《程序综合设计》课程过程文档-设计文档

1. 基本信息

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 项目名称 | | 全民飞机大战 | | | | 版本 | 1.00 |
| 学号 | 5120152534 | | | 专业 | 软件工程 | 姓名 | 李国印 |
| 邮件 | 1203251505@qq.com | | | | 电话 | 18681660719 | |
| Google code 地址 | | |  | | | | |

1. 类的设计

坦克大战，根据需求分析共设计15个类，分别如下：

① 资源 ② 主控制 ③ 飞机基类 ④ 玩家飞机  
 ⑤ 敌方飞机{有攻击型，无攻击型，boss} ⑥ 特效  
 ⑦ 道具 ⑧ 子弹 ⑨ 星星奖励 ⑩ 登录 注册

11 开始界面 12 游戏界面 13 结束界面

其中主要的类有 ①②③④⑤ 具体类图如下：

CPlane//飞机基类

int m\_posx; //坐标X int m\_posy; //坐标Y

int m\_blood; //血量 int m\_value; //价值·可得分数

int m\_attack; //攻击强度 int m\_pan; //判断是那种飞机

bool m\_Isclear; //是否被清除 int m\_dix;//X轴偏移量

int m\_diy; //Y轴偏移量 int m\_timer; //时间间隔

int m\_choose; //判断一类飞机下的那一辆飞机

int m\_pretimer;//存储当前时间int m\_timerCD; //技能冷却时间 IplImage \* m\_nowplane;

virtual void Draw(IplImage \*bg) = 0; //飞机绘制

virtual void Move(int height,int width) = 0; //飞机移动

virtual void Attack(CManager \*manger) = 0; //飞机的攻击

Void Destory(int atk){m\_blood -= atk;if (m\_blood <= 0)m\_Isclear = true;}//被攻击失血

IplImage\* GetNowImg() { return m\_nowplane; }

bool IsClear(){ return m\_Isclear; }//判断飞机是否死亡

int GetNowX(){ return m\_posx; }//得到飞机的坐标

int GetNowY() { return m\_posy; }

int GetValue() { return m\_value; }//返回所得分数

int GetJudge() { return m\_pan; } //返回值判断是那种飞机

CResource //资源类

IplImage \*myPuplane[9];//紫色飞机

IplImage \*myGlplane[9];//金色飞机

IplImage \*myPlplane[9];//土色飞机

IplImage \*enemyplane[2];//无攻击型敌机

IplImage \*enemyfightplane[4];//攻击型敌机

IplImage \*enemybossplane[3];//boss飞机

IplImage \*bultte[7];//子弹

IplImage \*number[10];//数字

IplImage \*score[7];//等级

IplImage \*texiao[4];//特效

IplImage \*map[4];//地图

IplImage \*buff[2];//道具

IplImage \*blood;//血条

IplImage \*startUI;//开始背景

IplImage \*endUI;//结束背景

IplImage \*ui[8];//图标

IplImage \*m\_award[7];//星星奖励

IplImage \*bomb[4];//炸弹

IplImage \*star;//一颗星星

void InitResource();//初始化

void ReleaseIplimage();//释放图片指针

CEnemyBossplane//敌方boss飞机类

(继承于plane基类)

IplImage \*m\_plane[3]; IplImage \*m\_bullet[7];

int m\_contral;

CEnemyBossplane(IplImage \*plane[], IplImage \*bullet[]); //重载函数

void InitData();//初始化

void Draw(IplImage \*bg);// 多态 绘制

void Move(int height, int width);//多态 移动

void Attack(CManager \*manger);//攻击

CMyplane //玩家飞机类(继承于plane基类)

IplImage \*myPuplane[9]; //紫色飞机

IplImage \*myGlplane[9]; //金色飞机

IplImage \*myPlplane[9]; //土色飞机

IplImage \*myplane[9]; //存放正在循环的飞机

IplImage \*m\_bullte[5]; //存放子弹

int Speed; //飞机移动速度

int m\_index,m\_speed; //切换照片

bool panduan; //来确定什么时候照片该反着切换

void SetNowImg(IplImage \*planepu[], IplImage \*planegl[], IplImage \*planepl[], IplImage \*bullet[]);

//初始化所需图片

void InitData();//初始化变量

void PlaneChoose(int buffnumber);//调整飞机类型

void Draw(IplImage \*bg); //多态 绘制

void Move(int height,int width); //多态 移动

void Attack(CManager \* manger);//攻击

int GetBlood(); //返回血量

void AddBlood();//加血

CEnemyplane//敌方无攻击飞机类

(继承于plane基类)

