# 第一部分 《Visual C++程序设计》实训

【内容导读】Visual C++是一门实践性很强的重要基础课程,而实训是教学的一个重要环节,也是检验学习效果、提高学生动手能力与学习兴趣的一种重要形式,对理解消化课堂内容非常有益。学生在学习这门课程时,要十分重视实践环节,做到独立编写程序与调试程序,建议在学完本课程后,结合实际做课程设计,完成一个小型应用软件的设计与开发。

本部分在选材时注重从教学的实际出发,结合教材内容,安排了14个实训、一个上机测 试环节与一个综合案例,供教师教学与学生学习时选用。其中的综合案例,在配套光盘中有全 部源代码,这对提高学生的编程能力大有裨益。

# 实训1 预备知识

### 💹 一、实训目的与要求

1. 熟悉 Windows 基本操作,掌握利用"我的电脑"或"Windows 资源管理器"进行文件 (夹)管理的基本方法。

2. 掌握屏幕截图的基本方法,能够利用画图工具或 Photoshop 对图片进行简单加工。

3. 熟练掌握利用 Word 撰写实验报告。

4. 掌握常用压缩工具 WinRAR、WinZIP 的使用。

5. 掌握 E-mail 与 FTP 的使用。

### 🧧 二、实训内容

1. 在 C 盘根目录下以你的学号建立一个文件夹, 然后再在其中建立一个文本文件 exam, 内容为: 实验1预备知识。

2. 打开"我的电脑"窗口,将该窗口的信息以图片形式存于上述文件夹的 Win.bmp 文件中。

3. 利用 Word 在上述文件夹中建立一个 SY1.DOC 文档, 然后将上述文本文件及图片文件 内容插入该文档, 并适当排版。

4. 将上述文件夹的内容以你的学号作为压缩文件名进行压缩,然后将此作为附件发给: ygwang21@163.com,标题为:实验报告1,内容为:学号(XXXXXX)作业见附件。

5. 对 4.中的压缩文件利用 FTP 方法上传至 210.45.212.43 (用户名: stud-sxx, 密码: 888888)。

#### 💼 三、实验设备及环境

1. 每人一台 PC 机器。

2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。

3. 网络畅通。

#### 📝 四、实训步骤

- 1. 启动计算机,进入系统。
- 2. 按照要求利用"我的电脑"或"Windows资源管理器"建立文件夹和文本文件。
- 3. 利用抓图工具或按 Alt+PrintScreen 组合键抓图并以指定文件存盘。
- 4. 启动 Word,利用"插入"菜单完成相应操作,排版存盘。
- 5. 利用压缩工具进行压缩,按要求操作。
- 6. 写出实验报告。

# 📌 五、注意事项

- 1. 注意开机的先后顺序。
- 2. 注意"我的电脑"→"工具"→"文件夹选项",使文件扩展名可见。
- 3. 注意 PrintScreen 与 Alt+PrintScreen 的区别。
- 4. 检查网络是否畅通。
- 5. 注意上机过程中的屏幕显示,以便整理出实验报告。

#### 💊 六、拓展与思考

- 1. 列出3种以上建立文件(夹)的方法。
- 2. 如何在 www.163.com 中申请一个免费邮箱?
- 3. 如何在互联网中搜索并下载常用的免费软件?

# 实训 2 Visual C++集成开发环境(IDE)

### 💹 一、实训目的与要求

- 1. 熟悉 Visual C++的集成开发环境及菜单与工具栏的使用。
- 2. 掌握用应用程序向导创建一个控制台程序的方法。
- 3. 掌握在 Visual C++环境中新建、编写、保存程序的方法与基本步骤。
- 4. 掌握 Visual C++项目的编译、连接和执行。
- 5. 掌握代码简单语法错误修正和调试的一般过程。

#### 🤁 二、实训内容

1. #include "iostream.h"

```
void main()
{
    cout<<"Hello!\n";</pre>
```

```
cout<<"Welcome to C++!"<<endl;
```

