

C#程序设计及应用第四次作业暨期末大作业报告

多功能多重对战方式五子棋游戏

1000010284 管毓清

一、 作业要求.....	2
二、 程序简介.....	3
三、 程序界面.....	4
四、 功能实现.....	5
五、 程序测试.....	7
六、 提交文件.....	8
七、 个人介绍.....	8

一、作业要求

程序要求:

- 1)程序有较大的工作量(指自己实现的部分)或者有一定的难度或者比较有创意;
- 2)程序有较完整的功能或者具有较好的可扩充性。

在作业提交时,请注意:

- 1)程序要有适应的中文注释(英文注释不算),变量名/函数名/类名要合适,建议控件名使用匈牙利命名法。
- 2)程序要有较清晰的逻辑结构,注意应用的分层,注意类与类之间的关系,注意函数的简洁,必要时使用重构手段改进代码。
- 3)程序的基本功调试通过并能出结果才进行提交(请不要使用 WPF 或 silverLight,vs.net2010);最好附一个快速使用说明,有密码时要注明;如果使用数据库,则要提供相关的库、文件或创建库表的 sql 脚本。
- 4)可以写一个 readme 文件,描述一下程序的功能,程序的特点,程序的模块,所用到的技术,以及尚未完成的功能(或者你认为可以扩展的方面),如果程序中参考了别人框架或程序,请注明,并注明那部分是你添加或修改的。如果是一个小组进行开发,一般不超过 3 人,并注明分工的情况。请留下联系方式,以便有问题时可以联系到你。可以写上你开发程序的体会。

二、程序简介

本程序实现了图形化的无禁手五子棋程序，同时内含通过极大极小值搜索实现的 AI，实现了服务端/客户端网络对战功能，支持存档和读取游戏，支持每局比赛结束后保存录像，播放（通过读取存档功能），播放录像可以调节播放速度或者暂停（将播放速度调为 0s/步自动暂停）。

我一直想写一个图形化的五子棋程序，去年我参加了院内 Botzone 网站举办的五子棋争霸赛，当时我的 AI 取得了较好的成绩。当时由于我技术有限，只做了一个控制台的外壳。本学期学习唐老师的 C#课后初步掌握了一些图形化程序编写的方法，所以我想实现图形化的五子棋程序功能。与此同时，为了纪念 Botzone 的比赛，我把我的 AI 玩家默认命名为 **Cheat**（当时我参赛时用的名称），人类玩家默认命名为 **Wang Cai**（Botzone 网站默认参赛名称）。

有点遗憾的是，我没有完全将 Botzone 参赛的原 AI 完全移植到 C#程序中，一方面由于 C#相比 C++来说可能慢了一些，我的 C++程序可以在 1s 中内推算未来 20 步的落子情况，但是 C#这样做太慢了要十几秒，所以我不得不缩小推算层数为 9 层（不过搜索层数仍为 6 层）。另外，我去掉了 C++的 AI 用到的五子棋终结者的先手必胜谱，我一直没有研究透那个棋谱的结构，所以当时读谱的时候就有错误，而且这个谱只帮我多赢了两场先手。我不想为了棋谱破坏这个程序的完整性，而且目前的 AI 水平也还可以，我很难下赢它的先手，但也可能是我很久没下过棋艺退步了。

注：

- 1.保存功能仅在非网络对战时支持保存，否则若网络对战中途存盘，读取存档后还要重新建立连接非常麻烦，而且常见的五子棋网络对战游戏也不提供保存下到一般的残局的功能；
- 2.保存功能仅在当前玩家为人类时可以保存，如果当前玩家 AI 因为 AI 每步落子不会超过 3 秒钟，所以等 AI 落子后人类可以保存；
- 3.网络对战时白子方为服务端，黑子方为客户端，必须先建立服务端，对战通过本机的 2255 端口，可能被防火墙拦截；
- 4.黑子方和白子方不能同时设为网络玩家。

三、程序界面



(主界面图，这是一次黑白双方均为 AI 的对战结果)

