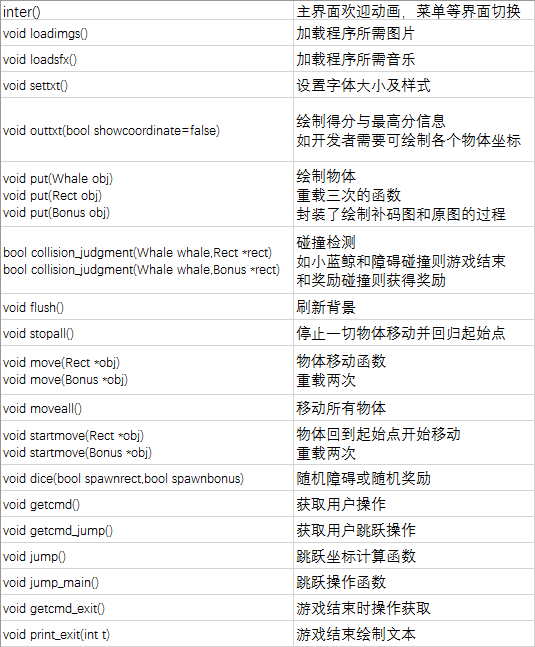
Crazy Whale——疯狂的小蓝鲸

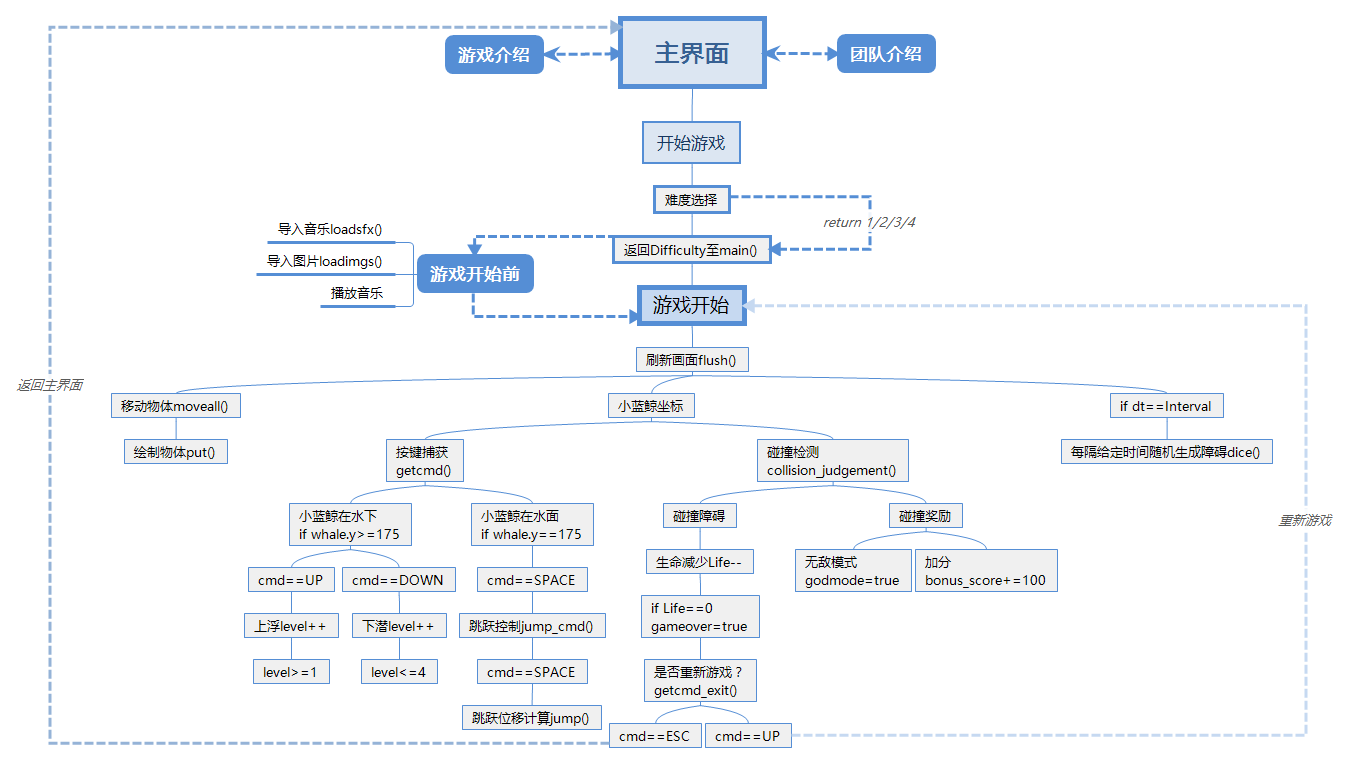
——C语言程序设计报告

工程管理学院 岳翔 芮雨琛 吴昊

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 组员信息 | 学号 | 姓名 | | | 手机号 | 邮箱 |
| 161271029 | 岳翔 | | | 15720618301 | 1196061706@qq.com |
| 161278025 | 芮雨琛 | | | 13952070991 | 161278025@smail.nju.edu.cn |
| 161278035 | 吴昊 | | | 13236551308 | 161278035@smail.nju.edu.cn |
| 程序功能 | 《Crazy Whale》是一款以南大吉祥物“小蓝鲸”为主角的跑酷游戏。本游戏通过操作“w”“s”或“↑”“↓”控制小蓝鲸上浮下潜以躲避海底障碍物，在水面上时按“空格键”可以让小蓝鲸跳跃以越过岛屿，游戏过程中小蓝鲸存活时间越久得分越高。与此同时，小蓝鲸吃到鱼可以获得加分或者无敌的奖励。难度方面共设计了四个难度，难度乘以时间决定最终的总分。除了游戏部分之外，玩家还可以查看规则及团队介绍。  本程序设计灵感来源于《flappy bird》，但实际上程序设计思路完全是团队原创。玩家在游戏中还能发现许多亮点，如千奇百怪的障碍物、奖励设置、跳跃操作、小蓝鲸吐槽等，界面精美，可玩性高，是一款操作简易娱乐性强的游戏。 | | | | | |
| 组员分工 | 岳翔 | | | Crazy Whale团队负责人，游戏总策划，主框架设计者。主要完成游戏部分代码实现。程序设计中解决的问题有：透明贴图绘制、结构体封装、坐标测量、文本输出、物体移动、随机障碍、奖励设置、按键捕捉、音乐导入。美工方面，解决的问题有：补码图绘制、图片信息记录。 | | |
| 芮雨琛 | | | Crazy Whale美术总策划，负责所有场景、角色、障碍物、音效及版面的设计与制作，参与玩家主界面及各种动态效果的构思与实现，对制作进程中的细节进行发掘和处理。 | | |
| 吴昊 | | | 负责部分功能的实现，主要包括碰撞的检测和判定、跳跃、游戏结束后的选择、最高分记录、多生命功能的实现；早期头文件的编写，主界面和游戏部分的黏合。 | | |
| 数据结构 | 本程序主要使用了Whale、Rect、Bonus结构体和wImg、wImgx数组，没有明显数据结构 | | | | | |
| 调试主要问题及思考 | 1. 透明贴图：因为物体绘制时需要去掉背景，所以如何实现透明贴图成了一大难题，最初是想通过导入Png实现，发现easyx不支持Png且只支持Bmp与Jpg格式。之后又从网上下载easyx第三方Png支持库，结果安装失败。最后通过绘制补码图和原图实现了透明贴图。 2. 随机障碍：一开始打算通过人工导入地图的方式来绘制地图，但发现数据不好处理，因此采用了随机算法每个一段时间生成障碍。在最初的Demo版本里，出现了随机物体组合导致小蓝鲸无法通过的情况，因为是纯随机生成，所以通过检测随机数以防出现通不过的障碍组合避免了该问题。 3. 操作捕捉：小蓝鲸移动过程中会出现“先动一下，再缓缓移动”的情况，也就是动作不流畅的问题。这一问题是由于使用getch()来捕捉按键导致的，getch()获取输入存在系统给定的0.5毫秒的间隔。之后我们实现了用GetAsyncKeystate解决了该问题，实现了平滑移动。与此同时还有一个问题是，玩家按住某一个键小蓝鲸会一直移动下去，灵敏度很高，之后我们设置了“按下并弹起”的条件，这样小蓝鲸的动作更加易于控制。 4. 移动问题：物体移动是游戏的一大难题，在最初物体移动是通过坐标自增减实现的，但物体的速度是通过Sleep控制的并且不稳定不直观。之后我们通过获取系统时间来测量间隔时间，据此计算出物体坐标，实现了物体更精确的移动。 5. 碰撞判定：物体碰撞需要知道坐标、长、宽三个量，而这和图片大小密切相关。我们游戏图片都是矩形，碰撞检测也以矩形为判定，如果物体的轮廓不接近矩形就会有很奇怪的碰撞判定，于是芮雨琛同学把物体做的更像矩形，缓解了该问题。 6. 图形移动过程中前一刻图像未擦除导致重复出现，在循环移动过程中不断覆盖背景图片以营造擦除效果；界面切换时鼠标不断获取信息进行判断，造成界面2的信息放到了界面1使用判断条件下，导致判断错误，界面切换乱套，在程序中放入多个goto结构使得信息去往正确的位置。 7. 跳跃功能：在实现跳跃功能时直接使用岳翔在物体移动中的相关函数会出现一些逻辑性和变量重复的问题，通过编写新的专用于跳跃判定的函数解决了这个问题。其他没有遇到问题。 | | | | | |
