# 武汉大学2023级计算机学院期末项目文档

## —文字三国杀（对战机器人版）

## 计算机学院周锦耀2023302111048

目录：

1. [功能](#标题1)
2. [项目细节](#标题2)
3. [测试用例](#标题3)
4. [日志](#标题4)
5. [反思](#标题5)

声明：该项目由本人独立完成，没有模仿，没有沿用任何平台、任何范例的代码。各种逻辑都是由本人独立设计，所以有很多地方可以优化的更好，但绝对是原创。参与测试人员为我以及浙江大学的一些同学们，他们提出了很多之前没有了解到的三国杀规则，非常感谢！特此说明。

提交方式为提供Github地址。

功能

本项目实现的是玩家和机器人进行三国杀卡牌游戏的功能。

整体过程可以分为：开始阶段(玩家和机器人选角色阶段)=>玩家游戏阶段=>机器人游戏阶段。注意：在每个阶段当任意一方血条归0时进入濒死阶段，不满足复活条件时将真正死亡。

项目细节

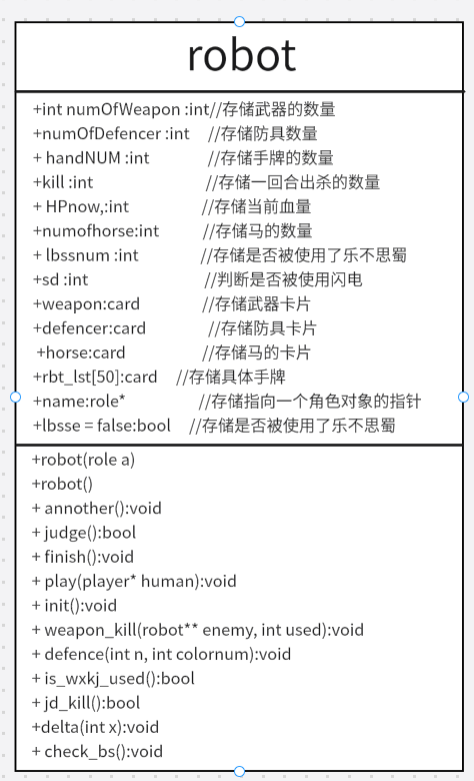
1）类的概述

本项目采用如下类：role，player，card，robot。UML图:

role类用来存储基本角色，并保存在变量role\_lst中。

表格

描述已自动生成

 表格

描述已自动生成

2）主要文件说明：

包含main.cpp,player.cpp,robot.cpp,头文件：role.h,player\_robot.h和card.h分别存储role类定义，player,robot,card类的定义。main中则是手牌数组和牌堆队列的初始化，以及player，robot的play函数定义和实现，主函数main的实现。

3）主要功能实现原理和函数：

1.**如何管理卡牌（牌堆）？**

采用队列pd管理，有抽牌则用pd.front()和pd.pop()模拟抽牌，有使用牌（弃牌）使用pd.push()模拟弃牌入牌堆底部。

card\_lst\_init()：把每个卡牌逐一赋值给对象数组card\_lst中

pd\_init():把card\_lst的顺序打乱（接用random\_shuffle()函数），然后再借用STL中的队列数据结构，让每一张打乱后的卡片使用pd.push()进入对象队列pd中，于是就借用了pd来管理牌堆。

2**.如何管理角色行为？**

首先，玩家（player）中用player\_lst管理已拥有的手牌。接着，每回合开始前从牌堆顶摸两张牌（管理方法见1），然后根据变量lbssnum/sd判断是否有乐不思蜀/闪电，有则从牌堆顶摸牌进行判定。根据annother函数和lbsse来判断是否要进行对应行为。闪电和乐不思蜀具体实现逻辑见后文。然后进入角色出牌阶段，每次有效出牌之后，都将把该牌放入牌堆底部并且在对应的手牌数组内，从后往前按序覆盖。当输入为4时进入另一方的回合。以上行为（包括判定阶段/结束阶段）对双方均适用。

**3.机器人的实现逻辑？**

机器人采用尽可能把牌出完的原则，按照先出非桃/杀牌，最后出桃/杀牌的逻辑实现，实现部分最优。然后没有可执行手牌时，退出出牌阶段。

**4.什么时候判定角色死亡？濒死和死亡逻辑实现？**

每个有可能造成伤害的操作之后，都将进行判断：通过两个类中的judge()函数来判断，当进入judge函数时，输出：“已经进入濒死状态！”，通过check\_bs()函数实现是否能够复活（桃的数量能否让角色恢复到1血），并输出状态：死亡/成功复活。

**5.角色技能实现？**

郭嘉：在每个可能受到伤害的地方调用判断函数，来实现技能的发动。注意：**这里有一个很复杂的点：当瞬间进入濒死状态时，需要判断能不能救活，如果能救活才能够发动技能，这里摸牌数量相当于之前扣除的血量\*2。就是说如果角色无法从濒死进入残存1血状态，那么无法发动郭嘉技能，这其实将与之前的逻辑有些冲突，具体实现代码如下：**

