**中山大学计算机学院**

**人工智能**

**本科生实验报告**

课程名称：Artificial Intelligence

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | **22336259** | 姓名 | **谢宇桐** |

# 实验题目

博弈树搜索--中国象棋

# 实验内容

1. 算法原理

博弈树启发式搜索策略即为结点排序技术。适用于此处的两个玩家对抗博弈的一种算法。在每次走出下一步时，玩家需要经过搜索算出对自己利益最大且对对方最不利的一步。

**极小极大算法（Minimax）：**

即深度优先搜索+逆向归纳法，是博弈树搜索的基本方法。首先假定，有一可以对所有的博弈进行评估的值。当值大于0时，表示博弈对我方有利，对对方不利。小于0时，表示博弈对我方不利，对对方有利：

1、当轮到我方走棋时，首先按照一定的搜索深度生成出给定深度d以内的所有状态，计算所有叶节点的评价函数值。然后从d-1层节点开始逆向计算：

2、对于我方要走的节点（用MAX标记，称为极大节点）取其子节点中的最大值为该节点的值（因为我方总是选择对我方有利的棋）。

3、对于对方要走的节点（用MIN标记，称为极小节点）取其子节点中的最小值为该节点的值（对方总是选择对我方不利的棋）。

4、一直到计算出根节点的值为止。获得根节点取值的那一分枝，即为所选择的最佳走步。

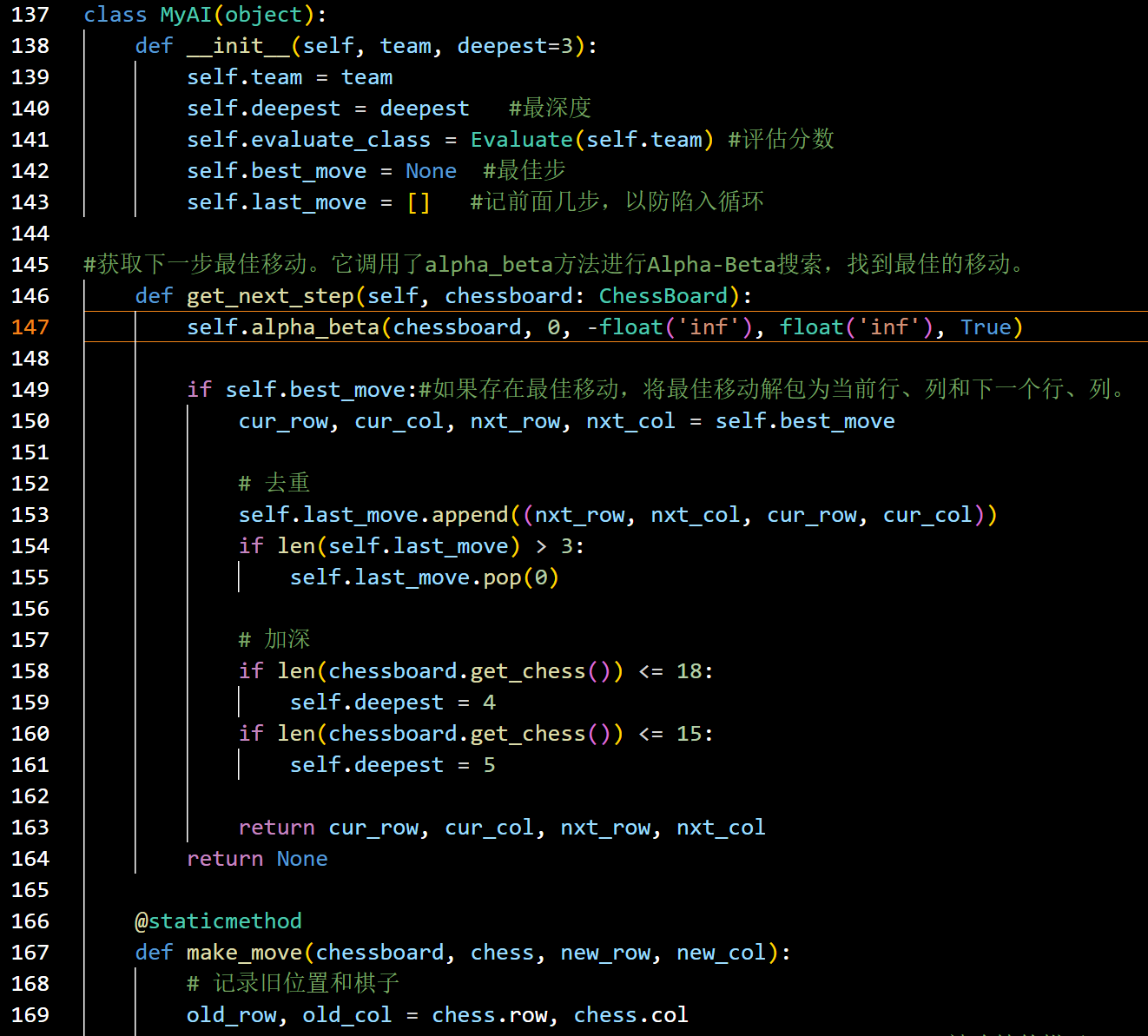
**Alpha-Beta剪枝：**

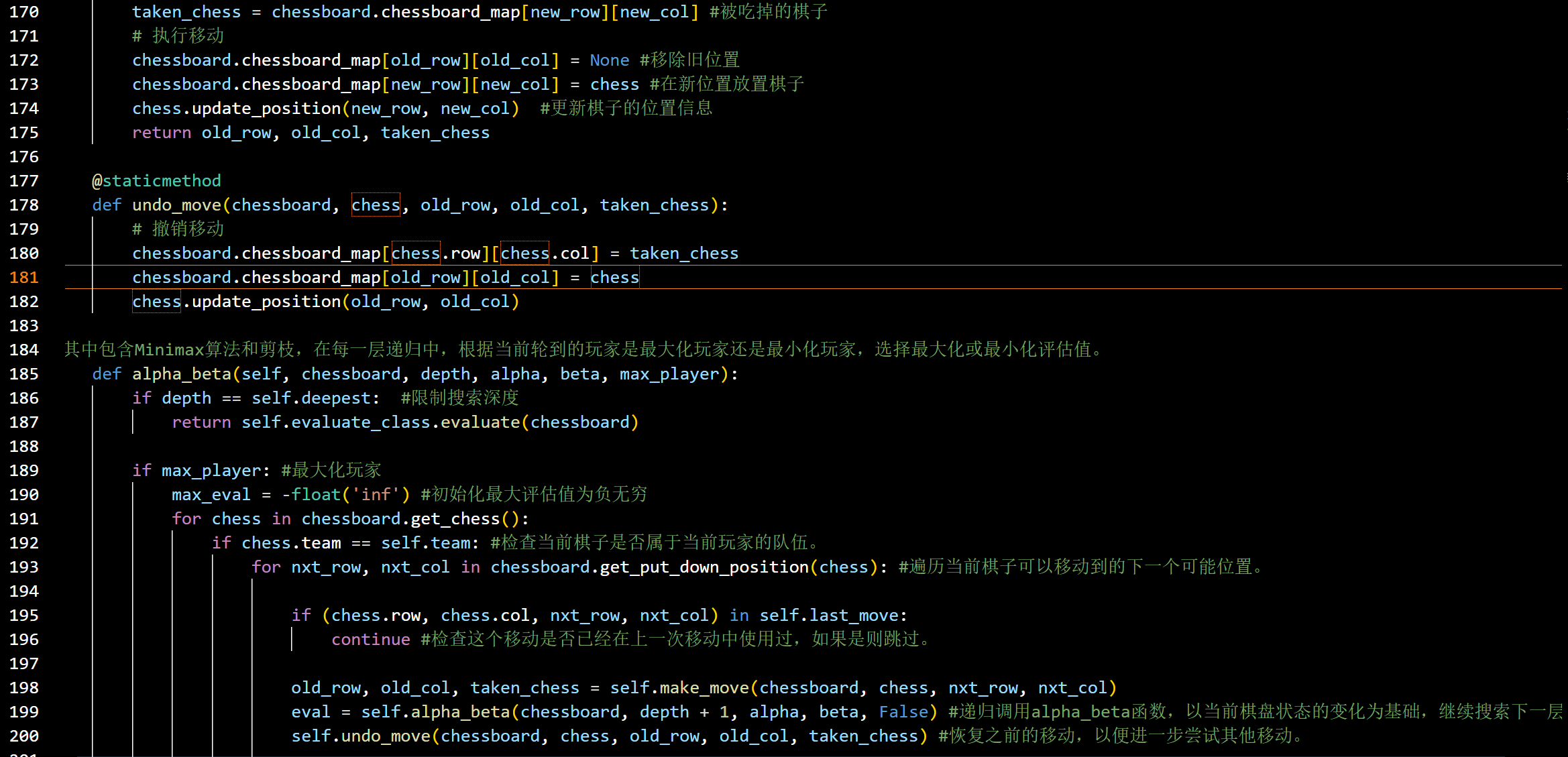
由于我们的搜索空间仍然非常庞大，在最开始的几层，可做的决策是相当多的。所以我们用到了α—β剪枝技术，减少没有必要的搜索，及时终止，从而加快了搜索速度。

