report

June 9, 2020

1 Build a Game Playing Agent - Knights Isolation

1.1 James McGuigan

- Source: https://github.com/JamesMcGuigan/udacity-artificial-intelligence/blob/master/Projects/3_Adversarial%20Search/

2 Unit Tests

[1]:
```
from agents.AlphaBetaPlayer import AlphaBetaAreaPlayer, AlphaBetaPlayer,␣
↪MinimaxPlayer
from agents.DistancePlayer import DistancePlayer, GreedyDistancePlayer
from agents.MCTS import MCTSMaximum, MCTSMaximumHeuristic, MCTSRandom,␣
↪MCTSRandomHeuristic
from agents.UCT import UCTPlayer
from sample_players import GreedyPlayer, RandomPlayer
from run_backpropagation import run_backpropagation, TEST_AGENTS
from isolation import Agent, logger
from run_match_sync import play_sync
import time
import gc

%load_ext autoreload
%autoreload 2
```

[2]:
```
! python3 -m unittest -v
```

```
test_get_action_midgame (tests.test_my_custom_player.CustomPlayerGetActionTest)
get_action() calls self.queue.put() before timeout in a game in progress … ok
test_get_action_player1 (tests.test_my_custom_player.CustomPlayerGetActionTest)
get_action() calls self.queue.put() before timeout on an empty board … ok
test_get_action_player2 (tests.test_my_custom_player.CustomPlayerGetActionTest)
get_action() calls self.queue.put() before timeout as player 2 … ok
test_get_action_terminal (tests.test_my_custom_player.CustomPlayerGetActionTest)
get_action() calls self.queue.put() before timeout when the game is over … ok
test_custom_player (tests.test_my_custom_player.CustomPlayerPlayTest)
```

```
CustomPlayer successfully completes a game against itself … ok

----------------------------------------------------------------------
Ran 5 tests in 18.558s

OK
```

3 Infrastructure

- run_match_sync.py
- run_backpropagation.py
- agents/DataSavePlayer.py

I've rewritten `run_match.py` with a synchronous implementation in `run_match_sync.py` and `run_backpropagation.py`. using signals rather than `multithreading.Pool()`, which allows for better profiling and a 2x performance speedup.

I have improved the CLI flags and logging ouput, with an extra `--verbose` flag which can be used to ASCII print out the board state after each turn.

DataSavePlayer now handles loading and atexit autosaving of cls.data, whilst also gzipping contents

4 Advanced Heuristic

- agents/AlphaBetaPlayer.py

  What features of the game does your heuristic incorporate, and why do you think those features matter in evaluating states during search?

The main analogy is with the game of Go. The goal is to surround your opponent and capture a larger territory.

Recursively computing liberties several moves ahead shows the area of the board that the opponent could potentially escape to.

At `depth=1` this `heuristic_area()` is equivalent to `#my_moves - #opponent_moves`

Early in the game the opponent effectively has access to the entire board, making this heuristic ineffective, hence `max_area` is used in addition to depth to shortcircuit the computational cost of expanding breadth-first search to all possible future moves on a mostly empty board when neither side is trapped.

This heuristic is endgame focused. It solves the simplified subproblem of local search without an adversary, and provides an upper-bound estimate for the difference in maximum number of total moves each player has remaining. It reaches maximum value for moves that trap a player within a self-contained section whilst leaving the other a means of escape. The player in the smaller territory will run out of moves first.

The other major impact on the performance of this agent is the addition of alphabeta pruning with the aggressive use of caching. This avoids the computation expense of recomputing previously

explored subtrees, and by caching on `@classmethod` parts of the cache can even be reused between runs. The net effect of this is to increase the maximum depth of iterative deepening before the timeout to beyond what `MinimaxPlayer(depth=3)` can compute.

Improvements over previous submission: - Iterative Deepening checks terminates early if action score is infinite - Caching has been redone: now keyed on (player_id,state) and ignores depth/alpha/beta - Alphabeta caching only stores infinite values, which is effectively an endgame table - Caching is persisted to disk ./data/*.zi p.pickle via DataSavePlayer base class - Persisted caching means Alphabeta can pretrained with a higher timeout before the match

Performance improvements mean the algorithm now dominates the course Minimax implementation - First percentage is the total winrate, second percentage is the rolling average winrate

[3]:
```
!rm -f ./data/A*.pickle
```

Greedy vs Minimax (at depth 2) depends on who gets the first turn. At standard depth 3, it has a 100% winrate.

[4]:
```
! python3 ./run_backpropagation.py  -a GREEDY -o MINIMAX --progress -r 70
```

```
---------------------------------------------------------------------- match_id:
70 |  72s |   0% ->   0% | Greedy vs Minimax
```

AlphaBeta gets 86%+ winrate vs both Greedy and Minimax

[5]:
```
! python3 ./run_backpropagation.py  -a ALPHABETA -o GREEDY --progress -r 70
```

```
++++++++++++++++++++++-++++++-+-+++++++-+++++++-++++++++++++++++++++ match_id:
70 | 290s |  93% ->  93% | AlphaBeta vs Greedy
wrote:  ./data/AlphaBetaPlayer.zip.pickle        |  0.8MB in  1.1s | entries:
55693
```

[6]:
```
! python3 ./run_backpropagation.py  -a ALPHABETA -o MINIMAX --progress -r 70
```

```
loaded: ./data/AlphaBetaPlayer.zip.pickle        |  0.9MB in  0.3s | entries:
55693
+++-+-+-+---++++++++++-++--+++-++++++++--+++-+++++-+-+++++++++-++++++- match_id:
70 | 304s |  76% ->  79% | AlphaBeta vs Minimax
wrote:  ./data/AlphaBetaPlayer.zip.pickle        |  1.6MB in  2.3s | entries:
107152
```

AlphaBetaAreaPlayer gets near 100% winrate vs Greedy and Minimax, plus 70% vs AlphaBetaPlayer

[7]:
```
! python3 ./run_backpropagation.py  -a AREA -o GREEDY --progress -r 70
```

```
++++++++++++++++++++++++++++++++-++++++++++++++-++++++++++++++++++++ match_id:
70 | 246s |  97% ->  97% | AlphaBeta Area vs Greedy
wrote:  ./data/AlphaBetaAreaPlayer.zip.pickle    |  5.8MB in  7.4s | entries:
368483
```

[8]:
```
! python3 ./run_backpropagation.py  -a AREA -o MINIMAX --progress -r 70
```

```
loaded: ./data/AlphaBetaAreaPlayer.zip.pickle    |  5.9MB in  2.0s | entries:
368483
++++++++++++++++++++++++++++++++++-++++++++++++++++++++++++-+++++++++ match_id:
70 | 327s |  97% ->  96% | AlphaBeta Area vs Minimax
wrote:  ./data/AlphaBetaAreaPlayer.zip.pickle    | 13.8MB in 18.1s | entries:
873487
```

[9]:
```
! python3 ./run_backpropagation.py  -a AREA -o ALPHABETA --progress -r 70
```

```
loaded: ./data/AlphaBetaPlayer.zip.pickle        |  1.6MB in  0.5s | entries:
107152
loaded: ./data/AlphaBetaAreaPlayer.zip.pickle    | 13.8MB in  6.2s | entries:
873487
+++++-+-+-+-+--+++-++-+-+-+-+-----+---+++-+-+++-++-+-----++++++++++- match_id:
70 | 623s |  56% ->  58% | AlphaBeta Area vs AlphaBeta
wrote:  ./data/AlphaBetaPlayer.zip.pickle        |  4.0MB in  4.5s | entries:
261045
wrote:  ./data/AlphaBetaAreaPlayer.zip.pickle    | 22.9MB in 25.2s | entries:
1452080
```

5 Depth Analysis

- MiniMax course default is depth 3
- AlphaBeta
  - has trouble getting past depth 1 on the first turn
  - can get to depth 5 on the second turn
  - remains at depth 4-6 for the early game
  - grows to depth 6-9 during the mid-game
  - expands out to max depth of 14 before finding a -inf lose condition
- AlphaBetaArea
  - is usually depth-2 compared to AlphaBeta
  - the area heuristic effectively adds an extra 4 layers of hidden depth
  - so for the same CPU cost, AlphaBetaArea has a depth 2 advantage
  - finds the inf win condition before AlphaBeta and at a lower depth

