目录

[1）产品概述 2](#_Toc138591202)

[1.1项目相关背景和调研情况 2](#_Toc138591203)

[1.2项目摘要 2](#_Toc138591204)

[1.3项目的类型及规则说明 2](#_Toc138591205)

[2） 项目开发设计 3](#_Toc138591206)

[2.1项目的系统结构、各模块/组件的描述 3](#_Toc138591207)

[2.2基本的对象和主要逻辑 5](#_Toc138591208)

[2.3主要的数据结构和用户界面的定义 5](#_Toc138591209)

[2.4核心算法的代码实现 7](#_Toc138591210)

[2.5源码 7](#_Toc138591211)

[3）项目测试与总结 46](#_Toc138591212)

[3.1 bug和debug 46](#_Toc138591213)

[3.2总结与不足 46](#_Toc138591214)

## 1）产品概述

### 1.1项目相关背景和调研情况

《超级马里奥》是任天堂开发的一款非常经典甚至是传奇的游戏。根据前期的调研显示，大部分游戏玩家对此游戏予以高度评价，并且对游玩此类游戏十分感兴趣。基于以上原因，我尝试制作了这个游戏。

### 1.2项目摘要

《关于我一定要救出前辈这件事》是基于《超级马里奥》的背景，利用C语言的easyX图形库来开发的一款游戏。游戏的设定主要是，玩家操纵一个下水管道工（马里奥的晚辈）来拯救被反派抓走的马里奥。游戏分为两个部分，第一部分是教学部分，难度简单。第二个部分是实战部分，难度稍高。实战部分玩家需要抓住时机，躲开地刺以及其他陷阱，然后拿到钥匙，解救马里奥前辈。

### 1.3项目的类型及规则说明

该游戏的类型主要是2D的闯关型游戏。

操作机制：玩家通过键盘的**AD**键来**左右移动**，**空格键**来**跳跃**以及**继续对话**。按键盘上的**up键和down键**（上下箭头）来选择**对话中的选项**。

死亡机制：

玩家开局有**五条命**，在教学关卡中，在获得星星后碰撞小怪会扣血，踩小怪可以踩死不扣血；在实战关卡中，碰到地刺会扣血，摔入深渊会扣血。

每一次扣血之后都会返回存档点。重新挑战是，存档点会随着剧情自动刷新。

奖励机制：游戏中会有星星，在主角碰撞之后会推进剧情。

## 2） 项目开发设计

### 2.1项目的系统结构、各模块/组件的描述

文本

描述已自动生成全项目共约1500行，main()函数结构如右图。

控制游戏的最基础代码如下（以下文档将其称为while结构）

init();//初始化

while (1)

{

show();

updateWithInput();

updateWithoutInput();

}

其中在while中把(1)换成用于判定是否过关的isStart等参数来控制游戏的进行。

**show()**函数主要用于展示游戏内容

**updateWithInput()**函数主要用于接受用户的输入。

**updateWithoutInput()**函数主要用于更新游戏进行时改变的参数

文本

描述已自动生成

其中，各个关卡的内容用**结构体**来写。

以一个结构体为一个组件，并且用结构体内定义的函数来分别调用函数名一样但函数内容不一样的函数。

在结构体中写出各个关卡需要用到的参数，如右图

以Surface结构体为例

IMAGE参数用于存储封面图像文本

描述已自动生成 标题 以及开始游戏的按钮

然后用while结构来控制游戏的进入

以Before结构体为例。

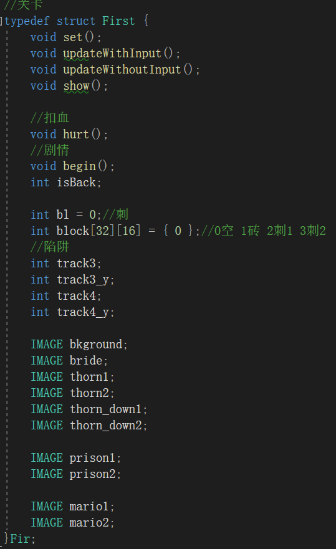
文本

描述已自动生成set()函数用于初始化Before结构体的内容，如左图。

inputname() introduce()等函数用于教学关卡的流程控制，内部的基础结构也是while结构

IMAGE的参数用于保存背景图片 该地形图片等

isStar isfinish star\_i等用于存储游戏流程的进度。

以First结构体为例。

hurt()函数用于扣血之后重置参数，播放扣血动画等

begin()用于展示游戏开始之前的对话

isBack用于记录玩家是否返回起点

bl用于记录地刺的升降

block[][]数组是该结构的核心参数，将全部游戏屏幕分为32\*16的格子，用于记录地形

track3及track3\_y和4用于记录一些陷阱的状态，

IMAGE存储展示的图片

文本

描述已自动生成

还有其他一些公用的函数以及参数当作全局变量

gameOver()与gameClear()函数用于展现游戏通关或游戏失败的场景

文本

描述已自动生成conversation()函数、dconver()以及converpause()函数用于展现对话框，其中conversation()函数展现对话内容只有单行的对话框而dconver()函数展现有两行的对话，converpause()函数用于玩家控制对话。