IplImage\*enemyplane[2];

CEnemyPlane::CEnemyPlane(IplImage \*plane[]);

//重载函数

void InitData();//初始化

void Draw(IplImage \*bg);// 多态 绘制

void Move(int height, int width);//多态 移动

void Attack(CManager \*manger);//攻击

CEnemyFightplane//敌方攻击型飞机类

(继承于plane基类)

IplImage\*m\_plane[4]; IplImage\*m\_bullet[6];

CEnemyFightplane(IplImage \*plane[], IplImage \*bullet[]); //重载函数

void InitData();//初始化

void Draw(IplImage \*bg);// 多态 绘制

void Move(int height, int width);//多态 移动

void Attack(CManager \*manger);//攻击

CManager//主控制类

string m\_name; int n; int m\_timer, m\_pretimer, m\_timerCD; //用于时间更新

int m\_timer1, m\_timerCD1; int m\_timer2, m\_timerCD2; int m\_timer3, m\_timerCD3;

int m\_timer4, m\_timerCD4; int m\_GradeNumber; //分数int m\_Starnumber;//星星数量

int m\_buffnumber;//吃到buff的数量 int m\_bgChoose;//更换背景

int m\_beclearednum; //被消灭的敌机数量 bool m\_bossAlive; //判断boss是否被消灭

bool m\_buffAlive; //判断buff是否被吃掉 bool m\_buffaccpet; //是否吃到buff

bool m\_IsSaveTxt; //缓存数据判断，int m\_starspeed;CStartUI m\_startui;

CEndUI m\_endui; GameState m\_state; CResource m\_resource;

CGameUI m\_gameUI; CMyplane m\_myplane; CEnemyPlane\* m\_enemyplane;

CEnemyFightplane\* m\_fightplane; CEnemyBossplane \*m\_bossplane;

vector <CPlane \*> m\_plane; vector <CAward \*> m\_award;

vector <CAward \*> m\_bomb; vector <CBullet\*> m\_Bullet;

vector <CBullet\*> m\_myplaneBullet; CBuffTool \*m\_buff;vector <CEffect\*> m\_effect;

CManager(string user);

void InitGameResource();//初始化资源

void ReleaseGameRes();//释放资源

void Run();//运行函数

void MyplaneUpdata();//我方飞机更新

void EnemyPlaneUpdata();//敌方飞机更新

void AwardUpdata();//星星更新

void BuffUpdata();//道具更新

void BullteCreat(IplImage \*bulltImg, int posx, int posy, int dix, int diy, int attack,int type);//创建子弹

void BullteDraw();//子弹绘制

void BullteMove();//子弹移动

void BullteClear();//子弹清除

void BullteUpdata();//子弹更新

void AddEnemyPlane();//添加敌方飞机

void TimeUpdata();//时间更新

void Timeupdata\_1();//无尽模式时间更新

void Crash();//碰撞函数

int GetGrade();//

int GetWJGrade();//

int GetStarnum();//

string GetName();//

void ResterGameInit();//

enum GameState { GameStart, GameRunning, GameEnd };//

图1 各类图

1. 重要函数流程图

1. 该流程图为总共过程，（可写出详细功能描述）。流程图如下：

开始

注册

登录注册

注册成功并登陆

登录成功

模式选择

无尽模式 一般模式 enter键

Enter键双击进入

m\_state

m\_state具有初始值0,1,2

m\_state=1

m\_state=2

m\_state=0

运行界面

结束界面

开始界面

AllUpdata

按键

按Enter键

按键

Enter键开始游戏

gameover| |Isalive

gameover=false || Isalive = false

继续运行

jixuyunxing

结束

按ESC键 按ESC键

图2 **Run()** 函数流程图

2. 该函数Move() 完成玩家坦克的移动功能，流程图如下：

开始

输入按键

判断按键

再次输入按键判定

按W键 按S键 按A键 按D键

飞机下移

飞机右移

飞机左移

飞机上移

结束

图3 玩家**Move()** 函数流程图

1. 该函数DrawAll() 完成菜单栏选项功能，流程图如下：

开始

↑ or ↓

↓ ↑

m\_choose初始化值为-1

m\_choose-2

m\_choose+2

i初始值为0

结束

变换图片

i+2

m\_choose == i+1？

m\_choose！=i+1

m\_choose==i+1