

README

Instructions:

In your command window, run:

```
FileLocation\Project_4_Konane> py -2.7 game.py
```

(or other python 2 version)

Next, specify if you wish to play the game against an ai, or if you want an ai to play another ai.

The options are "AlvAI" and "PlayervAI".

Next, specify which type of ai you want to play the game.

The options are "Random", "Minimax", and "MinimaxAlphaBeta".

Random - the ai picks a random legal move

Minimax - the ai uses minimax algorithm to pick a more optimal move

MinimaxAlphaBeta - the ai uses minimax algorithm with alpha beta pruning to pick a more optimal move

Note: for AlvAI, one AI is the specified AI, while the other is the Random AI.

Next, if you chose Minimax or MinimaxAlphaBeta, choose a depth of 2, 4, or 6.

If you chose PlayervAI:

Choose to "Start" (be Black player), or "Divert" to ai (be the white player).

If you chose "Start":

Choose 1, 4, 5, or 8 to remove the piece at (1, 1), (4, 4), (5, 5), or (8, 8) respectively.

If you chose "Divert":

The AI will choose a piece to remove.

Then, you are given possible moves. First enter the starting coordinate of the piece you wish to move: row, col. Next, enter the ending coordinate of the piece you wish to move: row, col.

This will continue on until the game is finished.

Example PlayervAI game:

```
C:\Users\Chloe Dorward\laf_databases\Project_4_Konane>py -2.7 game.py
```

```
  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W B W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

```
AIvAI or PlayervAI: PlayervAI
```

```
Random, Minimax, or MinimaxAlphaBeta: Random
```

```
Random
```

```
Start or Divert to ai: Start
```

```
Start
```

```
Choose 1, 4, 5, 8: 1
```

```
  1 2 3 4 5 6 7 8
1 . W B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W B W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

```
  1 2 3 4 5 6 7 8
1 . . B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W B W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

```

  1 2 3 4 5 6 7 8
1 . . B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W B W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

```

Possible Moves:

```

[((2, 0), (0, 0))]
[((2, 0), (0, 0))]
((2, 0), (0, 0))

```

Player W won the game

Coordinate of piece: 3, 1

Coordinate to move piece: 1, 1

```

  1 2 3 4 5 6 7 8
1 . . B W B W B W
2 . B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W B W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

```

```

  1 2 3 4 5 6 7 8
1 . . B W B W B W
2 . B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W B W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

```

Possible Moves:

```

[]
[]

```

Player W won the game

Player lost the game

('Total AI time: ', 0)

('Time elapsed: ', 38.25099992752075, ' seconds')

('Static Evaluations: ', 0)

Traceback (most recent call last):

File "game.py", line 380, in <module>

print("Average branching factor: ", num_branches/(calls+0.0))

ZeroDivisionError: float division by zero

Example AIvAI game:

```
C:\Users\Chloe Dorward\laf_databases\Project_4_Konane>py -2.7 game.py
```

```
  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W B W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

AIVAI or PlayervAI: AIVAI

Random, Minimax, or MinimaxAlphaBeta: Random

Random

(4, 4)

```
  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W B W B W B W B
5 B W B W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

(3, 4)

```
  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W B W B . B W B
5 B W B W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

```

  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W B W B W B W
4 W . . B . B W B
5 B W B W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

('Possible Moves for Random AI: ', [((1, 2), (3, 2)), ((1, 4), (3, 4)), ((3, 6), (3, 4)), ((3, 6),
(3, 2)), ((5, 2), (3, 2))])
  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W . W B W B W
4 W . . B . B W B
5 B W B W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W . W B W B W
4 W . . B . . W B
5 B W B W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

('Possible Moves for Random AI: ', [((1, 4), (3, 4)), ((5, 2), (3, 2))])
  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 W B W B W B W B
3 B W . W B W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 . B W B W B W .
3 B W . W B W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W

```

('Possible Moves for Random AI: ', [((1, 2), (1, 0)), ((1, 4), (3, 4)), ((3, 0), (1, 0))])

```
  1 2 3 4 5 6 7 8
1 B W B W B W B W
2 . B W B W B W .
3 B W . W . W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

```
  1 2 3 4 5 6 7 8
1 B . B W B W B W
2 . . W B W B W .
3 B W . W . W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

('Possible Moves for Random AI: ', [((0, 3), (0, 1)), ((3, 0), (1, 0))])

```
  1 2 3 4 5 6 7 8
1 B . B W B W B W
2 . . W B W B W .
3 . W . W . W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

```
  1 2 3 4 5 6 7 8
1 B . B W B W B W
2 . . W B . B W .
3 . W . W . W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B
```

('Possible Moves for Random AI: ', [((0, 3), (0, 1)), ((1, 2), (1, 4)), ((1, 6), (1, 4))])

