

课程设计：塔防游戏

Department of Computer Science
Nanjing University

April 25, 2024

Contents

1. 背景
2. 技术指标： 阶段一
3. 技术指标： 阶段二
4. 技术指标： 阶段三
5. 技术指标： 阶段四
6. 分数构成
7. 一些问题
8. 参考资料

Contents

1. 背景

2. 技术指标： 阶段一

3. 技术指标： 阶段二

4. 技术指标： 阶段三

5. 技术指标： 阶段四

6. 分数构成

7. 一些问题

8. 参考资料

背景

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》、《明日方舟》等。



Figure: 植物大战僵尸

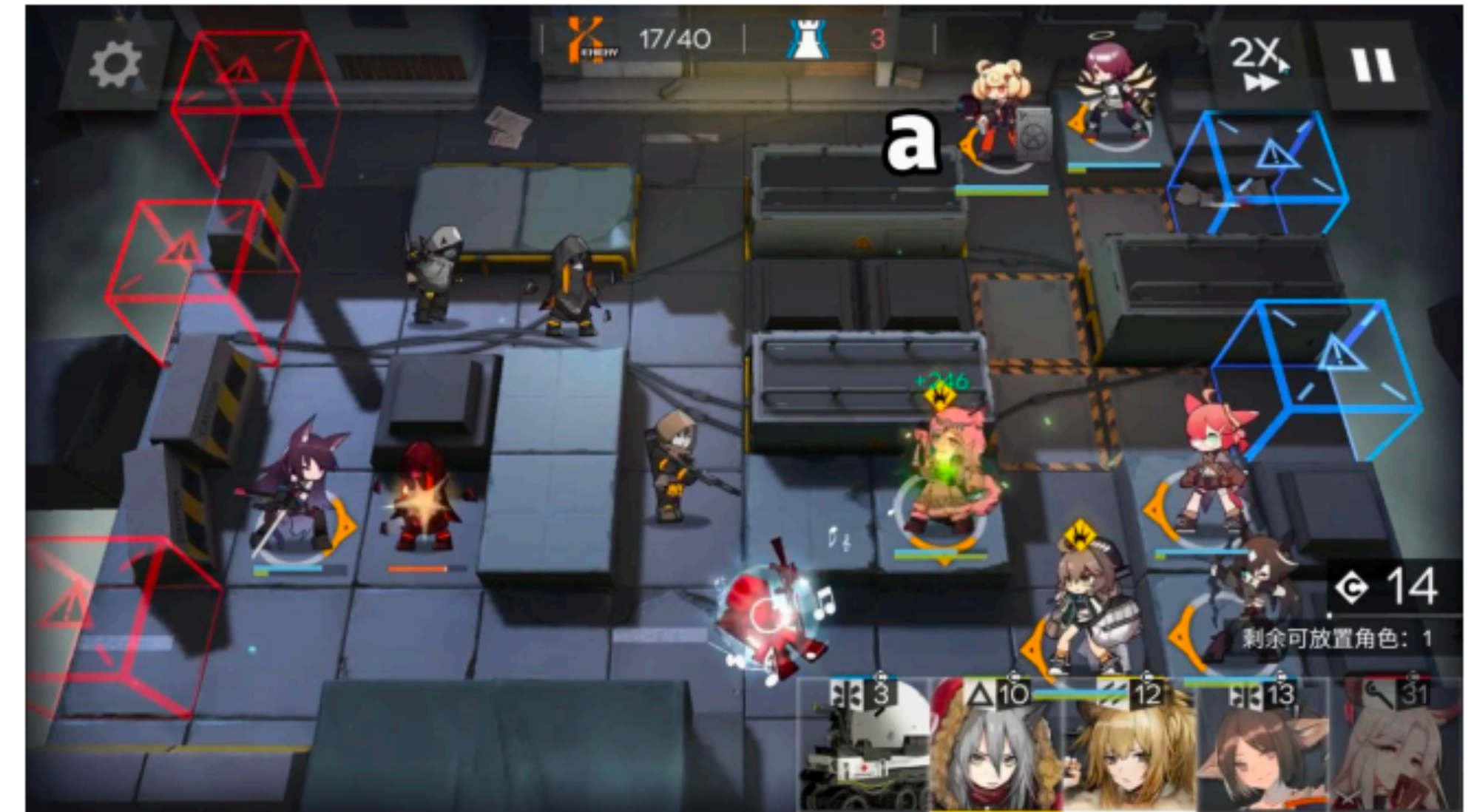


Figure: 明日方舟

实验安排

本次课程设计的目标是一个简单的塔防游戏。我们参考已有的一些塔防游戏，制订了这份课程设计手册。共分为四个阶段：

- 阶段一（无需提交与验收）：完成游戏基础设置的创建，比如地图、双方单位的创建等，进行简单的交战。
- 阶段二（无需提交，第12周验收，检查进度，不计分）：支持导入导出地图、导入导出存档、引入词缀机制来增加游戏趣味。
- 阶段三（第14周完成提交并验收，计分）：加入GUI（如果你前面阶段没有实现的话），展示单位的攻击和受伤效果。
- 阶段四（第16周完成提交并验收，计分）：自由发挥/按Extension部分拓展。（临近考试周，请自行斟酌）

注意：阶段三和阶段四都需要提交代码到课程网站。

开始之前

- **抄袭是被严厉禁止的。** 游戏必须是自己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等内容，这样可以集中精力设计和实现游戏的架构。