下面wchar\_t类型的参数基本上是用于outtextxy()函数输出文字

player\_x,player\_y表示玩家的位置，jumpup,jumpdown和isjump表示玩家跳跃的状态

通过控制playermove参数来改变IMAGE player1 2展现的内容以实现动画移动的效果

bkleft参数用于背景较长的时候控制背景实际展现的左边距离

star\_i用与控制star1 2的内容来实现星星的旋转动画效果

monster\_i起类似的作用

monster\_life用于表示怪物是否存活

isAnswer用于对话函数，用于表示对话框是否有回答选项，而player\_word1 2用于存储玩家要回答的话，choose用于表示玩家选择的选项。

start end与duration参数用于计时来控制机关还有一些剧情。

### 卡通人物 描述已自动生成2.2基本的对象和主要逻辑

1.主角

通过a d来控制左右 空格控制跳跃不能二段跳，不过如果在空中没有按过跳跃键可以跳

2.蘑菇头

 一般不会有攻击性，只是为了复现一下然后在丰富一下教学关卡的内容。不过玩家在对话完并且获得星星之后与其碰撞会扣血。

3.马里奥

剧情人物

4.地形方块

 人物踩上去能站立，能跳跃

5.星星

 会不断转动，人物碰撞星星会有奖励

6.地刺与陷阱

地刺刷新在地上，每四秒会出来或者回去，设置了3帧的地刺起落动画来控制地刺的起落

陷阱在角色经过特定地区会触发，从天上落下

地刺与陷阱与人物发生碰撞时都会造成伤害，不过地刺在交换起落的几毫秒中不会造成伤害

7.爱心

在左上角呈现，用于表示玩家剩余的血量

### 2.3主要的数据结构和用户界面的定义

主要的数据结构类型为结构体，内部存储各关卡需要的内容

主页面的界面有封面、标题与start按钮三部分组成，按动中间的start按钮就会进入教学关卡



电脑萤幕画面

中度可信度描述已自动生成 教学关卡如图所示，地形较为简单，所以没有用特定的参数来存储，左上角用于存储玩家的血量信息。最左边有一颗星星，游戏偏右侧有一个蘑菇头。

玩家在经过蘑菇头时会触发对话

形状

中度可信度描述已自动生成电脑萤幕画面

中度可信度描述已自动生成图形用户界面

描述已自动生成 踩死蘑菇头后来到最后边可以进入下一关。

关卡里的画面与教学关卡不同的主要是增添了地形以及一些地刺。玩家不能穿过地形方块，并且碰到地刺与陷阱会扣血。

获胜条件是避开陷阱拿到星星之后回到左上角马里奥的位置



失败页面如右图所示

由两个按钮、背景以及标题组成。玩家的鼠标移到按钮上时字会放大。按再加五条命可以满血复活继续游戏，而按退出游戏则会直接退出。



获胜界面同理，与失败界面相比少了加五条命按钮。

### 2.4核心算法的代码实现

### 2.5源码

#include <stdio.h>

#include <stdlib.h>

#include <Windows.h>

#include <time.h>

#include <conio.h>

#include <graphics.h>

#include <math.h>

// 引用 Windows Multimedia API

#pragma comment(lib,"Winmm.lib")

void init();//初始化

void gameOver();//游戏结束

void gameClear();//游戏通关

void conversation(wchar\_t character[], wchar\_t text[],int& isA);//对话

void dconver(wchar\_t character[], wchar\_t text1[], wchar\_t text2[],int& isA);//两行对话

void converpause(int& isA);

//通用的参数

//作者名字

wchar\_t author\_name[] = \_T("南 飞");

//游戏信息

const int width = 1024;

const int height = 512;

//玩家信息

IMAGE player1;

IMAGE player2;

int player\_x, player\_y;

int jumpup, jumpdown;//跳跃状态

int isjump;

int playermove = 0;//移动状态

wchar\_t player\_name[20];

wchar\_t mario\_name[20] = L"马里奥";

//血量显示

IMAGE heart1, heart2, heartempty;

int heart;

//背景

int bkleft;

//奖励

IMAGE star1, star2;

int star\_i = 0;

//怪

IMAGE monster1, monster2;

int monster\_left;

int monster\_i;

int monster\_life;

//对话

IMAGE conversation1, conversation2;

IMAGE answer1, answer2;

int isAnswer;

wchar\_t player\_word1[40], player\_word2[40];

int choose;

int isStart = 0;

//是否拿到星星

int isStar;

//时间

clock\_t start, end;

double duration = 0;

//获胜或失败封面

IMAGE win\_title1;

IMAGE win\_title2;

IMAGE lost\_title1;

IMAGE lost\_title2;

IMAGE relife\_title1;

IMAGE relife\_title2;

IMAGE relife\_small1;

IMAGE relife\_small2;

IMAGE exit\_title1;

IMAGE exit\_title2;

IMAGE exit\_small1;

IMAGE exit\_small2;

//封面

typedef struct Surface

{

IMAGE title1;

IMAGE title2;

IMAGE bkground;

IMAGE btn1;

IMAGE btn2;

void set();

void updateWithInput();

void updateWithoutInput();

void show();

}Sur;

Sur s;

//教学

typedef struct Before //进入游戏之前，填写玩家名字等

{

void set();

void updateWithInput();

void updateWithoutInput();

void show();

//流程

void inputname();

void introduce();

void teach();

void getStar();

IMAGE bkground;//背景

IMAGE grass1, grass2;//土块 48\*33

//剧情是否拿到星星

int isfinish;

}Bef;

Bef b = {};

int isBefore = 1;

//关卡

typedef struct First {

void set();

void updateWithInput();

void updateWithoutInput();

void show();

//扣血

void hurt();

//剧情

void begin();

int isBack;

int bl = 0;//刺

int block[32][16] = { 0 };//0空 1砖 2刺1 3刺2

//陷阱

int track3;

int track3\_y;

int track4;

int track4\_y;

IMAGE bkground;

IMAGE bride;

IMAGE thorn1;

IMAGE thorn2;

IMAGE thorn\_down1;

IMAGE thorn\_down2;

IMAGE prison1;

IMAGE prison2;

IMAGE mario1;

IMAGE mario2;

}Fir;

Fir f;

int main()

{

init();

//封面surface

s.set();

while (!isStart)

{

s.show();

s.updateWithInput();

s.updateWithoutInput();

}

//教学关卡begin

b.set();

b.show();

b.inputname();

b.introduce();

b.teach();

//正式关卡First

f.set();

f.begin();

while (heart && !isStar)

{

f.show();

f.updateWithInput();

f.updateWithoutInput();

if (!heart)

{

gameOver();

}

}

while (!f.isBack && heart)

{

f.show();

f.updateWithInput();

f.updateWithoutInput();

if (player\_x <= 80 && player\_y < 96)

{

f.isBack = 1;

}

if (!heart)

{

gameOver();

}

}

if (heart)

{

wchar\_t str[40];

swprintf(player\_word1, 40, L"下次可小心点吧");

swprintf(player\_word2, 40, L"建议回去之后给我升官");

isAnswer = 1;

f.show();

swprintf(str, 40, L"我的好后辈终于等到你来救我了，我们快一起出去吧");

conversation(mario\_name, str, isAnswer);

converpause(isAnswer);

isAnswer = 0;

swprintf(str, 40, L"哈哈，下次一定");

conversation(mario\_name, str, isAnswer);

converpause(isAnswer);

gameClear();

}

return 0;

}

//初始化

void init()

{

srand((unsigned)time(NULL));

//绘制窗口

initgraph(width, height);

BeginBatchDraw();

//玩家设定

loadimage(&player1, \_T("image/playermove1.jpg"));

loadimage(&player2, \_T("image/playermove2.jpg"));

player\_x = 50;

player\_y = height / 2 - 55;

jumpup = jumpdown = 0;

heart = 5;

