##### 南京大学

**课程设计报告**

课程名称 C程序设计

设计题目:\_ \_\_\_ \_\_\_五子棋小游戏 \_\_\_

姓 名: 龚亮 袁浩然 朴慧杰 佧米然

学 号: 231830131 231830153 221830213 231830222

指导教师: 潘亦

完成日期： 2023 年11月 27 日

**概 况**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 作品名称 | | 五子棋小游戏 | | | 总代码量（行） | 620 |
| 小组成员 | | | | | | |
| 学号 | 姓名 | 院系 | 手机号 | 在程序开发中承担的主要任务 | | |
| 231830131 | 龚亮 | 安邦书院 | 13382133508 | 代码编写  报告 | | |
| 231830153 | 袁浩然 | 安邦书院 |  | PPT  图片与音乐  初期代码编写和设计 | | |
| 221830213 | 朴慧杰 | 地理与海洋科学 | 17631403839 | 程序报告撰写  图片与音乐 | | |
| 231830222 | 佧米然 | 安邦书院 |  | PPT | | |
| 程序功能简介 | | 一款针对五子棋新手的练习提高小游戏 | | | | |
| 程序的实用性与特色 | | 进行五子棋的练习 | | | | |
| 使用的编译系统及  相关软件 | | EasyX，Dev-C++（Clion辅助） | | | | |
| 对程序运行的环境要求  （软硬件、资源文件存储位置等） | | 64位操作系统，背景图片与音频文件放到与可执行文件同一文件夹下，VS编译软件 | | | | |
| 在开发程序过程中自学了哪些内容 | | 图形界面设计，EasyX，C++语句，图片与音频输入 | | | | |
| 不足之处 | | 不能进行ai人机模拟练习 | | | | |
| 参考资料 | | 《C语言程序设计教程》  《EasyX\_Help》 | | | | |

**目录**

[一、设计内容与要求 4](#_Toc59990510)

[二、总体设计 4](#_Toc59990511)

[(1)模块划分 4](#_Toc59990512)

[(2)程序运行思路 5](#_Toc59990513)

[三、详细设计 6](#_Toc59990514)

[(1)数据结构 6](#_Toc59990515)

[(2)程序界面 6](#_Toc59990516)

[(3)主要函数的原型、功能 6](#_Toc59990517)

[(4)算法流程 8](#_Toc59990518)

[(5)调用关系 8](#_Toc59990519)

[四、开发过程中遇到的主要问题及解决方法 8](#_Toc59990520)

[五、完成课程设计的收获与体会 8](#_Toc59990521)

[六、学完本课程的收获体会及给老师的教学建议 9](#_Toc59990522)

[七、软件使用方法 9](#_Toc59990523)

八、源代码...............................................................................................................................9