- 共分四个阶段，一二阶段不要求有图形界面，第三阶段才要求完成图形界面。第四阶段则是完成前面阶段的拓展功能，或者你当然也可以自己提出并实现新的拓展功能，我们会根据难度进行评分。每阶段依然是两周，我们会在第三阶段验收前面所有基础功能，在第四阶段验收拓展功能。
- 我们同样也提供了实现图形化界面的一些参考资料以及**本项目基于Qt的初始界面代码**，供同学们参考。但是本项目的GUI开发**并不限制于Qt**。GUI可以用Qt之类第三方库实现的图形窗口，也可以用ncurses之类库实现的终端窗口，简而言之就是任意发挥，界面是否漂亮/酷炫不算在基础功能中。但这里我们需要强调，你实现的项目应该有C/C++代码文件，使用完全不涉及C/C++的引擎并不符合本课程要求。
- 本手册中，Question内容是我们对你们可能问题的预演和回答。Note内容是对我们认为可以自由发挥部分的进一步说明，仅供参考。Extension内容是第四阶段才要求的拓展功能。

Contents

1. 背景

2. 技术指标： 阶段一

3. 技术指标： 阶段二

4. 技术指标： 阶段三

5. 技术指标： 阶段四

6. 分数构成

7. 一些问题

8. 参考资料

地图

- 地图是游戏的舞台。
- 地图在逻辑上是一个 $m \times n$ 的矩形，存在边界。每个 1×1 的矩形是一个格子。格子可以共用，即多个单位可以处于同一个格子中。
- 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上下左右邻接的，并且序列中不存在重复的格子。
- 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。

