

学生姓名	学生学号	实验课成绩
冯钢果	0121610870910	
郑佳浩	0121610870929	

武汉理工大学

学 生 大 作 业 报 告 书

课 程 名 称 可视化编程

开 课 学 院 计算机科学与技术学院

指 导 老 师 胡伦

学 生 姓 名 冯钢果、郑佳浩

专 业 班 级 软件 1604 班

2017 — 2018 学 年 第 二 学 期

目录

一、项目背景和意义.....	3
1. 项目背景：	3
2. 项目意义：	3
二、系统设计.....	3
1. 明确实验需求和思路分析.....	3
2. 系统总体设计结构图.....	4
3. 功能模块划分和模块关系.....	4
4. 相关技术和分工.....	5
三、系统实现.....	5
1. 程序流程图	5
2. 成员变量和文件列表.....	6
3. 消息通知代码和处理函数.....	9
4. 运行界面.....	14
四、程序调试分析	14
1. 程序调试分析	14
2. 功能测试.....	15
五、总结.....	19

小组成员:

软件 1604 班冯钢果

软件 1404 班郑佳浩

一、项目背景和意义

1. 项目背景:

通过调查显示,大部分消费者都会使用电脑记事本功能。随着电脑记事本功能的不断加强与完善,电脑记事本对于人们的意义,已不仅仅像一般记事簿一样简单记录,而是向着更个性化人性化的方向发展。记事本在生活和工作也日益凸显出它的重要性,它能更好地帮助人们管理自己的工作和生活,极大的方便了人们的生活。用过电脑的人,几乎用过记事本,用记事本干嘛呢?顾名思义,是用来记叙事情的,但往往我们不是用来记录你今天做了什么事,不是当日记本来使用的,而是用它来保存一些文字,这些文字可以是一部小说,可以是一些代码:包括 C、java、c++、html、汇编源程序书写的;可以是一些资料文件,包括你得到的一些知识内容,把它记录起来,等等这些功能,当然,使用这个软件我们也能进行日程规划和记录自己做过的,但 Windows 系统默认记事本不能直观地表示自己的日程规划。

因此,综合考虑以上需求,我们可以将设计一个软件——日程规划软件,使用此软件,我们直观方便地进行记录自己将要做的事、自己做过的重大的事和一些重要的信息,并在实际应用中,可以到处自己的日程规划分享给重要的人,更加限度以满足用户的需求。

2. 项目意义:

本项目名称定位——日程规划项目,我们使用 VC++进行软件的界面设计和功能设计,因此,本项目具有以下意义:

- 1) 提升自己的编程能力;
- 2) 通过实验项目,培养自己的团队合作能力里;
- 3) 对 VC++有更深刻认识,是自己学会学以致用,理解 VC++函数的原理和应用,从而提高自己对 VC++的兴趣;
- 4) 最重要的是本项目的目的——开发出一款简单易用的日程规划软件,满足用户平时工作、学习和娱乐的规划。

