**C语言项目报告**

**项目名称：自动化考试系统**

**项目成员：沈元辰、孙语默**

**填写日期：2020.9.1**

**·一、摘要**

本学期我们最终选择了“自动化考试系统”作为C语言课程的项目。我们的预期目标是做出一个以界面为主体，通过鼠标点击进行交互的Windows应用程序，允许多名学生和老师进行注册和登录，出题、答题和查看反馈界面设置合理、简洁便利，同时兼具自动判分功能。

从项目最初开始筹划、设计实现路线、寻找解决方案到最终完成项目，我们经历了将近两个月的时间。在完成项目的过程中，我们丰富了更多的C语言知识，了解了许多解决问题的方法和技巧，也强化了编程的实战能力。最终的程序完成了预期的所用功能和效果。

作为一个线上考试平台系统，我们开发的这款程序创新点主要在于其界面明了简洁，几乎所有功能都是以鼠标点击的方式实现，更加便捷和人性化。答题也有设计好的答题卡，非常直观。同时系统具有自动保存功能，通过文件对信息进行保存，避免信息丢失。

**·二、问题描述**

受疫情影响，本学期我们只能通过线上的方式上课。而相应的，我们的许多考试也是在线上考试平台实现的。受此启发，我们组决定开发一个简易的自动化考试系统。

此系统为Windows应用程序，以简洁的界面呈现，用户可以以鼠标点击的方式进行操作。系统内分为学生端和教师端，学生和教师可分别在相应的端口进行注册和登录。

学生用户可在学生端内查看到题目并答题，教师用户可在教师端内出题、设置答案并查看学生成绩反馈。使用文件操作函数自动保存相关信息。

本系统实现了机考系统的基本功能并增加了一些更加人性化的功能和界面，可以作为线上考试系统使用。

**·三、组内分工**

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 学院 | 学号 | 工作比例 |
| 沈元辰 | 化学 | 16190108 | 50% |
| 孙语默 | 化学 |  | 50% |

具体分工：

沈元辰：答题卡开发，界面设计及界面操作函数，查看反馈函数，自动判分系统，PPT制作，项目报告撰写

孙语默：程序整体构架、教师及学生端逻辑函数，出题系统，文件函数，项目报告撰写。

**·四、分析**

此程序的目标是实现考试的自动化。首先需要为教师和学生设置不同的入口进行信息的注册和登录。同时需要对学生和教师的信息进行储存。为此我们建立了学生和教师结构体，学生结构体中储存学生用户名、密码、学生编号、作答答案、学生成绩和考试完成情况。教师结构体中储存教师用户名、密码和教师编号。同时我们还创建了题目结构体保存教师编辑的题目、选项和答案。

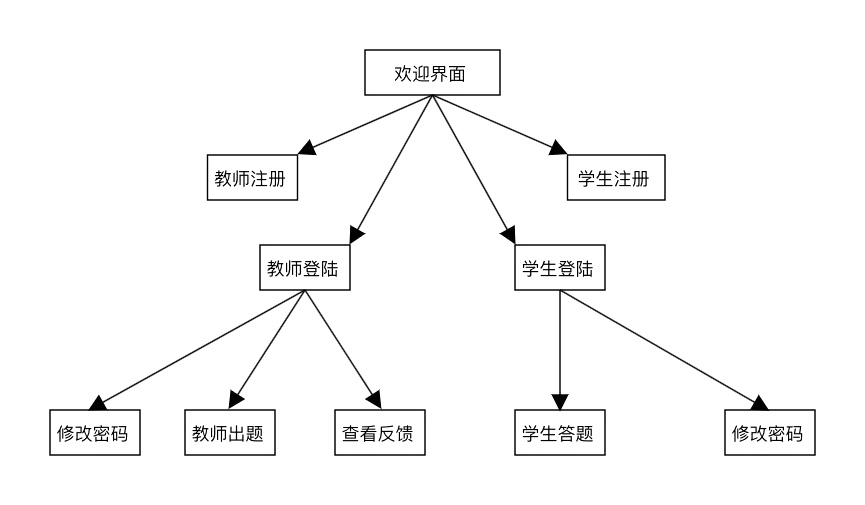
程序的主界面分为了学生端和教师端，即学生注册、学生登录、教师注册、教师登录。

为对注册的学生和教师进行统计，我们定义了全局变量nStudentNum、nTeacherNum分别代表注册的学生和教师人数。为了对登录的学生和教师进行标识，我们定义了全局变量studentnum和teachernum分别代表登录的学生和教师编号。

