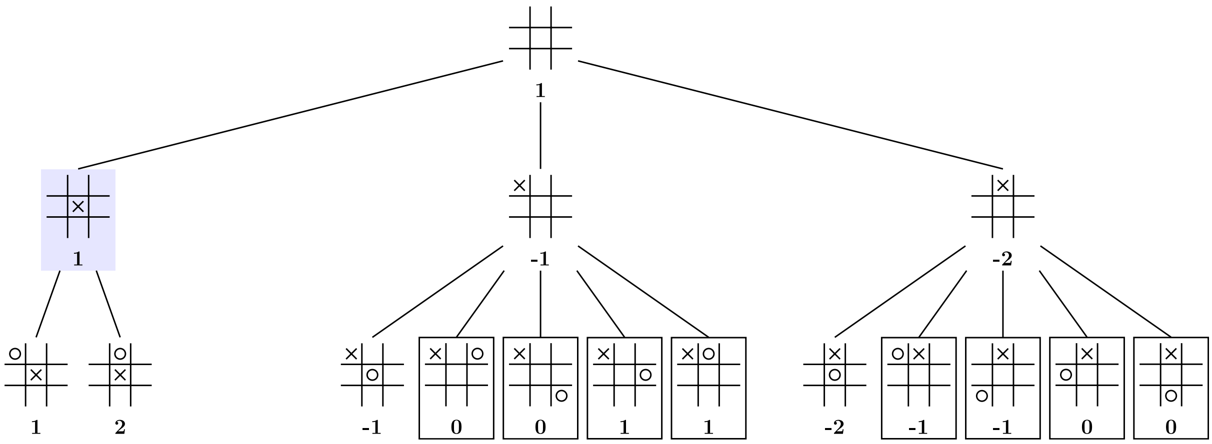
人工智能基础作业 4

傅申 PB20000051

5.9.

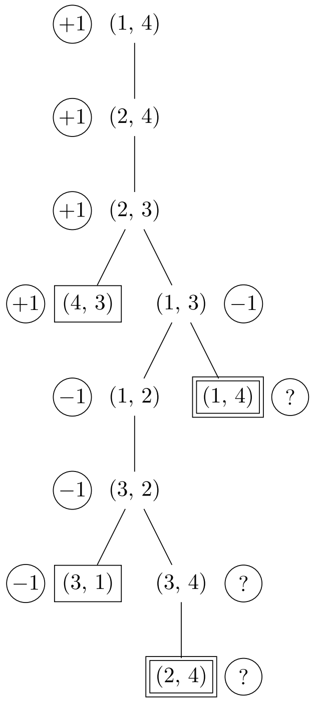
a. 可能的局数约为 9!.

b. c. d. e. 见下图, 最佳起始行棋为背景蓝色的结点, 用方框圈住的结点为被剪掉的结点.



5.8.

a. 见右图.



b. 若所有子结点的值为 ?, 则结点值为 ?; 否则, 按照如下规律计算值:

$$\max(+1, ?) = +1 \quad \max(-1, ?) = ?$$
$$\min(+1, ?) = ? \quad \min(-1, ?) = -1$$

这样处理没有违背 max 和 min 的性质.

c. 标准的 minimax 算法用到的深度优先搜索会陷入无限循环. 修正后的算法并不能处理所有包含循环的游戏, 比如在随机博弈中, 无法处理包含 ? 的平均值.

d. 显然 $n = 3$ 时 A 必败而 $n = 4$ 时 A 有必胜策略. 先考虑 $n > 4$ 为偶数的情况, 假设 A 一直往右走 (除非 A 位于 $n - 1$ 且 B 位于 n), 则不论 B 怎么移动, 在 A 和 B 第一次相邻时 (A 在 B 的左边) A 至少向右移动了 $\frac{n}{2}$ 步, B 至多向左移动了 $\frac{n}{2}$ 步, 且下一回合为 A 移动, 此时 A 能跳过 B 多移动一步, 无论 B 如何移动, A 总能比 B 先到达目标. 对于 $n > 4$ 为奇数的情况, B 采用类似的策略即可.

5.13.

a. 因为 $n_2 = \max\left(n_3, n_{31}, ..., n_{3b_3}\right) = \max\left(n_3, ..., \min(...), ..., n_{3b_3}\right)$, 其中 $\min(...)$ 与 n_j 相关 (不断递推), 所以

$$n_1 = \min\left(\max\left(n_3, ..., \min(...), ..., n_{3b_3}\right), n_{21}, ..., n_{2b_2}\right)$$

为用 n_j 表示的 n_1 的表达式.

b. 类似地, 有

$$n_1 = \min(l_2, \max(l_3, \min(l_4, ..., r_4), r_3), r_2)$$

递推的最后一项为 $\min(l_j, n_j, r_j)$.

c. 显然 $n_{2k} > l_{2k}$ 时 $\min(l_{2k}, n_{2k}, r_{2k})$ 与 n_{2k} 无关, $n_{2k-1} < l_{2k-1}$ 时 $\max(l_{2k-1}, n_{2k-1}, r_{2k-1})$ 与 n_{2k-1} 无关. 若要 n_j 对 n_i 造成影响, 显然有 $n_2 = n_3 = ... = n_{j-1} = n_j$, 则 n_j 需要满足

$$\max(l_3, l_5, ..., l_{j-1}) \leq n_j \leq \min(l_2, l_4, ..., l_j)$$

d. $\max(l_3, l_5, ..., l_j) \leq n_j \leq \min(l_2, l_4, ..., l_{j-1})$