**《C语言程序设计》**

**课程设计报告**

**专 业： 软件工程3+4**

**班 级： 181级**

**姓 名： 石明阳**

**指导教师： 丁玉忠**

**2019年 03月07日-2019年04月26日**

目录

[1 前言 2](#_Toc10969781)

[2 需求分析 2](#_Toc10969782)

[2.1要求 2](#_Toc10969783)

[2.2任务 3](#_Toc10969784)

[2.3运行环境 3](#_Toc10969785)

[2.4开发工具 3](#_Toc10969786)

[3 概要设计与详细设计 3](#_Toc10969787)

[3.1系统流程图 3](#_Toc10969788)

[4 编码与实现 3](#_Toc10969789)

[4.1分析 3](#_Toc10969790)

[4.2具体代码实现 7](#_Toc10969791)

[5 课程设计总结 14](#_Toc10969792)

[参考文献 14](#_Toc10969793)

1 前言

编写一个程序来实现游戏flappy bird。通过不同的函数来分别实现游戏的不同部分。

2 需求分析

2.1要求

（1）用C语言实现程序设计；

（2）利用循环，选择等实现游戏的动作。

（3）画出游戏运行的流程图；

（4）系统的各个功能模块要求用函数的形式实现；

（5）界面友好（良好的人机互交），程序要有注释。

2.2任务

（1）通过不同函数构成游戏

（2）通过键鼠可玩游戏

（3）画出所有模块的流程图；

（4）编写代码；

（5）程序分析与调试。

2.3运行环境

（1）WINDOWS10(v.17134)系统

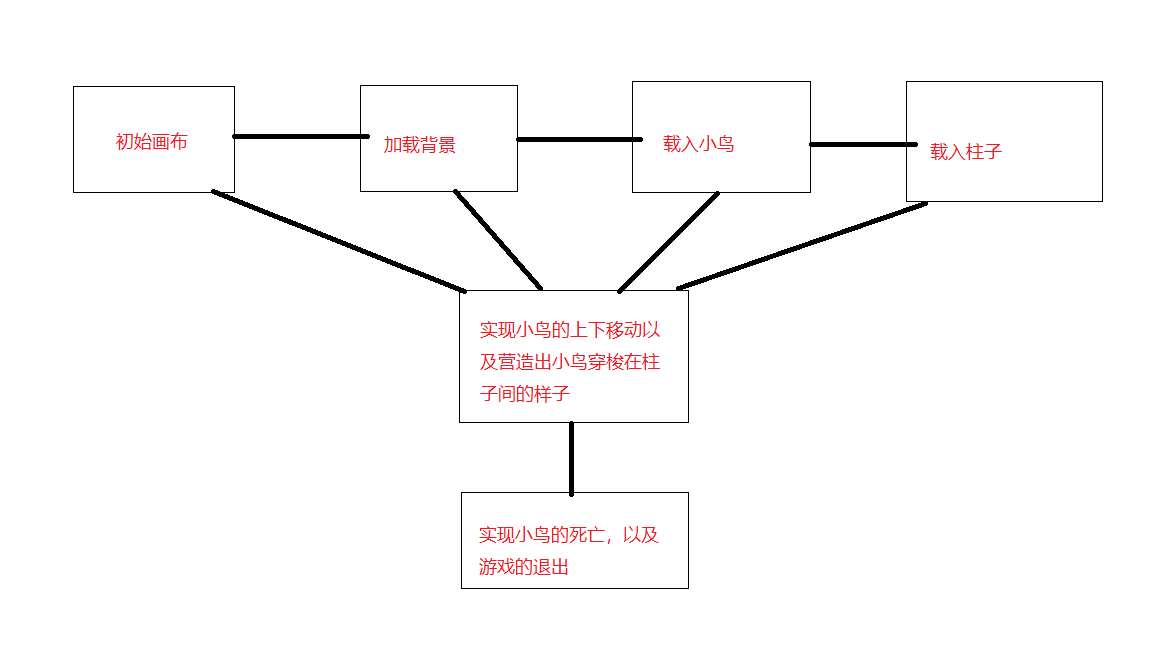
（2）Visual Studio 2017编译环境

2.4开发工具

C语言

3 概要设计与详细设计

3.1系统流程图



4 编码与实现

4.1分析

鸟和柱子的实现本质上就是利用他们左上的坐标变化的基础上使用库函数loadimage（）和putimage（）将素材输出到该坐标上，因此需要定义关于坐标的函数bird x和bird y，通过对坐标函数的定义来实现鸟的行为，柱子同理。

