南开大学 计算机大类

姓名:胡博浩

学号：2212998

班级:1012

2023年5月13日

高级语言程序设计

实验报告

目录

[**高级语言程序设计大作业实验报告 3**](#_Toc27543)

[**一、 作业题目： 3**](#_Toc7082)

[**二、 开发软件： 3**](#_Toc20029)

[**三、大体结构： 3**](#_Toc13852)

[（1）主界面： 3](#_Toc26002)

[（2） 游戏帮助界面：查看游戏规则和操作说明 4](#_Toc1499)

[（3） 游戏设置界面：打开或关闭音乐。 4](#_Toc17972)

[（4） 关卡界面：实现连续闯关 5](#_Toc23038)

[（5） 其它： 6](#_Toc14032)

[（6） 资源文件： 6](#_Toc14032)

[**四、具体代码实现： 7**](#_Toc21439)

[**五、遇到的问题： 1**](#_Toc17854)**1**

[（1） 界面的切换问题： 1](#_Toc21855)1

[（2） 关卡界面暂停后人物还能移动： 1](#_Toc24833)1

[（3） 游戏暂停后再开始，无法读取键盘事件： 1](#_Toc8509)1

[（6）音乐播放故障： 1](#_Toc16147)2

[**六、游戏的优势与不足： 1**](#_Toc29212)**2**

[**七、 收获： 13**](#_Toc25848)

[（1）git代码上传 ： 13](#_Toc7332)

[（2） Qt的使用： 13](#_Toc19046)

[（3） 项目的创建，开发和打包： 14](#_Toc23876)

[**八、 感想： 14**](#_Toc16620)

高级语言程序设计大作业实验报告

**一、** **作业题目：**

一款基于马里奥的高清横版2D闯关游戏

**二、 开发软件：**

Qt6.5.0（编程工具），git（代码上传工具）

**三、大体结构：**

（1）主界面：

主界面的类：mainwindow

主界面构成要素：

界面大小、图标、标题设置。

背景图片。

开始游戏按钮，退出游戏按钮，游戏帮助按钮，游戏设置按

钮。



1. 游戏帮助界面：查看游戏规则和操作说明

游戏帮助的类：gamehelp

游戏帮助界面构成要素：

界面大小、图标、标题设置。

背景图片。

返回主页按钮。



1. 游戏设置界面：打开或关闭音乐。

游戏设置的类：gameset

游戏设置界面构成要素：

界面大小、图标、标题设置。

背景图片。

打开音乐按钮、关闭音乐按钮、返回主页按钮。



1. 关卡界面：实现连续闯关

①关卡类：level1、level2、level3

②游戏暂停窗口（gamepause）：实现游戏暂停功能

构成要素：

窗口大小、位置设置

背景图片。

继续游戏按钮，重新开始按钮、退出游戏按钮。

③方块类：brick、unknown（包括金币coin）

④敌人类：master、bullet（子弹）、spikeweed(地刺)、kuba、shell、cannibal

⑤道具类：mushroom、flower、rainbow

⑥其他类：castle（城堡）、fire（火球）、pipe（管子）、flag、flagpole、bird

⑦主角类：mary





1. 其它：

①global：实现全局变量、函数的调控

②按钮类（mypushbutton）：设置按钮的配置

③音乐播放类（musicplayer）：实现音乐的配置、播放等

（6）资源文件：包括music（音乐）、photo、sound（音效）

**四、具体代码实现：由于文件较多，代码较长，仅论述几点，详情请前往gitee查看**

（1）游戏运行逻辑：timerEvent

游戏运行的核心在于游戏画面的实时更新，利用定时器来保证人物、敌人移动，判定碰撞、死亡事件，实现界面切换、画面播放及延迟效果。

（2）画面播放：paintEvent

所有的动画效果均由利用定时器快速播放绘制的每一帧内容来实现。

（3）实现主角的移动和跳跃动画、子弹发射、快跑等：KeyPressEvent、KeyReleaseEvent

①由mainwindow捕获键盘事件，播放朝什么方向的动画由按下的按键决定：right,left,space

（其中跳跃is\_jump，is\_jump\_end，is\_space\_release三个bool数据，保证mary只能一段跳且跳跃动画正常）

②然后若用于判断是否可以射击的bool型数据为1（主角通过吃两次蘑菇或获得花）并且按下x键就可以发射子弹。

外挂：w键可无限制释放子弹。

子弹释放方向以mary方向为准，释放位置为mary的位置。

③快跑的实现：按下移动键（right或left）的同时按下z键，主角朝着当前方向快速移动。

④主角吃道具（例如吃蘑菇加速，吃彩虹星变大并处于无敌状态15秒）后，主角移动画面相应改变。

（4）碰撞的判定：

由于Qt缺少相关的物理引擎，加之本人技术水平有限，只能采取矩形碰撞检测法。通过对不同情况分类讨论（主角跳跃或仅移动，目标物体是否移动，主角是否变大），根据游戏实际运行结果进行检测，粗略实现碰撞的判定。