二、系统设计

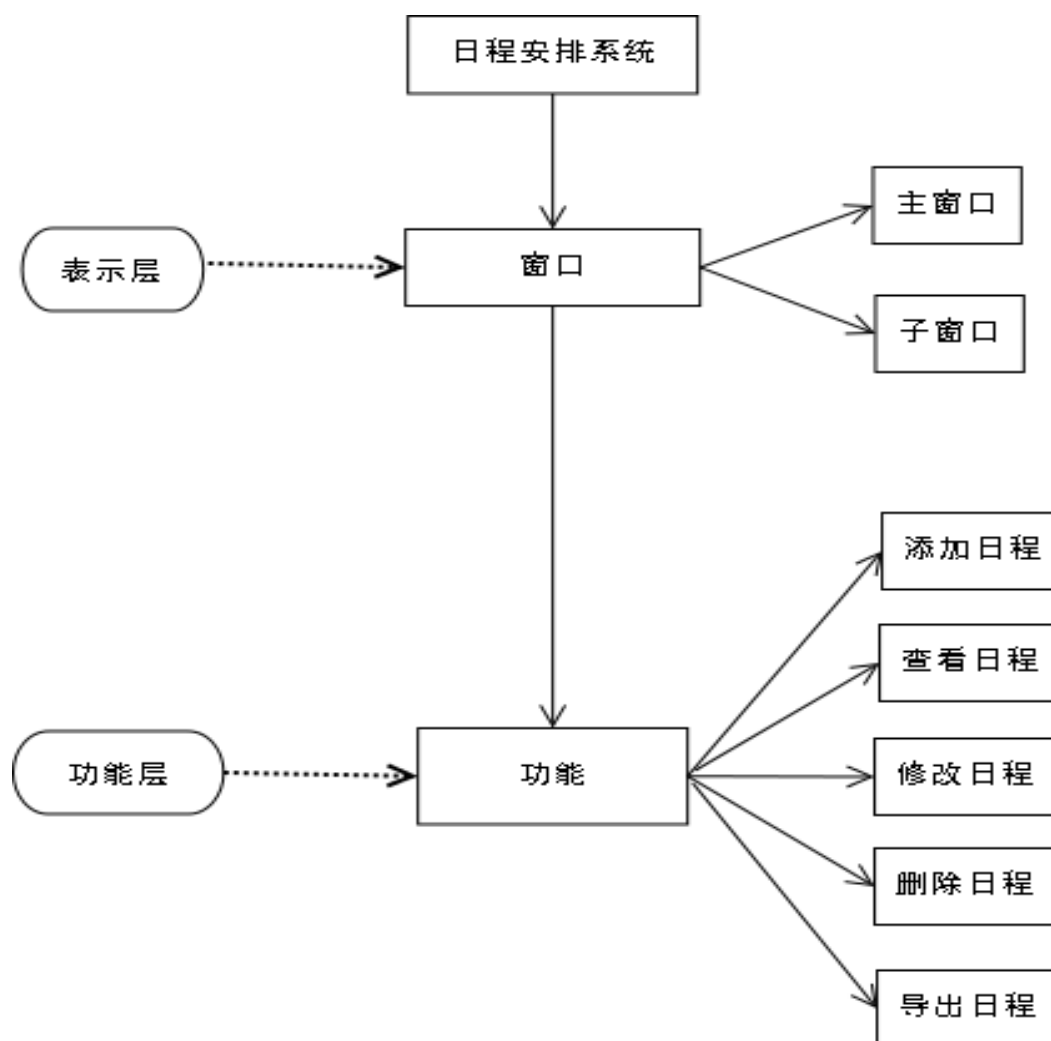
1. 明确实验需求和思路分析

系统需求:使用 VC++开发出一款 Windows 平台的、方便易用的、直观的并且面向免费用户的日程规划软件。

思路分析:想到日程规划我们都能想到:首先选择功能——记录事件、查看事件、修改事件、删除事件,选择事件段(开始时间 T_1 、结束事件 T_2 , $T_1 \leq T_2$),然后输入事件标题,接着输入事件的详细描述,点击确定按钮即可加入事件,这是我们能想到的,当然导出自己的日程后,我们也可以将自己的日程规划分享给重要的人。

