HW4

T1

Question:

5.9 本题以井字棋（圈与十字游戏）为例练习博弈中的基本概念。定义 X_n 为恰好有 n 个 X 而没有 O 的行、列或者对角线的数目。同样 O_n 为正好有 n 个 O 的行、列或者对角线的数目。效用函数给 $X_3 = 1$ 的棋局+1，给 $O_3 = 1$ 的棋局−1。所有其他终止状态效用值为 0。对于非终止状态，使用线性的评估函数定义为 $Eval(s) = 3X_2(s) + X_1(s) - (3O_2(s) + O_1(s))$。

a. 估算可能的井字棋局数。

b. 考虑对称性，给出从空棋盘开始的深度为 2 的完整博弈树（即，在棋盘上一个 X 一个 O 的棋局）。

c. 标出深度为 2 的棋局的评估函数值。

d. 使用极小极大算法标出深度为 1 和 0 的棋局的倒推值，并根据这些值选出最佳的起始行棋。

e. 假设结点按对α-β剪枝的最优顺序生成，圈出使用α-β剪枝将被剪掉的深度为 2 的结点。

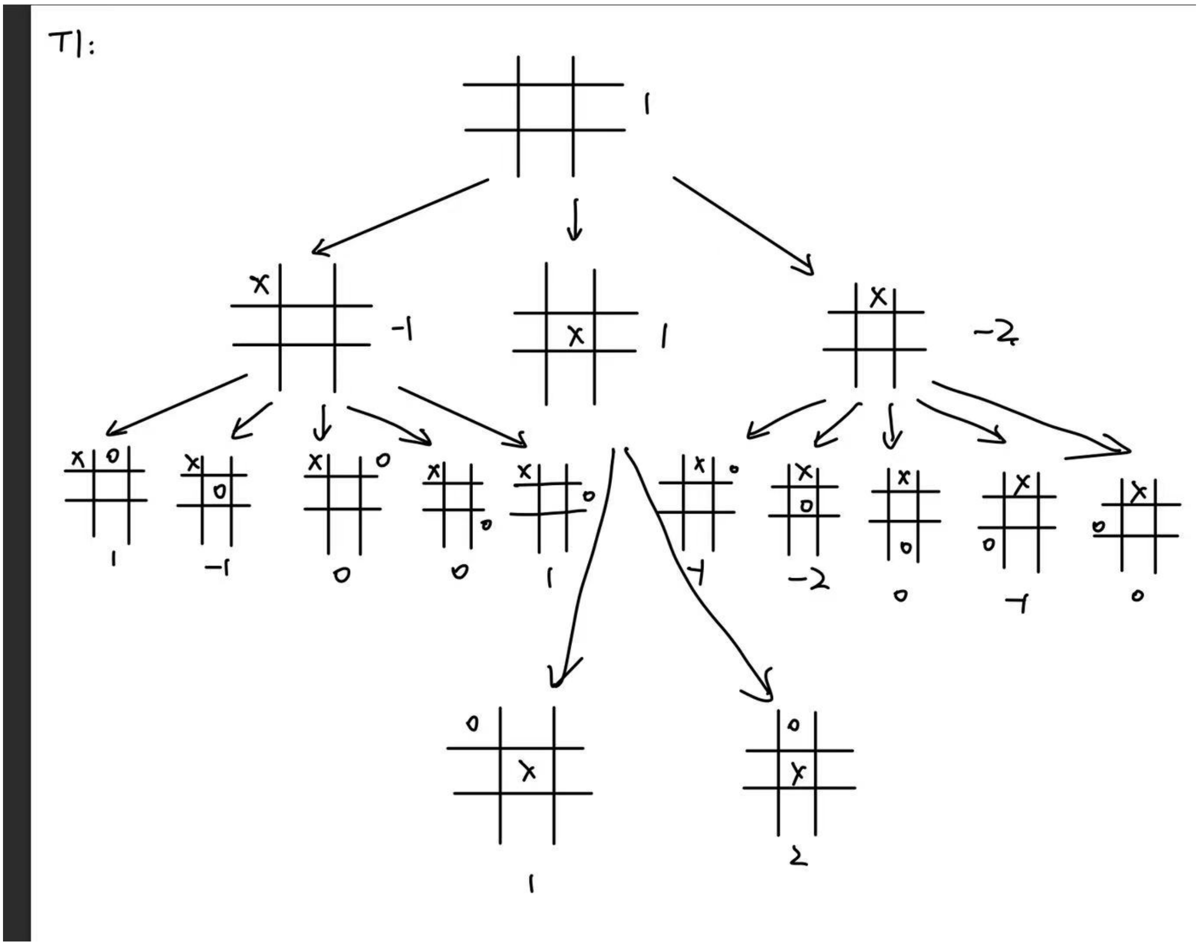
Answer:

a.

