

# 廈門大學



## 信息学院软件工程系

### 《JAVA 程序设计》实验报告

实验 x

姓名：陈澄

学号：32420212202930

学院：信息学院

专业：软件工程专业

完成时间：

## 一、实验目的及要求

实验目的：练习控制结构

实验要求：按照题目要求写代码，撰写实验报告，并在下周实验课上课前将源代码和实验报告提交到 FTP。

## 二、实验题目及实现过程

实验题目：请按照游戏规则，编程实现：随机生成六个筛子点数并判断得奖情况，注意需要考虑到多个奖项的情况，比如四进带一秀。  
博饼游戏规则如下图所示。

	【一秀】即秀才 有一粒为4点
	【二举】即举人 有两粒为4点
	【四进】即进士 有四粒点数相同（4除外）
	【三红】即探花 有三粒的点数为4点
	【对堂】即榜眼 六粒骰子为123456
	【状元】 有四粒骰子为4点即是状元
	【四红】四个4
	【五子】五个2或3或5或6
	【五王】五个4
	【状元插金花】四个4、二个1
	【六勃红】六个4
	【六勃黑】六个2或3或5或6
	【遍地锦】六个1

奖项	常用名	图示	图解
状元	状元插金花		444411
	六博红		444444
	六博黑		666666-555555-3333 33-222222-111111
	五王		44444X X=6>5>3>2>1
	五子登科		66666X>55555X>333 33X>22222X>11111X
	状元		4444XX
榜眼	对堂		123456
	三红		444XXX
探花	四进		6666XX>5555XX>333 3XX>2222XX>1111XX
进士	二举		44XXXX
举人	一秀		4XXXXX
秀才			

**拓展题目：**

- (1) 提醒用户输入玩家数（6-10）。
- (2) 循环为每个玩家生成六个筛子点数（1-6），根据上图的规则判断所产生的骰子对应的奖项，并输出。
- (3) 游戏结束时（所有奖项已经出完），输出每个玩家所获得的奖项以及每个奖项的个数。

奖项	份数
状元	1
对堂	2
三红	4
四进	8
二举	16
一秀	32

（一）实验环境（集成开发环境、jdk 版本、字符编码等）

集成开发环境：IntelliJ IDEA

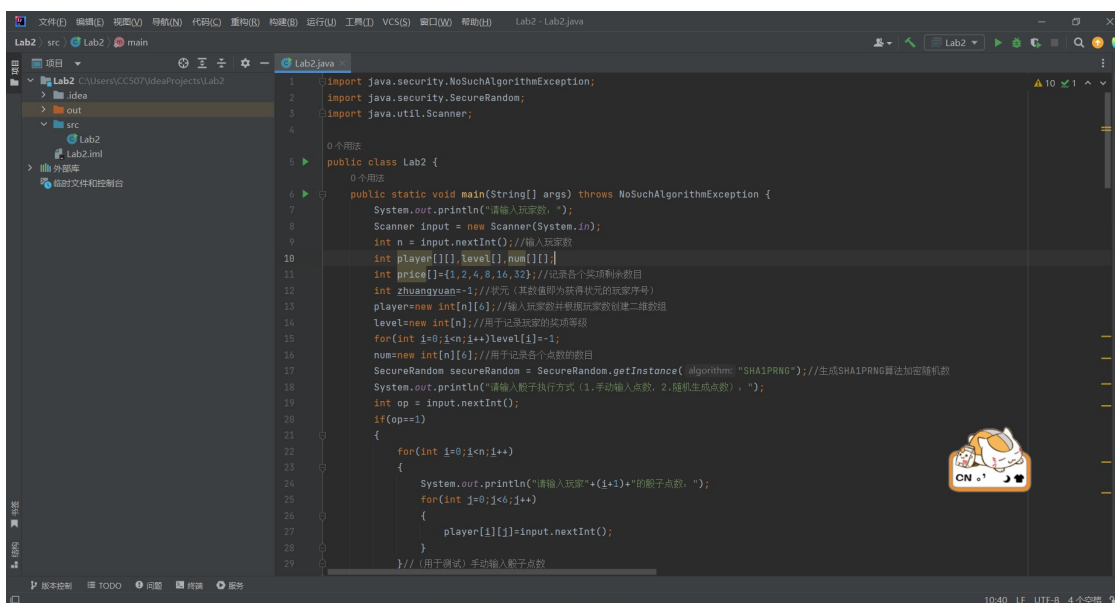
jdk 版本：java17.0.5

字符编码：ASCII

(二) 实现过程 (本部分为主要评分依据, 请描述解题思路, 比如总共设计几个类, 各个类的用途、成员、主要方法等及其之间调用关系等)