2. 系统总体设计结构图

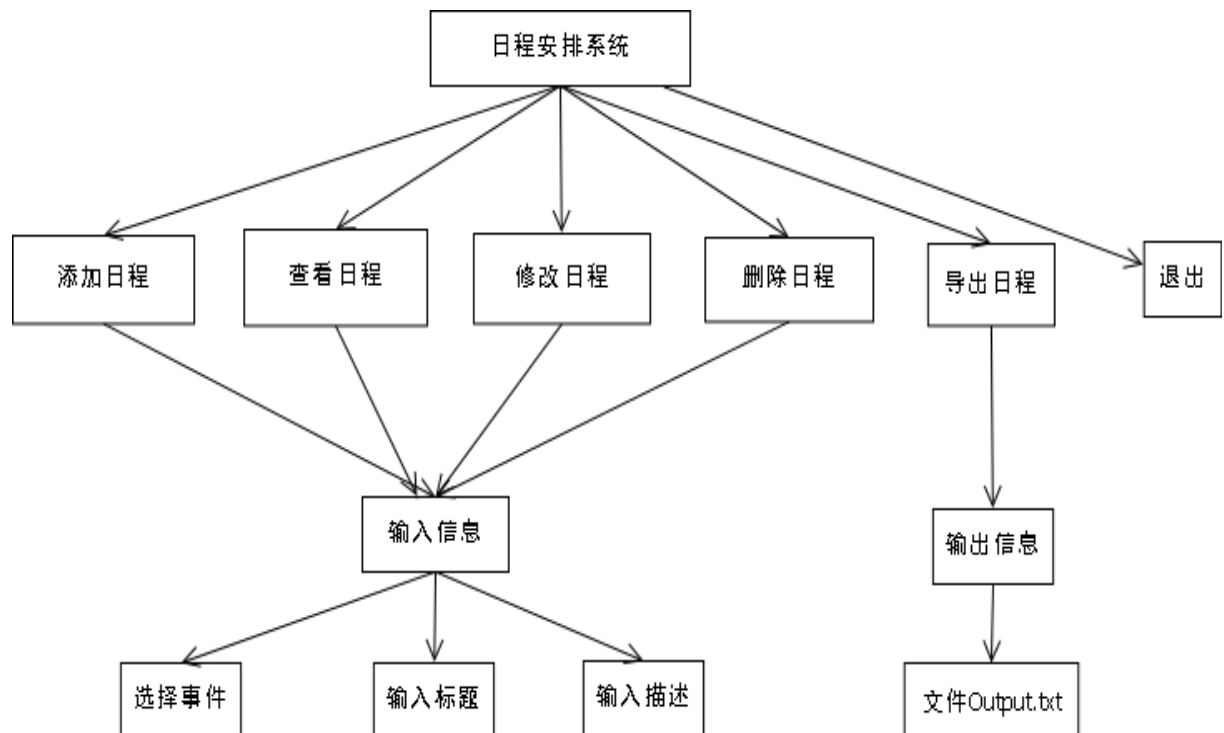
说明：对于日程规划系统软件设计我们可以考虑使用软件工程的调用返回风格——层次结构，将系统功能分为两个部分：窗口部分、功能实现部分；窗口部分包括主窗口、子窗口；功能实现部分包括添加日程、查看日程、修改日程、删除日程和导出日程。因此，应用以上的设计思想，可画出系统总体设计结构图如下：



3. 功能模块划分和模块关系

以上个部分的日程规划系统总体结构设计图作为设计功能模块的标准，因此我们可以主要基于系统的功能，将系统功能划分为6个部分：添加日程、查看日程、修改日程、删除日程、导出日程和退出程序。其中为了明确各个功能之间的联系，可以简单的将前4个功能提取相同点——输入日程相关信息如选择事件、输入标题、输入描述；可以将导出日程与输出到文件相关联；退出功能直接通过发送向窗体信息退出即可。

以下是功能模块划分和模块关系的示意图：



4. 相关技术和分工

使用工具：Microsoft Visual Studio 2010，HTML 编辑器（因为设计界面时主要用到了界面引用 HTML 当作背景，便于设计出更加人性化的背景，从而避免了 VC++ 中复杂的操作）