学生登录后进入学生菜单，包括开始答题、修改密码和返回三部分。当教师还没有出题时，点击开始答题系统会显示教师未出题。当教师已出题时，点击开始答题会同时显示题目和答题卡。学生作答完成提交后会弹出提示框显示学生的答案和分数。

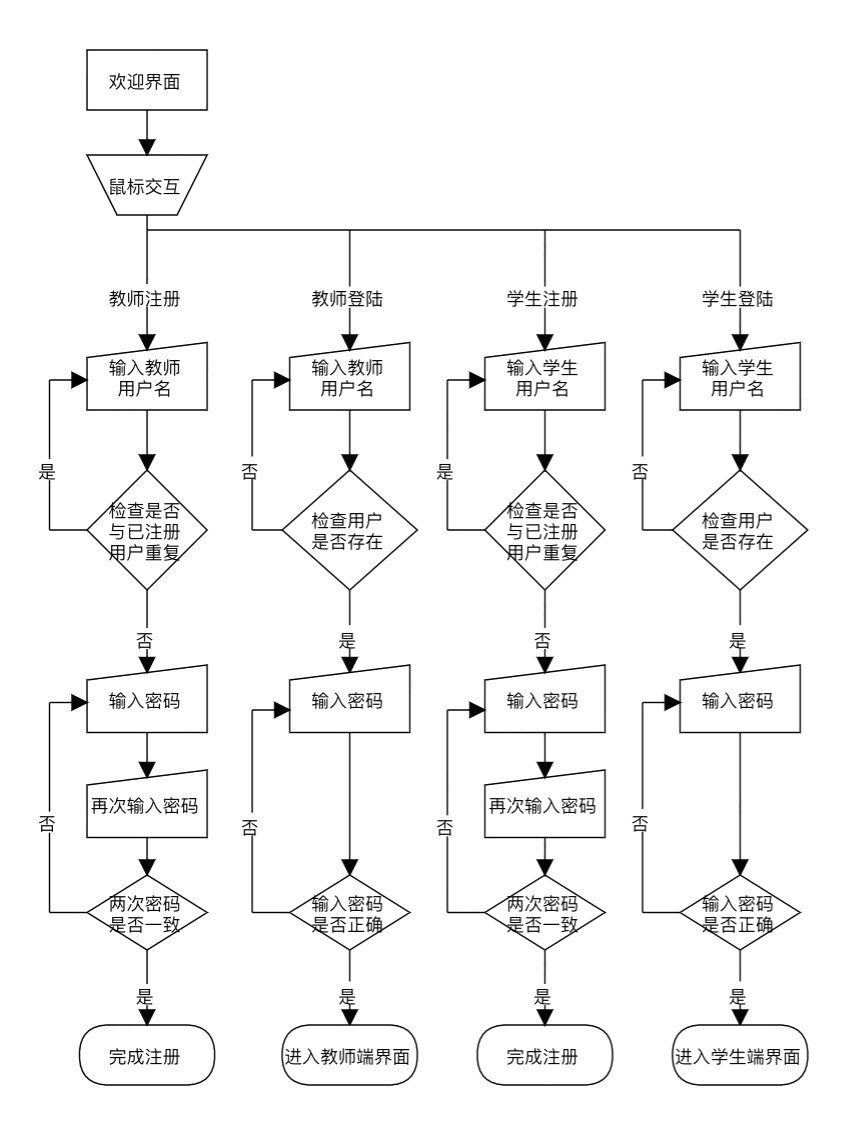
教师登录后进入教师菜单，包括开始出题、查看反馈、修改密码和返回四部分。点击开始出题进入题目编辑界面，可进行题目信息编辑，点击查看反馈进入反馈界面，可以看到学生姓名、学号和成绩。

程序中还包括文件保存函数，其功能是在文件中写入学生结构体、教师结构体、题目结构体中的全部信息和全局变量。当每一个包含有信息输入的函数运行结束后，文件保存函数会自动运行以及时保存信息。

