# AI大战

## 作者

计科1601 丁嘉禾

## 项目地址

Github：<https://github.com/dicarne/aifightworld>

## 开发环境

Unity2018.2.7f1

## 源起

最初想要开发这个AI对战游戏是因为在思考社团新学期的活动的时候，觉得可以给新社员提供一些有趣的活动，和编程相关的又不只是做题的，让对编程感兴趣的新同学可以通过这个游戏学习编程和算法。如果只是练习练习HelloWorld未免显得太枯燥无趣了，而如果是参与到游戏中、为了达成某个目标而编程的话，会显得更有意思，更真实，有目的性。

一开始了解过Code Combat，一个外国人开发的编程教育平台，可以通过Python、Javascript等语言控制英雄闯关、解谜，甚至可以编写自己的关卡。而国内网易代理了，取名为极客战记。但是在使用过程中，感觉即使使用网易的版本，加载也会非常的慢，甚至经常根本无法加载。而且也没有我们可控的多人游戏方式，很难组织同学使用这个东西学习。因为Code Combat是开源的，我成功搭建自己的Code Combat服务器后，发现问题就是传输的数据量很大，如果要提供很多人流畅加载的网络带宽服务器一个月可能就要几千上万，负担不起。于是就萌发了自己开发一个AI对战的游戏的想法。

后来以本游戏的实现思路做了一些尝试，完成了服务器与客户端的数据交换，通过RPC调用体验还可以，但客户端因为设计的有些问题所以暂时无法开发，便暂时搁置。直到最近重新开始一个新的客户端设计。

## 设计思路

这个游戏9月15日开始设计开发。第一天想要仿照Code Combat做成两个英雄对抗的游戏，然后搭建了脚本的运行环境、基本UI等，设计了一些API可以通过玩家写的脚本来控制游戏对象，比如移动、探测获取前方的东西等。费了一些功夫研究完了射线检测，代表玩家的小方块可以在一些逻辑下运动了。然后就被难住了，因为我不知道仅凭这几个功能是否真的能做出比如走迷宫寻找到对手这种看起来比较复杂的事，看起来就是一个简单的寻路，实则这些功能不应该直接写好提供给玩家，应该玩家自己来编程实现功能。而玩家是不能直接获得地图数据的，我觉得就像人在走迷宫一样，如果进入入口就直接知道了迷宫的整个地图，未免显得太作弊了点。应该由玩家自己来探索记录迷宫才对，而玩家能用的只有一个功能就是看看前方是什么，有多远，感觉短时间内我没有办法解决如何玩通这个游戏的问题，更别说如果是刚刚接触编程的同学了，而且要做的哪怕是一般好看也涉及到不少动画，时间技术上来不及，因此我觉得设计思路应该加以调整。

接着第二天，我废弃了前一天的场景，利用前一天研究的脚本运行环境和脚本与游戏相互调用的方法，重新设计了一个游戏场景。这次我将玩家对战的场景简单的放到一个10\*10的网格地图中，这样起码可移动的坐标就很确定，而不用像之前一天那样，整张地图完全没有标识，连坐标都不能直接的观察到。既然是对战，网格图，自然可以容易的想到围棋、象棋这种。我想做成类似于策略游戏攻城略地的那种感觉，就规定了占地必须在自己领地的边缘才能占领，防止直接占了人家大本营；占领的土地可以升级，类似于大富翁，升级后的建筑更难被敌人所占领；如果包围一个空地快，那么可以立即占领并搭建高级建筑等。

为了防止做出来一个无法完成的游戏，我在开始设计API之前先写了一段比较简单的Lua脚本当作玩家可能的输入，从写脚本的过程中发现我应该暴露什么样的信息给玩家才能完成这个AI。再对着这个脚本实现API，让这段脚本能够正常的运行。然后根据不同等级的建筑和阵营用不同颜色的地块作为标记。

后来，在打包游戏的过程中，发现了一些比较奇怪的问题，就是在编辑器里可以运行，但是发布出来就无法正常运行。因为中间使用了XLua作为Lua脚本解析运行的环境，按照官方给的方法并无法解决问题，最后将执行的地方换了一种实现方法绕过了之前遇到的问题。

## 实现说明

### 地图

地图动态生成，10\*10大小，每个地图块是一个TileCtrl类，用于更新显示出来的样式。另保存了地块的信息、所属玩家，建筑等级等。

### 脚本执行环境

本游戏玩家可以编写Lua脚本操控AI。其中与游戏访问的函数与属性等都在传入的参数self内。Self实际上是一个TileHeroData类，储存相关数据。内部方法无法直接修改游戏数据，但可以查询并做安排。TileHeroData类的方法主要做查询判断之用。只有当玩家脚本运行结束后，TileHero类才会执行玩家所做的安排。

TileHero继承自LuaRunner类，封装了一个XLua运行环境，通过调用ChangeCode()方法可以改变执行的代码。里面定义了一些时间，会分别在开始时、每帧更新时调用，可以重写基类的虚方法获得普通Unity的脚本拥有的Start与Update且不破坏基类的行为。

### 对手AI

对手AI的脚本储存在Assets/Resources目录下，在开始时会自动加载作为对手。

### 回合实现

回合实现在Level2Control类的Turn()方法。这是一个协程，在开始后每隔0.2秒执行一次，双方选手交替执行，当执行到一百回合则停止，判断得分高的玩家获胜。

## 改进事项

1. 可以增加服务器端，服务器储存玩家的脚本，然后通过客户端大厅选择下载到本地进行对战，然后将胜利的脚本再传输给服务器，做出排名表等。
2. 美化界面，为不同等级的地块用更好看的图片。
3. 给游戏中的代码框增加高亮提示、滚动条等功能。
4. 增加更多的API供脚本调用，与游戏本身有更多的交互功能。
5. 增加更多游戏机制，比如包围、轰炸等，为策略提供更多的选择。
6. 增加新的关卡类型，增添更多的游戏方式。
7. 对不同分辨率进行适配，现在是简单适配能应付部分分辨率情况，但有时也会出现错位严重的情况。
8. 从底层修复可以不判断执行条件违规操作的漏洞。

## 其他说明

发布出来的可执行文件位于Build目录下。