- 1.主界面：左边为棋盘，右边从上往下分别为用户栏、按钮栏、落子历史栏、录像播放速度栏；
- 2.棋盘：棋局进行时，如果当前是本地人类玩家，点击棋盘可以落子，棋局已有子情况可以在棋盘中显示，选中落子历史栏，高亮的棋子会在棋盘中以红色十字标出；
- 3.用户栏：包括黑色玩家和白色玩家栏，每个玩家都可以选是电脑，本地的人类，网络端，选中人类玩家可以输入名字，选择网络玩家必须输入对方 IP 地址；
- 4.按钮栏：第一个按钮会根据游戏情况显示“开始新局”、“投子认输”或“断开连接”、“停止录像”，第二个按钮在未进行游戏时为“读取游戏”，游戏进行时为“存储游戏”；
- 5.落子历史栏：显示当前或之前刚结束的棋局的落子情况，选中一个棋子可以在左边的棋盘中标志出它；
- 6.录像播放速度栏：只有播放棋局录像时启用，调节播放速度，或者将拉杆拉到最左端暂停播放。

四、功能实现

为了实现本程序，我自定义了 13 个类，并且在原有的窗体 frmMain 类中加入了很多代码，但是与之前程序不同的是，我没有修改 Program 静态类的内容，由于我自定义的 13 个类中有两个类仅仅作为 AI 棋子估值的比较器使用，所以下面主要介绍一下自定义的 11 个类和 frmMain 类的一些情况。

A. 棋子类型

1. AbstractStone 类：抽象类，所有棋子类型的基类，提供了根据当前棋盘、玩家颜色、所在坐标的构造方法，实现了落子 Down，提子 Up 方法；

2. RealStone 类：AbstractStone 类的派生类，提供了初始化棋子图像的静态方法 Init，提供了根据当前棋盘、玩家、坐标的构造方法，实现了在棋盘上绘出棋子 Draw 和被选中棋子 DrawHighlight、转换为字符串供 ListBox 显示 ToString、保存到二进制文件 Save 的方法；

3. VirtualStone 类：AbstractStone 类的派生类，AIPlayer 内的封闭类，使用了 IComparable 接口，构造函数中会根据当前棋盘清空得到进攻得分和攻防综合得分，按照比较方式不同的方式，可以按照纯进攻得分排序或攻防综合得分排序，内含 StoneComparer 和 StoneComparerWithDefence 比较器类型。

B. 玩家类型

4. AbstractPlayer 类：抽象类，所有玩家（选手）类的公共基类，根据当前游戏，选手颜色、选手名实现构造函数，提供鼠标点击棋盘 MouseDown、当前玩家开始准备落子 Play、游戏结束后续处理 Close 虚函数，转换为字符串 ToString；

5.HumanPlayer 类：人类玩家类，AbstractPlayer 类的派生类，NetPlayer 类基类，实现了点击棋盘落子方法 MouseDown;

6.NetPlayer 类：网络玩家类，HumanPlayer 类派生类，构造函数不需要用户名而是需要对方 IP 地址，构造时启动 Init 函数线程；初始化函数为 Init，通过 Socket 技术，如果网络玩家为黑子，本地建立服务端，否则建立客户端，保证了必然是黑棋与白棋建立连接。连接时需要服务端先启动，另一方再启动客户端；Input 函数通过 Lambda 表达式启动 Put 函数的新线程给对方发送本地落子，覆盖基类的 Play 函数创建 Get 函数的线程获取网络端反馈来的落子情况。Error 函数统一处理网络出错时的情况，Close 函数负责关闭网络创建/获取/发送的线程，覆盖了 HumanPlayer 类的 MouseDown 函数避免网络玩家下棋时仍然可以点棋盘落子。

7.VideoPlayer 类：录像播放器类，AbstractPlayer 派生类，覆盖基类的 Play 函数，将保存的棋子记录的第一个棋子出队列下到棋盘上；

8.AIPlayer 类：计算机玩家类，AbstractPlayer 派生类，默认命名玩家为 Cheat,内部有 VirtualStone 封闭类，Available 函数检测是否当前位置可以落子（距离棋盘有子位置是否太远），覆盖基类 Play 函数，创建 Computer 函数的线程进行 AI 计算，Close 函数关闭创建的线程。