## **设计内容与要求**

菜单界面：



实现基础五子棋的练习。同时实现记录时间功能，方便用户查阅相关记录。

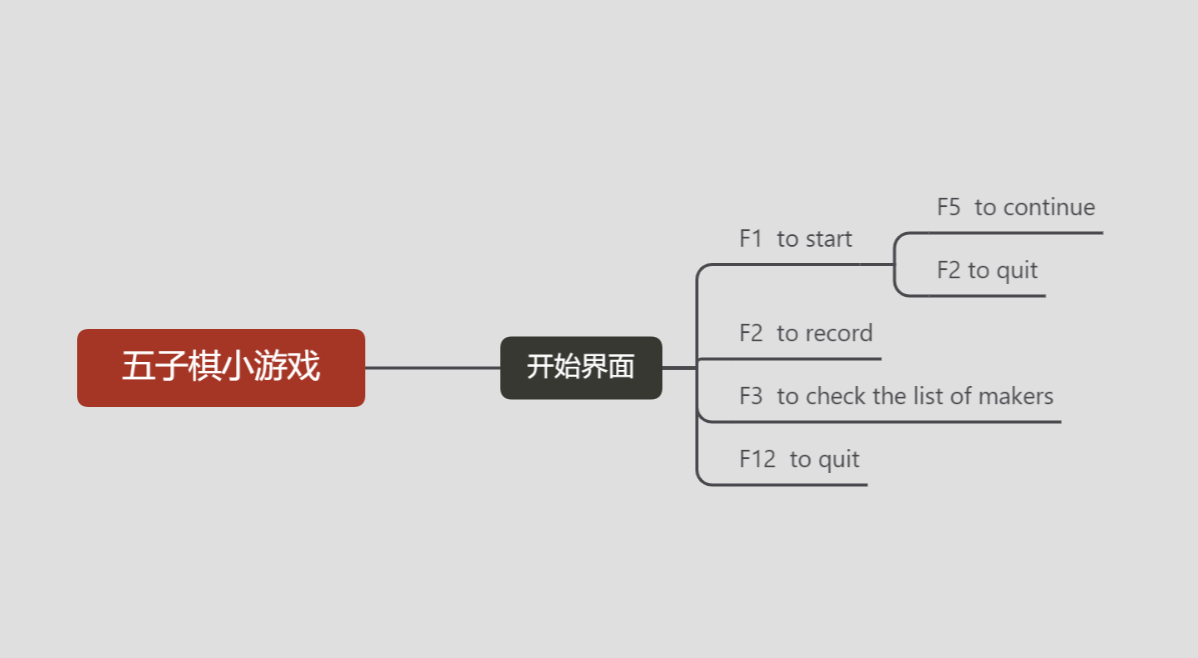
## **总体设计**

### (1)模块划分

头文件：

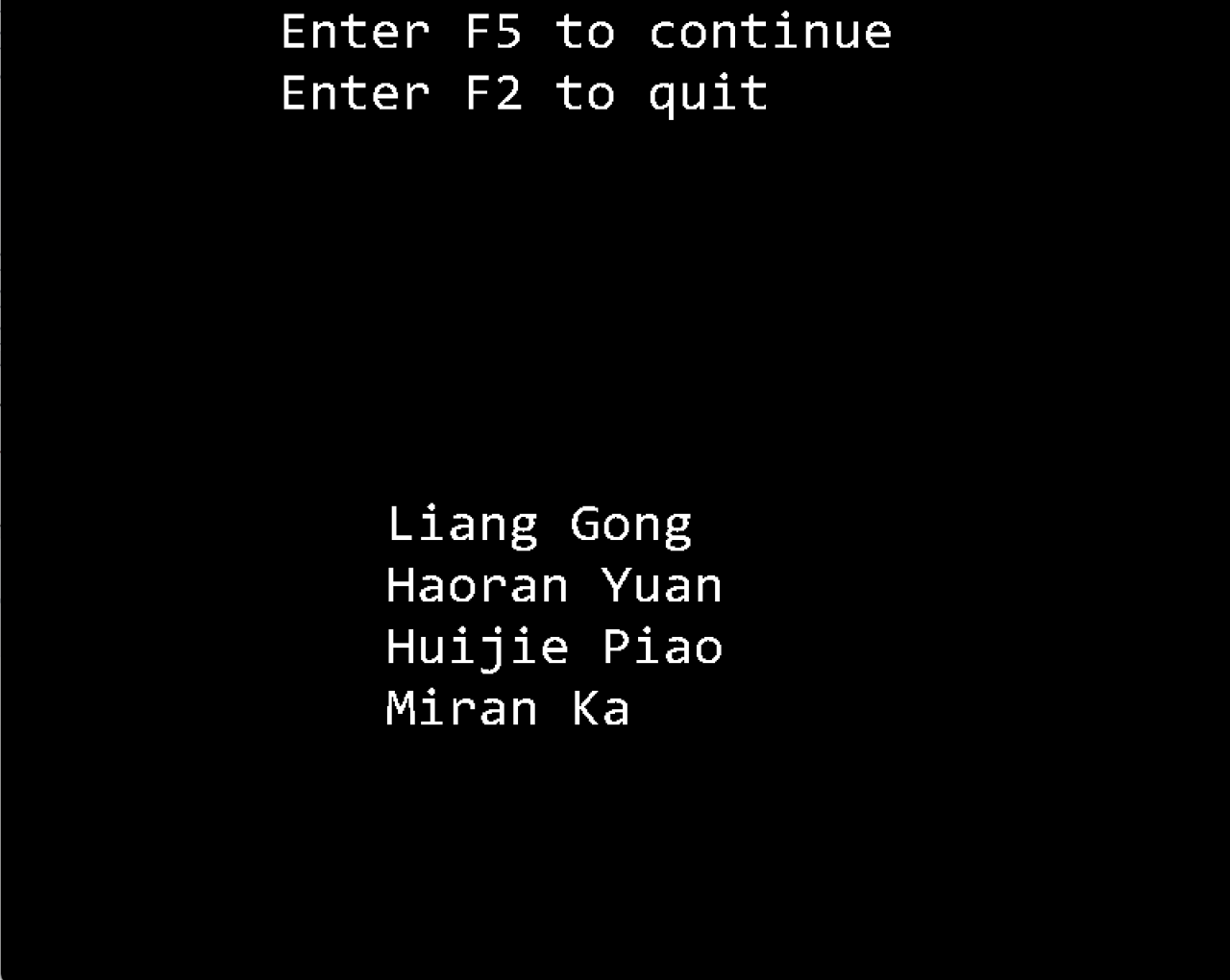
#include <graphics.h>  
#include <conio.h>  
#include <easyx.h>  
#include <winuser.h>  
#include<mmsystem.h>//包含多媒体设备接口头文件  
#include<windows.h>  
#include <fstream>//txt  
#include <iostream>//txt  
#include<cstdlib>//txt  
#include<istream>//txt  
#include <string.h>  
#include <stdlib.h>  
#include <math.h>  
#include <time.h>

