

لینک پروژه در گیتاب:

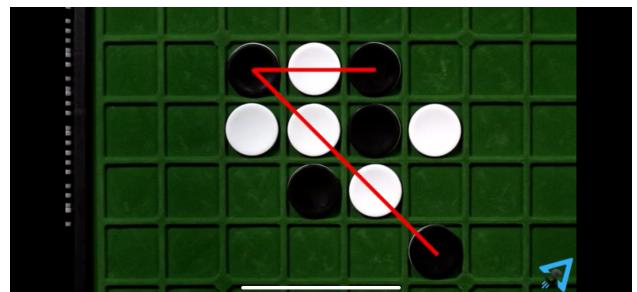
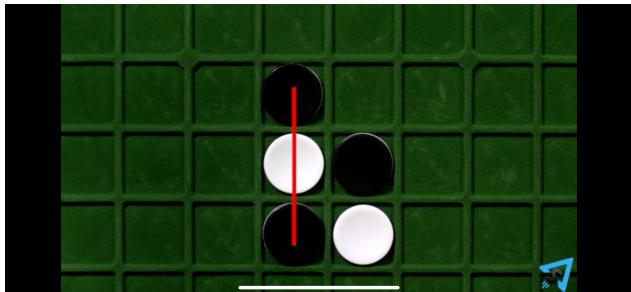
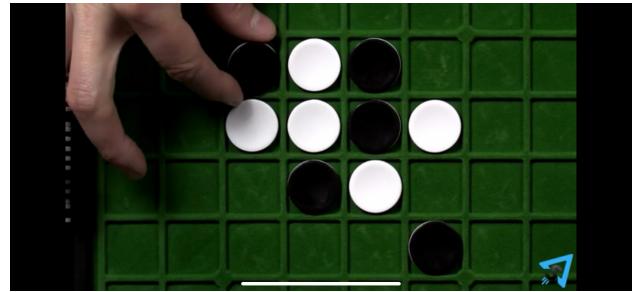
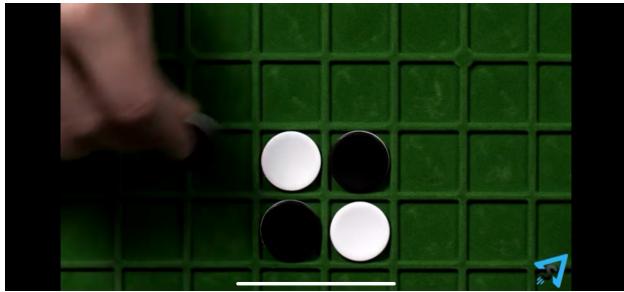
<https://github.com/nikashahabi/othello-game-with-adversarial-search>

توضیح بازی اتلو:

اتلو یک بازی دو نفره است که قصد هر بازیکن این است که در انتهای بازی بیشترین تعداد دیسک رنگ خود را روی صفحه داشته باشد. صفحه ای اتلو یک صفحه شطرنجی 8^*8 است که در ابتدا دیسک های دو بازیکن به این صورت روی صفحه قرار میگیرند:

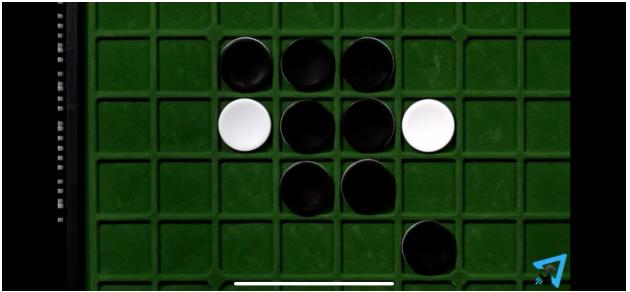


حال یک بازیکن با مهره های سفید بازی میکند و دیگری با مشکی. هر بازیکن هر زمانی که حرکتی بخواهد انجام دهد اجازه دارد دیسکی بردارد. بازیکنی که دیسک های مشکی را دارد شروع میکند و نوبت ها پس از آن یکی در میان عوض میشود. در هر نوبت بازی بازیکنی که نوبتش است باید دیسک رنگ خود را در یکی از خانه های بازی بگرد به کونه ای که در سطر، ستون و یا قطری که دیسک را قرار میدهد با قرارگیری دیسک در آن مرحله بین دیسکی که قرار داده شده و دیسک هم رنگ آن که روی صفحه قبل قرار داده شده بود تعدادی مهره با رنگ مخالف وجود داشته باشد. مثال هایی از حرکت های مناسب را میبینیم:



اگر در هیچ حرکتی این اتفاق نیوفتد ما مجاز به بازی نیستیم و نوبت نفر دیگر میشود.