[10]:
```
AlphaBetaPlayer.verbose_depth     = True
AlphaBetaAreaPlayer.verbose_depth = True

play_sync( ( Agent(AlphaBetaPlayer,'AlphaBeta'),␣
 ↪Agent(AlphaBetaAreaPlayer,'AlphaBetaArea') ) )
pass
```

```
loaded: ./data/AlphaBetaPlayer.zip.pickle        |  3.1MB in  1.1s | entries:
203452
loaded: ./data/AlphaBetaAreaPlayer.zip.pickle    | 18.0MB in  6.1s | entries:
```

```
1142325

AlphaBetaPlayer      | depth: 1
AlphaBetaAreaPlayer  | depth: 1 2
AlphaBetaPlayer      | depth: 1 2 3 4
AlphaBetaAreaPlayer  | depth: 1 2 3 4
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
AlphaBetaPlayer      | depth: 1 2 3 4
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5 6
AlphaBetaAreaPlayer  | depth: 1 2 3 4
AlphaBetaPlayer      | depth: 1 2 3 4 5 6
AlphaBetaAreaPlayer  | depth: 1 2 3 4
AlphaBetaPlayer      | depth: 1 2 3 4 5 6
AlphaBetaAreaPlayer  | depth: 1 2 3 4
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 inf
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
-inf
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 -inf
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 -inf
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 -inf
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 -inf
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 -inf
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 -inf
AlphaBetaPlayer      | depth: 1 inf
AlphaBetaAreaPlayer  | depth: 1 -inf
```

> If this can be done, then it turns out that alpha–beta needs to examine only O(bm/2) nodes to pick the best move, instead of O(bm) for minimax. This means that the effective branching factor becomes √b instead of b—for chess, about 6 instead of 35. Put another way, alpha–beta can solve a tree roughly twice as deep as minimax in the same amount of time. - Artificial Intelligence: A Modern Approach (p160)

In practice, AlphaBeta gets an smaller depth lead over MiniMax with an average of 1 and range of 0-2

[32]:
```
# don't load existing caches
class AlphaBeta(AlphaBetaPlayer):
    search_fn     = 'alphabeta'
    heuristic_fn  = 'heuristic_liberties'   # 'heuristic_liberties' |
 ↪'heuristic_area'
    verbose_depth = True
    def load(cls): pass
    def save(cls): pass

class MiniMax(AlphaBetaPlayer):
```

```
    search_fn     = 'minimax'
    heuristic_fn  = 'heuristic_liberties'  # 'heuristic_liberties' |␣
↪'heuristic_area'
    verbose_depth = True
    def load(cls): pass
    def save(cls): pass

play_sync( ( Agent(MiniMax,'MiniMax'), Agent(AlphaBeta,'AlphaBeta') ) )
pass
```

```
MiniMax              | depth: 1
AlphaBeta            | depth: 1 2
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5 6
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5 6
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5 6
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5 6
MiniMax              | depth: 1 2 3 4 5 6
AlphaBeta            | depth: 1 2 3 4 5 6 7
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5 6 7
```

```
MiniMax              | depth: 1 2 3 4 5 6
AlphaBeta            | depth: 1 2 3 4 5 6 7
MiniMax              | depth: 1 2 3 4 5 6
AlphaBeta            | depth: 1 2 3 4 5 6 7
MiniMax              | depth: 1 2 3 4 5 6
AlphaBeta            | depth: 1 2 3 4 5 6
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5 6 7
MiniMax              | depth: 1 2 3 4 5
AlphaBeta            | depth: 1 2 3 4 5 6
MiniMax              | depth: 1 2 3 4 5 6
AlphaBeta            | depth: 1 2 3 4 5 6 7
MiniMax              | depth: 1 2 3 4 5 6 7
AlphaBeta            | depth: 1 2 3 4 5 6 7 8 9
MiniMax              | depth: 1 2 3 4 5 6 7
AlphaBeta            | depth: 1 2 3 4 5 6 7 8 9
MiniMax              | depth: 1 2 3 4 5 6 7 8
AlphaBeta            | depth: 1 2 3 4 5 6 7 8 9
MiniMax              | depth: 1 2 3 4 5 6 7 8 9 10
AlphaBeta            | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 inf
MiniMax              | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 -inf
AlphaBeta            | depth: 1 inf
MiniMax              | depth: 1 -inf
```

6 Monty Carlo Tree Search

- agents/MCTS.py
- agents/UCT.py

I have implemented variations on Monty Carlo Tree Search.

6.1 MCTSMaximum

is designed to be reinforcement learning agents, to be trained by running repeatedly before the match. Initially they will make random moves. After each match, the `.backpropergate()` function is called, which will build a record of the win/loss ratio for each seen board position, and also compute the BestChild score `(w/n + c√(ln N/n))` for each seen node. The agent is runtime fast in the sense that it only needs to read from it's own cache to compute the max score of available actions, which assumes the agent has seen this exact board position before during training,

6.2 MCTSRandom

Is a variation inspired by the Ant Colony Optimization Algorithm. It removes the exploration term, and instead of selecting the action with the maximum score, it uses the difference in score as random weighting factor to stochastically select the next action.

6.3 MCTSRandomHeuristic + MCTSMaximumHeuristic

These are variants that add the liberties heuristic to the BestChild score

6.4 Results

- MCTSRandom wins against MCTSMaximum by 68%

[25]:
```
! rm -f ./data/MCTS*.pickle
! python3 ./run_backpropagation.py  -a MCR  -o MCM  -r 100 -l 100
```

```
 match_id:  100 |   1s |  70% ->  70% | MCTS Random vs MCTS Maximum
wrote:  ./data/MCTSMaximum.zip.pickle            |  0.1MB in  0.1s | entries:
4354
wrote:  ./data/MCTSRandom.zip.pickle             |  0.1MB in  0.1s | entries:
4354
```

MCTSRandom can be shown to be trainable against a Greedy opponent

[26]:
```
! python3 ./run_backpropagation.py  -a MCR -o GREEDY -r 1000  -l 100
```

```
loaded: ./data/MCTSRandom.zip.pickle             |  0.1MB in  0.0s | entries:
4354
 match_id:  100 |   1s |  39% ->  37% | MCTS Random vs Greedy
 match_id:  200 |   3s |  44% ->  46% | MCTS Random vs Greedy
 match_id:  300 |   4s |  47% ->  50% | MCTS Random vs Greedy
 match_id:  400 |   5s |  45% ->  46% | MCTS Random vs Greedy
 match_id:  500 |   7s |  45% ->  45% | MCTS Random vs Greedy
 match_id:  600 |   8s |  46% ->  47% | MCTS Random vs Greedy
 match_id:  700 |   9s |  48% ->  49% | MCTS Random vs Greedy
 match_id:  800 |  11s |  48% ->  49% | MCTS Random vs Greedy
 match_id:  900 |  12s |  48% ->  50% | MCTS Random vs Greedy
 match_id: 1000 |  13s |  48% ->  49% | MCTS Random vs Greedy
wrote:  ./data/MCTSRandom.zip.pickle             |  0.6MB in  0.8s | entries:
29185
```

MCTSRandom has a curious dip in performance against Minimax. - After training against Greedy, its performance is 17% - Repeatedly losing seems to amplify exploration, decreasing to a minimum 11% winrate - It eventually finds better countermoves, training itself back upto 18% winrate

[27]:
```
! python3 ./run_backpropagation.py  -a MCR -o MINIMAX -r 1000
```

```
loaded: ./data/MCTSRandom.zip.pickle             |  0.6MB in  0.1s | entries:
29185
 match_id:  100 |  32s |  17% ->  17% | MCTS Random vs Minimax
 match_id:  200 |  63s |  13% ->  11% | MCTS Random vs Minimax
 match_id:  300 |  92s |  13% ->  13% | MCTS Random vs Minimax
 match_id:  400 | 122s |  14% ->  14% | MCTS Random vs Minimax
 match_id:  500 | 153s |  14% ->  14% | MCTS Random vs Minimax
```

```
 match_id:  600 | 186s |  16% ->  17% | MCTS Random vs Minimax
 match_id:  700 | 216s |  16% ->  18% | MCTS Random vs Minimax
 match_id:  800 | 246s |  17% ->  19% | MCTS Random vs Minimax
 match_id:  900 | 275s |  17% ->  18% | MCTS Random vs Minimax
 match_id: 1000 | 303s |  17% ->  18% | MCTS Random vs Minimax
wrote:  ./data/MCTSRandom.zip.pickle             |  1.3MB in  1.8s | entries:
64980
```

It has a hard time against AlphaBeta, with only a 4-6% winrate

[28]:
```
! python3 ./run_backpropagation.py  -a MCR -o ALPHABETA -r 250 -l 50
```

```
loaded: ./data/MCTSRandom.zip.pickle             |  1.3MB in  0.3s | entries:
64980
 match_id:   50 | 192s |   4% ->   7% | MCTS Random vs AlphaBeta
 match_id:  100 | 377s |   6% ->   7% | MCTS Random vs AlphaBeta
 match_id:  150 | 570s |   6% ->   6% | MCTS Random vs AlphaBeta
 match_id:  200 | 762s |   6% ->   5% | MCTS Random vs AlphaBeta
 match_id:  250 | 957s |   6% ->   6% | MCTS Random vs AlphaBeta
wrote:  ./data/AlphaBetaPlayer.zip.pickle        |  4.7MB in  5.3s | entries:
308358
wrote:  ./data/MCTSRandom.zip.pickle             |  1.5MB in  2.1s | entries:
75992
```

What if we pretrain all the Monty Carlo agents against each other?

