

版本说明

1. 此平台用于参赛选手进行本地算法程序训练，属于离线版本
2. 离线版本只能维持4个算法程序进行游戏
3. 本次竞赛所采用的掇蛋规则遵循2017年国家体育总局发布的《淮安掇蛋竞赛规则》，以江苏省体育局发布的《江苏省掇蛋竞赛简易规则（2017版）》作为补充

平台运行要求

1. 离线版本目前支持Windows 平台和Linux平台（注：可能会被一些防护软件视做为病毒，添加信任即可或者无视即可）
2. 确认本地端口号23456空闲
3. 解压文件夹之后会得到3个目录及1个文件

- clients -- 包含解析的随机程序
- Ubuntu -- 在Linux系统下可使用的离线平台
- windows -- 在windows系统下可使用的离线平台
- 使用说明.pdf

4. Ubuntu使用方法

- ubuntu文件夹存放着文件guandan_offline_v1006
- 找到guandan_offline_v1006所在目录
- 在终端输入如下命令，看见Ready for connect.字样即运行成功（请勿占用23456端口）

```
./guandan_offline_v1006 10
```

- 其中10为游戏次数（即一方从2打到A，并且双下且当前等级为我方等级）。例如，100次游戏的命令如下

```
./guandan_offline_v1006 100
```

5. Windows使用方法

- windows文件夹存放着文件guandan_offline_v1006.exe
- 找到guandan_offline_v1006.exe所在目录
- 在命令行输入如下命令，看见Ready for connect.字样即运行成功（请勿占用23456端口）

```
guandan_offline_v1006.exe 10
```

- 其中10为游戏次数（即一方从2打到A，并且双下且当前等级为我方等级）。例如，100次游戏的命令如下

```
./guandan_offline_v1006.exe 100
```

6. clients文件夹下包含4个示例代码，分别为client1.py、client2.py、client3.py、client3.py，及动作类和状态类

- 示例代码使用方法：
 - 根据对应系统，按上面的使用方法成功运行后，依次运行client1.py, client2.py, client3.py, client4.py
 - 注意：由于本版本为命令行版本，所以连满4人之后游戏将自动开始，即在连入client4.py之后将自动开始

攒蛋AI的编写与连接

1. 不限制编程语言，算法程序通过Websocket来连接平台，平台与算法程序之间通过JSON数据的交互来进行游戏
2. 离线平台websocket连接地址：ws://127.0.0.1:23456/game/{user_info}
 - user_info字段表示用户信息，用于标识该算法程序
 - 若对所连接的地址有所疑惑，可以参考示例程序中的连接地址
 - **由于是本地的离线平台，不做掉线重连处理，且不会对所发出的错误的动作做任何处理，做出错误动作时服务器会输出错误动作，直至正确的算法程序发送了正确的动作**
3. 离线平台下无法选择游戏座位，第1个连接的算法程序和第3个连接的算法程序自动成为一队，同理，第2个连接的和第4个连接的一队。
4. **AI程序应针对不同的JSON数据结构进行相应的解析，并进行相对应的动作。JSON数据结构的说明见下**
5. **按上述说明运行离线平台后，依次运行所提供的4个示例代码即可进行随机对局**

JSON数据说明

1. 卡牌：使用长度为2的字符串表示，如'S2'表示黑桃2，下面进行详细说明
 - 花色：黑桃、红桃、梅花、方片分别对应字符S, H, C, D。特别地，小王的花色为S，大王的花色为H
 - 点数：A、2、3、4、5、6、7、8、9、10、J、Q、K分别对应A、2、3、4、5、6、7、8、9、T、J、Q、K。即点数10用字符T表示。特别地，小王的点数为字符B，大王的点数为字符R
 - 如'S2'表示黑桃2，'HQ'表示红桃Q；'SB'表示小王，'HR'表示大王，'PASS'表示PASS
2. 牌型：[牌型，点数，卡牌]
 - 一个牌型表示为一个长度为3的列表（数组），包含三个固定部分：牌型，点数，卡牌
 - 牌型的类型为字符串，取值范围为['Single', 'Pair', 'Trips', 'ThreePair', 'ThreeWithTwo', 'TripsPair', 'Straight', 'Straight', 'Boom', 'PASS', 'tribute', 'back']
 - 牌型的中英文对照见附件
 - 点数的类型为字符串，取值范围为['A', '2', '3', '4', '5', '6', '7', '8', '9', 'T', 'J', 'Q', 'K', 'B', 'R', 'PASS']
 - 卡牌用于表示该牌型所包含的全部卡牌，类型为列表数据结构，例子如下
 - 单张方片5表示为：['Single', '5', ['D5']]
 - 对4表示为：['Pair', '4', ['H4', 'C4']]
 - PASS: ['PASS', 'PASS', 'PASS']
 - ["tribute", "tribute", ['D5']]表示为进贡方片5，["back", "back", ['D5']]表示为还贡方片5
3. **整个游戏过程中，AI会收到平台发来的11种JSON数据格式**，可通过JSON中字段(stage, type)的值来判定是何种类型
 - stage表示游戏阶段，type表示消息类型，下面进行详细说明
 - "type": "notify", "stage": "beginning" 表示游戏开始阶段，告知每位玩家的手牌情况，例子如下：