**文本

描述已自动生成**

**利用类里的delta函数和get\_num\_before\_bs参数及check\_bs()里的**

**文本

描述已自动生成**

**部分实现，相当于巧妙利用两次回调，实现代码复用，有效减少代码量。见下图**

**图示

描述已自动生成**

**黄月英技能实现？**在确定为有效出锦囊牌（非长效锦囊牌）的时候，将直接摸一张牌。注意：这里考虑的情况也很多：例如当主动出无懈可击时容易不弃牌但触发技能；满血出桃园结义同上；对面没牌出顺手牵羊、过河拆桥同理。这些写着也比较麻烦，容易漏情况。

**6.怎么发动武器技能？在对面有防具的时候两个逻辑冲突怎么解决？**

这里我认为比较有意思的就是把出杀和出闪两个单拎出来写。利用Weapon\_kill和defence函数。Weapon\_kill传入的参数是对面的实例的指针的地址，然后用对应的指针指向该实例的指针，还有就是传入的使用杀手牌的编号（特殊情况见“借刀杀人”）。如果没有装备武器（numofweapon==0）那么就直接让对方出闪；如果装备武器，那么就使用weapon\_kill再根据对应的武器使用对应的效果。实现杀闪分离，减少代码量。**同时，比较有意思的就是defence传入的参数有一个int n，这是用来辨别防具用的，如果使用的武器是青钢剑（无视防具），那么传入0，然后如果这个和防具数量有一个为0，就无视防具。这样巧妙实现了到底要不要用防具的效果，以及各种武器的效果。**

**7.各种手牌是怎么使用的？**

**类别1：武器/防具/马。那么就是直接修改数量=1和对应的牌变量，输出“您装备了xxx”即可。**

**类别2：杀/闪/桃：杀一回合没有特殊情况只能用一次，所以添加kill参数；不能主动出闪；满血不能用桃。**

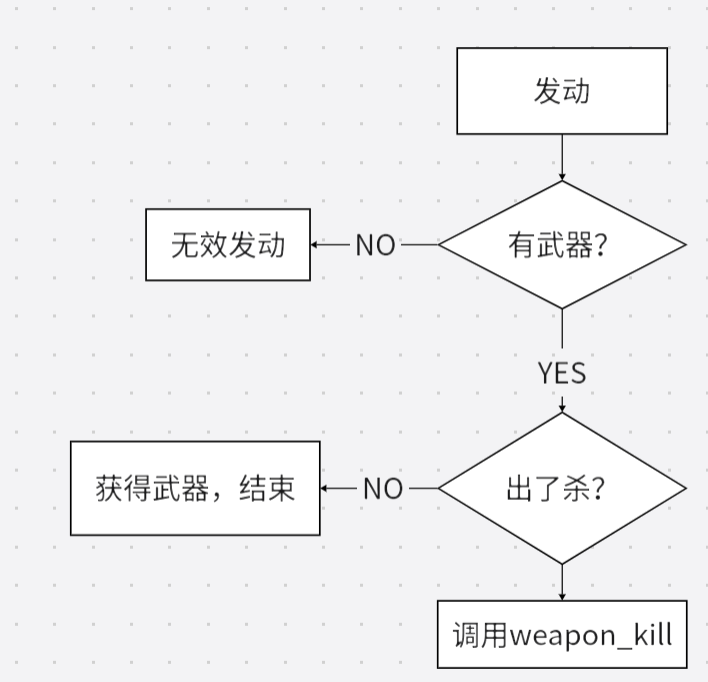
**类别3：锦囊：这里就是比较复杂了，按照各种种类来实现。**

**我就举几个比较难实现的逻辑。**

**1.乐不思蜀：**之前想过很多方案，但是都很难实现，**后面发现把它与回合开始函数融合在一起就行，先修改当前的lbssnum为1，表示判定区已经有了乐不思蜀。会在开始判断，如果结果不为红桃，那么跳过，修改lbsse为true，跳过出牌阶段，进入结束阶段后再把两个变量初始化就行。**

**2.无懈可击：**不能直接主动用，只有在对方使用锦囊牌（非无懈可击）的时候才能使用无懈可击，用于抵消对面发动的技能效果。所以我创建了一个bool类型的wxkj\_used方法，当使用锦囊牌的时候，**询问对方是否使用无懈可击，如果使用，那么返回true，抵消锦囊发动。**独立出来可以更好实现功能。