اگر ما مجاز به بازی باشیم باید بازی کنیم.
برگردانده شدن دیسک های طرف مقابل فقط باعث نتیجه می
مستقیمی از حرکت ما باشد.
همه می دیسک های وسط باید



برگردانده شوند.
اگر هر دو طرف نتوانند هیچ حرکتی انجام دهند بازی تمام است.
امتیاز هر شخص در انتهای بازی = تعداد دیسک های طرف مقابل در صفحه - تعداد دیسک های رنگ خودش

فرموله سازی مسئله به صورت یک مسئلهی Search

S_0 استیت اولیه، صفحه‌ی اتلوا با ۴ مهره‌ی چیده شده در وسط

PLAYER(s) : بازیکنی که در استیت مورد نظر نویش است.

ACTIONS(s, player) : لیست حرکاتی که یک بازیکن در یک استیت خاص میتواند انجام دهد.

RESULT(s,a) نتیجه‌ی حرکت a روی استیت s

TERMINAL-TEST(s) اگر استیت انتیت پایانی باشد true برابر میگردد.

UTILITY(s, p) : برای بازیکن P امتیاز در استیتنهایی S چقدر است.

این بازی یک بازی Zero-sum game است.

توضیح صورت مسئله:

با استفاده از agent ای طراحی کنید که اتلوا بازی کند.

چرا این الگوریتم و تحلیل آن به طور کلی:

با توجه به اینکه این بازی یک بازی تخاصمی است الگوریتم های جست و جوی تخاصمی روی آن جواب میدهند.

در این پژوهه

Min-max algorithm with alpha beta pruning with use of evaluation function for a specific cut-off

پیاده سازی شده است.

الگوریتم min-max از نظر تئوری اهمیت زیادی دارد. اما از نظر زمانی اصلاً الگوریتم خوبی نیست. این الگوریتم یک سرچ Depth- first روی درخت بازی میزند. اگر ماکسیمم عمق درخت M و b ، Branching factor باشد، زمان این الگوریتم از $O(b^m)$ است. چون در الگوریتم پیاده سازی شده همه‌ی Action ها با هم در یک لیست تولید میشوند مکان این الگوریتم از Branching factor $O(m)$ است.

برای بازی اتلوا حدوداً ۱۰ است. (منبع زیر) به این معنا که هر استیت در d از درخت از استیتی در عمق ۱-d در درخت ۱۰ برابر بیشتر زمان میبرد.

https://en.wikipedia.org/wiki/Game_complexity

عمق درخت هم ($O(\text{Dimension}^2)$) است.

با استفاده از alpha-beta pruning با اینکه $\text{order of time complexity}$ عوض نمیشود اما زمان در بهترین حالت نصف میشود ($b_{\text{new}} = \sqrt{b}$) اگر بچه ها رندوم بررسی شوند زمان میشود $b^{3m/4}$ با این حال باز هم انتو با صفحه i^8 زمان بسیار زیادی از ما میگیرد. با گذشتن یک cut-off و استفاده از یک evaluation function برای یک node به جای ادامه ای تمام مسیر تا برگ زمان بسیار مورد قبول تر میشود. در ادامه در مورد cut-off گشته شده و استفاده evaluation function توضیح داده میشود.

توضیح الگوریتم:

در این پروژه برنامه در مقابل خودش بازی میکند و هر دو agent از الگوریتم نام برده شده استفاده میکنند. هر agent که نوبتش شد و امکان بازی دارد max میشود و برای انتخاب حرکت آن و خانه ای که باید مهره را در آن قرار دهد

```
minMaxDecisionWithPruning(board, player)
```

صدای زده میشود که خانه i مورد نظر را برミگرداند. حرکت انتخاب شده باتابع

```
makeMove(move, board, player)
```

اجرا میشود. سپس بازیکن بعدی انتخاب میشود و همین الگوریتم برایش اجرا میشود. اگر هیچ بازیکنی امکان حرکت نداشت تابع انتخاب بازیکن بعدی None برミگرداند تا بازی تمام شود و امتیازات اعلام شوند.

```
minMaxDecisionWithPruning(board, player)
```

