|  |
| --- |
| [键入公司名称] |
| 飞机大战 |
| [键入文档副标题] |
|  |
| **anwu** |
| **[选取日期]** |

|  |
| --- |
| [在此处键入文档的摘要。摘要通常是对文档内容的简短总结。在此处键入文档的摘要。摘要通常是对文档内容的简短总结。] |

目录

[一、功能介绍 2](#_Toc464555094)

[1. 增加了buff种类 2](#_Toc464555095)

[2. 增加了敌机类型 2](#_Toc464555096)

[3. 增加了两个关卡 2](#_Toc464555097)

[4. 增加了背景音效 2](#_Toc464555098)

[5. 增加了背景移动效果 2](#_Toc464555099)

[6. 增加了闪烁和爆炸效果 2](#_Toc464555100)

[7. 优化了飞行物模型 3](#_Toc464555101)

[8. 增加了能量槽积累和生命值的动态显示功能 3](#_Toc464555102)

[9. 增加了键盘控制 3](#_Toc464555103)

[二、类结构： 3](#_Toc464555104)

[1. 基础类： 3](#_Toc464555105)

[2. 对象类： 4](#_Toc464555106)

[3. 行为类： 5](#_Toc464555107)

[4. main类： 6](#_Toc464555108)

[三、实现思路 6](#_Toc464555109)

[1. 飞机模型 6](#_Toc464555110)

[2. 路径控制 6](#_Toc464555111)

[3. 路径控制 6](#_Toc464555112)

[4. 爆炸效果 7](#_Toc464555113)

[5. 背景移动 7](#_Toc464555114)

[6. 能量槽和血条 7](#_Toc464555115)

[7. 追踪飞行物 7](#_Toc464555116)

[四、总结 7](#_Toc464555117)

# 一、功能介绍

本游戏在实现了王克晶老师飞机大战原有功能的基础上，新增了多种敌机、Boss机及相应的出场和飞行方式，并使敌机也具有了发射子弹的能力；增加了英雄机子弹的类型、防护罩等特效；增加了背景音效，并设置了三个关卡等。

## 增加了buff种类

游戏随机生成多种buff，吃到相应的buff，子弹会更换成相应的效果，并拥有自己的飞行轨迹和相应的威力值。双倍火力：一次发射两颗子弹；平行飞行；散弹：一次发射三颗子弹，发散状飞行；摇摆弹：一次发射两颗子弹，呈交叉装飞行；追踪弹：自动追踪离敌机最近的敌方飞机；防护罩：获得该buff后，英雄机会获得5s的无敌效果。

## 增加了敌机类型

敌机和boss机拥有各自的飞行轨迹、不同的生命值以及不同的子弹发射方式。深红色敌机A从窗体顶端生成，向下飞行直至飞出界面，期间定时发射直线飞行的子弹；蓝色敌机B从窗体两侧生成，横向往复移动，期间定时发射三颗散弹；军绿色敌机C从窗体顶端两侧生成，不具备发射子弹的功能，但是会时刻追踪英雄机的位置，并向其靠近直至撞毁或者被躲开；粉红色敌机D从窗体两侧生成，一次生成四架，呈队列状飞行；boss机从窗体上端生成，定期进行横向移动躲避英雄机子弹，并具有三种子弹类型。

## 增加了两个关卡

当关boss机被击毁后即代表通关，进入下一关卡，不同关卡通过设置敌机的产出频率和boss机的生命值来设置不同的难度。

## 增加了背景音效

游戏运行过程中会伴有背景音效，当鼠标移出窗体，游戏暂停时，背景音效随之暂停；当鼠标再次回到窗体，背景音效继续播放。

## 增加了背景移动效果

游戏运行期间，背景图会不断下移，模拟飞机飞行时场景不断变换的感觉。

## 增加了闪烁和爆炸效果

游戏中的各个飞行物的显示都是动态的，通过不断的突变切换来模拟各飞行物的动态飞行。在此之外，还增加了飞行物被击毁后的爆炸效果。

## 优化了飞行物模型

各飞行物根据其形状设置了相应的模型和碰撞方式，保证了游戏中各种物体尽可能真实的碰效果。

## 增加了能量槽积累和生命值的动态显示功能

英雄机子弹在击打到敌机后，会获得相应的能量值续集，当能量值续集到一定程度，可以通过鼠标左键出发相应的必杀技能。此外Boss机的生命值可以通过血条的形式动态的显示给游戏者。

