**高级语言程序设计**

**实验报告**

****

**南开大学 计算机学院**

**刘轩麟 2312289**

**2024年5月2日**

**目录**

1. **作业题目…………………………………………………………3**
2. **开发软件…………………………………………………………3**
3. **课题要求…………………………………………………………3**
4. **主要流程…………………………………………………………3**
5. **整体流程………………………………………………………3**
6. **具体实现………………………………………………………4**
7. **开始及准备界面……………………………………4**
8. **选择人物………………………………………………5**
9. **创建人物、球及球场背景………………………5**
10. **实现人物和球的运动………………………………5**
11. **加入碰撞检测及反弹效果………………………6**
12. **记录比赛时间及比分………………………………7**
13. **添加音效………………………………………………7**
14. **添加解说和庆祝彩蛋………………………………8**
15. **测试…………………………………………………………………8**
16. **收获…………………………………………………………………8**
17. **多文件协同……………………………………………………8**
18. **Qt基本功能和写法…………………………………………8**
19. **解决实际项目问题的能力…………………………………9**

**高级语言程序设计实验报告**

1. **作业题目**

SoccerGoal：双人足球对战小游戏的实现

1. **开发软件**

Visual Studio 2022、Qt Creator

1. **课题要求**

1）面向对象

2）多文件协同

3）模型创建

4）物理效果实现

5）测试验证

**四、主要流程**

**1.整体流程**

实现思路：

在用Qt的基本功能实现图形化界面的创建后，实现多界面交互，再利用多种方法完成足球比赛的效果实现。

创建的类如下

player： ball:

