

作业报告

小组：暂复组 小组成员：孙心豪 马唯硕 姚一伦

一. 程序功能介绍

在此次 QT 大作业中，我们组设计了名为“PKU FLYING PACMAN”的小游戏。运行项目可以看到开始界面。点击左上角的设置按钮可以进入设置界面，在那里可以选择游戏背景音乐播放，游戏人物皮肤和游戏难度，点击左下角的帮助界面可以查询游戏规则。点击正中央“开始”按钮即可开始游戏，点击“退出”则退出游戏。在游戏开始后。玩家将操纵 **pacman** 绕过天空和地上的障碍物，获取尽量多的金币。金币分为大金币、小金币与异色金币三种，小金币 5 个相当于大金币 1 个，但获得大金币后一段时间人物体积将会变大，同时免疫障碍物；小金币 2 个相当于异色金币 1 个，但获得异色金币后人物会变色，并且在屏幕的右上方会获得可以获得一次短距离冲刺机会的提示，释放冲刺后会变回原色，冲刺机会无法累加。金币的产生是随机的，其中大金币刷新的概率是 0.05，异色金币的刷新概率是 0.1，在吃豆时，会触发吃豆音效。在我们操纵人物通过一组障碍物后，我们会获得一金币的得分，实时得分会显示在屏幕的左上角。在游戏过程中，如果人物碰到上下方的游戏界面边框或者在非超级模式下触碰到障碍物，游戏就会结束，并显示本次得分以及历史最高得分。此时，我们点击屏幕的任意处，就可以重新开始游戏。玩家在游戏过程中可以随时按“S”键暂停，“C”键恢复。在不清除闯关记录的情况下，系统文件会在退出游戏时保留上次闯关的记录。

二. 项目设计细节

项目共有 8 个自主设计的头文件。以下分别介绍他们的功能。

game.h 文件是显示游戏开始是图片的文件，其中有设置和帮助控件，点击分别跳转到“**changewindow**”和“**helpwindow**”界面。点击“开始”控件将跳转到“**mainwindow**”界面。“**mainwindow**”界面是游戏的主界面。

Changewindow.h 文件控制了设置界面，可以调控三种功能。第一，操纵其中的按钮将直接改变背景音乐的播放情况；第二，可以操纵按钮选择不同的人物形象；

第三，可以操纵按钮选择不同的关卡难度。

Helpwindow.h 是帮助界面，其上有简单的文字游戏简介与游戏规则介绍，设计了退出按钮。

Bird.h 定义了玩家可以控制的人物类。所包含的函数和功能如下：

Updateposition()	实时更新人物位置
flap()	识别玩家按下空格键的操作并释放信号使得人物向上跳跃
Onsuper()/offsuper()	使得人物进入和离开超级状态
Issuper()	判断当前人物是否处于超级状态（即变大状态）
getSprint()	人物获得一次短距离冲刺机会
resetSprint()	人物冲刺后重置冲刺机会

Dot.h 是金币类，他是游戏中的金币。所包含的函数和功能如下：

Updateposition()	更新金币位置
Reset()	刷新产生新金币
Iscapsule()	判断金币类型,其中返回值 1 是大金币,返回值 2 是异色金币,返回值 0 是普通金币
rush	金币加速运动使人物产生冲刺的效果
resetSpeed	重置速度以终止人物的冲刺效果

Ghost.h 是障碍物类，定义了障碍物的运动和产生。其中函数和功能如下：

Updateposition()	随机产生障碍
Checkpos()	判断障碍位置是否合理
rush	障碍物加速运动使人物产生冲刺效果
resetSpeed	重置速度以终止障碍物的冲刺效果

Stoppage.h 定义了游戏过程中的暂停界面，游戏中暂停即可显示。

Mainwindow.h 是主游戏界面，包含检测是否通关，计算分数和道具数量等功能。其中函数和功能如下。

Init()	根据关卡数初始化各变量
Detectcollide()	判断是否撞到障碍
Updateposition()	更新个物体位置
Updatescore()	更新得分
Lose()	接受失败信号，展示失败界面
process	处理吃到金币后的得分、技能触发、技能获得情况
Read_maxscore()/write_maxscore()	读/写入最高分，用于记录

三. 小组成员分工情况

本次项目的分工情况大致如下：

孙心豪同学设计开始界面，设置界面，帮助界面，暂停界面，并负责各个界面之间的信号转接。项目中类 `helpwindow`、`changewindow`、`game`、`stoppage` 主要由孙同学完成。此外，孙同学同时负责完成功能设计文档；

马唯硕同学负责设计游戏中各种事物的运动，主要包括人物下落，人物跳起，人物获得奖励，人物技能的设计，人物放大，鬼平移，金币生成的效果展现等。项目中类 `bird`，`ghost`，`dot` 主要由马同学完成。同时，马同学还进行了关卡难易度的设计。

姚一伦同学负责设计游戏主界面，结束界面、部分人物技能的设计以及分数的具体规则。项目中类 `mainwindow` 主要由姚同学完成。此外，姚同学也参与了 `game` 界面与 `mainwindow` 界面的信号对接。

