2.1 Minimax and alpha-beta agents

The backbone of the "War" game is the **Board** class:

```
public class Board{
	/* Essential Info */
	public int rows;
	public int columns;
	public Tile[][] tile;

	/* Statuses */
	public Color playerTurn;
	public int blueScore;
	public int greenScore;
	public boolean gameEnded;
	public Color winner;
	ArrayList<Point> blitzPoint;

	public ArrayList<Board> getAdjacentBoards(){ /* Details omitted */ }

	public int utility(){ /* Details omitted */}

	// Other methods omitted
}
```

A **Board** is constructed by reading in a text file line by line. Each line of "{X}.txt" gives us a row of tiles. Reading the entire text file lets us create the **Tile**[][]. Once the **Tile**[][] is complete, we initialize the remaining variables. **getAdjacentBoards**() is a crucial function that determines which moves are valid on a certain **Board**.

The **Board** class relies on the simple **Tile** class:

```
public class Tile {
	public Color color;
	public final int value;

	// Methods omitted
}
```

A **Tile** has a *Color* that is either BLUE, GREEN, or NONE. Each tile also has a value that is an integer between 1 and 99.

Evaluation Function

The **utility()** function uses a simple evaluation function of: blueScore - greenScore for inner nodes, and "maxScore" for terminal nodes, which is a sum of all the values in the **Tiles.** This ensures the utility in the inner nodes never exceeds the max & min utilities at the terminal nodes.

One of our simple strategies is **Minimax** search:

```
public class Minimax {

	public int depth;
	public int nodesExpanded;
	public double elapsedTime;

	public Board minimax(Board board){ /* Details omitted */ }

	// Other methods omitted
}
```

This **Minimax** strategy is to choose the move that gives the best worst-case payoff. This strategy is optimal against an optimal opponent. The strength of this strategy is determined by depth, which is the number of levels of the search tree that we can search. The **minimax(Board board)** function returns the best calculated move as a **Board**.

An improved strategy is the **AlphaBeta** search strategy:

```
public class AlphaBeta {

	public int depth;
	public int nodesExpanded;
	public double elapsedTime;

	public Board alphaBeta(Board board){ /* Details omitted */ }

	// Other methods omitted
}
```

The basic premise that **AlphaBeta** search relies on is that it is possible to compute the exact **Minimax** decision without expanding every node in the game tree. This enables us to search at a larger depth to get a more accurate result.

The variables and functions are very similar to **Minimax** search.

To generate results on the 5 given gameboards, we set the depth of **Minimax** to 3, and the depth of **AlphaBeta** to 4. Blue player is the "Max" player and makes the first move. Green player is the "Min" player who makes the second move. Players alternate turns.

Move Ordering to Increase Pruning

To improve pruning, move ordering was optimized for **AlphaBeta** search. Specifically, when choosing which move to try first, the move that resulted in the largest immediate increase in utility was tried first. This was done by sorting the moves (well, in the implementation, it was each **Board**), by using a custom Comparator. This lets us search at a larger depth for **AlphaBeta** than for **Minimax**.

Results are shown on the following pages:

Keren.txt

```
1     1     1     1     1     1
1     1     1     1     1     1
1     1     1     1     1     1
1     1     1     1     1     1
1     1     1     1     1     1
1     1     1     1     1     1
```

Matchup: Minimax(3) vs. Minimax(3)

*** blueScore = 18   greenScore = 18 ***

```
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
```

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10488888888888888 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09133333333333332 seconds

Matchup: AlphaBeta(4) vs. AlphaBeta(4)

*** blueScore = 18   greenScore = 18 ***

```
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
```

Blue Player total nodes expanded = 367799
Blue Player Nodes expanded per move = 20433
Blue Player Time per move = 2.2006111111111113 seconds

Green Player total nodes expanded = 313602
Green Player Nodes expanded per move = 17422
Green Player Time per move = 1.8297777777777773 seconds

Matchup: Minimax(3) vs. AlphaBeta(4)

*** blueScore = 18   greenScore = 18 ***

```
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
```

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10933333333333334 seconds

Green Player total nodes expanded = 313602
Green Player Nodes expanded per move = 17422
Green Player Time per move = 1.857277777777778 seconds