- {


```

        "type": "notify",
        "stage": "beginning",
        "handCards": ['S2', 'H2', 'H2', 'C2', 'D2', 'S3', 'H3', 'D3', 'S4',
          'D5', 'C6', 'H7', 'C7', 'C8', 'D8', 'DT', 'HJ', 'DJ', 'HQ', 'HQ', 'CQ',
          'SK', 'H9', 'H9', 'C9', 'D9', 'HR'],
        "myPos": 1,
      
```

 }

- 该例子所对应的语义为：游戏开始，我是1号位，我的手牌为handCards

- "type": "notify", "stage": "play"表示出牌阶段，用于通知其他玩家做出的动作，例子如下：

- {


```

        "type": "notify",
        "stage": "play",
        "curPos": 1,
        "curAction": ['Single', '2', ['s2']],
        "greaterPos": 1,
        "greaterAction": ['Single', '2', ['s2']]
      
```

 }

- 该例子所对应的语义为：在游戏的出牌阶段，1号位打出单张-黑桃2，当前的最大动作位1号位打出的黑桃2

- type: "notify", "stage": "tribute"表示进贡阶段，用于通知所有玩家进贡者（们）都进贡了什么牌，例子如下：

- {


```

        "type": "notify",
        "stage": "tribute",
        "result": [[0, 3, 's2']]
      
```

 }

- 该例子所对应的语义为：在游戏的进贡阶段，0号位玩家向3号位玩家进贡了黑桃2

- 若result的值为[[0, 3, 'S2'], [2, 1, 'S2']]，则对应语义为，在游戏的进贡阶段，0号位玩家向3号位玩家进贡了黑桃2，2号玩家向1号玩家进贡了黑桃2

- "type": "notify", "stage": "anti-tribute"表示抗贡阶段，用于通知所有玩家，有人抗贡。其中antiNums的取值与antiPos数组的长度所对应，例子如下：

- {


```

        "type": "notify",
        "stage": "anti-tribute",
        "antiNums": 2,
        "antiPos": [0, 2]
      
```

 }

- 该例子所对应的语义为：在游戏的进贡阶段发生了抗贡，抗贡玩家有2名，抗贡的玩家的座位号分别为0号位玩家和2号位玩家

- "type": "notify", "stage": "back"表示还贡阶段，用于通知所有玩家还贡者（们）都还贡了什么牌，例子如下：

```
○ {
  "type": "notify",
  "stage": "back",
  "result": [[3, 0, 'S2']]
}
```

- 该例子所对应的语义为：在游戏的还贡阶段，3号位玩家向2号位玩家还贡了黑桃2
- 若result的值为[[3, 0, 'S2'], [1, 2, 'S2']], 则对应语义为，在游戏的还贡阶段，3号位玩家向0号位玩家还贡了黑桃2，1号玩家向2号玩家还贡了黑桃2

- "type": "notify", "stage": "episodeOver"表示小局结束阶段，用于通知所有玩家小局结束，例子如下：

```
○ {
  "type": "notify",
  "stage": "episodeOver",
  "order": [0, 1, 2, 3],
  "curRank": 'A',
  "restCards": [[3, ['C2']]]
}
```

- 该例子所对应的语义为：在游戏的出牌阶段，该小局结束，结束时所打的等级为A，各个玩家完牌的次序分别为0号位、1号位、2号位、3号位，其中在3号位所剩余的手牌为方片2
- 若完牌次序为[0, 2, 1, 3]，即第一完牌的玩家和第二完牌的玩家同属同一队伍时，该小局结束，restCards会收录1号位玩家的手牌和3号位玩家的手牌，例如restCards: [[3, ['C2']], [1, ['C2']]]

- "type": "notify", "stage": "gameOver"表示到达指定游戏次数游戏结束，用于通知所有玩家游戏结束，例子如下：

```
○ {
  "type": "notify",
  "stage": "gameOver",
  "curTimes": 1,
  "settingTimes": 1,
}
```

