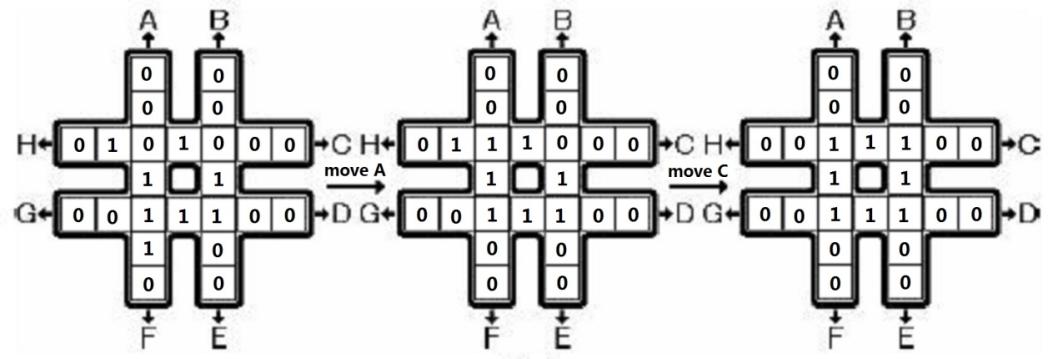
1. 全局搜索

第1题

旋转游戏使用一个井字形的棋盘，包含24个方格，其中8格上标着1，其它格上标着0。初始时棋盘上的1是随机分布的。你的任务是旋转其中一些行或列使得8个1到达内圈的8格上。图中给出了一个初始状态和操作的示意图，操作A表示将第一列的每个数字向上移动一格，最上面的数字会被移动到最下面的格子中，其余操作同理。经过AC两步操作后，棋盘已经到达目标状态。现在给定任意初始状态，希望用最少的操作到达目标状态，你决定用A\*算法解决该问题。（10分）



（1）A\*算法的伪代码填空（4分）

function A\*(start, goal, h):

openSet := {start}

cameFrom := empty map

g := map with default value of inf

g[start] := 0

f := map with default value of inf

f[start] := h(start)

while openSet is not empty:

（a）A

if current = goal:

return backtrack\_path(cameFrom, current)

openSet.remove(current)

for each neighbor of current:

（b）B

if （c）B:

cameFrom[neighbor] := current

（d）D

if neighbor not in openSet:

openSet.add(neighbor)

return failure

（a）

A current := node in openSet with lowest f

B current := node in openSet with lowest g

C current := node in openSet with highest f

D current := node in openSet with highest g

（b）

A t := f[current] + cost(current, neighbor)

B t := g[current] + cost(current, neighbor)

C t := f[current] + h(neighbor)

D t := g[current] + h(neighbor)

（c）

A t < f[neighbor]

B t < g[neighbor]

C t < h(neighbor)

D t < g[neighbor] + h(neighbor)

（d）

A f[neighbor] := t; g[neighbor] := g[current] + cost(current, neighbor)

B g[neighbor] := t; f[neighbor] := h(neighbor)

C g[neighbor] := t; f[neighbor] := g[current] + h(neighbor)

D g[neighbor] := t; f[neighbor] := g[neighbor] + h(neighbor)

（2）我们将每行每列两头的格子称为外圈，内圈与外圈之间的格子称为中圈。记内圈、中圈和外圈中1的个数为。

考虑以下四个启发式函数：

（a）判断这四个启发式函数是否满足可采纳性和一致性。若满足需要给出理由，不满足给出反例。（4分）可采纳：h1，h3 一致性:h1，h3

（b）判断这四个启发式函数是否一定使A\*算法找到最优解。如果有多个可以，选出你认为最好的一个并说明原因。（2分）h1废话

第2题

假设有两个人分别在两个不同的城市，现在有一张地图（每个城市是一个点，城市间有公路相连）。在每个时间步，每个人要走到他/她的相邻城市。从城市i 走到相邻城市 j 的时间花费是连接两座城市的公路距离 d(i, j)，但是每一次先到达自己相邻城市的人要等待另一个人也得到达了（另一个人的）相邻城市，他们才可以同时再出发。我们希望两个朋友尽快相见（到达同一座城市）。