- }
- 2. 下列程序是从键盘输入一个整数,然后计算它的平方,并以十六进制的形式输出。请

2

```
在 Visual C++开发环境输入、调试该程序。
        #include "iostream.h"
        void main()
        {
          int x;
          cin>>x;
          int y=x^*x;
          count << "x*x = =" << hex << y << endl;
        }
    3.【例 1.2】求两个数之和。
        //Ex1_2.cpp
        #include "iostream.h"
        int add(int x, int y)
        {
          int z;
          z=x+y;
          return z;
        }
        void main()
        {
          int a, b, c;
          cin>>a>>b;
          c=add(a,b);
          cout<<c<endl;
         }
```

#### 💼 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

#### 📝 四、实训步骤

1. 启动 Visual C++ 6.0 开发环境。

2. 创建一个工程。

(1) 单击"文件"→"新建"命令,显示"新建"对话框。

(2)单击"工程"标签,在"工程"选项卡中选择 Win32 Console Application (Win32 控制台应用程序),在"位置"文本框中指定一个路径,在"工程名称"文本框中为项目输入一个名字 Exam1\_1,如图 1.1 所示。

(3) 单击"确定"按钮后如图 1.2 所示,单击"完成"按钮后再单击"确定"按钮即进入开发环境。

新建			<u>1</u>
文件 工程   工作区   其它文档			
-프ATL COM AppWizard		工程名称(M):	
Cluster Resource Type Wizard		Exam1_1	
Database Project     Dev Studio Addia Wizard		位置(C):	
Extended Stored Proc Wizard		C:\Exam1_1	1.1
KiAPI Extension Wizard Makefile Makefile Marc ActiveX ControlWizard MFC AppWizard (Ext) MFC AppWizard (Ext) MFC AppWizard (Ext) MFC Application Min32 Console Application S) Win32 Console Application S) Win32 Static Library Min32 Static Library		<ul> <li>● 创建新的工作空</li> <li>● 添加到当前工作</li> <li>□ 从属于(0):</li> <li>□</li> <li>□</li> <li>〒</li> <li>〒</li> <li>〒</li> <li>●</li>     &lt;</ul>	<b>间(1)</b> 空间(4) 又
<u>-</u>		确定	取消
USI 111 Win32 Console Application - 歩載1共1歩	您想要创建什 。 <u>一个空</u> 。 一个面 。 一个面 。 一个支持	wife 么类型的控制台程F <u>「程</u> 匠] 約程序(S) 10、World <sup>19</sup> 程序(M) ド MFC 的程序(M)	2)×
	下一步>	完成	取消

图 1.2 "控制台应用程序"第1步

3. 创建一个 C++源文件。

(1) 单击"文件"→"新建"命令,显示"新建"对话框。

(2) 单击"文件"标签,在"文件"选项卡中选择 C++ Source File,在"文件名"下方 输入源程序文件名 SX1 后单击"确定"按钮,如图 1.3 所示。

90 Exam1_1 - Microsoft Visual C++ - [SX1.CPP]	
③ 文件(E) 编辑(L) 查看(Y) 插入(L) 工程(E) 细漆(L) 工具(L) 密□(W) 帮助(L)	_#×
12 😹 🖬 🖉 🖇 📾 🛍 🗅 × 🗠 × 📧 🎘 🚼 🐂 📃 🔽 🐂	
[Globals] 💽 [All global members - Create New Class] 🔍 👻 🗐 🛎 🚦	6 13
i al 20	14 P
Classy Frieview	<u> </u>
当 」 」 」 」 】 】 】 】 】 】 】 】 】 】 】 】 】 】 】	×
就绪 行 1. 列 1 REC COL 覆盖	读取 //

图 1.3 源程序编辑界面

4

(3) 在光标所在处输入实训内容。

4. 进行编译、连接、运行。

选择"组建"→"组建"命令或按快捷键 F7 系统开始对所输程序进行编译、连接,同时 在输出窗口中出现编译信息,如图 1.4 所示。

🎌 Exam1_1 - Microsoft Visual C++ - [SX1.CPP]		_ 🗆 ×
□ 文件(E) 编辑(E) 查看(V) 插入(I) 工程(P) 組建(B) 工具(I) 窗口(W) 帮助(H)		_ 8 ×
) 🖆 🚅 🕼 🐰 ங 📾 🗠 × 오 × 🚾 🗖 😤 🐂	- <b>*</b> *	
(Globals) (All global members 🗸 💊 main	• * •	🕸 🛗 👗 🚦 🕘
#include "iostream.h"         Image: Stream include "iostream.h"         Void main()         (cout<<"Hellot\n";		
ClassV FileView		<u>•</u>
Mark (油球) (油球) なび供口由 香港 ) なび供の団 (1)		
))))))))))))))))))))))))))))))))))))))	行고 제 20	
3/07日	11 3, 24 33	