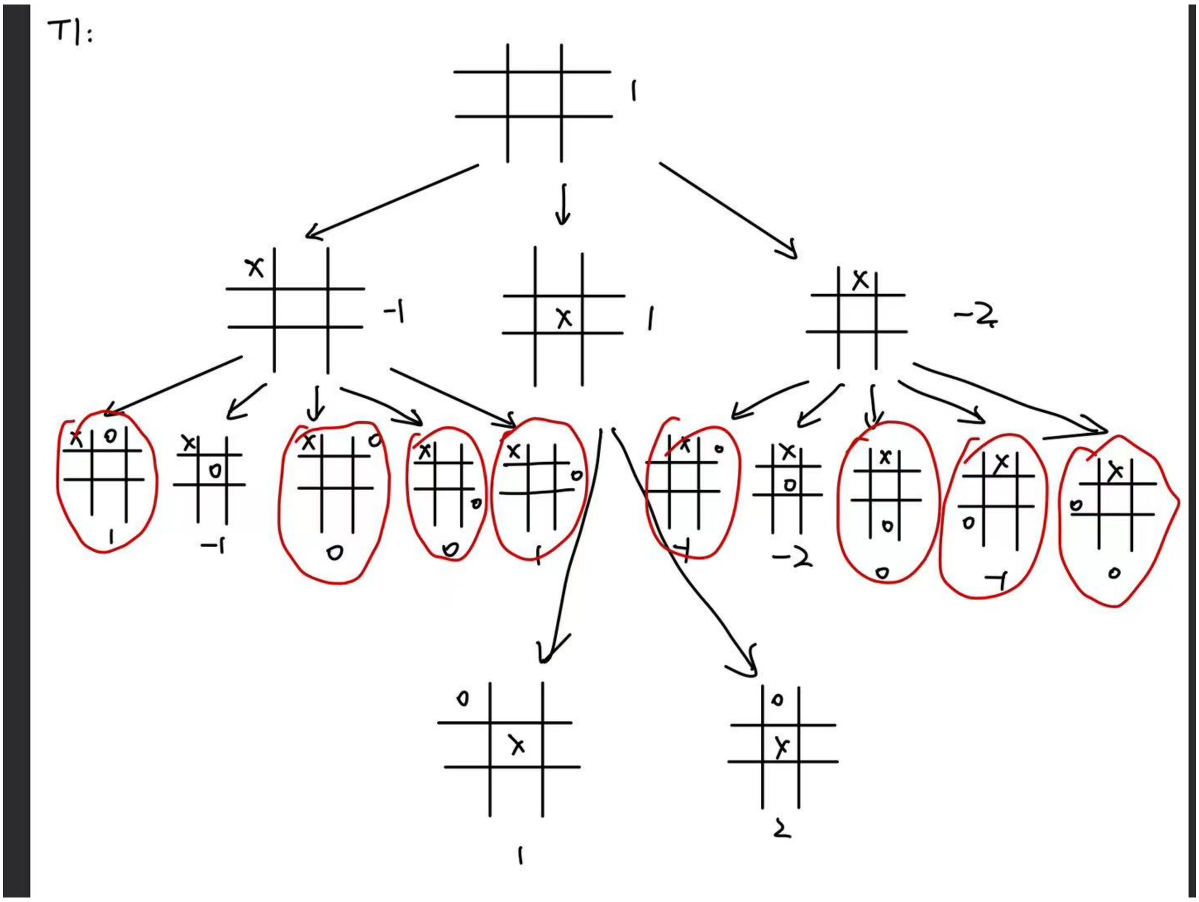
每个块有三种可能，棋盘的总可能情况不超过 3^9，路径总数不超过$9!$

b.c.d.