qreal v\_jump qreal x

qreal v\_run qreal y

QString name const qreal r

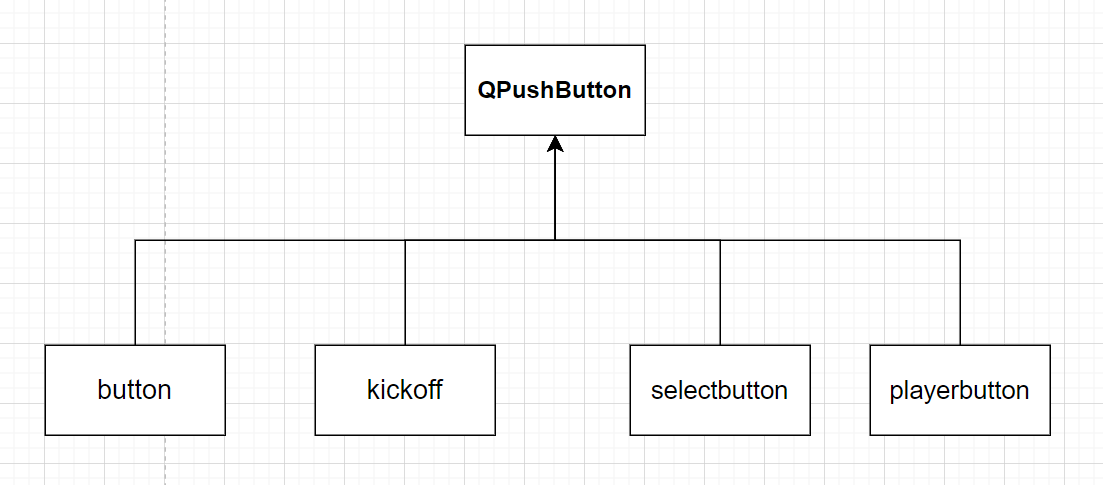
qreal left\right\top\bottom qreal vx

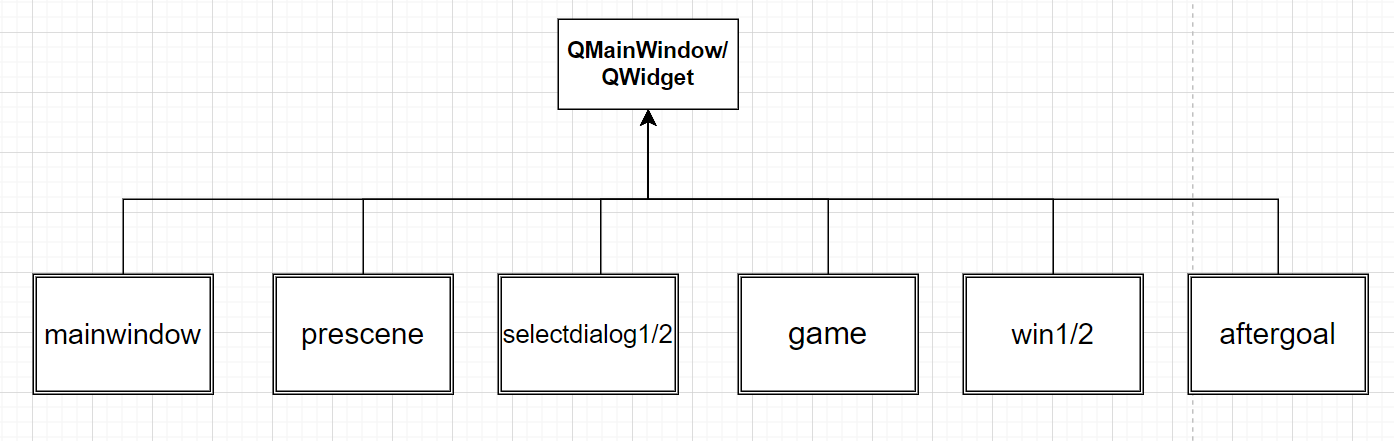
qreal gravity qreal vy

QString [30] scores qreal f

qreal g

qreal left/right/bottom/top





**2.具体实现**

（1）开始及准备界面

在网络上学习Qt的基本操作，利用重写paintEvent函数美化界面。在按钮设置方面，进行了动画设置，重新编写按钮的构造函数，通过动画效果两次加载图片实现按钮的按下和弹起效果。准备界面（prescene.cpp）先将开球按钮设为disable，只有双方都选择好球员后才可以按下；选择球员的按钮在完成选择后也被设为disable模式，避免重复选择。效果如下：



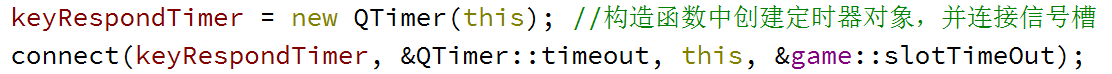
（2）选择人物

从网络上下载了一组相同风格的球员图片，保持风格的协调一致。在准备球员界面加载了十名球员的卡通图片，丰富了玩家的选择。

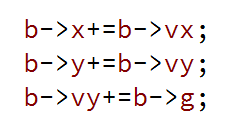
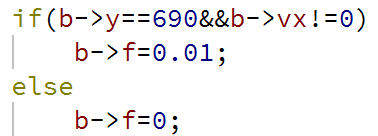
（3）创建人物、球及球场背景

创建人物用到了Qt关键的信号与槽机制，在选择好球员后，球员选项按钮会发射信号，“prescene”文件通过connect将球员与“game”场景联系起来。如图：利用槽函数在QLabel上显示出球员（每个球员的身高，宽度、速度及弹跳均有所不同）。同样，足球也是在QLabel上设置了图片以及合适的大小，显示在场内。由于本人绘画水平有限，为了得到满意的背景图，所以邀请了朋友帮忙绘制球场背景，自己根据背景图记录了地面和球门的位置，以及球门的高度，便于进行下一步设定。

（4）实现人物和球的运动

人物的运动主要通过键盘事件控制，将不同人物设置了不同的移动速度和弹跳速度，由于Qt常规的键盘事件无法同时捕获多个按键，会使对战游戏的体验感变差，所以我改用定时器和槽函数做连接，将部分键盘事件写在槽函数中，这样既解决了同时捕获多个按键的问题，又使球员的连续移动更加流畅。为了实现球的运动，我设置了水平、竖直两个方向的速度，用到了timerEvent函数，先设置定时器，定时在函数中进行较快的update，实现足球连续移动的效果，如果球在地面上，我也设置了