//对话

loadimage(&conversation1, \_T("image/conversation1.jpg"));

loadimage(&conversation2, \_T("image/conversation2.jpg"));

isAnswer = 0;

choose = 0;

loadimage(&answer1, \_T("image/conversation3.jpg"));

loadimage(&answer2, \_T("image/conversation4.jpg"));

//星星

loadimage(&star1, \_T("image/star1.jpg"));

loadimage(&star2, \_T("image/star2.jpg"));

star\_i = 0;

//血量

loadimage(&heart1, \_T("image/heart1.jpg"));

loadimage(&heart2, \_T("image/heart2.jpg"));

loadimage(&heartempty, \_T("image/heartempty.jpg"));

start = clock();

//win lost窗口

loadimage(&win\_title1, L"image/win\_title1.jpg");

loadimage(&win\_title2, L"image/win\_title2.jpg");

loadimage(&lost\_title1, L"image/lost\_title1.jpg");

loadimage(&lost\_title2, L"image/lost\_title2.jpg");

loadimage(&relife\_title1, L"image/relife\_title1.jpg");

loadimage(&relife\_title2, L"image/relife\_title2.jpg");

loadimage(&relife\_small1, L"image/relife\_small1.jpg");

loadimage(&relife\_small2, L"image/relife\_small2.jpg");

loadimage(&exit\_title1, L"image/exit\_title1.jpg");

loadimage(&exit\_title2, L"image/exit\_title2.jpg");

loadimage(&exit\_small1, L"image/exit\_small1.jpg");

loadimage(&exit\_small2, L"image/exit\_small2.jpg");

}

//结束

void gameOver()

{

mciSendString(L"close bkmusic", NULL, 0, NULL);//关闭背景音乐

int isrelife = 0;

int isexit = 0;

while (1)

{

cleardevice();

putimage(0, 0, width, height, &s.bkground, 0, 0);

putimage(6 \* 32 + 16, 2 \* 32, &lost\_title2, NOTSRCERASE);

putimage(6 \* 32 + 16, 2 \* 32, &lost\_title1, SRCINVERT);

MOUSEMSG m;

m = GetMouseMsg();

if (m.x >= 12 \* 32 && m.x <= 20 \* 32 && m.y <= 10 \* 32 && m.y >= 9 \* 32)

{

putimage(6 \* 32 + 16, 8 \* 32, &relife\_title2, NOTSRCERASE);

putimage(6 \* 32 + 16, 8 \* 32, &relife\_title1, SRCINVERT);

isrelife = 1;

}

else

{

putimage(6 \* 32 + 16, 8 \* 32, &relife\_small2, NOTSRCERASE);

putimage(6 \* 32 + 16, 8 \* 32, &relife\_small1, SRCINVERT);

isrelife = 0;

}

if (m.x >= 12 \* 32 && m.x <= 20 \* 32 && m.y <= 13 \* 32 && m.y >= 12 \* 32)

{

putimage(6 \* 32 + 16, 11 \* 32, &exit\_title2, NOTSRCERASE);

putimage(6 \* 32 + 16, 11 \* 32, &exit\_title1, SRCINVERT);

isexit = 1;

}

else

{

putimage(6 \* 32 + 16, 11 \* 32, &exit\_small2, NOTSRCERASE);

putimage(6 \* 32 + 16, 11 \* 32, &exit\_small1, SRCINVERT);

isexit = 0;

}

FlushBatchDraw();

if (GetAsyncKeyState(VK\_LBUTTON) & 0x8000)

{

if (isrelife)

{

heart = 5;

//背景音乐

mciSendString(L"open bkmusic.mp3 alias bkmusic", NULL, 0, NULL);//打开背景音乐

mciSendString(L"play bkmusic repeat", NULL, 0, NULL); // 循环播放

break;

}

else if (isexit)

{

EndBatchDraw();

closegraph();

exit(0);

}

}

}

}

//通关

void gameClear()

{

int isexit = 0;

while (1)

{

cleardevice();

putimage(0, 0, width, height, &s.bkground, 0, 0);

putimage(6 \* 32 + 16, 2 \* 32, &win\_title2, NOTSRCERASE);

putimage(6 \* 32 + 16, 2 \* 32, &win\_title1, SRCINVERT);

MOUSEMSG m;

m = GetMouseMsg();

if (m.x >= 12 \* 32 && m.x <= 20 \* 32 && m.y <= 10 \* 32 && m.y >= 9 \* 32)

{

putimage(6 \* 32 + 16, 8 \* 32, &exit\_title2, NOTSRCERASE);

putimage(6 \* 32 + 16, 8 \* 32, &exit\_title1, SRCINVERT);

isexit = 1;

}

else

{

putimage(6 \* 32 + 16, 8 \* 32, &exit\_small2, NOTSRCERASE);

putimage(6 \* 32 + 16, 8 \* 32, &exit\_small1, SRCINVERT);

isexit = 0;

}

FlushBatchDraw();

if (GetAsyncKeyState(VK\_LBUTTON) & 0x8000)

{

if (isexit)

{

EndBatchDraw();

closegraph();

exit(0);

}

}

}

}

//对话

void conversation(wchar\_t character[], wchar\_t text[], int& isA)

{

setbkmode(TRANSPARENT);

settextcolor(BLACK);

settextstyle(0, 0, \_T("宋体"));

if(isA)

{

putimage(width / 2 - 325, height - 270, &answer2, NOTSRCERASE);

putimage(width / 2 - 325, height - 270, &answer1, SRCINVERT);

outtextxy(width / 2 + 100, height - 245, player\_word1);

outtextxy(width / 2 + 100, height - 200, player\_word2);

}

else

{

putimage(width / 2 - 325, height - 200, &conversation2, NOTSRCERASE);

putimage(width / 2 - 325, height - 200, &conversation1, SRCINVERT);

}

outtextxy(width / 2 - 325 + 20, height - 200 + 10, character);

outtextxy(width / 2 - 325 + 20, height - 200 + 65, text);

FlushBatchDraw();

Sleep(500);

}

void dconver(wchar\_t character[], wchar\_t text1[], wchar\_t text2[], int& isA)

{

setbkmode(TRANSPARENT);

settextcolor(BLACK);

settextstyle(0, 0, \_T("宋体"));

if (isA)

{

putimage(width / 2 - 325, height - 300, &answer2, NOTSRCERASE);

putimage(width / 2 - 325, height - 300, &answer1, SRCINVERT);

outtextxy(width / 2 + 100, height - 245, player\_word1);

outtextxy(width / 2 + 100, height - 200, player\_word2);

}

else

{

putimage(width / 2 - 325, height - 200, &conversation2, NOTSRCERASE);

putimage(width / 2 - 325, height - 200, &conversation1, SRCINVERT);

