《面向对象程序设计与训练》实验报告

姓名 ***Steven*** 学号

实验名称 人机猜拳

实验成绩

实验时间 2021 年 3 月 29 日

实验二 人机猜拳

一、实验目的

1.掌握通过对象使用类的属性及方法。

2.理解并应用使用get…()方法传值的意义。

3. 掌握Java程序的开发步骤。

4. 掌握在控制台界面下使用输入输出语句与用户交互。

5.思考并实现猜拳的胜负判断。

6.掌握生成随机数的方法。

二、实验仪器设备及软件

（1）

（2）

（3）

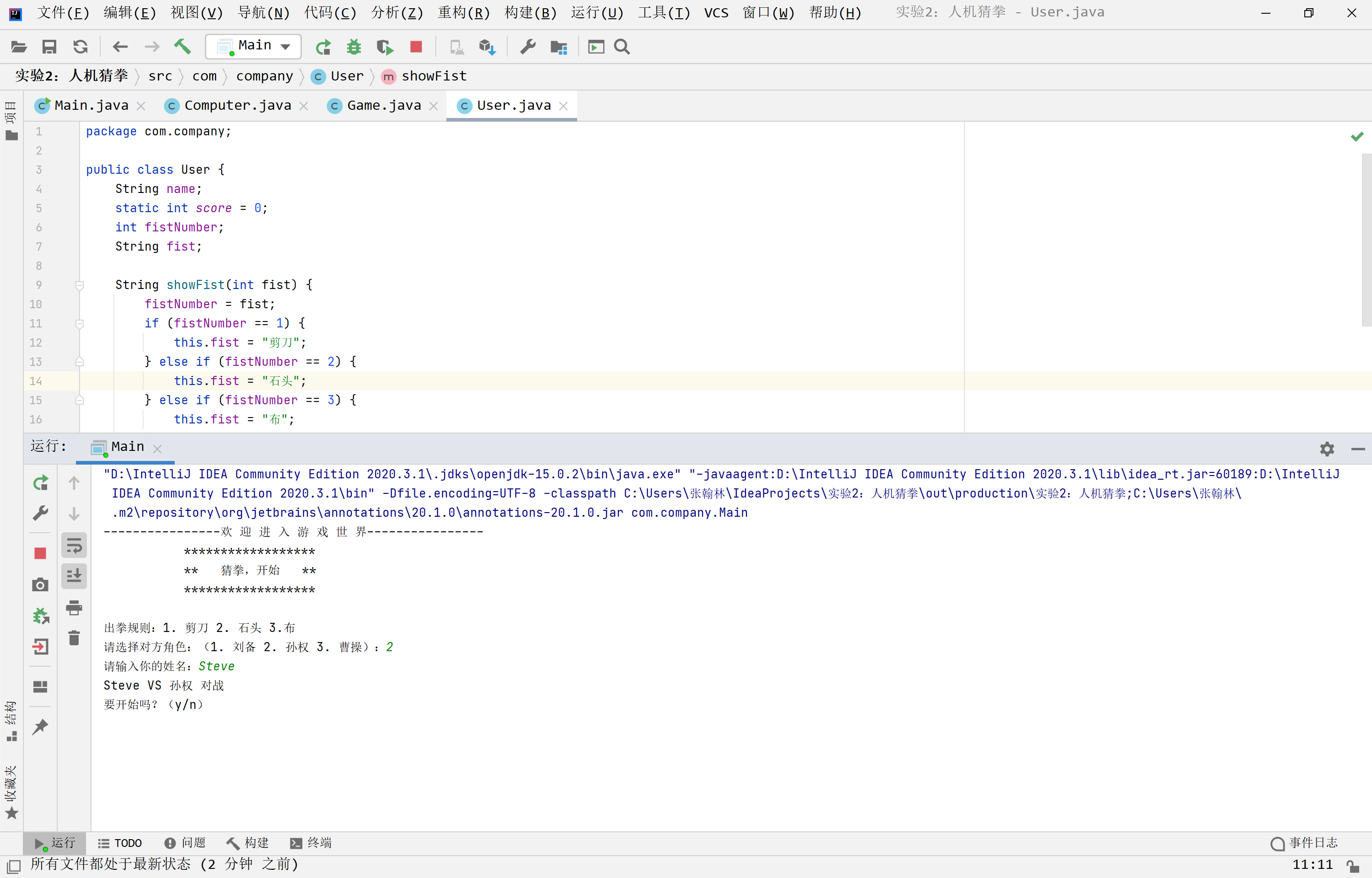
三、实验方案

人机猜拳包括如下的功能：

**1.实现菜单的输出显示**

输出菜单，告知用户如何进行交互以及代号对应的选项，接受用户对于对手的选择以及用户自定用户名。

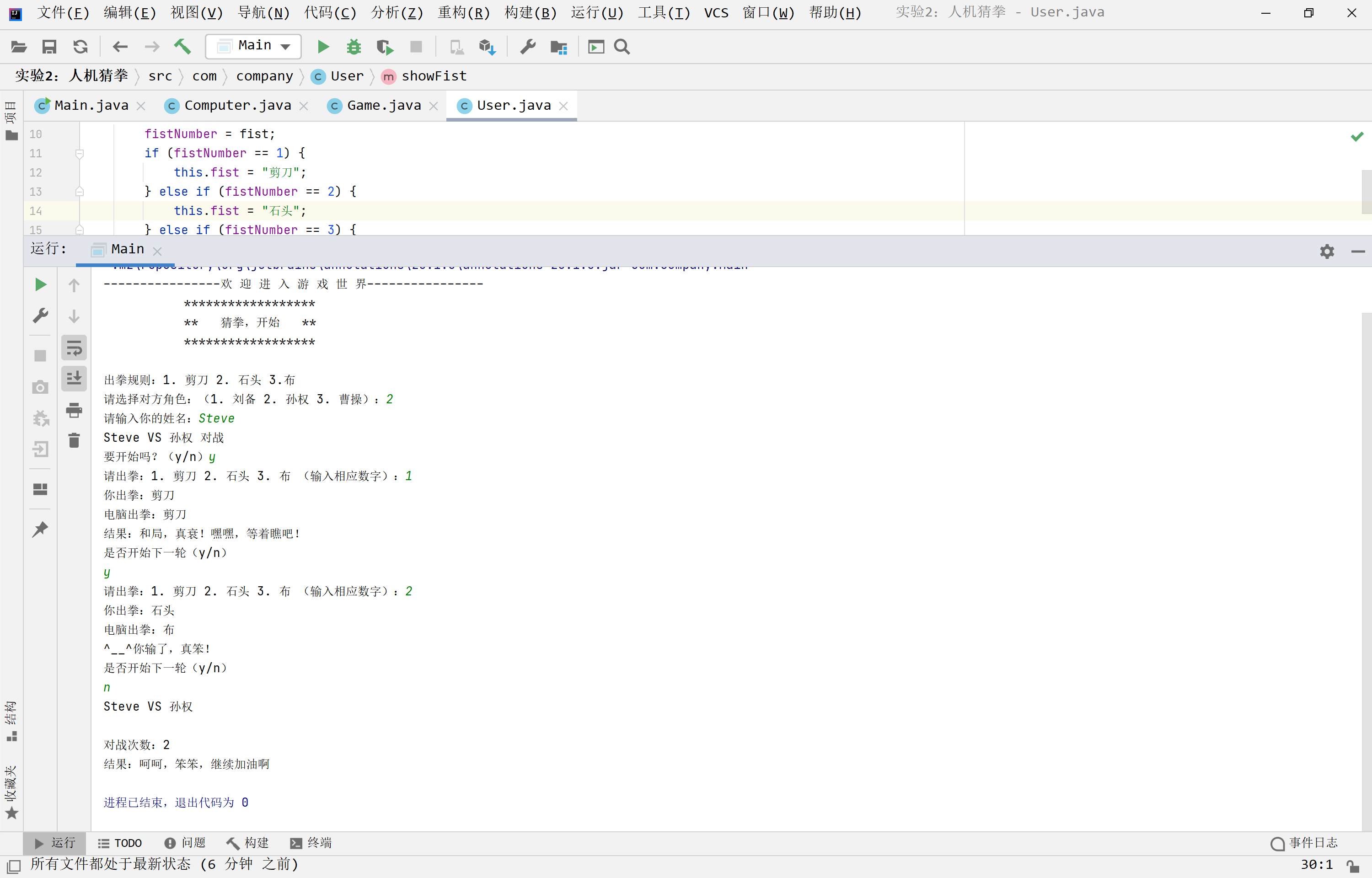
菜单的显示如下：



**2.实现循环执行功能**

系统询问用户是否开始，如果用户选择，则请求用户的出拳之后自动输出结果，并询问是否继续。若有一次用户选择否，则整个程序结束。