地图

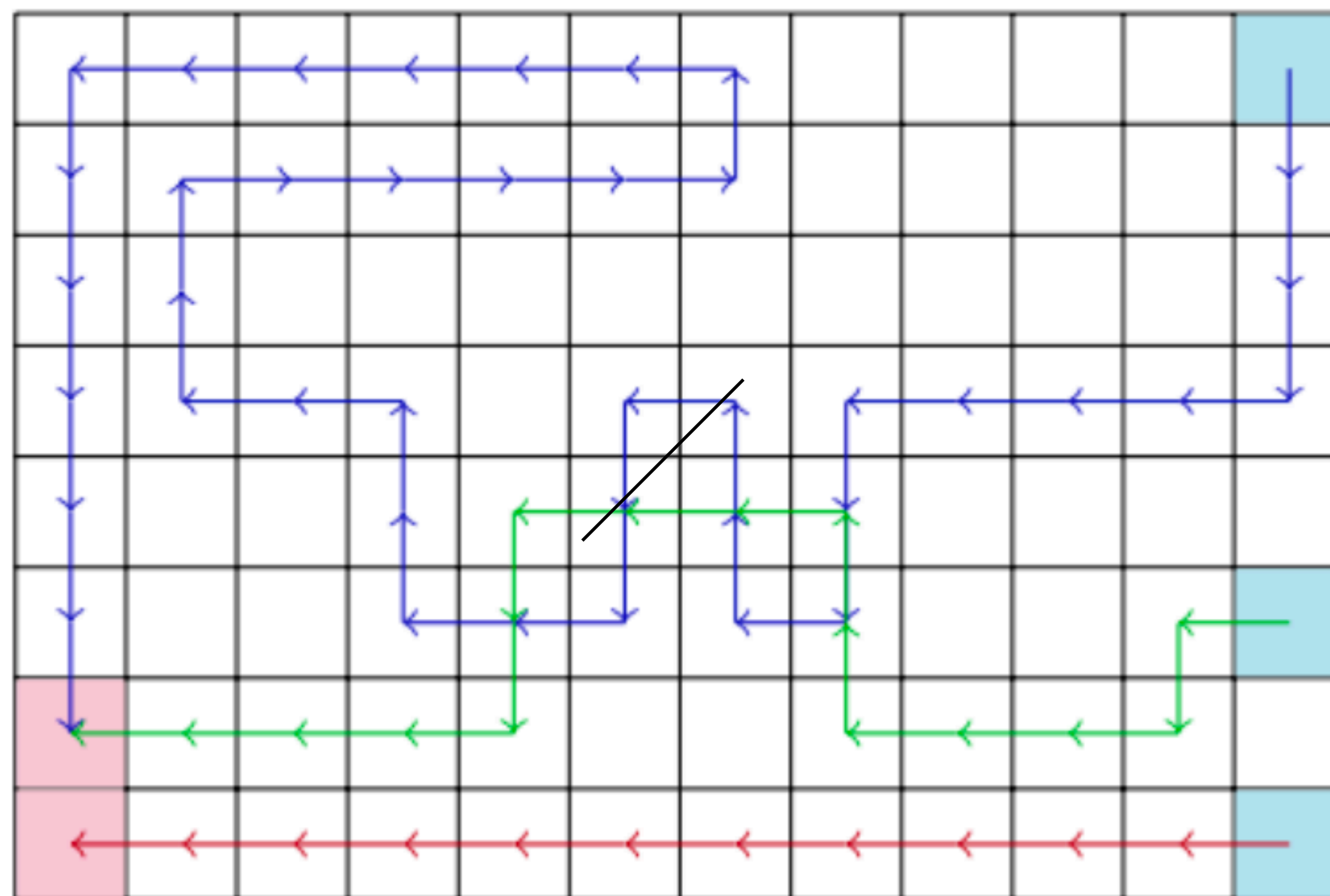


Figure: 地图、格子和路径的示意图。图中有红、绿、蓝三条路径。路径的起始格子被标记成蓝色，结束格子被标记成红色。

阵营

- 游戏中存在两个阵营，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。
 - ▶ 敌方的胜利目标是在有限次攻击之内穿过地图。
- 每个阵营都可以产生具有一定功能的单位。
- 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。
- 每个单位都有基础属性。

我方

- 我方的单位有近战塔、远程塔。（此时还没有引入阶段二的词缀机制，下述均为默认初始机制）
 - ▶ 近战塔被部署在路径上，攻击范围为1（即只攻击相邻格子的敌人），并且能阻拦敌方单位的前进。
 - ▶ 远程塔则被部署在路径之外，且与路径相邻的格子上，默认攻击范围自行设定。
 - ▶ 近战塔和远程塔都不具备范围伤害效果，一次只能攻击一个敌人。
- 我方单位的基础属性包括：生命值、攻击力。
- 我方每种单位类型里，你可以设定多个具有不同参数的单位。

敌方

- 敌方的单位将在路径的起始点随机生成，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。
- 敌方单位的基础属性包括：生命值、攻击力、移动速度。
- 你可以设定多个具有不同参数的敌方单位。

完成要求（无需提交）

在该阶段完成以下内容：

1. 一张地图，包含起点终点、路径、以及其他格子。注意我方远程塔对格子的布置要求。
2. 我方的（至少一个）近战塔和（至少一个）远程塔，以及部署操作。
3. 敌方（至少一个）单位的初始化以及移动。（整个项目除了词缀机制外，只要求这一种敌人，当然，如果你想另外实现其他的，完全可以）
4. 设定生命值、攻击力等，不要求显示这些，但要有基础的敌我双方对战过程。
5. 实现近战塔、远程塔和敌方单位时，必须使用C++中的面向对象编程（即类与继承）。

Extension

- 强化全局设定

- ▶ 增加多个关卡

- 每个关卡在地图和路线的复杂性上有区别；
 - 可以在开始游戏时由玩家自主选择关卡，或者通过前一关后解锁下一个关卡。

- ▶ 增加我方单位复活次数的设定

- ▶ 增加金币设定

- 例如每局可以通过消灭敌方单位获得金币；
 - 通过消耗关卡中收集的金币，玩家可以下一关中全局强化所有我方单位的基础属性、单位复活次数。

注意，这是永久的强化加成，不会随着一局游戏的结束而消失。

-

Q&A

- 本课程项目有STL、Linux限制吗？

没有。

- 我必须在阶段一实现一个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阶段一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。当然，我们认为一边实现核心逻辑，一边实现对应GUI功能是最稳妥的办法。