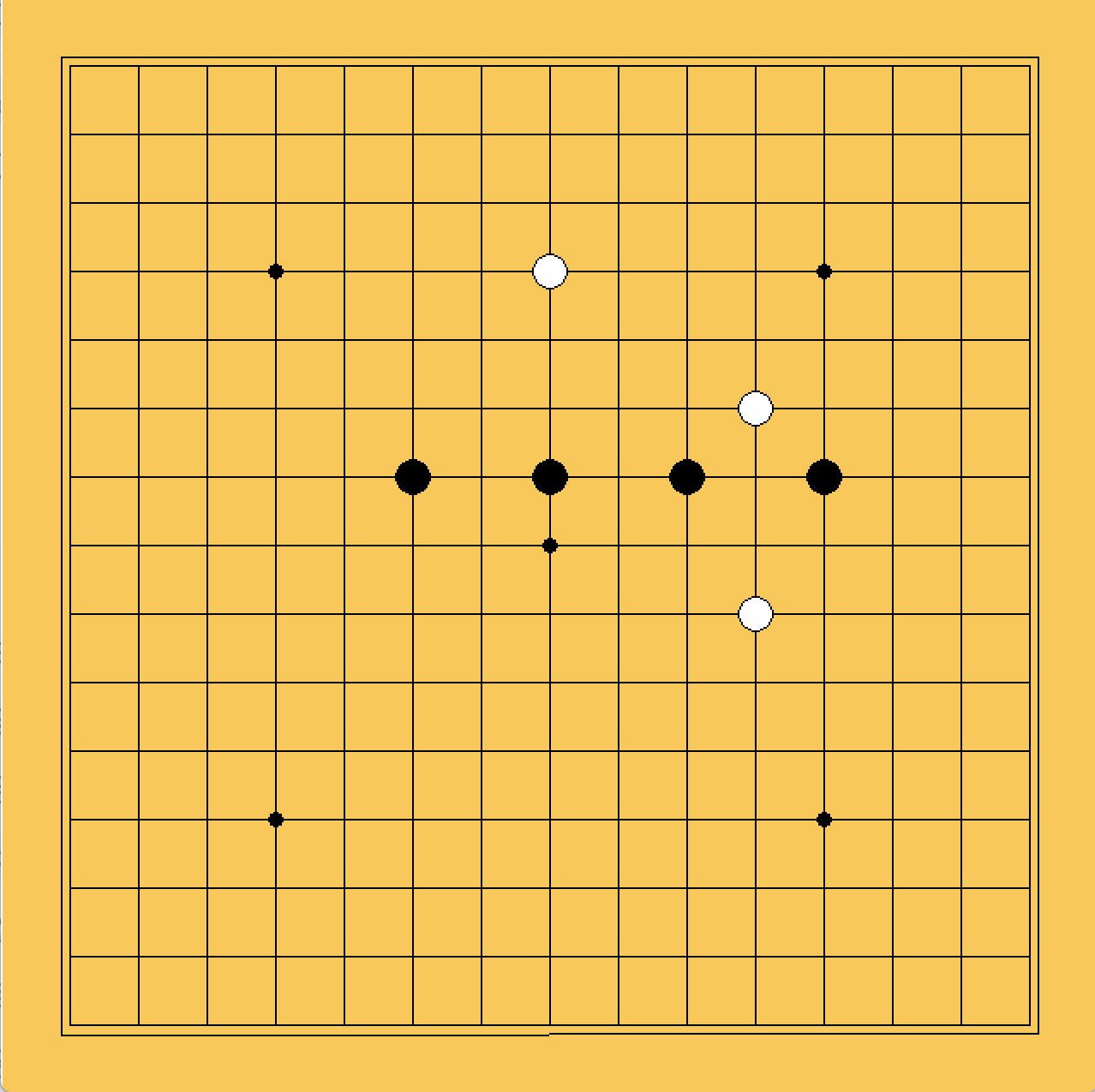
### (2)程序运行思路

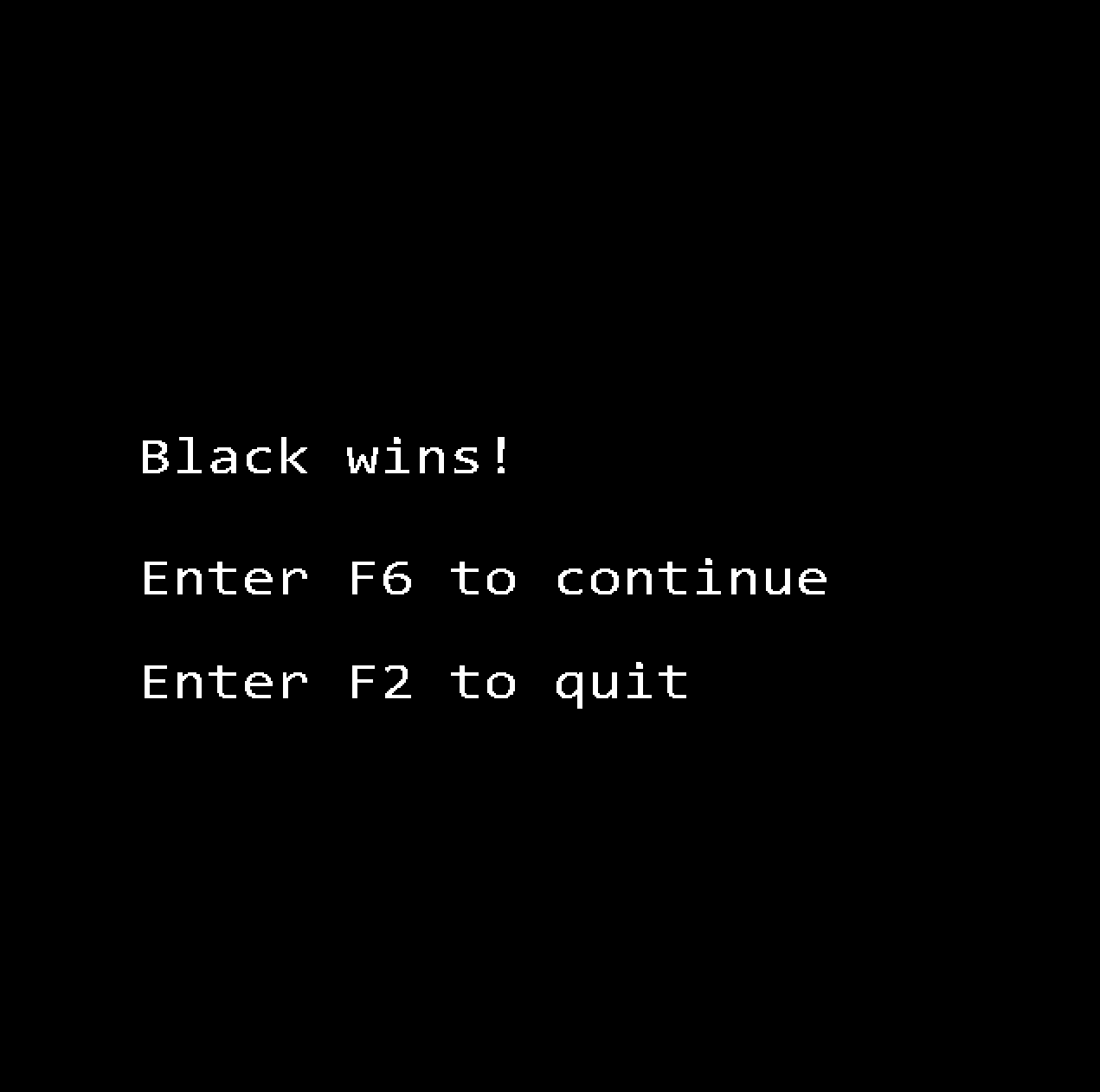


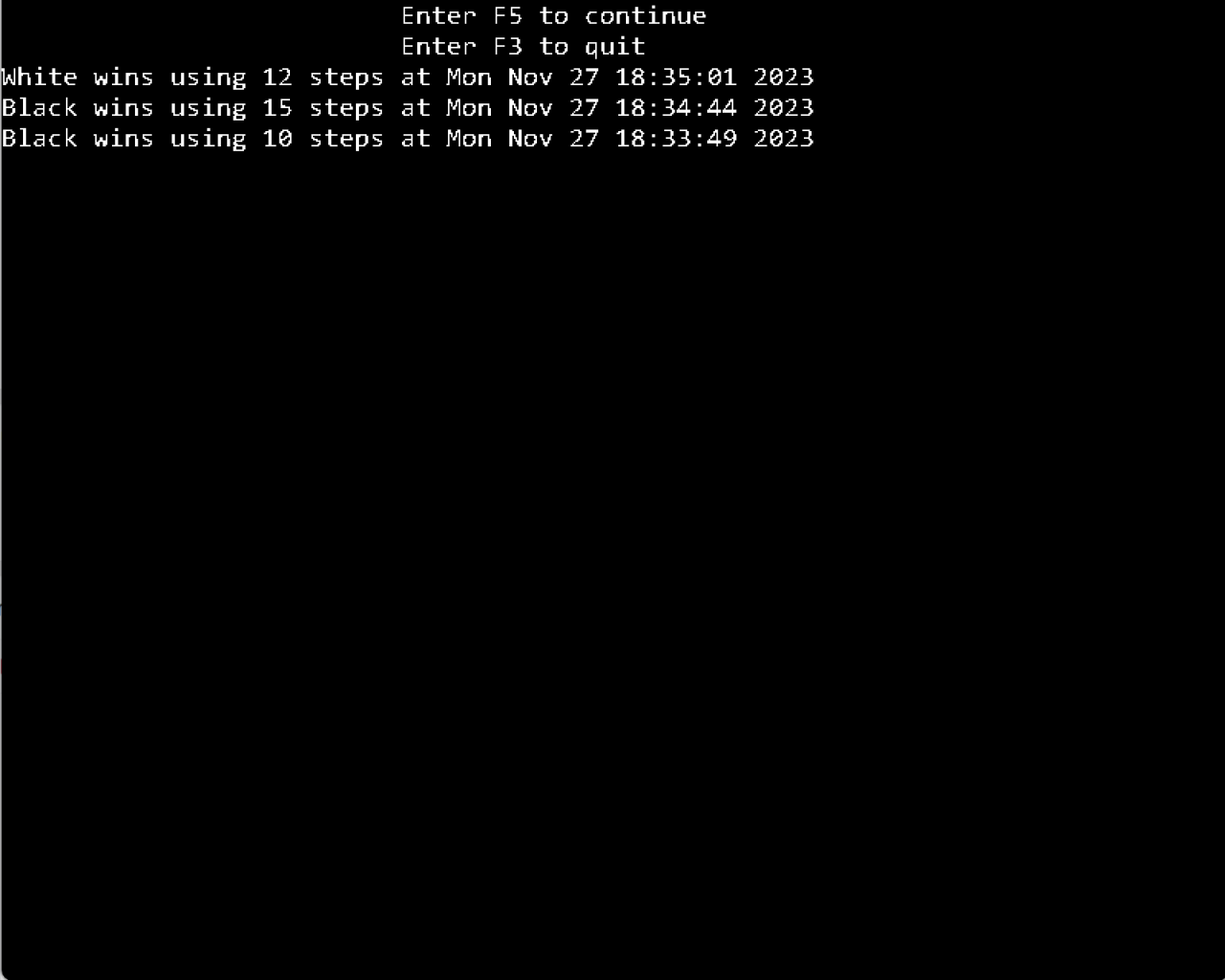
## **详细设计**

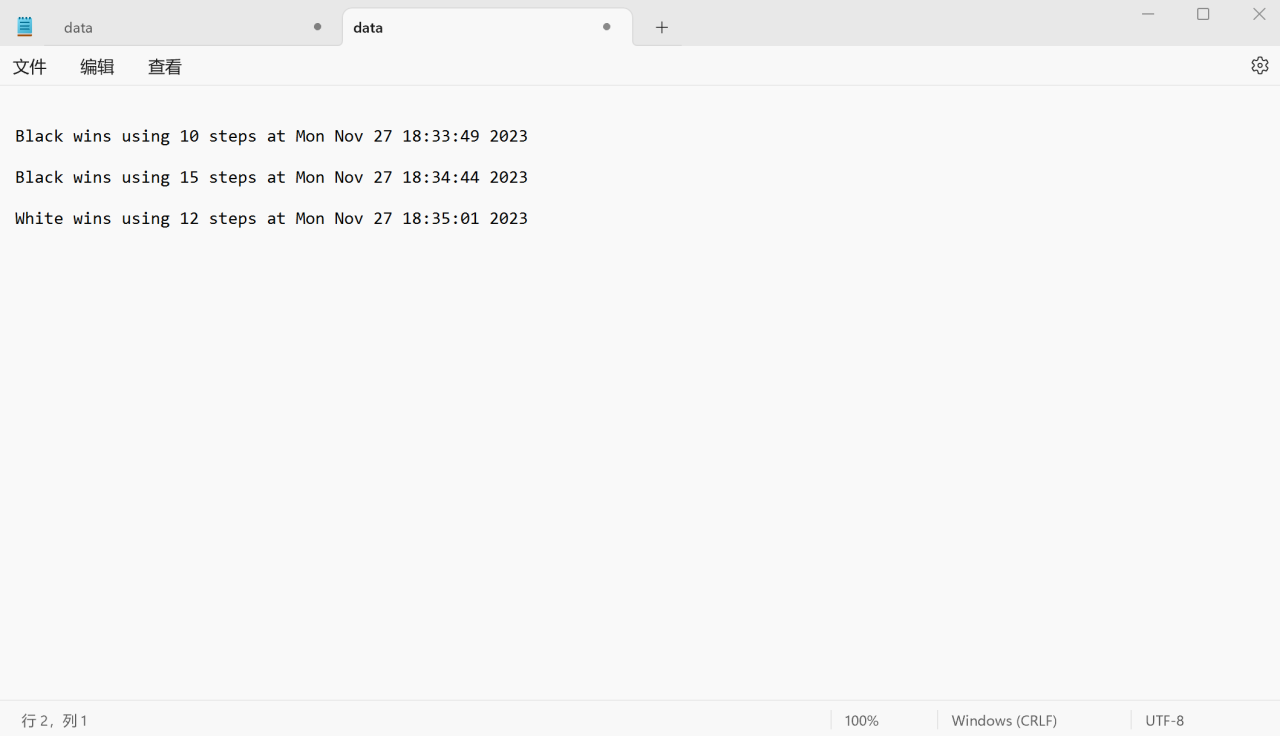
### (1)程序界面











### (3)主要函数的原型、功能

void start\_cover();//开始界面  
void game\_part();//游戏主程序部分  
void mousemeg(int\* x, int\* y);//读取鼠标点击的位置并赋值给x,y  
int mousxy(int\* x,int\* y);//标准化x,y  
int mousy(int y);//对y的判断  
int check\_map();//检查是否游戏结束  