相关的判定，增加了“摩擦力”，在空中时，也增加了“重力”，使上升下落过程中每次刷新时足球移动的距离不同，模拟真实的物理运动。如图：

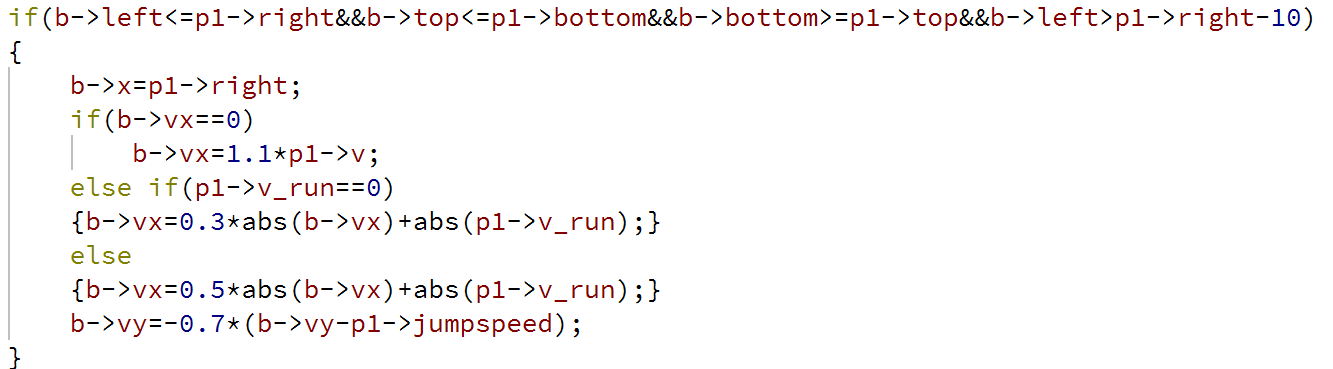


同时，timerEvent函数也启发我将人物弹跳时设置“重力”，使跳起效果更加真实。如下图：

（5）加入碰撞检测及反弹效果

我用矩形边界来实现碰撞检测，由于每次更新有时间差，碰撞检测并不能简单的用“等于”来实现，而应规定好边界所处的范围，这一点在我反复的测试中逐渐优化，尽管仍有不足，会出现不真实的情况，但已经比一开始进步太多。反弹效果上，我设置了每次反弹水平、竖直速度方向和大小的变化，也设置了球场左右边界的位置，球在地上反弹时，我设置了当竖直速度大小小于某一值时，球会停止在地面上，避免反复的振动。碰撞检测和反弹效果需要多次试验，获得较优解，因而是整个程序中较难的部分，耗时也较长。

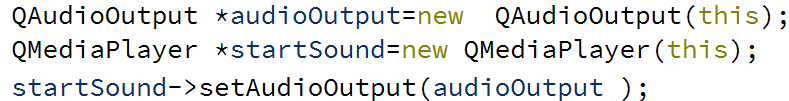
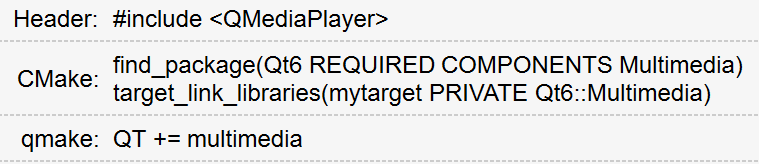
部分实现如下图：



（6）记录比赛时间及比分

Qt提供的QLCDNumber是一个可以创建计时器外观组件的类，各种方法的实现也较为容易，比分存在数组中，进球后会向后读取。

（7）添加音效

由于中途将qmake构建改为cmake构建，因此“添加多媒体”的方法有所不同，需要在CMakeLists文件中添加，添加方法可以在Qt帮助文档中查到。在文件编写时，也许是Qt版本不同（我最后用的是Qt6），多次试验后，发现用QAudioOutput类才能将声音放出来。循环播放等操作虽然与之前版本有所差异，但也可以根据Qt帮助文档查到相关用法。

（8）添加解说和庆祝彩蛋

为了进一步增加游戏的趣味性，在进球界面我加入了庆祝功能，点击按钮会显示一段动图，每名球员都有专属的庆祝动图；比赛结束后，会自动播放一段贺炜老师的解说，我一共截取了12段，播放是随机的。

**五、测试**

Release版本我上传到了GitHub网站（[30SC35KD/SocceGoal\_Grade1-FinalAssignment\_release\_version (github.com)](https://github.com/30SC35KD/SocceGoal_Grade1-FinalAssignment_release_version)），有需要可以下载。相关演示效果可以观看我在b站上传的视频 （https://www.bilibili.com/video/BV1WE42157Vy/?vd\_source=3282e7843a3d26537409ccf71b9fdd10）。

**六、收获**

1.多文件协同

在“高级语言程序设计”课程的学习中，我们接触到的编程往往只有一个.cpp文件，但这次的大作业给了我们独立完成一个project的经历。创建多个类，在头文件中定义属性，在cpp中实现相关功能，体现了c++语言面向对象的特性，使得整个项目更加清晰，也更具可维护性。

2.Qt基本功能和写法

除了提升编程能力，大作业让我接触到了Qt这一图形化编程的工具，掌握了如QPaintEvent、QTimer、QPushButton、QMediaPlayer等多种Qt类的使用方法，也学习了信号与槽机制，为图形化编程打好了基础。

3.解决实际项目问题的能力

由于是第一次独立完成项目，在实现过程中，经常会遇到问题，以上所提到的功能实现，都是在反复的试验或查询后才完成的，遇到的问题有时会困扰我很长时间，前前后后历时一个多月，我才完成了这份大作业。尽管时间较长，但在这个过程中，我自身发现问题并解决问题的能力大大提高，创新进取的意识不断地被激发，踏实沉稳的心性也有所培养。在之后面对项目或者遇到难以解决的困难时，这次经历带给我专业能力的提升以及意志品质的打磨都会有所帮助。