از دو تابع

```
maxValueWithPruning(board, player, alphabeta)
```

```
minValueWithPruning(board, player, alphabeta)
```

استفاده میکند. به طوری که حرکتی انتخاب میشود که ماکسیمم ارزش را بین حرکت های تولید شده که از نوع min node هستند داشته باشد. Min node ها ارزشی برابر با مینیمم نود های فرزند خود که max node هستند دارند و در مقابل node ها ارزشی برابر با ماکسیمم نودهای فرزند خود که min node هستند دارند. ارزش یک نود terminal همان امتیاز برای max است. ولی در این پروژه از evaluation function ای استفاده میشود که ارزش نود های غیرترمینالی را با شرط برقرار بودن یک cut off به طور مستقیم برミگرداند.

ایده ای این evaluation function از منابع زیر پیدا شد:

<https://stackoverflow.com/questions/12334216/othello-evaluation-function>

https://pats.cs.cf.ac.uk/@archive_file?p=708&n=final&f=1-Final_Report.pdf&SIG=48a08fc35e0169234716d776bec6f534c2e97babba9edc889d4c99291178a38

https://www.researchgate.net/publication/234785722_Othello_evaluation_function_comparison

ولی از evaluation function توضیح داده شده در لینک زیر به طور کاملا مشابه استفاده شد:

<http://www.cs.cornell.edu/~yuli/othello/othello.html>

(استفاده از تابع Eval و نه از کد پروژه ها)

در همه لینک ها به سه فاکتور برای محاسبه ارزش یک استیت برای یک بازیکن اشاره میشود و من از وزن های استفاده شده در پروژه اخیر برای وصل کردن این سه فاکتور به هم استفاده کردیم:

فاکتور اول: mobility : تعداد حرکت هایی که بازیکن میتواند در یک استیت انجام دهد. هر چه این عدد بزرگتر باشد به معنی باز بودن دست ما برای انجام حرکت بهتر و کوچک تر بودن این عدد برای حریف به معنی مجبور شدن او برای انجام حرکت نامناسب است. وزن هر حرکت یک گرفته شد.

فاکتور دوم: تعداد مهره های موجود در زمین بازی که متعلق به بازیکن است. امتیازدهی پایان بازی این شهود را به ما میدهد که از این فاکتور استفاده کنیم، اما تجربه نشان داده که این فاکتور اصلاً فاکتور مناسبی نیست یا حداقل وزن آن باید کم باشد. چون تعداد مهره ها خیلی سریع میتواند عوض شود و ورق برگردد. وزن هر مهره ۱۰۰۰۰۰ در نظر گرفته شده.

فاکتور سوم: داشتن موقعیت های مهم در بازی که در اینجا تعداد مهره های بازیکن در ۴ گوشه زمین فرض شده است. این موقعیت از این جهت اهمیت دارد که هیچ وقت توسط بازیکن حریف تصاحب نمیشود و تعدادی ازفلیپ کردن ها را از بین میبرد و حرکت های حریف را محدود میکند. وزن هر بازیکن در گوشه برای بازیکن ۱۰ در نظر گرفته شد.

این سه فاکتور با وزن های گفته شده جمع زده میشوند و تابع Eval را تشکیل میدهند.

Cut-off پیشنهاد داده شده در پروژه نام برد شده ۱۴ بود. یعنی زمانی که نزدیک به پایان بازی بود و ۱۴ یا کمتر خانه ۱۴ را

در زمین قرار دارد از این تابع به جای محاسبه مستقیم استفاده شود. ولی ران کردن با Cut-off کمتر از ۵۰ هم برای من زمان بر بود. من از Cut-off ، ۵۰ استفاده کردم و برنامه ای اجرا گرفته شده هم با همین عدد است.

علاوه بر این ها در این پروژه از Alpha-beta pruning هم استفاده شد که مینیمم و مаксیمم ارزش های پیدا شده ذخیره و اپدیت میشوند که کرانی برای ارزش نود ها پیدا شود و در صورت امکان نود هایی به طور کلی بررسی نشوند ولی تقاضتی در جواب داده شده ایجاد نشود.

