

Pre1 Lab代码与演进过程分析

1. 报告及讨论的内容：

本学期至今共完成了多个连续的下棋游戏的lab（Lab2~5），每次是一个功能的增量，相信同学们实现要求的功能都没有问题，但是设计得比较好的程序在增量时会比较容易完成，否则会工作量比较大，甚至会对原有功能造成破坏。

本次讨论要求每个小组对自己组员的代码及演进过程进行分析，在共同讨论的基础上，通过示例分别以匿名的方式给出：

- 1.1. (Lab5) 组内评选出最好的代码，给出分析介绍，包括优点和不足。
- 1.2. (Lab5) 组内评选出改进余地比较大的代码，分析问题并给出可行的改进建议和理由。
- 1.3. 在版本Lab[n]到Lab[n+1]转换的过程中，有因为Lab[n]的结构不好，导致工作量比较大，但是运用了目前已经掌握的OOP的方法[包括抽象、封装、组合、继承、多态]，Lab[n+1]版结构有了明显的优化。在组员中找到这样一个或两个案例，给出分析和说明。
- 1.4. 【可选】在讨论(1.1, 1.2)的优点和不足的时候，可以考虑一下目前的设计对之后实现图形化界面有什么影响。

2. 报告过程

- 2.1. 每组选一个组长，负责协调讨论，协调制作PPT和负责课上展示报告。
- 2.2. 每组最长15分钟
- 2.3. 除了组长以外的其他同学在报告结束后需要提交一份简单的评价文档，选出（除了自己组以外）：
 - 2.3.1. 代码最好的两个组(1.1部分)，并简单给出理由
 - 2.3.2. 1.2部分分析得最好的两个组，并简单给出理由
 - 2.3.3. 1.3部分分析得最好的两个组，并简单给出理由

组内对代码和设计的评价，如果组员之间存在无法协调的争议，也可以在Pre中做一个简单的说明。

3. 评分依据：

组员：

- 评分依据1：讨论部分（70%）。
 - 讨论评分的依据与选取组员代码的质量关联度较小，重点关注报告的条理性和分析的内容。

- 评分依据2：讨论结束后提交的2.3的评价文档（30%）

组长：

- 10分起步
- 讨论及报告部分（90%）

4. 相关时间节点

4月29日上课时间讨论。各组组长需在**4月28日晚上22:00之前(周一晚)**在e-learning上提交讨论PPT。