四. 项目总结与反思

总体来说，本项目利用 QT 并结合《pacman》的美术概念，完成了对经典游戏《Flappy Bird》的重新实现以及适当扩展。

项目的开发过程总体来说，经历了三个阶段。

第一阶段中，小组的各个成员分别构建自己的代码，最终合并在一起，产生初步可运行的程序。这一阶段的程序已完成应用主体玩法的构建，但在细节如音乐、得分机制、图片选取等方面并未完善。

第二阶段我们进行了细节的补充，设计了较为合理的得分机制，添加了背景音乐并添加了音乐的开关功能，并且对于开始界面与结束界面进行了美化。此版本投稿了路演视频。

第三阶段我们添加了更多的功能。在 changewindow 界面添加了更改人物形象、切换关卡难易程度的功能。同时我们在游戏界面中添加了异色金币，人物拾取异色金币后获得一次短暂冲刺的机会。并且，我们添加了 exe 文件的图标。

经历了这三个阶段，最终确定了项目程序的最终版本。

通过这一项目，首先我们进一步地掌握并理解了面向对象的编程理念，领会到如何通过设计类的方法抽象游戏中涉及到的对象，和封装、继承、多态的特征是如何清晰地处理对象间的复杂关系的，进一步加深了对本学期程序设计实习课程前半学期内容的理解，同时，我们了解了在程序中前端和后端是如何配合，形成一个完整的程序的。在本项目中，我们自定义了 8 个类，通过这些类的相互作用完成了游戏的主体。

其次，我们初步掌握了利用 QT 进行 GUI 编程的方法，了解到了 QT 有别于传统 C++ 的一些重要特征，包括 QObject 采用的对象树方法，以及槽与信号机制等等。对象树提供了一种结构性的方法组织并管理程序中的对象；当父对象被销毁时，子对象也会同时被销毁。这一特性简化了 C++ 中的内存管理，有效避免内存泄漏的可能。本项目中的大部分对象即通过对象树组织。槽与信号机制提供了一种处理事件传递与响应的快捷方式，也是 QT 的核心特性。在本项目中，我们充分利用了这一机制，如画面的更新、超级模式的开启与关闭、冲刺过程中背景的加速运动、难度的切换等功能的实现等等。

另外，我们藉此了解了 QT 的一些基本库。如<QTimer>、<QPixmap>、<QGraphicsPixmapItem>等等，熟悉这些库所提供的函数，有助于我们在日后的

QT 编程实践中，快速调用适当的方法来完成一些功能，或对这些功能给出更加优雅的实现。运用这些功能，我们学会了用 Qt 进行一些简单的动画效果处理和 ui 界面的设计。

最后，我们也积累了一些多人合作编程的经验，对 GitHub 的使用有了初步的了解，初次进入了软件开发的项目中。想要发挥一加一大于二的效果，组员间必须有清晰的沟通协作，明确的分工安排。此外，合理的版本控制也很重要。本项目全程使用 Github 来进行协作开发，每个人独立工作，并将各自的更改有序合并至主分支，这有效避免了可能的冲突。

纵观整个项目，还有一些方面可以进一步提升。

功能上可以进一步完善并丰富。本项目成功再现了《Flappy Bird》的核心玩法，但在主线之外，很多细节可以进一步打磨。比如，碰撞效果的展示、界面的进一步美化等等。并且，也可以考虑加入更多内容，比如各种技能，多样化的 dot，不同造型的 ghost，场景选择等，这样可以使游戏体验更充实。

游戏平衡性可以进一步提高。例如在跳跃过程中的速度变化可以采用与速度相关的函数，这样可以优化游戏体验，避免游戏中连续跳跃时人物跳得太高；同理，也可以将人物自由落体的运动改成具有风阻的运动，避免下落过快时难以通过操作挽回局面的情况发生。

玩法上可以进行扩充。本项目主要基于 flappy bird 的规则，玩法上添加了吃到特殊金币获得技能的机制，事实上，我们可以引入更多影响游戏性质的技能乃至人物自带的主动技能，来超出 flappy bird 的概念。另外，我们可以设计关卡制度，通过获得一定的分数来通关，序号大的关卡金币更少而障碍物跨越难度更大，背景运动速度更快，随着通关增加更多的技能。我们也可以通过不同技能的衔接打出得分更快的操作。

代码风格可以进一步统一并规范。基于可读性和效率等问题，也为了规避常见的漏洞，代码风格应当保持一致并规范。本项目中，不同人完成的部分存在较大风格差异，并且对于某些相同功能的实现不同两人实现的方法不同，导致代码合并时需要进一步修改，部分代码不够规范简洁，质量有待继续提高。

这次 Qt 大作业不仅是我们初步了解了 Qt 的使用方法,对计算机前端编程有了了解,更培养了我们自主学习、小组合作的能力,对我们未来我的学习生活中一定会产生深远的影响。