| 学习体会  和收获 | 岳翔 | | 见证了整个程序从最初起步到现在完成的全过程，可以说是非常具有成就感的一次程序设计。意想不到最初大家曾经想象过计划过的东西最终得以实现。在这个过程中团队内的每个人都功不可没，无论前端还是后端都很棒。所谓程序的精髓在于设计而非语法或者函数的运用，这次的程设正是加深了我对这一点的认知。 | | | |
| 芮雨琛 | | 最初的希望就是——努力制作出一款真正由我们设计和实现的高水准游戏。  即使它是一款只有一千行代码的小游戏，但是我们一直用玩家视角不断审视，比如鼠标移到按钮上的效果、跳跃和下落时的重力加速度等，我们将一开始天马行空的想象逐步落实到每一个细节。  为了这个目标，几乎每一个成员在前期都为了这款游戏熬夜到两三点甚至更晚：背景、角色、障碍物，都是亲自设计和制作，用PS、VS反复调试效果；为了更流畅更真实，碰撞判定、操作捕捉、物体移动，一次次讨论、修改……  但是当看到它一点点成长时，其中的喜悦溢于言表。  它小巧精致，倾注了我们所有的努力，而这些，都没有白费。 | | | |
| 吴昊 | | 从理论上说，一个完整的程序设计的开头应该将整个程序的整体结构和具体功能确定，但是由于对c语言的理解还不够，我们在早期的一些设定在后期被陆续推翻、修改、重建，当然主体框架得以保留；同时还有很多新的功能和想法在具体编写过程中出现。在这个过程中我们得到了类似于对游戏进行更新的体验，我从中认识到了一个良好的程序的框架的重要性：一个好的框架能够很好的适应修改和更新，对于维护也更加方便；而当程序现有框架不能很好的满足程序功能上的需求的时候应该果断重写框架，这样能够大大的提高效率。同时我还认识到了团队合作完成工程项目中沟通交流的重要性：在编写jump()函数的时候，因为我们是将岳翔同学的代码作为主体框架，在上面加以修改，而不同的代码编写习惯导致我直接阅读岳翔的代码时非常困难。在将jump()函数整合到主题代码中时，程序整体逻辑出现了很多问题，而后我加强了和岳翔之间的沟通和交流，后面的功能整合就基本没有遇到什么问题了。同时我也得到了一个启发：在编写团队合作的项目时最好确定一套统一的书写规范，同时多打注释多交流，这样能够很好的提高工作效率。 | | | |

模块功能介绍



程序框架图

源代码 Intergrity.h

#pragma once

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<graphics.h>

#include<math.h>

#include<time.h>

#pragma comment(lib, "Winmm.lib") // 给游戏添加音乐要用到它

#define UP 1

#define LEFT 2

#define DOWN 3

#define RIGHT 4

#define SPACE 5

#define Quantity 16

#define Q\_Rect 12

#define Q\_Bonus 4

#define ESC 27

#define SRC\_W 1000

#define SRC\_H 480

#define Interval\_godmode 5000

#define Interval\_talk 2000

#define Interval\_notalk 2000

#define TALKNUM 22

//主角

typedef struct Whale

{

IMAGE img;

IMAGE imgx;

int x; //矩形左上角x轴坐标

int y; //矩形左上角y轴坐标

int w; //矩形水平边像素长度

int h; //矩形竖直边像素长度

double v\_x;

double v\_y;

double a\_x;

double a\_y;

int st\_x; //起始坐标

int st\_y;

bool move;

double t0;

}Whale;

//方形

typedef struct Rect

{

IMAGE img;

IMAGE imgx;

int x; //矩形左上角x轴坐标

int y; //矩形左上角y轴坐标

int w; //矩形水平边像素长度

int h; //矩形竖直边像素长度

double v\_x;

double v\_y;

int st\_x; //起始坐标

int st\_y;

bool move;

double t0;

}Rect;

//奖励

typedef struct Bonus

{

IMAGE img;

IMAGE imgx;

int x; //矩形左上角x轴坐标

int y; //矩形左上角y轴坐标

int w; //矩形水平边像素长度

int h; //矩形竖直边像素长度

double v\_x;

double v\_y;

int st\_x; //起始坐标

int st\_y;

bool move;

bool collide;

double t0;