- 如果我在该阶段没有实现GUI的话，那么我该如何显示呢？

直接在控制台显示日志就可以。

- 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。

- 我需要把所有的Extension内容都实现吗？

不需要，Extension是供阶段四参考的可选内容。需要注意的是，完成一定量（不是凑代码行数）的Extension就可以得到相应分数，拓展满分并不需要完成所有可选内容。评分标准视具体情况而定。

Contents

1. 背景

2. 技术指标： 阶段一

3. 技术指标： 阶段二

4. 技术指标： 阶段三

5. 技术指标： 阶段四

6. 分数构成

7. 一些问题

8. 参考资料

地图的导入与导出

- 在阶段一中，我们可能只硬编码（hard code）了一张地图。但你应该已经自己设计了一套格式用来表达地图，包括尺寸、路径、起点终点位置等。此时只要添加导入导出的接口，在演示时修改、加载地图即可。

存档的导入与导出

- 在阶段一中，我们可能每次打开游戏都要重新开始。现在，我们要能让游戏支持存档的导入和导出。你只需为自己设计一套格式用来表达存档，包括地图、我方与敌方单位的游戏状态等，演示时修改、加载存档即可。

NOTE

- 被导出和等待导入的存档应当是具有特定语法的文本文件（可以理解成一种配置文件）。这样，玩家可以：
 1. 首先导出当前的存档
 2. 然后可以参考其格式，对其进行修改后成为新的存档；
 3. 最后将新的存档导入到游戏中并正常游玩。
- 各位同学可以自由设计自己的存档的语法。在这个语法下，一个合法的文本文件就唯一对应于一张存档。

词缀机制

在阶段二中，我们引入了词缀机制来丰富单位。

- 敌方单位被攻击后可以随机掉落词缀，我方拾取词缀。
- 当我方单位生命力低于一定值时，会掉落词缀，敌方拾取词缀。
- 拾取的词缀保存在词缀库中（你可以设计自动拾取，也可以设置拾取某类词缀后无数量限制，这都随意。但是很显然，你必须能通过选中词缀和单位进行安装/卸下）。
- 我方无论是近战塔还是远程塔都有两个词缀槽，一个槽可以安装一类词缀。
- 安装后的塔的属性（伤害、攻击范围、攻击间隔、其他机制）将发生改变。

我方词缀

- 近战塔，三种基本的词缀：
 - ▶ 狂暴的：附加狂暴属性，装备后伤害和攻击间隔都大幅增加（比如简单设定为都乘以2）
 - ▶ 冰系的：附加冰冻效果，装备后攻击能使得敌方单位停止攻击、移动一段时间
 - ▶ 群伤的：附加范围伤害，装备后能同时攻击多个敌人
- 远程塔，一种基本词缀：
 - ▶ 放血的：装备后对敌人的攻击附加流血效果，使得敌人在一定时间内持续扣血
- 根据游戏规则，显然，敌方单位掉落的必定是有益于我方的以上四种基础词缀，此外你也可以设定其他更多词缀。

敌方词缀

- 对于敌方，我们也要求有实现不同的词缀。敌方单位的词缀可以为空，也可以自由组合，为了方便考虑，我们同样规定一个敌方单位只能拥有不超过两类词缀。
- 基本的敌方词缀：
 - ▶ 闪现的：该单位能发动闪现，越过我方近战塔的阻挡前进，但需要有冷却时间
 - ▶ 神速的：该单位的移动速度要超过默认单位
- 同样的，我方单位掉落的必定是有益于敌方的以上两种基础词缀，此外你也可以设定其他更多词缀。

词缀显示

到了这个阶段，如果你有实现相应的图形界面，就需要对不同词缀的单位进行区分了：

- 直接在单位头上用文字标出如“群伤的”这类词缀
- 或者用不同的贴图来区分
- 或者更多有趣的效果.....