在开始需要定义之后的IMAGE对象即各个图像的名称。

所有全局变量在最前面定义，清晰明了。

1. **实现游戏前的导入：**建立一个初始化的变量startup();，用于导入大部分的素材的图片，注意起名要有规律可循，以免产生混乱。导入图片利用loadimage(&image对象名，”所使用图片的位置”)；

1.2 实现素材的显示 ：建立一个用于将素材打印到屏幕上的函数show();将图片输出到屏幕用putimage(横坐标，纵坐标，&图片变量名);

**2. 背景图片的显示：**将图片放置在窗口的左上角，使putimage(0,0,&img\_bk);即可实现。

**3.显示小鸟的图：仔**细观察，flappy bird 中小鸟虽然看起来是在穿梭柱子，但是小鸟并没有左右移动，只是柱子在向左移动，营造出了小鸟在向右移动的错觉，所以这里只需要做出小鸟的上下移动即可，顺便在屏幕左边中间的位置设置一个初始位置就好这里设置为了（50,200）然后用putimage()将小鸟的图片放上去。这里需要用到一个遮罩层的使用，利用小鸟的遮罩层bird1。bird1与bird2中的像素对应，bird1中的黑色像素将bird2中的白色像素隐藏，其中需要用到NOTSRCERASE与SRCERASE进行遮罩操作（注意图片需要是gif格式），之后柱子即分数的数字也是相同原理。效果如下：



代码如下：

putimage(bird\_x, bird\_y, &img\_bd1, NOTSRCERASE);

putimage(bird\_x, bird\_y, &img\_bd2, SRCERASE);

**4.小鸟的动作：**小鸟的动作分为三种情况。一种是键盘无输入时的掉落，另一种是按下鼠标左键后小鸟的上升以及按下鼠标右键的下落动作。单独做出一个关于小鸟的函数bird()。自由下落通过循环实现，无操作时bird\_x+=3;来进行缓慢下落，使用if（kbhit()）来检测键盘是否有输入。设置一个char类型的变量用于存储来自键盘的输入。键盘输入具体代码如下：

if (\_kbhit())

{

input = \_getch();

if (input == ' '&&bird\_y > 20)

bird\_y -= 60;

}

鼠标需要先前面设置一个MOUSEMSG m; m用于存储来自鼠标的信息。使用循环控制鼠标的输入，再用判断语句判断左右键分辨需要执行什么指令，独有的函数较多，所以直接贴上代码。

while (MouseHit())

{

// 获取一条鼠标消息

m = GetMouseMsg();

switch (m.uMsg)

{

case WM\_LBUTTONDOWN:

{

if (bird\_y > 20)//以防小鸟飞出外

bird\_y -= 60;

mciSendString("closejpmusic", NULL, 0, NULL);

mciSendString("open ..\\source\\Jump.mp3 alias jpmusic", NULL, 0, NULL);

mciSendString("open music", NULL, 0, NULL);

}

break;

case WM\_RBUTTONDOWN:

bird\_y += 60;

break;

}

}

**5.静态柱子的移动；**静态柱子的移动在bar()中实现，通过设置柱子的上坐标bar\_up\_x和下坐标bar\_down\_x的自减来使柱子向左移动，这里注意一下上下柱子的同步移动。

**6..柱子的刷新：**设定柱子的坐标bar\_up\_x,bar\_down\_x范围，当到达一个值时重新定义柱子的坐标，我的柱子的长度是140px，所以设定为了移到x=-140时重新刷新到屏幕最右侧的x=600代码如下：

if (bar\_up\_x == -140 && bar\_down\_x == -140)

{

bar\_up\_x = 600;

bar\_down\_x = 600;

