

# Project1

## 塔防游戏

Version 1.1.0

刘添翼\*

计算机科学与技术系，南京大学

2021 年 9 月 18 日

## 1 最后期限

Project1 的最后期限是北京时间 2021 年 10 月 10 日 23 时 59 分。

在此之前，每位同学需要提交课程项目的源码、可执行文件和项目报告到待定的某个指定地点。

## 2 概述

塔防游戏是一种很有趣的游戏。在过去的十多年间，该类型的游戏一直流行着，如《植物大战僵尸》、《王国保卫战》、《保卫萝卜》和《明日方舟》等。所以，让我们写一个塔防游戏吧！

## 3 原则

游戏必须是从第一行代码开始写成的，符合面向对象和泛型程序设计范式的，C++ 程序。推荐使用第三方的 C++ 图形库来完成图形的绘制，键盘鼠标事件的捕获和处理等内容，这样可以集中精力设计和实现游戏的架构。

## 4 技术指标

我们参考《明日方舟》和《植物大战僵尸》，制订了这份技术指标。本节是 Project1 的基本要求，其中描述的所有内容都需要完成，但是所有注记内容，如**注记 4.1**所述，是可选的内容，仅供参考。

**注记 4.1.** 写在这种（看起来就很显眼的）注记里的是可选的内容，或者是评注性质的段落。这些内容仅供参考。

---

\*tyi.liu@outlook.com

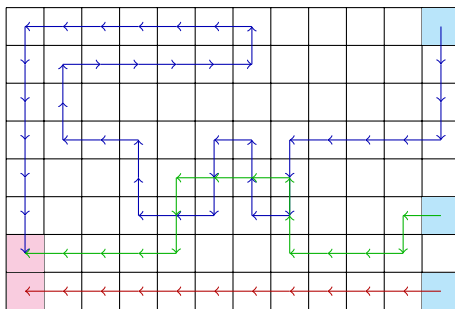


图 4.1: 地图、格子和路径的示意图。图中有红、绿、蓝三条路径。路径的起始格子被标记成蓝色，结束格子被标记成红色。

#### 4.1 世界观

游戏中存在两个对立阵营, 一个由玩家操控, 一个由程序操控, 分别称为**我方**和**敌方**。

每个阵营都有自己的胜利目标, 我方的胜利目标是在有限次攻击中阻止敌方穿过地图 (参见第 4.2 节), 敌方的目标是在有限次攻击之内穿过地图。

注记 4.2. 具体的胜败条件可以自由发挥。

作为参考, 失败条件可以像《植物大战僵尸》那样在路径的末端设置一个割草机; 也可以像其他很多塔防游戏一样给我方设置一个生命值, 有敌方单位到达路径的末端时扣除一定数值的生命。

而胜利条件, 可以是挺过了若干波敌方的进攻, 也可以简单地设定成我方无法胜利。

#### 4.2 地图

地图是游戏的舞台。

注记 4.3. 地图的数量和具体式样在 *Project1* 中不做要求, 但请在自己的项目中留好接口, 方便完成潜在的后续要求。

地图在逻辑上是一个  $m \times n$  的矩形, 每个  $1 \times 1$  的矩形是一个**格子**。当然, 在实现的时候不用考虑美术方面的问题, 能够达意就可以。

地图上存在若干条有向的**路径**。一条路径是格子的一个有穷序列, 序列中相邻的两个格子是**四邻接**的, 并且序列中不存在重复的格子。路径是有向的, 方向是从序列的第一个格子到最后一个格子。图 4.1 给出了一张  $12 \times 8$  的地图。

敌方的单位将从路径的起始格子出现, 按照序列, 移动到路径的结束格子。特别地, 不同路径可以共用格子, 包括起始格子和结束格子。

注记 4.4. 引入路径的概念让地图的设计变得灵活: 如果把所有的路径都设置成从右向左依次排开, 就成为经典的《植物大战僵尸》; 如果把路径设置成有很多转折, 就有一点《王国保卫战》的感觉。

#### 4.3 资源

我方可以在地图上合适的位置 (参见第 4.4 节), 以一定的**资源**为代价, 部署单位。当然, 实现上, 需要游戏为玩家提供一套界面, 让玩家能够选择要在何处部署何种单位。

