**实验报告**

1. 规则与功能
2. 规则

对局双方各执一色棋子，空棋盘开局，黑先、白后，交替下子，每次只能下一子，棋子下在棋盘的空白点上，棋子下定后，不得向其它点移动，不得从棋盘上拿掉或拿起另落别处。

1. 功能 //没有什么功能/(ㄒoㄒ)/~~
   1. 可以选择人机模式，或者自己和自己下模式 //这算什么功能
   2. 可以在人机模式里面选择先手或者后手
   3. 可以在棋局结束后选择当前模式下重开
2. 完成流程
3. 使用Microsoft Visual Studio2022版编写
4. 对如何写一个自己的GUI一无所知，所以直接用了easyX，在网上简单学了几个函数搬上来用了
5. 模块
   1. 绘制棋盘与初始化

initChess(void);

drawChessboard(void);

getReady(void);

goFirst(void);

* 1. 落子

coordinate(int x);

judgeBlank(int x, int y);

Mouse(void);

placePieces(int x, int y);

* 1. 判断输赢与重开

judgeWin(int x, int y);

judgeRestart(int x, int y);

restart(void)；

exitgame(void);

* 1. AI

boardJudge(int x, int y);

putScore(int x, int y, int p);

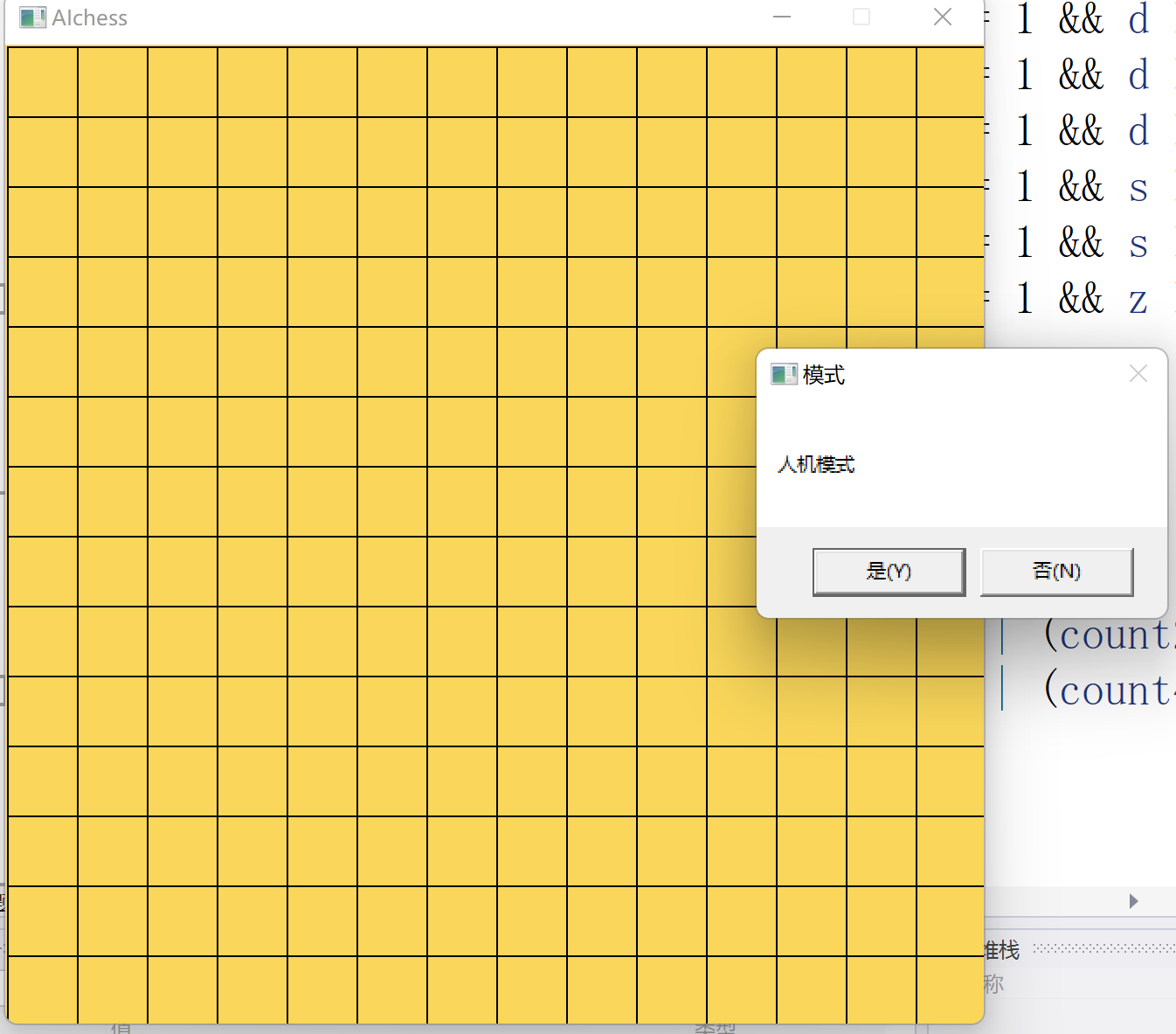
AIplace(void);

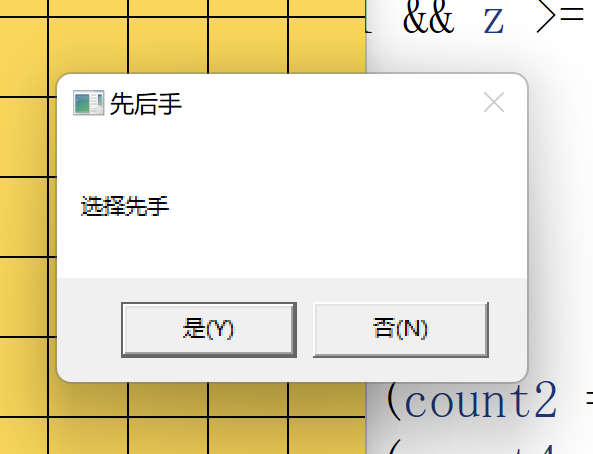
//非常可惜的是，本人的水平太烂，那个分成好几个cpp文件的模块化怎么都搞不好，所以只能交这个高耦合作业/(ㄒoㄒ)/~~