此程序整体流程图如下

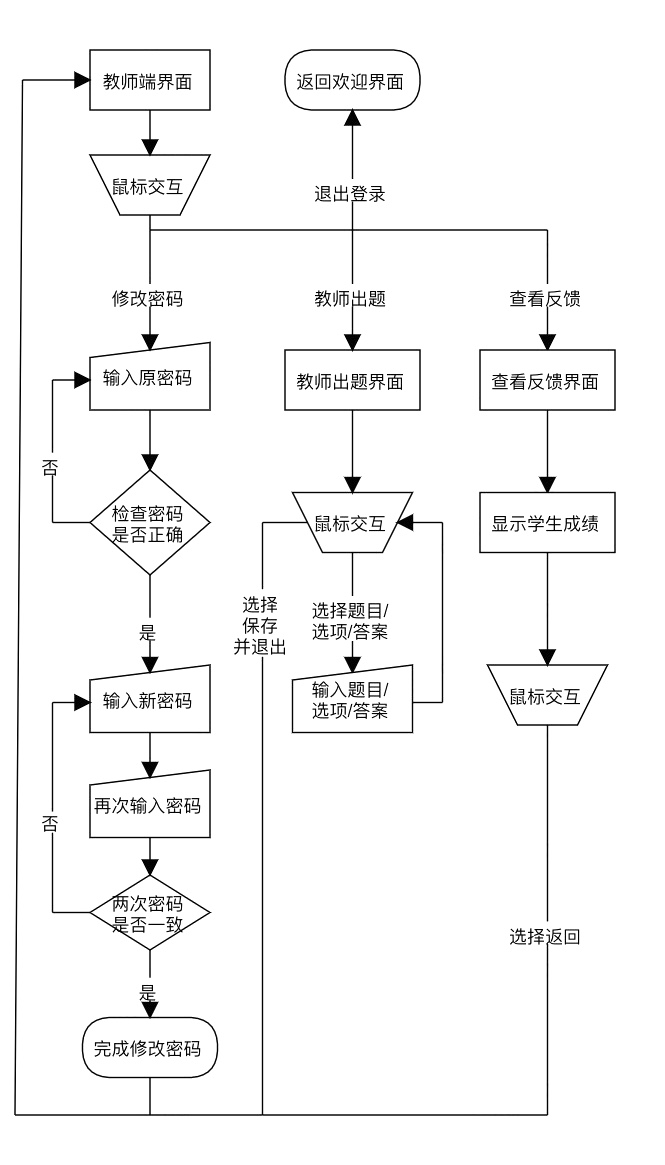
**·五、设计**

1. 欢迎界面



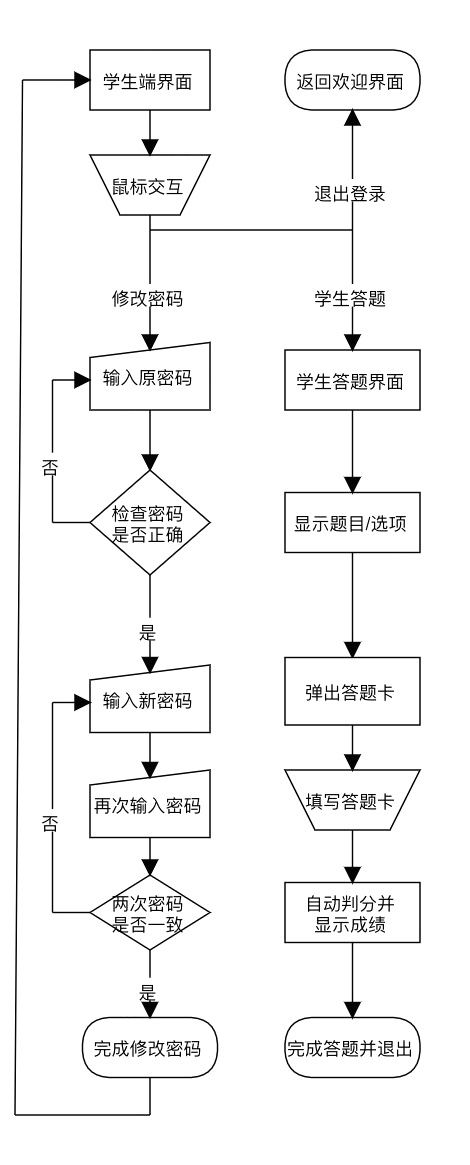
程序启动后，将会出现欢迎界面。欢迎界面带有四个按钮，点击按钮所在区域即可调用函数功能。注册功能首先让用户输入用户名，再输入两次密码进行确认，完成注册。登录功能先让用户输入用户名，再输入密码，当密码正确时登录成功，进入二级菜单，当密码错误时显示密码错误，重新输入密码。