## 增加了键盘控制

添加了键盘事件监听，可以使用键盘的方向键控制英雄机的移动。

# 二、类结构：

## 基础类：

AwardType枚举类：定义英雄机获得buff前后的状态变化，ADD\_LIFE代表加生命,DOUBLE\_FIRE代表双倍火力,DEFENDER代表添加防护罩,ROCK代表摇摆子弹,FRIENDPLANE代表僚机,SHOT\_GUN代表散弹,ROCKET代表跟踪弹,NONE代表什么都没有。

GameState枚举类：定义游戏状态，START代表开始,PAUSE代表暂停,RUNNING代表运行,GAME\_OVER代表游戏结束,LEVEL\_UP代表通关。

Model类：用于模拟飞行物碰撞体积，提供两个静态方法，getMod（）用来获取不同飞行物的模型，crash（用来判断飞行物是否发生碰撞）。

ObjectState枚举类：定义各种飞行物状态，alive代表存活、booming代表正在爆炸、destroyed代表已经爆炸完，可以销毁对象。

Path类：定义飞行物移动路径，提供飞行物的横向、纵向、斜向和追踪移动方法。

TimerAct计时器类：用来精准控制游戏运行过程中的各种行为触发时间，提供通用计时方法timerAction（）、防护罩时间计时timer（）、和boss机移动计时bossMoveTime（）。

实现FlyingObject对象比较方法的FlyingObjectComparator类，为FlyingObject集合的排序提供方法。

## 对象类：

FlyingObject飞行物父类：定义所有子类飞行物的公共属性和方法。属性包括飞行物坐标位置point、用来闪烁的图片集合images、当前显示图片image、飞行物当前状态state、飞行物横向移动速度speedX、飞行物纵向移动速度speedY、计时器timerAct、生命值life、爆炸时间boomTime、闪烁时间glintTime1、当前图片编号imageNum。方法包括碰撞方法、移动方法、爆炸方法、毁灭方法、越界判断方法、飞行物闪烁方法。

