Lab2说明文档

1. 类结构及相应位置使用的oop知识
2. 棋盘

属性：

1. 棋盘上所有可以落子的位置构成的二维数组（private，对外信息隐藏）
2. 取胜者（玩家/电脑/平局/游戏未结束）

主要方法：

1. 初始化棋盘（被主类调用）
2. 打印棋盘（private，对外信息隐藏）
3. 落子（private，对外信息隐藏），调用本类的B）方法以及玩家类和电脑类的获取落子列方法
4. 判断游戏是否结束（private，对外信息隐藏）
5. 封装C）和D）方法（实现封装，被主类调用）
6. 棋子

属性：玩家/电脑

主要方法：

获取落子列（由于玩家和电脑都需要这一方法而具体实现方式不同，该方法被子类重写，实现多态）

1. 玩家：继承棋子类，重写父类的获取落子类方法
2. 电脑：继承棋子类，重写父类的获取落子类方法
3. 输入：读入玩家输入，被玩家类中获取落子列方法调用
4. 输出：将输出写入文件
5. 游戏（主类）：调用棋盘、棋子、输出类的方法，实现四连环游戏

注：只有游戏类、输入类以及输出类对外可见（public）

1. 设计思路（oop知识使用原因）：

四连环游戏的主要操作对象是棋盘与两种棋子，因此我分别建立了棋盘类和棋子类。由于两种棋子操作不完全相同，我将玩家与电脑的棋子设为棋子类的两个子类，这样它们既可以继承父类的相同方法，又可以通过多态实现部分方法内容的不同。其中我对不必须被其他类调用的方法与属性设置为private，实现信息隐藏，对必须被外部类访问的方法进行封装，以减少外部类访问该类的次数。棋盘与棋子类对外不可见，通过被主类访问来实现其功能。由于输入和输出必须对外可见，不能写入棋盘和棋子类，所以我将它们单独设置了两个类，实现外部访问。