}

outtextxy(width / 2 - 325 + 20, height - 200 + 10, character);

outtextxy(width / 2 - 325 + 20, height - 200 + 65, text1);

outtextxy(width / 2 - 325 + 20, height - 200 + 90, text2);

FlushBatchDraw();

Sleep(500);

}

void converpause(int &isA)

{

if (isA)

{

isA = 0;

while (1)

{

if (GetAsyncKeyState(VK\_UP) & 0x8000)

{

choose = 1;

break;

}

else if (GetAsyncKeyState(VK\_DOWN) & 0x8000)

{

choose = 2;

break;

}

}

}

else

{

while (1)

{

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

{

Sleep(500);

break;

}

}

}

}

//Surface的一些函数

void Sur::set()

{

//背景音乐

mciSendString(L"open bkmusic.mp3 alias bkmusic", NULL, 0, NULL);//打开背景音乐

mciSendString(L"play bkmusic repeat", NULL, 0, NULL); // 循环播放

//背景

loadimage(&bkground, \_T("image/Sur\_bk.jpg"));

bkleft = 0;

//标题

loadimage(&title1, \_T("image/Sur\_title1.jpg"));

loadimage(&title2, \_T("image/Sur\_title2.jpg"));

//按钮

loadimage(&btn1, \_T("image/Sur\_startbtn1.jpg"));

loadimage(&btn2, \_T("image/Sur\_startbtn2.jpg"));

}

void Sur::updateWithInput()

{

//鼠标

MOUSEMSG m = {};

m = GetMouseMsg();

if ((GetAsyncKeyState(VK\_LBUTTON) & 0x8000) && m.x >= width / 2 - 128 && m.x <= width / 2 + 128 && m.y >= height / 2 && m.y <= height / 2 + 100)

{

isStart = 1;

}

}

void Sur::updateWithoutInput()

{}

void Sur::show()

{

//背景

putimage(0, 0, width, height, &bkground, bkleft, 0);

//标题

putimage(50, 0, &title2, NOTSRCERASE);

putimage(50, 0, &title1, SRCINVERT);

//开始按钮

putimage(width / 2 - 128, height / 2, &btn2, NOTSRCERASE);

putimage(width / 2 - 128, height / 2, &btn1, SRCINVERT);

//一些说明

FlushBatchDraw();

Sleep(10);

}

//Before的一些函数

void Bef::set()

{

//背景

loadimage(&bkground, \_T("image/Bef\_bk.jpg"), width + 200, height);

loadimage(&grass1, \_T("image/Bef\_grass1.jpg"));

loadimage(&grass2, \_T("image/Bef\_grass2.jpg"));

bkleft = 200;

//怪物

loadimage(&monster1, \_T("image/monster1\_1.jpg"));

loadimage(&monster2, \_T("image/monster1\_2.jpg"));

monster\_i = 0;

monster\_left = width / 3 \* 2;

monster\_life = 3;

//剧情

isStar = 0;

isfinish = 0;

star\_i = 0;

}

void Bef::show()

{

cleardevice();

//背景覆盖

putimage(0, 0, width, height, &bkground, bkleft, 0);

//地面

for (int i = 0; i <= width / 40; i++)

{

putimage(i \* 40, height / 2, 40, 33, &grass2, 4, 0, NOTSRCERASE);

putimage(i \* 40, height / 2, 40, 33, &grass1, 4, 0, SRCINVERT);

}

for (int i = 0; i < height / 2 / 18; i++)

{

for (int j = 0; j <= width / 40; j++)

{

putimage(j \* 40, height / 2 + 33 + i \* 18, 40, 18, &grass2, 4, 15, NOTSRCERASE);

putimage(j \* 40, height / 2 + 33 + i \* 18, 40, 18, &grass1, 4, 15, SRCINVERT);

}

}

//显示小怪

putimage(monster\_left - bkleft + 200, height / 2 - 25, 30, 30, &monster2, 91 - 30 \* monster\_i, 0, NOTSRCERASE);

putimage(monster\_left - bkleft + 200, height / 2 - 25, 30, 30, &monster1, 91 - 30 \* monster\_i, 0, SRCINVERT);

//显示玩家

putimage(player\_x, player\_y, 32, 58, &player2, 160 + playermove \* 32, 0, NOTSRCERASE);

putimage(player\_x, player\_y, 32, 58, &player1, 160 + playermove \* 32, 0, SRCINVERT);

//如果还没拿到星星

if (!isStar)

{

putimage(-bkleft, height / 2 - 25, 25, 24, &star2, star\_i \* 25, 0, NOTSRCERASE);

putimage(-bkleft, height / 2 - 25, 25, 24, &star1, star\_i \* 25, 0, SRCINVERT);

}

//显示血量

for (int i = 0; i < 5; i++)

{

putimage(i \* 27, 0, &heart2, NOTSRCERASE);

if (i < heart)

{

putimage(i \* 27, 0, &heart1, SRCINVERT);

}

else

{

putimage(i \* 27, 0, &heartempty, SRCINVERT);

}

}

Sleep(70);

FlushBatchDraw();

}

void Bef::updateWithInput()

{

//跳跃

if (jumpdown > 0)

{

if (jumpup > 0)

{

player\_y -= 15;

jumpup--;

}

else if (jumpdown > 0)

{

player\_y += 15;

jumpdown--;

}

}

else if (GetAsyncKeyState(VK\_SPACE) & 0x8000)

{

//跳动音效

mciSendString(L"close jpmusic", NULL, 0, NULL); // 先把前面一次的音乐关闭

mciSendString(L"open jump.mp3 alias jpmusic", NULL, 0, NULL); // 打开跳动音乐

mciSendString(L"play jpmusic", NULL, 0, NULL); // 仅播放一次

if (playermove >= 0)

{

playermove = 4;

}

else

{

playermove = -5;

}

jumpup = 4;

jumpdown = 5;

player\_y -= 15;

}

else if (!jumpdown)

{

if(playermove==4)

{

playermove = 0;

}

else if (playermove == -5)

{

playermove = -1;

}

}

//前进后退

if (GetAsyncKeyState(65) & 0x8000)

{

if (player\_x <= 100 && bkleft > 0)

{

bkleft -= 10;

}

else if (player\_x - 10 >= 0)

{

player\_x -= 10;

}

if (playermove==4||playermove==-5)

{

playermove = -5;

}

else if (playermove == -4 || playermove >= 0)

{

playermove = -1;

}

else

{

playermove--;

}

}

else if (GetAsyncKeyState(68) & 0x8000)