图 1.4 编译信息

当无错误及警告信息时表示可执行文件已经正确生成,可以通过"组建"→"执行"命 令或按快捷键 Ctrl+F5 执行了;否则再修改调试。

## 📌 五、注意事项

 标题栏文件名后有 "\*" 表示程序没有存盘,如选择"文件"→"保存" 命令则 "\*" 消失。

2. 一个工程中只能有一个 main()函数。

3. 要注意上机过程中屏幕上显示的信息,并学会根据计算机所给出的信息修改程序,这 是程序动态调试的步骤之一。

4. 如实记录上机过程中输入数据和输出结果的情况,以便整理出实验报告。

💊 六、拓展与思考

利用 Visual C++创建一个 Win32 控制台应用程序需要经过哪几步?每一步生成的文件扩展名是什么?

## 实训3 顺序结构程序设计

# 🗾 一、实训目的与要求

1. 掌握基本数据类型及常量、变量、运算符和表达式的使用。

2. 掌握 Visual C++程序中输入/输出数据的基本方法,能正确使用各种格式控制符。

3. 掌握编译预处理命令的使用。

4. 能阅读、分析、调试简单的程序。

5. 理解简单的 C++程序结构,能设计简单的顺序结构程序。

### 🧧 二、实训内容

1. 下列程序的功能是: 求 3 位学生的平均成绩,并对平均成绩四舍五入后保留两位小数。 试分析、调试程序(注意: 程序中有错误)。

```
#include "iostream.h"
void main()
{
  float s1,s2,s3;
  double aver;
  count<<"输入 3 个数据: "<<endl;
  cin>>s1>>s2>>s3;
  aver=(s1+s2+s3)/3;
  aver=aver*100+0.5;
  aver=(int)aver;
  aver=aver/100;
  cout <<"平均值="<<aver<<endl;</pre>
```

}

- 2. 编写一个计算梯形面积的程序。要求梯形的上底、下底和高在变量初始化时直接赋值。
- 3. 输入直角三角形的两条直角边长,调用平方根函数 sqrt 来求斜边的长度。
- 4. 根据公式C= $\frac{5}{9}$ (F-32)(其中C为摄氏温度,F为华氏温度)编写程序,将输入的华氏温度转换成摄氏温度。

人通度特殊风域以通度。

5. 已知地球体积的计算公式为:

 $v = \frac{4\pi r^3}{3}$  (地球的半径为 6356.91km, 平均密度为 5.52t/m<sup>3</sup>)

编写一个计算和估计地球重量的程序。要求地球半径从键盘输入,地球密度定义成符号常量。

### 🚺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

#### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试, 直至正确。

4. 试运行程序。按照各题中给定的已知条件,计算出各题的结果,记录程序调试情况。

5. 正式运行程序,记录数据输入和数据输出的情况。

6

6. 写出实验报告。

#### 📌 五、注意事项

 将各个程序输入计算机,并且编译和连接,如果在编译、连接中有错误信息,则编译 窗口可提示错误信息,双击错误提示信息,光标将自动定位在发生错误的代码行上,然后修改 程序,直至正确。

2. 如果只有警告(Warning)信息而没有错误信息,程序还可以运行,但很可能存在某种 潜在的错误。

#### 💊 六、拓展与思考

1. 如何修改本实训的第1题, 使之能输出他们之中的最高分?

2. 为什么第3题需要加<math.h>编译预处理命令?

3. 程序书写格式有哪些规范? 怎样遵守?