}Bonus;

//物体信息

typedef struct Data

{

int x; //矩形左上角x轴坐标

int y; //矩形左上角y轴坐标

int w; //矩形水平边像素长度

int h; //矩形竖直边像素长度

double v\_x;

double v\_y;

int st\_x; //起始坐标

int st\_y;

}Data;

//游戏变量

extern int Difficulty;

extern int dv;

extern int Interval;

extern double V\_X;

extern int talknum;

extern bool isgodmode;

extern bool talking;

extern int bonus\_score;

extern char str\_talk[TALKNUM][128];

extern int life;

extern int score\_max;

extern int score;

//系统变量

extern bool buttondown;

extern bool buttonup;

extern bool gameover;

extern bool exitgame;

extern IMAGE wImg[Quantity],wImgx[Quantity];

extern IMAGE bkimg;

extern int cmd;

extern bool spawn\_big;

extern bool moving;

extern int cmd\_exit;

//物体数组

extern Whale whale;

extern Rect rect[12];

extern Bonus bonus[4];

//时间变量

extern double dt;

extern int t0,t1;

extern int tmpt;

extern int hit;

extern int start1;

extern int godtime;

//文本信息

extern char str\_score[128];

extern char str\_interval[128];

extern char str\_bonus[128];

extern char bgm[4][20];

//空中跳跃用到的变量

extern int times\_jump,y0\_jump;

extern double t0\_jump,t1\_jump,dt\_jump;

extern bool jumping;

extern bool buttonspace;

extern int cmd\_jump;

//加载图片

void loadimgs()

{

int i;

loadimage(&bkimg,"res\\bkimgL.jpg");

char name[Quantity][20]={"whale","comb","island1","island2","land","rock","shrimp",

"squid","star1","star2","weed1","weed2","fish1","fish2","fish3","fish4"};

for(i=0;i<Quantity;i++){

char tmp[128];

sprintf\_s(tmp,"res\\%s.bmp",name[i]);

loadimage(&wImg[i],tmp);

sprintf\_s(tmp,"res\\%sx.bmp",name[i]);

loadimage(&wImgx[i],tmp);

}

}

//加载音效

void loadsfx()

{

mciSendString("open res\\easy.wma alias easy",NULL,0,NULL);

mciSendString("open res\\inwater.wma alias inwater",NULL,0,NULL);

mciSendString("open res\\outwater.wma alias outwater",NULL,0,NULL);

mciSendString("open res\\medium.wma alias medium",NULL,0,NULL);

mciSendString("open res\\hard.wma alias hard",NULL,0,NULL);

mciSendString("open res\\insane.wma alias insane",NULL,0,NULL);

mciSendString("open res\\gameover.wma alias gameover",NULL,0,NULL);

mciSendString("open res\\cantmove.wma alias cantmove",NULL,0,NULL);

mciSendString("open res\\move.wma alias move",NULL,0,NULL);

mciSendString("open res\\100.wma alias 100",NULL,0,NULL);

mciSendString("open res\\god.wma alias god",NULL,0,NULL);

mciSendString("open res\\youareweak.wma alias youareweak",NULL,0,NULL);

mciSendString("open res\\collide.wma alias collide",NULL,0,NULL);

mciSendString("open res\\jump.wma alias jump",NULL,0,NULL);

mciSendString("open res\\turn.wma alias turn",NULL,0,NULL);

}

void playbgm()

{

char str[128];

sprintf\_s(str,"play %s from 0 repeat",bgm[Difficulty-1]);

mciSendString(str,NULL,0,NULL);

}

void stopbgm()

{

char str[128];

sprintf\_s(str,"stop %s",bgm[Difficulty-1]);

mciSendString(str,NULL,0,NULL);

}

//字体设置

void settxt()

{

LOGFONT f;

gettextstyle(&f); // 获取当前字体设置

f.lfHeight=30; // 设置字体高度为 30

\_tcscpy\_s(f.lfFaceName,\_T("微软雅黑")); // 设置字体为“黑体”(高版本 VC 推荐使用 \_tcscpy\_s 函数)

f.lfQuality=ANTIALIASED\_QUALITY; // 设置输出效果为抗锯齿

settextstyle(&f);

}

//打印文本

void outtxt(bool showcoordinate=false)

{

score=((t1-t0)/1000+bonus\_score)\*Difficulty\*Difficulty;

sprintf\_s(str\_score,"得分:%d 生命:%d 历史最高分:%d",score,life,score\_max);

outtextxy(0,0,str\_score); //绘制位于左上角的分数

if(showcoordinate){

sprintf\_s(str\_interval,"系统间隔:%d HIT:%d 无敌间隔:%d",t1-start1,hit,t1-godtime);

outtextxy(SRC\_W-textwidth(str\_interval),0,str\_interval);

int h=textheight(str\_score);

char str[Quantity][128]={'\0'};

char rectname[Q\_Rect][20]={"whale","comb","island1","island2","land","rock","shrimp",

"squid","star1","star2","weed1","weed2"};

char bonusname[Q\_Bonus][20]={"fish1","fish2","fish3","fish4"};

int i;

for(i=0;i<Quantity;i++){

if(i==0) sprintf\_s(str[i],"%s x:%d y:%d",rectname[i],whale.x,whale.y);

else if(i<Q\_Rect) sprintf\_s(str[i],"%s x:%d y:%d",rectname[i],rect[i].x,rect[i].y);

else sprintf\_s(str[i],"%s x:%d y:%d",bonusname[i-Q\_Rect],bonus[i-Q\_Rect].x,bonus[i-Q\_Rect].y);

}

int h1=h;

for(i=0;i<Quantity;i++){

outtextxy(0,h1,str[i]);

h1+=h;

}

}

}

//绘制物体

void put(Whale obj)

{

int x=obj.x,y=obj.y;

putimage(x,y,&obj.imgx,SRCAND); //绘制补码图

putimage(x,y,&obj.img,SRCPAINT); //绘制原图

if(talking){

LOGFONT f;

gettextstyle(&f); // 获取当前字体设置

f.lfHeight=20; // 设置字体高度为 30

\_tcscpy\_s(f.lfFaceName,\_T("微软雅黑")); // 设置字体为“黑体”(高版本 VC 推荐使用 \_tcscpy\_s 函数)

f.lfQuality=PROOF\_QUALITY; // 设置输出效果为抗锯齿

settextstyle(&f);

outtextxy(obj.x,obj.y+whale.h,str\_talk[talknum%TALKNUM]);

settxt();

}

if(isgodmode){

int length=100\*(Interval\_godmode-(t1-godtime))/Interval\_godmode;