1. 写一个搜索问题模型（五元组）来描述这个问题。(城市对，相邻城市对乘积，，，）
2. 假设D(i, j)就是城市间的直线距离，下面的启发式函数哪个（些）是可采纳的（admissible）？(i) D(i, j); (ii) 2 · D(i, j); (iii) D(i, j)/2.（iii）
3. 是否存在下面的情况：在一个连通的地图上，给定了两个人的初始位置，却找不到一种移动方式使他们能够最终见面？请说明你的答案。当然
4. 是否存在这样的地图，无论两人最初身在何处（不同城市），必须有至少一个人要经过同一个城市两次，他们才能最终见面？请说明你的答案。不存在

第3题

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |
| 13 | 14 | 15 |  |

在4\*4的棋盘上摆有15枚棋子，分别标着1~15的数字，初始时棋盘上的棋子是随机分布的。每一回合你可以选择一枚和当前空格同行或者同列、且距离不超过2的棋子，将其移动到空格上。游戏目标是使用尽可能少的步骤将初始棋盘还原成目标状态。下图给出了一种初始状态（左）和目标状态（右）的示意图，图中的初始局面可以使用4步进行还原：按顺序选择1、2、10、12四枚棋子。你决定使用A\*算法解决该问题。（10分）

|  |  |  |  |
| --- | --- | --- | --- |
|  | 1 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 2 | 11 | 10 |
| 13 | 14 | 15 | 12 |

