



一、项目分工

学号	名字	角色	班级	职责	贡献
14331401	庄清惠	组长	周四班	游戏登录界面，分数提交，查看排名，界面跳转控制，音符序列	33.3%
14331370	张双涛	组员	周四班	游戏界面物理世界 碰撞 能量槽 额外奖励、计时器及清屏动画	33.3%
14331122	赖加成	组员	周四班	游戏界面音轨及布局 碰撞检测，分数级别判定；游戏帮助界面	33.3%

二、开发环境

Window10 下 VS2015, server.jar

三、项目阐述

名称：畜不及防

简介：

这是一款类似节奏大师的游戏，结合 B 站上火爆且富有带感的鬼畜音乐，让玩家在游戏中挑战手速、准确性和迷之笑点。

功能：

(1) **登录界面**：新玩家输入姓名，点击“Play”、“Help”、“Rank”或“Exit”，依次实现进入开始游戏、新手帮助、玩家排名和退出游戏界面。同时，实现了自动登录的功能，为每个玩家保存数据。

(2) **游戏界面**：(a) 玩家进入后会看到 6 个音轨，而且序列化的音符随机下落，左侧跳动的计时器，游戏分数和积聚的能量槽。还有供玩家直接退出的“exit”按钮。(b) 随着玩家游戏的进行，检测玩家的操作情况，并根据每次操作的准确性（即音符与相应位置的距离）会在屏幕上即时显示“Miss”“Good”“Perfect”三种操作级别。这里“Miss”操作不会增加玩家的成绩和能量槽。否则添加相应的分数和显示玩家操作级别，鼓励玩家更快、更准。当能量槽聚满之后，清屏条立即从上往下快速扫过，将所有屏幕上的音符消灭并有奖励分数，缓解玩家游戏压力，增加用户体验。(c) 在某些时间点会随机有音符飞出，用户必须在该音符穿过的时间内按相应的键来实现切水果的效果，如果成功操作，音符一分为二（分开的两部分也有自己的速度），然后受重力下落。否则该音符与边界碰撞后消失。

(3) **游戏结束**：玩家可以选择重新开始游戏或退出。

服务器：

在这款游戏中，登陆、上传分数和排名等功能是直接使用了 14 周 TA 给的服务器(win10 下)。使用时需在服务器所在目录打开命令提示符 CMD，输入 `java -jar server.jar` 则可运行服务器。

