南开大学 计算机大类

姓名 卻铭恺

学号 2012411

班级 2020级智能一班

2021年5月9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](#_Toc71500938)

[**一.** **作业题目** 3](#_Toc71500939)

[**二.** **开发软件** 3](#_Toc71500940)

[**三.** **课题要求** 3](#_Toc71500941)

[**四.** **主要流程** 3](#_Toc71500942)

[**1． 整体流程** 3](#_Toc71500943)

[**2． 算法或公式** 4](#_Toc71500944)

[**在工程内的source.h中定义了很多全局变量，用于计分、存档等。** 4](#_Toc71500945)

[**（1） 计分机制** 4](#_Toc71500946)

[**（2） 音符判定实现** 6](#_Toc71500955)

[**（3） 动画实现** 7](#_Toc71500957)

[**（4） 存档读档实现** 7](#_Toc71500959)

[**（5） 各种信号与槽的链接** 7](#_Toc71500960)

[**（6） 谱面下落动画** 7](#_Toc71500962)

[**3． 单元测试** 7](#_Toc71500964)

[**五.** **单元测试** 7](#_Toc71500965)

[测试结果 9](#_Toc71500966)

[**六.** **收获** 9](#_Toc71500967)

[**1． 文件读取操作** 9](#_Toc71500968)

[**2． 模板类的知识学习** 9](#_Toc71500969)

[**3． 在Qt中进一步深化对继承的理解** 10](#_Toc71500971)

[**4． 学会了在connect中使用lambda表达式编写程序内容** 10](#_Toc71500972)

[**5． 学会了分文件编写的有关内容** 10](#_Toc71500973)

[**6． 了解了Qt中很多类的使用** 10](#_Toc71500974)

[**七.** **反思** 10](#_Toc71500975)

[**八.** **关于谱面编辑器** 10](#_Toc71500977)

高级语言程序设计大作业实验报告

1. **作业题目**

使用QtCreator制作简单的音乐游戏，与《太鼓达人》类似。

1. **开发软件**

QtCreator5.15.2

1. **课题要求**

学生自选题目，使用C++语言完成一个图形化的小程序。

图形化平台不限，可以是MFC、QT等。

程序内容主题不限，可以是小游戏、小工具等。

1. **主要流程**
   1. **整体流程**

（1）音乐游戏背景介绍：

音乐游戏，又称为节奏游戏。其玩法的核心要领是“跟着节奏点按键（屏幕）”，也就是说，在音乐游戏里，背景音乐是不可或缺的一部分，音乐游戏里的所有操作都跟这段游戏时间内，游戏的背景音乐有关。音乐游戏广泛存在于我们的生活中，涉及平台多样，手机端、电脑端、街机平台上都有制作精良的音乐游戏。音乐游戏的玩法多种多样，有传统的下落式判定音乐游戏，如《节奏大师》、《古树旋律》、《音灵》等；有非常规的下落式判定，如《Arcaea》，《Chunithm》等；有判定线跟随音符移动的判定类型，如《Cytus》、《CytusII》等；有无判定线，只跟着节奏进行游戏的类型，如《太鼓达人》、《节奏医生》、《Osu！》等；还有判定线在屏幕上飞舞的《Phigros》，甚至在还未开放游玩的《Rotaeno》中音乐游戏和手机的陀螺仪联系在了一起，可见音乐游戏玩法多样。本人是音乐游戏深度玩家，在老师、同学们体验or批评指正本人的作品之余，强烈推荐各位去游玩一下比本人的作品更加完美、更加有趣的上述几款游戏。

（2）游戏界面、机制大体构思与编程构想：

大体界面：首先有一个MainWindow类，存放初始界面。然后可以通过初始界面进入PlayScene类选择谱面，进入选择难度，然后开始游戏，游戏结束以后可以查看成绩。程序中拥有与《Arcaea》类似的计分机制与potential机制，在后文会解释。游戏有存档读档机制，但是没有在线模块，它是一个单机游戏。