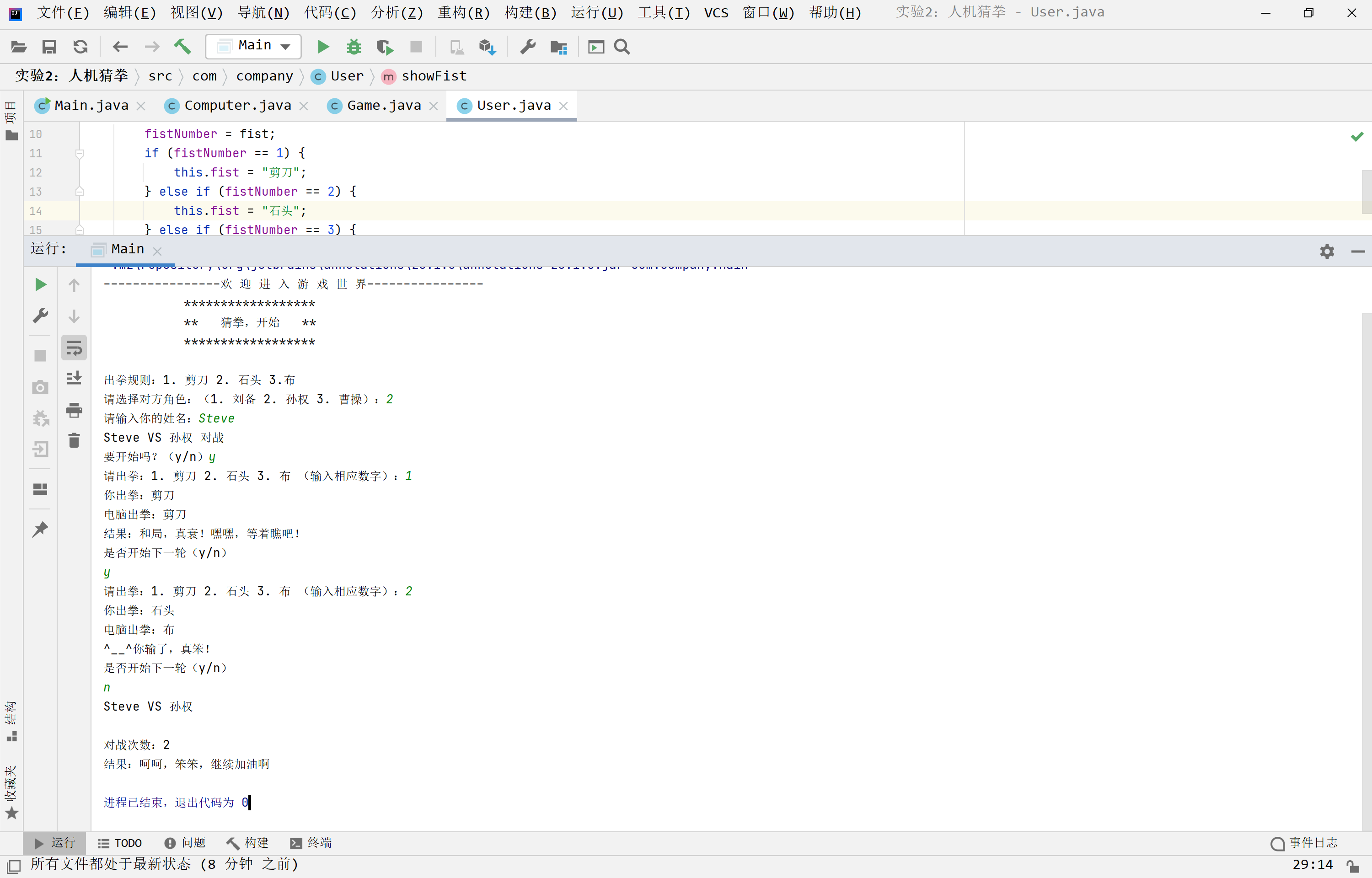
菜单的显示如下：



**3. 实现输出结果功能**

用户退出后，分别输出：用户名和电脑名、对局次数、对局结果。

菜单的显示如下：



四、实验步骤

**1.实现菜单的输入和输出**

菜单显示和执行相关的操作实质就是使用println打印菜单，提示用户操作，然后使用一个Scanner类的对象input三次，结合提示，分别接收用户选择的对手、用户名和是否开始。然后用if语句对是否开始的选项进行判断。