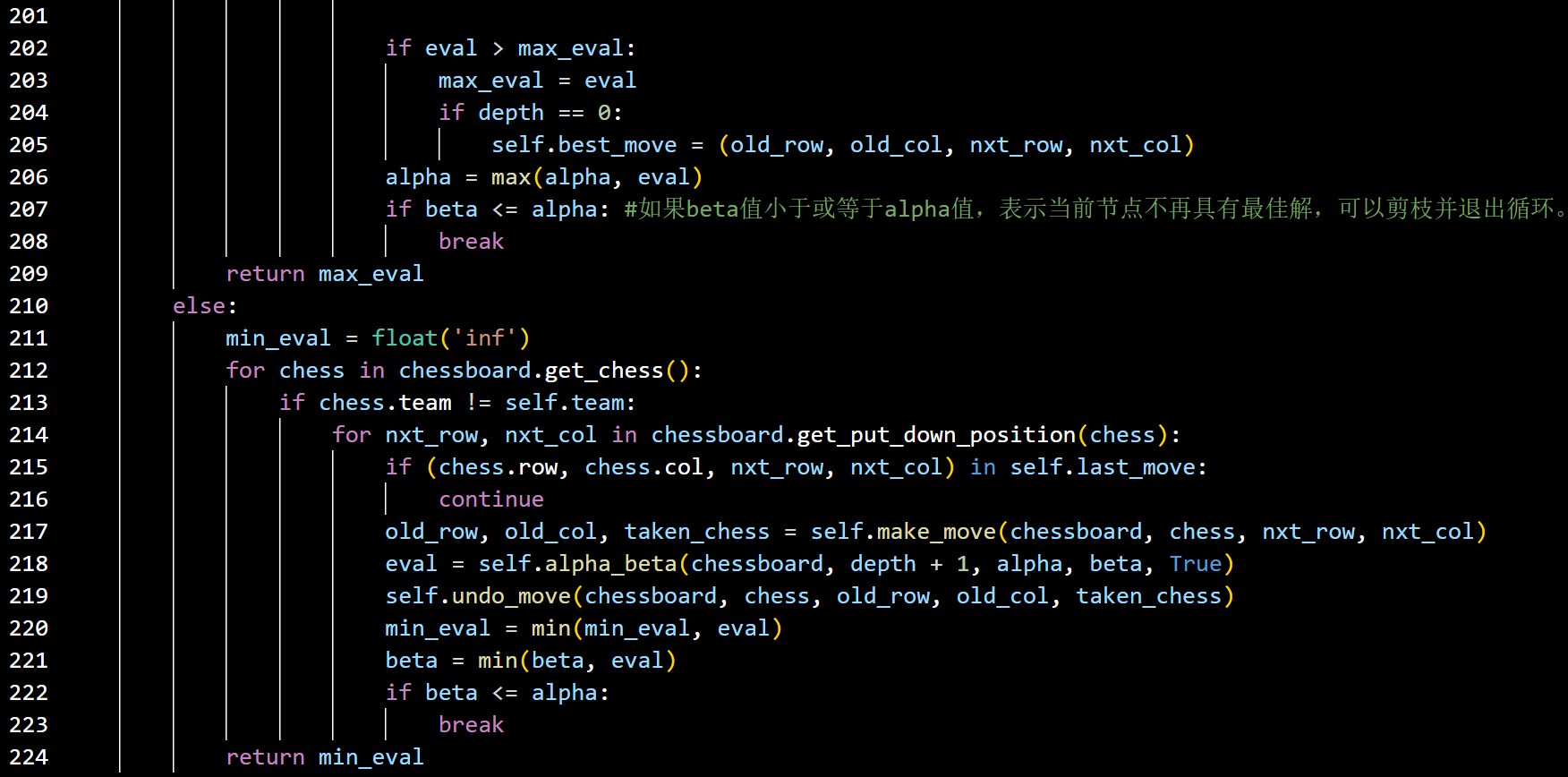
算法先到达底层，然后根据往上往下传承，以及更新规则更新。剪枝就发生在这个过程：Max 节点:其取子节点的最大值，α只更新更大，为下限。Max传承其父节点Min的β值，为上限。α根据 Max 节点的子枝信息由空更新到最大。在更新过程中如果出现α>=β，则 Max节点的余枝被剪枝。如果遍历 Max节点的子节点均未出现α>=β，将β更新成这个遍历以后的α。