LOGFONT f;

gettextstyle(&f); // 获取当前字体设置

f.lfHeight=20; // 设置字体高度为 30

\_tcscpy\_s(f.lfFaceName,\_T("微软雅黑")); // 设置字体为“黑体”(高版本 VC 推荐使用 \_tcscpy\_s 函数)

f.lfQuality=PROOF\_QUALITY; // 设置输出效果为抗锯齿

settextstyle(&f);

outtextxy(obj.x,obj.y-40,"无敌模式");

setlinecolor(RED);

line(obj.x,obj.y-15,obj.x+length,obj.y-15);

settxt();

}

}

void put(Rect obj)

{

int x=obj.x,y=obj.y;

putimage(x,y,&obj.imgx,SRCAND); //绘制补码图

putimage(x,y,&obj.img,SRCPAINT); //绘制原图

}

void put(Bonus obj)

{

int x=obj.x,y=obj.y;

putimage(x,y,&obj.imgx,SRCAND); //绘制补码图

putimage(x,y,&obj.img,SRCPAINT); //绘制原图

}

//碰撞检测

bool collision\_judgment(Whale whale,Rect \*rect)

{

bool case1,case2;

case1=(whale.x+whale.w)>rect->x && (whale.x-rect->w)<rect->x;

case2=(whale.y+whale.h)>rect->y && (whale.y-rect->h)<rect->y;

if(case1 && case2) return true;

else return false;

}

bool collision\_judgment(Whale whale,Bonus \*rect)

{

bool case1,case2;

case1=(whale.x+whale.w)>rect->x && (whale.x-rect->w)<rect->x;

case2=(whale.y+whale.h)>rect->y && (whale.y-rect->h)<rect->y;

if(case1 && case2) return true;

else return false;

}

//以bkimg刷新背景

void flush()

{

putimage(0,0,&bkimg);

} //用bkimg刷新画面

//停止物体

void stopall()

{

int i;

for(i=0;i<Quantity;i++){

if(i<Q\_Rect) rect[i].move=false;

else bonus[i-Q\_Rect].move=false;

}

}

//移动物体

void move(Rect \*obj)

{

if(obj->move==false){

return ;

}

else{

double t1=clock();

double dt=(t1-obj->t0)/1000;

int st\_x=obj->st\_x,st\_y=obj->st\_y;

obj->x=(int)(st\_x+obj->v\_x\*dt);

//obj->y=(int)(st\_y+obj->v\_y\*dt);

if((obj->x)<-500){

obj->move=false;

obj->x=obj->st\_x;

obj->y=obj->st\_y;

}

put(\*obj);

if(isgodmode==false && collision\_judgment(whale,obj)){

mciSendString("play collide from 0",NULL,0,NULL);

gameover=true;

}

}

}

void move(Bonus \*obj)

{

if(obj->move==false){

return ;

}

else if(obj->collide){

outtextxy(obj->x,obj->y,str\_bonus);

obj->y-=1;

if(fabs(obj->y-obj->st\_y)>200){

obj->move=false;

obj->collide=false;

}

return ;

}

else{

double t1=clock();

double dt=(t1-obj->t0)/1000;

int st\_x=obj->st\_x,st\_y=obj->st\_y;

obj->x=(int)(st\_x+obj->v\_x\*dt);

obj->y=(int)(st\_y+obj->v\_y\*dt);

if((obj->x)<-500){

obj->move=false;

obj->x=obj->st\_x;

obj->y=obj->st\_y;

}

put(\*obj);

if(collision\_judgment(whale,obj)){

int randnum=rand()%100;

if(randnum<60){

bonus\_score+=100;

obj->collide=true;

sprintf\_s(str\_bonus,"+%d00分",Difficulty\*Difficulty);

mciSendString("play 100 from 0",NULL,0,NULL);

}

else{

isgodmode=true;

obj->collide=true;

strcpy\_s(str\_bonus,"无敌模式");

mciSendString("play god from 0",NULL,0,NULL);

}

}

}

}

void moveall()

{

int i;

for(i=0;i<Quantity;i++){

if(i<Q\_Rect) move(&rect[i]);

else move(&bonus[i-Q\_Rect]);

}

}

void startmove(Rect \*obj)

{

obj->move=true;

obj->t0=clock();

}

void startmove(Bonus \*obj)

{

obj->move=true;

obj->t0=clock();

}

//摸奖

void dice(bool spawnrect,bool spawnbonus){

if(spawnrect){

if(spawn\_big==false){

int randnum=rand()%(Q\_Rect-1)+1;

if(rect[randnum].move==false){

if(randnum==1 || randnum==4) spawn\_big=true;

startmove(&rect[randnum]);

}

else{

dice(true,false);

}

}

else{

int randnum;

do{

randnum=rand()%(Q\_Rect-1)+1;

} while(randnum==10);

if(rect[randnum].move==false){

if(randnum==1 || randnum==4) spawn\_big=true;

else spawn\_big=false;

startmove(&rect[randnum]);

}

else{

dice(true,false);

}

}

}

else{

int randnum=rand()%Q\_Bonus;

if(bonus[randnum].move==false){

startmove(&bonus[randnum]);

}

else{

dice(true,false);

}

}

}

//获取操作

void getcmd()

{

cmd=0;

//if(GetAsyncKeyState(65) & 0x8000) cmd=LEFT;

//if(GetAsyncKeyState(68) & 0x8000) cmd=RIGHT;

if(GetAsyncKeyState(87)&0x8000 || GetAsyncKeyState(VK\_UP)&0x8000) buttonup=true;

if(GetAsyncKeyState(83)&0x8000 || GetAsyncKeyState(VK\_DOWN)&0x8000) buttondown=true;

if(buttonup){

if(!(GetAsyncKeyState(87)&0x8000) && !(GetAsyncKeyState(VK\_UP)&0x8000)){

buttonup=false;

cmd=UP;

}

}

if(buttondown){

if(!(GetAsyncKeyState(83)&0x8000) && !(GetAsyncKeyState(VK\_DOWN)&0x8000)){

buttondown=false;

cmd=DOWN;

}

}

/\*if(GetAsyncKeyState(VK\_SPACE) & 0x8000){

outtextxy(100,100,"游戏暂停");

system("pause");

}\*/

if(GetAsyncKeyState(27)&0x8000) cmd=ESC;

}

void getcmd\_jump()

{

cmd\_jump=0;

if(GetAsyncKeyState(VK\_SPACE)&0x8000) buttonspace=true;

if(!(GetAsyncKeyState(VK\_SPACE)&0x8000) && buttonspace)

{

cmd\_jump=SPACE;

buttonspace=false;

}

}

void jump()

{

if(whale.y>175)

{

mciSendString("play inwater from 0",NULL,0,NULL);

jumping=false;

moving=false;

times\_jump=0;

}

t1\_jump=clock();

dt\_jump=t1\_jump-t0\_jump;

if(jumping == true) whale.y=(int)(y0\_jump+whale.v\_y\*dt\_jump+0.5\*whale.a\_y\*dt\_jump\*dt\_jump);

else whale.y = 175;

