

项目二实验指导书

一、步骤 1: α - β 剪枝

不需要使用 Ai 平台, 直接在 Idea 中用 Java 完成即可。数据文件 tree.txt

二、步骤 2: 随机棋手-V0

1. 解压缩 AiDeveloper.zip, 里面是一个 Idea 项目, 已经设置好了对 AI 平台的依赖, 平台的源代码在另外一个压缩包 AiFramework_src.zip 中。同学们只需要重新设置 JDK 为你电脑上的 JDK, **Java11 以上**即可。



2. 在 src 文件夹下, 创建 stud 包, 在该包下已经创建了三个小组 (组号 77,88,99) 所对应的包, 在该包下创建相关的 AI, 这三个小组包下分别创建本步骤中所要求的三个随机棋手。同学们可以在此基础上, 按照这样的结构, 任意添加棋手 AI, 请添加你们组的一个 AI, 使用《项目 2-博弈搜索.pdf》中步骤 2 的第③点中的方法创建随机棋手。

3. AITester 类是用来帮助同学们完成实验的, 程序的输出可作为实验分析的数据。

三、六子 AI 大战

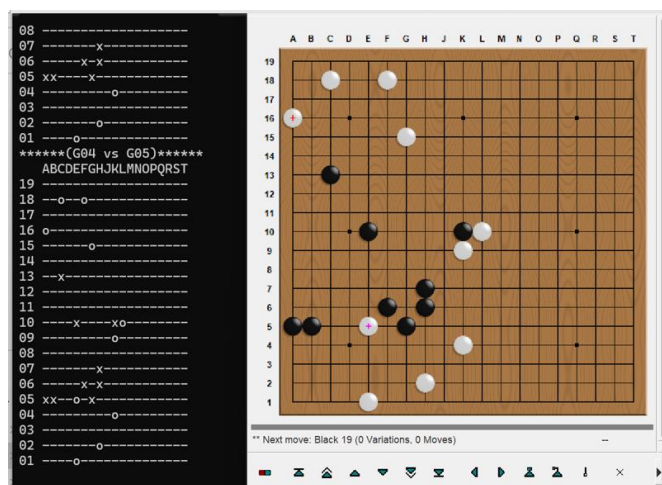
1. 请解压缩 oucLeague.zip, 进入生成的 oucLeague 子目录, 其内容如下。

```
-oucLeague
  -lib
    -aiFramework.jar
    -jagoclient.jar
  -ais
    -g04.jar
    -g05.jar
  -AITester.class
  -file.properties
  -run.bat
```

```
#the Ids of each player. class name "stud.g04.AI"
Player_Ids = 4,5

#the group who host the games with other groups.
# At all the games, host play first, i.e. white
Host = 4
```

2. 请确保已安装 **JDK11** 以上版本，点击 run.bat，g04 的 AI 作为 host，先手与棋手 Id 列表中的其他棋手各下一局棋。确认程序能够正常运行，出现以下界面。



3. 将你的每个 AI（你们组的 AI，以及其他三个随机棋手）所在的子目录（例如 stud/g01，注意是**斜杠**，不是反斜杠）单独打成一个 jar 包(命名为 g01.jar)。请参考《将一个子目录打成 jar 包.pdf》

4. 请清空 oucleague 的 ais 子目录，并将你的 jar 包复制到该子目录下。**修改 file.properties 文件，将你的所有 AI 的 id 添加到 player_ids 列表，并将 Host 设为你们组的编号。**重新运行 run.bat，你们组 AI 将先手和其他 AI 进行对局。

四、真正的 AI 开发及对战

1. 在 stud 下创建你们小组自己的包，在此编写调试你的 AI。如 g77 包中所示，你可以任意添加一些辅助的类（如 G77Board），用来帮助你编写你的 AI，尽可能把 AI 中的一些功能抽象封装为辅助类或者函数。

2. 项目截止后，各小组将自己的最终版 AI 打成 jar 包，发给助教。

3. 助教整合所有的小组的 jar 包后，打包成 group.zip，发给每个小组。

4. 小组间进行 AI 对战。每个小组出一台电脑，作为 host 与其他小组的 AI 先手对战。请清空 oucleague 的 ais 子目录，并将 group.zip 里的 jar 包解压到该子目录下。确保 player_ids 中包含所有小组的 id，Host 设为本小组的 id。

5. 运行 run.bat，host 与其他小组 AI 的对局将一局局的开始，每局结束后，请 host 小组的同学负责将即时结果录入腾讯共享 Excel 文件中。

	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	先手得分
G01		2	0	2	0	2	2	1	2	2	13
G02	0		0	2	2	2	2	0	2	2	10
G03	2	2		2	0	2	2	2	2	2	16
G04	0	2	0		0	2	0	0	2	2	8
G05	2	2	0	2		0	2	2	2	2	14
G06	2	2	0	0	2		2	0	2	2	12
G07	2	2	0	2	2	2		2	2	2	16
G08	2	2	0	2	2	2	2		2	2	16
G09	0	2	0	0	0	0	2	0		2	6
G10	2	2	0	2	0	0	2	2	0		10
后手得分	8	2	20	6	12	8	4	11	4	4	/
最后得分	21	12	36	14	26	20	20	27	10	14	/