（1）考虑以下四个启发式函数：

为不在目标位置的棋子个数；

为不在目标行的棋子个数+不在目标列的棋子个数（注：同时满足的棋子会被计算两次）；

为每个棋子与目标位置的曼哈顿距离（横向距离+纵向距离）之和

1. 判断这四个启发式函数是否分别满足可采纳性和一致性。（4分）

可采纳性：h1,h4

一致性:h1,h4

（b）判断这四个启发式函数是否一定使A\*算法找到最优解。如果有多个可以，选出你认为最好的一个并说明原因。（2分）h1

（2）对于一般问题上的A\*算法，判断以下命题的正确性并说明理由。（4分）

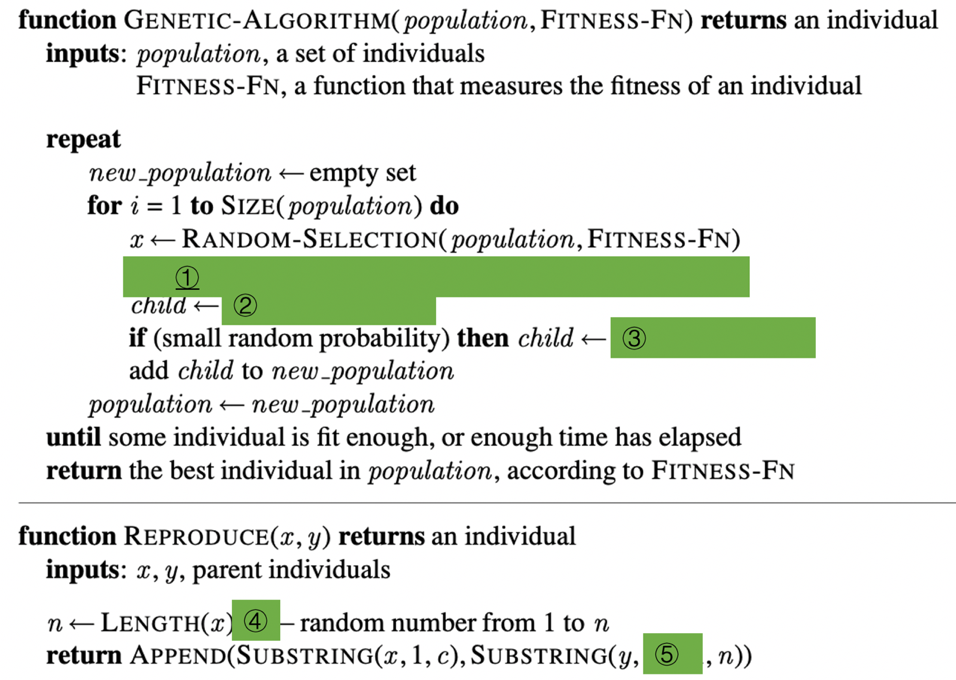
（a）可采纳的启发式函数一定是一致的。不一定

（b）一致的启发式函数一定是可采纳的。一定（要求目标点的h不大于0）

1. **局部搜索**

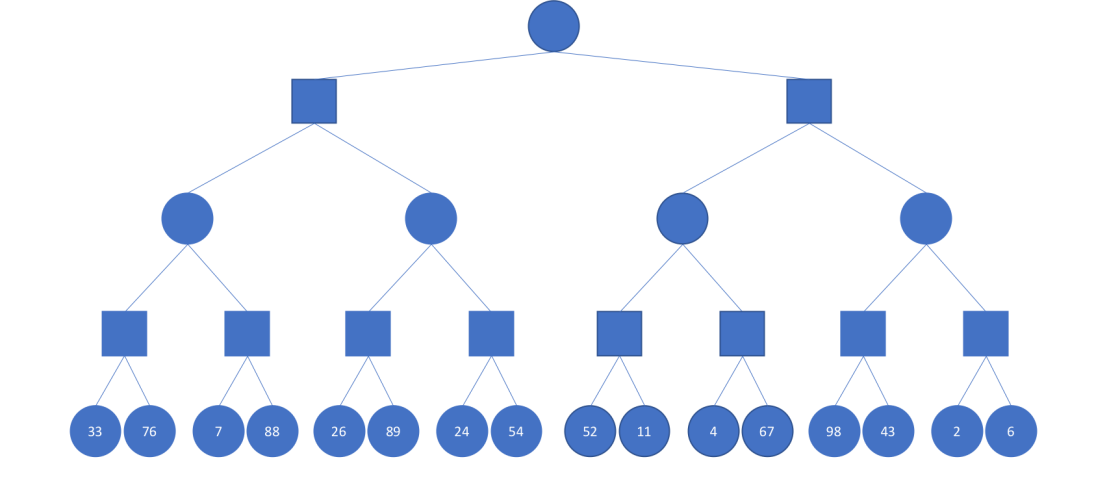
第4题

程序填空，下面是遗传算法的伪代码，请在空格出填上相应的代码。



1. **对抗搜索**

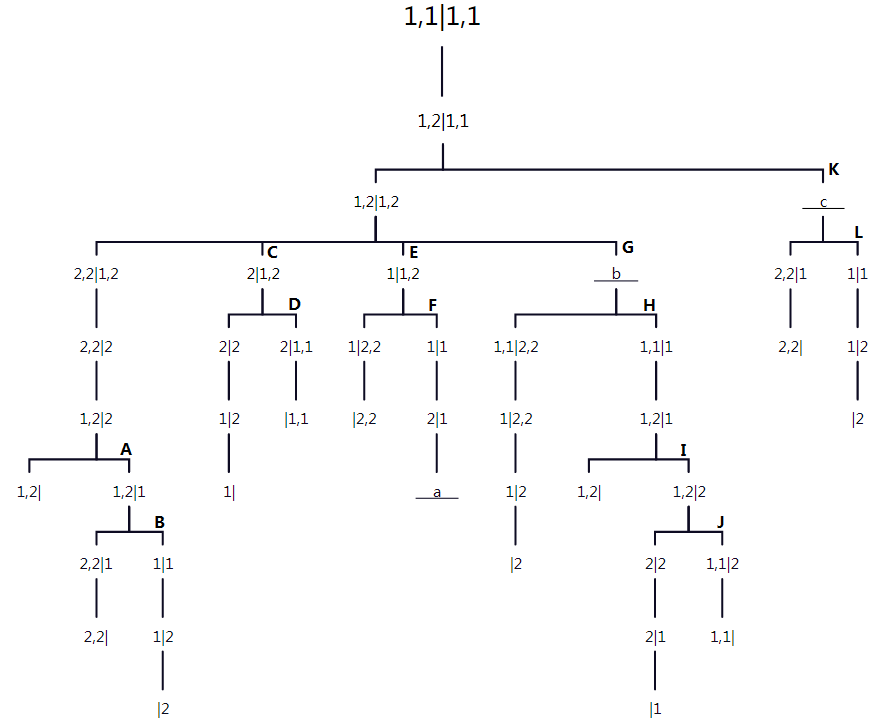
第5题：αβ剪枝



上图是一棵双人游戏的最大最小搜索树，圆形节点在子节点中求最大值，方形节点在子节点中求最小值，请在上图中没有填写数字的节点处填上正确数字。如果用αβ剪枝算法对这棵树做剪枝，请圈出哪些节点会被剪掉。