}

void jump\_main()

{

if(jumping==true && times\_jump>=2)

{

jump();

}

else if(jumping==true && times\_jump<2)

{

if(cmd\_jump==SPACE)

{

mciSendString("play jump from 0",NULL,0,NULL);

t0\_jump=clock();

y0\_jump=whale.y;

jump();

times\_jump++;

}

else jump();

}

else if(cmd\_jump==SPACE)

{

t0\_jump=clock();

y0\_jump=175;

jump();

times\_jump++;

jumping=true;

moving=true;

mciSendString("play jump from 0",NULL,0,NULL);

mciSendString("play outwater from 0",NULL,0,NULL);

}

}

void getcmd\_exit()

{

cmd\_exit=0;

if(GetAsyncKeyState(87) & 0x8000 || GetAsyncKeyState(VK\_UP) & 0x8000) buttonup=true;

if(GetAsyncKeyState(83) & 0x8000 || GetAsyncKeyState(VK\_DOWN) & 0x8000) buttondown=true;

if(buttonup) {

if(!(GetAsyncKeyState(87) & 0x8000) && !(GetAsyncKeyState(VK\_UP)&0x8000)) {

buttonup=false;

cmd\_exit=UP;

}

}

if(buttondown) {

if(!(GetAsyncKeyState(83) & 0x8000) && !(GetAsyncKeyState(VK\_DOWN) & 0x8000)) {

buttondown=false;

cmd\_exit=DOWN;

}

}

if(GetAsyncKeyState(27) & 0x8000) cmd\_exit=ESC;

}

void print\_exit(int t)

{

char str1[128];

sprintf\_s(str1,"按ESC键返回主界面,W或↑继续游戏,选择剩余时间:%ds",t);

outtextxy(SRC\_W/2-textwidth(str1)/2,SRC\_H/2-textheight(str1)/2,str1);

}

//主界面

void change(int a,int b,IMAGE flower,IMAGE maskbitmap)

{

putimage(a,b,&maskbitmap,SRCAND);

putimage(a,b,&flower,SRCPAINT);

}

int inter\_face(void)

{

initgraph(640,480);

MOUSEMSG m;

IMAGE main1,main2,main3,main4,main5,main11,main11y,main12,main12y,main13,main13y;

IMAGE main14,main21,main22,main23,main24,main31,main51,main52,main53,main54,main55;

IMAGE main14y,main21y,main22y,main23y,main24y,main31y,main51y,main52y,main53y,main54y,main55y;

loadimage(&main1,"res\\main1.bmp");//天空主背景

loadimage(&main2,"res\\main2.bmp");// 菜单背景

loadimage(&main3,"res\\main3.bmp");//人员介绍背景

loadimage(&main4,"res\\main4.bmp");//规则介绍背景

loadimage(&main5,"res\\main5.bmp");//难度选择背景

loadimage(&main11,"res\\main11.bmp");//游戏名称图像

loadimage(&main11y,"res\\main11y.bmp");//游戏名称掩码图

loadimage(&main12,"res\\main12.bmp");//小蓝鲸图像

loadimage(&main12y,"res\\main12y.bmp");//小蓝鲸图像掩码图

loadimage(&main13,"res\\main13.bmp");//气泡图像

loadimage(&main13y,"res\\main13y.bmp");//气泡图像掩码图

loadimage(&main14,"res\\main14.bmp");//欢迎字体

loadimage(&main14y,"res\\main14y.bmp");//欢迎字体掩码图

loadimage(&main21,"res\\main21.bmp");//菜单栏游戏介绍

loadimage(&main21y,"res\\main21y.bmp");//菜单栏游戏介绍掩码图

loadimage(&main22,"res\\main22.bmp");//菜单栏团队介绍

loadimage(&main22y,"res\\main22y.bmp");//菜单栏团队介绍掩码图

loadimage(&main23,"res\\main23.bmp");//菜单栏开始游戏

loadimage(&main23y,"res\\main23y.bmp");//菜单栏开始游戏掩码图

loadimage(&main24,"res\\main24.bmp");//菜单栏退出游戏

loadimage(&main24y,"res\\main24y.bmp");//菜单栏退出游戏掩码图

loadimage(&main31,"res\\main31.bmp");//back按钮

loadimage(&main31y,"res\\main31y.bmp");//back按钮掩码图

loadimage(&main51,"res\\main51.bmp");//难度选择：脑残

loadimage(&main51y,"res\\main51y.bmp");//难度选择：脑残掩码图

loadimage(&main52,"res\\main52.bmp");//难度选择：正常

loadimage(&main52y,"res\\main52y.bmp");//难度选择：正常掩码图

loadimage(&main53,"res\\main53.bmp");//难度选择：不正常

loadimage(&main53y,"res\\main53y.bmp");//难度选择：不正常掩码图

loadimage(&main54,"res\\main54.bmp");//难度选择：肯定过不了

loadimage(&main54y,"res\\main54y.bmp");//难度选择：肯定过不了掩码图

//游戏开始欢迎动画 ,背景音乐

mciSendString("open res\\intro.wma alias interfacemusic",NULL,0,NULL);

mciSendString("open res\\turn.wma alias turn",NULL,0,NULL);

mciSendString("setaudio turn volume to 999",NULL,0,NULL);

mciSendString("play interfacemusic repeat",NULL,0,NULL);

//起始页动画

int i;

BeginBatchDraw();

/\*putimage(0,0,&main1);

for(i=0;i<=30;i++){

putimage(0,0,&main1);

change(100,i,main11,main11y);

FlushBatchDraw();

Sleep(10);

}

for(i=0;i<=30;i++){

putimage(0,0,&main1);

change(100,30,main11,main11y);

change(i,200,main12,main12y);

FlushBatchDraw();

Sleep(10);

}

for(i=480;i>=300;i--){

putimage(0,0,&main1);

change(100,30,main11,main11y);

change(30,200,main12,main12y);

change(i,210,main13,main13y);

FlushBatchDraw();

Sleep(1);

}

Sleep(200);

putimage(0,0,&main1);

change(100,30,main11,main11y);

change(30,200,main12,main12y);

change(300,210,main13,main13y);

change(320,220,main14,main14y);

FlushBatchDraw();

EndBatchDraw();\*/

for(i=1;i<=44;i++){

char tmp[128];

IMAGE bk;

sprintf\_s(tmp,"res\\%d.jpg",i+1);

loadimage(&bk,tmp);

putimage(0,0,&bk);

FlushBatchDraw();

Sleep(50);

}

Sleep(2000);

//用于返回跳跃

loop1:{

putimage(0,0,&main2);

while(true){

m=GetMouseMsg();

//选择游戏介绍

if(m.x>98&&m.x<213&&m.y>148&&m.y<191){

change(75,140,main21,main21y);