4. Visual C++中常用的编辑键有哪些? 掌握 Ctrl+C、Ctrl+X、Ctrl+V、Ctrl+Z、Ctrl+Y、Ctrl+L 的作用。

# 实训4 选择结构程序设计

# 🗓 一、实训目的与要求

1. 熟练掌握关系及逻辑表达式在程序设计中的运用。

- 2. 正确利用 if 语句的各种形式实现不同分支选择的方法。
- 3. 掌握使用 switch 语句实现多分支选择的方法。

#### 🥏 二、实训内容

1. 下列程序的功能是求两个非 0 整数相除的商和余数。程序有若干错误(包括 Warning 错误),试修改并调试。

#include "iostream.h"
main()
{
 int x,y,r1,r2;
 cin>>x>>y;
 if(x=0||y=0)
 cout<<"Input Error!"<<endl;
 else
 if(x>y)
 r1=x/y;
 r2=x%y;
 else
 r1=y/x;
 r2=y%x;

cout<<"商="<<r1<<",余数="<<r2<<endl;

2. 已知某网吧上网费用的计算方法如下:

但上网费用超过50元时,将强制结账退网。试编程根据输入上网的时间计算上网费用。

3. 某商店店庆促销,规定顾客购买商品时所花的钱 x,按下述方法对所付钱 y 打折支付:

$$y = \begin{cases} x & x < 500 \\ 0.9x & 500 \le x < 1000 \\ 0.8x & 1000 \le x < 1500 \\ 0.7x & x \ge 1500 \end{cases}$$

试分别利用 if 和 switch 语句两种形式实现。

- 4. 从键盘输入 x、y、z 三个数,按 "XX>XX>XX" 格式从大到小显示。
- 5. 已知一元二次方程  $ax^2+bx+c=0$  的两个实根  $x_1$ 、 $x_2$  的计算公式为:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

要求系数 a、b、c 通过键盘输入,编程计算并输出一元二次方程的根。

#### 🝺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

#### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试, 直至正确。

4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。

5. 正式运行程序,记录数据输入和数据输出的情况。

6. 写出实验报告。

# 📌 五、注意事项

Visual C++编辑器功能非常强大,它具有许多优点,你可以在不断的探索中对其进行了解。 下面的特点你可能已有所体会:

(1) 自动语法:用高亮度和不同颜色的字来显示关键字和注释内容。

(2) 自动缩进:帮助你排列源代码,使其可读性更强。

}

(3)参数帮助:显示预定义的 Windows 函数的参数。

(4)集成的关键字帮助:能够使你得到任何关键字、MFC 类或 Windows 函数的帮助信息(按 F1 键即可)。

(5)拖放编辑:因为你需要写很多类似的代码行,借助于拖放功能,可以方便地实现代码的移动或复制。具体操作方式为:

1)将鼠标放置在要复制的内容的开始部分,按下鼠标左键不放,拖动鼠标,直到要复制 内容的结束部分,放开鼠标,此时所选的部分成为反显。

2)鼠标放在选中内容的任意部位,按下鼠标左键,此时鼠标右下方出现一个虚的长方形标志,该标志就表示你将要拖动的内容。如果你想复制所选的内容,则再按住 Ctrl 键(默认为移动操作),此时鼠标右下方的长方形标志中间出现了一个十字形。

3) 按住鼠标左键不放,拖动鼠标,你会看到一个虚的光标跟随鼠标移动,将它移动到想 要放代码的新位置,松开鼠标左键(如果按了 Ctrl 键,在松开鼠标以后再松开按键)。

4)所选的代码即可移动(或复制)到新的位置。

(6) 自动错误定位: 能自动将光标移动到有编译错误的源代码处。

# 💊 六、拓展与思考

如何在调试程序时查看一个变量或表达式的值?

实训5 循环控制结构程序设计

### 💹 一、实训目的与要求

- 1. 熟练掌握 while、do…while 和 for 三种循环语句的使用。
- 2. 了解 break 和 continue 语句在循环中的辅助控制作用。
- 3. 掌握利用循环语句求解问题的常用算法。

#### 🤁 二、实训内容

1. 下列程序是计算 20 以内的奇数和。找出并修改程序中的错误,然后上机验证。 #include "iostream.h"

```
void main()
{
    int n,sum;
    for(n=1;;n+=2);
    sum=sum+n;
    if (n==20) break;
    cout<<"sum="<<sum<<endl;
    }
2. 下面的程序通过键盘输入给定的误差δ来计算e的近似值。
    #include "iostream.h"
    void main()</pre>
```

```
{
double e=1.0,x=1.0,y,datax;
```

```
int i=1;
cout<<"Enter a datax:";
cin>>datax;
y=1/x;
while(y>=datax)
{
    x=x*i;
    y=1/x;
    e=e+y;
    ++i;
}
cout<<"e="<<e<endl;
}
```

(1) 写出程序所依据的计算公式。

(2)当输入的 datax 各是什么值时,能分别使程序按下述要求运行:不进入循环、只循环一次、只循环两次、死循环?