void list\_of\_makers();//开发者名单  
int fileinput(char sen[]);//txt  
void result(char sen[],int outcome, int n);//将结果存入data.txt  
void recordings();//显示链表内容  
struct node\* add\_to\_record(struct node\* list, int n, int steps);//链表加长函数  
void record(int outcome, int steps);//将结果存入链表

### (4)算法流程

（1）初始化窗口

（2）绘制界面

（3）开始游戏

（4）输入并显示结果

（5）清空棋盘

（6）显示最终成绩

（7）数据记录与退出程序

### (5)调用关系

Main调用startover函数，startover函数再调用其他函数。

## **开发过程中遇到的主要问题及解决方法**

音乐、图片无法播放导入，没有解决（在mingW编译器上无法完成）；

ai棋手没有完成，时间有限，课业压力较大；

对于easyx图形库相对陌生，一开始无从下手，最后通过仔细阅读easyx文档并在csdn博客上查阅相关资料完成开发

## **完成课程设计的收获与体会**

完成大作业的过程中，我们组同学共同的感受就是：c语言实训和平时上课所接触的程序是大有不同的，所经受的考验和克服的困难是平时所无法比拟的。好在同组的搭档们精诚合作，分工明确，有问题共同解决，攻克了c语言实训的复杂程序。

## **学完本课程的收获体会及给老师的教学建议**

对于初学者而言，学习C语言，不仅仅是学习一门新的编程语言，更帮助我们初步了解了计算机的工作原理，包括它的数据存储方法与运算逻辑等等。这是一个崭新的世界，它让我们看到了计算机的“逻辑”，帮助我们从一个全新的角度去看待与解决问题，也让我们的思维逻辑更加缜密，提高了我们拆分与解决复杂问题的能力。