亮点：

鬼畜音乐的结合，玩家数据保存及查看排名情况 网络物理世界与碰撞 游戏动画 汇聚的能量槽 向量队列等数据结构 调度器及事件分发。

四、项目展示

游戏开始界面 右侧的四个按钮点击后跳转相应的界面。



畜不及防

Player Name

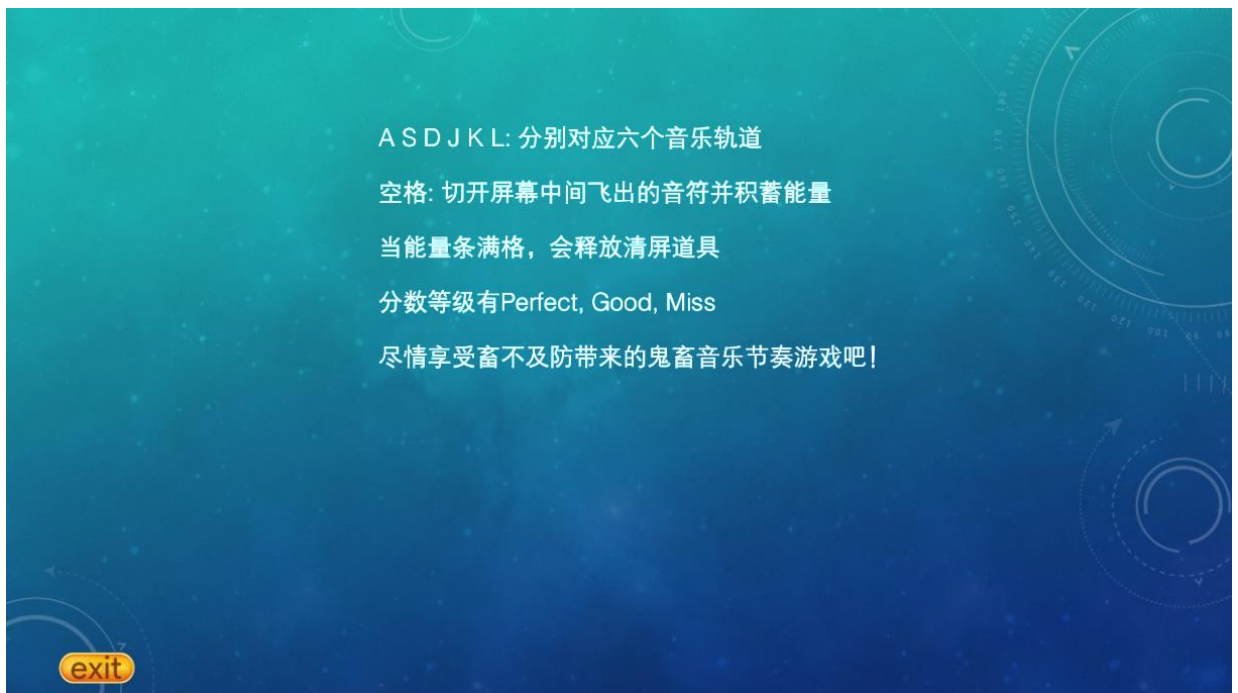
Help

Play

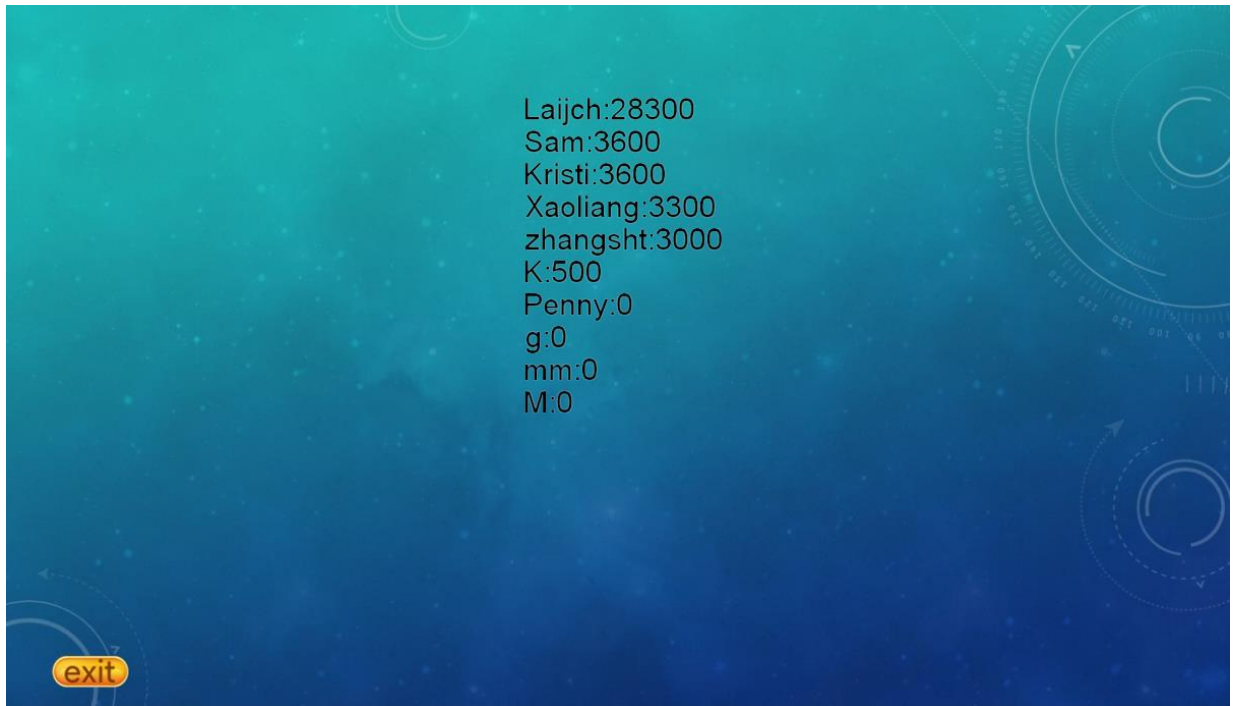
Rank

Exit

