大作业报告

2015080121 李在弦

目标

用QT编程做跟植物大战僵尸相似的我自己的游戏。

原理

用鼠标玩游戏。

如何编程和运行，结果



这是打开游戏之后的界面。

创造一个“MyGame”类，然后在此类做一个“MyBackground”函数。

在这个函数做一个“QLabel”，然后用“QPixmap”画背景图。（用了setPixmap函数）

再做一个“QPushButton”，然后用“QIcon”和QPushButton的几个函数（例如 setStyleSheet，setMask 等）画冒险模式按钮的图片。



按下按钮之后，从土里突出僵尸的手，同时按钮发光。

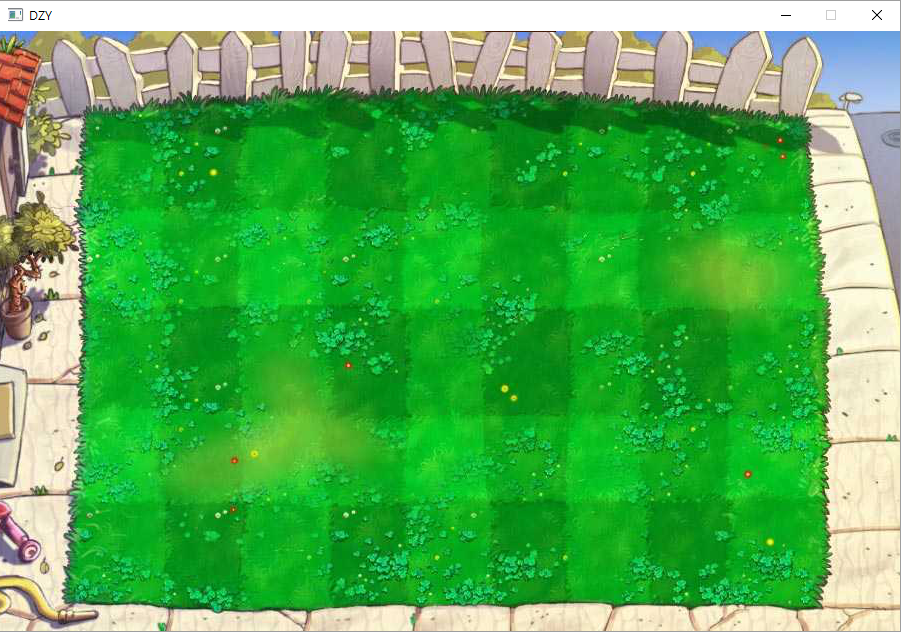
突出僵尸手的编程是这样的，做一个QTimer和QLabel，然后给QTimer定时间（例如 0.1秒），如果QTimer的时间到了就发信号换QLabel的图片。

按钮发光也跟上面一样。



按下按钮后再过5秒之后，删除按钮，换背景。





换背景以后背景图片从左到右移动，然后再从右到左移动。

在这里用的是QPropertyAnimation。（用setDuration函数定时间，用setStartValue函数定初始位置，用setEndValue函数定最后位置，用setEasingCurve函数添加一个效果（背景的移动速度有变化））

背景图片到了某个位置之后，植物卡，铲子，太阳标签，这三个从上下来。

这个也是用了QPropertyAnimation。（这次用的函数跟上面一样）

在这里我的游戏和原来的植物大战僵尸游戏之间有不一样的玩的方式。

我游戏摆放植物的方式是Drag方式。

如果按植物卡就会产生植物图片 (做一个DragItem类，它的父类是QLabel。在mousePressEvent函数中实现按植物卡产生植物图片的功能。如果mousePressEvent函数的参数（QMouseEvent\* event）的位置在植物卡的区域内，会产生图片。)

然后我按着鼠标左键移动鼠标到某个草坪上（用了mouseMoveEvent）

然后释放鼠标左键就摆放成功植物 (用了mouseReleaseEvent。首先换背景图片时创造一个QLabel型的二维数组（QLabel \* MyBox[9][5]），每个MyBox[i][j]指的是 i 行 j 列的草坪，如果mouseReleaseEvent的参数（QMouseEvent\* event）的位置在某个草坪的区域内，就会摆放植物。)



这三个标签下来之后做一个QTimer，做这个QTimer后再过15秒后僵尸开始出来。

各个僵尸有他自己的类。

首先在MyGame类中做 ClassicZombie \* CZombie[5] ， PoleZombie \* PZombie[5] ， BucketZombie \* BZombie[5]。三个标签出现之后，在看不见的窗口的右侧（x坐标大于等于900的区域，窗口的宽度是900）创造总共15个僵尸（每种僵尸每行一个，比如CZombie[0]是在第一行，CZombie[1]是在第二行，这样的），如果跟各自僵尸响应的QTimer发信号时他们开始动。

普通僵尸的生命值是100，他的攻击力是30（每次普通僵尸攻击植物时，植物的生命值会减小30。），他从窗口的最右边到最左边移动时需要90秒（用QPropertyAnimation实现僵尸的移动）。他的生命值减少到10时，他会丢头 (做新的 QLabel \* head)，他的生命值达到 0 时他会死 (在普通僵尸的类中做一个QTimer。僵尸出现后每次过0.1秒时会发信号确认他的生命值是不是10，还是0)

铁桶僵尸的生命值是170，他的生命值达到100之前的攻击力是50，达到100之后他的攻击力会减小到30，同时他会丢铁桶。他从窗口的最右边到最左边移动时需要75秒（实现功能的原理跟普通僵尸的的一样）。他丢头，死亡的条件也跟普通僵尸的的一样，查生命值的方法也一样。

撑杆僵尸的生命值是150，他的攻击力是40。他有他自己的特点，第一次遇到植物时跳过植物，然后会攻击植物（用QMovie和QLabel换他的动作，同时用QPropertyAnimation实现跳过植物的功能）。他跳过植物之后他的移动速度变很慢。别的基本功能跟上面一样。

各个植物也有它自己的类。

豌豆射手的生命值是120，他的攻击力是10 (豌豆射手射的子弹跟僵尸冲突时，僵尸的生命值减小10。)。如果在跟它同行上有僵尸，而且他们正在动，它会发射子弹。如果不是，不会发子弹。在它的类中做一个长度为10的豌豆子弹数组。做一个QTimer \* CheckPea，时间间隔是0.01秒。每次过0.01秒后发信号查询每个元素是不是存在。如果不存在，以时间间隔为1.4秒的PeaTime（QTimer \* PeaTime）发信号，在CreatePea曹创造新的元素。如果子弹跟僵尸冲突时会被删除（delete PeaL[i]）。做确认子弹跟僵尸是不是冲突的功能的函数是在MyGame类中的CheckPZ曹，它收到CheckPeaZombie（QTimer \* CheckPeaZombie , 它发信号的时间间隔是0.01秒）的信号之后确认子弹跟僵尸是不是冲突了。

向日葵的生命值是120，它被摆放在草坪上之后，在它类中的QTimer（QTimer \* A ， 时间间隔为20秒）发信号时会生产阳光，点击阳光的话，阳光移动到太阳标签（这个也是用QPropertyAnimation实现的），然后阳光值会增加25。

坚果墙的生命值是300，他不会攻击僵尸，但用它可以争取时间。



如果僵尸都死了，就胜利



如果任何一个僵尸的x坐标小于0时，就失败。

结论

通过这次作业，我会了解怎么用QT了，还有这个作业给了我能提高编程序能力，解决问题的能力的机会。下次有机会想试一试做别的游戏。