FlushBatchDraw();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("play turn from 0",NULL,0,NULL);

loop2:{

putimage(0,0,&main4);

FlushBatchDraw();

m=GetMouseMsg();

if(m.x>531&&m.x<623&&m.y>403&&m.y<468){

loop7:{

change(540,433,main31,main31y);

FlushBatchDraw();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("play turn from 0",NULL,0,NULL);

goto loop1;

}

m=GetMouseMsg();

if(m.x>531&&m.x<623&&m.y>403&&m.y<468){

goto loop7;

}

else goto loop2;

}

}

else goto loop2;

}

}

}

//选择团队介绍

if(m.x>98&&m.x<213&&m.y>247&&m.y<289){

change(75,239,main22,main22y);

FlushBatchDraw();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("play turn from 0",NULL,0,NULL);

loop3:{

putimage(0,0,&main3);

FlushBatchDraw();

m=GetMouseMsg();

if(m.x>531&&m.x<623&&m.y>403&&m.y<468){

loop6:{

change(540,433,main31,main31y);

FlushBatchDraw();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("play turn from 0",NULL,0,NULL);

goto loop1;

}

m=GetMouseMsg();

if(m.x>531&&m.x<623&&m.y>403&&m.y<468){

goto loop6;

}

else goto loop3;

}

}

else goto loop3;

}

}

}

//选择开始游戏

if(m.x>98&&m.x<213&&m.y>345&&m.y<391){

change(75,342,main23,main23y);

FlushBatchDraw();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("play turn from 0",NULL,0,NULL);

loop4:{

putimage(0,0,&main5);

FlushBatchDraw();

m=GetMouseMsg();

if(m.x>551&&m.x<623&&m.y>433&&m.y<468){

loop8:{

change(540,433,main31,main31y);

FlushBatchDraw();

m=GetMouseMsg();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("play turn from 0",NULL,0,NULL);

goto loop1;

}

if(m.x>551&&m.x<623&&m.y>433&&m.y<468){

goto loop8;}

else goto loop4;

}

}

//选择脑残模式，返回 1

if(m.x>98&&m.x<213&&m.y>156&&m.y<192){

loop9:{

change(70,150,main51,main51y);

FlushBatchDraw();

m=GetMouseMsg();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("stop interfacemusic",NULL,0,NULL);

mciSendString("close interfacemusic",NULL,0,NULL);

closegraph();

return 1;

}

if(m.x>98&&m.x<213&&m.y>156&&m.y<192)

goto loop9;

else goto loop4;

}

}

//选择正常模式，返回 2

if(m.x>98&&m.x<213&&m.y>236&&m.y<274){

loop10:{

change(70,230,main52,main52y);

FlushBatchDraw();

m=GetMouseMsg();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("stop interfacemusic",NULL,0,NULL);

mciSendString("close interfacemusic",NULL,0,NULL);

closegraph();

return 2;

}

if(m.x>98&&m.x<213&&m.y>236&&m.y<274)

goto loop10;

else goto loop4;

}

}

//选择不正常模式，返回3

if(m.x>98&&m.x<213&&m.y>316&&m.y<354){

loop11:{

change(70,310,main53,main53y);

FlushBatchDraw();

m=GetMouseMsg();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("stop interfacemusic",NULL,0,NULL);

mciSendString("close interfacemusic",NULL,0,NULL);

closegraph();

return 3;

}

if(m.x>98&&m.x<213&&m.y>316&&m.y<354)

goto loop11;

else goto loop4;

}

}

//选择肯定过不了模式，返回 3

if(m.x>98&&m.x<213&&m.y>396&&m.y<436){

loop12:{

change(70,390,main54,main54y);

FlushBatchDraw();

m=GetMouseMsg();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("stop interfacemusic",NULL,0,NULL);

mciSendString("close interfacemusic",NULL,0,NULL);

closegraph();

return 4;

}

if(m.x>98&&m.x<213&&m.y>396&&m.y<436)

goto loop12;

else goto loop4;

}

}

if((m.x>551&&m.x<623&&m.y>433&&m.y<468||m.x>98&&m.x<213&&m.y>156&&m.y<192&&m.uMsg==WM\_LBUTTONDOWN||m.x>98&&m.x<213&&m.y>236&&m.y<274&&m.uMsg==WM\_LBUTTONDOWN||m.x>98&&m.x<213&&m.y>316&&m.y<354&&m.uMsg==WM\_LBUTTONDOWN||m.x>98&&m.x<213&&m.y>396&&m.y<436&&m.uMsg==WM\_LBUTTONDOWN)==0)

goto loop4;

}

}

}

//选择退出游戏

if(m.x>480&&m.x<623&&m.y>410&&m.y<468){

loop5:{

change(475,418,main24,main24y);

FlushBatchDraw();

m=GetMouseMsg();

if(m.uMsg==WM\_LBUTTONDOWN){

mciSendString("stop interfacemusic",NULL,0,NULL);

mciSendString("close interfacemusic",NULL,0,NULL);

closegraph();

return -1;

}

if(m.x>480&&m.x<623&&m.y>410&&m.y<468)

goto loop5;

else goto loop1;

}

}

if((m.x>98&&m.x<213&&m.y>148&&m.y<191||m.x>98&&m.x<213&&m.y>247&&m.y<289||m.x>98&&m.x<213&&m.y>345&&m.y<391||m.x>502&&m.x<623&&m.y>427&&m.y<468)==0){

putimage(0,0,&main2);

FlushBatchDraw();

}

}

EndBatchDraw();

}

}

源代码 Crazy whale.cpp

#include"Integrity.h"

//临时变量

//系统变量

bool buttondown;

bool buttonup;

bool gameover=false; //判断是否碰撞

bool exitgame=false;

IMAGE wImg[Quantity],wImgx[Quantity];

IMAGE bkimg;

int cmd; //用户输入

int cmd\_exit;

bool spawn\_big=false; //判断产生大物体

bool moving=false; //判断蓝鲸是否在移动

int pos[4]={175,255,335,400}; //蓝鲸y位置数组

int level; //蓝鲸位置

errno\_t err;

//游戏变量

int Difficulty;

int dv;

int Interval;

int talknum;

double V\_X;

bool isgodmode;

bool talking;

int bonus\_score;

int life;

int score\_max;

int score;

char str\_talk[TALKNUM][128]={

"你好呀( • ̀ω•́ )✧",

"欢迎你哦(づ｡◕ᴗᴗ◕｡)づ",

"我的中文名叫“小蓝鲸”",

"英文名叫……‘blue whale’ (〃'▽'〃)",

"如果你是手残党，初次尝试请选择‘脑残’关卡(≖ᴗ≖)✧",

"如果你是six的话，那就选择‘正常’关卡吧(\*•ω•)",

"啥？你想玩‘过不去’(｀ﾟДﾟ´)ゞ？嘿嘿嘿……",

"你以为‘过不去’是能够轻易尝试的吗？(￣^￣)"