}

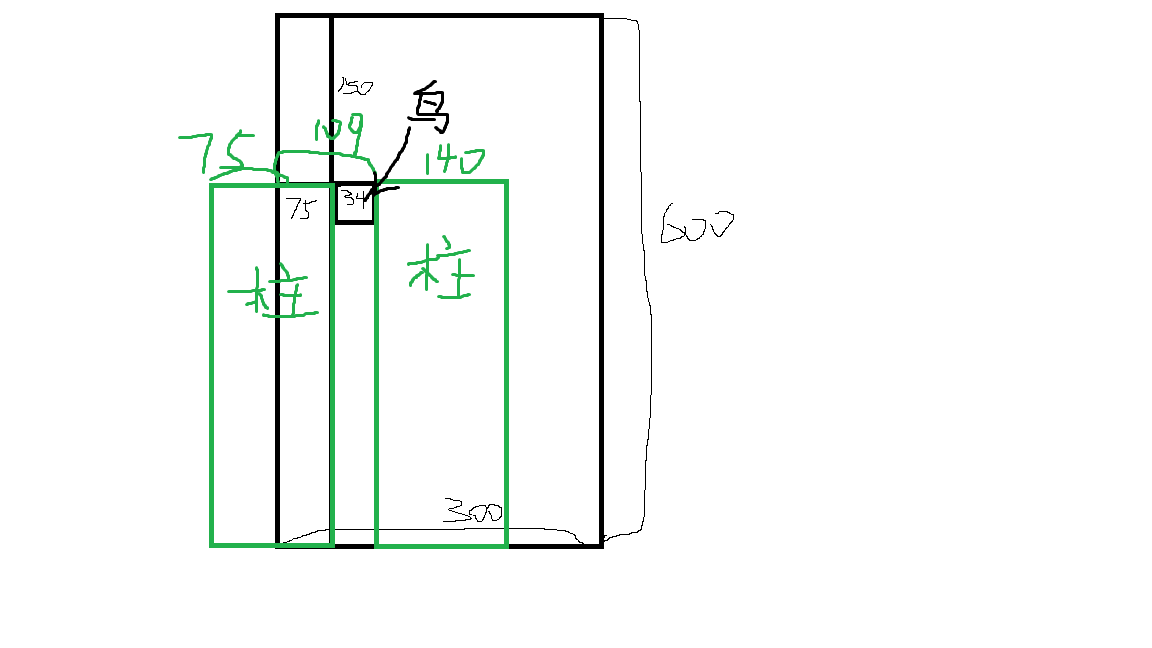
**7.柱子的随机长度:**要让柱子产生随机长度的效果，需要用到rand()，位于stdlib.h头文件中。由于flappy bird中柱子的x坐标是一样的，所以只需要考虑一个珠子的长度即可，另外一个柱子只需要加上相应的值即可。给定随机函数一个范围。这里使用上柱子的坐标进行运算，给出的范围是bar\_up\_y=rand()-310-500;，下柱子在上柱子的基础上加上700，然后再重新输出柱子的贴图，就可以实现柱子的随机刷新了。随机柱子的坐标代码如下：

bar\_up\_y= rand() % -310 - 550;//随机上方柱子出现的坐标

bar\_down\_y = bar\_up\_y+700;//下方柱子紧跟上方柱子

8.多根柱子：为了之后的三关做准备，所以多准备几根，bar2(),bar3(),与第一根柱子一样，这里特别的注意要把控好多根柱子自己的数值以及坐标。

**9实现小鸟的死亡，判断碰撞区间：**需要判定柱子与小鸟坐标的位置，大体思路是先判断横坐标上的柱子与小鸟的位置关系，横坐标判定柱子是否进入小鸟所在区间，当判定进入小鸟所在区间时再进行纵坐标上的柱子与小鸟的区间判断，判断完成后，执行小鸟死亡的界面。