توضیح توابع مسئله:

به صورت کامنت در بین کد

تحلیل زمانی و مکانی الگوریتم:

در قسمت چرا این الگوریتم و تحلیل آن الگوریتم pruning و استفاده از min max تحلیل شد.

b^d nodes

Number of terminal nodes $O(b^d)$

Computation of utility of each terminal node: $O(\text{dimension}^2)$

$\rightarrow O(\text{dimension}^2 * b^d)$

وقتی از heuristic eval function استفاده میکنیم زمان:

$b^{d_{\text{new}}}$ nodes

Number of Terminal nodes $O(b^{d_{\text{new}}})$

Computation of evaluation of each node that is bellow a cut-off : $O(\text{dimension}^2)$

$\rightarrow O(\text{dimension}^2 * \text{bdnew})$

$D_{\text{new}} = O(\text{dimension}^2 - \text{cut-off})$

مکان در هر صورت چون همه move‌های بعدی با هم تولید می‌شوند از $O(m)$ است.

نمونه‌ی خروجی و ورودی مربوطه:

Cut-off = 50

Input : playOthello(8)

Output :

کپی شده است:

Users/nayvaka/Documents/University/Term6/AI/AIProjects/Othello/venv/bin//
python /Users/nayvaka/Documents/University/Term6/AI/AIProjects/Othello/
main.py

:initial board

```
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 2 1 0 0 0  
0 0 0 1 2 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0
```

turn = Black

the decision is made. to a node with value : 3.07

the selected move is = (2, 4)

board after the selected move

```
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 1 0 0 0 0  
0 0 0 1 1 0 0 0  
0 0 0 1 2 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0
```

turn = White

the decision is made. to a node with value : 14.05999999999999

the selected move is = (2, 5)

board after the selected move

```
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0  
0 0 2 1 0 0 0 0  
0 0 0 2 1 0 0 0  
0 0 0 1 2 0 0 0  
0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0
```

0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 2.07

the selected move is = (2, 6)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 1 1 0 0 0 0
0 0 0 2 1 0 0 0
0 0 0 1 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 10.06

the selected move is = (4, 5)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 1 1 0 0 0 0
0 0 0 2 1 0 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 3.07

the selected move is = (3, 5)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 1 1 0 0 0 0
0 0 1 1 1 0 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 17.06

the selected move is = (2, 2)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 1 1 0 2 0 0
0 0 1 1 2 0 0 0

0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 6.07

the selected move is = (3, 2)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 1 1 0 2 0 0
0 0 1 1 1 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 18.060000000000002

the selected move is = (2, 3)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 1 1 2 2 0 0
0 0 1 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 8.06

the selected move is = (1, 2)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0
0 1 1 1 2 1 0 0
0 0 1 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 11.07

the selected move is = (1, 1)

board after the selected move

0 0 0 0 0 0 0 0

0 0 0 0 0 1 2 0
0 1 1 1 2 2 0 0
0 0 1 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 9.07

the selected move is = (2, 1)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 0 1 2 0
0 1 1 1 1 1 1 0
0 0 1 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 9.07

the selected move is = (1, 3)

board after the selected move

0 0 0 0 0 0 0 0
0 0 0 0 2 2 2 0
0 1 1 1 2 1 1 0
0 0 1 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 20.08

the selected move is = (0, 0)

board after the selected move

0 0 0 0 0 0 0 1
0 0 0 0 2 2 1 0
0 1 1 1 2 1 1 0
0 0 1 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 13.09

the selected move is = (1, 5)
board after the selected move

0	0	0	0	0	0	0	1
0	0	2	0	2	2	1	0
0	1	2	2	2	1	1	0
0	0	2	2	2	1	0	0
0	0	2	2	2	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

turn = Black

the decision is made. to a node with value : 18.07

the selected move is = (0, 2)
board after the selected move

0	0	0	0	0	1	0	1
0	0	2	0	2	1	1	0
0	1	2	2	2	1	1	0
0	0	2	2	2	1	0	0
0	0	2	2	2	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

turn = White

the decision is made. to a node with value : 8.12

the selected move is = (0, 1)
board after the selected move

0	0	0	0	0	1	2	1
0	0	2	0	2	2	1	0
0	1	2	2	2	1	1	0
0	0	2	2	2	1	0	0
0	0	2	2	2	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