参考资料：《Visual C++ 面向对象与可视化程序设计》

使用技术：vc++ 的基本技术、MFC 的程序设计；

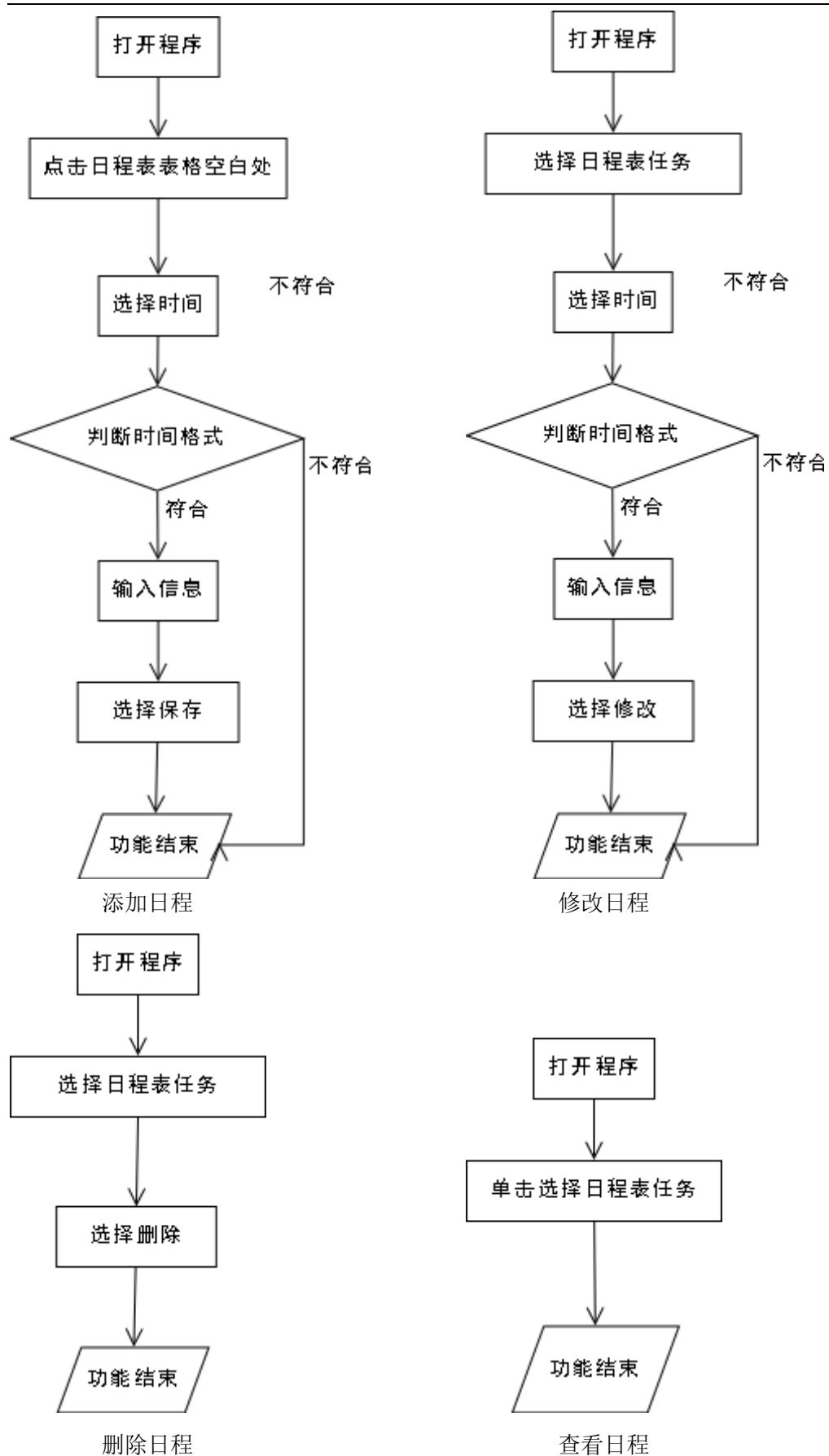
分工说明：

姓名	冯钢果	郑佳浩
任务	写实验代码、 测试程序、 总结程序功能、 绘制系统设计图	分析需求、 写实验报告、 提交改进建议

三、 系统实现

1. 程序流程图

以下是程序的添加日程、修改日程、查看日程、删除日程的流程图,最主要的是添加日程。流程图如下：



2. 成员变量和文件列表

详细的工程变量记录如下：

文件名	类名	变量名	功能
SchedulerDlg.h	CSchedulerDlg	CArray <CMyscheduler> m_scheduler_array	存储已保存的信息
		COleVariant m_month	变量的月份
		COleVariant m_day	变量的天数
		HICON m_hIcon	窗口图标
MyScheduler.h	CMyscheduler	int m_iMonth;	当前月数
		int m_iDay;	当前天数
		int m_iYear;	当前年数
		int m_iStartHour;	开始的小时
		int m_iStartMinutes;	开始的分钟
		int m_iEndHour;	结束的小时
		int m_iEndMinutes;	结束的分钟
		int m_iId;	时间的 ID
		CString m_szTitle	事件标题
		CString m_szDescription	事件描述
InputNewEventDlg.h	CEntryInfoDlg	BOOL m_fUpdateScheduleDisplay;	判断修改后更新事件显示
		BOOL m_fDeleteSchedule;	删除标志
		BOOL m_fModifySchedule	修改标志
		BOOL m_fClearSchedule;	清除标志
		BOOL m_fCancelSchedule;	取消标志
		BOOL m_fVisibleButtons;	按钮可视化标志
		BOOL m_fVisibleDefaultButtons;	默认按钮标志
		CString m_szStartTime;	开始时间字符串
		CString m_szEndTime;	结束事件字符串
		CString m_szTitle;	事件标题
		CString m_szDescription;	事件描述
		CString m_szStartTimeHour;	事件开始小时信息
		CString m_szStartTimeMinute;	事件开始分钟信息
		CString m_szEndTimeHour;	事件结束小时信息
		CString m_szEndTimeMinute;	事件结束分钟信息
		int m_iStartHour;	开始小时数 暂存变量
		int m_iEndHour	结束小时暂 存变量