NOTE: Cross-training in a league is the method used by the Starcraft AlphaStar agent.

[29]:
```
for agent in TEST_AGENTS.keys():
    if agent.startswith('MC'):
        TEST_AGENTS[agent].agent_class.verbose = False
        TEST_AGENTS[agent].agent_class.load()

for agent in TEST_AGENTS.keys():
    for opponent in TEST_AGENTS.keys():
        if agent.startswith('MC') and opponent.startswith('MC'):
            time.sleep(0.1)
            run_backpropagation({
                "agent":      agent,
                "opponent":   opponent,
                "rounds":     1000,
                "logging":    1000,
                "progress":   False,
                "time_limit": 0   # reduce freak TimeoutErrors
            })

for agent in TEST_AGENTS.keys():
    if agent.startswith('MC'):
        TEST_AGENTS[agent].agent_class.save()
```

```
 match_id: 1000 |   9s |  51% ->  50% | MCTS Maximum vs MCTS Maximum 2
 match_id: 1000 |   9s |  26% ->  26% | MCTS Maximum vs MCTS Random
 match_id: 1000 |  10s |  53% ->  51% | MCTS Maximum vs MCTS Maximum Heuristic
 match_id: 1000 |  20s |  31% ->  32% | MCTS Maximum vs MCTS Random Heuristic
 match_id: 1000 |  11s |  70% ->  72% | MCTS Random vs MCTS Maximum
 match_id: 1000 |  10s |  49% ->  49% | MCTS Random vs MCTS Random 2
 match_id: 1000 |  12s |  74% ->  74% | MCTS Random vs MCTS Maximum Heuristic
 match_id: 1000 |  14s |  53% ->  55% | MCTS Random vs MCTS Random Heuristic
 match_id: 1000 |  10s |  53% ->  51% | MCTS Maximum Heuristic vs MCTS Maximum
 match_id: 1000 |  12s |  24% ->  24% | MCTS Maximum Heuristic vs MCTS Random
 match_id: 1000 |  12s |  45% ->  46% | MCTS Maximum Heuristic vs MCTS Maximum
Heuristic 2
 match_id: 1000 |  17s |  32% ->  32% | MCTS Maximum Heuristic vs MCTS Random
Heuristic
 match_id: 1000 |  22s |  69% ->  70% | MCTS Random Heuristic vs MCTS Maximum
 match_id: 1000 |  15s |  46% ->  47% | MCTS Random Heuristic vs MCTS Random
 match_id: 1000 |  17s |  67% ->  65% | MCTS Random Heuristic vs MCTS Maximum
Heuristic
 match_id: 1000 |  22s |  48% ->  48% | MCTS Random Heuristic vs MCTS Random
Heuristic 2
```

Then attempt a rematch against AlphaBeta, which results in a small but significant winrate improvement.

[30]:
```
! python3 ./run_backpropagation.py  -a MCR -o ALPHABETA -r 250 -l 50
```

```
loaded: ./data/AlphaBetaPlayer.zip.pickle        |  4.7MB in  1.4s | entries:
308358
loaded: ./data/MCTSRandom.zip.pickle             |  7.2MB in  2.1s | entries:
369486
 match_id:   50 | 194s |   4% ->   5% | MCTS Random vs AlphaBeta
 match_id:  100 | 378s |   4% ->   4% | MCTS Random vs AlphaBeta
 match_id:  150 | 565s |   5% ->   5% | MCTS Random vs AlphaBeta
 match_id:  200 | 752s |   5% ->   5% | MCTS Random vs AlphaBeta
 match_id:  250 | 947s |   5% ->   5% | MCTS Random vs AlphaBeta
wrote:  ./data/AlphaBetaPlayer.zip.pickle        |  9.5MB in 12.0s | entries:
629184
wrote:  ./data/MCTSRandom.zip.pickle             |  7.4MB in 10.1s | entries:
380447
```

6.5 UCTPlayer

UCTPlayer will use its 150ms of time to simulate MCTSMaximum from the current board position before returning an answer based on the current scores of current available actions.

[31]:
```
! python3 ./run_backpropagation.py  -a UCT -o GREEDY -r 70 --progress
```

```
loaded: ./data/MCTSMaximum.zip.pickle            |  5.4MB in  1.5s | entries:
280851
loaded: ./data/MCTSMaximum.zip.pickle            |  5.4MB in  1.5s | entries:
280851
-+----++--+-----------++--++----+-+--------+---+-+----+------+--++-- match_id:
70 | 266s |  24% ->  24% | UCT vs Greedy
wrote:  ./data/MCTSMaximum.zip.pickle            |  5.7MB in  9.0s | entries:
295859
```

[32]:
```
! python3 ./run_backpropagation.py  -a UCT -o MINIMAX -r 70 --progress
```

```
loaded: ./data/MCTSMaximum.zip.pickle            |  5.7MB in  2.0s | entries:
295859
loaded: ./data/MCTSMaximum.zip.pickle            |  5.7MB in  2.0s | entries:
295859
---------------------------------------+-------------------------+-- match_id:
70 | 282s |   3% ->   4% | UCT vs Minimax
wrote:  ./data/MCTSMaximum.zip.pickle            |  6.0MB in  8.3s | entries:
307631
```

[33]:
```
! python3 ./run_backpropagation.py  -a UCT -o ALPHABETA -r 70 --progress
```

```
loaded: ./data/AlphaBetaPlayer.zip.pickle        |  9.5MB in  2.8s | entries:
629184
loaded: ./data/MCTSMaximum.zip.pickle            |  6.0MB in  1.8s | entries:
307631
loaded: ./data/MCTSMaximum.zip.pickle            |  6.0MB in  1.7s | entries:
307631
--------------+-----------------+-----------------------------------+ match_id:
70 | 529s |   4% ->   5% | UCT vs AlphaBeta
wrote:  ./data/MCTSMaximum.zip.pickle            |  6.2MB in  8.8s | entries:
319067
wrote:  ./data/AlphaBetaPlayer.zip.pickle        | 10.8MB in 16.7s | entries:
719376
```

7 League Tables

Lets run every agent against every other agent and compare results - AlphaBetaArea has a 100% winrate against everything except AlphaBeta and MCTS Random - AlphaBetaArea vs AlphaBeta scores 60/40 both ways, which may depend on who has the bigger cache - MCTS Random scored a maximum 90% vs Alphabeta and 24% vs AlphaBeta Area in reverse matchups, - MCTS Random got beaten 100% Alphabeta and AlphaBeta later when in was in first position, this may be cache related