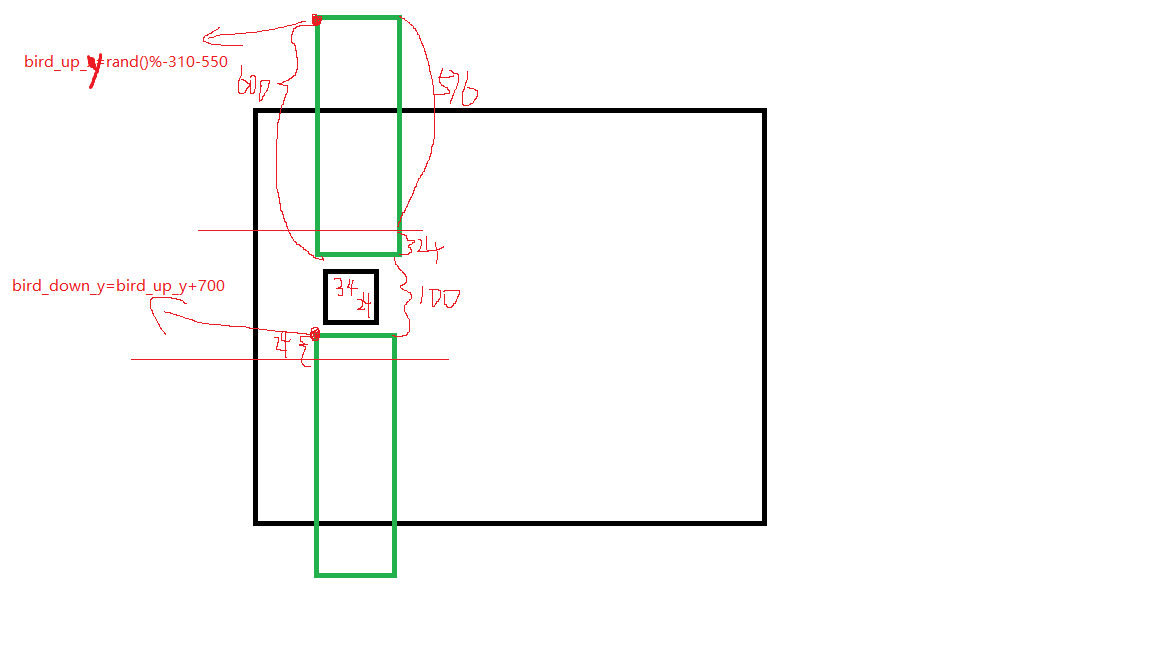
横坐标的位置图如下：

我的小鸟大小为34\*24，坐标为（75,150），柱子规格为140\*600，首先是横坐标的判断，柱子的横坐标从75（鸟横坐标）+34（鸟的长度）移动到75（鸟的横坐标）-140（柱子横坐标），即-75—109的横坐标区间。

横坐标处判断代码如下：

if ((bar\_up\_x < 109&&bar\_down\_x<109)&&(bar\_up\_x > -75 && bar\_down\_x > -75))

纵坐标位置图如下：



纵坐标上由于柱子的横坐标一致，所以柱子只需要考虑上柱子坐标即可。要控制好两根柱子的存活区间，上柱子与下柱子之前我设置的值时是差了100px的距离，再结合柱子的高度600px所以上柱子判断是上柱子横坐标加上柱子的高度和一个容错的值，可以使用小鸟的高度，下柱子的判定只需在上柱子的基础上加上100 即可。

纵坐标处判断代码如下：

if ((bird\_y < (bar\_up\_y + 576) && bird\_y < (bar\_up\_y + 600))||(bird\_y > (bar\_up\_y + 676) && bird\_y > (bar\_up\_y + 700)))

再加上跌落底部的死亡，y坐标上不让其大于566.。其余柱子的判定同理，注意各个的坐标不要写混。

10.分数的计算 我的是在碰撞判断的函数那里写的，如果柱子们的横坐标到了小鸟横坐标附近，分数加一，这里需要注意，如果速度有不同的值的话，这个加分判定的数值要为所有速度大小的公倍数，否则柱子坐标无法在相应的速度下到达加分判断的那个数值。代码如下：

if ((bar\_up\_x == 80)|| (bar\_up\_x2 == 80)|| (bar\_up\_x3 == 80))//注意，判定的值需要同时是不同速度的公倍数，否则会错过

{

score++;

}

11.分数的显示：我的最大分数是两位数，所以设置了两个变量right，left来对个位数和十位数进行显示，总分数在判定碰撞的函数内进行计算，如果通过分数加1，个位数对总分数进行取余10，十位数先除10，后取余10。然后在屏幕指定位置显示。

12.整合 建立主函数，设置所有的初始数值，设置一个死循环在里面放入之前写的函数让其一直循环。

4.2具体代码实现

#include <graphics.h>

#include <conio.h>

#include <stdio.h>

#include <stdlib.h>

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

IMAGE img\_bk;

IMAGE img\_bd1, img\_bd2;//鸟的图片

IMAGE img\_bar\_up1, img\_bar\_up2, img\_bar\_up2\_1, img\_bar\_up2\_2, img\_bar\_up3\_1, img\_bar\_up3\_2;//上方柱子的图片

