****

中南大学课程实践报告

实践名称：计算机程序设计实践

实践题目：推箱子游戏

指导教师：张 伟

专 业：计算机与通信类

班 级：2215

姓 名：陈康坤

学 号：8208221512

2021年 12月 26日

目录

一 实践题目及设计目标

二 算法流程图及说明

三 调试及运行分析

四 源代码清单

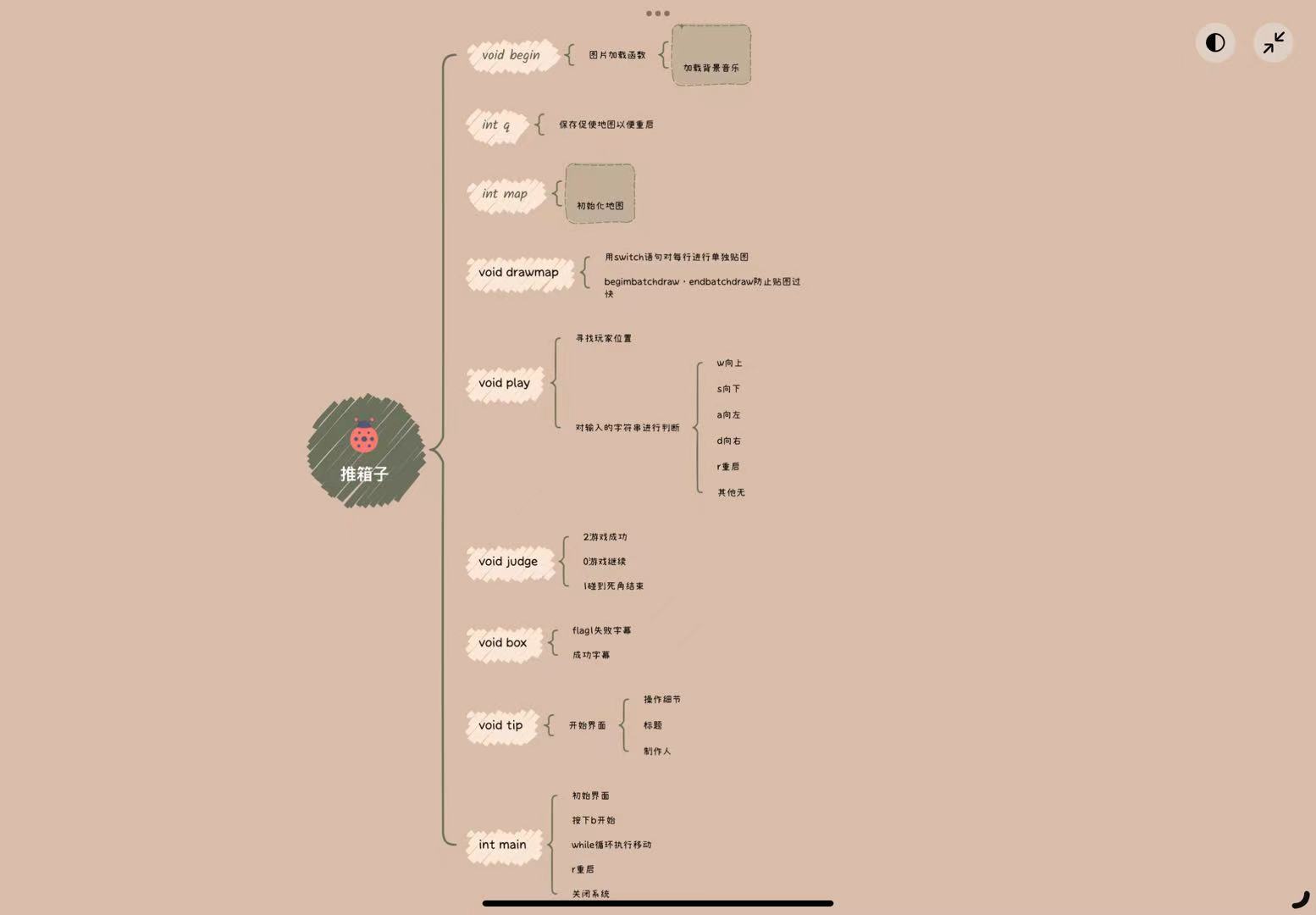
五 总结

一 实践题目及设计目标

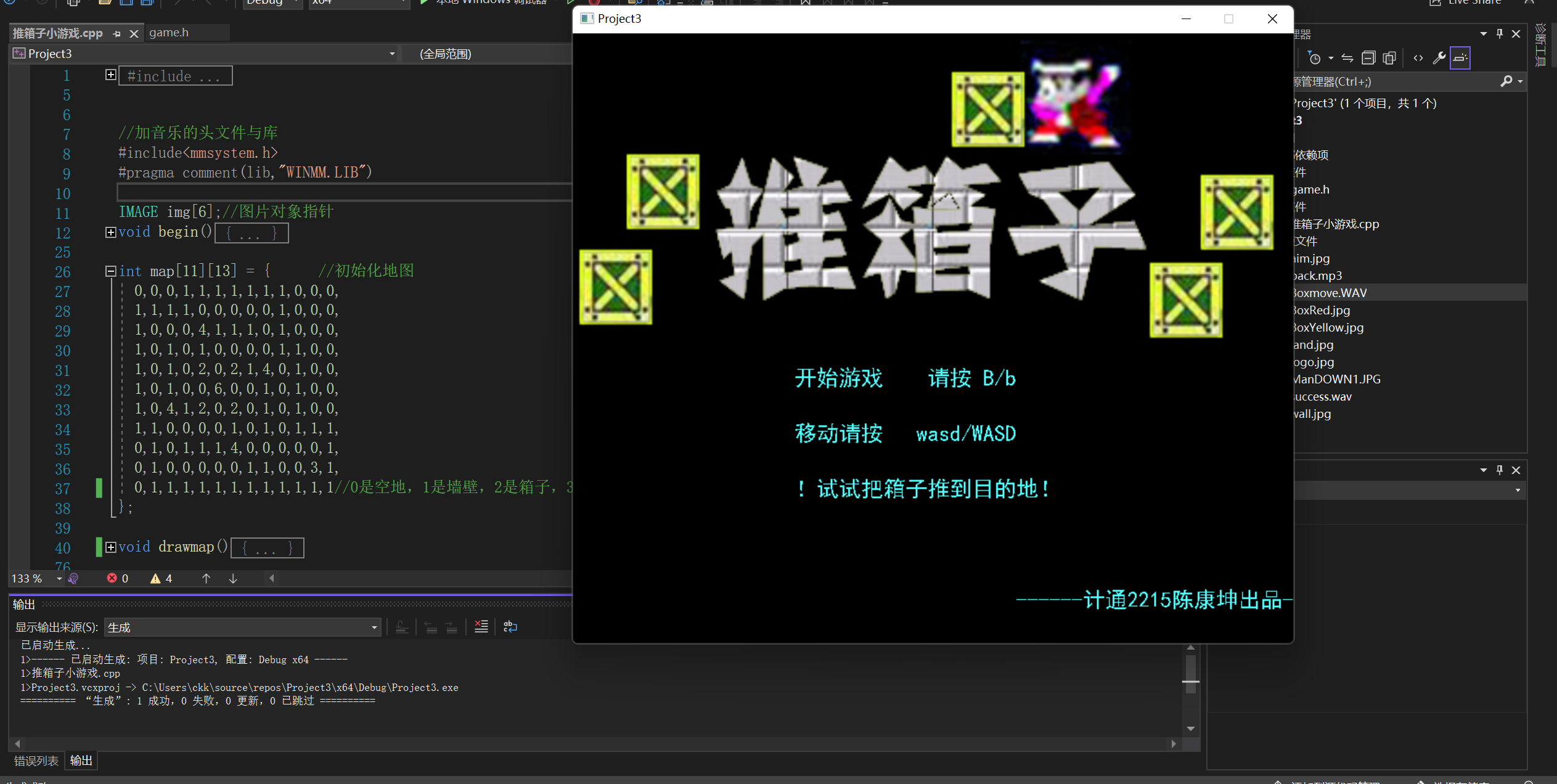
推箱子游戏 ：推箱子算法及图形编程库函数

掌握程序设计的基本方法，训练我们运用编程语言进行基本的结构化程序设计。使我们加深对编程语言的理解，培养我们设计、编写、调试程序的能力，同时培养我们的程序设计思维。

二 算法流程图及说明



三 调试及运行分析



四 源代码清单

#include<stdio.h>

#include<easyx.h>

#include<conio.h>

//加音乐的头文件与库

#include<mmsystem.h>

#pragma comment(lib,"WINMM.LIB")