**2.实现循环执行功能**

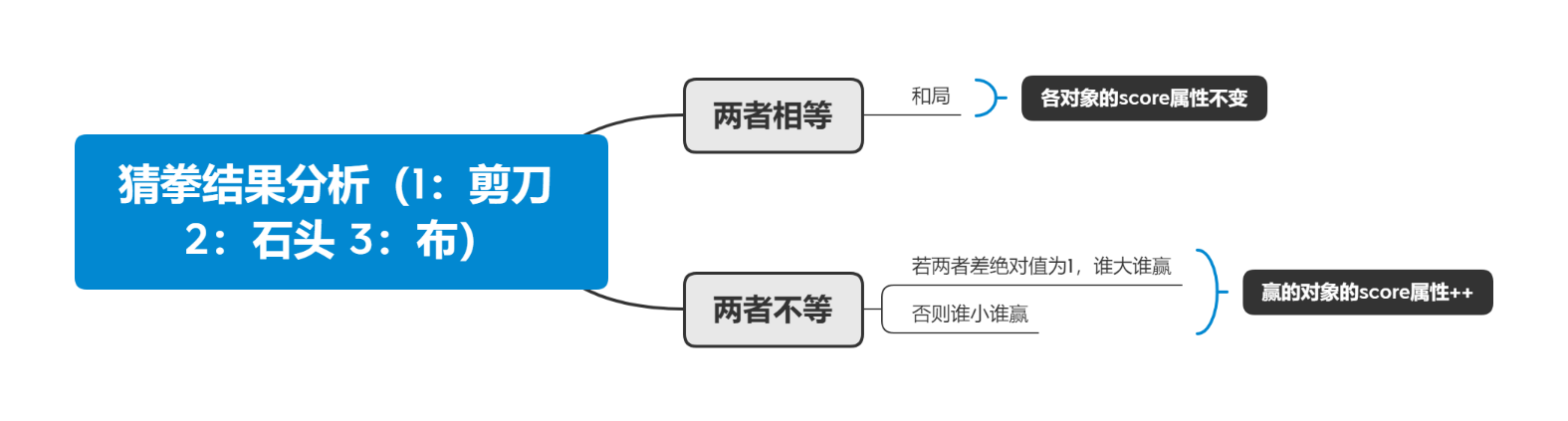
在显示菜单的代码外套一个while (true)循环，如果输入“y”则继续执行循环,如果输入“n”则退出循环。在每次猜拳结束后，对局次数++。

**3.实现计算机出拳功能**

每次在决定计算机的出拳的时候用fistNumber = 1 + (int) (Math.*random*() \* 3);语句生成一个范围在1~3的随机整数，分别代表3种出拳结果。

