2048 AI

1. 问题分析：

1、 小游戏2048是一款数字类的小游戏，本人也玩过很多次。自己在玩的过程中也曾发现一些规律：为以下几点：

1. 最大的数尽量要放在一个角落，并且在角落中后就不要再去改变其位置。
2. 保证空格数尽量的多。
3. 选择最大数所在的行（列）作为主行（列），即在这一行（列）上的数从最大数开始尽可能的保持递减，这个行（列）选定后就尽量不变，除非在极短的步骤内列（行）上的单调性更好，或行（列）上的数恰好合并完了，而列（行）上又有数 时可以考虑主行（列）转成主列（行）。
4. 数据尽量保持一种相近性，相邻的数之间的绝对值之差要小。
5. 在保持主行（列）的条件下，主行（列）相邻行的单调性如果能保持衔接对局势会很有利。比如如下的排列方式（主列）：

|  |  |  |  |
| --- | --- | --- | --- |
| 256 | 128 | 2 |  |
| 512 | 64 | 2 |  |
| 1024 | 32 | 4 |  |
| 2048 | 16 | 8 |  |

1. 发现如果排列是按zig\_zag的顺序递减，其也很容易收拢合并。比如如下方式：

|  |  |  |  |
| --- | --- | --- | --- |
| 32 |  |  |  |
| 64 | 16 |  |  |
| 1024 | 128 | 8 | 4 |
| 2048 | 512 | 256 | 4 |

其按上、左、左、左、下、左、左、上、左、下、下 的方式可以不管出现的数字是什么而最终完成2048的收拢。所以该种形势也是有利的。

2、预计完成所需时间：

根据以上可以写出一个评估函数，可以由当前局面进行深度搜索，朝哪个方向走的评估函数值高，就返回哪个方向。总计一个评估函数，一个深度搜索。 预计两天可以完成。

1. 算法设计：
2. 将其抽象成信息对称双人对弈模型（玩家向四个方向中的一个移动，然后计算机在某个空格中填入2或4），假设机器会考虑局面，将局面形势带入较差（即：评估值最小）的情形，而人则尽量让场上形势最优（即：评估最大）。于是就可以看成一种对弈。这样就可以选取Minimax算法，为了增加搜索深度，采用了带Alpha-Beta 剪枝的Minimax算法。
3. 由于实际上2048游戏中，机器走的时候是随机在剩余生成2或4的，所以上面1中假设机器每次都将局面带入最劣是有一些问题的，个人觉得在机器走的时候对机器所有的可能走法算期望值或许更好。相对于上一种方法在这里只需将Min步求取评估最小的走法改为计算所有可能走法的均值。
4. 最终的设计中选取了第二种方式。
5. 实现：

程序给的接口中，给出的是当前局面的数字（4\*4的数组）。

1. 类的设计：

类内完成上、下、左、右的走法，当前局面下单调性计算，平滑性计算，空格的计算，最大值的计算，评估函数的实现。

1. 递归函数：
2. 结果：

基本上能以80%~90%的概率实现2048，偶尔能实现4096。得分平均在30000分左右。