{

if (player\_x + 32 + 200 >= width && bkleft < 200)

{

bkleft += 10;

}

else if (player\_x + 32 + 10 <= width)

{

player\_x += 10;

}

if (playermove == 4 || playermove == -5)

{

playermove = 4;

}

else if (playermove == 3 || playermove < 0)

{

playermove = 0;

}

else

{

playermove++;

}

}

else if (playermove != 4 && playermove != -5)//走路姿势复位

{

if (playermove >= 0)

{

playermove = 0;

}

else

{

playermove = -1;

}

}

}

void Bef::updateWithoutInput()

{

if (isStar == 0 && player\_x <= 25 && bkleft == 0 && player\_y <= height / 2 + 55)

{

getStar();

}

else if (!isStar)

{

if (star\_i == 7)

{

star\_i = 0;

}

else

{

star\_i++;

}

}

if (isStar && isfinish)

{

//踩怪

if (monster\_i != 3 && player\_x + 32 >= monster\_left - bkleft + 200 && player\_x <= monster\_left - bkleft + 200 + 30 && jumpdown == 1)

{

jumpup += 4;

jumpdown += 5;

player\_y -= 15;

monster\_i = 3;

}

//碰怪扣血

else if (monster\_i != 3 && monster\_left - bkleft + 200 <= player\_x + 30 && monster\_left - bkleft + 200 + 30 >= player\_x && jumpdown < 2)

{

//背景音乐

mciSendString(L"close bkmusic", NULL, 0, NULL);//关闭背景音乐

//扣血

mciSendString(L"open dead.mp3 alias deadmusic", NULL, 0, NULL); // 打开扣血

mciSendString(L"play deadmusic", NULL, 0, NULL); // 仅播放一次

show();

Sleep(1000);

int i = 3;

while (player\_y <= height)

{

if (i > 0)

{

player\_y -= 20;

i--;

}

else

{

player\_y += 50;

}

show();

Sleep(100);

}

heart--;

player\_x = 50;

player\_y = height / 2 - 55;

jumpup = jumpdown = 0;

if(heart)

{

mciSendString(L"open bkmusic.mp3 alias bkmusic", NULL, 0, NULL);//打开背景音乐

mciSendString(L"play bkmusic repeat", NULL, 0, NULL); // 循环播放

}

}

}

}

void Bef::inputname()

{

//字符输出

wchar\_t str[] = \_T("欢迎来到游戏，我是本游戏的作者，首先能否告诉我你的名字");

wchar\_t str2[] = \_T("（按空格继续对话，如果有选项的按上下键选择）");

dconver(author\_name, str, str2, isAnswer);

converpause(isAnswer);

while (!InputBox(player\_name, 20, L"请输入您的游戏名"));

Sleep(300);

}

void Bef::introduce()

{

wchar\_t str[40], str2[40];

swprintf\_s(str, 40, L"感谢您的配合，%s", player\_name);

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

swprintf\_s(str, 40, L"接下来由我，为你介绍游戏背景");

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"你叫%s，是一个平平无奇的下水道工人。",player\_name);

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"一天，你看到恶龙库巴从你们的公司离开，而你的前辈马里奥好像被");

swprintf(str2, 40, L"带走了");

show();

dconver(author\_name, str, str2, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"为了救回你敬爱的马里奥前辈，你打算立马冲进库巴的城堡去营救。");

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"是不是十分简单易懂。");

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

}

void Bef::teach()

{

wchar\_t str[40], str2[40];

swprintf(str, 40, L"好了不要偷偷吐槽这么简单的设定了，让我教教你最基本的操作。");

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"按a向左走，按d向右走，按空格键跳跃");

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"想必你平常也不怎么锻炼，没病先走两步");

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

while (player\_x < width / 2)

{

updateWithInput();

updateWithoutInput();

show();

}

swprintf(str, 40, L"看来你已经完全适应人类的活动方式了呢/doge。看到了吗，前面有一");

swprintf(str2, 40, L"个小怪。想办法过去吧。");

show();

dconver(author\_name, str, str2, isAnswer);

converpause(isAnswer);

while (player\_x <= monster\_left - bkleft + 200 + 20)

{

updateWithInput();

updateWithoutInput();

show();

}

wchar\_t monster\_name[] = L"蘑菇头";

swprintf(str, 40, L"喂！");

while (jumpdown > 0)

{

if (jumpup > 0)

{

player\_y -= 15;

jumpup--;

}

else if(jumpdown>=0)

{

player\_y += 15;

jumpdown--;

}

show();

}

playermove = 0;

show();

conversation(monster\_name, str, isAnswer);

converpause(isAnswer);

while (monster\_left - bkleft + 200 <= player\_x + 35)

{

monster\_i++;

if (monster\_i == 2)

{

monster\_i = 0;

}

monster\_left += 10;

show();

}

swprintf(str, 40, L"没看到本大爷在这里看着吗？！你是个什么品种的东西。");

show();

isAnswer = 1;

swprintf(player\_word1, 40, L"我是新来的保安队长");

swprintf(player\_word2, 40, L"我是库巴的朋友");

conversation(monster\_name, str, isAnswer);

converpause(isAnswer);

if (choose == 1)

{

swprintf(str, 40, L"你是故意找茬是不是，我就是保安队长，趁本大爷心情好，哪凉快哪");

swprintf(str2, 40, L"呆着去");

}

else if (choose == 2)

{

swprintf(str, 40, L"我靠，怎么搞诈骗都搞到我头上来了，我还以为你是瞎子拉琴呢，和");

swprintf(str2, 40, L"我瞎扯老半天。库巴大人没有你这么搓的朋友。哪凉快哪呆着去");

}

show();

dconver(monster\_name, str, str2, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"事先说好，你可别想着踩扁我，我和你可不在一个图层/笑");

show();

conversation(monster\_name, str, isAnswer);

converpause(isAnswer);

swprintf(str, 40, L"（看来得想想办法过去了）");

show();

conversation(player\_name, str, isAnswer);

converpause(isAnswer);

isfinish = 1;

if(!isStar)

{

while (player\_x >= 100 && bkleft >= 0)//回到左边再给一个教程

{

updateWithInput();

if (player\_x >= monster\_left - bkleft + 200)//不能超过怪物

{

player\_x -= 10;

}

updateWithoutInput();

show();

}

swprintf(str, 40, L"哦，刚才忘了告诉你了，左边这个转圈的星星里面可能会有好东西哦");

show();

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

while (!isStar)

{

updateWithInput();

if (player\_x >= monster\_left - bkleft + 200)//不能超过怪物

{

player\_x -= 10;