最优起始行棋：正中央



e.



T2

Question：

5.8 考虑图 5.17 中描述的两人游戏。

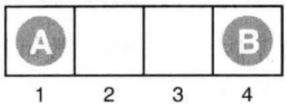


图 5.17　一个简单游戏的初始棋局

选手 A 先走。两个选手轮流走棋，每个人必须把自己的棋子移动到任一方向上的相邻空位中。如果对方的棋子占据着相邻的位置，你可以跳过对方的棋子到下一个空位。（例如，A 在位置 3，B 在位置 2，那么 A 可以移回 1。）当一方的棋子移动到对方的端点时游戏结束。如果 A 先到达位置 4，A 的值为+1；如果 B 先到位置 1，A 的值为−1。

a. 根据如下约定画出完整博弈树：

- 每个状态用(s_A, s_B)表示，其中 s_A 和 s_B 表示棋子的位置。
- 每个终止状态用方框画出，用圆圈写出它的博弈值。
- 把循环状态（在到根结点的路径上已经出现过的状态）画上双层方框。由于不清楚他们的值，在圆圈里标记一个“？”。

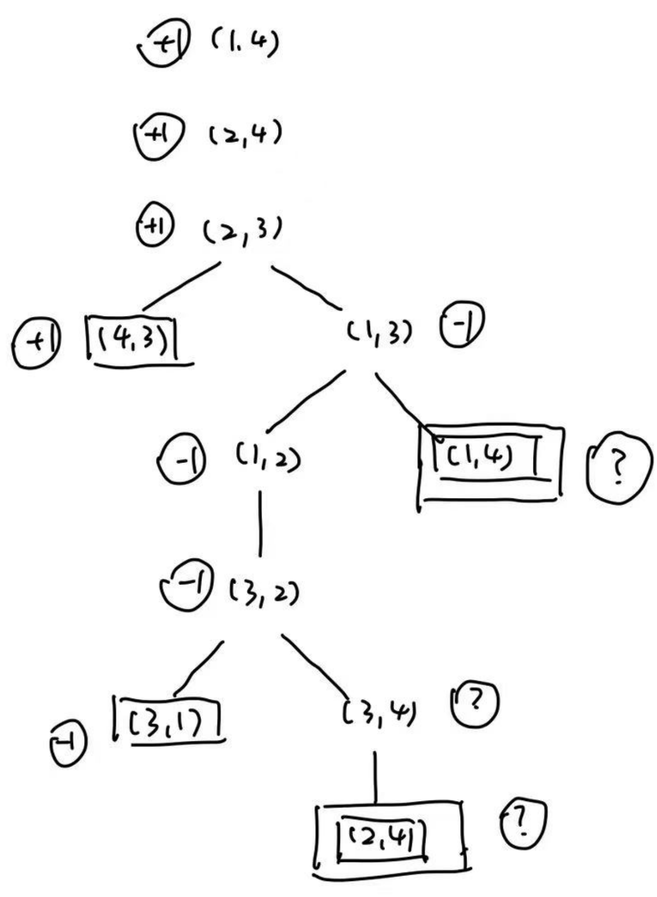
b. 给出每个结点倒推的极小极大值（也标记在圆圈里）。解释怎样处理“？”值和为什么这么处理。

c. 解释标准的极小极大算法为什么在这棵博弈树中会失败，简要说明你将如何修正它，在（b）的图上画出你的答案。你修正后的算法对于所有包含循环的游戏都能给出最优决策吗？