给老师的教学建议：希望老师能讲解一下一些特殊功能的用法。（播放音乐和导入图片在easyx文档中没有提到，将游戏数据存到文档中的方法对于C语言初学者并不友好）

## **软件使用方法**

界面1为程序的首页，进行程序使用说明

界面2为开发者的姓名介绍

开始后即进入游戏界面，一个五子棋盘

结束后，会公布得分及胜出的一方，游戏的历史数据可以在\Great\_Work\cmake-build-debug\data中查找到

## **完整源代码**

#include <graphics.h>

#include <easyx.h>

#include <winuser.h>

#include<mmsystem.h>//包含多媒体设备接口头文件

#include<windows.h>

#include<istream>//txt

#include <string.h>

#include <time.h>

#include <stdlib.h>

#include <stdio.h>

using namespace std;//txt

void start\_cover();//开始界面

void game\_part();//游戏主程序部分

void mousemeg(int\* x, int\* y);//读取鼠标点击的位置并赋值给x,y

int mousxy(int\* x,int\* y);//标准化x,y

int mousy(int y);//对y的判断

int check\_map();//检查是否游戏结束

void list\_of\_makers();//开发者名单

int fileinput(char sen[]);//txt

void result(char sen[],int outcome, int n);//将结果存入data.txt

void recordings();//显示链表内容

struct node\* add\_to\_record(struct node\* list, int n, int steps);//链表加长函数

void record(int outcome, int steps);//将结果存入链表

//void ai\_game();//ai游戏模式

//void ai(int \*x, int \*y);//ai下子选择

//int outcome\_ai();

#pragma comment(lib,"winmm.lib")//音乐

void bgm();

int game\_condition = 1;//是否重复游戏了

ExMessage m;//消息处理

int map[17][17] = {0};//初始化棋盘

int vectors[8][2] = {{1,0},{-1,0},{1,1},{-1,1},{0,1},

{1,-1},{0,-1},{-1,-1}};//辅助数列

struct node{

int outcome;

int steps;

char time[30];

struct node \*next;

};

struct node \*first = NULL;

int main()

{

bgm();

start\_cover();//开始界面

while(game\_condition == 0){

start\_cover();

}

closegraph();

return 0;

}

/\*start\_cover();//开始界面

while(game\_condition == 0){

start\_cover();

}

closegraph();

return 0;\*/

void start\_cover(){

initgraph(800,640);

fillrectangle(300,350,500,400);

setlinecolor(0);

settextstyle(40, 20, \_T("Consolas"));

outtextxy(350,350,"Start");

setlinecolor(0xFFFFFF);

fillrectangle(300,400,500,450);

setlinecolor(0);

outtextxy(350,400,"Record");

setlinecolor(0xFFFFFF);

fillrectangle(300,450,500,500);

setlinecolor(0);

outtextxy(350,450,"Makers");

setlinecolor(0xFFFFFF);

fillrectangle(300,500,500,550);

setlinecolor(0);

outtextxy(350,500,"Quit");

setlinecolor(0xFFFFFF);

settextstyle(20, 10, \_T("Consolas"));

outtextxy(0,0,"Enter F1 to start");

outtextxy(0,20,"Enter F2 to the temporary recordings");

outtextxy(0,40,"Enter F3 to check the list of makers");

outtextxy(0,80,"Enter F12 to quit");

outtextxy(0,60,"Enter F4 to play with ai(Not finished!Enter to continue.)");

outtextxy(0,100,R"(Enter \Great\_Work\cmake-build-debug\data to the permanent recordings)");

outtextxy(0,120,"Don't click your mouse too fast.");

outtextxy(0,140,"Otherwise, the program will break down!");

IMAGE img1;

loadimage(&img1, \_T("cover.jpg"),200,180);

putimage(300,170,&img1);

IMAGE img2;

loadimage(&img2, \_T("left.jpg"),300,640-170);

putimage(0,170,&img2);

IMAGE img3;

loadimage(&img3, \_T("right.jpg"),300,640-170);

putimage(500,170,&img3);

flushmessage();

while(1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x7B){

game\_condition = 1;

return;

} else if (m.vkcode == 0x70){//按F1开始游戏

game\_part();//开始游戏

return;

} else if (m.vkcode == 0x71){

recordings();

return;

} else if (m.vkcode == 0x72){

list\_of\_makers();

return;

} else if (m.vkcode == 0x73){

game\_condition = 0;

return;

}

}

}

void game\_part(){

//初始化棋盘

for (int i = 0; i < 17; i++){

for (int j = 0; j < 17; j++){

map[i][j] = 0;

}

}

if (game\_condition == 0){

cleardevice();

game\_condition = 1;

}

initgraph(640,640);

setfillcolor(RGB(249,200,91));

fillrectangle(0,0,640,640);

setlinecolor(0);

for (int i = 40; i < 640; i += 40){

line(40,i,600,i);

}

for (int i = 40; i < 640; i+= 40){

line(i,40,i,600);

}

line(35,35,35,605);

line(35,35,605,35);

line(605,35,605,605);

line(35,606,605,605);

setfillcolor(0);

fillcircle(40\*4,40\*4,4);

fillcircle(40\*12,40\*4,4);

fillcircle(40\*4,40\*12,4);

fillcircle(40\*12,40\*12,4);

fillcircle(40\*8,40\*8,4);

int countb = 0;

int countw = 0;

//下棋部分

int outcome = 0;

int flag = 1;

while (1){

int x, y;

mousemeg(&x,&y);

if (flag == 1){

for (int i = 1; i <= 15; i++){

for (int j = 1; j <= 15; j ++){

if (x == i && y == j){

setlinecolor(0);

setfillcolor(0);

fillcircle(i\*40,j\*40,10);

map[i][j] = 1;

countb++;