turn = Black

the decision is made. to a node with value : 18.08

the selected move is = (1, 4)
board after the selected move

0	0	0	0	0	1	2	1
0	0	2	1	1	1	1	0
0	1	2	2	1	1	1	0
0	0	2	2	2	1	0	0
0	0	2	2	2	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 8.11

the selected move is = (0, 3)

board after the selected move

0 0 0 0 2 2 2 1
0 0 2 2 2 1 1 0
0 1 2 2 2 1 1 0
0 0 2 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 18.08

the selected move is = (1, 6)

board after the selected move

0 0 0 0 2 2 2 1
0 1 1 1 1 1 1 0
0 1 2 2 2 1 1 0
0 0 2 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 21.130000000000003

the selected move is = (0, 7)

board after the selected move

2 0 0 0 2 2 2 1
0 2 1 1 1 1 1 0
0 1 2 2 2 1 1 0
0 0 2 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 19.11

the selected move is = (0, 4)

board after the selected move

2 0 0 1 1 1 1 1
0 2 1 1 1 1 1 0
0 1 2 2 2 1 1 0
0 0 2 2 2 1 0 0

0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 18.119999999999997

the selected move is = (0, 5)

board after the selected move

2 0 2 1 1 1 1 1
0 2 2 2 1 1 1 0
0 1 2 2 2 1 1 0
0 0 2 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 18.130000000000003

the selected move is = (0, 6)

board after the selected move

2 1 1 1 1 1 1 1
0 1 2 2 1 1 1 0
0 1 2 2 2 1 1 0
0 0 2 2 2 1 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 16.13

the selected move is = (1, 0)

board after the selected move

2 1 1 1 1 1 1 1
0 1 2 2 2 2 2 2
0 1 2 2 2 1 2 0
0 0 2 2 2 2 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 21.11

the selected move is = (2, 0)

board after the selected move

2 1 1 1 1 1 1 1

0 1 2 2 2 2 1 1
0 1 2 2 2 1 1 1
0 0 2 2 2 2 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 14.16

the selected move is = (1, 7)

board after the selected move

2 1 1 1 1 1 1 1
2 2 2 2 2 2 1 1
0 2 2 2 2 1 1 1
0 0 2 2 2 2 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 20.130000000000003

the selected move is = (3, 6)

board after the selected move

2 1 1 1 1 1 1 1
2 1 2 1 2 2 1 1
0 1 1 2 2 1 1 1
0 1 2 2 2 2 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 13.15

the selected move is = (2, 7)

board after the selected move

2 1 1 1 1 1 1 1
2 1 2 1 2 2 1 1
2 2 2 2 2 1 1 1
0 2 2 2 2 2 0 0
0 0 2 2 2 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 20.15

the selected move is = (3, 7)
board after the selected move

2 1 1 1 1 1 1 1

turn = White

the decision is made, to a node with value : 12.17

The selection is **initially** set so
the selected move is = (3, 0)

the selected move is $(3, 3)$
board after the selected move

turn = Black

the decision is made to a node with value : 20 17

the selected move is = (3, 1)

the selected move is – (3, 1)
board after the selected move

turn = White

the decision is made to a node with value : 14 1299999999999999

the selected move is = (1, 0)

the selected move is = (4, 0)
board after the selected move

0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 18.22

the selected move is = (4, 2)

board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	2	2	1	1	
1	1	1	1	1	1	1	2	
0	0	2	2	2	1	0	2	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	

turn = White

the decision is made. to a node with value : 14.15

the selected move is = (4, 1)

board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	2	2	1	1	
1	1	1	1	1	2	1	2	
0	0	2	2	2	2	2	2	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	

turn = Black

the decision is made. to a node with value : 16.22

the selected move is = (5, 0)

board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
1	1	1	1	1	1	1	1	
0	0	2	2	2	2	1	1	
0	0	0	0	0	0	0	1	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	

turn = White

the decision is made. to a node with value : 11.129999999999999

the selected move is = (4, 6)

board after the selected move

2	1	1	1	1	1	1	1	
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
1	1	2	1	1	1	1	1	