（5）人物移动效果的实现：

将人物移动中一连串的图片在极短的时间内逐一释放，从而达到逼真的移动效果。

（6）人物跳跃效果的实现：

设置两个参数（height，distance），利用定时器使主角每次移动距离改变，从而达到有加速度的效果。

if (distance > 0) {

y -= height - distance;

distance++;}

1. 碰撞效果的实现：

①主角碰到物体被反弹：x不变，y改变。

②方块破碎：方块消失；方块破碎后的残片往两边先斜上升、在下落；残片消失。

③怪物被压扁：绘图时怪物图片长不变，高减半；且之后怪物消失。

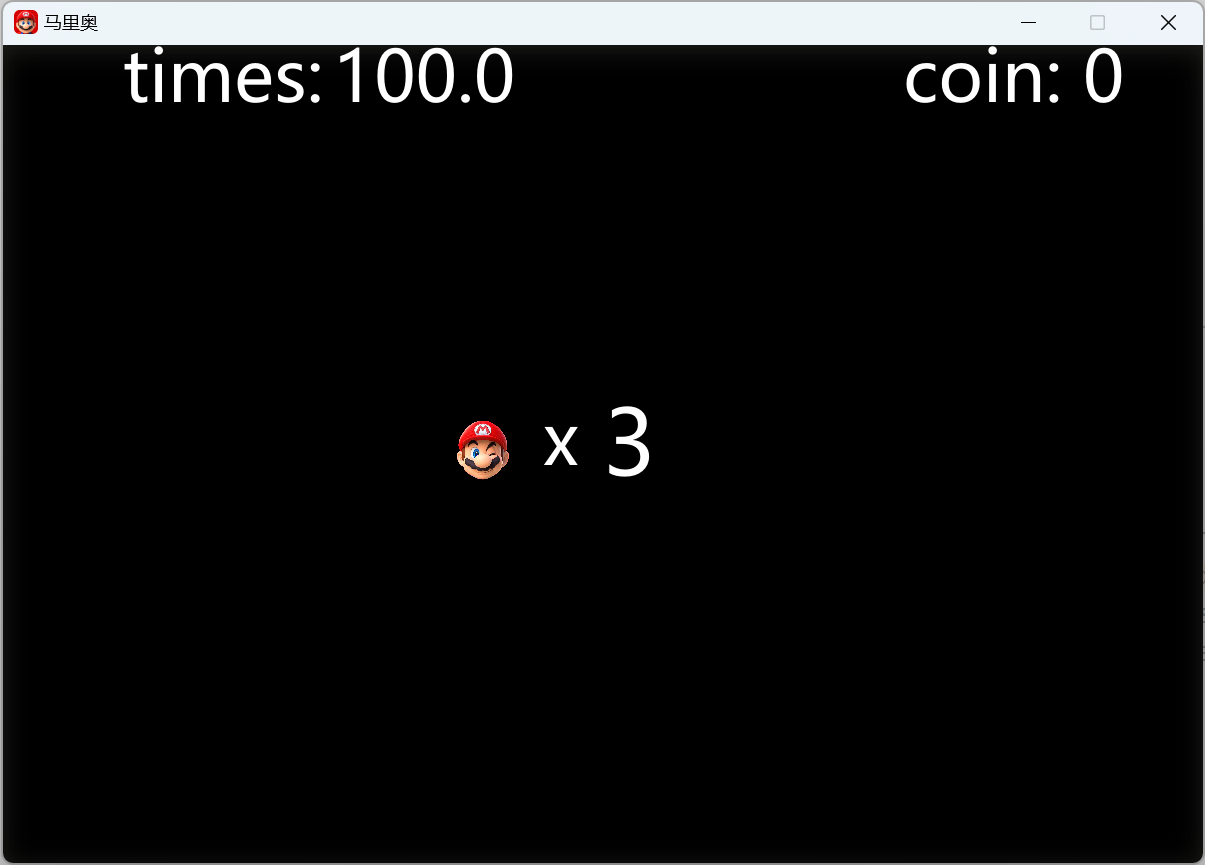
（8）死亡效果的实现：

将人物死亡时一连串的图片在极短的时间内逐一释放；人物死亡过程中x不变，y先上升后下降；人物消失，播放游戏准备画面。

（9）复活效果的实现：

人物在无和有中切换，持续一段时间。即绘画人物时先画，再不画，再画......（设置is\_invincible参数，随时间改变，奇数时不画，偶数时画）。

（10）在关卡切换和人物死亡、复活间隔的过程中，播放游戏准备画面。



（11）关卡胜利画面：

关闭所有定时器；若不是最后一关，延时切换到下一关；若为最后一关，延时返回主页面。



（12）游戏失败画面：

关闭所有定时器，延时返回主页面。



（13）加入背景音乐，游戏音效：

定义一个musicplayer类，在其中定义控制音乐的一系列函数；在global类中声明并定义一系列string变量，将music、sound资源的相对路径存放其中；在需要音乐的地方添加音乐，并在适当位置关闭音乐。

**五、遇到的问题：**

1. 界面的切换问题：

由于一开始对Qt的学习不扎实，对于信号和槽这一块儿没有掌握好，切换界面时原界面的东西还存在。后来学习了this指针，show、hide、back函数，在切换界面时利用信号和槽机制，实现了界面的出现、隐藏、返回，消去了这个大bug。

1. 关卡界面暂停后人物还能移动：

使用setWindowModality函数，禁止其他界面响应，只有退出游戏暂停界面其他界面才能响应。（有点类似恶性弹窗）

this->setWindowModality(Qt::ApplicationModal);

1. 游戏暂停后再开始，无法读取键盘事件：

可能是由于第2点的解决方法的问题，暂停后重新开始或者继续游戏，人物移动出现bug。后来发现是键盘事件的读取出现了故障，在关卡初始化时将它的焦点策略设置为“强焦点”后问题解决了。

this->setFocusPolicy(Qt::StrongFocus);