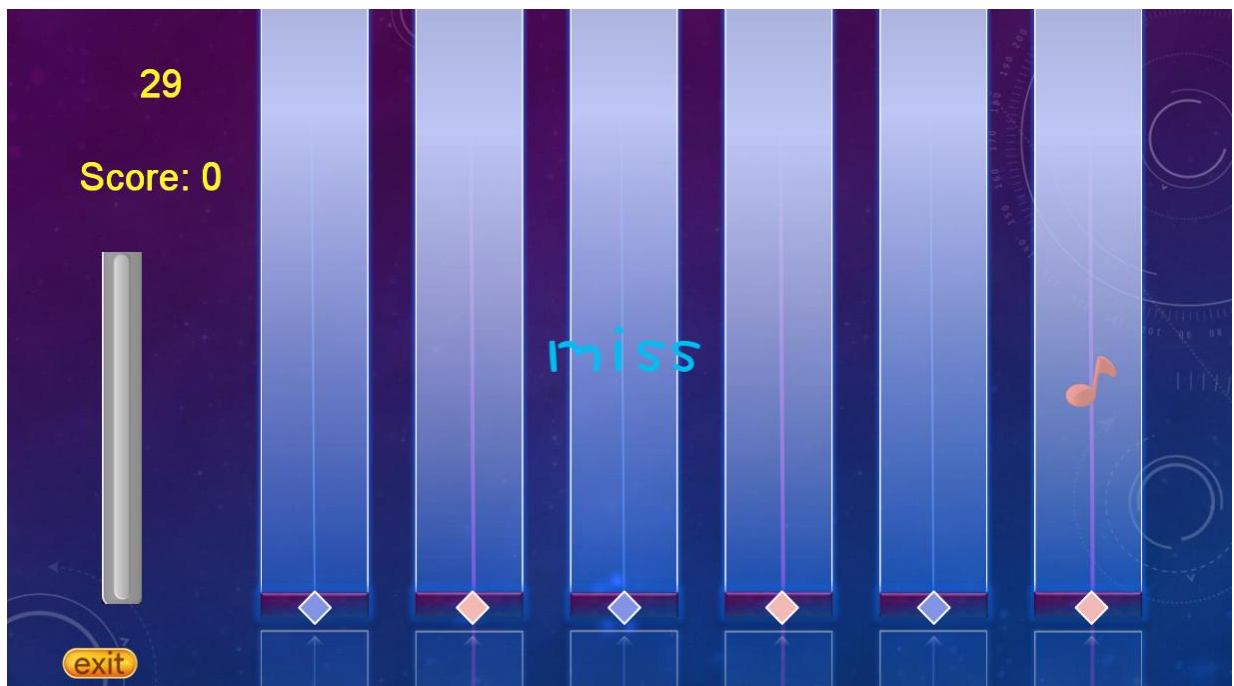
Help 界面教程



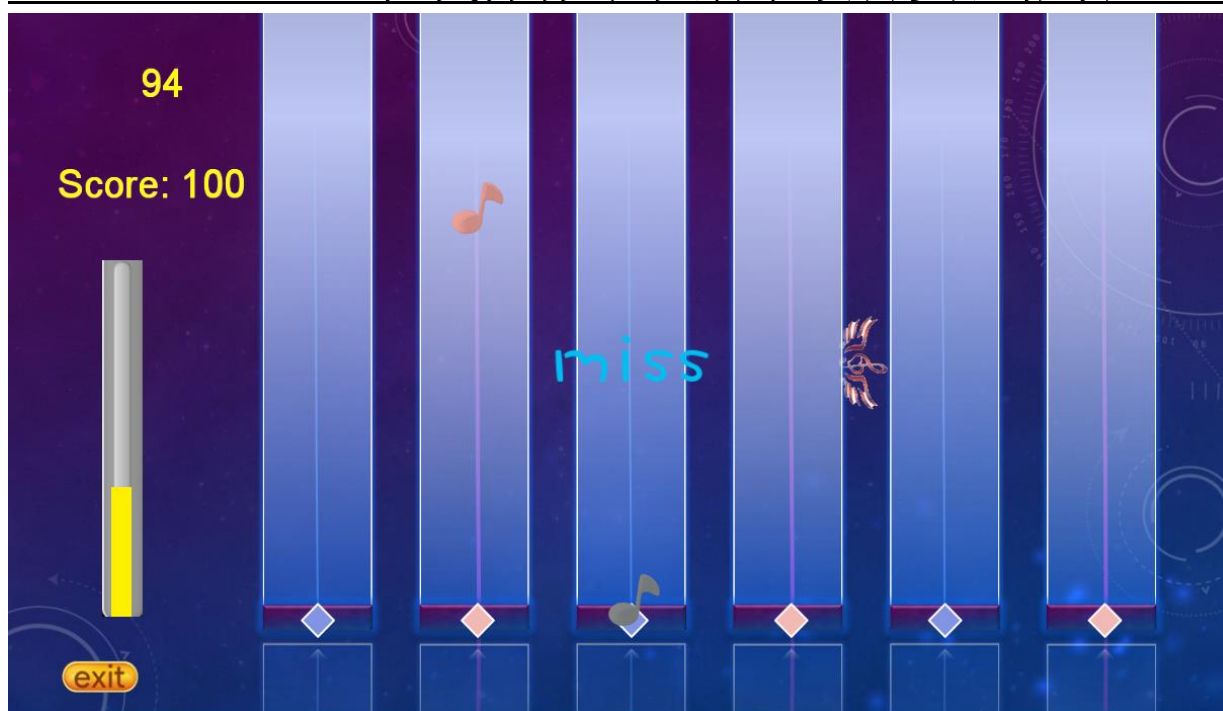
点击查看排名



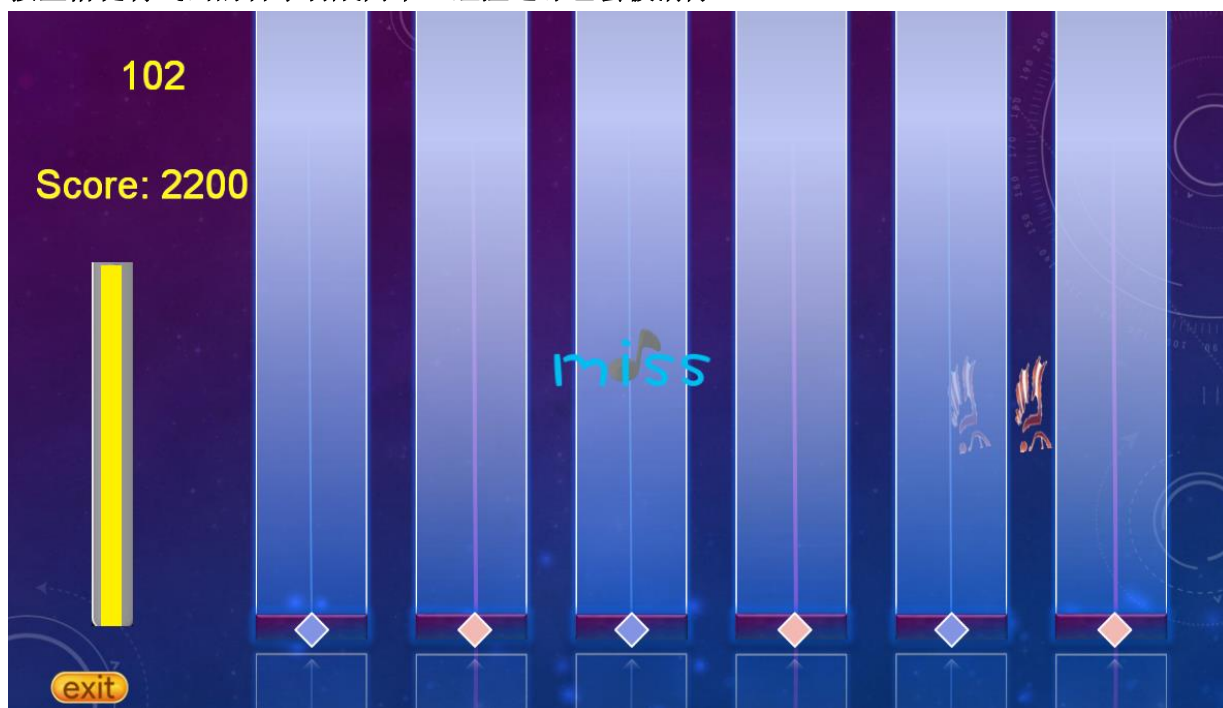