- 该例子的语义为：当前训练次数为2, 设定的游戏次数为3，本次游戏结束
- 注：一次游戏是指：0、1、2、3四位玩家中有任意一方处于A等级时（且当前所打的等级为该方的A等级）同时以第一名、第二名的完牌次序进行完牌，则该次游戏结束

- "type": "notify", "stage": "gameResult"表示到达指定游戏次数游戏结束，用于通知所有玩家游戏结束，例子如下：

```
○ {
  "type": "notify",
  "stage": "gameResult",
  "victoryNum": [2, 0, 2, 0],
  "draws": [0, 0, 0, 0]
}
```

- 该例子的语义为：达到设定的场次，其中0号位胜利2次，1号位胜利0次，2号位胜利2次，3号位胜利2次，在所进行的场次游戏中发生的平局次数为：0号位胜利0次，1号位胜利0次，2号位胜利0次，3号位胜利0次

- 注：根据规则，当有一方处于A等级，经过2小局游戏后仍未获得游戏胜利，则自动降为2等级，继续从2等级打至A等级。理论上可能会发生死循环的情况（即从A-2，再从2-A），在离线平台中做如下处理

- 离线平台：任意一方从A降级为2，记为一次，当这一次数达到50次时，该次游戏记为平局
- 当这一次数达到50次时，在线平台与离线平台的处理略有不同，以大赛官网所公布的规则为准

- "type": "act", "stage": "play"表示出牌阶段，用于通知该玩家做出动作，例子如下：

```

{
  "type": "act",
  "handCards": ['S2', 'H2', 'H2', 'C2', 'D2', 'S3', 'H3',
'D3', 'S4', 'D5', 'C6', 'H7', 'C7', 'C8', 'D8', 'DT', 'HJ', 'DJ', 'HQ',
'HQ', 'CQ', 'SK', 'H9', 'H9', 'C9', 'D9', 'HR'],
  "publicInfo": [
    {'rest': 22, 'playArea': ['ThreewithTwo', 'A', ['SA',
'SA', 'DA', 'H8', 'D8']]},
    {'rest': 23, 'playArea': ['Bomb', '4', ['H4', 'H4',
'C4', 'D4']]},
    {'rest': 23, 'playArea': ['Bomb', 'A', ['HA', 'HA',
'CA', 'DA']]},
    {'rest': 27, 'playArea': None}
  ],
  "selfRank": 'K',
  "oppoRank": '9',
  "curRank": '9',
  "stage": "play",
  "curPos": 2,
  "curAction": ['Bomb', 'A', ['HA', 'HA', 'CA', 'DA']],
  "greaterAction": 2,
  "greaterPos": ['Bomb', 'A', ['HA', 'HA', 'CA', 'DA']],
  "actionList": [
    ['PASS', 'PASS', 'PASS'],
    ['Bomb', '9', ['H9', 'H9', 'C9', 'D9']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'C2', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'C2', 'D2']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'D2', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'H9', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'C2', 'D2', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'C2', 'H9', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'D2', 'H9', 'H9']],
    ['Bomb', '2', ['S2', 'C2', 'D2', 'H9', 'H9']],
    ['Bomb', '2', ['H2', 'H2', 'C2', 'D2', 'H9']],
    ['Bomb', '2', ['H2', 'H2', 'C2', 'H9', 'H9']],
    ['Bomb', '2', ['H2', 'H2', 'D2', 'H9', 'H9']],
    ['Bomb', '2', ['H2', 'C2', 'D2', 'H9', 'H9']],
    ['Bomb', '3', ['S3', 'H3', 'D3', 'H9', 'H9']],
    ['Bomb', 'Q', ['HQ', 'HQ', 'CQ', 'H9', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'C2', 'D2', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'C2', 'H9', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'D2', 'H9', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'C2', 'D2', 'H9', 'H9']],
    ['Bomb', '2', ['H2', 'H2', 'C2', 'D2', 'H9', 'H9']],
    ['Bomb', '2', ['S2', 'H2', 'H2', 'C2', 'D2', 'H9',
'H9']]
  ],
  "indexRange": 21,
}