(3) 增加什么样的语句能知道程序运行的次数?

(4) 若把源程序中 while 语句中的 y>=datax 分别改为: y>datax、y=datax、y<datax、y<=datax、fl况有什么变化?

(5) 若把源程序中 while 语句前的 y=1/x;语句去掉,观察程序的运行将会发生什么样的变化。

(6) 若把源程序中++i;换成 i++;,观察程序的运行发生了什么样的变化。

- 3. 用 $\frac{\pi}{4} = 1 \frac{1}{3} + \frac{1}{5} \frac{1}{7} + \dots$ 公式求 $\pi$ 的近似值,直到最后一项的绝对值小于 10<sup>-5</sup>为止。
- 4. 已知求立方根 ∛a 的迭代公式为:

$$x_{i+1} = \frac{2}{3}x_i + \frac{a}{3x_i^2}$$

假定迭代到|x<sub>i+1</sub> - x<sub>i</sub> |<10<sup>-5</sup> 为止。试分别计算 a=3、27 的值,并通过调用 pow(a,1.0/3)函数 加以验证。

### 🝺 三、实验设备及环境

1. 每人一台 PC 机器。

- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试, 直至正确。 4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。

5. 正式运行程序,记录数据输入和数据输出的情况。

6. 写出实验报告。

### 📌 五、注意事项

Visual C++内置了强大的调试功能。在已经成功地进行了编译、连接,得到了可执行程序, 但是程序执行的结果不正确的情况下,调试是修改代码以便它能够正确工作的过程。Developer Studio 提供了许多工具帮助你跟踪和定位错误,调试系统提供特殊的菜单、窗口、对话框等来 为开发者提供帮助。

1. 调试命令:有关调试的命令分散在 Build、Debug、View 和 Edit 菜单中。Build 菜单包 含一个 Start Debug 子菜单,其中的命令是 Debug 菜单命令的子集。包括:启动调试过程(Go)、 单步跟踪(Step Into)和运行到光标处(Run To Cursor)。当启动调试进程后,Build 菜单会被 Debug 菜单代替,Debug 菜单包含各种控制程序执行的命令,如单步执行、进入函数体、从函 数体中出来、运行到光标所在位置等。View 菜单包含一些命令,可以控制显示各种与调试有 关的窗口,如变量窗口(Variables Window)、调用栈窗口(Call Stack Window)等。Edit 菜单 下的 BreakPoints 命令可以打开一个对话框,在其中可以插入、删除、启动、停止各个断点。

2. 设置断点:你可以控制程序直接运行到指定地点,然后查看运行到这个地方时程序的状态,如变量的值、调用栈的情况等。可以通过设置断点来达到这一目的。设置断点的方式是:将光标移到要设置断点的地方,按 F9 键,这时会有一个红的圆点出现在代码行的左边。

如果想取消断点,将光标移动到设置断点的代码行,按F9键。

3. 启动调试:按 F5 键或者在 Build 菜单中选择 Start Debug 再选择 Go,即可启动调试程序。程序会一直运行到需要用户输入或者有断点的代码处。

4. 查看变量值: 查看变量值有多种方式,可以选择你喜欢的方式进行。

(1) 可以将鼠标移动到程序的变量名处,系统会自动为你显示变量的值。

(2)复杂变量(如对象)可以通过 QuickWatch 查看,方法是:将光标定位到所要查看值的变量处,右击,选择 QuickWatch 选项即可看到变量值。

(3)启动调试程序后,屏幕下方将会出现两个输出窗口:一个是 Watch,另一个是 Variable。 Watch 窗口显示变量名和变量值,可以在 Watch 窗口中加上你想观察值的变量名,也可以直接 从源代码中选择变量名,并把它拖动到 Watch 窗口中。Variable 窗口显示程序当前运行上下文 涉及的变量的值。