进入游戏界面



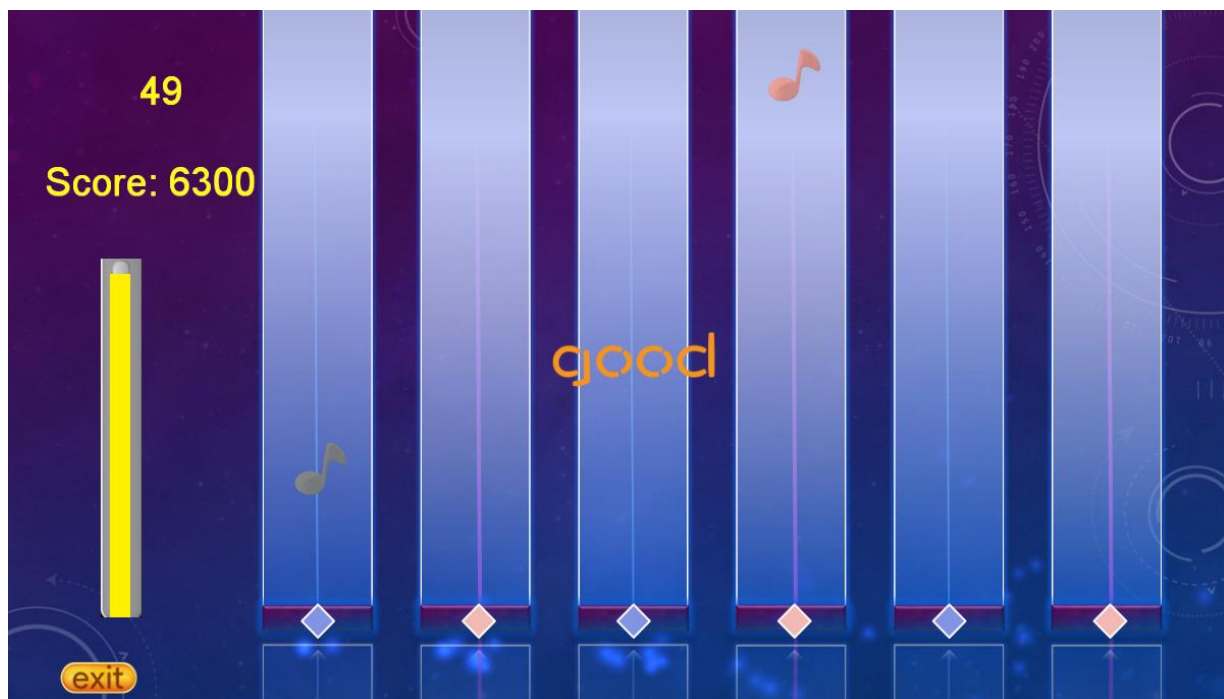
不及时按钮“切掉”音符的话，音符走 与边界碰撞后消失



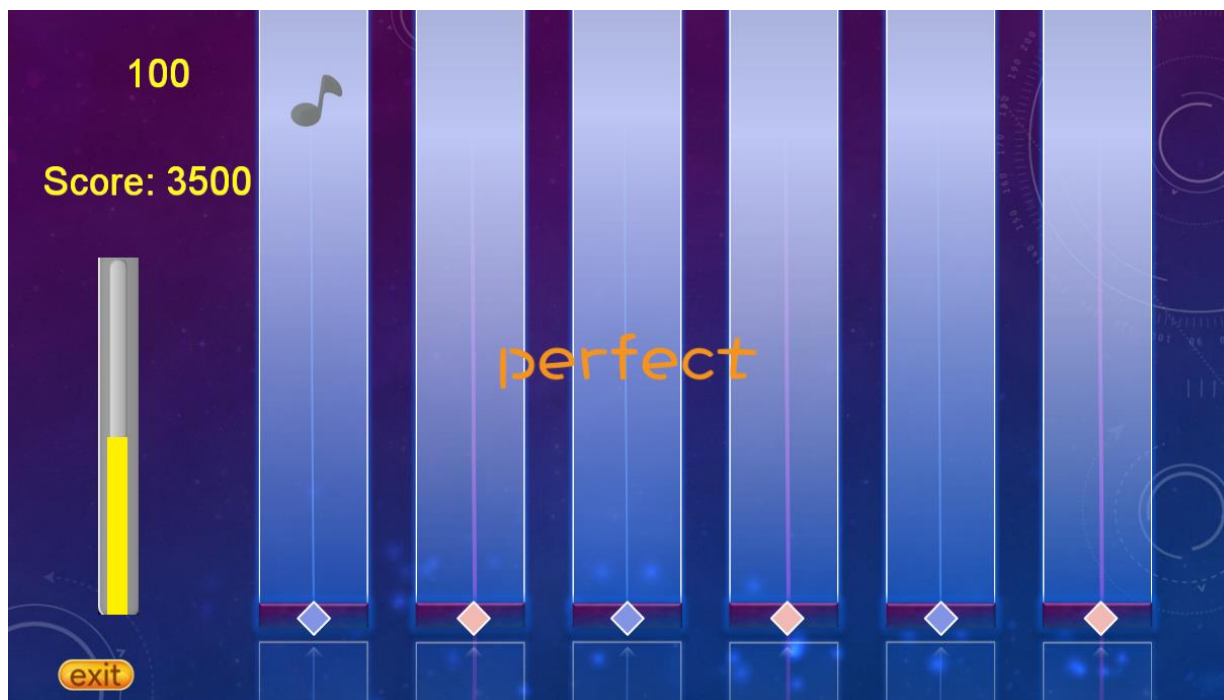