注记 4.5. 纯键盘操作当然是可行的。为了更好的游戏体验，也可以尝试加入对鼠标的支持。

资源随时间流逝而自动增加。我方也可以手动撤退已经部署的单位。

注记 4.6. 这里的设定完全可以随各位同学开心，不过想要强调游戏平衡性的同学可以在这里稍微多思考一点。

比如，可以设置一些特殊的可以产出资资源的单位，也可以设定成在手动撤退单位的时候返还若干资源。

## 4.4 单位

每个阵营都可以产生若干种具有一定功能的单位。不同阵营的单位用各自的方式互相攻击，为对应的阵营争取胜利。每个单位都有一定的生命值，如果生命值过低，单位将被破坏。

我方的单位有近战单位和远程单位。

近战单位需要部署在路径所在的格子上。近战单位可以阻挡一定数量的敌方单位跨过自己所在的格子，也可以主动攻击自己攻击范围内的敌方单位。

注记 4.7. 如果把近战单位的攻击范围设定得比较大，但是不能阻拦敌方单位，敌方单位也不能攻击自己，就可以实现《植物大战僵尸》中的地刺；如果把攻击力设定成 0 但是血量很高，就可以实现坚果。

我们并没有阻止同学们把近战单位部署在已经有其他近战单位的格子之上。或许可以用这个机制实现南瓜。

远程单位不能部署在路径所在的格子上，但需要部署在路径的附近。远程单位不能阻拦敌方的单位，但是攻击范围一般较近战单位大。

注记 4.8. 远程单位在 *Project1* 中不做要求，但请在自己的项目中留好接口，方便完成潜在的后续要求。

敌方的单位，当然，不受我方控制。它们将在路径的起始点随机生成，按照一定的策略向路径的结束点移动，并攻击沿途的处在攻击范围内的我方单位。

注记 4.9. 敌方单位的具体情况在 *Project1* 中可以自由发挥，但请在自己的项目中留好接口，方便完成潜在的后续要求。

## 5 Q&A

问题 5.1. 我可以使用 *EasyX*、*Unreal Engine*、*NoesisGUI* 之类的库吗？

可以，只要它们为程序的开发者提供的 API 是 C++ 的即可。

过于复杂和高级的库可能并不容易学习掌握，打包和发布程序或许也会相对困难；过于简单的库则缺乏一些必要的基础设施，会不必要地加大任务量。

一般来说，我们会推荐初学者使用一些成熟的、有良好文档的、容易上手的库，这样的库不必是 Qt，或许某些（不为我们所熟悉的）专为游戏开发设计的库是最适合这个项目的。如果有同学对此有心得，欢迎联系我们。

**问题 5.2.** 我必须在 *Project1* 中实现一个 GUI 吗？

并非如此。我们只要求项目在最终有一个可用的 GUI，并不强求各位同学在 *Project1* 就实现它的部分或者全部。

不过为了避免不必要的重构，思考如何设计程序的核心逻辑使其能在各种 UI（包括 Graphical User Interface 和 Command Line Interface）上展示是有必要的。

**问题 5.3.** 我不喜欢这个游戏，我有一个新的想法。

请带上新项目的文档来联系我们，我们很欢迎新的想法。

不过考虑到这是一门程序设计课的课程项目，我们可能会对新项目的扩展性有比较高的要求，因此可能难免会提出一些（或许看起来和这个项目并不搭的）要求。总体来说新的项目是被允许的，但是如果新的项目不能很好地体现这门课程的宗旨，或许项目的最高分数会打一些轻微的折扣。

**问题 5.4.** 我不喜欢某些规定动作，我可以改变/简化/删去它们吗？

参见 [问题 5.3](#)。请带上你的新想法的文档来联系我们。

如果经过我们评估，认为新的想法的难度适中，可以体现本课程的宗旨，那么是可以的；如果新的想法显然弱化了项目的难度或者和项目的根本宗旨不合，可能会轻微地影响项目的得分。

**问题 5.5.** 我没有美术天赋，我是不是需要自己硬着头皮画地图/单位/动画/…？

完全不需要。

参见 [第 4.2 节](#)，只要能够达意，不论是自制的美术素材，还是经授权使用的其他人的美术素材，或者只是简单的文本和字符，都是可以的。