**4.实现猜拳结果分析功能**

每次猜拳之后将用户的输入和计算机生成的随机数进行比较（1：剪刀 2：石头 3：布），比较过程如下：

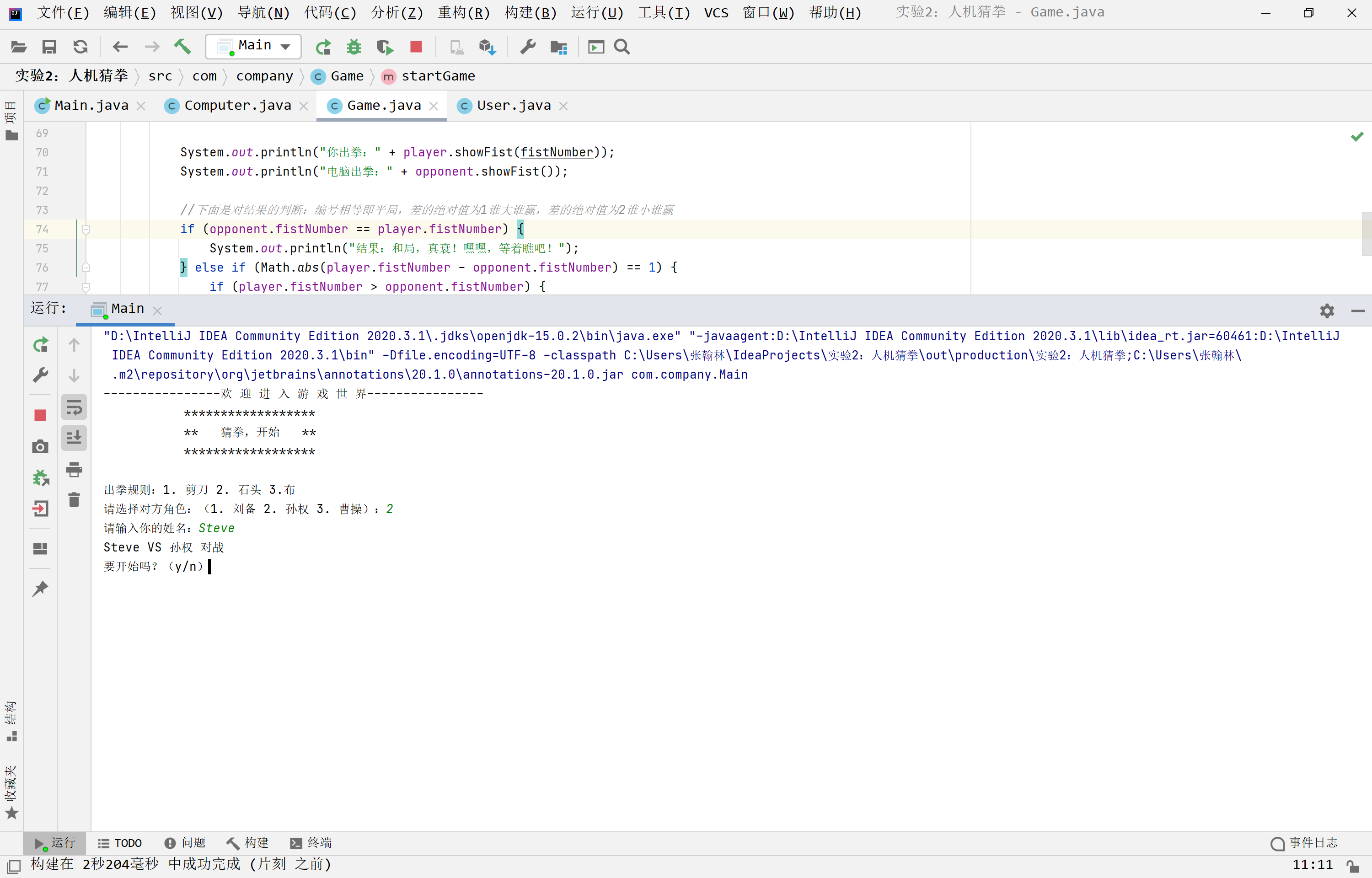


**（1.判定的大概思路）**

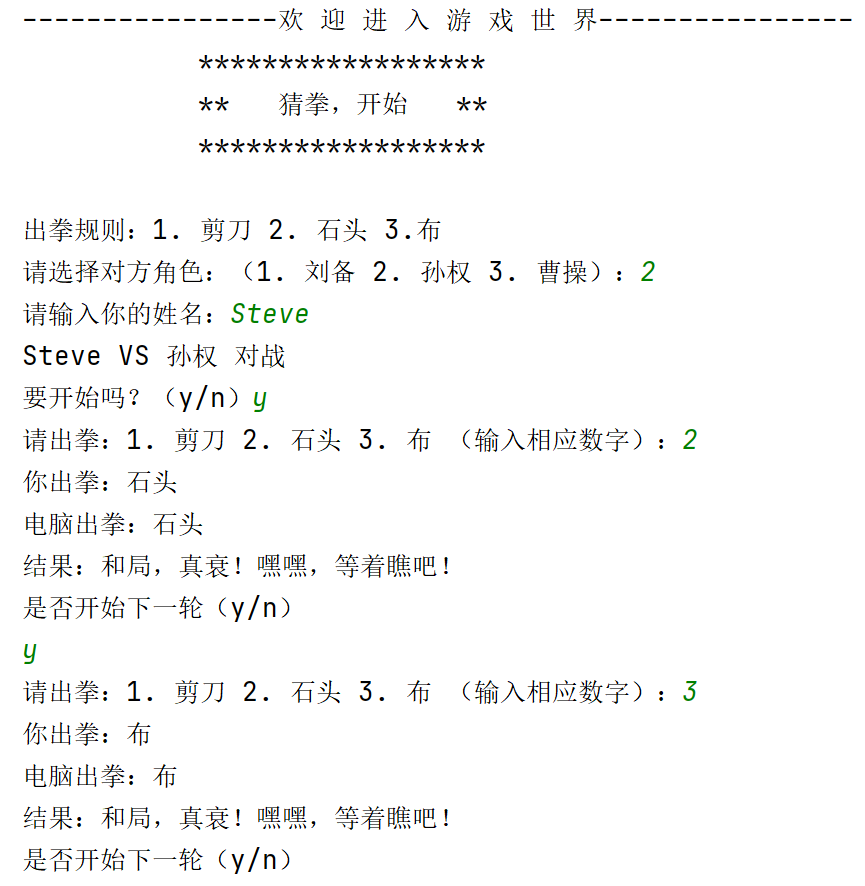
**5.实现最终结果输出功能**

按格式分别输出：双方的用户名、对局次数、并根据双方的分数确定谁胜谁负，以此为依据输出最终结果。

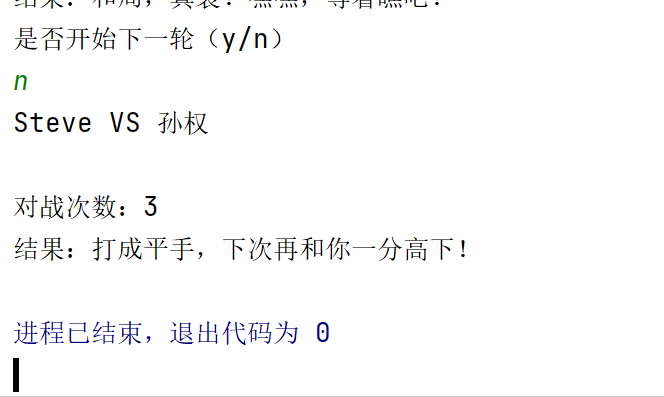
五、实验结果及分析



**（1.初始化对局双方示意图）**



**（2.循环进行猜拳示意图）**



**（3.结果输出示意图）**

1.按照预期结果，实验先是弹出了菜单，随后选择对方角色和设定用户名。

2.在初始化完成之后输入“y”将会开始游戏，输入“n”则会直接结束整个进程。

3.开始游戏后每一轮都会请求用户选择，并输出双方的出拳以及本局结果。输出完成后将会继续请求用户选择是否继续。选择方式和2.相同

4.这时再输入“n”之后将会输出本次若干轮猜拳后总的结果，并附带一句“祝福语”。然后结束整个进程。