文件列表及其用途：

文件名	文件用途
Scheduler.vcproj	应用程序向导生成的 VC++ 项目的主要项目文件，包含有关生成该文件的 Visual C++ 版本的信息、有关使用应用程序向导选择的平台、配置和项目功能的信息；
Scheduler.h	应用程序的主头文件，包含其他项目特定的头文件（包括 Resource.h）并声明 CSchedulerApp 应用程序类；
Scheduler.cpp	包含应用程序类 CSchedulerApp 的主应用程序源文件；
Scheduler.rc	该程序使用的所有 Microsoft Windows 资源的列表；包括存储在 RES 子目录中的图标，位图和光标；
Scheduler.ico	这是一个图标文件，用作应用程序的图标；
Scheduler.rc2	包含不由 Microsoft Visual C++ 编辑的资源；
SchedulerDlg.h	对话框头文件，定义了你的应用程序主对话框的行为；
SchedulerDlg.cpp	对话框源文件，定义了你的应用程序主对话框的行为；
StdAfx.h,	构建一个预编译头、预编译类型文件；
StdAfx.cpp	构建一个预编译头、预编译类型文件；
Resource.h	定义资源 ID；
EntryInfoDlg.htm	新建任务对话框背景；
Scheduler.htm	主窗体对话框背景；
Output.txt	保存导出的的日程信息；
Scheduler.sln	VC++2010 的解决方案。