1. 教师端



进入教师端，包括开始出题、查看反馈、修改密码和返回四部分。点击开始出题进入题目编辑界面，点击题目和选项的位置即可弹出对话框对题目信息进行编辑。点击查看反馈进入反馈界面，可以看到学生姓名、学号和成绩。点击修改密码首先输入原密码进行检验，检验通过后输入新密码两次完成密码修改。点击返回则返回到欢迎界面。

1. 学生端



进入学生端，包括开始答题、修改密码和返回三部分。当教师还没有出题时，点击开始答题系统会显示教师未出题。当教师已出题时，点击开始答题会同时显示题目和答题卡。学生作答完成提交后会弹出提示框显示学生的答案和分数。点击修改密码首先输入原密码进行检验，检验通过后输入新密码两次完成密码修改。点击返回则返回到欢迎界面。

**·六、实现**

定义学生、教师、题目结构体

//定义学生结构体数组

struct student

{

wchar\_t szName[STRLENGTH];//定义学生用户名

wchar\_t password1[NUMLENGTH];//定义学生账户密码

wchar\_t password2[NUMLENGTH];//定义学生二次输入密码

int nSerialNum;//定义学生序号(从1开始)

char stuAnswer[NUMQUEST];//定义学生答案

int stuScore;//定义学生成绩

int stuTest = 2;//定义学生完成考试情况

}stu[NUMBER];

//定义教师结构体

struct teacher

{

wchar\_t szName[STRLENGTH];//定义教师用户名

wchar\_t password1[NUMLENGTH];//定义教师账户密码

wchar\_t password2[NUMLENGTH];//定义教师二次输入密码

int nSerialNum;//定义老师序号(从1开始)

}teacher[NUMBER];

//定义题目结构体

struct question

{

wchar\_t Quest[100];//定义题目

wchar\_t ChoiceA[20];//定义A选项

wchar\_t ChoiceB[20];//定义B选项

wchar\_t ChoiceC[20];//定义C选项

wchar\_t ChoiceD[20];//定义D选项

wchar\_t teacherAnswer[1];//定义教师输入正确答案

char TeacherAnswer[1];//定义正确答案

}question[NUMQUEST];

鼠标交互函数

MOUSEMSG msg; //定义鼠标坐标

FlushMouseMsgBuffer();

while (true)

{

while (MouseHit())

{

msg = GetMouseMsg();//获取鼠标坐标

if (WM\_LBUTTONDOWN == msg.uMsg)

{

if (msg.x > 108 && msg.x < 291 && msg.y > 687 && msg.y < 790)//判断鼠标坐标

}

}

}

用户注册函数

以学生注册为例

void Stu\_Register(FILE\* fp)

{

/\*输入用户名\*/

int i;

do {

InputBox(stu[nStudentNum].szName, 10, \_T("请输入用户名"));

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

{

if (wcscmp(stu[i].szName, stu[nStudentNum].szName) == 0)

{

MessageBox(NULL, \_T("该用户已经存在，请重新输入"), \_T("创建账户"), MB\_ICONSTOP | MB\_SETFOREGROUND);

break;

}

}

if (i >= nStudentNum)

break;

} while (1);

/\*输入密码\*/

do {

InputBox(stu[nStudentNum].password1, 10, \_T("请设置密码"));

InputBox(stu[nStudentNum].password2, 10, \_T("请确认密码"));

if (wcscmp(stu[nStudentNum].password1, stu[nStudentNum].password2) != 0)

MessageBox(NULL, \_T("两次输入不一致，请重新输入"), \_T("创建账户"), MB\_ICONSTOP | MB\_SETFOREGROUND);

else

{

MessageBox(NULL, \_T("注册成功！"), \_T("创建账户"), MB\_SETFOREGROUND);

nStudentNum++;

stu[nStudentNum - 1].nSerialNum = nStudentNum;