IMAGE img[6];//图片对象指针

void begin()//图片加载函数

{

loadimage(&img[0],"land.jpg",60,60);//加载图片

loadimage(&img[1],"wall.jpg",60,60);//(IMAGE 对象指针,图片文件名,宽度,高度)

loadimage(&img[2],"BoxYellow.jpg",60,60);

loadimage(&img[3],"ManDOWN1.jpg",60,60);

loadimage(&img[4],"aim.jpg",60,60);

loadimage(&img[5],"BoxRed.jpg",60,60);//箱子在目的地时

//加载音乐

mciSendString("open back.mp3 alias bgm",0,0,0);

mciSendString("play bgm repeat",0,0,0);

}

int map[11][13] = { //初始化地图

0,0,0,1,1,1,1,1,1,1,0,0,0,

1,1,1,1,0,0,0,0,0,1,0,0,0,

1,0,0,0,4,1,1,1,0,1,0,0,0,

1,0,1,0,1,0,0,0,0,1,1,0,0,

1,0,1,0,2,0,2,1,4,0,1,0,0,

1,0,1,0,0,6,0,0,1,0,1,0,0,

1,0,4,1,2,0,2,0,1,0,1,0,0,

1,1,0,0,0,0,1,0,1,0,1,1,1,

0,1,0,1,1,1,4,0,0,0,0,0,1,

0,1,0,0,0,0,0,1,1,0,0,3,1,

0,1,1,1,1,1,1,1,1,1,1,1,1//0是空地，1是墙壁，2是箱子，3是人物，4是目的地，6是箱子在目的地上面

};

void drawmap()

{

BeginBatchDraw();// 防止贴图过快

for(int i=0;i<11;i++)//行

{

for(int j=0;j<13;j++)//列

{

//每行单独贴图

switch(map[i][j])

{

case 0://空地

putimage(j\*60,i\*60,&img[0]);//(x坐标,y坐标,对象指针)

break;

case 1://墙

putimage(j\*60,i\*60,&img[1]);

break;

case 2://箱子

putimage(j\*60,i\*60,&img[2]);

break;

case 3:

case 7://人物

putimage(j\*60,i\*60,&img[3]);

break;

case 4://目的地

putimage(j\*60,i\*60,&img[4]);

break;

case 6://箱子到达目的地

putimage(j\*60,i\*60,&img[5]);

break;

default:

break;

}

}

}

EndBatchDraw();

}

void play()//定位

{

//找人的位置

int x,y;

for(int i=0;i<11;i++)

{

for(int j=0;j<13;j++)

{

if(map[i][j] == 3|| map[i][j] == 7)

{

x=i;

y=j;

}

}

}

switch(\_getch())

{

case 'W':

case 'w':

//往上

if(map[x-1][y]==0||map[x-1][y]==4) //人上方是空地或目的地

{

map[x-1][y]+=3;//人走上去 +3

map[x][y]-=3;//原位置-3

}

else if(map[x-1][y]==2||map[x-1][y]==6) //上方为箱子或箱子在目的地上

{

if(map[x-2][y]==0||map[x-2][y]==4) //箱子上方是空地或目的地

{

//人走上去，箱子也推上去

map[x-2][y]+=2;

map[x-1][y]+=1;//-2 +3 --+1

map[x][y]-=3;

}

}

break;

case 'S':

case 's':

//往下

if(map[x+1][y]==0||map[x+1][y]==4) //人下方是空地或目的地

{

map[x+1][y]+=3;//人走下去 +3

map[x][y]-=3;//原位置-3

}

else if(map[x+1][y]==2||map[x+1][y]==6) //下方为箱子或箱子在目的地上

{

if(map[x+2][y]==0||map[x+2][y]==4) //箱子下方是空地或目的地

{

//人走下去，箱子也推下去

map[x+2][y]+=2;

map[x+1][y]+=1;//-2 +3 =+1

map[x][y]-=3;

}

}

break;

case 'A':

case 'a':

//往左

if(map[x][y-1]==0||map[x][y-1]==4) //人左方是空地或目的地

{

map[x][y-1]+=3;//人往左走 +3

map[x][y]-=3;//原位置-3

}

else if(map[x][y-1]==2||map[x][y-1]==6) //左方为箱子或箱子在目的地上

{

if(map[x][y-2]==0||map[x][y-2]==4) //箱子左方是空地或目的地

{

//人往左走，箱子也推向左边

map[x][y-2]+=2;

map[x][y-1]+=1;//-2 +3 --+1

map[x][y]-=3;