* 1. **算法或公式**

**在工程内的source.h中定义了很多全局变量，用于计分、存档等。**

* + 1. **计分机制**

在source.h中存在一个get\_score值，用来计算当前歌曲的得分；存在一个get\_potential值，用来计算在当前歌曲中你获得的potential；存在一个get\_level值，用来计算在当前歌曲中你获得的评价。Key数的定义是本首曲目的谱面中共有多少个音符。

**分数计算：**

类似arcaea，满分10000000，连击不算分，每个key算10000000除以key总数的分数。

共有四种判定，分别是max perfect，perfect，great，miss。

perfect的分值是10000000除以key总数，max perfect的分值是perfect的分值+1

great的分值是perfect的分值的50%，miss不得分。

满分为10000000+key数。

**获得的歌曲评价与potential计算：**

if(get\_score < 8000000)

{

get\_level = "迷失";

get\_potential =current\_potential \* (double)get\_score / (double)9500000;

}

else if(get\_score >= 8000000 && get\_score < 8800000)

{

get\_level = "略知";

get\_potential = current\_potential \* (double)get\_score / (double)9500000;

}

else if(get\_score >= 8800000 && get\_score < 9200000)

{

get\_level = "了解";

get\_potential = current\_potential \* (double)get\_score / (double)9500000;

}

else if(get\_score >= 9200000 && get\_score < 9500000)

{

get\_level = "熟悉";

get\_potential = (current\_potential - 1) + ((double)get\_score - 9200000.0)/30000.0;

}

else if(get\_score >= 9500000 && get\_score < 9800000)

{

get\_level = "通晓";

get\_potential = (current\_potential) + ((double)get\_score - 9500000.0)/30000.0;

}

else if(get\_score >= 9800000 && get\_score < 9900000)

{

get\_level = "洞悉";

get\_potential = (current\_potential + 1) + ((double)get\_score - 9800000.0)/20000.0;

}

else if(get\_score >= 9900000 && get\_score < 10000000)

{

get\_level = "洞悉+";

get\_potential = (current\_potential + 1) + ((double)get\_score - 9800000.0)/20000.0;

}

else if(get\_score >= 10000000 && get\_score < 10000000+key\_num)

{

get\_level = "征服";

get\_potential = current\_potential + 2;

}

else if(get\_score == 10000000 + key\_num)

{

get\_level = "制霸";

get\_potential = current\_potential + 2;

}

else

{

get\_level = "错误";

get\_potential = 0;

}

**Potential计算：**

每首歌曲有一个potential的基准值，这个值存在存档文件中。根据不同的歌曲评价，获得的potential如上述代码计算。

* + 1. **音符判定实现**

利用自定义类Note组成QVector<Note\*>Notes，在label中加载pixmap作为音符的展示。利用QTimer实现判定，方法是当Notes的成员变量note\_start\_time与GameWindow中的bgtimer时间重合时启动判定，在±100ms的判定区间内给予单个音符的评价。

* + 1. **动画实现**

利用自定义控件mylabel组成音符单端数组QVector<mylabel\*>float\_key，利用move函数改变mylabel的坐标实现音符流动动画。

* + 1. **存档读档实现**

利用QFile和QTextStream实现存取文本文件。

* + 1. **各种信号与槽的链接**

利用connect实现。

* + 1. **谱面下落动画**

逐次检测，到音符下落时启动动画，详见讲解视频和代码。

* 1. **单元测试**

点击play，选中一首音乐（默认第一首），然后继续play，点击开始玩即开始游戏。期间可以通过点击pause实现暂停。与大多数音游相同，游戏结束后会弹出result界面，可以看到自己获得的成绩，并可以再次进行游戏。

