttt.swi](#): example implementation of the tic-tac-toe game showing how the game tree search engine can be used.

Note that the .pl files have been renamed .swi so that our web server will let you download them.

FAQ

- From N.B.: *I'm just wondering about the `initBoard` predicate, you defined the board to be*

```
[. . . . .]
[. . . . .]
[. . 1 2 . .]
[. . 2 1 . .]
[. . . . .]
[. . . . .]
```

and since player 1 is the human and player 2 is the computer wouldn't that mean that player 1 is white but in the description of the assignment black is supposed to move first. Shouldn't the board be

```
[. . . . .]
[. . . . .]
[. . 2 1 . .]
[. . 1 2 . .]
[. . . . .]
[. . . . .]
```

Use the initial configuration contained in the starter code. Note that the row numbering used in the figures in the handout is bottom-to-top. So, if you flip the board in the figures vertically you get the configuration used in the starter code. (Of course the starter code uses a 6x6 board). In addition, Player 1 (Human) plays first.

- From N.B.: *I have a question about the ordering of `MvList`. In the assignment you say that the list should be sorted left to right, top to bottom, but since in the code the pair `[x, y]` denotes column `x` and row `y` this would mean that the given list represents this type of matrix*

```
[0,0] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
[_,_] [1,1] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
[0,2] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
[_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
```

```

[_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
[_,_] [1,5] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
[_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
[_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]
[_,_] [2,7] [_,_] [_,_] [_,_] [_,_] [_,_] [_,_]

```

So that would mean that the list is sorted top to bottom, left to right not the other way around so I'm just wondering if this is the right interpretation or are we supposed to take the pair $[x, y]$ to mean row x and column y .

$[x, y]$ in the handout refers to column x and row y (each between 0 and 5). So, for example the up position (i.e. fix the column but go one row up) for $[3, 2]$ is $[3, 1]$. The order we are looking for the moves is first all moves in column 1 then column 2 and so on where the order in each column is top to bottom. So, we get $[0, 0], [0, 1], \dots, [0, 5], [1, 0], \dots, [1, 5], \dots, [5, 0], \dots, [5, 5]$.