

高级程序设计

Project1

181220027, 李博瑞

2021 年 10 月 9 日

实验进度

我选择”塔防游戏”作为课程项目. 我完成了文档中要求必做的所有内容.

整体设计

```
/
├── bin
│   ├── Project
│   └── ...
├── include
│   ├── common.h
│   ├── game.h
│   ├── map.h
│   ├── plant.h
│   └── zombie.h
├── src
│   ├── map
│   │   ├── grid.cpp
│   │   └── map.cpp
│   ├── plant
│   │   ├── plant.cpp
│   │   └── shop.cpp
│   ├── zombie
│   │   └── zombie.cpp
│   ├── game.cpp
│   └── main.cpp
├── doc
└── Makefile
```

我完成的塔防游戏主要基于控制台, 其地图大小为 4*4(可随时进行修改), 己方单位包括向日葵 (Sunflower), 坚果 (Wallnut), 地刺 (Spikeweed), 南瓜 (Pumpkin), 敌方单位包括普通僵尸 (Zombie) 和路障僵尸 (Conehead), 其效果与《植物大战僵尸》中的单位效果类似. 游戏采用阳光 (sun) 作为资源, 随着时间流逝会自动增加. 部署单位需要一定的阳光, 手动铲除单位会返还一半阳光.

游戏开始一段时间后, 会在地图右下角自动生成僵尸, 僵尸会沿着两条路径进行移动, 分别经由左下角和右上角, 到达左上角. 如果僵尸到达路径终点 (即将移出左上角), 则游戏失败; 如果玩家坚持了两分钟, 则玩家获胜.

实现方式

- 对于植物, 我将其分成两个部分: 与战斗有关的属性和与购买有关的属性. 战斗方面的属性, 例如血量, 攻击速度等, 因为很多植物都有相似的属性, 只有少数不同, 因此考虑使用派生类, 即建立一个 Plant 类, 各种植物使用其派生类; 对于购买有关的属性, 将其置于 Menu 类中, 进一步使用 Shop 类包含所有的 Menu 对象.
- 僵尸的实现与植物战斗部分的实现类似. 比较遗憾的是, 由于在屏幕上显示时的大小限制, 每个格子只能显示 3 个僵尸, 因此我设定每个格子里最多只能有 3 个僵尸, 如果有新僵尸想要进入这个格子时就会被强行阻塞, 无法进入.
- 可以看出, 所有动作应该都是发生在地图上的一个个格子上的, 因此我创建了 Grid 类和 Map 类, 其中 Map 包含了所有的 Grid, 每个 Grid 中都可能包含若干个 Plant 和 Zombie. 每 1 单位时间 (1/30s) 对地图状态进行更新, 即添加新购买或新生成的单位, 移除血量为 0 的单位, 各单位进行冷却、攻击或移动.
- 最后是 Game 类, 主要在这里对输入进行处理, 并完成 Shop 类和 Map 类的交互.
- 在绘图方面, 我使用 curses 进行终端绘图.