

# AI2603-01 人工智能基础

## 期末作业

### 1. 任务简介

对于本次作业，你需要在跳棋环境中：

- 实现带有 Alpha-Beta 剪枝的 minimax 算法
- 实现更优的方法（强化学习等）

你的智能体应评估当前棋盘状态，尽可能提高战胜对手的概率。详细规则和指南请参阅第 2 节。

### 2. 游戏规则

#### 2.1 游戏设置

- 棋盘如图一所示，尺寸及棋子数量由参数triangle\_size控制（本作业要求triangle\_size不可小于2）。
- **Player 1**（代码中键值为player\_0）先手，控制红色棋子，初始位置在棋盘上端。
- **Player 2**（代码中键值为player\_3）后手，控制绿色棋子，初始位置在棋盘下端。

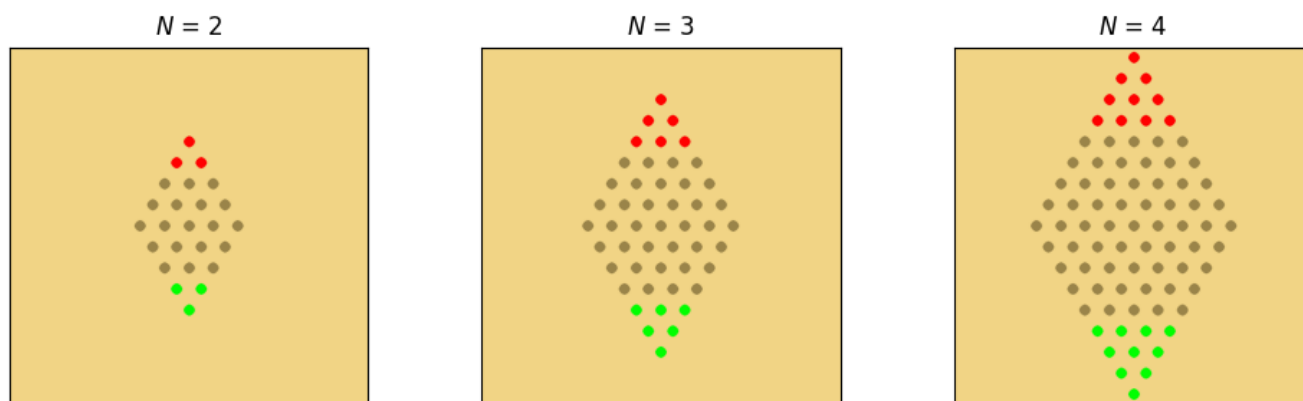


图1、不同尺寸的棋盘，从左至右依次为 triangle\_size=2, 3, 4

#### 2.2 移动规则

每名玩家一回合仅可移动一枚棋子，可选的移动方式包含以下两种：

- **移动到相邻区域：**棋子可以移动到周围6个直接相邻且未被任意一方的棋子占据的位置，之后切换到对方的回合。

- **越过相邻的棋子：**如果选定棋子周围6个直接相邻位置存在其他棋子，且沿相邻棋子所处方向的下一个位置未被占据，那么玩家可以将选定棋子越过相邻棋子直接占据该位置。允许连续跳跃，但必须遵守以下规则：
  - 连续跳跃一旦开始，始终只能移动同一枚棋子。
  - 不允许跳回上一步的位置。

若玩家中途放弃连续跳跃或无法继续跳跃，则切换到对方的回合。

## 2.3 胜利条件

先用己方棋子占满目标区域（对方玩家的初始三角区域）的玩家获胜。

# 3. 代码

## 3.1 设备要求

本作业的代码一般只能在 **支持GPU加速** 的 **Linux系统** 下执行，无法满足此设备要求的同学可以按之前发布的教程使用云计算平台

## 3.2 环境配置

代码要求python>=3.10，推荐同学们按照以下步骤进行环境配置

代码块

```
1  conda create --name env_checker python=3.10
2  conda activate env_checker
3  conda install anaconda::swig # 若不进行此步骤，将无法正常编译box2d-py库
4  # 跳转到项目代码下，安装依赖库
5  cd your_project
6  pip install -r requirements.txt
```

## 3.3 代码描述

- **ChineseChecker/**
  - **env/：**定义了棋盘游戏环境。
  - **models/：**定义了强化学习模块。
- **agents.py：**实现了随机策略和贪心策略，你在实现minimax策略时，可以将它们作为格式的参考。

- **train.py & rllib-marl.py**: 定义了强化学习的训练流程，你可以在train.py中修改神经网络结构和超参数等。按照以下指令运行train.py即可开启训练流程：

代码块

```
1 python train.py -t --triangle_size 2
```

- **play.py**: 定义了测试流程（需要补充minimax策略接口）。如果你需要测试你实现的agent，可按照如下指令运行play.py：

代码块

```
1 # 以测试基于强化学习的agent为例，假设训练好的模型路径为your_checkpoint
2 python play.py --use_rl --checkpoint your_checkpoint
```

### 3.4 任务描述

本次作业，你至少需要完成以下任务：

- 在agents.py中实现带有Alpha-Beta剪枝的minimax算法。
- 实现至少一种其他算法。项目提供了一种强化学习框架，可在此基础上进行修改，也可另行实现其他算法。
- 尽可能提高算法在与贪心策略与预训练策略对弈时的胜率。

## 4. 评分标准

各项评分标准及其占比如下：

- 代码占5%，必须能够正常运行，**允许一切不改变游戏规则的修改**。
- 报告占75%，主要检查以下两部分：
  - 实验过程：详细介绍想法、算法实现和算法优化的过程。
  - 自测结果：记录minimax agent和自定义agent与贪心策略、预训练的强化学习基线分别对弈至少20次的胜率，需要截图证据。
  - 说明各组员的贡献。
  - 如果包含引用，请在报告末尾标出。
- 最终测试占20%，助教再次测试提交算法的胜率，根据真实表现进行打分。

## 5. 提交要求

- 报告语言不限，篇幅不超过**15页**普通A4纸。
- 以压缩包的形式提交，每组提交一份。压缩包格式应遵循以下规范：

**{组员1姓名}\_{组员2姓名}\_project.zip**

**code/**

**agents.py**

**{your best checkpoint}**

**其他文件...**

**report.pdf**

- **Deadline:** 2026.01.03