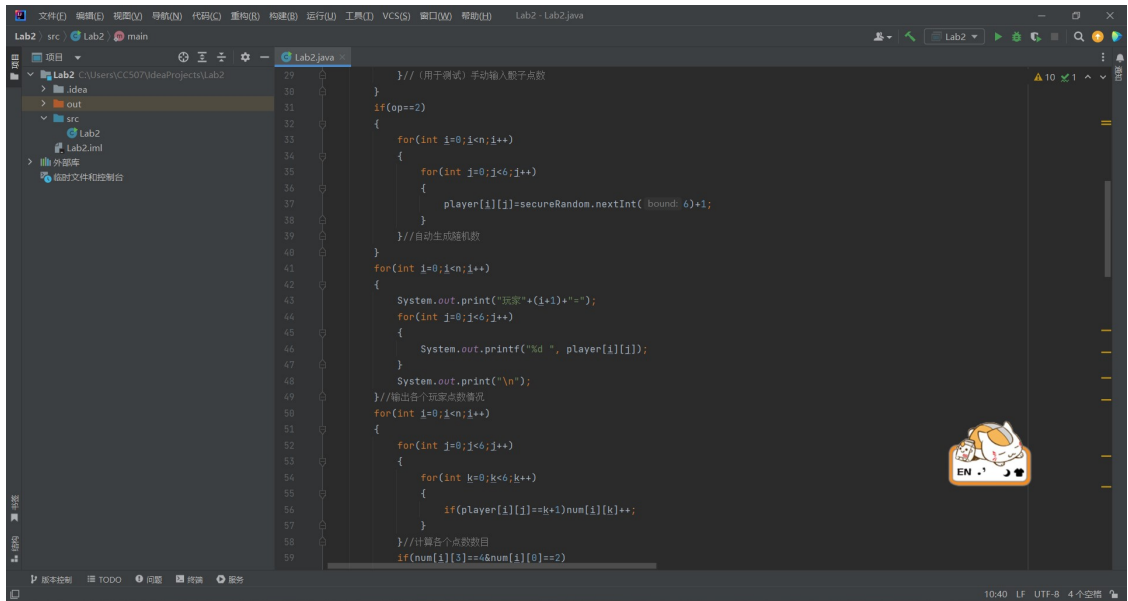
1. 创建整形 `n` 用于记录玩家数, 创建二维数组 `player[n][6]` 用于记录每个玩家所获得的骰子点数, 创建数组 `level[n]` 用于记录各个玩家获得奖项等级, 创建二维数组 `num[n][6]` 用于记录各个玩家获得的 6 种点数数目, 整形变量 `zhuangyuan` 用来记录状元玩家的编号, 数组 `price[6]` 用于记录各个奖项的剩余情况。
2. 根据玩家数目, 使用 `SecureRandom` 方法生成 1-6 的随机数, 计入 `player` 数组
3. 依次判断各个玩家获得各个点数数目, 计入 `num` 数组
4. 根据各个点数数目判断获得的奖项, 并且判断新状元的是否大于旧状元, 若大于, 则改变状元获得者。
5. 输出各个玩家的获奖情况。
6. 输出剩余的奖项数目。

(三) 过程截图 (本部分为主要评分依据, 一张全屏截图 (必须)、若干运行结果展示图 (可选), 主要代码 (可选))