```
  1 2 3 4 5 6 7 8
1 B . B W B W B W
2 . . . . W B W .
3 . W . W . W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
```

('Possible Moves for Random AI: ', [((0, 3), (0, 1)), ((1, 2), (1, 4)), ((1, 6), (1, 4))])

	1	2	3	4	5	6	7	8
1	B	.	B	W	B	W	B	W
2	W	B	W	.
3	.	W	.	W	.	W	B	W
4	W	.	.	B	.	.	W	B
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	B	W	B	W	B	W	B	W
8	W	B	W	B	W	B	W	B

	1	2	3	4	5	6	7	8
1	B	.	B	W	B	W	B	W
2	B	W	.
3	.	W	.	W	.	W	B	W
4	W	.	.	B	.	.	W	B
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	B	W	B	W	B	W	B	W
8	W	B	W	B	W	B	W	B

('Possible Moves for Random AI: ', [((0, 3), (0, 1)), ((1, 6), (1, 4))])

	1	2	3	4	5	6	7	8
1	B	W	.	.	B	W	B	W
2	B	W	.
3	.	W	.	W	.	W	B	W
4	W	.	.	B	.	.	W	B
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	B	W	B	W	B	W	B	W
8	W	B	W	B	W	B	W	B

	1	2	3	4	5	6	7	8
1	B	W	.	.	B	W	B	W
2	B	W	.
3	.	W	.	W	.	W	B	W
4	W	.	.	B	.	.	W	B
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	B	W	B	W	B	W	B	W
8	W	B	W	B	W	B	W	B

('Possible Moves for Random AI: ', [((0, 5), (0, 3)), ((1, 6), (1, 4))])

	1	2	3	4	5	6	7	8
1	B	W	.	W	.	.	B	W
2	B	W	.
3	.	W	.	W	.	W	B	W
4	W	.	.	B	.	.	W	B
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	B	W	B	W	B	W	B	W
8	W	B	W	B	W	B	W	B


```

  1 2 3 4 5 6 7 8
1 B W . W . . B W
2 . . . . . W B
3 . W . W . W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

```

('Possible Moves for Random AI: ', [((0, 7), (0, 5))])

```

  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W B
3 . W . W . W B W
4 W . . B . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

```

```

  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W B
3 . W . W . W B W
4 W . . . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

```

('Possible Moves for Random AI: ', [((2, 7), (0, 7))])

```

  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W .
3 . W . W . W B W
4 W . . . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 W B W B W B W B

```

```

  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W .
3 . W . W . W B W
4 W . . . . W B
5 B W . W . W B W
6 W B W B W B W B
7 B W B W B W B W
8 . . . . . B W .

```

('Possible Moves for Random AI: ', [((5, 0), (7, 0)), ((5, 2), (7, 2)), ((5, 4), (7, 4)), ((7, 6), (7, 4))])

	1	2	3	4	5	6	7	8
1	B	W	.	W	.	W	.	.
2	W	.
3	.	W	.	W	.	W	B	W
4	W	W	B
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	B	W	.	W	B	W	B	W
8	B	W	.

	1	2	3	4	5	6	7	8
1	B	W	.	W	.	W	.	.
2	W	.
3	.	W	.	W	.	W	B	W
4	.	.	.	B	.	.	W	.
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	B	W	.	W	B	W	B	W
8	B	W	.

('Possible Moves for Random AI: ', [((5, 0), (3, 0)), ((5, 0), (7, 0)), ((5, 4), (7, 4)), ((7, 6), (7, 4))])

	1	2	3	4	5	6	7	8
1	B	W	.	W	.	W	.	.
2	W	.
3	.	W	.	W	.	W	B	W
4	.	.	.	B	.	.	W	.
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	.	W	.	W	B	W	B	W
8	B	W	.

	1	2	3	4	5	6	7	8
1	B	W	.	W	.	W	.	.
2	W	.
3	.	W	.	W	.	W	B	W
4	W	.
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B
7	.	W	.	W	B	W	B	W
8	B	W	.

('Possible Moves for Random AI: ', [((5, 0), (3, 0)), ((5, 4), (7, 4)), ((7, 6), (7, 4))])

	1	2	3	4	5	6	7	8
1	B	W	.	W	.	W	.	.
2	W	.
3	.	W	.	W	.	W	B	W
4	W	.
5	B	W	.	W	.	W	B	W
6	W	B	W	B	W	B	W	B

```

  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W .
3 . W . W . W B W
4 . . . . . W .
5 B W . . . W B W
6 W B W B W B W B
7 . W . W B W B W
8 . . . . W . . .

('Possible Moves for Random AI: ', [((5, 0), (3, 0)), ((5, 6), (7, 6)), ((6, 3), (4, 3))])
  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W .
3 . W . W . W B W
4 . . . . . W .
5 B W . . . W B W
6 W B W . W B W B
7 . W . W B W B W
8 . . . . W . . .

  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W .
3 . W . W . W B W
4 . . . . . W .
5 B W . . . W B W
6 . . . . W B W .
7 . W . W B W B W
8 . . . . W . . .

('Possible Moves for Random AI: ', [((5, 6), (7, 6))])
  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W .
3 . W . W . W B W
4 . . . . . W .
5 B W . . . W B W
6 . . . . W B W .
7 . W . W B W . W
8 . . . . W . . .

  1 2 3 4 5 6 7 8
1 B W . W . W . .
2 . . . . . W .
3 . W . W . W B W
4 . . . . . W .
5 B W . . . W B W
6 . . . . W B W .
7 . W . W B W . W
8 . . . . W . . .

```

```
('Possible Moves for Random AI: ', [])
Player B won the game
('Total AI time: ', 0.178999662399292)
('Time elapsed: ', 5.575000047683716, ' seconds')
('Static Evaluations: ', 0)
Traceback (most recent call last):
  File "game.py", line 374, in <module>
    print("Average branching factor: ", num_branches/(calls+0.0))
ZeroDivisionError: float division by zero

C:\Users\Chloe Dorward\laf_databases\Project_4_Konane>
```