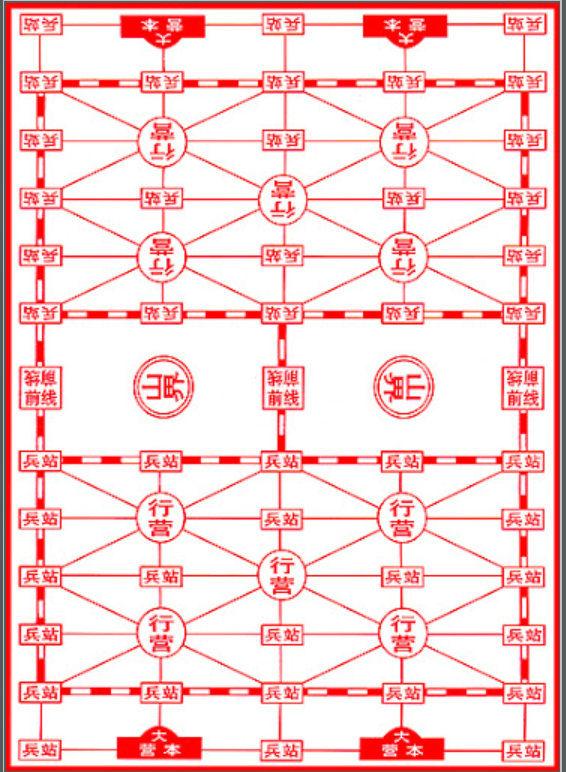
# 人工智能课程实践部分

本课程实践部分需要完成并提交一个陆军旗AI程序，每四-五人为一组。陆军棋，又称陆战棋，简称军棋，是[中国](https://zh.wikipedia.org/wiki/%E4%B8%AD%E5%9C%8B" \o "中国)近代的一种两人棋类，设计根据军队中的[军衔](https://zh.wikipedia.org/wiki/%E8%BB%8D%E9%9A%8E" \o "军衔)。每一方有25枚棋子，先夺得对方军旗者为胜。军棋有三种模式：明棋、暗棋和翻棋，**比赛使用明棋规则**。

# 下棋规则

# 棋盘



1. 停靠点分为四类：兵站、前线、行营和大本营。棋盘上每方有三十个停靠点， 其中有二十三个兵站，五个行营和两个大本营。两方势力通过三条铁路连接，每两条铁路之间有一个“山界”。在某些规则中，山界可以当作行营使用，但这里不考虑山界，只将其视为背景。
2. 路线分为两类：公路线和铁路线，细线为公路线，粗线为铁路线。

# 棋子

1. **每方棋子的种类和个数：**

司令\*1、军长\*1、师长\*2、旅长\*2、团长\*2、营长\*2、连长\*3、排长\*3、工兵\*3、地雷\*3、炸弹\*2、军棋\*1。（共25枚棋子）

1. **行棋规则：**

每步只能移动一枚棋子，地雷和军旗无法移动。

在公路线上，棋子只能沿公路方向移动一格。在铁路上的棋子，可以沿着铁路直线行动不限格数，但不能跨过其他棋子；若该棋子为工兵，可以沿着未被阻挡的铁路线到达任一位置（可以转弯）。

1. **吃子规则：**

同方棋子不可互相攻击，在行营内的棋子不可被攻击。

当棋子移动的目的地有棋子时，移来的棋子攻击未动的棋子，此时按优先级规则决定胜负，胜者覆盖负者或双方同归于尽。若目的地的棋子不能被攻击，则不能如此移动。

1. **吃子优先级：**

司令 > 军长 > 师长 > 旅长 > 团长 > 营长 > 连长 > 排长 > 工兵 > 军旗，相同棋子同归于尽。炸弹与任何敌方棋子相遇都会同归于尽。工兵吃地雷，其他棋子和地雷相碰则同归于尽。军旗被吃则游戏结束，被吃的一方为输家。