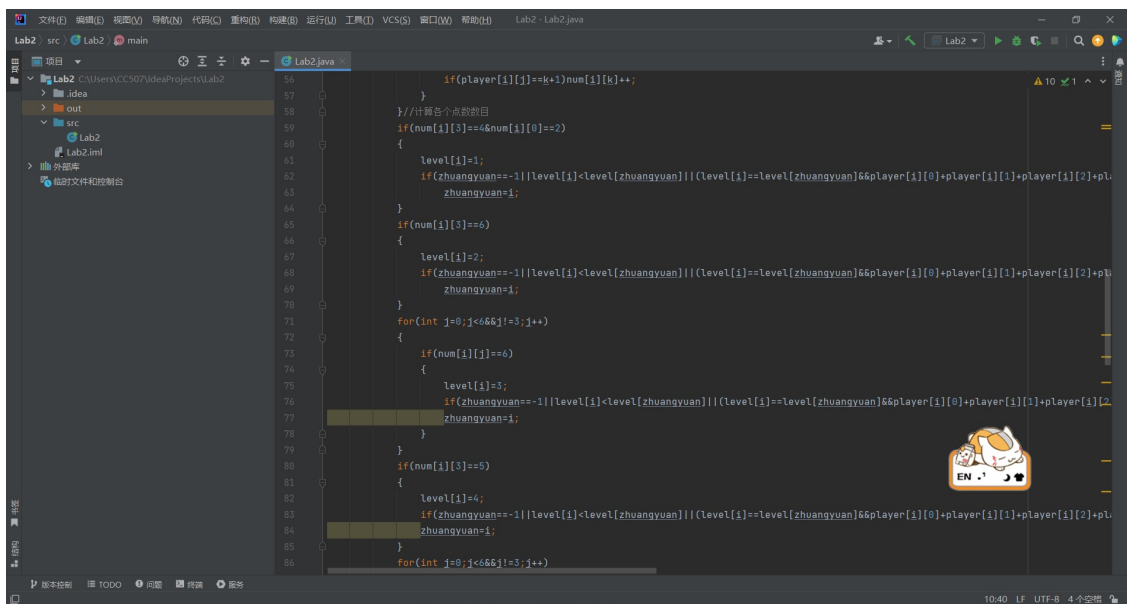


```
1 import java.security.NoSuchAlgorithmException;
2 import java.security.SecureRandom;
3 import java.util.Scanner;
4
5 public class Lab2 {
6     public static void main(String[] args) throws NoSuchAlgorithmException {
7         System.out.println("请输入玩家数:");
8         Scanner input = new Scanner(System.in);
9         int n = input.nextInt(); // 输入玩家数
10        int player[][] = new int[n][6]; // 记录各个奖项剩余数目
11        int price[] = {1, 2, 4, 8, 16, 32}; // 记录各个奖项的剩余数目
12        int zhuangyuan = -1; // 状元 (其数值即为获得状元的玩家序号)
13        player = new int[n][6]; // 输入玩家数并记录玩家数创建二维数组
14        level = new int[n]; // 用于记录玩家的奖项等级
15        for (int i = 0; i < n; i++) level[i] = -1;
16        num = new int[n][6]; // 用于记录各个点数的数目
17        SecureRandom secureRandom = SecureRandom.getInstance("SHA1PRNG"); // 生成SHA1PRNG算法加密随机数
18        System.out.println("请输入骰子执行方式 (1. 手动输入点数, 2. 随机生成点数):");
19        int op = input.nextInt();
20        if (op == 1)
21        {
22            for (int i = 0; i < n; i++)
23            {
24                System.out.println("请输入玩家" + (i + 1) + "的骰子点数:");
25                for (int j = 0; j < 6; j++)
26                {
27                    player[i][j] = input.nextInt();
28                }
29            } // (用于测试) 手动输入骰子点数
```