**反思：**

这次实验中交互的部分初次设计欠佳，最大的问题就是用户选择的时候需要输入数字，但是如果不小心输入了一个其他字符就会引起报错导致整个程序终止。当然现在看来这个问题很简单，加一个if(hasNextInt())就能解决，但这并不意味着这是个小问题，以后在最初设计的时候需要考虑的完善一点，这样就能减少在测试阶段花费的时间。

此外询问是否继续的是不是有点太频繁了….测试的时候老是输入这个很累，于是在确定这一部分没问题之后就设置成循环10次自动退出，的确快了不少。

六、实验总结及体会

和上次问题基本一样，理解要求和流程之后用代码复现的流程太不连贯，其实本质上还是代码写的少的问题（只不过这次的要求相对来说更简单一些，所以花费时间比上次短）。此外还有随机数生成仍然不熟练的问题存在。

但此外我想夸自己一句，这次类倒是建的还可以，在调用的时候也熟练了不少，同时也沿袭了上次对于重复代码用类封装的习惯，使得Game类里面简洁了不少。

最后还有一个小问题，就是在debug的时候数据流总是在各个对象之间跳来跳去，并且会经过各种处理。现在类比较简单的时候还好，要是类复杂的时候我感觉理解成本会直线上升，很多时候我们并不能第一时间就反应过来某个方法是干什么的，某个属性代表什么。目前的做办法是多写注释，然后在较为复杂的方法中把代码按功能分块，尽量好理解一点，但我认为这并不是最优解。

七、教师评语