按空格键将飞出的音乐切成两半，碰撞边缘也会被清除



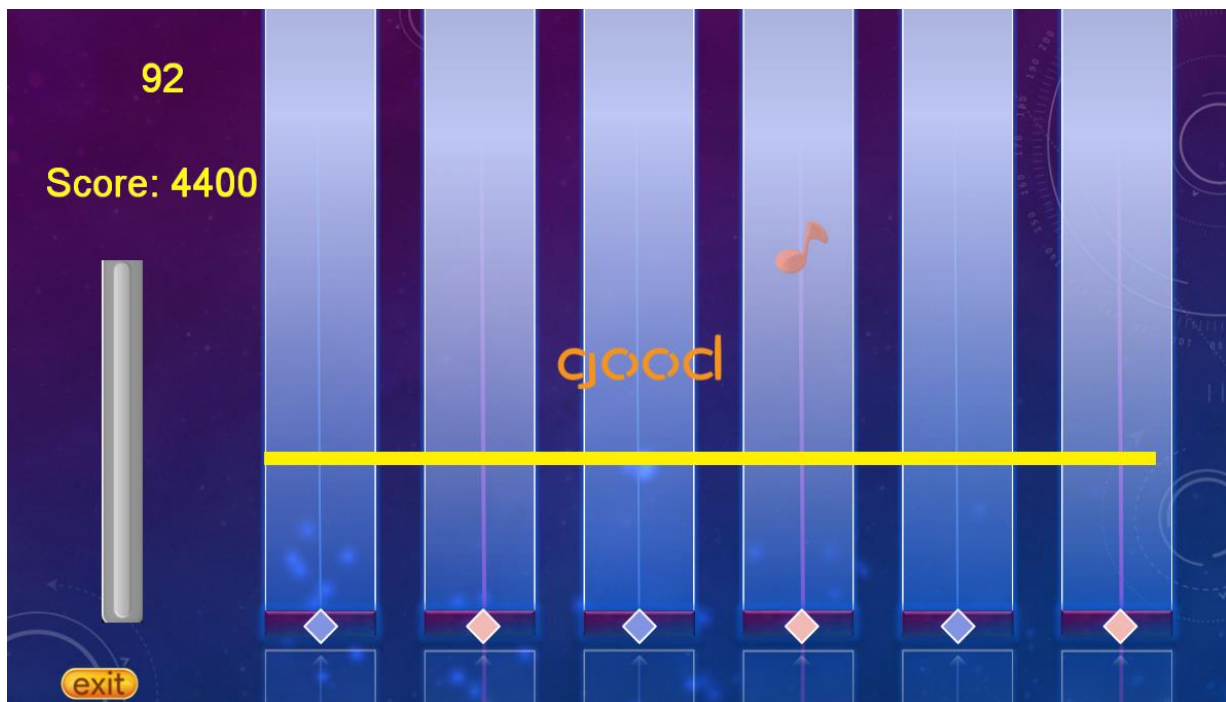
下面的图是 good 等级操作，意味着当玩家按下对应音轨的键时音符和固定的位置还有一定距离