# 布阵

双方在开始行棋前进行布阵，每方有25枚棋子，25个合法布阵点（大本营+兵站），布阵需满足如下规则：

1. 棋子可以摆放在己方的兵站或大本营中，不能摆放在行营中；
2. 炸弹不能放在第一排；
3. 地雷必须放在最后两排；
4. 军旗必须放在大本营中。

布阵结束，双方亮棋，确定先后手，轮流行棋。**在此次对战评审的实际操作过程中，布阵这一部分使用人工的方式完成（即由参赛方自拟布阵方案，将我方布阵方案以中心对称的方式扩展至红黑两方，以NCN串表示方法，存储于txt格式文件首行。）。**

注：对于每一组提交，如果先手则取该组提交的红方布镇；如果后手则取该组提交的黑方布镇。

# 胜负和

**对局中，出现下列情况之一，本方算输，对方赢：**

1. 己方的军旗被对方棋子吃掉；
2. 己方无子可走；
3. 己方违规走子

**出现下列情况之一，为和棋：**

1. 双方均无可能取胜的简单局势；
2. 符合“自然限着”的回合规定，即在连续15个回合中（也可根据具体情况酌减），双方都没有吃过一个棋子。

# 二、军棋AI引擎编写规范

# 1. 通讯方法

军旗AI引擎跟界面之间通过“标准输入”和“标准输出”(例如C/C++语言中的stdin和stdout)来通讯。**需要完成的AI对应这一部分。**

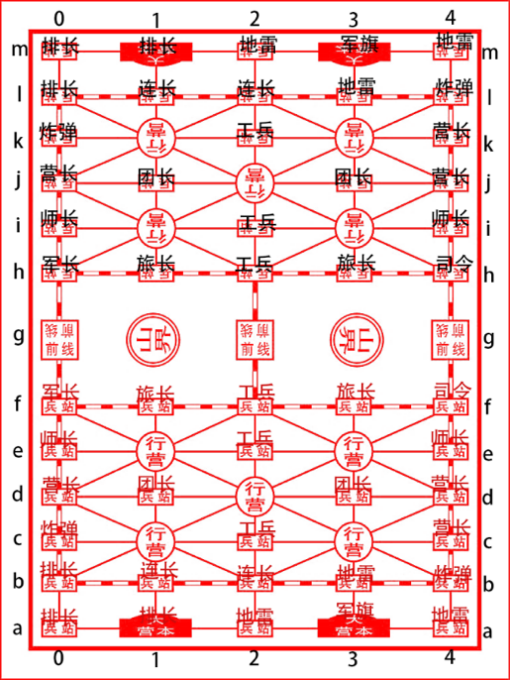
通常，界面向引擎发送的信息称为“指令”，而引擎向界面发送的信息称为“反馈”。不管是指令还是反馈，都是以“行”为单位的，即每条指令和反馈都必须以“回车换行”(例如大多数编程语言中的\n)结束。

注意：引擎用缓冲方式发出反馈(例如C/C++语言中直接将字符串写入stdout)，因此每输出一行都必须用fflush()语句刷新缓冲区。

# 2. 着法和棋盘表示

1. 棋盘表示

NCN(Naive Chessboard Notation)是本门课程用来表示军棋局面的记录规范，在这个规范下，一个局面可以简单地用一行“NCN格式串”来表示。



依照上图，举例说明当前棋盘的NCN串：

ppdfd/plldz/z1g1y/yt1ty/s1g1s/jvgva/5/JVGVA/S1G1S/YT1TY/Z1G1Y/PLLDZ/PPDFD 0 0

　 ① 红色区域，表示棋盘布局。这里要注意两点，一是棋盘有13行，所以要用12个“/”把每一行隔开；二是棋子名称用英文字母表示，小写表示红方，大写表示黑方。其中数字表示有对应数量的空位。例如上例中的“5”表示“g”这一行有5个空位。其余字母的表示方法如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 棋种 | 代表字母 | 棋种 | 代表字母 | 棋种 | 代表字母 |
| 军旗 | f,F | 排长 | p,P | 旅长 | v,V |
| 地雷 | d,D | 连长 | l,L | 师长 | s,S |
| 炸弹 | z,Z | 营长 | y,Y | 军长 | j,J |
| 工兵 | g,G | 团长 | t,T | 司令 | a,A |