以上是文件和变量详细列表，文件主要集中在主窗体和子窗体的函数中。

3. 消息通知代码和处理函数

```
//文件名: InputNewEventDlg.cpp
//类名: CEntryInfoDlg
//该函数功能: 点击保存按钮时检查输入信息是否有误
HRESULT CEntryInfoDlg::OnButtonSave(IHTMLElement *pElement)
{
    UpdateData(true);
    if (CheckValidTitle() && CheckValidDescription() && CheckStartEndTime())
    {
        ASSERT(m_szTitle);
        ASSERT(m_szDescription.GetLength() <= 200);
        m_fUpdateScheduleDisplay = true;
        OnOK();
    }
    else
        m_fUpdateScheduleDisplay = false;
    return S_OK;
}
```

```
//文件名: InputNewEventDlg.cpp
//类名: CEntryInfoDlg
//该函数功能: 设置日程时取消保存而退出
HRESULT CEntryInfoDlg::OnButtonCancel(IHTMLElement* /*pElement*/)
{
    m_fCancelSchedule = true;
    OnCancel();
    return S_OK;
}
```

```
//文件名: InputNewEventDlg.cpp
//类名: CEntryInfoDlg
//该函数功能: 设置日程时, 清除已设置的信息
HRESULT CEntryInfoDlg::OnButtonClear(IHTMLElement* /*pElement*/)
{
    m_fClearSchedule = true;
    m_szTitle.Empty();
    m_szDescription.Empty();
    UpdateData(false);
    return S_OK;
}
```

```
//文件名: InputNewEventDlg.cpp
//类名: CEntryInfoDlg
//该函数功能: 修改日程, 并判断日程是否符合格式

HRESULT CEntryInfoDlg::OnButtonModify(IHTMLInputElement*)
{
    UpdateData(true);
    if (CheckValidTitle() && CheckValidDescription() && CheckStartEndTime()){
        ASSERT(m_szTitle);
        ASSERT(m_szDescription.GetLength() <= 200);
        m_fModifySchedule = true;
        OnOK();
    }
    else
        m_fModifySchedule = false;
    return S_OK;
}
```

```
//文件名: InputNewEventDlg.cpp
//类名: CEntryInfoDlg
//该函数功能: 删除日程
HRESULT CEntryInfoDlg::OnButtonDelete(IHTMLInputElement*)
{
    m_fDeleteSchedule = true;
    OnCancel();
    return S_OK;
}
```

```
//文件名: InputNewEventDlg.cpp
//类名: CEntryInfoDlg
//该函数功能: 使鼠标经过按钮时, 上面字样能够有变化
HRESULT CEntryInfoDlg::OnMouseOverButtons(IHTMLInputElement* pElement)
{
    CComPtr<IHTMLStyle> phtmlStyle;
    COleVariant varColor;
    pElement->get_style(&phtmlStyle);
    ASSERT(phtmlStyle);
    if (phtmlStyle)
    {
        varColor = AQUA;
        phtmlStyle->put_color(varColor);
        phtmlStyle->put_cursor(L"hand");
    }
    return NULL;
}
```

```
//文件名: InputNewEventDlg.cpp
//类名: CEntryInfoDlg
//该函数功能: 使鼠标经过按钮后 (与上面函数不同), 上面字样能够有变化
HRESULT CEntryInfoDlg::OnMouseOutButtons(IHTMLElement* pElement)
{
    CComPtr<IHTMLStyle> phtmlStyle;
    COleVariant varColor;
    pElement->get_style(&phtmlStyle);
    ASSERT(phtmlStyle);
    if (phtmlStyle)
    {
        varColor = YELLOW;
        phtmlStyle->put_color(varColor);
    }
    return NULL;
}
```

```
//文件名: SchedulerDlg.cpp
//类名: CSchedulerDlg
//该函数功能: 导出日程到文件
void CSchedulerDlg::OnSave(){
    ofstream fout;
    fout.open("Output.txt");
    fout << "日期\t开始 - 结束\t标题\t描述\n";
    int cnt = m_scheduler_array.GetSize();
    CMyScheduler p;
    for(int i = 0; i < cnt; i++)
    {
        p = m_scheduler_array.ElementAt(i);
        fout << p.m_iYear << "." << p.m_iMonth << "." << p.m_iDay << "\t";
        fout << p.m_iStartHour << ":" << p.m_iStartMinutes << " - ";
        fout << p.m_iEndHour << ":" << p.m_iEndMinutes << "\t";
        fout << p.m_szTitle << "\t" << p.m_szDescription << endl;
    }
}
```

```
//文件名: SchedulerDlg.cpp
//类名: CSchedulerDlg
//该函数功能: 退出程序
void CSchedulerDlg::OnExit()
{
    CDialog::SendMessage(WM_CLOSE);
}
```

```
//文件名: SchedulerDlg.cpp
//类名: CSchedulerDlg
//该函数功能: 新建事件日程
HRESULT CSchedulerDlg::OnClickNewElements(IHTMLElement *pElement)
{
    BSTR getbstr = NULL;
    CEntryInfoDlg InfoDlg;
    CMyScheduler scheduler;
    CString szStr;
    CComPtr<IHTMLElement> phtmlElement;
    int iDivID, iScheduleInfo;
    pElement->get_parentElement(&phtmlElement);
    ASSERT(phtmlElement);
    if (phtmlElement)
        phtmlElement->get_id(&getbstr);
    ASSERT(getbstr);
    szStr = getbstr;
    InfoDlg.m_szStartTime = GetStartTimeFormatted(szStr);
    pElement->get_id(&getbstr);
    ASSERT(getbstr);
    szStr = getbstr;
    iDivID = atoi(szStr);
    for (iScheduleInfo=0; iScheduleInfo < m_scheduler_array.GetSize(); iScheduleInfo++){
        scheduler = m_scheduler_array.GetAt(iScheduleInfo);
        if (scheduler.m_iId == iDivID)
            break;
    }
    ASSERT(scheduler.m_iId==iDivID);
    InfoDlg.m_szTitle = scheduler.m_szTitle;
    InfoDlg.m_szDescription = scheduler.m_szDescription;
    InfoDlg.m_fVisibleButtons = true;
    InfoDlg.m_fVisibleDefaultButtons = false;
    InfoDlg.DoModal();
    if (InfoDlg.m_fDeleteSchedule) {
        m_scheduler_array.RemoveAt(iScheduleInfo);
        OnDeleteScheduleDisplay(scheduler);
    }
    else if (InfoDlg.m_fModifySchedule) //If the schedule needs to be modified{
        RemoveInfoFromTable();
        OnModifyScheduleDisplay(m_scheduler_array.ElementAt(iScheduleInfo),
InfoDlg);
        DateSchedules();
    }
    else
        InfoDlg.SetDefaultSettings();
    return S_OK;
}
```

```
//文件名: SchedulerDlg.cpp
//类名: CSchedulerDlg
//该函数功能: 清除所有日程
HRESULT CSchedulerDlg::OnButtonClearAll(IHTMLElement*)
{
    RemoveInfoFromTable();
    m_scheduler_array.RemoveAll();
    SetDefaultProperties();
    return S_OK;
}
```

```
//文件名: SchedulerDlg.cpp
//类名: CSchedulerDlg
//该函数功能: 显示“关于”窗口
void CSchedulerDlg::OnHelpAbout(void)
{
    CAboutDlg dlg;
    dlg.DoModal();
}
```

```
//文件名: SchedulerDlg.cpp
//类名: CSchedulerDlg
//该函数功能: 更新窗口内事件信息显示
void CSchedulerDlg::OnUpdateScheduleDisplay(const CEntryInfoDlg& InfoDlg)
{
    CMyScheduler schedulerEntry;

    schedulerEntry.m_iMonth = m_month.iVal;
    schedulerEntry.m_iDay = m_day.iVal;
    schedulerEntry.m_iYear = m_year.iVal;
    schedulerEntry.m_szTitle = InfoDlg.m_szTitle;
    schedulerEntry.m_szDescription = InfoDlg.m_szDescription;
    schedulerEntry.m_iStartHour = InfoDlg.m_iStartHour;
    schedulerEntry.m_iEndHour = InfoDlg.m_iEndHour;
    schedulerEntry.m_iStartMinutes = atoi(InfoDlg.m_szStartTimeMinute);
    schedulerEntry.m_iEndMinutes = atoi(InfoDlg.m_szEndTimeMinute);
    schedulerEntry.m_iId = m_iLastDivID++;

    m_scheduler_array.Add(schedulerEntry);

    RemoveInfoFromTable();

    DateSchedules();
}
```