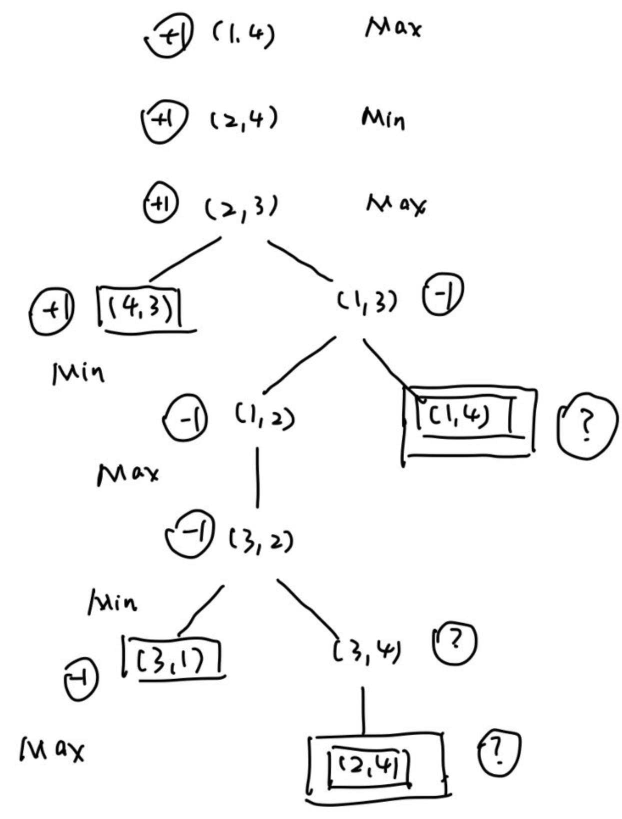
d. 这个 4-方格游戏可以推广到 n 个方格，其中 $n > 2$。证明如果 n 是偶数 A 一定能赢，而 n 是奇数则 A 一定会输。

Answer：

a.


+1 (1,4)
+1 (2,4)
+1 (2,3)
+1 (4,3)
(1,3) -1
-1 (1,2)
(1,4) ?
-1 (3,2)
-1 (3,1)
(3,4) ?
(2,4) ?

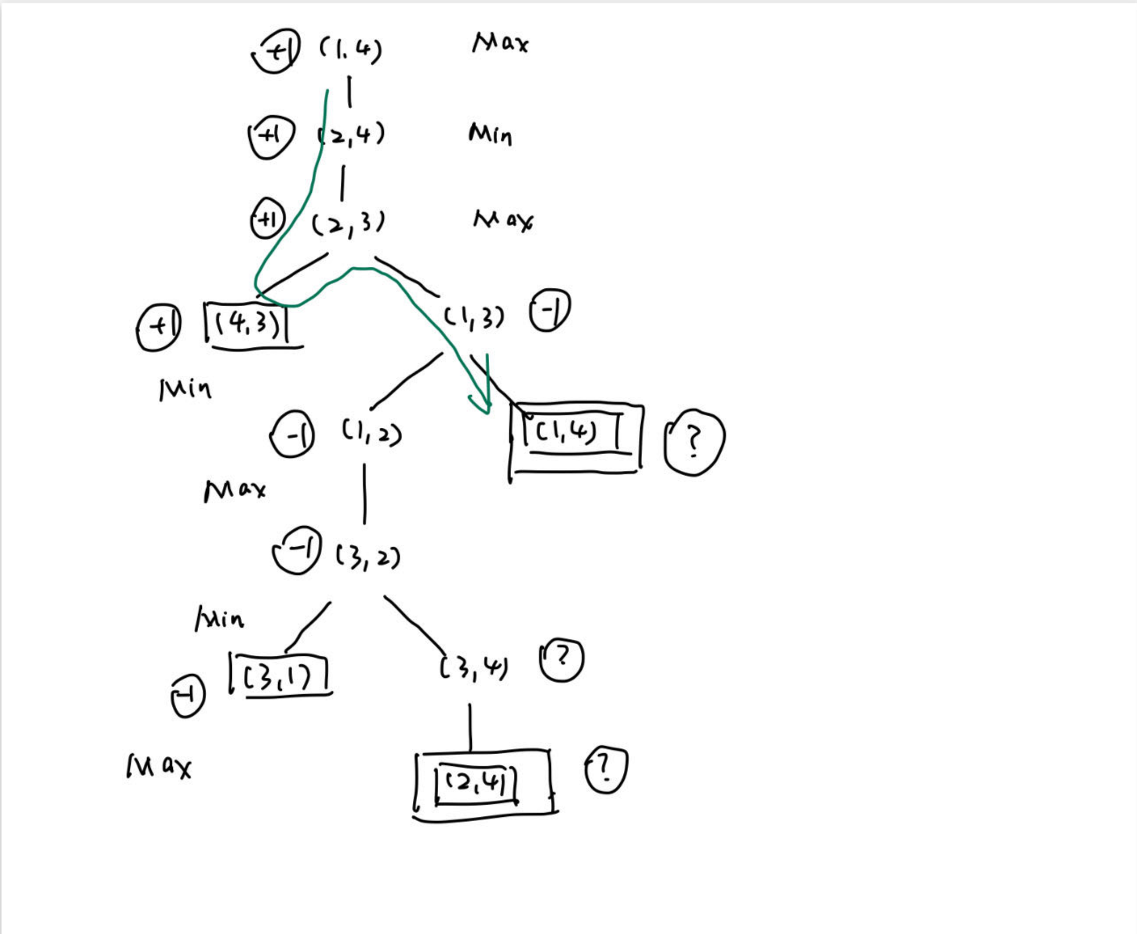
b.