Matchup: AlphaBeta(4) vs. Minimax(3)

*** blueScore = 18   greenScore = 18 ***

```
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
B     G     B     G     B     G
G     B     G     B     G     B
```

Blue Player total nodes expanded = 367799
Blue Player Nodes expanded per move = 20433
Blue Player Time per move = 2.2546111111111102 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09627777777777775 seconds

Narvik.txt

```
99    1    99    1    99    1
1    99    1    99    1    99
99    1    99    1    99    1
1    99    1    99    1    99
99    1    99    1    99    1
1    99    1    99    1    99
```

Matchup: Minimax(3) vs. Minimax(3)

*** blueScore = 1096   greenScore = 704 ***

```
B    G    B    G    G    G
G    B    B    G    B    G
B    B    B    G    G    G
B    B    G    G    B    G
B    G    B    B    B    G
B    B    G    B    G    G
```

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.11127777777777778 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09844444444444443 seconds

Matchup: AlphaBeta(4) vs. AlphaBeta(4)

*** blueScore = 1096   greenScore = 704 ***

```
B    G    B    G    G    G
G    B    B    G    B    G
B    B    B    G    G    G
B    B    G    G    B    G
B    G    B    B    B    G
B    B    G    B    G    G
```

Blue Player total nodes expanded = 400790
Blue Player Nodes expanded per move = 22266
Blue Player Time per move = 2.2977222222222218 seconds

Green Player total nodes expanded = 354988
Green Player Nodes expanded per move = 19721
Green Player Time per move = 1.852888888888889 seconds

Matchup: Minimax(3) vs. AlphaBeta(4)

*** blueScore = 1096   greenScore = 704 ***

B G B G G G
G B B G B G
B B B G G G
B B G G B G
B G B B B G
B B G B G G

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10999999999999999 seconds

Green Player total nodes expanded = 354988
Green Player Nodes expanded per move = 19721
Green Player Time per move = 1.916888888888889 seconds

Matchup: AlphaBeta(4) vs. Minimax(3)

*** blueScore = 1096   greenScore = 704 ***

B G B G G G
G B B G B G
B B B G G G
B B G G B G
B G B B B G
B B G B G G

Blue Player total nodes expanded = 400790
Blue Player Nodes expanded per move = 22266
Blue Player Time per move = 2.3285 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09166666666666667 seconds

Sevastopol.txt

1	1	1	1	1	1
2	2	2	2	2	2
4	4	4	4	4	4
8	8	8	8	8	8
16	16	16	16	16	16
32	32	32	32	32	32

Matchup: Minimax(3) vs. Minimax(3)

*** blueScore = 189   greenScore = 189 ***

G B G B G B
B G B G B G
G B G B G B
B G B G B G
G B G B G B
B G B G B G

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10638888888888888 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09849999999999998 seconds

Matchup: AlphaBeta(4) vs. AlphaBeta(4)

*** blueScore = 189   greenScore = 189 ***

G B G B G B
B G B G B G
G B G B G B
B G B G B G
G B G B G B
B G B G B G

Blue Player total nodes expanded = 431445
Blue Player Nodes expanded per move = 23969
Blue Player Time per move = 2.5416666666666665 seconds

Green Player total nodes expanded = 245831
Green Player Nodes expanded per move = 13657
Green Player Time per move = 1.2976666666666672 seconds

Matchup: Minimax(3) vs. AlphaBeta(4)

*** blueScore = 189   greenScore = 189 ***

G B G B G B
B G B G B G
G B G B G B
B G B G B G
G B G B G B
B G B G B G

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10316666666666666 seconds

Green Player total nodes expanded = 245831
Green Player Nodes expanded per move = 13657
Green Player Time per move = 1.2546111111111111 seconds

Matchup: AlphaBeta(4) vs. Minimax(3)

*** blueScore = 189   greenScore = 189 ***

G B G B G B
B G B G B G
G B G B G B
B G B G B G
G B G B G B
B G B G B G

Blue Player total nodes expanded = 431445
Blue Player Nodes expanded per move = 23969
Blue Player Time per move = 2.366277777777777 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.0913333333333333 seconds