}

updateWithoutInput();

show();

}

}

while (monster\_i != 3 && heart)

{

updateWithInput();

updateWithoutInput();

show();

if (!heart)

{

gameOver();

}

}

while (player\_x + 32 + 10 <= width)

{

updateWithInput();

updateWithoutInput();

show();

}

swprintf(str, 40, L"恭喜你成功通过第一关，接下来就是城堡里面的冒险了");

conversation(author\_name, str, isAnswer);

converpause(isAnswer);

}

void Bef::getStar()

{

wchar\_t str[40];

isStar = 1;

swprintf(str, 40, L"进入了怪物的图层");

show();

conversation(player\_name, str, isAnswer);

converpause(isAnswer);

}

//First的一些函数

void Fir::set()

{

start = clock();

int i;

loadimage(&bkground, \_T("./image/First\_bkground.jpg"), width, height);

loadimage(&bride, \_T("./image/bride.jpg"));

loadimage(&thorn1, \_T("./image/thorn1.jpg"));

loadimage(&thorn2, \_T("./image/thorn2.jpg"));

loadimage(&thorn\_down1, \_T("./image/thorn\_down1.jpg"));

loadimage(&thorn\_down2, \_T("./image/thorn\_down2.jpg"));

loadimage(&mario1, \_T("./image/mario1.jpg"));

loadimage(&mario2, \_T("./image/mario2.jpg"));

loadimage(&prison1, \_T("./image/prison1.jpg"));

loadimage(&prison2, \_T("./image/prison2.jpg"));

isStar = 0;

isBack = 0;

//玩家

player\_x = 0;

player\_y = 32 \* 6 - 58;

isjump = 0;

//陷阱

track3 = 0;

track3\_y = -32;

track4 = 0;

track4\_y = -32;

//block设置

for (i = 0; i < 9; i++)

{

block[i][3] = 1;

}

for (i = 0; i < 5; i++)

{

block[i][6] = 1;

}

for (i = 2; i < 9; i++)

{

block[i][10] = 1;

}

for (i = 4; i < 7; i++)

{

block[i][9] = 2;

}

for (i = 4; i < 10; i++)

{

block[7][i] = 1;

block[8][i] = 1;

}

for (i = 0; i < 14; i++)

{

block[i][14] = 1;

block[i][15] = 1;

}

block[9][7] = 3;

block[9][8] = 1;

block[11][4] = 2;

block[11][5] = 1;

block[11][10] = 2;

block[11][11] = 1;

for (i = 3; i < 14; i++)

{

block[12][i] = 1;

block[13][i] = 1;

}

block[18][3] = 1;

block[18][2] = 2;

for (i = 23; i < 28; i++)

{

block[i][3] = 1;

}

for (i = 14; i < 25; i += 4)

{

block[i][6] = 2;

block[i + 1][6] = 2;

block[i + 2][6] = 3;

block[i + 3][6] = 3;

}

block[30][4] = 3;

block[30][5] = 1;

block[31][8] = 1;

for (i = 14; i < 28; i++)

{

block[i][7] = 1;

}

for (i = 19; i < 25; i += 4)

{

block[i][10] = 3;

block[i + 1][10] = 3;

block[i + 2][10] = 2;

block[i + 3][10] = 2;

}

block[24][10] = 0;

block[23][10] = 0;

block[27][10] = 3;

block[28][10] = 3;

for (i = 16; i < 32; i++)

{

block[i][11] = 1;

}

block[14][11] = 3;

block[14][12] = 1;

block[14][15] = 1;

block[15][15] = 1;

for (i = 18; i < 28; i += 3)

{

block[i][15] = 1;

block[i][14] = 2 + i % 2;

}

for (i = 28; i < 32; i++)

{

block[i][15] = 1;

}

}

void Fir::begin() {

wchar\_t str[40];

wchar\_t str2[40];

show();

isAnswer = 1;

swprintf(str, 40, L"嘿，%s！是我，马里奥", player\_name);

swprintf(player\_word1, 40, L"终于找到前辈您了");

swprintf(player\_word2, 40, L"你谁啊喂");

conversation(mario\_name, str, isAnswer);

converpause(isAnswer);

show();

if(choose==1)

{

swprintf(str, 40, L"我太感动了，你居然会来救我 T\_T");

}

else

{

swprintf(str, 40, L"怎么翻脸不认人啊喂");

}

conversation(mario\_name, str, isAnswer);

converpause(isAnswer);

show();

swprintf(str, 40, L"总之，我需要你帮我找到钥匙带回来。这里的地刺好像是4秒变一次");

swprintf(str2, 40, L"要小心");

dconver(mario\_name, str, str2, isAnswer);

converpause(isAnswer);

}

void Fir::show()

{

cleardevice();

//背景覆盖

putimage(0, 0, &bkground);

//如果还没拿到星星

if (!isStar)

{

putimage(width - 25, height - 32 - 24, 25, 24, &star2, star\_i \* 25, 0, NOTSRCERASE);

putimage(width - 25, height - 32 - 24, 25, 24, &star1, star\_i \* 25, 0, SRCINVERT);

}

//显示前辈

if (!isBack)

{

putimage(0, 32, &prison2, NOTSRCERASE);

putimage(0, 32, &prison1, SRCINVERT);

}

else

{

putimage(16, 32, &mario2, NOTSRCERASE);

putimage(16, 32, &mario1, SRCINVERT);

}

//显示血量

for (int i = 0; i < 5; i++)

{

putimage(i \* 27, 0, &heart2, NOTSRCERASE);

if (i < heart)

{

putimage(i \* 27, 0, &heart1, SRCINVERT);

}

else

{

putimage(i \* 27, 0, &heartempty, SRCINVERT);

}

}

//显示地形

for (int i = 0; i < 32; i++)

{

for (int j = 0; j < 16; j++)

{

if (block[i][j] == 1)

{

putimage(i \* 32, j \* 32, &bride);

}

else if (block[i][j] == 2)//地刺1

{

if ((int)(duration \* 10) % 30 == 0 && bl || (int)(duration \* 10) % 30 == 1 && !bl)

{

putimage(i \* 32, j \* 32 + 20, &thorn2, NOTSRCERASE);

putimage(i \* 32, j \* 32 + 20, &thorn1, SRCINVERT);

}

else if ((int)(duration \* 10) % 30 == 1 && bl || (int)(duration \* 10) % 30 == 0 && !bl)

{

putimage(i \* 32, j \* 32 + 10, &thorn2, NOTSRCERASE);

putimage(i \* 32, j \* 32 + 10, &thorn1, SRCINVERT);