SaveFile(fp);

break;

}

} while (1);

}

用户登录函数

以学生登录为例

void Stu\_Login(FILE\* fp,

HINSTANCE hInstance,

HINSTANCE hPrevInstance,

PSTR szCmdLine,

int iCmdShow

) {

/\*输入用户名\*/

int i;

wchar\_t username[STRLENGTH];

wchar\_t password[NUMLENGTH];

do {

InputBox(username, 10, \_T("请输入用户名"), (LPCTSTR)0, (LPCTSTR)0, 0, 0, false);

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

if (wcscmp(username, stu[i].szName) == 0)

{

studentnum = i;

break;

}

if (i >= nStudentNum)

MessageBox(NULL, \_T("该用户不存在"), \_T("学生登录"), MB\_ICONSTOP | MB\_SETFOREGROUND);

else

break;//

} while (1);

/\*输入密码\*/

do {

InputBox(password, 10, \_T("请输入密码"));

if (wcscmp(password, stu[studentnum].password1) == 0)

{

MessageBox(NULL, \_T("登录成功！"), \_T("学生登录"), MB\_SETFOREGROUND);

nStuFlag = 1;

break;

}

else

MessageBox(NULL, \_T("密码错误"), \_T("学生登录"), MB\_ICONSTOP | MB\_SETFOREGROUND);

} while (1);

OutputStu\_SecondMenu(fp,

hInstance,

hPrevInstance,

szCmdLine,

iCmdShow

);//进入学生界面

}

自动判分函数

int Scoring(int StudentNum) {

int i;

int score = 0;

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

{

if (stu[StudentNum].stuAnswer[i] == question[i].TeacherAnswer[0])

score += 10;

}

return score;

}

教师出题函数

void Teacher\_Question(FILE\* fp,

HINSTANCE hInstance,

HINSTANCE hPrevInstance,

PSTR szCmdLine,

int iCmdShow) {

loadimage(&background, L"background.jpg");//导入背景

putimage(0, 0, &background);//显示背景

wchar\_t temp[1000];

setbkmode(TRANSPARENT);

settextcolor(RGB(127, 0, 255));

settextstyle(15, 0, \_T("xxx"));

\_stprintf\_s(temp, \_T("保存并返回"));

outtextxy(1100, 850, temp);

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

\_stprintf\_s(temp, \_T("%d、%s"), i, question[i - 1].Quest);

outtextxy(10, 10 + i \* 60 - 50, temp);

\_stprintf\_s(temp, \_T("A、%s"), question[i - 1].ChoiceA);

outtextxy(10, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("B、%s"), question[i - 1].ChoiceB);

outtextxy(260, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("C、%s"), question[i - 1].ChoiceC);

outtextxy(510, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("D、%s"), question[i - 1].ChoiceD);

outtextxy(760, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("答案:%c"), question[i - 1].teacherAnswer[0]);

outtextxy(910, 10 + i \* 60 - 50, temp);

}

MOUSEMSG msg;

FlushMouseMsgBuffer();

while (true)

{

while (MouseHit())

{

msg = GetMouseMsg();

if (WM\_LBUTTONDOWN == msg.uMsg)

{

for (int k = 0; k < NUMQUEST; k++)

{

if (msg.x > 10 && msg.x < 900 && msg.y > 20 + k \* 60 && msg.y < 50 + k \* 60)

{

InputBox(question[k].Quest, 1000, \_T("请输入题目"));

loadimage(&background, L"background.jpg");//导入背景

putimage(0, 0, &background);//显示背景

\_stprintf\_s(temp, \_T("保存并返回"));

outtextxy(1100, 850, temp);

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

\_stprintf\_s(temp, \_T("%d、%s"), i, question[i - 1].Quest);

outtextxy(10, 10 + i \* 60 - 50, temp);

\_stprintf\_s(temp, \_T("A、%s"), question[i - 1].ChoiceA);

outtextxy(10, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("B、%s"), question[i - 1].ChoiceB);

outtextxy(260, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("C、%s"), question[i - 1].ChoiceC);

outtextxy(510, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("D、%s"), question[i - 1].ChoiceD);

outtextxy(760, 10 + i \* 60 - 20, temp);