Smolensk.txt

66	76	28	66	11	9
31	39	50	8	33	14
80	76	39	59	2	48
50	73	43	3	13	3
99	45	72	87	49	4
80	63	92	28	61	53

Matchup: Minimax(3) vs. Minimax(3)

*** blueScore = 855 greenScore = 798 ***

B B G B B B
G B B G B B
B B G G G G
G B G G G G
B G B G G G
G B G B G G

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10549999999999998 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09294444444444443 seconds

Matchup: AlphaBeta(4) vs. AlphaBeta(4)

*** blueScore = 990 greenScore = 663 ***

B B G B G B
B G B G B B
B B B B G B
G B B B B G
B G G G B G
G B G G G G

Blue Player total nodes expanded = 386407
Blue Player Nodes expanded per move = 21467
Blue Player Time per move = 1.995888888888889 seconds

Green Player total nodes expanded = 318159
Green Player Nodes expanded per move = 17675
Green Player Time per move = 1.5657777777777777 seconds

Matchup: Minimax(3) vs. AlphaBeta(4)

*** blueScore = 1022   greenScore = 631 ***

B B G B G B
G B B B B B
B B B B B B
G B B B G G
B G B G G B
G B G G G G

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.11161111111111108 seconds

Green Player total nodes expanded = 316693
Green Player Nodes expanded per move = 17594
Green Player Time per move = 1.5320000000000005 seconds

Matchup: AlphaBeta(4) vs. Minimax(3)

*** blueScore = 948   greenScore = 705 ***

B B G B G G
G B B B G G
B B G B B G
G B B B G G
B B B G G G
G B G B G G

Blue Player total nodes expanded = 364928
Blue Player Nodes expanded per move = 20273
Blue Player Time per move = 1.930333333333333 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09505555555555557 seconds

Westerplatte.txt

1	1	1	1	1	1
1	3	4	4	3	1
1	4	2	2	4	1
1	4	2	2	4	1
1	3	4	4	3	1
1	1	1	1	1	1

Matchup: Minimax(3) vs. Minimax(3)

*** blueScore = 31   greenScore = 41 ***

B	B	G	B	G	G
B	B	B	G	G	G
B	B	B	G	G	G
B	B	B	G	G	G
B	B	G	G	G	G
B	G	B	G	G	G

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10666666666666663 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.093 seconds

Matchup: AlphaBeta(4) vs. AlphaBeta(4)

*** blueScore = 34   greenScore = 38 ***

B	B	G	B	G	G
B	B	B	G	G	G
B	B	B	G	G	G
B	B	G	G	G	G
B	B	B	G	G	G
B	B	G	B	G	G

Blue Player total nodes expanded = 358009
Blue Player Nodes expanded per move = 19889
Blue Player Time per move = 2.0475555555555562 seconds

Green Player total nodes expanded = 268359
Green Player Nodes expanded per move = 14908
Green Player Time per move = 1.4438888888888888 seconds

Matchup: Minimax(3) vs. AlphaBeta(4)

*** blueScore = 31   greenScore = 41 ***

B B G B G G
B B B G G G
B B B G G G
B B B G G G
B B G G G G
B G B G G G

Blue Player total nodes expanded = 217740
Blue Player Nodes expanded per move = 12096
Blue Player Time per move = 0.10705555555555554 seconds

Green Player total nodes expanded = 267513
Green Player Nodes expanded per move = 14861
Green Player Time per move = 1.4253333333333333 seconds

Matchup: AlphaBeta(4) vs. Minimax(3)

*** blueScore = 34   greenScore = 38 ***

B B G B G G
B B B G G G
B B B G G G
B B G G G G
B B B G G G
B B G B G G

Blue Player total nodes expanded = 358009
Blue Player Nodes expanded per move = 19889
Blue Player Time per move = 2.0593333333333335 seconds

Green Player total nodes expanded = 194736
Green Player Nodes expanded per move = 10818
Green Player Time per move = 0.09211111111111109 seconds

Animations (Extra Credit)

I have included 20 animations (in .gif format as “animated GIFS”), corresponding to All 5 game boards, and all 4 matchups.