Computer 是 AI 落子函数，Search 是极大极小值搜索函数，如果电脑为黑子第一个会占住天元（棋盘几何中心），否则通过搜索解决，搜索深度为 6 层，搜索完后会继续最优情况推演到第 9 步，统计出最好的子，下到棋盘上。

由于极大极小值搜索比较复杂，AI 算法也不在 C#课要求内，在这里略去相关的内容，请见谅。

C.自定义静态类

9.Constants 类：静态类，包含了程序所有需要的常量；

10.Analyzer 类，静态类，分析盘面情况，GetChess 方法获取当前棋盘上某坐标的棋子，避免数组越界，AnalyzeLine 针对以某棋子为中心的行、列、对角线，判断棋型，GetValue 给出棋子或棋子坐标，根据 AnalyzeLine 函数提供的棋型，求得棋子的估值情况；

D.游戏核心类

11.Game 类，游戏类，提供构造函数（参数为主窗体对象），清空游戏函数 Clear，开始游戏函数 Start，当前玩家放弃函数 GiveUp，游戏关闭后续处理函数 Close（如网络对战在这时关闭所有 Socket），落子函数 Down，鼠标点击棋盘函数 MouseDown 调用玩家对象的 MouseDown 函数，读写存档函数 Load/Save 读写二进制存档文件，棋盘重绘绘图函数 Draw 枚举列表框中所有棋子调用它们的 Draw 和 DrawHighlight，播放录像时播放录像的 timerVideo 的 Tick 事件注册函数 VideoTicker 调用了录像对象的 Play 函数；

12.frmMain 类，主窗体类，自己实现的函数都在 frmMain.cs 内，游戏中关闭窗体处理函数 frmMain_FormClosing，在网络对战时相当于弃权，开始游戏 StartGame 结束游戏函数 StopGame，文件打开保存函数 LoadGame,SaveGame，还有一些和控件属性相关的函数，在此就不再一一赘述。

五、程序测试

一开始程序有些小的 bug，经修正后解决。网络对战时，由于 WINDOWS 防火墙一开始不经提示直接拦截了本程序（但后来又莫名奇妙好了），导致不能连上，后来防火墙弹出提示，允许访问网络后问题解决。经过多次测试没有再发现问题，但由于时间比较紧的缘故（放假后一直留京上 GRE，每天只有下午晚上可以写程序），也许会有一些疏漏没有被发现。

六、提交文件

提交的是 VC#2008Express 创建的整个文件夹，其中

项目文件：	\FiveStones\FiveStones.sln
源代码：	\FiveStones\FiveStones*.cs
编译好的程序（Debug 版）：	\FiveStones\FiveStones\bin\Debug\FiveStones.exe
一个录像文件（AI vs AI）	\FiveStones\FiveStones\bin\Debug\untitled.ren
说明文件：	\FiveStones\期末大作业报告.pdf

七、个人介绍

我是信息科学技术学院的一名大二学生，编程基础可能在年级算是中等偏上吧，高中时被老师顺带着参加过 NOIP 的比赛，虽然有省一，但也是省内三十多名的样子，而高中阶段也确实没有那么多的时间用来写程序。上大学后选择了计算机专业，但是在学本门课之前，一直没有接触过太多图形化编程的东西。C#对面向对象实现得非常彻底，这有助于拓宽我们对面向对象程序的思路，而且在这个课上，我接触到了微软的 .Net 技术，第一次尝试了编写可视化程序、网络程序，无论是唐老师教的还是自己自学的都学到了很多新鲜的内容，而且我还把这学期老师教的 .Net 技术用在了我们专业课数算实习的大作业上，为我们的大作业写了一个 CLI C++ 的界面。我觉得这门课是我这学期收获最大的一门课，而且实现各种自己感兴趣的程序是一件非常有意思的事情。在这里，我想感谢老师和助教，希望以后还能有机会上这种有趣又有益的课！

祝唐老师和赵助教寒假愉快，新春快乐！

学生
管毓清
1000010284
2012 年 1 月 14 日