[3]:
```
for agent in TEST_AGENTS.keys():
    for opponent in TEST_AGENTS.keys():
        try: TEST_AGENTS[agent].agent_class.load()
```

```
        except: pass

        time.sleep(0.1)
        TEST_AGENTS[agent   ].agent_class.verbose = False
        TEST_AGENTS[opponent].agent_class.verbose = False
        is_slow = any(
            name in TEST_AGENTS[agent].name + TEST_AGENTS[opponent].name
            for name in ['Alpha', 'Area', 'UCT', 'Minimax', 'Custom']
        )
        run_backpropagation({
            "agent":      agent,
            "opponent":   opponent,
            "time_limit": 150,
            "rounds":     10  if is_slow else 100,
            "progress":   False,
            "exceptions": False,
        })
    print()

for agent in TEST_AGENTS.keys():
    try: TEST_AGENTS[agent].agent_class.save()
    except: pass
```

```
match_id:  100 |   1s |  52% ->  54% | Random vs Random 2
match_id:  100 |   1s |  17% ->  17% | Random vs Greedy
match_id:  100 |   1s |  61% ->  62% | Random vs Distance
match_id:  100 |   2s |  22% ->  21% | Random vs Greedy Distance
match_id:   10 |   3s |   0% ->   0% | Random vs Minimax
match_id:   10 |  41s |  30% ->  23% | Random vs AlphaBeta
match_id:   10 |  42s |   0% ->   0% | Random vs AlphaBeta Area
match_id:  100 |   2s |  70% ->  69% | Random vs MCTS Maximum
match_id:  100 |   4s |  50% ->  48% | Random vs MCTS Random
match_id:  100 |   2s |  70% ->  69% | Random vs MCTS Maximum Heuristic
match_id:  100 |   1s |  58% ->  62% | Random vs MCTS Random Heuristic
match_id:   10 |  35s |   0% ->   0% | Random vs UCT
match_id:   10 |  35s |   0% ->   0% | Random vs Custom TestAgent

match_id:  100 |   1s |  75% ->  75% | Greedy vs Random
match_id:  100 |   1s |  50% ->  50% | Greedy vs Greedy 2
match_id:  100 |   2s |  50% ->  50% | Greedy vs Distance
match_id:  100 |   2s |  50% ->  50% | Greedy vs Greedy Distance
match_id:   10 |   3s |   0% ->   0% | Greedy vs Minimax
match_id:   10 |  39s |  10% ->   1% | Greedy vs AlphaBeta
match_id:   10 |  31s |   0% ->   0% | Greedy vs AlphaBeta Area
match_id:  100 |   1s |  87% ->  89% | Greedy vs MCTS Maximum
match_id:  100 |   1s |  67% ->  64% | Greedy vs MCTS Random
match_id:  100 |   1s |  91% ->  90% | Greedy vs MCTS Maximum Heuristic
```

```
 match_id:  100 |   2s |  83% ->  82% | Greedy vs MCTS Random Heuristic
 match_id:   10 |  36s | 100% -> 100% | Greedy vs UCT
 match_id:   10 |  37s |   0% ->   0% | Greedy vs Custom TestAgent

 match_id:  100 |   1s |  40% ->  39% | Distance vs Random
 match_id:  100 |   1s |  50% ->  50% | Distance vs Greedy
 match_id:  100 |   2s |  50% ->  50% | Distance vs Distance 2
 match_id:  100 |   2s |   0% ->   0% | Distance vs Greedy Distance
 match_id:   10 |   2s |   0% ->   0% | Distance vs Minimax
 match_id:   10 |  51s |   0% ->   0% | Distance vs AlphaBeta
 match_id:   10 |  60s |   0% ->   0% | Distance vs AlphaBeta Area
 match_id:  100 |   2s |  50% ->  49% | Distance vs MCTS Maximum
 match_id:  100 |   2s |  31% ->  29% | Distance vs MCTS Random
 match_id:  100 |   2s |  22% ->  17% | Distance vs MCTS Maximum Heuristic
 match_id:  100 |   2s |  46% ->  48% | Distance vs MCTS Random Heuristic
 match_id:   10 |  40s |  50% ->  41% | Distance vs UCT
 match_id:   10 |  48s |   0% ->   0% | Distance vs Custom TestAgent

 match_id:  100 |   2s |  87% ->  90% | Greedy Distance vs Random
 match_id:  100 |   2s |  50% ->  50% | Greedy Distance vs Greedy
 match_id:  100 |   2s | 100% -> 100% | Greedy Distance vs Distance
 match_id:  100 |   2s |  50% ->  50% | Greedy Distance vs Greedy Distance 2
 match_id:   10 |   3s |  50% ->  45% | Greedy Distance vs Minimax
 match_id:   10 |  38s |  20% ->  10% | Greedy Distance vs AlphaBeta
 match_id:   10 |  35s |   0% ->   0% | Greedy Distance vs AlphaBeta Area
 match_id:  100 |   2s |  65% ->  68% | Greedy Distance vs MCTS Maximum
 match_id:  100 |   2s |  81% ->  80% | Greedy Distance vs MCTS Random
 match_id:  100 |   2s |  82% ->  76% | Greedy Distance vs MCTS Maximum
Heuristic
 match_id:  100 |   2s |  82% ->  79% | Greedy Distance vs MCTS Random Heuristic
 match_id:   10 |  33s |  90% ->  99% | Greedy Distance vs UCT
 match_id:   10 |  44s |  10% ->  17% | Greedy Distance vs Custom TestAgent

 match_id:   10 |   3s | 100% -> 100% | Minimax vs Random
 match_id:   10 |   3s | 100% -> 100% | Minimax vs Greedy
 match_id:   10 |   2s | 100% -> 100% | Minimax vs Distance
 match_id:   10 |   3s | 100% -> 100% | Minimax vs Greedy Distance
 match_id:   10 |   6s |  50% ->  45% | Minimax vs Minimax 2
 match_id:   10 |  44s |  20% ->  24% | Minimax vs AlphaBeta
 match_id:   10 |  42s |   0% ->   0% | Minimax vs AlphaBeta Area
 match_id:   10 |   2s | 100% -> 100% | Minimax vs MCTS Maximum
 match_id:   10 |  10s | 100% -> 100% | Minimax vs MCTS Random
 match_id:   10 |   2s | 100% -> 100% | Minimax vs MCTS Maximum Heuristic
 match_id:   10 |   3s | 100% -> 100% | Minimax vs MCTS Random Heuristic
 match_id:   10 |  37s | 100% -> 100% | Minimax vs UCT
 match_id:   10 |  47s |   0% ->   0% | Minimax vs Custom TestAgent

 match_id:   10 |  35s | 100% -> 100% | AlphaBeta vs Random
```

```
match_id:   10 |  37s |  90% ->  85% | AlphaBeta vs Greedy
match_id:   10 |  46s | 100% -> 100% | AlphaBeta vs Distance
match_id:   10 |  38s | 100% -> 100% | AlphaBeta vs Greedy Distance
match_id:   10 |  43s |  80% ->  92% | AlphaBeta vs Minimax
match_id:   10 |  96s |  40% ->  38% | AlphaBeta vs AlphaBeta 2
match_id:   10 |  84s |  20% ->  24% | AlphaBeta vs AlphaBeta Area
match_id:   10 |  25s | 100% -> 100% | AlphaBeta vs MCTS Maximum
match_id:   10 |  38s |  90% ->  95% | AlphaBeta vs MCTS Random
match_id:   10 |  31s | 100% -> 100% | AlphaBeta vs MCTS Maximum Heuristic
match_id:   10 |  36s | 100% -> 100% | AlphaBeta vs MCTS Random Heuristic
match_id:   10 |  91s | 100% -> 100% | AlphaBeta vs UCT
match_id:   10 |  86s |  40% ->  38% | AlphaBeta vs Custom TestAgent

match_id:   10 |  34s | 100% -> 100% | AlphaBeta Area vs Random
match_id:   10 |  31s | 100% -> 100% | AlphaBeta Area vs Greedy
match_id:   10 |  43s | 100% -> 100% | AlphaBeta Area vs Distance
match_id:   10 |  34s | 100% -> 100% | AlphaBeta Area vs Greedy Distance
match_id:   10 | 127s | 100% -> 100% | AlphaBeta Area vs Minimax
match_id:   10 |  89s |  40% ->  64% | AlphaBeta Area vs AlphaBeta
match_id:   10 |  82s |  60% ->  52% | AlphaBeta Area vs AlphaBeta Area 2
match_id:   10 |  30s | 100% -> 100% | AlphaBeta Area vs MCTS Maximum
match_id:   10 | 313s |  80% ->  76% | AlphaBeta Area vs MCTS Random
match_id:   10 |  30s | 100% -> 100% | AlphaBeta Area vs MCTS Maximum Heuristic
match_id:   10 |  37s | 100% -> 100% | AlphaBeta Area vs MCTS Random Heuristic
match_id:   10 |  75s | 100% -> 100% | AlphaBeta Area vs UCT
match_id:   10 |  90s |  50% ->  45% | AlphaBeta Area vs Custom TestAgent

match_id:  100 |   2s |  34% ->  34% | MCTS Maximum vs Random
match_id:  100 |   2s |  13% ->  15% | MCTS Maximum vs Greedy
match_id:  100 |   2s |  58% ->  57% | MCTS Maximum vs Distance
match_id:  100 |   2s |  19% ->  28% | MCTS Maximum vs Greedy Distance
match_id:   10 |  10s |   0% ->   0% | MCTS Maximum vs Minimax
match_id:   10 |  24s |   0% ->   0% | MCTS Maximum vs AlphaBeta
match_id:   10 |  36s |   0% ->   0% | MCTS Maximum vs AlphaBeta Area
match_id:  100 |   2s |  52% ->  51% | MCTS Maximum vs MCTS Maximum 2
match_id:  100 |   3s |  29% ->  36% | MCTS Maximum vs MCTS Random
match_id:  100 |   2s |  46% ->  41% | MCTS Maximum vs MCTS Maximum Heuristic
match_id:  100 |   2s |  36% ->  33% | MCTS Maximum vs MCTS Random Heuristic
match_id:   10 |  34s |  40% ->  44% | MCTS Maximum vs UCT
match_id:   10 |  33s |   0% ->   0% | MCTS Maximum vs Custom TestAgent

match_id:  100 |   1s |  46% ->  48% | MCTS Random vs Random
match_id:  100 |   1s |  32% ->  34% | MCTS Random vs Greedy
match_id:  100 |   2s |  64% ->  67% | MCTS Random vs Distance
match_id:  100 |   2s |  18% ->  16% | MCTS Random vs Greedy Distance
match_id:   10 |   4s |   0% ->   0% | MCTS Random vs Minimax
match_id:   10 | 425s |   0% ->   0% | MCTS Random vs AlphaBeta
match_id:   10 |  56s |   0% ->   0% | MCTS Random vs AlphaBeta Area
```

```
 match_id:  100 |   2s |  77% ->  81% | MCTS Random vs MCTS Maximum
 match_id:  100 |   2s |  44% ->  44% | MCTS Random vs MCTS Random 2
 match_id:  100 |   2s |  74% ->  75% | MCTS Random vs MCTS Maximum Heuristic
 match_id:  100 |   2s |  48% ->  49% | MCTS Random vs MCTS Random Heuristic
 match_id:   10 |  32s |  50% ->  39% | MCTS Random vs UCT
 match_id:   10 |  47s |   0% ->   0% | MCTS Random vs Custom TestAgent

 match_id:  100 |   1s |  29% ->  27% | MCTS Maximum Heuristic vs Random
 match_id:  100 |   2s |  14% ->  13% | MCTS Maximum Heuristic vs Greedy
 match_id:  100 |   2s |  64% ->  69% | MCTS Maximum Heuristic vs Distance
 match_id:  100 |   2s |  35% ->  34% | MCTS Maximum Heuristic vs Greedy
Distance
 match_id:   10 |   6s |   0% ->   0% | MCTS Maximum Heuristic vs Minimax
 match_id:   10 |  34s |   0% ->   0% | MCTS Maximum Heuristic vs AlphaBeta
 match_id:   10 |  32s |   0% ->   0% | MCTS Maximum Heuristic vs AlphaBeta Area
 match_id:  100 |   2s |  48% ->  46% | MCTS Maximum Heuristic vs MCTS Maximum
 match_id:  100 |   2s |  35% ->  33% | MCTS Maximum Heuristic vs MCTS Random
 match_id:  100 |   2s |  46% ->  50% | MCTS Maximum Heuristic vs MCTS Maximum
Heuristic 2
 match_id:  100 |   2s |  35% ->  36% | MCTS Maximum Heuristic vs MCTS Random
Heuristic
 match_id:   10 |  32s |  40% ->  52% | MCTS Maximum Heuristic vs UCT
 match_id:   10 |  31s |   0% ->   0% | MCTS Maximum Heuristic vs Custom
TestAgent
```