IMAGE img\_bar\_down1, img\_bar\_down2, img\_bar\_down2\_1, img\_bar\_down2\_2, img\_bar\_down3\_1, img\_bar\_down3\_2;//下方柱子的图片

IMAGE num1[10], num2[10];//计分数字图片

int bird\_x, bird\_y;//鸟坐标

int bar\_up\_x, bar\_up\_y, bar\_up\_x2, bar\_up\_y2,bar\_up\_x3, bar\_up\_y3;//上方柱子的坐标

int bar\_down\_x, bar\_down\_y, bar\_down\_x2, bar\_down\_y2, bar\_down\_x3, bar\_down\_y3;//下方柱子的坐标

int i = 0, left = 0, right = 0, n = 5;

int score = 0;//分数&分数左右

int a,b;//界面大小

int c, d;//死亡图标位置

int e1,e2, f=50;//分数位置

MOUSEMSG m;

void startup()//初始化

{

initgraph(a, b);//设置画布大小

BeginBatchDraw();

loadimage(&img\_bk, "..\\source\\background.jpg");//背景

loadimage(&img\_bd1, "..\\source\\bird1.jpg");

loadimage(&img\_bd2, "..\\source\\bird2.jpg");//鸟

loadimage(&img\_bar\_up1, "..\\source\\bar\_up1.gif");

loadimage(&img\_bar\_up2, "..\\source\\bar\_up2.gif");//上方柱子图片

loadimage(&img\_bar\_down1, "..\\source\\bar\_down1.gif");

loadimage(&img\_bar\_down2, "..\\source\\bar\_down2.gif");//下方柱子图片

loadimage(&num1[0], "..\\source\\01.jpg");

loadimage(&num2[0], "..\\source\\02.jpg");

loadimage(&num1[1], "..\\source\\11.jpg");

loadimage(&num2[1], "..\\source\\12.jpg");

loadimage(&num1[2], "..\\source\\21.jpg");

loadimage(&num2[2], "..\\source\\22.jpg");

loadimage(&num1[3], "..\\source\\31.jpg");

loadimage(&num2[3], "..\\source\\32.jpg");

loadimage(&num1[4], "..\\source\\41.jpg");

loadimage(&num2[4], "..\\source\\42.jpg");

loadimage(&num1[5], "..\\source\\51.jpg");

loadimage(&num2[5], "..\\source\\52.jpg");

loadimage(&num1[6], "..\\source\\61.jpg");

loadimage(&num2[6], "..\\source\\62.jpg");

loadimage(&num1[7], "..\\source\\71.jpg");

loadimage(&num2[7], "..\\source\\72.jpg");

loadimage(&num1[8], "..\\source\\81.jpg");

loadimage(&num2[8], "..\\source\\82.jpg");

loadimage(&num1[9], "..\\source\\91.jpg");

loadimage(&num2[9], "..\\source\\92.jpg");//数字

mciSendString("open ..\\source\\background(1).mp3 alias bkmusic", NULL, 0, NULL);

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

}

void show()

{

putimage(0, 0, &img\_bk);//载入背景

putimage(bird\_x, bird\_y, &img\_bd1, NOTSRCERASE);

putimage(bird\_x, bird\_y, &img\_bd2, SRCERASE);//载入小鸟

putimage(bar\_up\_x, bar\_up\_y, &img\_bar\_up1, NOTSRCERASE);

putimage(bar\_up\_x, bar\_up\_y, &img\_bar\_up2, SRCERASE);//载入上方柱子

putimage(bar\_down\_x, bar\_down\_y, &img\_bar\_down1, NOTSRCERASE);

putimage(bar\_down\_x, bar\_down\_y, &img\_bar\_down2, SRCERASE);//输出下方柱子

putimage(bar\_up\_x2, bar\_up\_y2, &img\_bar\_up1, NOTSRCERASE);

putimage(bar\_up\_x2, bar\_up\_y2, &img\_bar\_up2, SRCERASE);//载入上方柱子

putimage(bar\_down\_x2, bar\_down\_y2, &img\_bar\_down1, NOTSRCERASE);

putimage(bar\_down\_x2, bar\_down\_y2, &img\_bar\_down2, SRCERASE);//输出下方柱子

putimage(bar\_up\_x3, bar\_up\_y3, &img\_bar\_up1, NOTSRCERASE);

putimage(bar\_up\_x3, bar\_up\_y3, &img\_bar\_up2, SRCERASE);//载入上方柱子

putimage(bar\_down\_x3, bar\_down\_y3, &img\_bar\_down1, NOTSRCERASE);

putimage(bar\_down\_x3, bar\_down\_y3, &img\_bar\_down2, SRCERASE);//输出下方柱子

if (score < 10)