"否则它为什么叫‘过不去’？╮(╯▽╰)╭",

"你要是实在想玩",

"我也没办法阻止你不是⊂(￣(ｴ)￣)⊃",

"按w和s或者↑和↓来控制我呦！",

"按空格键我就跳起来啦~",

"游啊游，游到九乡河(\*/ω＼\*)……",

"这里好美啊٩(๑>◡<๑)۶ ~ ",

"我萌不萌呀(◕ᴗ◕✿)",

"头上的角是不是超级独特！不二的对不对！(︶.̮︶✽)",

"翔子他们把我做出来花了很多功夫呢(..›ᴗ‹..)",

" emmmmmm……加油ヽ(￣▽￣)ﾉ",

"我有点累了\_(:з」∠)\_",

"大神你坚持了好久啊✿✿ヽ(ﾟ▽ﾟ)ノ✿✿！",

"棒得不行哦！偷偷看你|ω・）"

};

//物体数组

Whale whale;

Rect rect[12];

Bonus bonus[4];

/\*

//物体变量信息清单 img imgx x y w h v\_x v\_y

Whale whale： {wImg[0],wImgx[0] {75 , 175,58, 48},

Rect comb： {wImg[1],wImgx[1] {1000,44, 262,160,V\_X,0 },

Rect island1： {wImg[2],wImgx[2] {1000,109,57 ,94,V\_X,0 },

Rect island2： {wImg[3],wImgx[3] {1000,110,70 ,98,V\_X,0 },

Rect land： {wImg[4],wImgx[4] {1000,136,236,75 ,V\_X,0 },

Rect rock： {wImg[5],wImgx[5] {1000,335,246,144,V\_X,0 },

Rect shrimp： {wImg[6],wImgx[6] {1000,265,43, 58 ,V\_X,0 },

Rect squid： {wImg[7],wImgx[7] {1000,279,12, 94,V\_X,0 },

Rect star1： {wImg[8],wImgx[8] {1000,428,37, 36 ,V\_X,0 },

Rect star2： {wImg[9],wImgx[9] {1000,265,43, 39 ,V\_X,0 },

Rect weed1： {wImg[10],wImgx[10] {1000,261,31 ,219,V\_X,0 },

Rect weed2： {wImg[11],wImgx[11] {1000,335,37,148,V\_X,0 },

Bonus fish1： {wImg[12],wImgx[12] {1000,189,39, 32 ,V\_X,0 },

Bonus fish2： {wImg[13],wImgx[13] {1000,270,51, 25 ,V\_X,0 },

Bonus fish3： {wImg[14],wImgx[14] {1000,340,41, 20 ,V\_X,0 },

Bonus fish4： {wImg[15],wImgx[15] {1000,442,42, 18 ,V\_X,0 },

\*/

//时间变量

double dt;

int t0,t1;

int tmpt=t0;

int hit=0;

int start1;

int godtime;

int talktime;

//文本信息

char str\_score[128];

char str\_interval[128];

char str\_bonus[128];

char bgm[4][20]={"easy","medium","hard","insane"};

//空中跳跃用到的变量

int times\_jump=0,y0\_jump,jumpornot=0;

double t0\_jump,t1\_jump,dt\_jump,V\_Y,A\_Y;

bool jumping=false;

int cmd\_jump;

bool buttonspace;

//主函数

int main(){

int i;

FILE \*fpr,\*fpw;

if(err=fopen\_s(&fpr,"max.txt","r"))

{

puts("Cannot open file max!");

exit(1);

}

fscanf\_s(fpr,"%d",&score\_max);

fclose(fpr);

//加载图片

loadimgs();

//加载音效

loadsfx();

while(1)

{

initgraph(640,SRC\_H);

Difficulty=inter\_face();

closegraph();

initgraph(SRC\_W,SRC\_H);

if(Difficulty==-1) break;

switch(Difficulty) {

case 1: {

dv=1;

Interval=2000;

V\_X=-125;

V\_Y=-0.2;

A\_Y=0.0002;

break;

}

case 2: {

dv=2;

Interval=1000;

V\_X=-250;

V\_Y=-0.4;

A\_Y=0.0008;

break;

}

case 3: {

dv=5;

Interval=500;

V\_X=-500;

V\_Y=-0.8;

A\_Y=0.0032;

break;

}

case 4: {

dv=15;

Interval=250;

V\_X=-1000;

V\_Y=-1.6;

A\_Y=0.0128;

break;

}

}

Data data[Quantity]={

{75,175,58,48,0,V\_Y},

{1000,44,262,160,V\_X,0},

{1000,109,57,94,V\_X,0},

{1000,110,70,98,V\_X,0},

{1000,136,236,75,V\_X,0},

{1000,335,246,144,V\_X,0},

{1000,265,43,58,V\_X,0},

{1000,279,12,94,V\_X,0},

{1000,428,37,36,V\_X,0},

{1000,265,43,39,V\_X,0},

{1000,261,31,219,V\_X,0},

{1000,335,37,148,V\_X,0},

{1000,189,39,32,V\_X,0},

{1000,270,51,25,V\_X,0},

{1000,340,41,20,V\_X,0},

{1000,442,42,18,V\_X,0},

};

//Sleep(10000);

//物体初始化

whale.img=wImg[0];

whale.imgx=wImgx[0];

whale.x=data[0].x;

whale.y=data[0].y;

whale.h=data[0].h;

whale.w=data[0].w;

whale.v\_x=data[0].v\_x;

whale.v\_y=data[0].v\_y;

whale.a\_y=A\_Y;

whale.st\_x=data[0].st\_x;

whale.st\_y=data[0].st\_y;

for(i=1; i<Quantity; i++) {

if(i<Q\_Rect) {

rect[i].img=wImg[i];

rect[i].imgx=wImgx[i];

rect[i].x=data[i].x;

rect[i].y=data[i].y;

rect[i].h=data[i].h;

rect[i].w=data[i].w;

rect[i].v\_x=data[i].v\_x;

rect[i].v\_y=data[i].v\_y;

rect[i].st\_x=data[i].st\_x;

rect[i].st\_y=data[i].st\_y;

}

else {

bonus[i-Q\_Rect].img=wImg[i];

bonus[i-Q\_Rect].imgx=wImgx[i];

bonus[i-Q\_Rect].x=data[i].x;

bonus[i-Q\_Rect].y=data[i].y;

bonus[i-Q\_Rect].h=data[i].h;

bonus[i-Q\_Rect].w=data[i].w;

bonus[i-Q\_Rect].v\_x=data[i].v\_x;

bonus[i-Q\_Rect].v\_y=data[i].v\_y;

bonus[i-Q\_Rect].st\_x=data[i].st\_x;

bonus[i-Q\_Rect].st\_y=data[i].st\_y;