**3.借刀杀人。这个是最麻烦的一个函数。**先判断是否有效发动：对面是否装备武器。如果装备且有杀那么出杀**，最麻烦的是对方出杀的同时调用武器的技能，并且要获取出杀的花色。所以我用pd.back()的花色来实现。**

****

以上就是重点注意的部分函数如何实现的说明。

测试用例

开始界面如图所示：

图片包含 图形用户界面

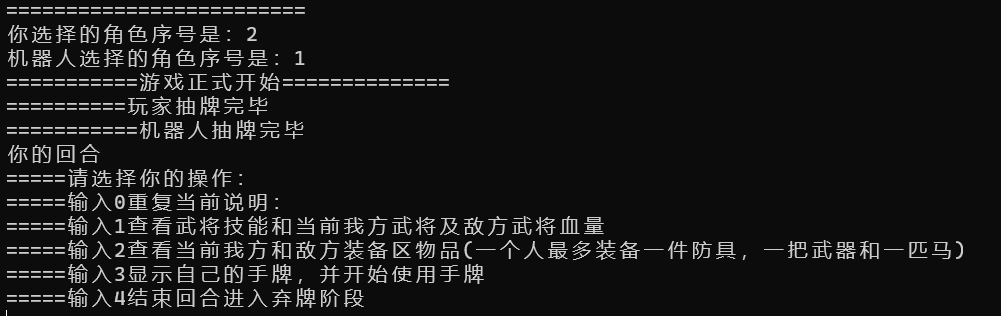
描述已自动生成

玩家按1即可进入选角色界面

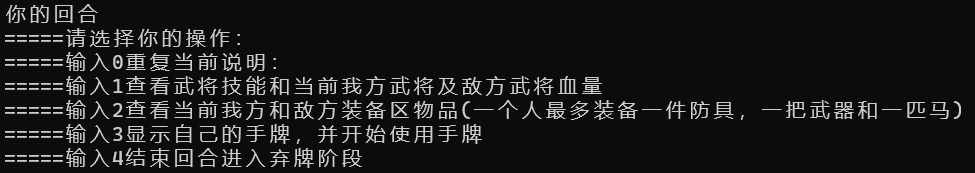
文本

描述已自动生成

玩家按1初始化手牌的同时可以在两个角色中自主选择，选好了之后机器人将会随机选一个角色，并开始游戏。



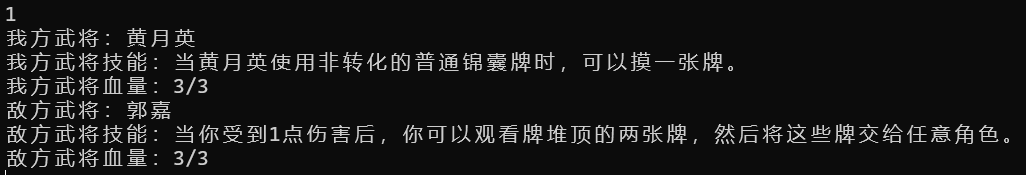
选好角色后将会进入玩家的出牌阶段



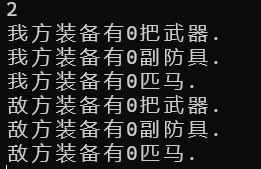
根据提示：

按0会重复当前说明；

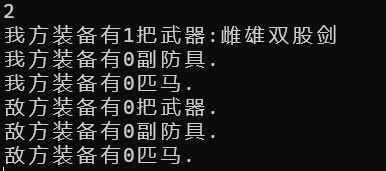
按1将查看武将技能和武将的血量



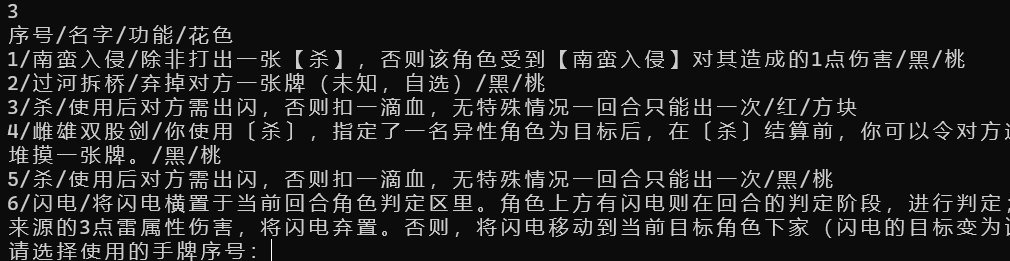
按2将查看当前我方和敌方装备区物品



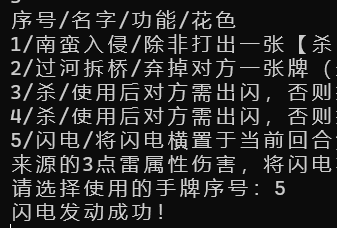
当装备了武器/防具/马的时候，会显示数目及名字



按3可以展示自己的手牌，包含序号/名字/功能/花色四种属性。并提示出牌序号。当输入出牌序号且在范围内时会出牌。



例如：当输入序号4：出现如下：

当输入闪电的序号：等等等等

出完牌之后可以选择敲入操作数4，来结束自己的回合，进入机器人的回合。

机器人将自动出牌并需要你响应。（完整测试用例在“测试用例.txt”）

日志

总共耗时约15天，总代码超2k行。

该项目大致时间分配如下图所示：

图表, 瀑布图

描述已自动生成

反思

本次项目做起来颇有挑战性，给我的逻辑设计增加了很多经验，对我而言是一次学以致用的好机会，极大地锻炼了我设计项目、独立思考和物化项目的能力。同时，我也注意到有一些类、一些方法可以设置的更简洁，让我在以后的设计中会考虑得更加充分。