**启动和效果**

运行Start.java

//正式的

Game g = new Game(new ArrayList<>(33), new Random(), ps);

g.procedue();

//测试用例

TestGame g = new TestGame(new ArrayList<>(33), new Random(), ps);

g.procedue();

测试用例效果：

初始状态

P0 : 1万 2万 3万 4万 5万 6万 7万 8万 9万 1条 2条 3条 4条

P1: 1万 1万 2万 3万 4万 5万 6万 7万 7万 7万 4条 5条 6条

P2: 2万 3万 4万 4万 5万 6万 2条 3条 4条 5条 6条 7条 8条

P3: 2条 3条 4条 5条 6条 7条 8条 9条 3柄 4柄 5柄 6柄

P1得到牌

p1分到了9万

p1:[0 1万],[1 2万],[2 3万],[3 4万],[4 5万],[5 6万],[6 7万],[7 8万],[8 9万],[9 9万],[10 1条],[11 2条],[12 3条],[13 4条],

请输入你是否要胡牌:1、是 2、不是

2

请输入你是否要杠:1、是 2、不是

2

请输入你是否要碰:1、是 2、不是

2

P1出牌

p1:[0 1万],[1 2万],[2 3万],[3 4万],[4 5万],[5 6万],[6 7万],[7 8万],[8 9万],[9 9万],[10 1条],[11 2条],[12 3条],[13 4条],

请在下行选择出一张牌

6

p1打出了7万

p2:[0 1万],[1 1万],[2 2万],[3 3万],[4 4万],[5 5万],[6 6万],[7 7万],[8 7万],[9 7万],[10 4条],[11 5条],[12 6条],

请输入你是否要杠:1、是 2、不是

1

杠这张牌7万

成功杠

…….

胡牌和碰也可以如此这般

**关于胡牌的判定：**

认为132张牌，33,4形式存储，33为总类型数，4为每种个数

//[0,8] 万 ，[9-17] 条 ，[18-26] 柄， [27,32] 风中发

认为胡牌规则：风发中不具有连续性

在此前提下

1. 先把数据按照类别分类（万，条，杠）
2. 判断每一类是否无杂牌，同时统计有规则的牌的数量
3. 由于牌的连续性，需要对牌进行合适的切分。例如：

1 1 1 2 3 4 4 4 5 6

可以有多种划分方法，对于此类，在先不考虑类别超出范围的前提下，使用穷举法，即a、先按照顺优先划分串，看是否能得到无冗余、可分类的数据

b、如果不行，再按照碰优先的顺序划分串，看是否能得到无冗余、可分类的数据

如果两次都失败，则认为该串不可用。

以下皆可用上述方法，不过第一和第二种需要先排除掉不与其他牌相连的碰

//0 杠 ， 1 碰 ， 2 顺 ， 3 对

s = "1 1 1 5 6 7 8 8 8 9 9"; //0 2 1 1

s = "1 1 1 5 5 5 5 6 7 7 7 7"; //0 3 1 0

s = "3 4 5 5 5 5 6 6 6 7 8 9 9 9"; //0 2 2 1

s = "2 3 4 4 4 4 5 5 5 6 6 6 7 7"; //0 0 4 1

s = "2 3 4 4 4 4 5 5 5 6 7 7 8 9"; //0 1 3 1

4、整合所有的类别

s = "3 4 5 9 10 11 19 20 21 28 28 28 29 29 33 33";

最后可以判别出此序列

1. 查表，判断类别

假设顺子数量为n

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 胡法 | 杠（已公示） | 顺 | 碰 | 对 |
| 1 | 0 | n | 4-n | 1 |
| 2 | 1 | n | 3-n | 1 |
| 3 | 2 | n | 2-n | 1 |
| 4 | 3 | n | 1-n | 1 |
| 5 | 0 | 0 | 0 | 7 |