{

putimage(e1, f, &num1[score], NOTSRCERASE);

putimage(e1, f, &num2[score], SRCINVERT);

}

else

{

right = score % 10;

left = score / 10;

putimage(e1, f, &num1[left], NOTSRCERASE);

putimage(e1, f, &num2[left], SRCINVERT);

putimage(e2, f, &num1[right], NOTSRCERASE);

putimage(e2, f, &num2[right], SRCINVERT);

}

FlushBatchDraw();

Sleep(50);

}

void bird()

{

char input;

if (\_kbhit())

{

input = \_getch();

if (input == ' ')

\_getch();

}

else if (bird\_y < 580)

bird\_y += 3;

while (MouseHit())

{

// 获取一条鼠标消息

m = GetMouseMsg();

switch (m.uMsg)

{

case WM\_LBUTTONDOWN:

{

if (bird\_y > 20)

bird\_y -= 60;

mciSendString("closejpmusic", NULL, 0, NULL);

mciSendString("open ..\\source\\Jump.mp3 alias jpmusic", NULL, 0, NULL);

mciSendString("open music", NULL, 0, NULL);

}

break;

case WM\_RBUTTONDOWN:

bird\_y += 60;

break;

}

}

}

void bar()

{

bar\_up\_x = bar\_up\_x - n;

bar\_down\_x = bar\_down\_x - n;

if (bar\_up\_x == -140 && bar\_down\_x == -140)

{

bar\_up\_x = 1140;

bar\_down\_x = 1140;

bar\_up\_y = rand() % -310 - 550;//随机上方柱子出现的坐标

bar\_down\_y = bar\_up\_y + 700;//下方柱子紧跟上方柱子

putimage(bar\_up\_x, bar\_up\_y, &img\_bar\_up1, NOTSRCERASE);

putimage(bar\_up\_x, bar\_up\_y, &img\_bar\_up2, SRCERASE);//载入上方柱子

putimage(bar\_down\_x, bar\_down\_y, &img\_bar\_down1, NOTSRCERASE);

putimage(bar\_down\_x, bar\_down\_y, &img\_bar\_down2, SRCERASE);//输出下方柱子

}

}

void bar2()

{

bar\_up\_x2 = bar\_up\_x2 - n;

bar\_down\_x2 = bar\_down\_x2 - n;

if (bar\_up\_x2 == -140 && bar\_up\_x2 == -140)

{

bar\_up\_x2 = 1140;

bar\_down\_x2 =1140;

bar\_up\_y2 = rand() % -310 - 550;//随机上方柱子出现的坐标

bar\_down\_y2 = bar\_up\_y2 + 700;//下方柱子紧跟上方柱子

putimage(bar\_up\_x2, bar\_up\_y2, &img\_bar\_up1, NOTSRCERASE);

putimage(bar\_up\_x2, bar\_up\_y2, &img\_bar\_up2, SRCERASE);//载入上方柱子

putimage(bar\_down\_x2, bar\_down\_y2, &img\_bar\_down1, NOTSRCERASE);

putimage(bar\_down\_x2, bar\_down\_y2, &img\_bar\_down2, SRCERASE);//输出下方柱子

}

}

void bar3()

{

bar\_up\_x3 = bar\_up\_x3 - n;

bar\_down\_x3 = bar\_down\_x3 - n;

if (bar\_up\_x3 == -140 && bar\_up\_x3 == -140)

{

bar\_up\_x3 = 1140;

bar\_down\_x3 = 1140;

bar\_up\_y3 = rand() % -310 - 550;//随机上方柱子出现的坐标

bar\_down\_y3 = bar\_up\_y3 + 700;//下方柱子紧跟上方柱子

putimage(bar\_up\_x3, bar\_up\_y3, &img\_bar\_up1, NOTSRCERASE);

putimage(bar\_up\_x3, bar\_up\_y3, &img\_bar\_up2, SRCERASE);//载入上方柱子

putimage(bar\_down\_x3, bar\_down\_y3, &img\_bar\_down1, NOTSRCERASE);

putimage(bar\_down\_x3, bar\_down\_y3, &img\_bar\_down2, SRCERASE);//输出下方柱子

}

}

void endgame()