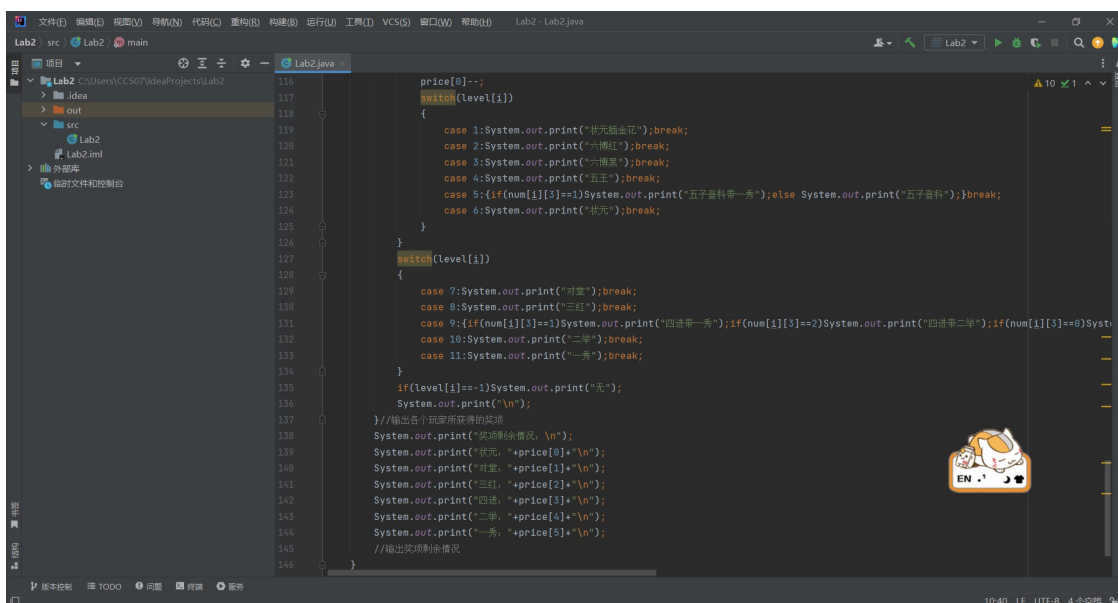
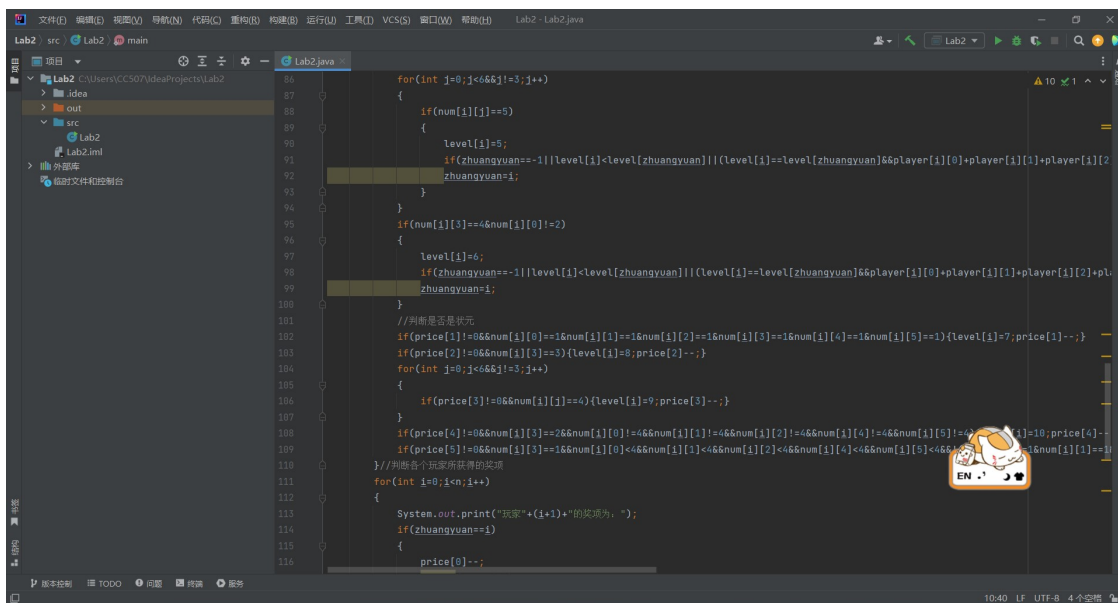
## 《Java 程序设计—王美红》实验报告