敌机A、敌机B、敌机C、敌机D、boss机类，不同的是移动方法move（）和加分方法。

敌方子弹A、敌方子弹B、敌方子弹C类，不同的是移动方法move（）。

英雄机子弹A、英雄机子弹B、英雄机子弹C类，不同的是飞行方法move（）。

英雄机

僚机A、僚机B（目前没有实现）。

Buff类额外定义了奖励类型获取方法generateAward（）和生成相应图标集合的方法getImages（）。

奖励接口定义了返回奖励类型的方法。

敌机接口定义了返回积分方法。

## 行为类：

背景移动类：定义了背景移动的静态方法。

飞行物碰撞类：定义了各种飞行物之间碰撞的方法。

飞行物销毁类：定义了飞行物判断为可销毁状态后的销毁方法。

飞行物销毁检查类：遍历各飞行物集合，判断是否已经结束爆炸时间，并标志为可销毁的。

游戏状态检查类：检查游戏当前处于那种状态，提供游戏结束和通关检查方法。

飞行物生成类：定义了各种飞行物的生成方法。

监听事件类：定义了主窗体中要注册的各种键盘和鼠标事件。

飞行物移动类：遍历各飞行物集合，调用各自的移动方法。

音效类：定义了背景音乐的播放方法。

飞行物越界类：定义了飞行物的越界检查和销毁方法。

绘图类：定义了主窗体中各种飞行物和物体的绘制。

## main类：

是整个程序的控制中心，负责各个类之间的信息交互。

# 三、实现思路

## 飞机模型

由于各种飞行物的形态不同，为了保证各飞行物间的碰撞切实是飞行物与飞行物的碰撞，而不是图片边框与图片边框的碰撞，需要单独创建一个Model类来模拟各飞行物的形态，并提供不同的碰撞判断方法。将Model类的对象作为飞行物主类的属性，在飞行物碰撞判断时直接调用该Model对象的碰撞方法。本游戏中具体的做法是将boss机模拟成长轴水平的椭圆，将英雄机和敌方飞机模拟成圆，判断时计算两飞行物坐标点的距离，如果该距离大于或等于当前方位各自飞行物半径的和，则视为未碰撞，反之则视为发生碰撞。

## 路径控制

飞行物的移动路径需要计时和相应的界限判断，需要解决一系列类似于“什么位置移动、移动多久或者多远、到什么位置停止、何时进行后续移动….”的问题。而整个程序相关操作的运行又都是在计时间隔为10ms的timer中进行的，由于关系到其他操作，该计时器不能随意调整，所以选择单独创建TimerAct类构建TimerAction方法来实现精准计时操作，程序每10ms访问一次该方法，而我们可以通过输入相应的参数来控制其在指定秒数之后以指定的时间间隔连续返回指定次数的true。

## 路径控制

有了时间的精准控制方法，还需要各种方式的移动方法，二者配合就可以灵活的完成飞行物的各种移动方式。这里选择单独创建一个Path类来控制路径，该类中创建了飞行物横向、纵向、斜向以及追踪的静态方法，每次访问相应移动方法时，只需提供给飞行物当前坐标和横、纵向移动速度，就能得到飞行物下一个方位的坐标位置。在此基础至上，今后可以做更多复杂的移动方式，像正弦函数的波浪状轨迹等等，由于时间有限，暂时还没有实现。

## 爆炸效果

在王克晶老师原先的飞机大战设计中，敌机生命值一旦小于等于0即代表可以销毁了，若此时进行飞行物对象的销毁，则飞行物图片会很突兀的从画面中消失。为了解决这一问题，给飞行物增加了几种状态，分别是：ALIVE代表存活，BOOMING代表正在爆炸，DESTROYED代表可以销毁。只有当飞行物的状态为DESTROYED时才会进行销毁，而在BOOMING期间则可以进行爆炸效果的显示（切换爆炸图片组合）。

## 背景移动

游戏背景移动会随着游戏的运行不断进行，要使用尽可能少的图片完成这一过程，需要将图片反复的进行切换。使用两张图片，一张显示在窗体中，另一张用来替换前者下移时露出的空间，一旦发现第一张图片全部移动出窗体，则将两张图片的绘制坐标重置。如此反复达到背景持续移动的效果。

## 能量槽和血条

本游戏中使用两张图片完成能量槽或者血条的显示，一张边框图和一张条形图，反复获取游戏中的能量值或者boss生命值，使用drwaImage的缩放功能，按比例缩放条形图的宽度，并显示出来，从而达到动态显示的效果。

## 追踪飞行物

反复获取被追踪物的坐标位置，计算追踪物与被追踪物之间最短路径的斜率，根据追踪物的纵向移动速度，计算当前按照该斜率移动的横向速度，从而得出下一个坐标点的位置。英雄机子弹追踪敌机需要控制追踪的优先级，需要使用集合的排序，排序顺序为：离追踪子弹最近且纵坐标在子弹之上的敌机、在子弹之上离子弹稍远的敌机、在子弹之下离子弹较近的敌机、在子弹之下离子弹较远的敌机，经过此排序后，追踪子弹在依据先前的追踪方法对敌机进行打击。

# 四、总结

由于时间有限，本游戏成果与最初的预期有些诧异，预期的功能还应该有：