4. 运行界面

打开软件，然后运行添加事件（点击事件列表空白处即可完成添加），添加几个事件后程序如下图：



四、 程序调试分析

1. 程序调试分析

- 1) 再次打开运行软件，发现上次添加任务已经在任务列表中；
- 2) 先测试“添加日程”功能，先单击事件列表空白处，然后会弹出“新建任务”的对话框，然后分别选择时间（11:30-6:00）、输入标题(Eat)、输入描述（Go out!），单击保存；
- 3) 再测试“修改日程”功能，先单击已经存在的任务标题“3”，再在弹出的对话框中修改标题为“ps”，单击修改即可；
- 4) 再测试“删除日程”功能，先单击已经存在的任务标题“1”，再在弹出的对话框中单击删除即可；
- 5) 再测试“查看日程”功能，单击已经存在的任务标题“Eat”即可查看相关信息；
- 6) 测试“导出日程”功能，单击“选项”下拉菜单中的“保存”即可，保存文件在运行目录；
- 7) 测试“关于”功能，单击“帮助”下拉菜单中的“关于”即可；
- 8) 测试“退出”功能，单击“选项”下拉菜单中的“退出”即可。

2. 功能测试

新建任务

开始

11:30 am

结束

6:00 pm

标题

Eat

详细描述

Go out!

保存

取消

清除

新建任务
窗口

添加任务后效果

任务管理助手

选项 帮助

任务表

一键清空

时间	事件
8:00 am	1: 8:00 am - 6:00 pm (Friday, June 15, 2018)
8:30 am	2: 8:30 am - 6:00 pm (Friday, June 15, 2018)
9:00 am	3: 9:00 am - 6:00 pm (Friday, June 15, 2018)
9:30 am	4: 9:30 am - 6:00 pm (Friday, June 15, 2018)
10:00 am	
10:30 am	
11:00 am	
11:30 am	
12:00 pm	Eat: 12:00 pm - 4:30 pm (Friday, June 15, 2018)
12:30 pm	
1:00 pm	
1:30 pm	
2:00 pm	
2:30 pm	
3:00 pm	
3:30 pm	
4:00 pm	
4:30 pm	
5:00 pm	
5:30 pm	
6:00 pm	