+1 (1,4) Max
+1 (2,4) Min
+1 (2,3) Max
+1 (4,3)
(1,3) -1
Min
-1 (1,2)
(1,4) ?
Max
-1 (3,2)
Min
-1 (3,1)
(3,4) ?
Max
(2,4) ?

? 在 {-1, 1}中取值

- Min（-1，？） = -1
- Max（1，？） = 1

c.



标准的极大极小值算法会重复绿色路径陷入死循环

可以通过添加一个bool数组记录所有state是否被visit

不能保证解决所有问题，? 可能会有平局出现

d.

pf:

对于 (1,n)， next state是 (2,n) , (2, n-1)

- 因此 若n-2时A/B由必胜策略，则 n时也有必胜策略

递归到 n = 3/4情况

n = 3时， $(1,3) \rightarrow (2,3) \rightarrow (2,1)$，B有必胜策略

n = 4时，$(1,4) \rightarrow (2,4) \rightarrow (2,3) \rightarrow (4,3)$，A有必胜策略

T3

Question：

5.13 请给出α-β剪枝正确性的形式化证明。要做到这一点需考虑图 5.18。问题为是否要剪掉结点 n_j，它是一个 MAX 结点，是 n_1 的一个后代。基本的思路是当且仅当 n_1 的极小极大值可以被证明独立于 n_j 的值时，会发生剪枝。

a. n_1 的值是所有后代结点的最小值：$n_1 = \min(n_2, n_{21}, \cdots, n_{2b_2})$。请为 n_2 找到类似的表达式，以得到用 n_j 表示的 n_1 的表达式。

b. 深度为 i 的结点 n_i 的极小极大值已知，l_i 是在结点 n_i 左侧结点的极小值（或者极大值）。同样，r_i 是在 n_i 右侧的未探索过的结点的极小值（或者极大值）。用 l_i 和 r_i 的值重写 n_1 的表达式。

c. 现在重新形式化表达式，来说明为了向 n_1 施加影响，n_j 不能超出由 l_i 值得到的某特定界限。

d. 假设 n_j 是 MIN 结点的情况，请重复上面的过程。

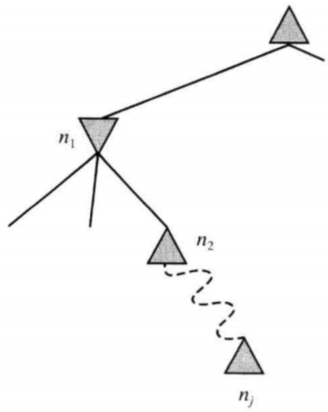


图 5.18　是否剪掉结点 n_j 时的情形

Answer：

a.

$n_2 = Max(n_3, n_{31}, \ldots, n_3 b_3)$，继续展开直到 n_j

b.

$n_1 = Min(l_2, r_2, n_2) = Max(l_2, r_2, Max(l_3, r_3, n_3))$，继续展开，直到$n_j$

c.

$n_{2k-1} = Min(l_{2k}, r_{2k}, Max(l_{2k+1}, r_{2k+1}, n_{2k+1}))$

$n_1 = Min(l_2, \ldots, Min(l_4, \ldots, Min, \ldots, Min(l_j, \ldots)))$

$n_j \le Min(l_2, \ldots l_j)$才会对$n1_1$施加影响

d.

$n_j \ge Max(l_3, \ldots, l_j)$才会对 n_1施加影响