\_stprintf\_s(temp, \_T("答案:%c"), question[i - 1].teacherAnswer[0]);

outtextxy(910, 10 + i \* 60 - 50, temp);

}

}

else if……

else if (msg.x > 1090 && msg.x < 1150 && msg.y > 830 && msg.y < 870)

{

for (int n = 0; n < NUMBER; n++) {

stu[n].stuTest = 0;

}

SaveFile(fp);

Sleep(100);

OutputTeacher\_SecondMenu(fp,

hInstance,

hPrevInstance,

szCmdLine,

iCmdShow

);

}

}

}

}

}

}

文件保存函数

void SaveFile(FILE\* fp)

{

errno\_t err;

err = fopen\_s(&fp, "alldate.dat", "wb");

if (err != 0) {

printf("失败了");

}

else {

//向文件输出全局变量

fwrite(&nStudentNum, 4, 1, fp);

fwrite(&nTeacherNum, 4, 1, fp);

fwrite(&nStuFlag, 4, 1, fp);

fwrite(&nTeacherFlag, 4, 1, fp);

fwrite(&studentnum, 4, 1, fp);

fwrite(&teachernum, 4, 1, fp);

//输出学生结构体数据

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

fwrite(&stu[i], sizeof(struct student), 1, fp);

//输出教师结构体数组

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

fwrite(&teacher[i], sizeof(struct teacher), 1, fp);

//输出题目结构体数据

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

fwrite(&question[i], sizeof(struct question), 1, fp);

}

fclose(fp);

}

答题卡函数

int answer(HINSTANCE hInstance, HINSTANCE hPrevInstance, PSTR szCmdLine, int iCmdShow, FILE\* fp)

{

static TCHAR szClassName[] = TEXT("Win32Demo");

HWND hwnd;

MSG msg;

WNDCLASS wndclass;

hInst = hInstance;

wndclass.style = CS\_HREDRAW | CS\_VREDRAW;

wndclass.lpfnWndProc = WndProc;

wndclass.cbClsExtra = 0;

wndclass.cbWndExtra = 0;

wndclass.hInstance = hInstance;

wndclass.hIcon = LoadIcon(NULL, IDI\_APPLICATION);

wndclass.hCursor = LoadCursor(NULL, IDC\_ARROW);

wndclass.hbrBackground = (HBRUSH)GetSysColorBrush(COLOR\_BTNFACE);

wndclass.lpszMenuName = NULL;

wndclass.lpszClassName = szClassName;

RegisterClass(&wndclass);

hwnd = CreateWindow(

szClassName,

TEXT("答题卡"),

WS\_OVERLAPPEDWINDOW,

CW\_USEDEFAULT,

CW\_USEDEFAULT,

800,

500,

NULL,

NULL,

hInstance,

NULL

);

ShowWindow(hwnd, iCmdShow);

UpdateWindow(hwnd);

while (GetMessage(&msg, NULL, 0, 0)) {

TranslateMessage(&msg);

DispatchMessage(&msg);

}

return msg.wParam;

}

LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam) {

int wmId, wmEvent;

PAINTSTRUCT ps;

HDC hdc;

// LPCWSTR La1 = (LPCWSTR)(question[1].Quest);

static TCHAR szBufquestion1[1];

……

static TCHAR szBufSubmit[100];

static TCHAR szBufTmp[10];

static HFONT hFont;

static HWND labquestion1;

static HWND radio1A;

static HWND radio1B;

static HWND radio1C;

static HWND radio1D;

……

static HWND btnSubmit;

switch (message) {

case WM\_CREATE:

hFont = CreateFont(

-14, -7, 0, 0, 400,

FALSE, FALSE, FALSE, DEFAULT\_CHARSET,

OUT\_CHARACTER\_PRECIS, CLIP\_CHARACTER\_PRECIS, DEFAULT\_QUALITY,

FF\_DONTCARE, TEXT("微软雅黑")