新建任务

开始

9:00 am

结束

6:00 pm

标题

ps

详细描述

44

取消

修改

删除

修改任务功能

修改日程后效果

任务管理助手

选项 帮助

任务表

一键清空

时间	事件
8:00 am	1: 8:00 am - 6:00 pm (Friday, June 15, 2018)
8:30 am	2: 8:30 am - 6:00 pm (Friday, June 15, 2018)
9:00 am	ps: 9:00 am - 6:00 pm (Friday, June 15, 2018)
9:30 am	4: 9:30 am - 6:00 pm (Friday, June 15, 2018)
10:00 am	
10:30 am	
11:00 am	
11:30 am	
12:00 pm	Eat: 12:00 pm - 4:30 pm (Friday, June 15, 2018)
12:30 pm	
1:00 pm	
1:30 pm	
2:00 pm	
2:30 pm	
3:00 pm	
3:30 pm	
4:00 pm	
4:30 pm	
5:00 pm	
5:30 pm	
6:00 pm	

新建任务

开始

8:00 am

▼

结束

6:00 pm

▼

标题

1

详细描述

First

取消

修改

删除

删除任务功能

删除日程后效果

任务管理助手

选项 帮助

任务表

一键清空

单击可以清空日程

时间	事件
8:00 am	
8:30 am	2: 8:30 am - 6:00 pm (Friday, June 15, 2018)
9:00 am	ps: 9:00 am - 6:00 pm (Friday, June 15, 2018)
9:30 am	4: 9:30 am - 6:00 pm (Friday, June 15, 2018)
10:00 am	
10:30 am	
11:00 am	
11:30 am	
12:00 pm	Eat: 12:00 pm - 4:30 pm (Friday, June 15, 2018)
12:30 pm	
1:00 pm	
1:30 pm	
2:00 pm	
2:30 pm	
3:00 pm	
3:30 pm	
4:00 pm	
4:30 pm	
5:00 pm	
5:30 pm	
6:00 pm	

新建任务

开始

12:00 pm

结束

6:00 pm

标题

Eat

详细描述

Go out!

取消

修改

删除

查看日程

关于



提醒:保存文件在工程目录

制作人:冯钢果、郑佳浩

退出

关于窗口

导出日程文件
截图

Output.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

日期	开始 - 结束	标题	描述
2018.6.14	11:0 - 18:0	1	first2
2018.6.15	8:30 - 18:0	2	Second
2018.6.15	9:0 - 18:0	ps	44
2018.6.15	9:30 - 18:0	4	Go out!
2018.6.15	12:0 - 16:30	Eat	Go out!

五、 总结

1. 日程规划软件完成情况:

- 1) 打开软件后可以查看上回添加日程;
- 2) 可以对日程进行以下功能: 添加日程、修改日程、删除日程、查看日程、导出日程;
- 3) 可以对所有日程进行清空。

2. 本次使用技术的总结:

- 1) 在开发过程中, 由于是面向桌面使用用户, 也因此牵涉到 MFC 的应用, 其中使用了 MFC 所设计界面;
- 2) 在设计文件保存中, 使用了 c++ 的 ifstream 中的 ofstream 定义, 因此, 牵涉到文件流操作;
- 3) 在设计消息通知函数和处理函数应用到 MFC 中如何进行添加处理时间的操作。
- 4) 在设计界面布局时结合来了 HTML 的灵活, 将 HTML 载入对话框, 避免了很多麻烦, 设计出简单易用的界面。

3. 本软件不足和改进:

- 1) 本软件没有提醒功能, 比如像闹钟那样提醒;
- 2) 本软件后台不能运行, 并且运行平台太单一;
- 3) 本软件不能设置任务到云端服务器保存, 使用户多台电脑同步使用;
- 4) 可以加一个天气预报并提醒不良天气功能;
- 5) 可以结合邮箱使用。

4. 体会:

本次项目开发中, 我们制作了日程规划, 一个具有经济可行性、技术可行性, 当下还非常流行又非常实用的一个软件。在这次项目开发中, 我们感受到了开发软件过程中的困难以及成功开发后的喜悦, 更加深入地了解软件开发的重要性, 相信对我们将来的软件开发具有非常大的帮助, 同样也学习到了不少新的知识和技术, 比如将界面与 HTML 结合就是在与专业认识交流时他提的建议。因此, 在学习技术时要注意与他人交流和合作学习, 补充自己的不足。