0 2 2 2 2 2 1 1
0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 17.27

the selected move is = (4, 7)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 2 1 2 1 1
1 1 2 1 1 1 1 1
1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 16.1

the selected move is = (5, 1)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 2 1 2 1 1
1 1 2 1 2 1 1 1
1 1 1 1 1 2 1 1
0 0 0 0 0 0 2 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 14.30999999999999

the selected move is = (5, 2)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 2 1 2 1 1
1 1 2 1 2 1 1 1
1 1 1 1 1 1 1 1
0 0 0 0 0 1 1 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 17.11

the selected move is = (5, 3)

board after the selected move

2 1 1 1 1 1 1 1

2 1 1 1 2 2 1 1
2 1 2 2 1 2 1 1
1 1 2 1 2 1 1 1
1 1 1 2 2 1 1 1
0 0 0 0 2 1 1 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 14.32

the selected move is = (5, 4)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 2 1 2 1 1
1 1 2 1 2 1 1 1
1 1 1 1 1 1 1 1
0 0 0 1 1 1 1 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 17.11

the selected move is = (5, 5)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 2 1 2 1 1
1 1 2 1 2 1 1 1
1 1 2 2 1 1 1 1
0 0 2 1 1 1 1 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = Black

the decision is made. to a node with value : 13.34

the selected move is = (5, 6)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 2 1 2 1 1
1 1 2 1 2 1 1 1
1 1 1 2 1 1 1 1
0 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

turn = White

the decision is made. to a node with value : 18.119999999999997

the selected move is = (5, 7)
board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
2	1	2	1	2	1	1	1	
2	2	1	2	1	1	1	1	
2	1	1	1	1	1	1	1	
0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	

turn = White

the decision is made. to a node with value : 17.16

the selected move is = (6, 0)
board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
2	1	2	1	2	1	1	1	
2	2	1	2	1	2	1	1	
2	1	1	1	1	1	2	1	
0	0	0	0	0	0	0	2	
0	0	0	0	0	0	0	0	

turn = Black

the decision is made. to a node with value : 22.32

the selected move is = (7, 0)
board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
2	1	2	1	2	1	1	1	
2	2	1	2	1	2	1	1	
2	1	1	1	1	1	2	1	
0	0	0	0	0	0	0	1	
0	0	0	0	0	0	0	1	

turn = White

the decision is made. to a node with value : 16.18

the selected move is = (6, 1)
board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
2	1	2	2	2	1	1	1	
2	2	1	2	2	2	1	1	
2	1	1	1	1	2	2	1	
0	0	0	0	0	0	2	1	

0 0 0 0 0 0 0 1

turn = Black

the decision is made. to a node with value : 23.31

the selected move is = (6, 2)

board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
2	1	2	2	2	1	1	1	
2	2	1	2	2	1	1	1	
2	1	1	1	1	1	1	1	
0	0	0	0	0	1	1	1	
0	0	0	0	0	0	0	1	

turn = White

the decision is made. to a node with value : 16.18

the selected move is = (6, 3)

board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	2	1	2	1	1	
2	2	2	2	2	1	1	1	
2	2	2	2	2	1	1	1	
2	1	1	2	2	1	1	1	
0	0	0	0	2	1	1	1	
0	0	0	0	0	0	0	1	

turn = Black

the decision is made. to a node with value : 23.32

the selected move is = (6, 4)

board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	1	1	2	1	1	
2	2	2	1	2	1	1	1	
2	2	2	1	2	1	1	1	
2	1	1	1	1	1	1	1	
0	0	0	1	1	1	1	1	
0	0	0	0	0	0	0	1	

turn = White

the decision is made. to a node with value : 15.17

the selected move is = (6, 5)

board after the selected move

2	1	1	1	1	1	1	1	
2	1	1	1	2	2	1	1	
2	1	2	1	1	2	1	1	
2	2	2	1	2	1	1	1	

2 2 2 1 2 1 1 1
2 2 2 2 1 1 1 1
0 0 2 1 1 1 1 1
0 0 0 0 0 0 0 1

turn = Black

the decision is made. to a node with value : 23.36

the selected move is = (6, 6)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 1 1 2 1 1
2 1 2 1 2 1 1 1
2 1 2 1 2 1 1 1
2 1 1 2 1 1 1 1
0 1 1 1 1 1 1 1
0 0 0 0 0 0 0 1

turn = White

the decision is made. to a node with value : 27.16

the selected move is = (7, 7)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 1 1 2 1 1
2 1 2 1 2 1 1 1
2 1 2 2 2 1 1 1
2 1 2 2 1 1 1 1
0 2 1 1 1 1 1 1
2 0 0 0 0 0 0 1