{

IMAGE gameover1, gameover2;

loadimage(&gameover1, "..\\source\\gameover1.gif");

loadimage(&gameover2, "..\\source\\gameover2.gif");

putimage(c, d, &gameover1, NOTSRCERASE);

putimage(c, d, &gameover2, SRCINVERT);

FlushBatchDraw();

\_getch();

score = 0;

}

void judge()

{

if ((bar\_up\_x < 109 && bar\_down\_x < 109) && (bar\_up\_x > -75 && bar\_down\_x > -75))//柱子与小鸟在横向坐标上的碰撞判断

{

if ((bird\_y < (bar\_up\_y + 576) && bird\_y < (bar\_up\_y + 600)) || (bird\_y > (bar\_up\_y + 676) && bird\_y > (bar\_up\_y + 700)))//柱子与小鸟在纵向坐标上的碰撞判断

endgame();

}

else if((bar\_up\_x2 < 109 && bar\_down\_x2 < 109) && (bar\_up\_x2 > -75 && bar\_down\_x2 > -75))//柱子与小鸟在横向坐标上的碰撞判断

{

if ((bird\_y < (bar\_up\_y2 + 576) && bird\_y < (bar\_up\_y2 + 600)) || (bird\_y > (bar\_up\_y2 + 676) && bird\_y > (bar\_up\_y2 + 700)))//柱子与小鸟在纵向坐标上的碰撞判断

endgame();

}

else if ((bar\_up\_x3 < 109 && bar\_down\_x3 < 109) && (bar\_up\_x3 > -75 && bar\_down\_x3 > -75))//柱子与小鸟在横向坐标上的碰撞判断

{

if ((bird\_y < (bar\_up\_y3 + 576) && bird\_y < (bar\_up\_y3 + 600)) || (bird\_y > (bar\_up\_y3 + 676) && bird\_y > (bar\_up\_y3 + 700)))//柱子与小鸟在纵向坐标上的碰撞判断

endgame();

}

else if (bird\_y > 566)

endgame();

if ((bar\_up\_x == 80)|| (bar\_up\_x2 == 80)|| (bar\_up\_x3 == 80))//注意，判定的值需要同时是不同速度的公倍数，否则会错过

{

score++;

}

}

int main()

{

char num;

printf("————————————————————————————————————————————\n");

printf("操作指示：空格暂停，鼠标左键上升一格，右键下降一格,死亡后长按空格至安全地带方可继续\n");

printf("————————————————————————————————————————————\n");

printf("请选择难度1/2/3（难度1至3依次变难）: ");

scanf\_s("%c", &num);

switch (num)

{

case '1':a = 350, b = 600,c=80,d=200,e1=290,e2=300; break;

case '2':a = 1000, b = 600,c=400,d=200,e1=900,e2=910; break;

case '3':a = 1000, b = 600, n = 10,c=400,d=200,e1=900,e2=910; break;

}

bird\_x = 75;

bird\_y = 200;//小鸟的初始坐标

//↓↓↓柱子们的初始坐标

bar\_up\_x = 740;

bar\_down\_x = 740;

bar\_up\_y = rand() % -310 - 550;

bar\_down\_y = bar\_up\_y + 700;

bar\_up\_x2 = 1240;

bar\_down\_x2 = 1240;

bar\_up\_y2 = rand() % -310 - 550;

bar\_down\_y2 = bar\_up\_y2 + 700;

bar\_up\_x3 = 1640;

bar\_down\_x3 = 1640;

bar\_up\_y3 = rand() % -310 - 550;

bar\_down\_y3 = bar\_up\_y3 + 700;

//↑↑↑初始坐标

startup();

while (\_kbhit)

{

show();

bar();

bar2();

bar3();

bird();

judge();

}

\_getch();

}

5 课程设计总结

看似游戏的程序很长很难，其实并不，比如我的这个flappy bird， 真正影响游玩的核心代码其实也就不超过20行，好比房梁，剩下的周边只需要像砖头一样慢慢垒上去就好。把它细化成一个一个的函数，一个一个的来其实也没那么的难。

参考文献

[1] 童晶，C语言课程设计与游戏开发实践教程。清华大学出版社