第6题：

甲乙两个人在玩一个纸条游戏，每个人有两张纸条并且互相可见。游戏开始时所有纸条上都写着1，双方轮流决策，每次每个人选择自己的一张纸条和对方的一张纸条，将自己纸条上的数字改成两者的和模3，如果得到0就扔掉自己的这张纸条, 谁先扔完两张就获胜，双方分数为对方剩下的纸条数量减去己方剩下的纸条数量（如果获胜时对面剩下一张，分数为1和-1，如果对面剩两张，分数为2和-2）。该游戏的博弈树已给出，如图所示。（10分）



1. 在博弈树中，状态用两个玩家纸条上的数字表示，每个玩家的数字用逗号隔开，玩家之间用竖线隔开，如1,1|2表示甲剩两张纸条都写着1，乙剩一张纸条写着2。请填写图中留空的三个状态。（3分）

（a）

（b）

（c）

1. 甲乙两名玩家都是理性的，求甲先手能得到的最高分。（3分）

第7题

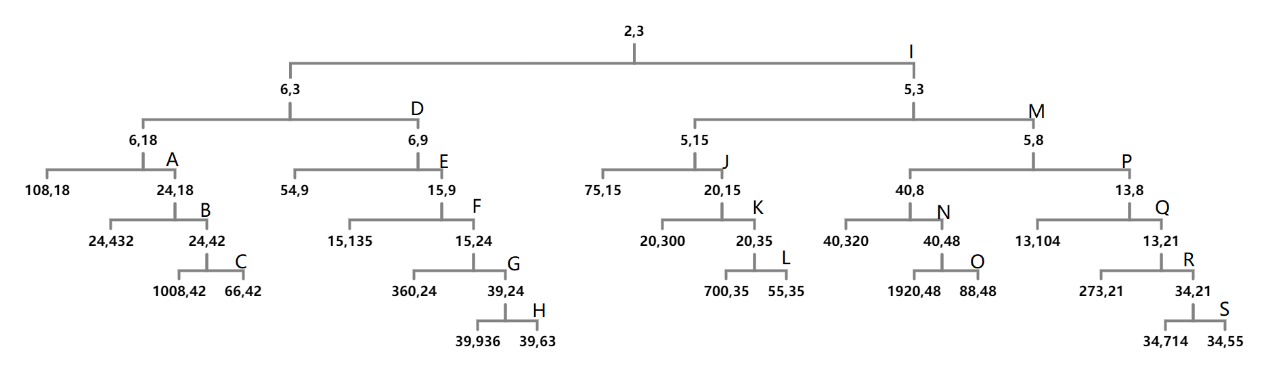
对于双人零和游戏，MAX 一方使用 MINIMAX 算法，在对阵使用最优策略（最小化敌方收益） 的 MIN 时会不会比它对阵使用非最优策略的 MIN 时收益更低?为什么?你能否给出 一个博弈树的例子，在 MIN 方使用了非最优策略时，MAX 方使用一个非MINIMAX算法从而获得比使用 MINIMAX 算法更大的收益?

第8题

1. 请描述一下minimax 算法如何能转换成解决两人非零和游戏的算法？假设两个人每人有一个独立的得分规则，并且两个人都知道彼此的得分规则。
2. 如果对两个人结束游戏时的得分没有限制（非零和），我们是否还能用alpha–beta 算法进行剪枝?

第9题

甲乙两人在玩一个小游戏，双方各拥有一个数字。游戏开始时甲的数字是2，乙的数字是3，双方轮流决策，甲先手。每个回合当前玩家可以选择将双方的数字相乘或者相加，用得到的结果替换自己的数字，谁的数字先超过50谁就获胜，每个人的得分是自己的数字减掉对手的数字。该游戏的博弈树已给出，如图所示，状态为甲和乙的数字。（10分）



（1）已知甲乙两名玩家都是理性的，求甲先手能得到的最高分。（5分）

（2）对图中的博弈树进行alpha-beta剪枝，选出所有剪枝的位置（A~S）。（5分）