}

}

break;

case 'D':

case 'd':

//往右

if(map[x][y+1]==0||map[x][y+1]==4) //人右方是空地或目的地

{

map[x][y+1]+=3;//人往右走 +3

map[x][y]-=3;//原位置-3

}

else if(map[x][y+1]==2||map[x][y+1]==6) //右方为箱子或箱子在目的地上

{

if(map[x][y+2]==0||map[x][y+2]==4) //箱子右方是空地或目的地

{

//人往右走，箱子也推向右边

map[x][y+2]+=2;

map[x][y+1]+=1;//-2 +3 --+1

map[x][y]-=3;

}

}

break;

default ://不满足所有case的情况下，会进default, 若不写则直接跳出switch

break;

}

PlaySound("Boxmove.WAV",nullptr,SND\_FILENAME | SND\_ASYNC);

}

int flag;

void judge()

{

flag=2;//初值

for(int i=0;i<11;i++)//行

{

for(int j=0;j<13;j++)//列

{

if(map[i][j]==2)

{

//表明未结束

flag=0;

if(map[i-1][j]==1||map[i+1][j]==1)

{

//找死角的墙

if(map[i][j-1]==1||map[i][j-1]==1)

{

//游戏结束

flag=1; //flag 分类

return; //=0 继续

//=1 结束

//=2 赢

}

}

}

}

}

}

void box()

{

if(flag==1)

{

MessageBox(GetHWnd(),"抱歉，你输了。","LOSE",MB\_OK);//头文件为windows.h

}

else

{

PlaySound("success.wav",nullptr,SND\_FILENAME | SND\_ASYNC);

MessageBox(GetHWnd(),"恭喜，你赢了!","VICTORY",MB\_OK);

}

}

IMAGE iae;

void tip()

{

loadimage(&iae,"logo.jpg",780,360);

putimage(0,0,&iae);

settextcolor(LIGHTCYAN);

settextstyle(24,0,\_T("黑体"));

outtextxy(240,360,\_T("开始游戏 请按 B/b "));

outtextxy(240,420,\_T("移动请按 wasd/WASD"));

outtextxy(240,480,\_T("！试试把箱子推到目的地！"));

outtextxy(480,600,\_T("------计通2215陈康坤出品------"));

}

int main()

{

initgraph(780,660);//每个方块大小60\*60 窗口大小780\*660

tip();

if(\_getch()=='B'||\_getch()=='b')

{

begin();

while(1)

{

drawmap();

judge();

if(flag) break;//判断是否结束

play();

}

box();

}

getchar();

closegraph();//关闭窗口

return 0;

}

五 总结

图形库函数

1.putimage() 输出图像函数

功能： 函数putimage()将一个先前保存在内存中的图像输出到屏幕上。

用法： 此函数调用方式为void putimage(int left,int top,void \*buf,int ops);

说明： 参数(left,top)为输出屏幕图像的左上角，即输出图像的起始位置。buf指向要输出的内存中图像。参数ops控制图像以何种方式输出到屏幕上。表1-13给出了图像输出方式。

2.outtextxy

这个函数用于在指定位置输出字符串。 void outtextxy(     int x,     int y,     LPCTSTR str ); void outtextxy(     int x,     int y,     TCHAR c ); 参数： x 字符串输出时头字母的 x 轴的坐标值 y 字符串输出时头字母的 y 轴的坐标值。 str 待输出的字符串的指针。 c 待输出的字符

3.播放声音

PlaySound(L"F:\\c.wav",NULL,SND\_FILENAME | SND\_ASYNC);

SND\_ASYNC  
 用异步方式播放声音，PlaySound函数在开始播放后立即返回。  
   
SND\_FILENAME  
 pszSound参数指定了WAVE文件名。  
 4.函数名: initgraph()   
　　功 能: 初始化 [图形系统](http://baike.so.com/doc/2963725.html)

这个函数用于初始化绘图环境。

HWND initgraph(

int width,

int height,

int flag = NULL

);

参数：

width

绘图环境的宽度。

height

绘图环境的高度。

flag

绘图环境的样式，默认为 NULL。可为以下值：