}

else if (bl)

{

putimage(i \* 32, j \* 32, &thorn2, NOTSRCERASE);

putimage(i \* 32, j \* 32, &thorn1, SRCINVERT);

}

else

{

putimage(i \* 32, (j + 1) \* 32, &thorn2, NOTSRCERASE);

putimage(i \* 32, (j + 1) \* 32, &thorn1, SRCINVERT);

}

}

else if (block[i][j] == 3)//地刺2

{

if ((int)(duration \* 10) % 30 == 0 && !bl || (int)(duration \* 10) % 30 == 1 && bl)

{

putimage(i \* 32, j \* 32 + 20, &thorn2, NOTSRCERASE);

putimage(i \* 32, j \* 32 + 20, &thorn1, SRCINVERT);

}

else if ((int)(duration \* 10) % 30 == 0 && bl || (int)(duration \* 10) % 30 == 1 && !bl)

{

putimage(i \* 32, j \* 32 + 10, &thorn2, NOTSRCERASE);

putimage(i \* 32, j \* 32 + 10, &thorn1, SRCINVERT);

}

else if (!bl)

{

putimage(i \* 32, j \* 32, &thorn2, NOTSRCERASE);

putimage(i \* 32, j \* 32, &thorn1, SRCINVERT);

}

else

{

putimage(i \* 32, (j + 1) \* 32, &thorn2, NOTSRCERASE);

putimage(i \* 32, (j + 1) \* 32, &thorn1, SRCINVERT);

}

}

}

}

//陷阱3

if (track3 && track3\_y != -100)

{

putimage(10 \* 32, track3\_y, &thorn\_down2, NOTSRCERASE);

putimage(10 \* 32, track3\_y, &thorn\_down1, SRCINVERT);

}

//陷阱4

if (track4 && track4\_y != -100)

{

putimage(15 \* 32, track4\_y, &thorn\_down2, NOTSRCERASE);

putimage(15 \* 32, track4\_y, &thorn\_down1, SRCINVERT);

}

//显示玩家

putimage(player\_x, player\_y, 32, 58, &player2, 160 + playermove \* 32, 0, NOTSRCERASE);

putimage(player\_x, player\_y, 32, 58, &player1, 160 + playermove \* 32, 0, SRCINVERT);

FlushBatchDraw();

Sleep(70);

}

void Fir::updateWithInput()

{

//左走

if (GetAsyncKeyState(65) & 0x8000)

{

if (player\_x % 32 >= 8)

{

player\_x -= 8;

}

else if (player\_x >= 32 && block[(player\_x - 8) / 32][player\_y / 32] != 1 && block[(player\_x - 8) / 32][player\_y / 32 + 1] != 1 && block[(player\_x - 8) / 32][(player\_y + 57) / 32] != 1)

{

player\_x -= 8;

}

if (playermove == 4 || playermove == -5)//跳跃

{

playermove = -5;

}

else if (playermove == -4 || playermove >= 0)

{

playermove = -1;

}

else

{

playermove--;

}

}

//右走

else if (GetAsyncKeyState(68) & 0x8000)

{

if (player\_x % 32 <= 32 - 8)

{

if (player\_x % 32 == 0)

{

if (player\_x / 32 + 1 < 32 && block[player\_x / 32 + 1][player\_y / 32] != 1 && block[player\_x / 32 + 1][player\_y / 32 + 1] != 1 && block[player\_x / 32 + 1][(player\_y + 57) / 32] != 1)

{

player\_x += 8;

}

}

else

{

if (block[player\_x / 32 + 1][player\_y / 32] != 1 && block[player\_x / 32 + 1][(player\_y + 57) / 32] != 1)

{

player\_x += 8;

}

}

}

else if (player\_x <= 32 \* 15 && block[(player\_x + 32 + 8) / 32][player\_y / 32] != 1 && block[(player\_x + 32 + 8) / 32][(player\_y + 57) / 32] != 1)

{

player\_x += 8;

}

if (playermove == 4 || playermove == -5)

{

playermove = 4;

}

else if (playermove == 3 || playermove < 0)

{

playermove = 0;

}

else

{

playermove++;

}

}

else if (playermove != 4 && playermove != -5)//走路姿势复位

{

if (playermove >= 0)

{

playermove = 0;

}

else

{

playermove = -1;

}

}

//跳跃

if (GetAsyncKeyState(VK\_SPACE) & 0x8000 && isjump == 0)

{

//跳动

mciSendString(L"close jpmusic", NULL, 0, NULL); // 先把前面一次的音乐关闭

mciSendString(L"open jump.mp3 alias jpmusic", NULL, 0, NULL); // 打开跳动音乐

mciSendString(L"play jpmusic", NULL, 0, NULL); // 仅播放一次

isjump = 1;

jumpup = 12;

if (playermove >= 0)

{

playermove = 4;

}

else

{

playermove = -5;

}

}

}

void Fir::updateWithoutInput()

{

//陷阱1

if (player\_x >= 5 \* 32 && player\_x <= 9 \* 32 && player\_y >= 10 \* 32 && block[6][14] != 0)

{

for (int i = 6; i < 10; i++)

{

block[i][14] = 0;

block[i][15] = 2;

}

}

else if (block[6][14] == 0 && (player\_x < 5 \* 32 || player\_x>9 \* 32 || player\_y < 10 \* 32))

{

for (int i = 6; i < 10; i++)

{

block[i][14] = 1;

block[i][15] = 1;

}

}

//陷阱2

if (player\_x >= 5 \* 32 && player\_x <= 6 \* 32 && player\_y <= 3 \* 32&& block[5][3] != 0)

{

block[5][3] = 0;

block[6][3] = 0;

block[5][0] = 1;

block[6][0] = 1;

}

else if (!(player\_x >= 5 \* 32 && player\_x <= 6 \* 32 && player\_y <= 3 \* 32) && block[5][3] == 0)

{

block[5][3] = 1;

block[6][3] = 1;

block[5][0] = 0;

block[6][0] = 0;

}

//陷阱3

if (!track3 && player\_x >= 10 \* 32 - 16 && player\_x <= 11 \* 32 && player\_y >= 12 \* 32 && player\_y <= 13 \* 32)

{

track3 = 1;

track3\_y = -32;

}

else if (track3 && track3\_y != -50)

{

if (track3\_y > height)//一次性

{

track3\_y = -100;

}

else if (track3\_y + 32 >= player\_y && track3\_y + 32 <= player\_y + 50 && player\_x < 11 \* 32 && player\_x > 9 \* 32)