　　② 绿色区域，表示双方没有吃子的走棋步数(半回合数)，通常该值达到120就要判和(六十回合无吃子)，一旦当前局面的上一步操作进行了吃子，就这里重置为“0”。

　　③ 蓝色区域，表示当前的操作数，随着局势的发展逐渐增加。

1. 着法格式

引擎回答界面应该走哪个着法，这样的着法都用4个字符表示。只要知道起始格子和到达格子，就确定了着法。例如h2i2。

**4. 引擎运行参数相关约定**

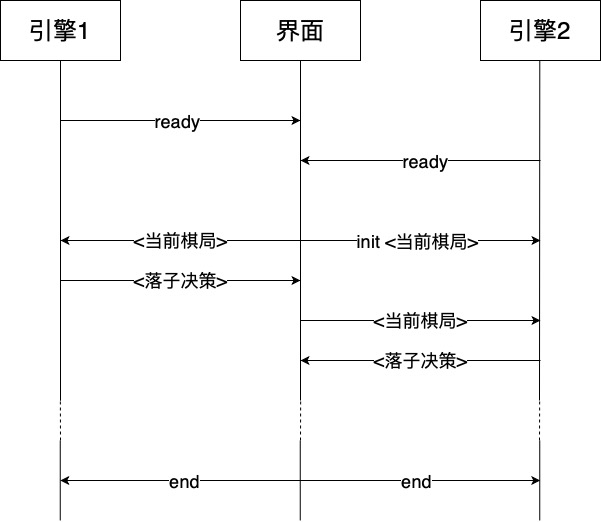
引擎应当通过启动时的运行参数对于单步决策时间以及引擎所属角色（红蓝方）进行设置。

例如: python ai.py –-role 1 –-time 60

注： 参数role：0表示红方、1表示黑方。

参数time：表示单步决策时间，单位秒。

# 5. 指令和反馈



注：指令用红色表示，反馈用蓝色表示。

1. ready

引擎启动后向界面发送ready，随后引擎处于空闲状态。

1. <当前棋局>

界面将当前棋局以NCN的形式使用标准输入传递给引擎，引擎开始推理。

例如: ppdfd/plldz/z1g1y/yt1ty/s1g1s/jvgva/5/JVGVA/S1G1S/YT1TY/Z1G1Y/PLLDZ/PPDFD 0 0

1. init <当前棋局>

用于将开局时的棋局传递给后手的引擎，保证开始运算的时间基本一致。

例如: init ppdfd/plldz/z1g1y/yt1ty/s1g1s/jvgva/5/JVGVA/S1G1S/YT1TY/Z1G1Y/PLLDZ/PPDFD 0 0

1. <落子决策>

使用标准输入的方式反馈运算结果，此后引擎返回空闲状态亦或是继续运算。

例如: h2i2

1. end

棋局结束，引擎退出。

# 三、比赛计分规则

# 1. 比赛判分

1）单局比赛：按照上述胜负和条件，判断一局比赛的胜负。

2）时间限制：若一方走棋时间超过限时，则直接判负。

3）初赛中，任意两组进行一场两局对决（分别为先后手），对一场比赛的判分，胜者计2分，平局双方各计1分，负者计0分。**注：每局可以采用不同的阵型。**

# 2. 违规说明

1）一方有违规落子，计0分，判负。

2）违规行为包括：

a)AI把棋子下在已有己方子的位置。

b)AI未按照棋子走棋规则下棋的。

c)AI把棋下出边界的。

d)其他未满足前文“须实现的功能”的。

# 3. 初赛结束后，排名方式如下

1）优先按照总积分（大分）多少排名。

2）总积分相同的，按照战胜的对手的积分和（小分）排名

3）大小分相同的，按照提交AI的先后顺序排名。

# 4. 初赛结束后，总积分进行公示，前两名进行决赛，决赛为展示性质的表演赛。

# 四、关于使用已有的开源代码

* 1. 不允许使用任何一手的开源代码。即，所有的代码必须由组员亲手打出。