These animations can be found in the ZIP file in the “Animations” folder.

1. **Keren.txt**
    a. Minimax(3) vs. Minimax(3)
    b. AlphaBeta(4) vs. AlphaBeta(4)
    c. Minimax(3) vs. AlphaBeta(4)
    d. AlphaBeta(4) vs. Minimax(4)

2. **Narvik.txt**
    a. Minimax(3) vs. Minimax(3)
    b. AlphaBeta(4) vs. AlphaBeta(4)
    c. Minimax(3) vs. AlphaBeta(4)
    d. AlphaBeta(4) vs. Minimax(4)

3. **Sevastopol.txt**
    a. Minimax(3) vs. Minimax(3)
    b. AlphaBeta(4) vs. AlphaBeta(4)
    c. Minimax(3) vs. AlphaBeta(4)
    d. AlphaBeta(4) vs. Minimax(4)

4. **Smolensk.txt**
    a. Minimax(3) vs. Minimax(3)
    b. AlphaBeta(4) vs. AlphaBeta(4)
    c. Minimax(3) vs. AlphaBeta(4)
    d. AlphaBeta(4) vs. Minimax(4)

5. **Westerplatte.txt**
    a. Minimax(3) vs. Minimax(3)
    b. AlphaBeta(4) vs. AlphaBeta(4)
    c. Minimax(3) vs. AlphaBeta(4)
    d. AlphaBeta(4) vs. Minimax(4)

Custom Boards (Extra Credit)

I was curious as to how **Minimax** search and **AlphaBeta** search performed on different size boards. Speficially, I wanted to test how **Minimax** search and **AlphaBeta** search on a smaller 4x4 **Board**, and a larger 8x8 **Board**.

Custom 4x4 Board

On a 4x4 **Board**, there is a much smaller branching factor. There are only 16 positions on a 4x4 **Board** as opposed to 36 positions on a 6x6 **Board**. **Minimax** search was able to run very quickly on a depth of 5, and **AlphaBeta** search was able to run quickly also at a depth of 7. The results for the **custom4x4.txt** maze is shown below.

custom4x4.txt

8	8	7	9
7	4	2	3
4	1	6	7
3	7	2	6

Matchup: Minimax(5) vs. Minimax(5)

```
*** blueScore = 52   greenScore = 32 ***

G     B     G     B
B     G     B     G
G     G     B     B
G     B     G     B

Blue Player: total nodes expanded = 992688
Blue Player: nodes expanded per move = 55149
Blue Player: Time per move = 0.2147222222222223 seconds

Green Player: total nodes expanded = 656720
Green Player: nodes expanded per move = 36484
Green Player: Time per move = 0.1373888888888889 seconds
```

Matchup: AlphaBeta(7) vs. AlphaBeta(7)

*** blueScore = 45   greenScore = 39 ***

G B B B
B B G B
G G G G
G B G G

Blue Player: total nodes expanded = 1037985
Blue Player: nodes expanded per move = 57665
Blue Player: Time per move = 0.7547222222222223 seconds

Green Player: total nodes expanded = 614490
Green Player: nodes expanded per move = 34138
Green Player: Time per move = 0.3972777777777778 seconds

Matchup: Minimax(5) vs. AlphaBeta(7)

*** blueScore = 60   greenScore = 24 ***

G B G B
B G B G
B B B B
B B G B

Blue Player: total nodes expanded = 992688
Blue Player: nodes expanded per move = 55149
Blue Player: Time per move = 0.21616666666666667 seconds

Green Player: total nodes expanded = 611242
Green Player: nodes expanded per move = 33957
Green Player: Time per move = 0.37955555555555553 seconds

Matchup: AlphaBeta(7) vs. Minimax(5)

*** blueScore = 52   greenScore = 32 ***

G B G B
B G B G
G G B B
G B G B

Blue Player: total nodes expanded = 1038506
Blue Player: nodes expanded per move = 57694
Blue Player: Time per move = 0.7288333333333334 seconds

Green Player: total nodes expanded = 656720
Green Player: nodes expanded per move = 36484
Green Player: Time per move = 0.1441111111111111 seconds

Custom 8x8 Board

On an 8x8 **Board**, there is a much larger branching factor. There are 64 positions on an 8x8 **Board** as opposed to 36 positions on a 8x8 **Board**. **Minimax** had to run at a slower depth of 2, and **AlphaBeta** search was able to run quickly at a depth of 3. The results for the **custom8x8.txt** maze is shown below.