{

hurt();

}

track3\_y += 50;//下落

}

//陷阱4

if (!track4 && player\_x >= 14 \* 32-8 && player\_x <= 15 \* 32)

{

track4 = 1;

track4\_y = -32;

}

else if (track4 && track4\_y != -50)

{

if (track4\_y > height)//一次性

{

track4\_y = -100;

}

else if (track4\_y + 32 >= player\_y && track4\_y + 32 <= player\_y + 50 && player\_x < 11 \* 32 && player\_x >= 9 \* 32)

{

hurt();

}

track4\_y += 50;//下落

}

//地刺

end = clock();

duration = (double)(end - start) / CLK\_TCK;

if (duration >= 3)

{

start = clock();

bl = 1 - bl;

}

//星星

if (isStar == 0 && player\_x + 32 > width - 25 && player\_y + 58 >= height - 32 - 25)

{

isStar = 1;

wchar\_t str[] = L"获得了钥匙🔑";

conversation(player\_name, str, isAnswer);

converpause(isAnswer);

}

else if (!isStar)

{

if (star\_i == 7)

{

star\_i = 0;

}

else

{

star\_i++;

}

}

//跳跃及下落

if (isjump && jumpup)//跳跃

{

//if ((player\_y - 8) < 0)//撞天花板

//{

// jumpup = 0;

//}

//else

if (player\_x % 32 == 0 && block[player\_x / 32][(player\_y - 8) / 32] != 1)

{

player\_y -= 8;

jumpup -= 1;

}

else if (block[player\_x / 32][(player\_y - 8) / 32] == 1 || block[player\_x / 32 + 1][(player\_y - 8) / 32])//撞墙

{

jumpup = 0;

}

else

{

player\_y -= 8;

jumpup -= 1;

}

}

else if (player\_x / 32 + 1 <= 32 && player\_y / 32 + 2 < 16)//下落

{

if (player\_x / 32 + 1 != 32 && block[player\_x / 32][(player\_y + 58) / 32] != 1 && block[player\_x / 32 + 1][(player\_y + 58) / 32] != 1)

{

player\_y += 8;

}

else if (player\_x % 32 == 0 && block[player\_x / 32][(player\_y + 58) / 32] != 1)

{

player\_y += 8;

}

else

{

isjump = 0;

if (playermove == 4)

{

playermove = 0;

}

else if (playermove == -5)

{

playermove = -1;

}

}

}

else if ((player\_y + 58) / 32 + 2 >= 16)

{

player\_y += 8;

if (player\_y / 32 >= 16)//掉下去

{

hurt();

}

}

//地刺判定

if (block[player\_x / 32][(player\_y + 57) / 32] >= 2)

{

if (block[player\_x / 32][(player\_y + 57) / 32] == 2 && bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

else if (block[player\_x / 32][(player\_y + 57) / 32] == 3 && !bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

}

else if (block[player\_x / 32][player\_y / 32 + 1] >= 2)

{

if (block[player\_x / 32][player\_y / 32 + 1] == 2 && bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

else if (block[player\_x / 32][player\_y / 32 + 1] == 3 && !bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

}

else if (player\_x % 32 != 0)

{

if (block[player\_x / 32 + 1][player\_y / 32 + 1] >= 2)

{

if (block[player\_x / 32 + 1][player\_y / 32 + 1] == 2 && bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

else if (block[player\_x / 32 + 1][player\_y / 32 + 1] == 3 && !bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

}

else if (block[player\_x / 32 + 1][(player\_y + 57) / 32] >= 2)

{

if (block[player\_x / 32 + 1][(player\_y + 57) / 32] == 2 && bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

else if (block[player\_x / 32][(player\_y + 57) / 32] == 3 && !bl && (int)(duration \* 10) % 30 > 1)

{

hurt();

}

}

}

}

void Fir::hurt()

{

//背景音乐

mciSendString(L"close bkmusic", NULL, 0, NULL);//关闭背景音乐

//扣血

mciSendString(L"close deadmusic", NULL, 0, NULL);//关闭扣血音乐

mciSendString(L"open dead.mp3 alias deadmusic", NULL, 0, NULL); // 打开扣血

mciSendString(L"play deadmusic", NULL, 0, NULL); // 仅播放一次

Sleep(500);

int i = 3;

while (player\_y <= height)

{

if (i > 0)

{

player\_y -= 20;

i--;

}

else

{

player\_y += 30;

}

show();

Sleep(10);

}

heart--;

if (heart)

{

mciSendString(L"open bkmusic.mp3 alias bkmusic", NULL, 0, NULL);//打开背景音乐

mciSendString(L"play bkmusic repeat", NULL, 0, NULL); // 循环播放

}

if (isStar)//复位角色

{

player\_x = 31 \* 32;

player\_y = 14 \* 32 - 58;

}

else

{

player\_x = 0;

player\_y = 32 \* 6 - 58;

}

//复位陷阱

track3\_y = -32;

track3 = 0;

track4\_y = -32;

track4 = 0;

}

## 3）项目测试与总结

### 3.1 bug和debug

开发过程中遇到的一些bug

1. 对话框的鼠标点击问题
   1. 内容：鼠标点击读取次数有误，鼠标点一次直接过了好多对话
   2. 解决方法
      1. 加上Sleep()函数，但是效果不好
      2. 换成键盘敲击加Sleep函数，效果相对前面较好
2. 主角的跳跃与下落问题
   1. 内容：在第一关中，地形只有平地，因此不会由太大的跳跃问题产生，但是第二关出现了不一样的地形
   2. 解决方法
      1. 根据主角的坐标与地形的block[][]数组来判定主角是否踩在空地上并且将跳跃与下落的if判定放在同一个if-else语句里以免发生既跳跃又下落的情况。

### 3.2总结与不足

先说好的，总体而言，本次用C语言来做这个游戏让我感觉还不错，毕竟也确实做出了一个还算有模有样的游戏，尽管过程有些艰辛。能够利用自己所学来完成一项工作对于我来说还是十分有成就感的。在写程序的时候也遇到了不少bug，或有所放弃，或有所改进。在网络上搜索 bug产生的原因，如何解决等都让我有所提升。最后做出来的游戏也还不错，总的还是在及格线之上的。

再说说不足之处。本来只是想借用Mario的素材来做一个综合各项游戏模式（2d枪战 2d格斗 甚至是回合制）的游戏，但是着实能力有限，并且选的专业课有点多，大作业和期末考试两大重压下不断取舍取舍，最后除了教学关卡只完成了一关，而且玩法也比较单一。实在令我感到非常遗憾。