8 Opening Book

8.1 Monty Carlo Method

We can derive an opening book by looking at the cached scores in MCTSRandom - score priorities exploration of 100% winrate nodes that have only been explored once - sorting by score actually suggested the best countermove was the corner square - sort by wins * wins/count to find the best move

- Monty Carlo suggests the best opening strategy is:
  - defend the corner on the 3,3 point
  - attack the knights blind spot on the 4,3 point
    - knight requires three turns to move one space sideways
  - the first player is attempting to move into position to directly attack their opponent
  - the second player is attempting to remain in the blind spot adjcent to the first player
  - the pieces only diverge after move 8

[4]:
```
from agents.MCTS import MCTSRandom
from isolation.isolation import Isolation, DebugState
MCTSRandom.load()
```

```python
def best_response(parent_state=Isolation(), verbose=True):
    actions      = parent_state.actions()
    child_states = [ parent_state.result(action) for action in actions ]
    child_states = [ child for child in child_states if child in MCTSRandom.
↪data ]
    records      = [ MCTSRandom.data[child] for child in child_states  ]
    best_state, best_record = max(zip(child_states, records),
                                  key=lambda key_value: key_value[1].wins *␣
↪key_value[1].wins/key_value[1].count)
    best_move    = DebugState.ind2xy(best_state.locs[parent_state.player()])
    if verbose:
        print('Best move '+str(best_state.ply_count)+' is:')
        print(best_move, best_record)
        print( DebugState.from_state(best_state).bitboard_string )
        print( DebugState.from_state(best_state) )
    return best_move, best_record, best_state

best_state = Isolation()
for i in range(8):
    try:
        best_move, best_record, best_state = best_response(best_state)
    except: pass
```

```
loaded: ./data/MCTSRandom.zip.pickle             |  7.4MB in  2.6s | entries:
380447
Best move 1 is:
(2, 2) MCTSRecord(wins=977, count=1487, score=0.6567967698519516)
1111111111100111111111110011111111111001111111111100111111111110011111111111110011
11111101100111111111110011111111111

+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 1 |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
```

```
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
```

```
Best move 2 is:
(3, 2) MCTSRecord(wins=89, count=184, score=0.48633879781420764)
11111111111001111111111100111111111110011111111111001111111111100111111111110011
11111001100111111111110011111111111
```

```
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | 2 | 1 |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
```

```
Best move 3 is:
(1, 4) MCTSRecord(wins=50, count=85, score=0.5833333333333334)
11111111111001111111111100111111111110011111111111001111111110100111111111110011
11111001100111111111110011111111111
```

```
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | 1 |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | 2 | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
```

```
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
Best move 4 is:
(2, 4) MCTSRecord(wins=27, count=59, score=0.46551724137931033)
11111111111001111111111100111111111111001111111111100111111111001001111111111110011
1111100110011111111111100111111111
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 2 | 1 |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
Best move 5 is:
(0, 6) MCTSRecord(wins=32, count=56, score=0.5636363636363636)
11111111111001111111111100111111111110001111111111100111111111001001111111111110011
1111100110011111111111100111111111
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   | 1 |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 2 | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
|   |   |   |   |   |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

Best move 6 is:
(4, 3) MCTSRecord(wins=18, count=34, score=0.5454545454545454)
11111111111001111111111100111111111110001111111111100111111111001001111111011110011
11111001100111111111110011111111111

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   | 1 |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | 2 |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

Best move 7 is:
(2, 5) MCTSRecord(wins=16, count=32, score=0.4838709677419355)
11111111111001111111111100111111111110001111111101100111111111001001111111011110011
11111001100111111111110011111111111

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | 1 |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
|   |   |   |   |   |   |   | 2 |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
Best move 8 is:
(3, 5) MCTSRecord(wins=16, count=28, score=0.5925925925925926)
11111111111001111111111100111111111110001111111001100111111111001001111111011110011
11111001100111111111111001111111111
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 2 | 1 |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
```

8.2 AlphaBetaArea Method

Alteratively we could see how AlphaBetaArea plays when given different depths to search during the opening

Opening Move: - Monty Carlo: likes the 3,3 corner position - Depth 1: main diagonal, one off from center - Depth 2: short centerline, one off from center - Depth 3: center,3 position - Depth 4: corner 4,4 position - Depth 5: long-edge but one off from center

Subsequent Moves: - Usually ends up in either an attacking or adjacent square - Except at depth 4, when p1 approaches diagonally, p2 jumps away

[5]:
```
AlphaBetaAreaPlayer.load()
class AlphaBetaOpening(AlphaBetaAreaPlayer):
    data             = AlphaBetaAreaPlayer.data  # import cache
    verbose_depth    = True
for depth in range(1,6):
    AlphaBetaOpening.search_max_depth = depth
    play_sync(
        agents=(
            Agent(AlphaBetaOpening, 'AlphaBetaOpening'),
            Agent(AlphaBetaOpening, 'AlphaBetaOpening'),
        ),
        verbose=True,
        time_limit=0,
        max_moves=4,
    )
```

```
AlphaBetaOpening     | depth: 1
AlphaBetaOpening     | depth: 1
match: 0 | move: 2 | 0.06s | AlphaBetaOpening(1) => (6, 4)

+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | 2 |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | 1 |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

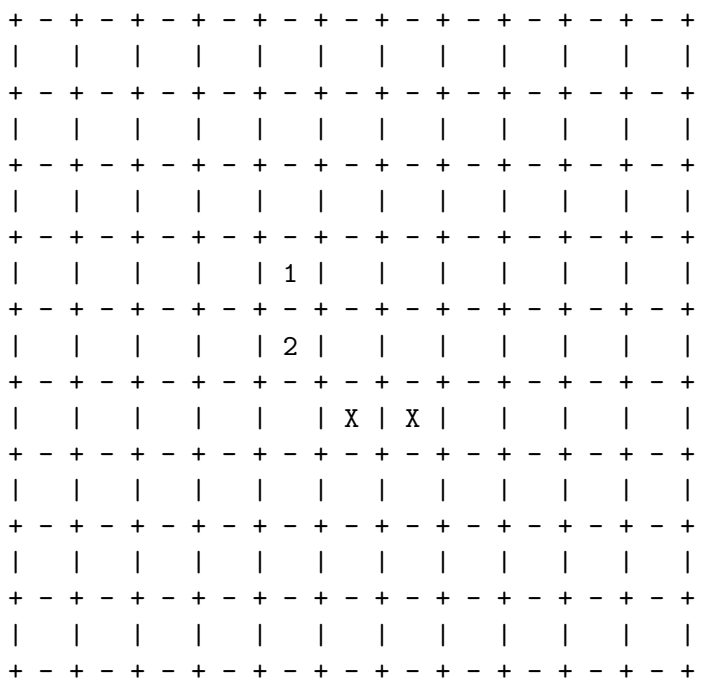
```
AlphaBetaOpening     | depth: 1
AlphaBetaOpening     | depth: 1
match: 0 | move: 4 | 0.01s | AlphaBetaOpening(1) => (11, 0)