}

}

for(i=0; i<Quantity; i++) {

if(i<Q\_Rect) {

rect[i].st\_x=rect[i].x;

rect[i].st\_y=rect[i].y;

rect[i].move=false;

}

else {

bonus[i-Q\_Rect].st\_x=bonus[i-Q\_Rect].x;

bonus[i-Q\_Rect].st\_y=bonus[i-Q\_Rect].y;

bonus[i-Q\_Rect].move=false;

}

}

setlinestyle(PS\_SOLID | PS\_ENDCAP\_ROUND,3);

//物体部分参数设置

//暂无

//游戏部分

//字体初始化

settxt();

settextcolor(WHITE);

setbkmode(TRANSPARENT);

//测试物体位置

/\*

BeginBatchDraw();

for(i=0;i<Q\_Rect;i++){

MOUSEMSG m;

while(true){

m=GetMouseMsg();

if(m.mkLButton) break;

flush();

outtxt(true);

rect[i].x=m.x;

rect[i].y=m.y;

put(rect[i]);

FlushBatchDraw();

}

}

for(i=0;i<Q\_Bonus;i++){

MOUSEMSG m;

while(true){

m=GetMouseMsg();

if(m.mkLButton) break;

flush();

outtxt(true);

bonus[i].x=m.x;

bonus[i].y=m.y;

put(bonus[i]);

FlushBatchDraw();

}

}

EndBatchDraw();

closegraph();

return 0;

\*/

while(1)

{

flush();

BeginBatchDraw(); //开始批量绘图

t0=clock(); //游戏开始时间

start1=t0;

life=3;

bonus\_score=0;

gameover=false;

exitgame=false;

playbgm();

while(!gameover&&life>0)

{

//字体设置

settxt();

//获取起始时间

srand((unsigned int)(time(NULL))); //随机数种子设置

dice(true,false);

level=1;

godtime=0;

isgodmode=false;

talking=false;

talktime=0;

talknum=0;

stopall();

whale.y=175;

jumping=false;

moving=false;

times\_jump=0;

//开始游戏

while(true) {

flush(); //重绘背景图

t1=clock(); //计算新时间

dt=(t1-tmpt)\*0.001;

outtxt(false); //输出左上角文本

if(gameover) {

stopbgm();

mciSendString("play gameover from 0",NULL,0,NULL);

Sleep(1000);

mciSendString("play youareweak from 0",NULL,0,NULL);

cleardevice();

char str[128];

sprintf\_s(str,"生命还剩下:%d",life-1);

outtextxy(SRC\_W/2-textwidth(str)/2,SRC\_H/2-textheight(str)/2,str);

FlushBatchDraw();

Sleep(2000);

playbgm();

break;

}

if(exitgame) break;

//无敌状态

if(isgodmode) {

if(godtime==0) godtime=t1;

if(t1-godtime>Interval\_godmode) {

isgodmode=false;

godtime=0;

}

}

//说话

if(!talking && t1-talktime>Interval\_notalk) {

talking=true;

talktime=t1;

}

if(talking && t1-talktime>Interval\_talk) {

talking=false;

talknum++;

talktime=t1;

}

if(!moving) {

getcmd();

switch(cmd) {

case UP: {

if(level>1) {

level--;

moving=true;

mciSendString("play move from 0",NULL,0,NULL);

}

else {

mciSendString("play cantmove from 0",NULL,0,NULL);

}

break;

}

case DOWN: {

if(level<4) {

level++;

moving=true;

mciSendString("play move from 0",NULL,0,NULL);

}

else {

mciSendString("play cantmove from 0",NULL,0,NULL);

}

break;

}

case ESC:exitgame=true; break;

}

}

if(moving) {

switch(cmd) {

case UP: {

whale.y-=dv;

if(fabs(whale.y-pos[level-1])<=dv) {

moving=false;

whale.y=pos[level-1];

}

break;

}

case DOWN: {

whale.y+=dv;

if(fabs(whale.y-pos[level-1])<=dv) {

moving=false;

whale.y=pos[level-1];

}

break;

}

case ESC:exitgame=true; break;

}

}

if(level==1)

{

getcmd\_jump();

jump\_main();

}

put(whale); //绘制蓝鲸

if((t1-start1)>Interval) {

if(hit%20==0) {

dice(false,true);

start1=t1;

hit++;

}

else {

dice(true,false);

start1=t1;

hit++;

}

}

moveall(); //移动物体

FlushBatchDraw(); //刷新画面

Sleep(1);

}

life--;

gameover=false;

}

if(!exitgame) gameover=true;

EndBatchDraw(); //结束批量绘制

flush();

if(gameover) {

stopbgm();

score=((t1-t0)/1000+bonus\_score)\*Difficulty\*Difficulty;

sprintf\_s(str\_score,"您的最终得分:%d",score);

outtextxy(SRC\_W/2-textwidth(str\_score)/2,SRC\_H/2-textheight(str\_score)/2,str\_score);

Sleep(3000);

if(err=fopen\_s(&fpw,"max.txt","w"))

{

puts("Cannot open file max.txt");

exit(1);

}

if(score>=score\_max)

{

flush();

score\_max=score;

sprintf\_s(str\_score,"您创造了新的记录:%d！",score);

outtextxy(SRC\_W/2-textwidth(str\_score)/2,SRC\_H/2-textheight(str\_score)/2,str\_score);

Sleep(3000);

}

fprintf(fpw,"%d",score\_max);

fclose(fpw);

}

flush();

bool exitjudge=false;

int t0\_exitj,t1\_exitj;

t0\_exitj=clock();

BeginBatchDraw();

while(1)

{

int dt,t;

t1\_exitj=clock();

dt=t1\_exitj-t0\_exitj;

if(dt<1000)

{

t=5;

flush();

print\_exit(t);

}

else if(dt>=1000&&dt<2000)

{

t=4;

flush();

print\_exit(t);

}

else if(dt>=2000&&dt<3000)

{

t=3;

flush();

print\_exit(t);

}

else if(dt>=3000&&dt<4000)

{

t=2;

flush();

print\_exit(t);

}

else if(dt>=4000)

{

t=1;

flush();

print\_exit(t);

}

FlushBatchDraw();

getcmd\_exit();

if(cmd\_exit==ESC)

{

exitgame=true;

break;

}

else if(cmd\_exit==UP)

{

exitgame=false;

break;

}

else if(t1\_exitj-t0\_exitj>5000)

{

exitgame=true;

break;

}

}

EndBatchDraw();

flush();

cmd=0;

if(exitgame) {

stopbgm();

char str[128];

sprintf\_s(str,"感谢游戏！");

outtextxy(SRC\_W/2-textwidth(str)/2,SRC\_H/2-textheight(str)/2,str);

Sleep(2000);

break;

}

}

}

closegraph();

return 0;

}