}

}

}

flag = 0;

outcome = check\_map();

} else {

for (int i = 1; i <= 15; i++){

for (int j = 1; j <= 15; j++) {

if (x == i && y == j) {

setlinecolor(0);

setfillcolor(0xFFFFFF);

fillcircle(i \* 40, j \* 40, 10);

map[i][j] = 2;

countw++;

}

}

}

flag = 1;

outcome = check\_map();

}

if (outcome == 1){

record(1,countb);

cleardevice();

char sen[50] = {0};

result(sen,outcome,countb);

fileinput(sen);

char black\_win[20] = "Black wins!";

settextstyle(35, 20, \_T("Consolas"));

outtextxy(80,250,black\_win);

char if\_return1[30] = "Enter F6 to continue";

outtextxy(80,320,if\_return1);

char if\_return2[20] = "Enter F2 to quit";

outtextxy(80,380,if\_return2);

flushmessage();

while (1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x75){

game\_condition = 0;

return;

} else if (m.vkcode == 0x71){

game\_condition = 1;

return;

}

}

} else if (outcome == 2){

record(2,countw);

cleardevice();

char sen[50] = {0};

result(sen,outcome,countw);

fileinput(sen);

char white\_win[20] = "White wins!";

settextstyle(35, 20, \_T("Consolas"));

outtextxy(80,250,white\_win);

char if\_return1[30] = "Enter F7 to continue";

outtextxy(80,320,if\_return1);

char if\_return2[20] = "Enter F2 to quit";

outtextxy(80,380,if\_return2);

//

flushmessage();

while (1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x76){

game\_condition = 0;

return;

} else if (m.vkcode == 0x71){

game\_condition = 1;

return;

}

}

} else if (outcome == 3){

record(3,0);

cleardevice();

char sen[50] = {0};

result(sen,outcome,countb);

fileinput(sen);

char black\_win[20] = "Draw!";

settextstyle(35, 20, \_T("Consolas"));

outtextxy(80,250,black\_win);

char if\_return1[30] = "Enter F6 to continue";

outtextxy(80,320,if\_return1);

char if\_return2[20] = "Enter F2 to quit";

outtextxy(80,380,if\_return2);

flushmessage();

while (1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x75){

game\_condition = 0;

return;

} else if (m.vkcode == 0x71){

game\_condition = 1;

return;

}

}

}

}

}

void mousemeg(int\* x, int\* y){

while (true) {

// 判断鼠标是否点击

if (MouseHit()) {

// 获取鼠标消息

MOUSEMSG msg = GetMouseMsg();

// 判断是否为左键按下事件

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

// 获取鼠标点击坐标

\*x = msg.x;

\*y = msg.y;

if (mousxy(x,y) == 1 && map[\*x][\*y] == 0){

break;

}

}

}

}

return;

}

int mousxy(int\* x, int\* y){

int result = 0;

for (int i = 40; i < 800; i += 40){

if (\*x >= i - 20 && \*x <= i + 20){

\*x = i / 40;

\*y = mousy(\*y);

result = 1;

}

}

return result;

}

int mousy(int y){

for (int i = 40; i < 640; i+= 40){

if (y >= i - 20 && y <= i + 20){

return i / 40;

}

}

return 0;

}

int check\_map(){

for (int i = 1; i < 16; i++){

for (int j = 1; j < 16; j++){

if (map[i][j] == 1){

int ti = i;

int tj = j;

for (int m = 1; m < 8; m++){

int count = 1;

while (map[ti + vectors[m][0]][tj + vectors[m][1]] == 1){

count++;

ti += vectors[m][0];

tj += vectors[m][1];

}

if (count >= 5){

return 1;

}

}

}

}

}

for (int i = 1; i < 16; i++){

for (int j = 1; j < 16; j++){

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

int ti = i;

int tj = j;

for (int m = 1; m < 8; m++){

int count = 1;

while (map[ti + vectors[m][0]][tj + vectors[m][1]] == 2){

count++;

ti += vectors[m][0];

tj += vectors[m][1];

}

if (count >= 5){

return 2;

}

}

}

}

}

int sum = 0;

for (int i = 1; i < 16; i++){

for (int j = 1; j < 16; j++){

if (map[i][j] == 1){

sum = 1;

}

}

}

if (sum == 0){

return 3;

}

return 0;

}

void list\_of\_makers(){

initgraph(800,640);

char maker1[30] = "Liang Gong";

settextstyle(40, 20, \_T("Consolas"));

outtextxy(250,320,maker1);

char maker2[30] = "Haoran Yuan";

outtextxy(250,360,maker2);

char maker3[30] = "Huijie Piao";

outtextxy(250,400,maker3);

char maker4[30] = "Miran Ka";

outtextxy(250,440,maker4);

char if\_return1[21] = "Enter F5 to continue";

outtextxy(180,0,if\_return1);

char if\_return2[20] = "Enter F2 to quit";

outtextxy(180,40,if\_return2);

flushmessage();

while (1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x74){

game\_condition = 0;

return;

} else if (m.vkcode == 0x71){

game\_condition = 1;

return;

}

}

}

/\*

void ai\_game() {

//初始化棋盘

for (int i = 1; i < 16; i++){

for (int j = 1; j < 16; j++){

map[i][j] = 0;

}

}

for (int i = 0 ; i < 17; i++){

map[0][i] = 3;

map[16][i] = 3;

