MoveSetting 控制棋子移动的类

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| nMoveCount | int |  | 总的走法数量 |
| CHESSMOVE | struct |  | 定义走法结构 |
| CHESSMANPOS | struct |  | 定义棋子位置的结构 |
| IsBlack | bool | (int PieceNum) | 判断一个棋子是不是黑色 |
| IsRed | bool | (int PieceNum) | 判断一个棋子是不是红色 |
| IsSameSide | bool | (int PieceNumX, int PieceNumY) | 判断两个棋子是不是同颜色 |
| CreatePossibleMove | int | (int[,] position,  int nPly,  bool nSide) | 产生给定棋盘上所有合法的走法，并将走法存入nMoveCount中 |
| ClickChessMoveDraw | void | (int fromx,  int fromy) | 棋子可能移动的位置的绘制 |

Rules类 判断下棋规则的类

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| Jiang\_x | int |  | 将的X坐标 |
| Jiang\_y | int |  | 将的Y坐标 |
| Shuai\_x | int |  | 帅的X坐标 |
| Shuai\_y | int |  | 帅的Y坐标 |
| IsSameSide | bool | (int PieceNumX,  int PieceNumY) | 判断两个棋子是不是同颜色 |
| KingBye | bool | (int [,]CHESS,  Int FromX,  int FromY,  int ToX,  int ToY) | 判断该步棋老将是否对面 |
| IsValidMove | bool | (int [,]CHESS,  Int FromX,  int FromY,  int ToX,  int ToY) | 判断该步棋是否是正确的走法 |
| JiangJunCheck | void | () | 判断是否被将军，并显示文字提示。若将被吃了游戏结束 |

ChessControl类 控制棋子移动和吃子事件并调用AI对弈

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| posthread | bool |  | 判断Invoke函数是否执行完毕 |
| TrueOrFalse | bool |  | 判断这个时候输赢状态能否走棋 |
| FromX, FromY, ToX, ToY | int |  | 存储位置移动的起点和终点信息 |
| BlackNameOrRedName | void | (GameObject ClickItem) | 判断点击的棋子是黑方还是红方 |
| threm | void | () | AI走下一步棋 |
| GetToPositionObjName | string | (MoveSetting.CHESSMOVE move) | 得到一步棋目标位置gameobject的对象名字 |
| GetFromPositionObjName | string | (MoveSetting.CHESSMOVE move) | 得到一步棋开始位置gameobject的对象名字 |

BackStepChess类 控制悔棋功能

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| QIZI | struct |  | 将对弈的过程存储起来 |
| HUIQIposition | struct |  |  |
| IsOnePersonWin | bool |  | 判断现在是否单人模式用户胜利 |
| Count | int |  | 统计的步数 |
| AddChess | void | (int count,int fromx,int fromy,  int tox,int toy,  bool IsTrueOrfalse  ,int ID1,int ID2) | 添加棋子移动开始位置的坐标和终点位置的坐标 |
| IloveHUIQI | void | () | 点击事件实现悔棋功能 |

BtnControl 按钮控制类

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| isFirstStart | bool |  |  |
| ChessPeople | int |  |  |
| BackMainScene | void | () | 返回主界面 |
| OnePeopleModel | void | () | 开始单人模式 |
| TwoPeopleModel | void | () | 开始双人模式 |
| SetUpButton | void | () | 点击设置按钮 |
| GameOver | void | () | 点击游戏结束按钮 |

SearchEngine AlphaBeta剪枝算法AI类

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| m\_nMaxDepth | int |  | AI算法最大搜索深度 |
| m\_nSearchDepth | int |  | AI算法搜索深度 |
| m\_cmBestMove | CHESSMOVE |  | 存放最佳走法的变量 |
| BaseValue | int [15] |  | 棋子基本价值 |
| FlexValue | int [15] |  | 棋子灵活性分数数组 |
| AttackPos | int [10,9] |  | 每个位置威胁信息 |
| GuardPos | int [10,9] |  | 存放每个位置被保护的信息 |
| FlexibilityPos | int [10,9] |  | 存放每个位置上的棋子灵活性 |
| chessValue | int [10,9] |  | 存放每个位置的棋子总价值 |
| nPosCount | int |  | 记录棋子的相关位置个数 |
| count | int |  | 用来统计调用了估值函数的也子节点次数 |
| BA0 | int [10,9] |  | 红兵的附加值数组 |
| BA1 | int [10,9] |  | 黑兵附加值数组 |
| MakeMove | int | (MoveSetting.CHESSMOVE move) | 根据传入的走法改变棋盘  ,move是要进行的走法, 返回被吃掉的棋子 |
| UnMakeMove | void | (MoveSetting.CHESSMOVE move, int nChessID) | 根据走法恢复棋盘,move是要恢复的走法,nChessID是原棋盘上move位置的棋子类型 |
| IsGameOver | int | (int [,]position,int nDepth) | 检查游戏是否结束,如果结束，返回0，否则返回计算的极大极小值 |
| PrincipalVariation | int | (int depth,int alpha,int beta) | Alpha-beta 剪枝算法，返回走法计算的价值 |
| SearchAGoodMove | MoveSetting.CHESSMOVE | (int[,] position) | 搜索最优走法，并返回 |
| GetBingValue | int | (int x,int y,int [,]CurSituation) | 获取每一个兵附加值 |
| Eveluate | int | (int [,]position,bool bIsRedTum) | 计算棋盘每个棋子价值 |
| GetRelatePiece | void | (int [,]position,int j,int i ) | 列举与指定位置的棋子相关的棋子 |
| CanTouch | bool | (int [,]position,int nFromX,int nFromY,int nToX,int nToY) | 判断from 位置能否走到to |
| AddPoint | void | (int x,int y) | 这个函数将一个位置加入数组relatepas中 |

ReStarChess 重新开始控制类

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| NextTips | Text |  | 提示下一步的信息 |
| ChessStart | void | () | 重新开始游戏，进行棋盘的初始化 |

Board棋盘界面类

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| DiffcultLevelText | Text |  | AI难度的文本显示 |
| chess | int[10, 9] |  | 存储棋盘的数组 |
| SetPiecePos | void | () | 动态添加了90个棋盘的格子 |
| InitPiece | void | () | 初始化棋子 |
| ChessInit | void | () | 棋盘初始化 |

ChessClick类 棋子的点击事件

|  |  |  |  |
| --- | --- | --- | --- |
| 属性/方法名 | 类型 | 参数 | 功能含义 |
| clickItemOrChess | void | () | 为每个棋子预设体添加点击事件 |