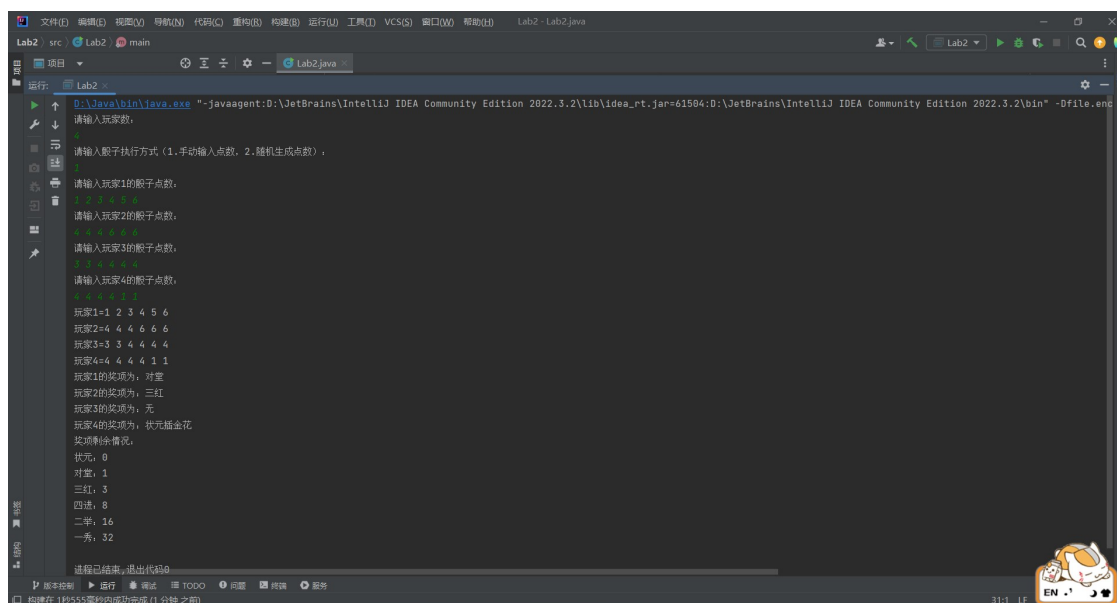
```
// (用于测试) 手动输入骰子点数
}
}
if(op==2)
{
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<6;j++)
        {
            player[i][j]=secureRandom.nextInt( bound: 6)+1;
        }
    }
    // 自动生成随机数
    for(int i=0;i<n;i++)
    {
        System.out.print("玩家"+(i+1)+"=");
        for(int j=0;j<6;j++)
        {
            System.out.printf("%d ", player[i][j]);
        }
        System.out.print("\n");
    }
    // 输出各个玩家的点数情况
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<6;j++)
        {
            for(int k=0;k<6;k++)
            {
                if(player[i][j]==k+1) num[i][k]++;
            }
        }
    }
    // 计算各个玩家的数目
    if(num[i][3]==4&&num[i][0]==2)
```



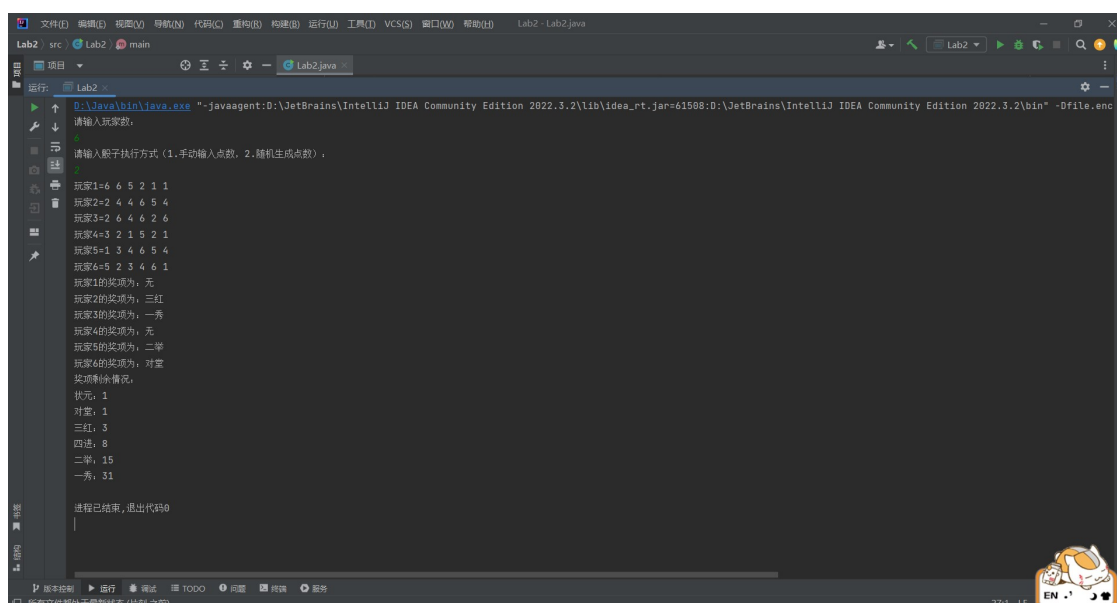
```
if(player[i][j]==k+1) num[i][k]++;
}
}
// 计算各个玩家的数目
if(num[i][3]==4&&num[i][0]==2)
{
    level[i]=1;
    if(zhuangyuan==1 || (level[i]<level[zhuangyuan] || (level[i]==level[zhuangyuan]&&player[i][0]+player[i][1]+player[i][2]+player[i][3]>player[zhuangyuan][0]+player[zhuangyuan][1]+player[zhuangyuan][2]+player[zhuangyuan][3]))
    {
        zhuangyuan=i;
    }
}
if(num[i][3]==6)
{
    level[i]=2;
    if(zhuangyuan==1 || (level[i]<level[zhuangyuan] || (level[i]==level[zhuangyuan]&&player[i][0]+player[i][1]+player[i][2]+player[i][3]>player[zhuangyuan][0]+player[zhuangyuan][1]+player[zhuangyuan][2]+player[zhuangyuan][3]))
    {
        zhuangyuan=i;
    }
}
for(int j=0;j<6&&j!=3;j++)
{
    if(num[i][j]==6)
    {
        level[i]=3;
        if(zhuangyuan==1 || (level[i]<level[zhuangyuan] || (level[i]==level[zhuangyuan]&&player[i][0]+player[i][1]+player[i][2]+player[i][3]>player[zhuangyuan][0]+player[zhuangyuan][1]+player[zhuangyuan][2]+player[zhuangyuan][3]))
        {
            zhuangyuan=i;
        }
    }
}
if(num[i][3]==5)
{
    level[i]=4;
    if(zhuangyuan==1 || (level[i]<level[zhuangyuan] || (level[i]==level[zhuangyuan]&&player[i][0]+player[i][1]+player[i][2]+player[i][3]>player[zhuangyuan][0]+player[zhuangyuan][1]+player[zhuangyuan][2]+player[zhuangyuan][3]))
    {
        zhuangyuan=i;
    }
}
for(int j=0;j<6&&j!=3;j++)
```