玩家判断精确，操作 perfect!



能量槽聚满，清屏条快速向上消除所有音符，玩家获得相应的 bonus



游戏时间到结束



五、项目难点及解决方案

在代码中碰到两个关于碰撞的问题：（1）碰撞后在`removeFromParentAndCleanup(true)`函数后相应的指针并没有设置成空指针，这样导致运行时中断，通过处理相应位置的指针后问题依然存在，这是因为在游戏过程中不断的创建音符，所以需要用到向量来处理是比较合适的，这里解决方案是设置状态变量来解决问题。（2）对`CategoryBitmask`、`CollisionBitmask`、`ContactTestBitmask`这三个之间的关系的理解，一开始为了省事直接用的原来作业的参数。但是由于本次碰撞的对象较多，所以处理时有些问题，所以可以看出`ContactTestBitmask`是标明对和谁的碰撞检测感兴趣，请注意这不表示物体会发生碰撞，而是会不会传达碰撞。理解这之间的差异，才可以明晰系统会不会调用监听器做出反馈。

另外就是击打音符分数等级判定的问题。分数等级有三种判定：Perfect, Good, Miss。因



为音符是储存在队列中的，因此当音符消失时（击中音符或与清屏道具碰撞等情况）将队头的音符出队，故在队头的音符总是最接近音键的音符（即高度最低的音符）。当没有点击相应轨道的按钮时，音符直接与音键碰撞，则判定为Miss，这是在音符（note）与音键（key）的碰撞相应事件中判定的。而其他两种情况，则是在击打音键时计算音键与最近的音符（即队头的音符）的距离来判断分数等级。点击相应的按钮，在键盘响应事件onKeyPressed中则会调用checkMeet函数，如点击A键，则会向checkMeet函数传一个参数0，表示击打的是第0条轨道，此时checkMeet函数就会计算第0条轨道中音键（key）与最近音符（note）的距离：（注：音键与轨道的锚点位置相同，故计算时可直接用轨道的位置计算）

```
float distance = tracks[trackNum]->getPosition().getDistance(notes.front()->getPosition());
```

当距离在半个音符大小范围内则判定为Perfect，当距离大于该距离且在一个音符大小范围内则判定为Good。

六、项目总结

（1） 经过两次的课程项目，体会最深的是在实践中才能更好地掌握技术，以作业为驱动，在写自己项目的过程中重温和学习以前的知识，尤其是在完成某一功能的过程中，你不得不去面对一大堆不曾想过的问题，面对曾经模糊两可的知识点。这可能会打击一部分同学的积极型和继续向前的勇气，但是只要我们相信自己，耐心去查资料、查文档，我想问题还是可以得以解决的，这之后是收获感、是成就感、是快乐和能量。

（2） 注重团队合作提高开发效率。本次课程的队友都是相当有想法灵感的，在完成作业的过程中也非常认真，一起讨论分工合作，讨论遇到的问题。让我们感觉到团队的强大，互相学习、一起解决问题。