

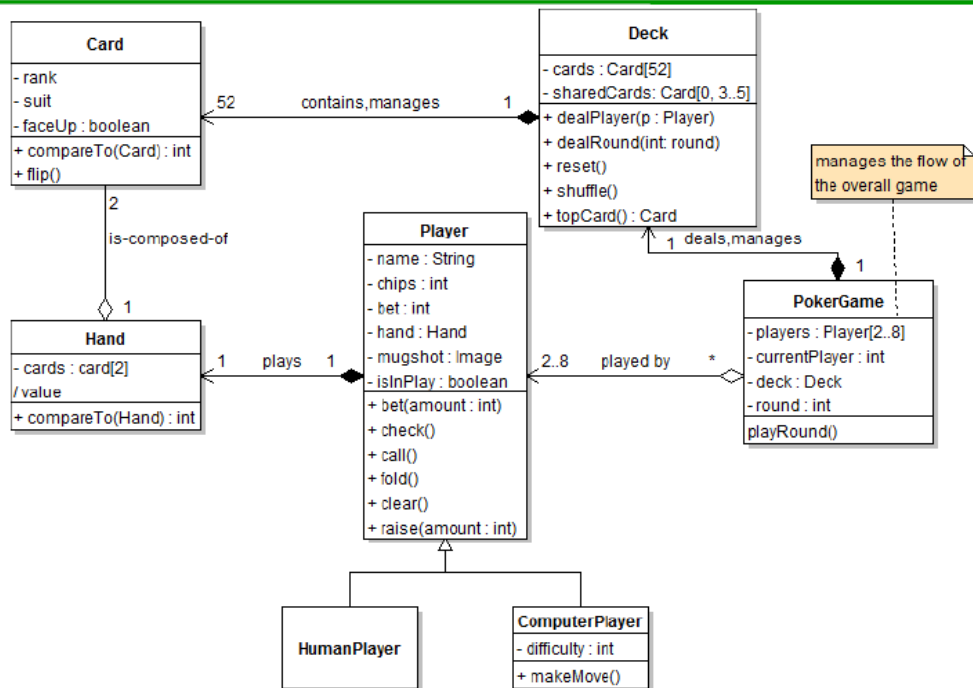
Project 补充说明

两个 project 都要求实现人、机间的相互博弈，在操作界面中应可自由选择两人互玩、人机对战或者是两个 NPC 对战，并任意指定先后手。

本次 project 主要考察面向对象编程中类的封装，继承和多态等知识点，因此要求使用面向对象方法设计程序，请不要使用面向过程的程序设计方法，也不要简单地只定义一个类完成所有工作。程序设计时要求实现 `game`, `player`, `playerHuman`, `playerComputer` 以及 `board/digits` 这些类，但不必局限于此。程序结构设计可参考课件中扑克类的设计：

Software Design II – C++ Polymorphism

Poker Class Diagram



April 18, 2016

SUN YAT-SEN UNIVERSITY

3 / 75

封装：需要对游戏中的数据进行相应的封装保护。

在 Project A 井字棋中的棋盘数据可以被 `player` 读取并放置下一步棋，但 `player` 不应该有权限对棋盘进行随意改动。

在 Project B 猜数字中的数字不能被 `player` 所直接获取，但需要对每次 `player` 的猜测做出反馈。

继承：`playerHuman` 和 `playerComputer` 派生于 `player`。

多态：`game` 类只调用 `player`。`Player` 可以作为虚基类，通过纯虚函数实现多态

AI 设计：两个题目的 AI 设计都不做要求，但必须实现 `player` 类的派生类 `playerComputer`。当然完全随机的策略也是一种 AI 的实现方法，但实现较优的 AI 策略将会获得一定的附加分。例如 Project A 井字棋中的最有策略已经在提示中给出，它可以保证后手不败。Project B 中猜数字的策略并没有给出任何提示，希望大家有所思考，也鼓励大家查阅资料以解决问题。

Project 的评估包括程序正确、界面友好以及文档报告三个方面。前两部分会在实验课上进行验收，文档报告要求大家上传到 FTP。

ftp://smie2.sysu.edu.cn

用户名: student0004

密码: student0004

评分标准:

程序正确（50%）：程序正确运行 25%，实现对数据的封装 15%，继承 10%。

界面友好（20%）：界面美观清晰，用户可在没有任何说明的帮助的情况下独立使用为标准。

文档报告（30%）：以报告清晰，能够阐明程序思路为标准，鼓励使用 UML。

相关信息：http://smie2.sysu.edu.cn/~hg/sdii_smie151/