turn = Black

the decision is made. to a node with value : 21.39

the selected move is = (6, 7)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 1 1 2 1 1
2 1 2 1 2 1 1 1
2 1 2 2 2 1 1 1
2 1 2 2 1 1 1 1
1 1 1 1 1 1 1 1
2 0 0 0 0 0 0 1

turn = White

the decision is made. to a node with value : 25.19

the selected move is = (7, 1)

board after the selected move

2 1 1 1 1 1 1 1

2 1 1 1 2 2 1 1
2 1 2 1 1 2 1 1
2 1 2 1 2 1 1 1
2 1 2 2 2 1 1 1
2 1 2 2 2 1 1 1
1 1 1 1 1 2 1 1
2 0 0 0 0 0 2 1

turn = Black

the decision is made. to a node with value : 21.39

the selected move is = (7, 2)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 1 1 2 1 1
2 1 2 1 2 1 1 1
2 1 2 2 2 1 1 1
2 1 2 2 2 1 1 1
1 1 1 1 1 1 1 1
2 0 0 0 0 1 1 1

turn = White

the decision is made. to a node with value : 23.2

the selected move is = (7, 3)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 1 1 2 1 1
2 1 2 1 2 1 1 1
2 1 2 2 2 1 1 1
2 1 2 2 2 1 1 1
1 1 1 2 2 1 1 1
2 0 0 0 2 1 1 1

turn = Black

the decision is made. to a node with value : 21.4

the selected move is = (7, 4)

board after the selected move

2 1 1 1 1 1 1 1
2 1 1 1 2 2 1 1
2 1 2 1 1 2 1 1
2 1 2 1 2 1 1 1
2 1 2 1 2 1 1 1
2 1 2 1 2 1 1 1
1 1 1 1 1 1 1 1
2 0 0 1 1 1 1 1

turn = White

the decision is made. to a node with value : 21.18

the selected move is = (7, 5)
board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	1	1	2	1	1	
2	1	2	1	2	1	1	1	
2	1	2	1	2	1	1	1	
2	1	2	1	2	1	1	1	
1	2	2	2	1	1	1	1	
2	0	2	1	1	1	1	1	

turn = Black

the decision is made. to a node with value : 20.43

the selected move is = (7, 6)
board after the selected move

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	1	1	2	1	1	
2	1	2	1	2	1	1	1	
2	1	2	1	2	1	1	1	
2	1	2	1	2	1	1	1	
1	1	1	2	1	1	1	1	
2	1	1	1	1	1	1	1	

.no legal moves available

2	1	1	1	1	1	1	1	1
2	1	1	1	2	2	1	1	
2	1	2	1	1	2	1	1	
2	1	2	1	2	1	1	1	
2	1	2	1	2	1	1	1	
2	1	2	1	2	1	1	1	
1	1	1	2	1	1	1	1	
2	1	1	1	1	1	1	1	

white's score = -28

black's score = 28

run time : 548.084743976593

Process finished with exit code 0

ایده برای بهتر کردن پروژه:

استفاده از Transposition table

بررسی حرکت هایی که حول حرکتی هستند که حریف زده زودتر صورت بگیرد.

استفاده از Iterative deepening

استفاده از quiescence search

References:

1. https://en.wikipedia.org/wiki/Game_complexity_branching_factor تحلیل زمان مسئله و پیدا کردن
2. <http://dhconnelly.com/paip-python/docs/paip/othello.html#board> برای ساختار کلی پروژه، از توابع Makemove, makeflips, findbracket, nextplayer, legal moves, has/ any legal moves آن تقریباً به طور مستقیم استفاده شد.
3. <http://www.cs.cornell.edu/~yuli/othello/othello.html> استفاده از تابع eval، و نه کد آن
4. https://www.researchgate.net/publication/234785722_Othello_evaluation_function_comparison eval ایده برای تابع
5. https://pats.cs.cf.ac.uk/@archive_file?p=708&n=final&f=1-Final_Report.pdf&SIG=48a08fc35e0169234716d776bec6f534c2e97babba9edc889d4c99291178a38_eval اتلوجیست و استراتژی کلیم چه باشد، ایده ای تابع eval
6. <https://stackoverflow.com/questions/12334216/othello-evaluation-function> eval ایده برای تابع eval