

简介

Java大作业：植物大战僵尸

需求见 `need.pdf`

程序架构与命名规范

文件

- MainWindow: UI
- GameController: 游戏逻辑
- Plant: 植物
- Zombie: 僵尸

UI层读取用户操作，调用GameController的接口进行处理，然后GameController返回新的画面信息交给UI层。也许不要game controller了。

注：今天我觉得gay还是要的！不然太乱了！让gay后入就好了

todo

- 增强鲁棒性，比如输入一亿个僵尸就骂用户一顿
- 植物和僵尸的一些固定属性改成static

bug:

- 被减速之后 由于int向下取整，直接不动了
- 如果有太阳 tick就不走

一些东西

- 强制16:9，不是就按倒霉处理
- 不要按钮了 直接响应鼠标区域

具体实现

架构：Mainwindow, Ground, Plant, Zombie

mainwindow负责绘图，Ground控制场面信息和逻辑

鼠标操作 -- mainwindow的鼠标监听器 -- 发给ground处理逻辑

Timer类每50ms执行一次Ground中的tick函数，处理各种事件，修改场面信息。tick函数的末尾调用repaint重新绘制画面，这是一切事件的核心。

定时的实现方式，以每1秒发射一个子弹为例，GPlant类中的tickTime和tickTimeRaw设定为1000，每调用一次tick，tickTime -= 50，然后判断tickTime <= 0时，执行操作，并执行tickTime = tickTimeRaw重置时钟。

在Ground.tick()中依次处理的内容和实现方式如下：

植物的事件（产生子弹/阳光）：

Plant基类中有tickEvent函数，子类重载这一函数实现具体的事件内容。tick函数中修改tickTime，如果时间

死了咋办：返回false，然后mainwindow日抛，直接渣了这个ground

性能优化

资源预加载，使用static块将图片、音乐等资源提前加载到内存，每次使用直接从内存中读取，避免频繁的IO操作。

优点

采用面向对象编程，代码耦合性很低，便于扩展、修改