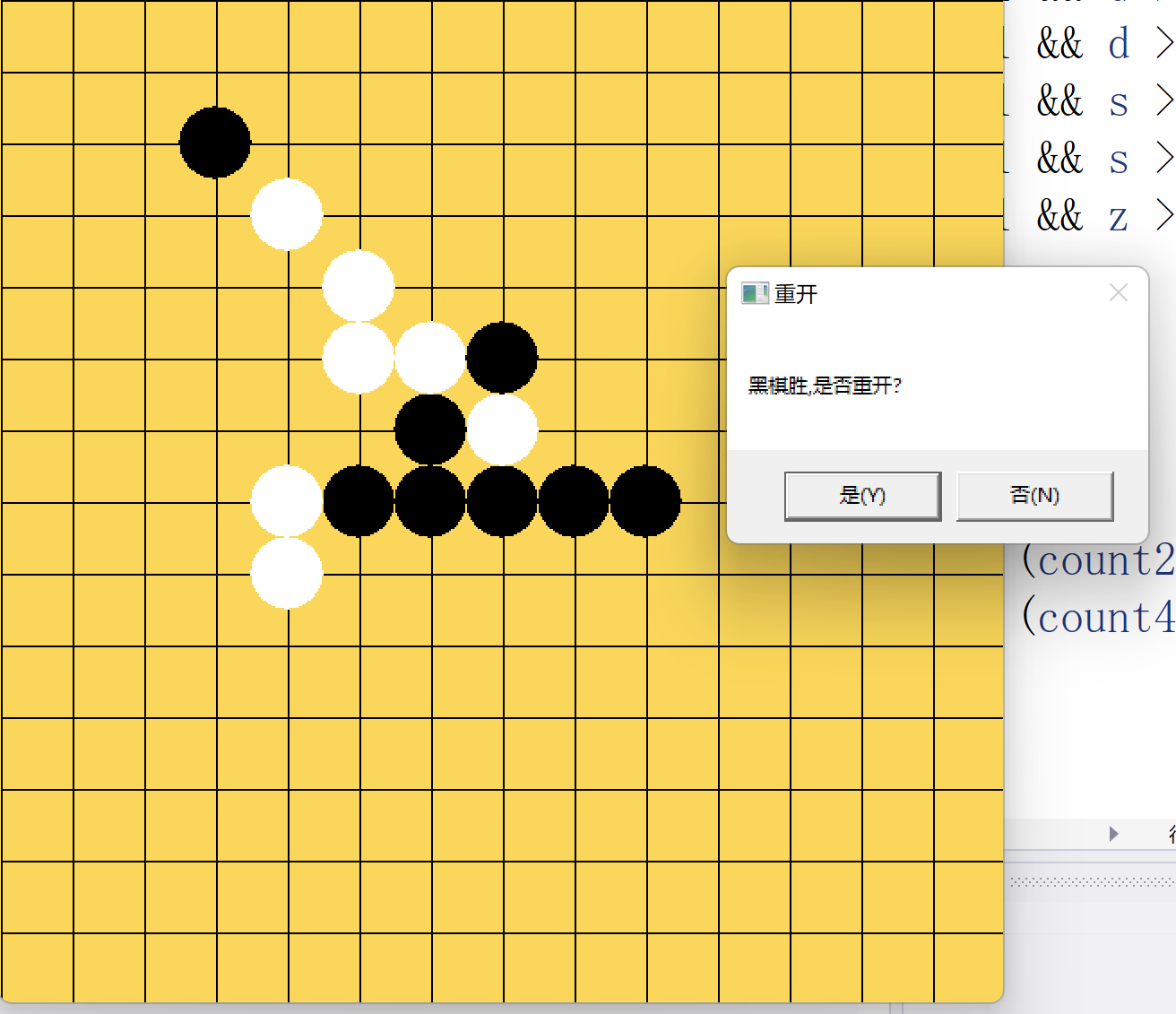
//在划分函数时我尽量按照各个模块作用进行划分，常用代码比如placePieces之类的都把它打包进一个函数了。但是各个函数之间仍然还是藕断丝连（？），引用来引用去的。。。

1. 关键代码

我认为是这个judgeScore判断五子棋场上局势以做出落子点判断的函数。其中，用WASD这个游戏常用键区的八个字母代表模拟棋盘的移动方向，count代表横竖斜各个方向的分值，然后进行各个点统计每个方向的count。通过count来为处于双活四、活四等各个情况下的点进行赋分。

1. 运行情况
2. 界面





1. 运行速度和程序大小

很快，因为这个程序比较简单/(ㄒoㄒ)/~~

1. 棋力

不怎么样，还在学习中的剪枝跟着两百多个Bug遥遥无期了（搜了一下要搞到四层普通人才能下赢，我的程序搞到四层以后AI直接下不出来卡住了）。只能说，这个AI只能战胜不怎么下棋的菜鸡。

但是因为我一直跟它下棋，改了好几次赋分的数据，所以也不是太离谱

1. 参考文献

C++超详细五子棋游戏（AI实现人机对弈+双人对弈+EasyX图形化界面+详细介绍）

【c语言五子棋】简单ai算法初步（实际）

好像还有好多，收藏夹里面找不到了