+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | 1 | 2 |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

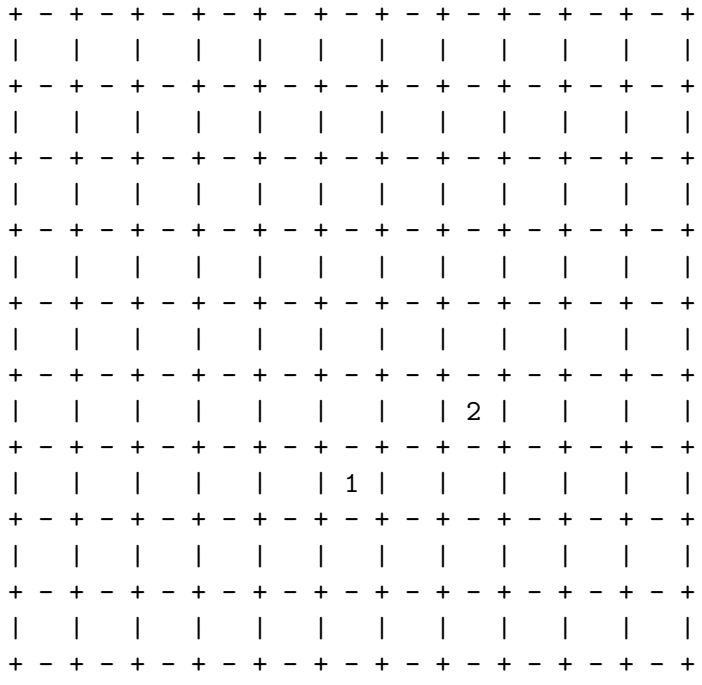
```
AlphaBetaOpening     | depth: 1 2
AlphaBetaOpening     | depth: 1 2
match: 0 | move: 2 | 0.73s | AlphaBetaOpening(1) => (4, 3)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | 1 | 2 |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaOpening     | depth: 1 2
AlphaBetaOpening     | depth: 1 2
match: 0 | move: 4 | 0.04s | AlphaBetaOpening(1) => (2, 1)
```

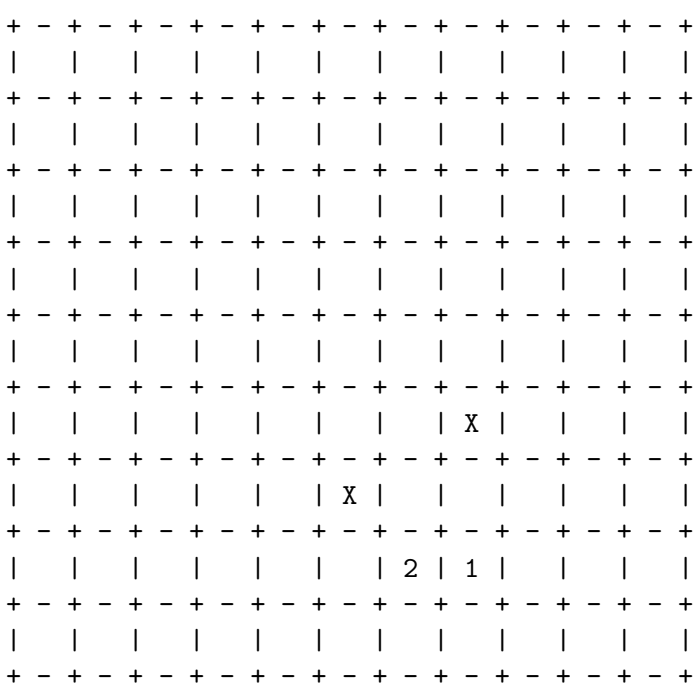



```
AlphaBetaOpening     | depth: 1 2 3
AlphaBetaOpening     | depth: 1 2 3
match: 0 | move: 2 | 1.65s | AlphaBetaOpening(1) => (3, 3)
```

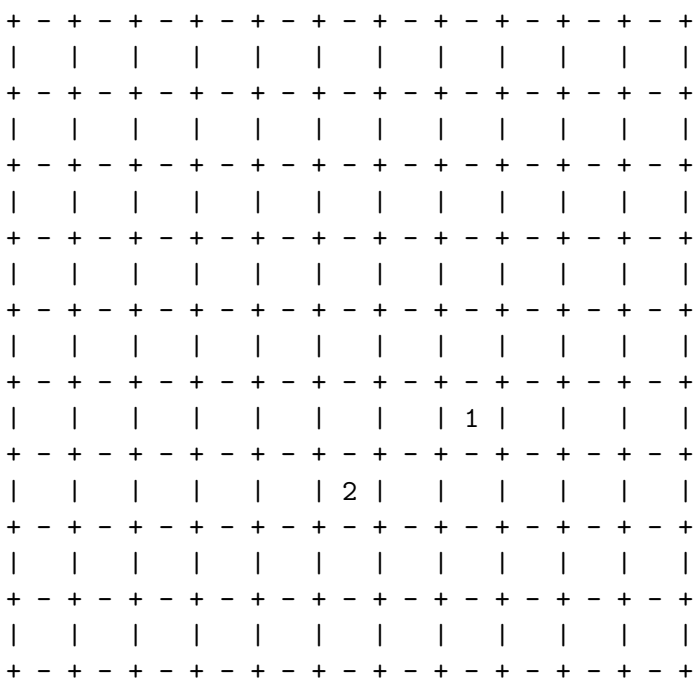



```
AlphaBetaOpening     | depth: 1 2 3
```

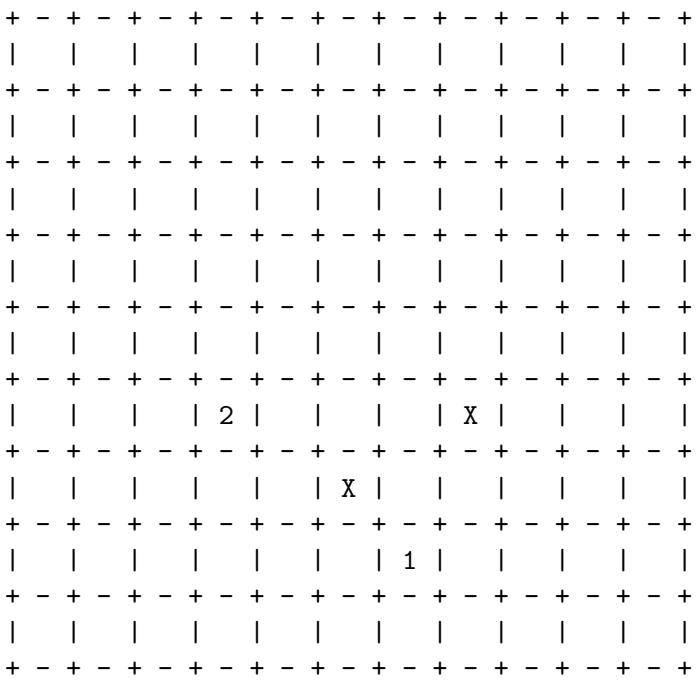
```
AlphaBetaOpening     | depth: 1 2 3
match: 0 | move: 4 | 0.11s | AlphaBetaOpening(1) => (1, -2)
```