1. 关键代码展示（可选）

MyAI.py:





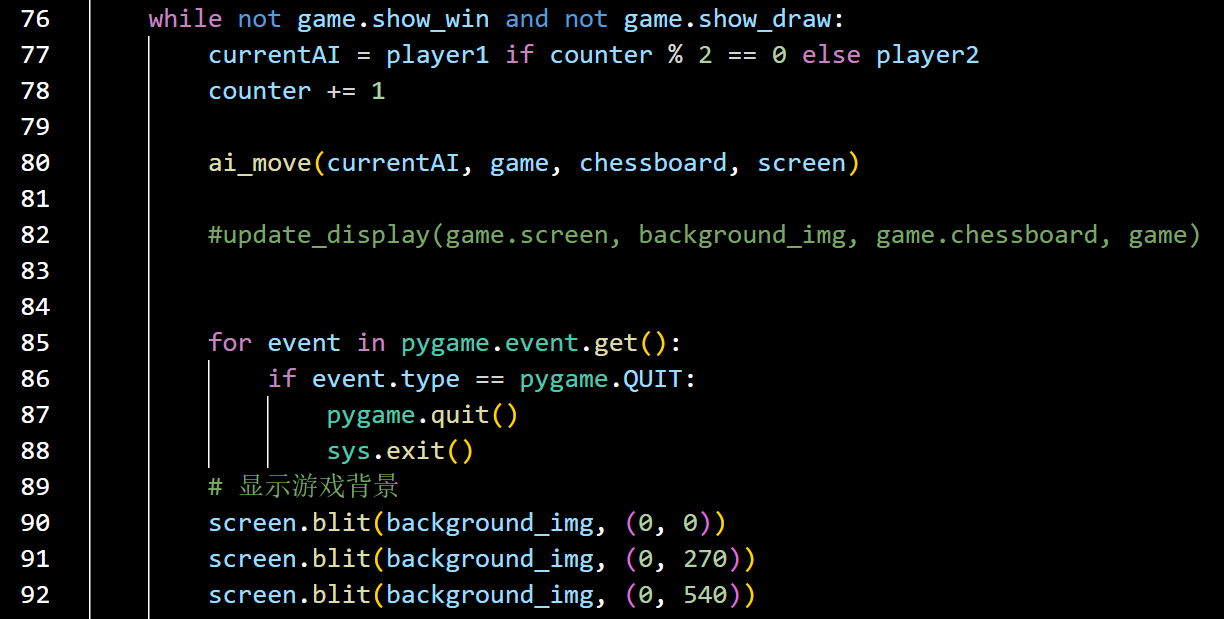


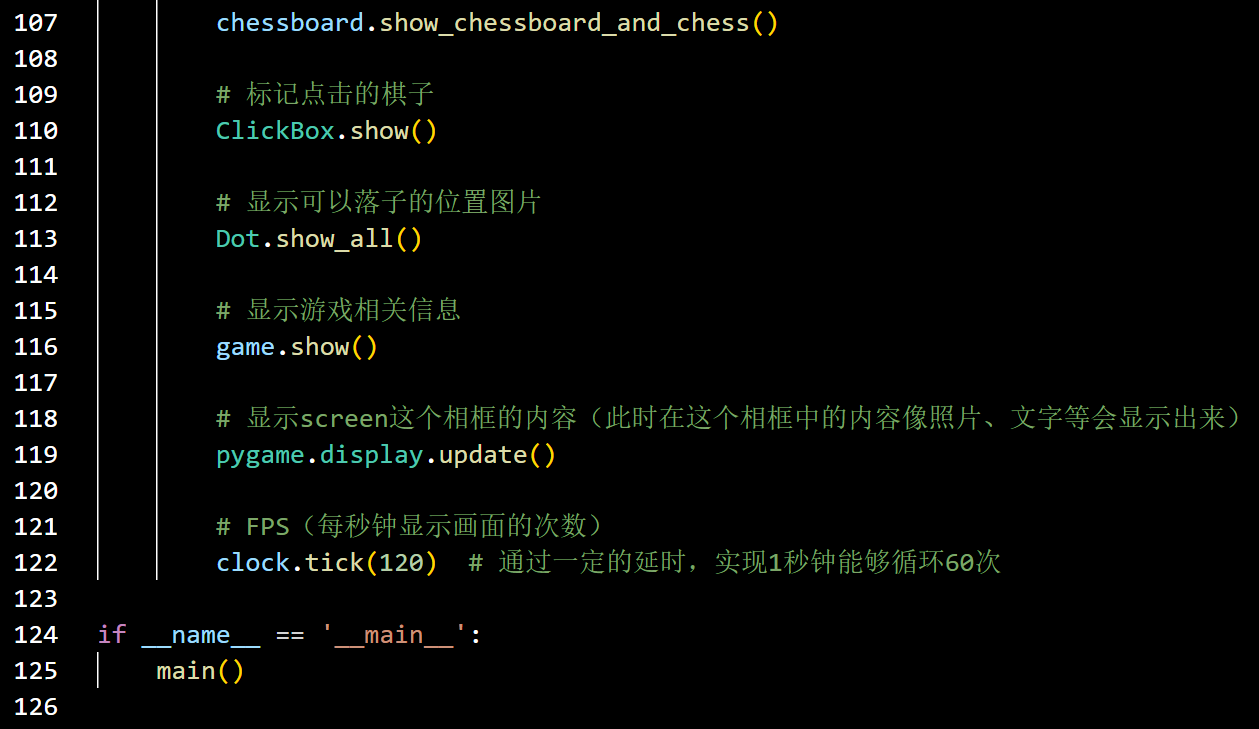
main.py:

因为之前的主函数不能实现我们的AI与程序之间的对战，所以我们重新写主函数，同时写一个ai\_move函数



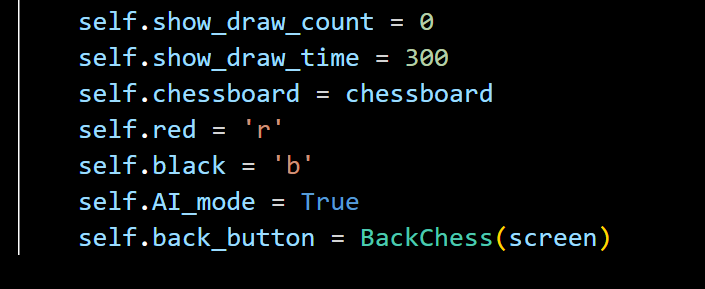






Game.py:

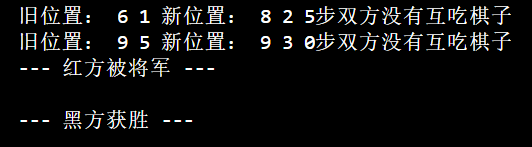
这里的文件做一些小改动。



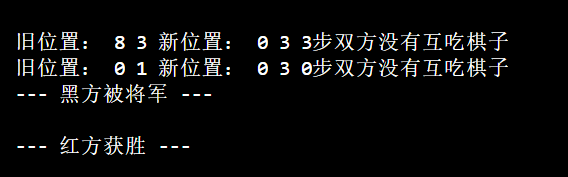
# 实验结果及分析

1. 实验结果展示示例（可图可表可文字，尽量可视化）

**AI红方先手：**



**我方先手：**



2. 评测指标展示及分析（机器学习实验必须有此项，其它可分析运行时间等）

根据实验要求，我们需要分别使用 MyAI 先手（执红棋）和后手（执黑棋）的对局结果。可以看到，我们的AI运行时间要比对方AI运行时间要短很多，这得益于α-β剪枝算法。但同时也有个问题，在对面AI运行时退出此程序非常卡，且刚开始时我的程序会黑屏，再次亮起时已经走了第一步。咨询同学后得知是因为我方AI太快，对方AI应加密过，pyarmor解密较慢导致。具体代码解析已将注释标注在代码中，下面是一些补充分析：

搜索深度控制：通过deepest参数控制搜索的深度，可以根据棋盘状态的复杂性来调整搜索深度。根据当前棋盘上棋子的数量，动态调整搜索深度，使得在关键时刻可以更深入地搜索，提高AI的水平。

移动选择策略：在最大化和最小化玩家之间切换，通过递归搜索所有可能的移动，以找到最优的走法。当评估值更高时，更新当前最佳移动步骤，以便在搜索结束时返回最佳的移动。

移动历史记录：通过last\_move列表记录前几步的移动，防止陷入循环，避免重复选择相同的移动序列。

# 参考资料

[最大最小法及α-β剪枝算法图解——CSDN](https://blog.csdn.net/xrying621/article/details/127254462?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522171309252316800226579871%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request_id=171309252316800226579871&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_ecpm_v1~rank_v31_ecpm-3-127254462-null-null.142%5ev100%5epc_search_result_base3&utm_term=%E5%8D%9A%E5%BC%88%E6%A0%91%E6%90%9C%E7%B4%A2%CE%B1%E2%80%94%E2%80%94%CE%B2%E5%89%AA%E6%9E%9D&spm=1018.2226.3001.4187)

[博弈树与α-β剪枝](https://blog.csdn.net/m0_51897707/article/details/125461537?ops_request_misc=&request_id=&biz_id=102&utm_term=%E5%8D%9A%E5%BC%88%E6%A0%91%E6%90%9C%E7%B4%A2%CE%B1%E2%80%94%E2%80%94%CE%B2%E5%89%AA%E6%9E%9D&utm_medium=distribute.pc_search_result.none-task-blog-2~all~sobaiduweb~default-2-125461537.142%5ev100%5epc_search_result_base3&spm=1018.2226.3001.4187)——CSDN

[博弈树 α-β剪枝——CSDN](https://blog.csdn.net/weixin_45693672/article/details/117296468?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522171309252316800226579871%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request_id=171309252316800226579871&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_ecpm_v1~rank_v31_ecpm-1-117296468-null-null.142%5ev100%5epc_search_result_base3&utm_term=%E5%8D%9A%E5%BC%88%E6%A0%91%E6%90%9C%E7%B4%A2%CE%B1%E2%80%94%E2%80%94%CE%B2%E5%89%AA%E6%9E%9D&spm=1018.2226.3001.4187)