5. 控制程序执行:可以控制程序单步执行(F10)、跟踪到一个函数内部(F11)、从一个 函数运行出来(Shift+F11)、运行到光标所在位置(Ctrl+F10),以便方便地调试程序。这些命 令用于在某个局部范围详细地调试程序。也可以通过设置断点(F9)然后用直接运行(GO或 F5)来控制程序直接运行到断点位置。如果设置了多个断点,程序将会在遇到的第一个断点 处停下来。要从断点处继续运行,可以用上面所说的各种命令(F5、F10、F11、Shift+F11、 Ctrl+F10)。

6. 结束调试:要结束调试,可以按 Shift+F5 键或者选择 Debug→Stop Debugging 命令。 当结束调试后,所有调试窗口会自动关闭, Debug 菜单也会自动还原为 Build 菜单。

# 💊 六、拓展与思考

- 1. 循环嵌套程序的执行过程是什么?
- 2. 循环与分支嵌套的形式是什么?

# 实训6 数组、字符串和指针

# 🗓 一、实训目的与要求

- 1. 掌握一维数组、二维数组的定义、初始化和输入/输出的方法。
- 2. 掌握字符数组和字符串的处理方法。
- 3. 掌握指针变量的定义、使用以及指针与变量、数组的关系。
- 4. 掌握结构体类型的定义和使用,了解链表的基本操作。
- 5. 掌握数组中的常用算法。

# 긷 二、实训内容

- 1. 模拟产生10个学生的百分制成绩,试编程:
  - (1) 编程输出他们的成绩、最高分、最低分及平均分。
  - (2) 按照从高到低的顺序排序输出。
- 2. 随机生成一个元素值在[1,9]之间的 3×3 矩阵 A。要求:
  - (1) 将 A 矩阵转置。
  - (2) 求 A 矩阵次对角线元素之和。
- 3. 输入一个小于 10 的正整数 n, 输出具有 n 行的杨辉三角形, 形式如下:
  - 1
  - 1 1
  - 1 2 1
  - 1 3 3 1
  - 1 4 6 4 1

```
•••••
```

4. 编写一个程序,将一串字符倒序存放后输出。例如原数组 a 的内容为"Visual C++ Programming",倒序后数组 a 中的内容为 "gnimmargorP ++C lausiV"。

5. 输入一串字符,分别利用字符数组、指针变量两种方式将字符串中的大写字母转换成 小写字母,并分别显示。

6. 找出并改正下列程序中的错误,然后上机调试运行。

```
#include "iostream.h"
struct{int a,b;}s1,s2;
struct S{int a,b;}s3,s4;
void main()
{
    cin>>s1>>s3;
```

```
s2=s1;
s4=s3;
cout<<s2<<s4;
}
```

\*7. 建立具有 5 个节点的单向链表, 然后实现数据的输出。要求每个节点含有: 学号、 姓名、年龄。

🝺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

#### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试,直至正确。

- 4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。
- 5. 正式运行程序,记录数据输入和数据输出的情况。
- 6. 写出实验报告。

# 📌 五、注意事项

- 1. 数组名与指针的关系。
- 2. 指针使用三步曲(定义、指向、使用)及其在处理字符串中的应用。

#### 🔛 六、拓展与思考

- 1. 输入一个字符串,将其中所有的字符's'删除。
- 2. 如何在一个已排序的整型数组中插入一个数,使之仍然有序?

## 实训7 函数及其应用

### 🗓 一、实训目的与要求

- 1. 掌握函数的定义、调用与说明的方法。
- 2. 掌握形参与实参之间的关系及"传值调用"与"传址调用"的区别。
- 3. 掌握递归函数的设计思想。
- 4. 掌握重载函数、内联函数、带有默认参数的函数及模板函数的定义和使用。

### 🧧 二、实训内容

1. 编写判断 m 是否为素数的函数,函数形式为:

bool isprime(int m);

在主函数中使用该函数求1000以内的素数平均值。

2. 编写判断一字符串是否为回文(回文是指顺读和倒读都一样的字符串,如"1221"和 "level"是回文)的函数,函数形式为:

#### int huiw (char s[ ]);

在主函数中对输入的5个字符串统计其中回文的个数。

3. 编写一个递归函数,实现将任意的十进制正整数转换为八进制。

4. 编写一个函数,实现将学生成绩按从高分到低分排序,并统计优秀与不及格的人数。 函数形式为:

void fun(int s[ ],int n,int &x,int &y);

在主函数中输入10个整数,调用该函数输出排序结果及优秀与不及格的人数。

如将函数形式改为: void fun(int s[],int n,int \*x,int \*y);, 又如何修改?