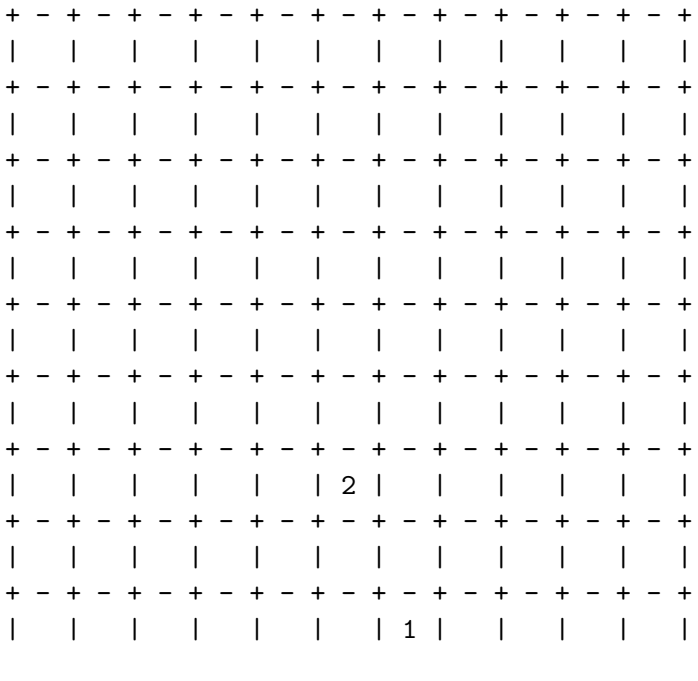

```
AlphaBetaOpening     | depth: 1 2 3 4
AlphaBetaOpening     | depth: 1 2 3 4
match: 0 | move: 2 | 9.82s | AlphaBetaOpening(1) => (5, 2)
```



```
AlphaBetaOpening     | depth: 1 2 3 4
AlphaBetaOpening     | depth: 1 2 3 4
match: 0 | move: 4 | 0.28s | AlphaBetaOpening(1) => (2, 1)
```




```
AlphaBetaOpening     | depth: 1 2 3 4 5
AlphaBetaOpening     | depth: 1 2 3 4 5
match: 0 | move: 2 | 32.56s | AlphaBetaOpening(1) => (5, 2)
```



```
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +


AlphaBetaOpening     | depth: 1 2 3 4 5
AlphaBetaOpening     | depth: 1 2 3 4 5
match: 0 | move: 4 | 1.46s | AlphaBetaOpening(1) => (11, 0)

+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 2 |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | 1 |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
```

9 Full Game (Final Section)

Here is a full game between AlphaBeta and AlphaBetaArea, with 1 second per move - Curiously, AlphaBeta at depth 2 picks the Monty Carlo 3,3 opening position - AlphaBetaArea(2) figures out forced checkmate in 26 on move 54 delivers it on move 65 (11 moves later) - The checkmate strategy for player 2 involves forcing 1 into a trapped corner

[18]:
```
AlphaBetaPlayer.verbose_depth     = True
AlphaBetaAreaPlayer.verbose_depth = True
play_sync(
    agents=(
        Agent(AlphaBetaPlayer,     'AlphaBeta'),
        Agent(AlphaBetaAreaPlayer, 'AlphaBetaArea'),
    ),
    verbose=True,
    verbose_depth=True,
    time_limit=1*1000,
)
```

```python
AlphaBetaPlayer.verbose_depth     = False
AlphaBetaAreaPlayer.verbose_depth = False
```

```
AlphaBetaPlayer      | depth: 1 2
AlphaBetaAreaPlayer  | depth: 1 2 3
match: 0 | move: 2 | 1.00s | AlphaBetaAreaPlayer(1) => (6, 2)

+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | 2 |   |   |   | 1 |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +


AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 4 | 1.00s | AlphaBetaAreaPlayer(1) => (2, 1)

+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | 1 |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   | 2 |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X |   |   |
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6
match: 0 | move: 6 | 1.00s | AlphaBetaAreaPlayer(1) => (12, 1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   | 1 |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | 2 |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 8 | 1.00s | AlphaBetaAreaPlayer(1) => (11, 0)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | 2 |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   |   |   | 1 |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   |   |   |   |   |   |   |
```

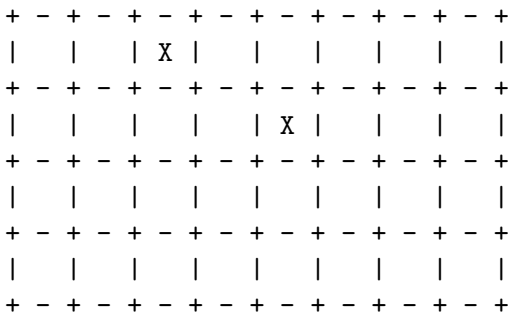
```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 10 | 1.01s | AlphaBetaAreaPlayer(1) => (11, -2)
```

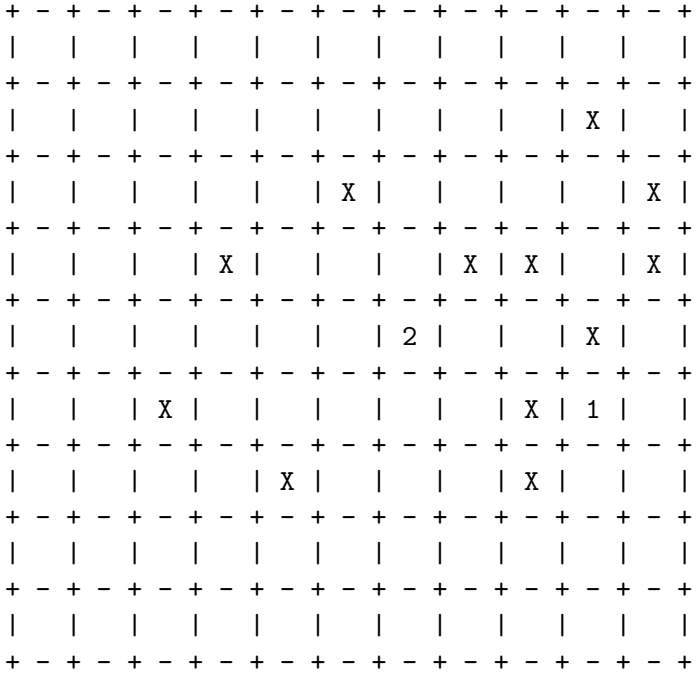
```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | 1 |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | 2 | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7
match: 0 | move: 12 | 1.00s | AlphaBetaAreaPlayer(1) => (12, -3)
```

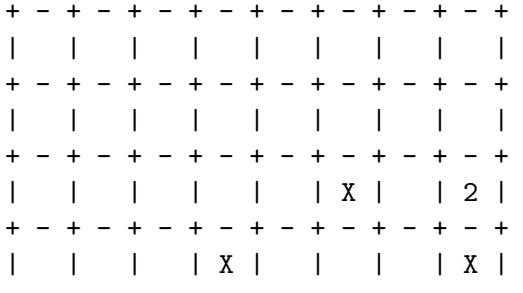
```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X | X |   | 1 |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```



```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6
match: 0 | move: 14 | 1.00s | AlphaBetaAreaPlayer(1) => (2, 1)
```



```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 16 | 1.01s | AlphaBetaAreaPlayer(1) => (12, 1)
```



```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 1 |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 18 | 1.00s | AlphaBetaAreaPlayer(1) => (11, -2)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   | X |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   | X | X | 2 | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | 1 |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6
match: 0 | move: 20 | 1.00s | AlphaBetaAreaPlayer(1) => (2, -1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   | X |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X | 2 |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   |   |   |   | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | 1 | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6
match: 0 | move: 22 | 1.00s | AlphaBetaAreaPlayer(1) => (2, -1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   | X |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X | X | 1 | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   | 2 |   |   | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 24 | 1.01s | AlphaBetaAreaPlayer(1) => (2, -1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   | X |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X | X | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   | X | 1 |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | 2 | X |   |   | X | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6
match: 0 | move: 26 | 1.00s | AlphaBetaAreaPlayer(1) => (12, 1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   | X |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | 2 |   |   | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   | X | X |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | 1 |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6
match: 0 | move: 28 | 1.00s | AlphaBetaAreaPlayer(1) => (2, 1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   | X |   |   | X |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | 2 | X |   |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   |   | X | X |   | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X | X |   |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | 1 |   |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6
match: 0 | move: 30 | 1.00s | AlphaBetaAreaPlayer(1) => (11, 0)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | 2 | X |   |   | X |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | 1 |   |   |   |   | X |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 32 | 1.00s | AlphaBetaAreaPlayer(1) => (1, -2)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X | X |   |   | X |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | 2 | X |   |   | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
| 1 |   |   | X |   |   |   | X | X |   |   | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X | X |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   |   | X |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