### 运行截图



```
运行: D:\Java\bin\java.exe "-javaagent:D:\JetBrains\IntelliJ IDEA Community Edition 2022.3.2\lib\idea_rt.jar=61584:D:\JetBrains\IntelliJ IDEA Community Edition 2022.3.2\bin" -Dfile.encoding=UTF-8
请输入玩家数:
4
请输入骰子执行方式 (1.手动输入点数, 2.随机生成点数):
1
请输入玩家1的骰子点数:
1 2 3 4 5 6
请输入玩家2的骰子点数:
4 4 4 4 4 4
请输入玩家3的骰子点数:
3 3 4 4 4 4
请输入玩家4的骰子点数:
4 4 4 4 1 1
玩家1=1 2 3 4 5 6
玩家2=4 4 4 6 6 6
玩家3=3 3 4 4 4 4
玩家4=4 4 4 4 1 1
玩家1的奖项为: 对堂
玩家2的奖项为: 三红
玩家3的奖项为: 无
玩家4的奖项为: 状元插花
奖项剩余情况:
状元: 0
对堂: 1
三红: 3
四红: 8
二半: 16
一秀: 32
进程已结束,退出代码0
```



```
运行: D:\Java\bin\java.exe "-javaagent:D:\JetBrains\IntelliJ IDEA Community Edition 2022.3.2\lib\idea_rt.jar=61588:D:\JetBrains\IntelliJ IDEA Community Edition 2022.3.2\bin" -Dfile.encoding=UTF-8
请输入玩家数:
6
请输入骰子执行方式 (1.手动输入点数, 2.随机生成点数):
2
玩家1=6 6 5 2 1 1
玩家2=2 4 4 6 5 4
玩家3=2 4 4 6 2 6
玩家4=3 2 1 5 2 1
玩家5=1 3 4 6 5 4
玩家6=5 2 3 4 6 1
玩家1的奖项为: 无
玩家2的奖项为: 三红
玩家3的奖项为: 一秀
玩家4的奖项为: 无
玩家5的奖项为: 二半
玩家6的奖项为: 对堂
奖项剩余情况:
状元: 1
对堂: 1
三红: 3
四红: 8
二半: 15
一秀: 31
进程已结束,退出代码0
```

### 三、实验总结与心得记录

本部分根据实验过程的所得所想描述,记录可供以后复习回看 {可以记录调试过程遇到的问题,自己哪些知识点掌握不够,设计是否有缺陷(比如耗时?耗内存?)是否有亮点,是否有精妙的算法,或者设计模式的应用,可吐槽,也可与其他语言作适当对比。} (本部分不作为平时评分依据)

备注：

建议附带代码提交的方式：导出工程压缩包。

平时实验成绩以考查参与度为主，所有实验要求自己完成，一旦发现抄袭或者其他投机取巧，取消所有平时成绩