5. 编一函数形式为: double fun(double x,double eps=1e-5)的函数,求级数:

$$e^{x} = \sum_{n=0}^{\infty} \frac{x^{n}}{n!}$$

的部分和,当第n项的精度小于 eps (默认值为 10<sup>5</sup>)时结束。

6. 使用函数重载的方法定义两个重名函数,分别求出整型数的两点间距离和浮点型数两 点间的距离,调试成功后,再将其改为用函数模板实现。函数形式为:

double dist (int x1,int y1,int x2,int y2);

double dist (double x1,double y1,double x2,double y2);

#### 💼 三、实验设备及环境

1. 每人一台 PC 机器。

2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。

3. 网络畅通。

4. 投影仪一台。

#### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试, 直至正确。

4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。

- 5. 正式运行程序,记录数据输入和数据输出的情况。
- 6. 写出实验报告。

# 📌 五、注意事项

- 1. 函数的原型声明。
- 2. "传值调用"与"传址调用"的区别。
- 3. 重载函数的定义和使用。

## 😪 六、拓展与思考

- 1. 函数的定义与函数的说明是一样的吗? 函数在什么情况下必须说明?
- 2. 理解函数递归调用的概念,考虑在什么情况下才使用递归算法?为什么?

# 实训8 变量的作用域和生存期

# 💭 一、实训目的与要求

- 1. 熟悉不同变量的作用域和不同存储变量的性质。
- 2. 掌握自动变量、静态局部变量与全局变量之间的联系与区别。

### 🧧 二、实训内容

1. 输入以下程序,观察编译情况(注意变量j的作用域)。 #include "iostream.h" void main() { int i=10; { int j=100; cout<<"i="<<i<"j="<<j<<endl; } cout<<"j="<<j<<endl; } 分别将程序第5行和第8行的"{"和"}"删除后再观察运行情况。 2. 输入以下程序,观察运行结果(注意静态变量的默认初始值)。 #include "iostream.h" void main() { static int a[5]; for(int i=0;i<5;i++) cout<<a[i]<<endl;

### }

将程序第3行数组说明语句中的 static 删除,再观察运行结果。

```
3. 输入以下程序,观察运行结果。
```

```
#include "iostream.h"
static int x;
```

```
void main()
{
    void fun();
    int x=1;
    {
        int x=2;
        cout<<x++<<" ";
    }
    {
        extern int x;
        cout<<++x<<" ";
    }
    fun();
}
void fun()</pre>
```

void fun()
{ cout<<x++<<" ";}</pre>

将程序第 12 行中的 extern int x;语句删除后,再运行并观察结果(注意: 若全局变量与局 部变量同名时,局部变量优先)。

```
4. 输入如下程序,观察运行结果。
```

```
#include "iostream.h"
#include "iomanip.h"
void row()
{
int a=1;
int b;
 for(b=1;b<=9;++b)
   cout<<setw(5)<<a*b;
 cout<<endl;
 ++a;
}
void main()
{
int k;
for(k=1;k<=9;++k)
row();
}
```

将程序第5行的 int a=1;改写成 static int a=1;,观察其运行情况。

```
5. 建立一个控制台程序,在其中建立两个 cpp 文件,一个文件名为 file1.cpp,内容如下:
#include "iostream.h"
extern int sum;
extern void fun();
void main()
{
fun();
sum++;
cout<<"sum="<<sum<<endl;
}
```

```
另一个文件名为 file2.cpp,内容如下:
#include "iostream.h"
int sum=0;
void fun()
{
sum++;
cout<<"fun:sum="<<sum<<endl;
}
```

试编译、链接、运行程序。如果把 file2.cpp 中的语句 int sum=0;移入函数体内可否通过编译?为什么?

## 🝺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

#### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试, 直至正确。

- 4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。
- 5. 正式运行程序,记录数据输入和数据输出的情况。
- 6. 写出实验报告。

## 📌 五、注意事项

- 1. 对各题运行结果进行分析。如果程序未能通过,应分析其原因。
- 2. 总结各题的编程思路,谈谈本次实训的收获与经验。

### 💊 六、拓展与思考

- 1. 什么情况下,要用外部变量的引用性声明? 全局变量与局部变量同名时如何处理?
- 2. 静态变量如何初始化?