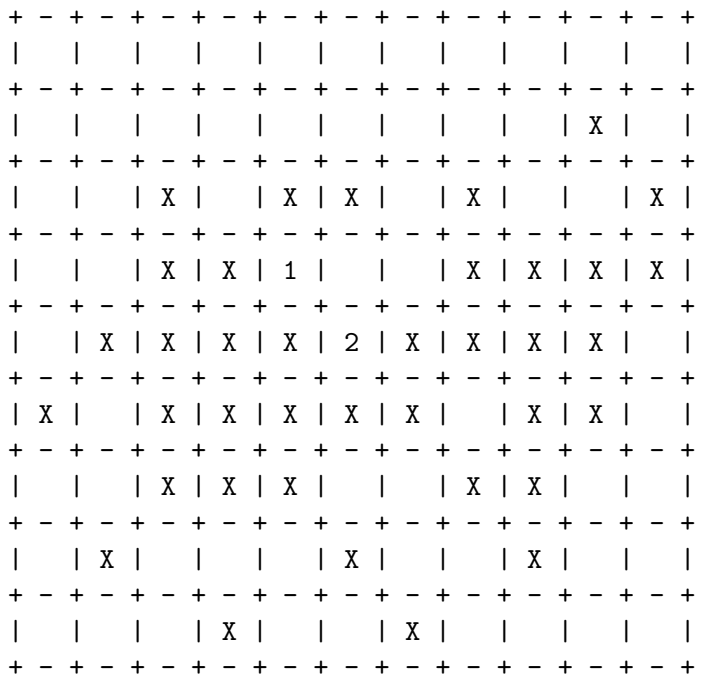
```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
match: 0 | move: 34 | 1.11s | AlphaBetaAreaPlayer(1) => (1, 2)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | 2 |   |   | X | X |   |   | X |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   |   |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X | X |   |   |   | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
| X |   |   | X |   |   |   | X | X |   |   | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | 1 | X | X |   |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   |   | X |   |   |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - + - +
```

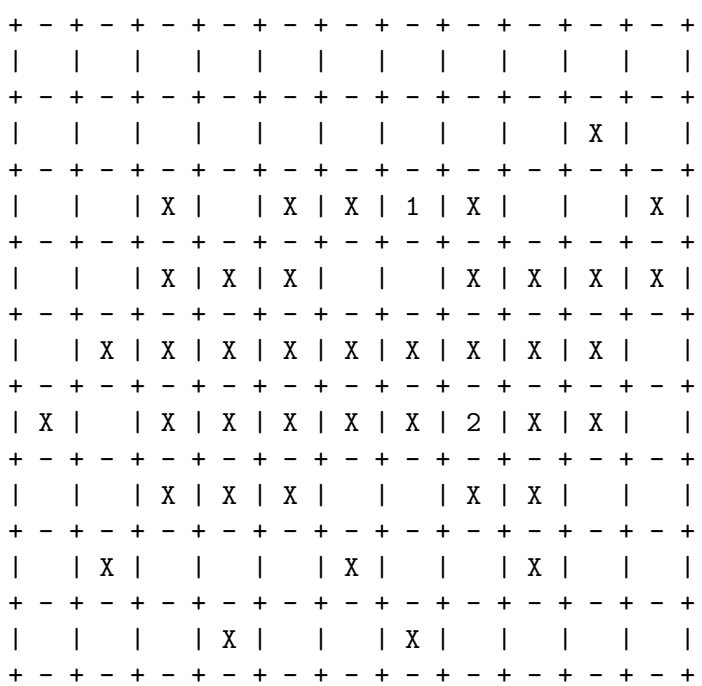
```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5
```

```
match: 0 | move: 36 | 1.02s | AlphaBetaAreaPlayer(1) => (1, -2)

+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X |   | X |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X |   |   |   | X | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | 2 |   | X | X |   | X | X | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X |   | 1 | X | X |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   |   |   | X |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   | X |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +


AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7
match: 0 | move: 38 | 1.01s | AlphaBetaAreaPlayer(1) => (11, -2)

+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X |   | X |   |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X |   |   |   | X | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | 1 | X | X |   | X | X | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X | 2 | X | X | X |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   |   |   | X |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   | X |   |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8
match: 0 | move: 40 | 1.00s | AlphaBetaAreaPlayer(1) => (11, 0)
```



```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10
match: 0 | move: 42 | 1.00s | AlphaBetaAreaPlayer(1) => (11, -2)
```



```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10
match: 0 | move: 44 | 1.00s | AlphaBetaAreaPlayer(1) => (1, -2)

+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 1 |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   | X | X | X | X |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   |   | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   | X | 2 |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11
match: 0 | move: 46 | 1.00s | AlphaBetaAreaPlayer(1) => (2, -1)

+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | X |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   | X | X | X | X | 1 |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   |   | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
|   |   |   | X | 2 |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14
match: 0 | move: 48 | 1.00s | AlphaBetaAreaPlayer(1) => (2, 1)

+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | X |   | 1 |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   |   |   |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   | X | X | X | X | X |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   |   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | 2 |   |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   | X | X |   |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14
match: 0 | move: 50 | 1.00s | AlphaBetaAreaPlayer(1) => (1, 2)

+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   |   | 1 |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X |   |   | X | X | X | X | X |   |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | 2 | X | X | X | X | X | X | X | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X | X |   |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24
match: 0 | move: 52 | 0.99s | AlphaBetaAreaPlayer(1) => (1, 2)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | 1 |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   |   |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X | X | X | X |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| 2 |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 -inf
AlphaBetaAreaPlayer  | depth: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 inf
match: 0 | move: 54 | 0.31s | AlphaBetaAreaPlayer(1) => (12, 1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | 2 |   | 1 |   |   |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X | X | X | X |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | X |   |
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 -inf
AlphaBetaAreaPlayer  | depth: 1 inf
match: 0 | move: 56 | 0.00s | AlphaBetaAreaPlayer(1) => (11, -2)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   |   |   | X |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   | X |   |   |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | 1 | X | 2 | X | X | X | X | X |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 -inf
AlphaBetaAreaPlayer  | depth: 1 inf
match: 0 | move: 58 | 0.00s | AlphaBetaAreaPlayer(1) => (12, 1)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | 1 |   | 2 | X |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   | X |   |   |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 -inf
AlphaBetaAreaPlayer  | depth: 1 inf
match: 0 | move: 60 | 0.00s | AlphaBetaAreaPlayer(1) => (11, -2)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X |   | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   | X | 1 |   | 2 | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X |   | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 -inf
AlphaBetaAreaPlayer  | depth: 1 inf
match: 0 | move: 62 | 0.01s | AlphaBetaAreaPlayer(1) => (11, 0)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X | 1 | X | 2 | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   | X | X |   | X | X |   | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X |   | X |
```

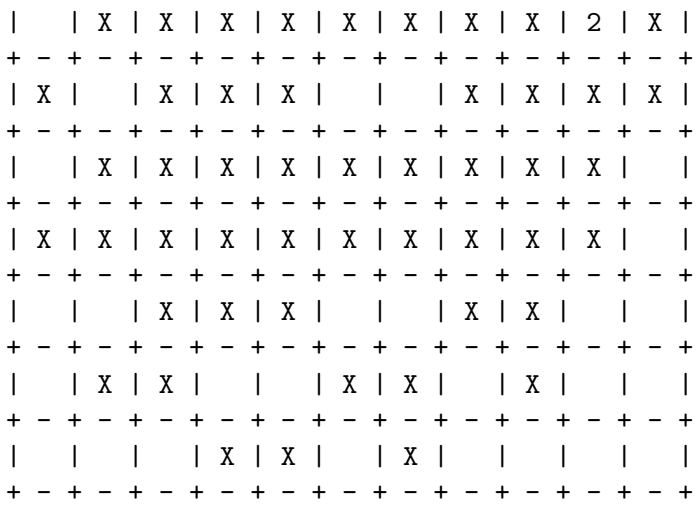
```
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 -inf
AlphaBetaAreaPlayer  | depth: 1 inf
match: 0 | move: 64 | 0.00s | AlphaBetaAreaPlayer(1) => (12, -3)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   | X | X |   | X | X | 1 | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | 2 | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X |   | X | X | X |   |   | X | X | X | X |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
| X | X | X | X | X | X | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X | X | X |   |   | X | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X | X |   |   | X | X |   | X |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   |   | X | X |   | X |   |   |   |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```

```
AlphaBetaPlayer      | depth: 1 -inf
match: 0 | move: 65 | 0.00s | AlphaBetaPlayer(0) => (11, 0)
```

```
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   |   | X |   | X | X | X | X | X | X | 1 |
+ - + - + - + - + - + - + - + - + - + - + - + - +
|   | X |   | X | X |   | X | X | X | X |   |
+ - + - + - + - + - + - + - + - + - + - + - + - +
```


	X	X	X	X	X	X	X	X	2	X
X		X	X	X			X	X	X	X	X
	X	X	X	X	X	X	X	X	X	X		
X	X	X	X	X	X	X	X	X	X	X		
			X	X	X			X	X			
	X	X				X	X		X			
				X	X		X					