完成要求（无需提交）

在该阶段完成下面的内容：

1. 不同类型的格子和地图的导入导出
2. 存档的导入与导出
3. 基础的我方词缀（显然，词缀的安装/卸下操作也是其中一部分）
4. 基础的敌方词缀

Extension

- 实现更多词缀的设定
-

Q&A

- GUI问题

同阶段一。

- Extension问题

同阶段一。

Contents

1. 背景
2. 技术指标： 阶段一
3. 技术指标： 阶段二
- 4. 技术指标： 阶段三**
5. 技术指标： 阶段四
6. 分数构成
7. 一些问题
8. 参考资料

回顾

在此之前我们已经完成了一个支持词缀的塔防游戏，如果你此前没有加入GUI，那么你需要在这阶段完成它。此外，你还需要实现一些（容易使用GUI实现的）实用功能了！

展示单位的攻击和受伤效果

- 在阶段二中，我们实现了游戏的核心逻辑。但是，如果远程单位在攻击时一声不响，近战单位在攻击时不展示任何动作，单位在受到攻击时也不会有任何变化，那么双方单位的血量就悄悄地减少着。这就会损害游玩的体验。因此，在阶段三中，我们需要在游戏的界面中展示双方单位的攻击/受伤效果：
 - ▶ 单位发起、受到攻击时，需要绘制出相应的效果。

NOTE

- Tips:

- ▶ 不必播放一整段的动画，可以简单地把表示这个单位的图形闪烁一下；
- ▶ 或者可以在它的头上飘过一个数字，表示这个单位受到了多少伤害；
- ▶ 又或者在头上绘制血条显示；
- ▶ 非常欢迎其他的想法。总之，它只要动了就可以。

提交要求

在该阶段完成下面的内容：

1. 图形界面显示前面阶段的各种单位的部署以及单位的移动
2. 展示单位的攻击/受伤效果

Extension

- 远程攻击显示
 - ▶ 对于我方或敌方远程单位,还可以表示出它的伤害区域。如果攻击不是立即生效的, 可以发射一些飞向伤害区域的飞行物; 如果是立即生效的, 也可以绘制一道光束之类。
-

Q&A

- 我必须在阶段三实现一个GUI吗？

是的。但如果很喜欢终端窗口的话，也可以用 ncurses 之类的库。如果不使用相关的库的话，可能需要自己重新造轮子来完成规定动作。如果某些动作在已有的程序架构下难以完成，强烈建议换一种展示思路或者实现思路，也可以不实现，但不实现可能会影响项目的分数。

- 终端GUI扣分吗/用字符画GUI可以吗？

不扣/可以，但做起来不一定能简单。

Contents

1. 背景

2. 技术指标： 阶段一

3. 技术指标： 阶段二

4. 技术指标： 阶段三

5. 技术指标： 阶段四

6. 分数构成

7. 一些问题

8. 参考资料

回顾

一个“有趣”的塔防游戏已经完成了。为了让它更加的有趣，此阶段是扩展阶段，你可以单纯按照我们提供的Extension内容来补充，或者实现任何你觉得很酷的想法。

一些参考

- 每一阶段可选的Extension
- 更多自由发挥：
 - ▶ 可以为游戏添加适当的音效
 - ▶ 可以设计酷炫的动画来提升用户体验
 - ▶ 任何你觉得有意义的都可以

Q&A

- 加个故事背景算拓展吗？

做成图鉴之类的，只要有代码量就算，只加了张图不算。

- 可以在词缀外增加技能机制吗？

可以，算拓展。

Contents

1. 背景

2. 技术指标： 阶段一

3. 技术指标： 阶段二

4. 技术指标： 阶段三

5. 技术指标： 阶段四

6. 分数构成

7. 一些问题

8. 参考资料

分数构成

- 基础功能（前三阶段规定必须完成的内容）： 70%
 - ▶ 注意阶段三里的基础功能只占5%，这是为了照顾能力有限的同学，毕竟课程的主旨还是在代码设计上，花费太多功夫在GUI上略有不妥
- 选做内容： 20%
 - ▶ 每个选做内容的难度不同，会根据实际情况给相应的分数
- 代码风格： 10%

评分标准

- 评分标准包括：
 - ▶ 是否完成了所有规定动作
 - ▶ 是否有一个好用的（而不关心好看的）界面
 - ▶ 是否实现了有趣的（而不关心凑数的）功能
 - ▶ 代码是否符合面向对象和泛型程序设计范式的要求

Contents

1. 背景

2. 技术指标： 阶段一

3. 技术指标： 阶段二

4. 技术指标： 阶段三

5. 技术指标： 阶段四

6. 分数构成

7. 一些问题

8. 参考资料

如何提问/提出问题的错误方式

- (不相干的问题) 电脑出现了故障应该如何处理。

请联系有能力处理这些故障的人（例如：<https://itxia.club/>）而不是你的程序设计课的助教。

当然,如果是和编程强相关的、软件方面的故障,而你自己毫无头绪,你当然可以问一问。

- (可以在搜索引擎里非常轻松地找到答案的问题) 编译器抱怨发现了一个 error 并停止编译。

编译器讲的是人话(虽然是英语), 真的。我们不想成为复读机, 尤其是当问题的答案就在问题的描述里的时候。当然,如果你相信问题不出在你的代码上, 互联网上也没有相关的讨论的时候, 欢迎和我们一起探讨。

- 我的程序又bug了, 我怎么办?