);

labquestion1 = CreateWindow(

TEXT("static"), TEXT("question1："),

WS\_CHILD | WS\_VISIBLE | SS\_CENTERIMAGE | SS\_RIGHT,

25, 10, 80, 26,

hWnd, (HMENU)1, hInst, NULL

);

radio1A = CreateWindow(

TEXT("button"), TEXT("A"),

WS\_CHILD | WS\_VISIBLE | BS\_LEFT | BS\_AUTORADIOBUTTON | WS\_GROUP,

105, 10, 50, 26,

hWnd, (HMENU)2, hInst, NULL

);

radio1B = CreateWindow(

TEXT("button"), TEXT("B"),

WS\_CHILD | WS\_VISIBLE | BS\_LEFT | BS\_AUTORADIOBUTTON,

165, 10, 50, 26,

hWnd, (HMENU)3, hInst, NULL

);

radio1C = CreateWindow(

TEXT("button"), TEXT("C"),

WS\_CHILD | WS\_VISIBLE | BS\_LEFT | BS\_AUTORADIOBUTTON,

225, 10, 50, 26,

hWnd, (HMENU)4, hInst, NULL

);

radio1D = CreateWindow(

TEXT("button"), TEXT("D"),

WS\_CHILD | WS\_VISIBLE | BS\_LEFT | BS\_AUTORADIOBUTTON,

285, 10, 50, 26,

hWnd, (HMENU)5, hInst, NULL

);

……

btnSubmit = CreateWindow(TEXT("button"), TEXT("提交"),

WS\_CHILD | WS\_VISIBLE | WS\_BORDER | BS\_FLAT,

95, 350, 200, 36,

hWnd, (HMENU)51, hInst, NULL

);

SendMessage(labquestion1, WM\_SETFONT, (WPARAM)hFont, NULL);

SendMessage(radio1A, WM\_SETFONT, (WPARAM)hFont, NULL);

SendMessage(radio1B, WM\_SETFONT, (WPARAM)hFont, NULL);

SendMessage(radio1C, WM\_SETFONT, (WPARAM)hFont, NULL);

SendMessage(radio1D, WM\_SETFONT, (WPARAM)hFont, NULL);

……

SendMessage(btnSubmit, WM\_SETFONT, (WPARAM)hFont, NULL);

break;

case WM\_COMMAND:

wmId = LOWORD(wParam);

wmEvent = HIWORD(wParam);

if (wmEvent == BN\_CLICKED) {

switch (wmId) {

case 2:

case 3:

case 4:

case 5:

GetWindowText((HWND)lParam, szBufquestion1, 10);

stu[studentnum].stuAnswer[0] = \*(TCHARToChar(szBufquestion1));

break;

……

case 51:

stu[studentnum].stuScore = Scoring(studentnum);

/\*因不可调用SaveFile函数，故直接保存\*/

……

/\*保存完成\*/

wsprintf(szBufSubmit,

TEXT("你的第一题答案：%s\n你的第二题答案：%s\n你的第三题答案：%s\n你的第四题答案：%s\n你的第五题答案：%s\n你的第六题答案：%s\n你的第七题答案：%s\n你的第八题答案：%s\n你的第九题答案：%s\n你的第十题答案：%s\n你的成绩：%d"),

szBufquestion1, szBufquestion2, szBufquestion3, szBufquestion4, szBufquestion5, szBufquestion6, szBufquestion7, szBufquestion8, szBufquestion9, szBufquestion10, stu[studentnum].stuScore);

学生答题函数

学生答题函数调用了题目显示函数和答题卡函数，首先判断出题状态。在已出题的情况下按顺序显示题目，并弹出答题卡，通过按钮获取点击的选项。最后通过判分函数判分显示成绩。

void Stu\_Answer(FILE\* fp,

HINSTANCE hInstance,

HINSTANCE hPrevInstance,

PSTR szCmdLine,

int iCmdShow

) {

if (stu[studentnum].stuTest == 1)

MessageBox(NULL, \_T("您已完成考试！"), \_T("学生答题"), MB\_SETFOREGROUND);

else if (stu[studentnum].stuTest == 2)

MessageBox(NULL, \_T("教师未出题！"), \_T("学生答题"), MB\_SETFOREGROUND);

else

{

Stu\_Question(fp);

answer(hInstance, hPrevInstance, szCmdLine, iCmdShow, fp);

stu[studentnum].stuTest = 1;

}

SaveFile(fp);

}

**·七、测试**

欢迎界面

教师端

出题界面

查看反馈界面

学生端

答题界面