```

- 该例子的语义为：

- 0号位打出['ThreeWithTwo', 'A', ['SA', 'SA', 'DA', 'H8', 'D8']]
 - 1号位打出['Bomb', '4', ['H4', 'H4', 'C4', 'D4']]
 - 2号位打出['Bomb', 'A', ['HA', 'HA', 'CA', 'DA']], 最大动作为2号位打出的['Bomb', 'A', ['HA', 'HA', 'CA', 'DA']]
 - 我方等级: K, 对方等级: 9, 当前等级9
 - 当前动作为2号-动作['Bomb', 'A', ['HA', 'HA', 'CA', 'DA']], 最大动作为2号-动作['Bomb', 'A', ['HA', 'HA', 'CA', 'DA']], 目前可选动作如actionList所示
 - 可选动作范围为: 0至21 (包含0和21)
- 注: 若某一位玩家打出某牌型, 其余玩家均PASS, 又轮到该玩家重新出牌时, 如下的部分字段表示当前的无任何人做出动作, 也无任何最大动作。即率先出牌时、接风时、进贡时、退贡时, 该部分字段为-1和None

```
{...
  "curPos": -1,
  "curAction": None,
  "greaterAction": None,
  "greaterPos": -1,
  ...}
```

- "type": "act", "stage": "tribute"表示进贡阶段, 用于该玩家进贡, 例子如下:

```
{
  "type": "act",
  "handCards": [['H3', 'D3', 'D4', 'H5', 'C5', 'S6', 'C6',
'D6', 'S7', 'S7', 'H7', 'D7', 'H8', 'C8', 'C8', 'H9', 'C9', 'HT', 'DT',
'HJ', 'DQ', 'HK', 'DK', 'SA', 'H2', 'H2', 'D2']],
  "publicInfo": [
    {'rest': 27, 'playArea': None},
    {'rest': 27, 'playArea': None},
    {'rest': 27, 'playArea': None},
    {'rest': 27, 'playArea': None}
  ],
  "selfRank": '2',
  "oppoRank": '9',
  "curRank": '2',
  "stage": "tribute",
  "curPos": -1,
  "curAction": None,
  "greaterAction": -1,
  "greaterPos": None,
  "actionList": [['tribute', 'tribute', ['D2']]],
  "indexRange": 0,
}
```

- 该例子的语义为: 游戏进贡阶段开始, 我方手牌为handCards, 轮到自己进贡, 可进贡的牌为方片2 (红桃配系统会自动跳过), 可选动作索引范围为0 (包含0)

- "type": "act", "stage": "back"表示还贡阶段, 用于该玩家进贡, 如下所示:

```
{
  "type": "act",
  "handCards": ['H2', 'S3', 'H3', 'C3', 'D3', 'S4', 'C4',
'S6', 'C6', 'C7', 'C7', 'C8', 'H9', 'CJ', 'HQ', 'SK', 'SK', 'CK', 'DK',
'SA', 'SA', 'DA', 'H5', 'C5', 'SB', 'HR', 'HR'],
  "publicInfo": [
    {'rest': 27, 'playArea': None},
  ],
}
```

```

        {'rest': 27, 'playArea': None},
        {'rest': 27, 'playArea': None},
        {'rest': 27, 'playArea': None}
    ],
    "selfRank": '5',
    "oppoRank": '9',
    "curRank": '5',
    "stage": "back",
    "curPos": -1,
    "curAction": None,
    "greaterAction": -1,
    "greaterPos": None,
    "actionList": [
        ['back', 'back', ['H2']],
        ['back', 'back', ['S3']],
        ['back', 'back', ['H3']],
        ['back', 'back', ['C3']],
        ['back', 'back', ['D3']],
        ['back', 'back', ['S4']],
        ['back', 'back', ['C4']],
        ['back', 'back', ['S6']],
        ['back', 'back', ['C6']],
        ['back', 'back', ['C7']],
        ['back', 'back', ['C8']],
        ['back', 'back', ['H9']]
    ]
}

```

- 该例子的语义为：游戏还贡阶段开始，我方手牌为handCards，轮到自己还贡，可进贡的牌为actionList（系统会自动跳过等级5的牌），可选动作索引范围为0至11（包含0和11）

4. 发送动作

- 相比之前的版本，动作的发送方式有所变化
 - 选手需要在所给定的动作列表中进行选择并给出指定动作的下标，如动作列表为：[['PASS', 'PASS', 'PASS'], ['Single', '5', ['C5']]]，则选手的可选动作范围为0-1
 - 选手只需返回形如JSON：{"actIndex": 0}的形式即可
 - 在所给的示例代码中包含动作解析类，解析完成后会返回随机索引，并由程序发送至平台
 - 其他详细内容，详见所提供的示例代码
- 在需要做出动作时，算法程序所发送的动作必须符合平台所给的动作范围

附件-中英文对照表

牌型名称	英文
单张	Single
对子	Pair
三张	Trips
三连对	ThreePair
三带二	ThreeWithTwo
钢板 (两个三张)	TwoTrips
顺子	Straight
同花顺	StraightFlush
炸弹	Bomb
进贡	tribute
还贡	back
过	PASS