1. 各种类似于人物、敌人以意料之外的方式移动、攻击或者消失等：运行项目，检测结果，修改代码循环进行，直到问题得到解决。

（5）在子弹、敌人、道具消失时忘记break（只是绘图不出现，但它依旧存在），那么主角仍然会判断子弹、敌人、道具，会导致bug；绘图时人物、方块等图片与实际坐标不一致。这种情况比较好处理。

（6）音乐播放故障：

由于是新版Qt，与音乐播放相关的类变化较大，之前播放方法不能适用。然后采用QSoundEffect类播放音乐，发现音乐还是不能播放。查找资料发现需要转为wav格式，转完之后有些音频仍不能播放。最后发现是比特率过高的问题，降到705比特率后问题完美解决了。

**六、游戏的优势与不足：**

1. 优势：

宏观上看：

（1）内容丰富且游戏流程完整。

（2）画面清晰且美观。

（3）敌人种类丰富多样，变化较多。

（4）道具多样，富有可玩性。

（5）拥有丰富的细节、游戏视觉效果好。

（6）操作流畅顺手，富有游戏性。

（7）创新且富有创意。

代码上看：

（1）分类得当，易于修改、补充。

（2）代码封装性好，且进行了多次的优化，经过大量的、充足的、多方位的测试。

（3）变量名函数名直观易懂，有详细的、足量的、易懂的注释，可读性强。

1. 不足：

（1）工程量大

（2）代码较为复杂

（3）仍存在一些bug

**七、 收获：**

（1）git代码上传 **：**

由于一开始不太了解git的使用方法，在初次代码时花费了很多时间。除了给的参考资料外，其他问题主要是上网查找资料加上自己不断摸索解决的。

1. Qt的使用：

学会了窗口，按钮控件，信号和槽机制，lambda表达式，对话框，键盘事件，定时器，绘图事件等的使用。但是对于ui界面的使用还是不熟悉。

1. 项目的创建，开发和打包：

从创建项目，建立头文件、源文件，到资源文件的导入及使用，从窗口的创建到界面之间的跳转，从类的定义到函数之间的相互作用，经过这次游戏的开发，我学会了一整个程序开发的流程，更加深刻地理解了类、函数、指针的作用，对代码有了全新的认识。

1. **感想：**

从一个月前开始学习Qt，学习了一周时间后就开始敲代码。那个月……在我的记忆里挥之不去。非常累，在每天满满的行程下，还要抽出时间打代码。尤其是在项目创建初期，满脑子都是如何完善游戏，就连做梦都是在想怎么修bug。第二天一起床就迫不及待地把梦里的想法拿去实现一下（居然还有几次是成功了的）。有时候我会想，为了一个可能根本不会有人玩的游戏花这么多心思，花这么多时间到底是为什么，就为了那可怜的十分吗？后来随着游戏的日益完善，在目睹了游戏运行效果的渐渐提高，获得了一次又一次的成就感后，我才渐渐明白，在过程中得到快乐，让我的角色或多或少地动起来，这就足够了。