原则上, 不应该让别人帮你debug, 如果实在是无法解决 (搜索了全网都没办法解决, 折腾了好久), 可以寻求TA帮助。

不应该一遇到error就找TA寻求帮助。在寻求帮助的时候, 最好需要给出你的运行环境, 你的错误是什么, 你尝试了哪些解决方法。

发生过的问题

- Qt的路径带有中文可能会出问题导致找不到文件；
- 打包提交到课程网站上，验收时下载下来不能直接跑（不只是Qt可能发生），得一个个导入资源文件；
- 验收时跑着跑着崩溃了；
-

被反复问到的

- 可以参考网络代码，但会查重（不限于Github），如果你盯着网络代码改，多半查重会出事；
- 可以参考PVZ只做直线路径，地图导入导出可以不修改尺寸，但不允许只支持尺寸 $1 \times n$ 的地图；
- 可以在拓展里参考PVZ实现更多的单位种类，但基础部分70分要求得按PPT来；
- 参与课堂演示会有bonus，加在100%分数之外，优先选择没有明显参考已有塔防项目的同学上台演示。

Contents

1. 背景
2. 技术指标： 阶段一
3. 技术指标： 阶段二
4. 技术指标： 阶段三
5. 技术指标： 阶段四
6. 分数构成
7. 一些问题
- 8. 参考资料**

图形库推荐

如果对C++的图形库并不熟悉,这里我们从简单到复杂给出三个库供参考:

- FLTK (<https://www.fltk.org/>) 是一个相当轻量级的C++图形库。
- wxWidgets (<https://www.wxwidgets.org/>) 是一个相对轻量级的C++图形库。
- Qt (<https://www.qt.io/>) 是非常流行的C++图形应用开发框架。

其中的每一个都有足够多的文档和示例程序, 互联网上也有足够多的讨论内容, 可供学习使用。

图形库的安装

- 对于 FLTK , Windows 用户可能需要安装MSYS2 (<https://www.msys2.org/>) 才能从中得到预编译的库;
- 大部分的 Linux 发行版通过自己的包管理系统提供打包好的库和头文件;
- macOS 用户可以使用Homebrew (<https://brew.sh/>) 作为自己的包管理器。
- 当然, 总是可以直接下载它的源代码然后自行编译—这其实不是一件困难的事情, 但是想要完全理解整个过程, 需要综合运用各位同学在计算机系的专业课(主要是计算机系统基础)上学到的知识。
- 对 wxWidgets 和 Qt 而言, 在主流的操作系统下都有打包好的安装程序。

Qt基础教程

为了减轻同学们的工作量，我们给出以下一些基础的Qt教程的链接，以及一份使用Qt来实现图形化界面的初始代码包。需要注意的是，以上内容仅供参考，你可以另寻其他更高阶的Qt教程，且本课程完全不强制使用Qt。

- Qt Download: <https://www.qt.io/free-trial>

- 教程参考1: <https://qtguide.ustclug.org/>

(贴合QtCreator这个软件，从最基础的底层原理开始教学，深入浅出，想学的都有)

- 教程参考2: <https://wizardforcel.gitbooks.io/qt-beginning/content/12.html>

(相较于第一个教程简洁了一点，从实际开发app的角度开始)

Qt初始代码包

- 我们提供了一份使用Qt来实现图形化界面的初始代码包(即课程网站中的“proj2”文件)。简单地实现了地图的展示、敌方单位的展示的基础功能。
 - ▶ 请先阅读/proj2/main.cpp中的README，了解代码结构和运行代码的注意事项；
 - ▶ 代码文件均包含详细注释，可自行查看。
- 需要注意的是，这只是一份参考代码，并不意味着你一定要按照参考代码中的结构和逻辑实现本项目（同时也不意味着一定要选择Qt实现图形化界面）。

The End