}

for (int i = 0; i < 17; i++){

map[i][0] = 3;

map[i][16] = 3;

}

if (game\_condition == 0){

cleardevice();

game\_condition = 1;

}

initgraph(640,640);

setfillcolor(RGB(249,200,91));

fillrectangle(0,0,640,640);

setlinecolor(0);

for (int i = 40; i < 640; i += 40){

line(20,i,620,i);

}

for (int i = 40; i < 640; i+= 40){

line(i,20,i,620);

}

int countb = 0;

int countw = 0;

int flag = 1;

int outcome = 0;

\*/

//游戏部分

/\*setlinecolor(0xFFFFFF);

setfillcolor(0xFFFFFF);

fillcircle(10\*40,5\*40,10);

map[5][5] = 2;\*/

/\*

while (1){

int x,y;

if (flag == 1){

mousemeg(&x,&y);

for (int i = 1; i < 16; i++){

for (int j = 1; j < 16; j ++){

if (x == i && y == j){

setlinecolor(0);

setfillcolor(0);

fillcircle(i\*40,j\*40,10);

map[j][i] = 1;

countb++;

}

}

}

flag = 0;

outcome = check\_map();

} else {

ai(&x,&y);//死循环？

setlinecolor(0xFFFFFF);

setfillcolor(0xFFFFFF);

fillcircle(x\*40,y\*40,10);

map[x][y] = 2;

countw++;

flag = 1;

outcome = check\_map();

}

if (outcome == 1){

cleardevice();

char sen[50] = {0};

result(sen,outcome,countb);

fileinput(sen);

char black\_win[20] = "The black wins!";

settextstyle(35, 20, \_T("Consolas"));

outtextxy(80,250,black\_win);

char if\_return1[30] = "Enter F5 to continue";

outtextxy(80,320,if\_return1);

char if\_return2[20] = "Enter F2 to quit";

outtextxy(80,380,if\_return2);

flushmessage();

while (1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x74){

game\_condition = 0;

return;

} else if (m.vkcode == 0x71){

game\_condition = 1;

return;

}

}

} else if (outcome == 2){

cleardevice();

char sen[50] = {0};

result(sen,outcome,countw);

fileinput(sen);

char white\_win[20] = "The white wins!";

settextstyle(35, 20, \_T("Consolas"));

outtextxy(80,250,white\_win);

char if\_return1[30] = "Enter F5 to continue";

outtextxy(80,320,if\_return1);

char if\_return2[20] = "Enter F2 to quit";

outtextxy(80,380,if\_return2);

//

flushmessage();

while (1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x74){

game\_condition = 0;

return;

} else if (m.vkcode == 0x71){

game\_condition = 1;

return;

}

}

}

}

}

int aix = 5;

int aiy = 5;

void ai(int \*x, int \*y){

for (int i = 1; i < 16; i++){

for (int j = 1; j < 16; j++){

if (map[i][j] == 0){

\*x = j;

\*y = i;

return;

}

}

}

}

\*/

void bgm()

{

//打开音乐

mciSendString("open ./su.mp3", 0, 0, 0);//mci:多媒体设备接口 send：发送 string:字符串

//播放音乐 repeat：重复播放

mciSendString("play ./su.mp3 repeat", 0, 0, 0);

}

int fileinput(char sen[])

{

FILE \*p1=fopen("data.txt","a+");

fprintf(p1,sen);

fclose(p1);

return 0;

}

void result(char sen[],int outcome,int n){

if (outcome == 1) {

strcpy(sen, "Black wins using");

} else if (outcome == 2){

strcpy(sen, "White wins using");

}

sen[16] = ' ';

char tem[3];

strcat(sen,itoa(n,tem,10));

strcat(sen," steps at ");

time\_t t;

struct tm \*local\_time;

time(&t);

local\_time = localtime(&t);

strcat(sen,asctime(local\_time));

strcat(sen,"\n");

return;

}

void recordings(){

cleardevice();

outtextxy(260,0,"Enter F5 to continue");

outtextxy(260,20,"Enter F3 to quit");

int rank = 0;

struct node\* temp = first;

while (temp != NULL){

char sen[50] = {0};

if (temp->outcome == 1){

strcpy(sen,"Black wins using");

} else {

strcpy(sen,"White wins using");

}

sen[16] = ' ';

char tem[3] = {0};

strcat(sen,itoa(temp->steps,tem,10));

strcat(sen," steps at ");

strcat(sen,temp->time);

temp = temp->next;

outtextxy(0,(rank+2) \* 20, sen);

rank++;

}

while (1){

m = getmessage(EX\_KEY);

if (m.vkcode == 0x74){

game\_condition = 0;

return;

} else if (m.vkcode == 0x72){

game\_condition = 1;

return;

}

}

}

void record(int outcome, int steps){

first = add\_to\_record(first,outcome,steps);

}

struct node\* add\_to\_record(struct node\* list, int n, int steps){

struct node\* new\_node;

new\_node = static\_cast<node \*>(malloc(sizeof(struct node)));

if (new\_node == NULL){

printf("Error: malloc failed in add\_to\_list\n");

exit(EXIT\_FAILURE);

}

new\_node->outcome = n;

new\_node->steps = steps;

time\_t t;

struct tm \*local\_time;

time(&t);

gmtime(&t);

local\_time = localtime(&t);

strcpy(new\_node->time,asctime(local\_time));

new\_node->next = list;

return new\_node;

}