As the size of the **Board** increases, the difficulty in playing this game optimally is higher for both human players and computer players. The fact that the computer player will have to search at a lower depth does not mean that it will do worse against a human player. The human player may also have trouble finding the optimal move to do since there are an increased number of possibilities.

custom8x8.txt

66	76	28	66	11	9	42	11
31	39	50	8	33	14	51	85
80	76	39	59	2	48	54	85
50	73	43	3	13	3	15	25
99	45	72	87	49	4	76	52
80	63	92	28	61	53	42	45
44	43	72	37	49	84	76	52
40	63	22	28	66	52	42	46

Matchup: Minimax(2) vs. Minimax(2)

*** blueScore = 1688 greenScore = 1364 ***

B	B	G	B	G	B	B	G
B	B	B	B	B	B	B	B
B	B	B	B	G	B	G	B
G	B	B	B	B	G	B	G
B	G	B	B	G	G	G	B
G	G	G	B	G	G	B	B
G	B	G	G	G	G	G	B
G	G	B	G	G	G	B	G

Blue Player: total nodes expanded = 45760
Blue Player: nodes expanded per move = 2542
Blue Player: Time per move = 0.04655555555555557 seconds

Green Player: total nodes expanded = 43680
Green Player: nodes expanded per move = 2426
Green Player: Time per move = 0.04233333333333335 seconds

Matchup: AlphaBeta(3) vs. AlphaBeta(3)

*** blueScore = 1662   greenScore = 1390 ***

B B G B G B B B
G B B G B B B B
B B B B B B G B
G B B B B G B G
B G B G G G G B
G B G G G G G G
B G B G B G G G
G B G B G G B G

Blue Player: total nodes expanded = 163183
Blue Player: nodes expanded per move = 9065
Blue Player: Time per move = 1.9447222222222227 seconds

Green Player: total nodes expanded = 132128
Green Player: nodes expanded per move = 7340
Green Player: Time per move = 1.8043888888888888 seconds

Matchup: Minimax(2) vs. AlphaBeta(3)

*** blueScore = 1479   greenScore = 1573 ***

B B B B B B B G
B B B B B B B B
B B G B G B G B
G B G G B G B G
B G B G G G G B
G G G G G G B G
G G G G G G G B
B G B G G G B G

Blue Player: total nodes expanded = 45760
Blue Player: nodes expanded per move = 2542
Blue Player: Time per move = 0.047388888888888904 seconds

Green Player: total nodes expanded = 144026
Green Player: nodes expanded per move = 8001
Green Player: Time per move = 1.819444444444444 seconds

Matchup: AlphaBeta(3) vs. Minimax(2)

*** blueScore = 1778   greenScore = 1274 ***

B B G B B B B G
G B B G B G B B
B B G B B B B B
G B B B B G B G
B G B B G G G B
G B G B G G G G
B G B G B G G G
G B G B G G B G

Blue Player: total nodes expanded = 164374
Blue Player: nodes expanded per move = 9131
Blue Player: Time per move = 1.7036111111111114 seconds

Green Player: total nodes expanded = 43680
Green Player: nodes expanded per move = 2426
Green Player: Time per move = 0.038722222222222234 seconds

Custom 16x16 Board

These results tell us something. We notice that for larger **Boards**, our A.I. will still be able to play them and get a result. A **custom16x16.txt** **Board** was created and played by 2 computer players. We can keep creating larger **Boards** and playing on them, but the depth of the search may need to decrease. The results are shown for:

Matchup: AlphaBeta(2) vs. AlphaBeta(2).

```
*** blueScore = 5890   greenScore = 6318 ***
66      76      28      66      11      9       42      11      66      76      28      66      11      9
31      39      50      8       33      14      51      85      31      39      50      8       33      14
80      76      39      59      2       48      54      85      80      76      39      59      2       48
50      73      43      3       13      3       15      25      50      73      43      3       13      3
99      45      72      87      49      4       76      52      99      45      72      87      49      4
80      63      92      28      61      53      42      45      80      63      92      28      61      53
44      43      72      37      49      84      76      52      44      43      72      37      49      84
40      63      22      28      66      52      42      46      40      63      22      28      66      52
66      76      28      66      11      9       42      11      66      76      28      66      11      9
31      39      50      8       33      14      51      85      31      39      50      8       33      14
80      76      39      59      2       48      54      85      80      76      39      59      2       48
50      73      43      3       13      3       15      25      50      73      43      3       13      3
99      45      72      87      49      4       76      52      99      45      72      87      49      4
80      63      92      28      61      53      42      45      80      63      92      28      61      53
44      43      72      37      49      84      76      52      44      43      72      37      49      84
40      63      22      28      66      52      42      46      40      63      22      28      66      52

B       G       B       G       G       G       B       B       B       G       G       G       G       G
B       B       G       G       G       B       G       B       G       G       G       B       G       G
B       G       G       G       G       G       B       B       G       B       G       G       B       G
B       B       B       G       B       B       G       G       B       G       G       G       G       G
B       B       B       B       B       G       B       B       G       B       G       G       B       G
B       G       B       B       B       B       B       G       G       G       G       G       G       B
B       G       G       B       G       B       B       G       B       G       G       G       G       G
G       G       G       G       B       G       G       B       G       G       G       G       G       B
B       G       G       G       G       G       B       B       B       G       B       G       G       G
G       B       G       G       G       G       G       B       B       G       G       B       G       B
B       G       G       G       G       G       B       B       B       B       G       G       G       G
B       B       B       G       B       B       G       G       B       G       G       G       G       B
B       G       B       B       B       B       B       B       G       G       G       G       B       G
B       B       B       G       B       B       B       G       G       G       G       B       G       B
B       B       B       B       B       B       B       G       G       B       G       G       G       G
B       B       B       B       B       B       G       G       B       G       B       G       G       B

Blue Player: total nodes expanded = 91208
Blue Player: nodes expanded per move = 5067
Blue Player: Time per move = 12.541888888888892 seconds

Green Player: total nodes expanded = 78172
Green Player: nodes expanded per move = 4342
Green Player: Time per move = 12.320166666666678 seconds
```

2.2 Extended Rules (Extra Credit)

Extended Rule: Battle

New Evaluation Function: To account for both the scores of each player, and the unit strength (which will highly determine who wins Battles), we will use the following weighted evaluation function (for inner nodes):

```
(blueScore - greenScore) + 3 * (blueUnitStrength - greenUnitStrength)
```

Game strategy: The above evaluation function will favor choosing high-resource spots more heavily than just attempting to get a maximum score. This strategy helps gain points near the endgame since capturing spots becomes easier with a high unit strength. A consequence of this strategy is that you may lose utility for capturing low resource spots, which is exactly what is intended since the goal of the evaluation function is to maintain a high "unit strength"

Clustering of results: Clustering happens the most in the **Smolensk.txt** **Board**. The most scattering happens in the **Westerplatte.txt** **Board**. This leads us to conclude that moderate clustering occurs for the Battle rule.

Results are shown below for 3 select **Boards**.

