# 厦門大學



# 信息学院软件工程系

# 《JAVA 程序设计》实验报告

实验 2

姓名: 黄勖

学号: 22920212204392

学院:信息学院

专业: 软件工程系

完成时间: 2023.3.7

## 一、实验目的及要求

- 练习 Java 的基本语法
- 熟悉控制结构

## 二、实验题目及实现过程

#### (一) 解题过程(基本+拓展)

题目大意:完成博饼游戏流程,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。



#### 拓展部分具体要求如下:

- (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。

#### 代码编写:

● 生成骰子点数(保留了测试入口,可以自行输入)

```
private static int[] randomDice() { //随机产生六个骰子点数
     var dice = new int[7];
     if (test == false) {
          Random rnd = new Random();
                dice[rnd.nextInt( bound: 6) + 1]++;
          for (int i = 1; i <= 6; i++)
                \underline{\text{dice}}[0] = \text{Math.max}(\underline{\text{dice}}[0], \underline{\text{dice}}[\underline{i}]);
          dice = new int[7];
          for(int i=0; i<6; i++)
                int temp = scan.nextInt();
                if(temp<1||temp>6)
                     System.out.println("输入错误,请重新输入");
                     dice[temp]++;
          for (int i = 1; i <= 6; i++)
                \underline{\text{dice}}[0] = \text{Math.} max(\underline{\text{dice}}[0], \underline{\text{dice}}[\underline{i}]);
```

判断是否为状元

```
private static boolean isZhuangYuan(int[] dice) { //判断是否为状元 return dice[4] >= 4 || dice[0] >= 5; }
```

● 判断普通奖项

```
private static int[] judgePrize(int[] dice) { //判断普通奖项
    if (isZhuangYuan(dice)) {
        if (dice[4] == 1)//五子登科带一秀
        {
            int[] ret = {1, 6};
            return ret;
        } else {
            int[] ret = {1};
            return ret;
        }
    } else {
        int[] ret = {2};
        return ret;
    } else if (dice[0] == 4)//四进
        {
            int[] ret = {4, 7 - dice[4]};
            return ret;
        } else {
            int[] ret = {4};
            return ret;
        } else if (dice[4] > 0) //三红二举一秀
        {
            int[] ret = {7 - dice[4]};
        }
}
```

······代码过长截取了部分,具体在实验附带 Lab2/src/中查看

● 判断状元奖项类型

```
private static int[] distinguishZhuangYuan(int[] dice) { //判断状元奖项类型
    int[] ret = new int[3];
    if (dice[1] == 2)//状元括金花
        ret[0] = 1;
    else if (dice[4] == 6)//六博红
        ret[0] = 2;
    else if (dice[0] == 6)//六博黑
    {
        ret[0] = 3;
        for (int i = 1; i <= 6; ++i) {
            if (dice[i] == 6) {
                ret[1] = i;
                break;
            }
        }
        else if (dice[4] == 5)//五王
        {
        ret[0] = 4;
        for (int i = 1; i <= 6; ++i) {
            if (dice[i] != 0 && i != 4) {
                ret[1] = i;
                break;
            }
        }
        else if (dice[0] == 5) //五子登科
        {
        ret[0] = 5;
        for (int i = 1; i <= 6; ++i) {
```

● 最后确认唯一的状元得主

判断游戏是否结束(还有没有剩余奖项)

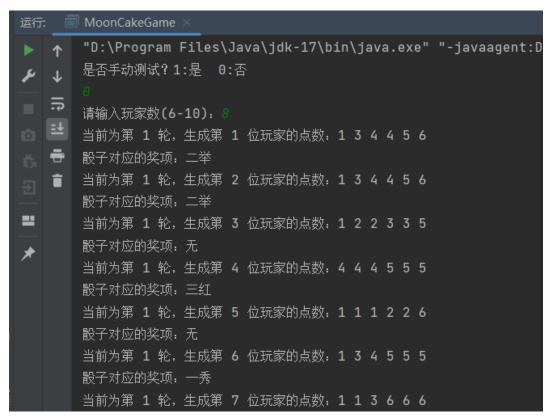
```
private static boolean isGameOver() { //判断游戏是否结束(还有没有剩余奖项)
    int sum = 0;
    for (int i : Remain_Prize)
        sum += i;
    if (sum == 0)
        return true;
    else
        return false;
}
```

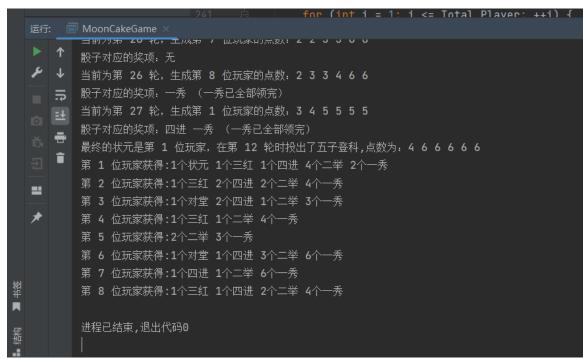
● Main 方法编写游戏全流程

#### (二) 结果展示

● 打开测试入口:

● 关闭测试入口并随机生成骰子:





## 三、实验总结与心得记录

通过本次实验,在博饼模拟过程中,我们需要使用 Java 语言来实现整个游戏的逻辑,这需要对 Java 基本语法和控制结构有一定的了解。

Java 基本语法包括变量声明、数据类型、运算符、条件语句、循环语句等等。 在博饼模拟过程中,我们需要定义多个变量来表示骰子点数、博饼结果、玩家得分 等信息。同时,我们需要使用运算符进行加减乘除等运算,以及使用条件语句和循 环语句来控制游戏的逻辑。

控制结构是编程中非常重要的一部分,包括顺序结构、分支结构和循环结构。 在博饼模拟过程中,我们需要使用分支结构和循环结构来实现游戏的逻辑。例如, 在进行博饼时,如果骰子点数为"五子",则直接计分并结束本轮游戏;如果骰子 点数为"状元插金花",则计分并再次掷骰子。我们需要使用 if 语句来实现这些逻辑。

同时,我们也需要使用循环结构来实现游戏的多轮进行。例如,当玩家选择开始游戏后,我们需要不断进行骰子掷点和计分的过程,直到玩家选择结束游戏为止。 在这里,我们需要使用 while 语句来实现。

总之,博饼模拟过程是一个较为简单的 Java 编程练习,通过实现游戏的逻辑,可以加深对 Java 基本语法和控制结构的理解和掌握。