\*更多细节请在程序中体验。

1. **单元测试**

测试案例：

表 1：测试案例

|  |  |  |
| --- | --- | --- |
| **输入** | **反应** | **目的** |
| 在mainwindow中点击play按钮 | 一切正常 | 验证Play是否能召出play窗口 |
| 在mainwindow中点击edit按钮 | 弹出模态对话框（一切正常） | 验证Edit是否起作用 |
| 在mainwindow中点击OPTION按钮 | 弹出option窗口（一切正常） | 验证option按钮是否能召出option窗口 |
| 在mainwindow中点击exit按钮 | 退出程序（一切正常） | 验证exit是否起作用 |
| 在option中更改用户名，头像 | 一切正常 | 验证option是否起对应作用 |
| 在play中点击任意一个toolbutton | 切换封面，播放预览，改变全局变量值（一切正常） | 验证三个toolbutton是否起作用 |
| 在play中点击back按钮 | 回到mainwindow，播放音乐（一切正常） | 验证back是否起作用 |
| 在play中点击play按钮（不点击toolbutton） | 加载paradise的谱面，召出sele窗口（一切正常） | 验证nameofsong和current\_song的默认值 |
| 在play中点击play按钮（点击任意一个toolbutton） | 加载对应的谱面，召出sele窗口（一切正常） | 验证play是否起对应作用 |
| Sele中显示界面 | 一切变量正常显示 | 验证初始化是否起作用 |
| 点击sele中radiobutton或拖动slider、点击spinbox | 不起作用 | 验证调速、选择难度功能是否正常（源代码中尚未实现，不是bug） |
| 点击sele中play按钮 | 一切正常 | 验证谱面是否加载成功，游戏能否正常启动 |
| play中显示界面 | 大体正常 | 音符可以正常下落，分数、连击数label起对应作用，判定正常，但精准度欠佳 |
| Play中点击pause按钮 | 弹出模态窗口paus，阻断游戏进行，等待下一步操作（一切正常） | 验证pause按钮是否起作用 |
| Paus中点击continue按钮 | 继续游戏（一切正常） | 验证continue按钮是否起作用 |
| Paus中点击restart按钮 | 继续游戏（不起作用） | 验证restart按钮是否起作用（起了与continue按钮相同的作用） |
| Paus中点击exit按钮 | 退出游戏界面，显示res窗口并正常显示到此为止的成绩（一切正常） | 验证exit按钮是否起作用 |
| game中点击任意一个pushbutton | 依照时间正常确定判定（大体正常） | 验证游戏能否正常进行 |
| game中点击过pushbutton并等到窗消失 | 正常进行游戏，一切正常 | 验证游戏能否正常进行 |
| game中从不点击pushbutton直到窗消失 | 正常进行游戏，一切正常，但是你没分 | 验证游戏能否正常进行 |
| game的任一种可以导致窗口消失的情况（除了点击x或者alt+f4） | 弹出此时对应的res窗口，一切正常 | 验证成绩能否正常显示 |
| 在res中点击continue按钮 | 窗口消失，显示play窗口（一切正常） | 验证能否进行二次游戏 |
| 二次进行游戏 | 轻微异常，谱面下落会莫名其妙变慢 | 验证能否正常进行二次游戏 |
| 在任意时刻点击存在小猫图片的地方 | 喵 | 验证彩蛋能否正常播放 |

### 测试结果

大多数功能一切正常，少部分功能稍有漏洞，彩蛋一切正常，存档读档功能一切正常，谱面加载一切正常。

1. **收获**
   1. **文件读取操作**

在以上代码的执行过程中，实现存档读档功能必须学会使用文件存取操作，在qt中使用包括但不限于qfile、qtextstream这些方便写入文本文件的存取，也同时了解了fstream、reinterpret\_cast运算符的有关知识。

* 1. **模板类的知识学习**

了解了模板类QVector（Vector）、QString（String）的有关知识，学会了使用QVector。

* 1. **在Qt中进一步深化对继承的理解**
  2. **学会了在connect中使用lambda表达式编写程序内容**
  3. **学会了分文件编写的有关内容**
  4. **了解了Qt中很多类的使用**

1. **反思**

在这一段时间里，我充分体会到了开发程序的艰难。由于对自身的高要求以及眼高手低，本人没能在规定时间内完整地完成大作业的开发。但是我会尽快补全大作业的有关内容。虽然作业提交的截止日期已经到了，但是这个游戏的开发还没有结束。现在虽然只是简单地复刻，以后也可能会开发出创新型的玩法，并且我愿意为了开发出一款属于自己的游戏而努力。

1. **关于谱面编辑器**

为了更完整地体验本游戏，我也将会本游戏所属的谱面编辑器完整地上传到github中，供各位编辑自己的谱子。到时候我会在readme中贴上谱面编辑器的使用方法，敬请期待。