Sevastopol.txt

1	1	1	1	1	1
2	2	2	2	2	2
4	4	4	4	4	4
8	8	8	8	8	8
16	16	16	16	16	16
32	32	32	32	32	32

Matchup: AlphaBeta(4) vs. AlphaBeta(4)

```
*** blueScore = 191   greenScore = 187 ***

B     B     B     G     G     G
B     G     B     G     G     G
B     B     G     B     G     B
G     B     B     G     G     B
B     B     G     G     G     B
B     G     B     G     B     G

Blue Player: total nodes expanded = 67292
Blue Player: nodes expanded per move = 3738
Blue Player: Time per move = 0.29966666666666675 seconds

Green Player: total nodes expanded = 57831
Green Player: nodes expanded per move = 3212
Green Player: Time per move = 0.1904444444444444 seconds
```

Smolensk.txt

66	76	28	66	11	9
31	39	50	8	33	14
80	76	39	59	2	48
50	73	43	3	13	3
99	45	72	87	49	4
80	63	92	28	61	53

Matchup: AlphaBeta(4) vs. AlphaBeta(4)

*** blueScore = 859   greenScore = 794 ***

B	B	G	G	G	G
B	B	B	G	G	G
B	B	B	G	G	G
G	B	B	G	B	G
B	B	B	G	G	B
G	G	G	G	G	B

Blue Player: total nodes expanded = 190286
Blue Player: nodes expanded per move = 10571
Blue Player: Time per move = 0.6804444444444445 seconds

Green Player: total nodes expanded = 122241
Green Player: nodes expanded per move = 6791
Green Player: Time per move = 0.5117777777777779 seconds

Westerplatte.txt

1	1	1	1	1	1
1	3	4	4	3	1
1	4	2	2	4	1
1	4	2	2	4	1
1	3	4	4	3	1
1	1	1	1	1	1

*** blueScore = 43   greenScore = 29 ***

B	B	G	B	G	B
B	B	B	G	B	B
G	B	G	G	B	G
B	B	G	G	G	B
G	B	G	B	B	G
B	G	G	B	B	G

Blue Player: total nodes expanded = 153410
Blue Player: nodes expanded per move = 8522
Blue Player: Time per move = 0.5822222222222222 seconds

Green Player: total nodes expanded = 112793
Green Player: nodes expanded per move = 6266
Green Player: Time per move = 0.37561111111111123 seconds

Extended Rule: Duel

New Evaluation Function: To account for how likely a player is going to win a duel, using a simple evaluation function that takes into account each player's unit strength works surprisingly well:

blueUnitStrength - greenUnitStrength

Game strategy: The above evaluation function will favor choosing high-resource spots more heavily than just attempting to get a maximum score. In fact, it will purposely not give any weight to a player's score. This will maximize unit strength instead of score, which will help a player win Duels near the endgame. This evaluation function makes sure we don't collect low-resource spots.

Results are shown below for 3 select **Boards**.

Clustering of results: Very minimal clustering of same color pieces happens in the results. Tiling of pieces happens in **Sevastopol.txt**. However, lots of clustering happens in the **Westerplatte.txt** **Board**.

Results are shown below for 3 select **Boards**.

Sevastopol.txt

1	1	1	1	1	1
2	2	2	2	2	2
4	4	4	4	4	4
8	8	8	8	8	8
16	16	16	16	16	16
32	32	32	32	32	32

*** blueScore = 189 greenScore = 189 ***

G	B	G	B	B	G
B	G	B	G	B	G
G	B	G	B	G	B
B	G	B	G	B	G
G	B	G	B	G	B
B	G	B	G	B	G

Blue Player: total nodes expanded = 38821
Blue Player: nodes expanded per move = 2156
Blue Player: Time per move = 0.14016666666666663 seconds

Green Player: total nodes expanded = 32283
Green Player: nodes expanded per move = 1793
Green Player: Time per move = 0.12122222222222225 seconds

Smolensk.txt

66	76	28	66	11	9
31	39	50	8	33	14
80	76	39	59	2	48
50	73	43	3	13	3
99	45	72	87	49	4
80	63	92	28	61	53

*** blueScore = 848 greenScore = 805 ***

G	B	B	G	B	G
B	B	B	G	B	B
B	B	G	G	G	B
G	G	G	B	G	G
B	G	G	B	B	G
G	B	G	G	B	G

Blue Player: total nodes expanded = 106718
Blue Player: nodes expanded per move = 5928
Blue Player: Time per move = 0.26994444444444454 seconds

Green Player: total nodes expanded = 192062
Green Player: nodes expanded per move = 10670
Green Player: Time per move = 0.4326666666666666 seconds

Westerplatte.txt

1	1	1	1	1	1
1	3	4	4	3	1
1	4	2	2	4	1
1	4	2	2	4	1
1	3	4	4	3	1
1	1	1	1	1	1

*** blueScore = 36 greenScore = 36 ***

B	B	G	B	G	G
B	B	B	G	G	G
B	B	B	G	G	G
B	B	G	B	G	G
B	B	B	G	G	G
B	B	G	B	G	G

Blue Player: total nodes expanded = 73304
Blue Player: nodes expanded per move = 4072
Blue Player: Time per move = 0.21777777777777774 seconds

Green Player: total nodes expanded = 73288
Green Player: nodes expanded per move = 4071
Green Player: Time per move = 0.20433333333333334 seconds

Extended Rule: Attrition

Implementing Attrition: The main change that was necessary was to change "utilities" to be "floats" instead "ints". We decrease each resource by 10% each turn. It was also necessary to calculate scores to account that losing resources doesn't mean losing points. In this implementation, when a player steals a resource spot, the player only gets the resources of that spot with the amount of attrition that was applied to the spot at the time of capture. That means the final scores will not sum up to the total resources available at the beginning of the game.

New Evaluation Function: To account for the quickly decreasing resource values, we weight the score differential by a factor 3 to try increase points as fast as possible. We also take into account the "unit strength" differential so that we can try to get only high resource spots.

```
3 * (blueScore - greenScore) + (blueUnitStrength - greenUnitStrength)
```

Game strategy: The above evaluation function will aim to gather as many spots with a high score as possible. There is only a slight weighting to gather spots that will increase unit strength.

Clustering of results: A very high amount of clustering of similar color pieces occurs in the results for all the **Boards**

Results are shown below for 3 select **Boards**. This is for playing with "Battle" rules and "Attrtition"

Sevastopol.txt

1	1	1	1	1	1
2	2	2	2	2	2
4	4	4	4	4	4
8	8	8	8	8	8
16	16	16	16	16	16
32	32	32	32	32	32

```
*** blueScore = 164.33046   greenScore = 213.66956 ***

B     G     G     G     G     B
B     G     G     G     G     B
B     B     G     G     G     B
G     B     G     G     G     B
B     B     B     G     G     G
B     G     B     G     G     G

Blue Player: total nodes expanded = 151930
Blue Player: nodes expanded per move = 8440
Blue Player: Time per move = 0.7283888888888889 seconds

Green Player: total nodes expanded = 121450
Green Player: nodes expanded per move = 6747
Green Player: Time per move = 0.4579444444444445 seconds
```

Smolensk.txt

66	76	28	66	11	9
31	39	50	8	33	14
80	76	39	59	2	48
50	73	43	3	13	3
99	45	72	87	49	4
80	63	92	28	61	53

*** blueScore = 760.7892   greenScore = 892.2108 ***

B	B	B	G	G	B
B	B	B	G	G	G
G	G	G	G	G	G
G	G	G	G	G	G
B	B	G	G	B	B
B	B	G	B	G	B

Blue Player: total nodes expanded = 209432
Blue Player: nodes expanded per move = 11635
Blue Player: Time per move = 0.9967777777777778 seconds

Green Player: total nodes expanded = 135191
Green Player: nodes expanded per move = 7510
Green Player: Time per move = 0.5324444444444443 seconds

Westerplatte.txt

1	1	1	1	1	1
1	3	4	4	3	1
1	4	2	2	4	1
1	4	2	2	4	1
1	3	4	4	3	1
1	1	1	1	1	1

*** blueScore = 32.494656   greenScore = 39.505344 ***

B	B	B	G	G	G
B	B	B	G	G	G
B	B	B	B	G	G
B	B	G	G	G	G
B	B	B	G	G	G
G	B	B	G	G	G

Blue Player: total nodes expanded = 393337
Blue Player: nodes expanded per move = 21852
Blue Player: Time per move = 2.332 seconds

Green Player: total nodes expanded = 244759
Green Player: nodes expanded per move = 13597
Green Player: Time per move = 1.4201111111111113 seconds