### 实训9 类和对象

### 💹 一、实训目的与要求

- 1. 掌握面向对象程序设计的基本概念与设计方法。
- 2. 掌握类的定义和构造方法。

- 3. 掌握对象的声明和使用方法。
- 4. 能正确区分类中成员的3种权限。

5. 掌握构造函数的定义与作用,理解析构函数只能有一个和肯定有一个的理由。

#### 🦾 二、实训内容

1. 请完善类 T 的定义, 使类 T 能实现两个整数 x、y 的减法运算, 使程序的运行结果如 图 1.5 所示。



图 1.5 运行结果

程序如下:

```
#include "iostream.h"
class T
{
 int x,y;
 public:
    T(int a_____,int b____
                               )
    {____=a;___=b;}
    void Print()
    {cout<<_
                               << endl; }
};
void main()
{
 T t1(50,8),t2(13),t3;
 t1.Print();
 t2.Print();
 t3.Print();
}
```

2. 定义一个描述书店中书籍的类 Book,数据成员包括:书名、数量和单价;成员函数包括计算总价的函数、构造函数和输出函数。请实现并测试。

3. 设计一个楼房类 Building,要求包含楼的长、宽、楼层、每平米单价等数据成员,能够用构造函数和 set 成员函数两种形式为对象的数据成员赋值,并具有求楼房的面积及总价等功能。

4. 设计一个矩形类 Rect,具有长、宽属性,类还具有求解并显示矩形的周长和面积的功能以及用对象作参数求两个矩形面积和的功能。

5. 设计一个字符串类 MyString:

#include "iostream.h" class MyString

{	
char *ptr;	
int len;	
public:	
MyString(char *s=NULL);	//将 s 所指字符串送到 ptr 所指的动态内存中
MyString(MyString &t);	//将对象 t 中的字符串复制到当前的对象中
bool isIn(char c);	//判断字符串中是否包含指定的字符 c
void Print()	
{if(ptr)cout< <ptr<<endl;}< th=""><th></th></ptr<<endl;}<>	
~MyString();	//释放 ptr 所指的动态内存
1	

};

试完成上述4个成员函数的定义,并编程测试所定义的类。

#### 🝺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试,直至正确。

4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。

5. 正式运行程序,记录数据输入和数据输出的情况。

6. 写出实验报告。

### 📌 五、注意事项

1. 在什么地方初始化类的静态数据成员?

2. 非静态成员函数能否访问静态数据成员?

## 💊 六、拓展与思考

类与对象的概念与定义是什么?从访问权限的角度看,类中的数据成员和函数成员一般 被说明成什么?

# 实训 10 类的继承和多态性

# 💭 一、实训目的与要求

1. 掌握利用类的继承关系定义派生类的方法。

2. 熟悉不同继承方式下派生类对基类成员的访问权限。

3. 掌握用友元函数和成员函数重载运算符的方法。

4. 理解虚函数、纯虚函数和抽象类的特性,掌握通过虚函数实现多态性的方法。

#### 🤁 二、实训内容

1. 在二维直角坐标系上,定义一个点类 Point,由类 Point 派生定义描述一个线段的类 Line。 要求 Line 类的成员函数能计算线段的长度。设计一个主函数测试所定义的类。

2. 定义一个圆类 Circle,具有保存数据成员半径、为半径赋值的构造函数和计算圆面积的公有成员函数。再由圆类派生出球类 Sphere、圆柱类 Cylinder 和圆锥类 Cone,并计算它们的表面积与体积。

3. 设计复数类 Complex,具有复数设置、输出、两个复数的加法和乘法功能,其中,加 法和乘法通过重载 "+"和 "\*"运算符来实现。

4. 以下程序通过虚函数计算圆面积,其运行结果如图 1.6 所示。



图 1.6 运行结果

请将程序补充完整。

```
#include "iostream.h"
class Shape
{
 public:
        (1) void draw()=0;
};
class Point:public Shape
{
 protected:
    int x,y;
 public:
    Point(int a=0,int b=0)
    {x=a;y=b;}
    void draw()
    {cout<<"画点:x="<<x<",y="<<y<<endl;}
};
class Circle:public Point
{
 protected:
    int r;
 public:
    Circle(int a,int b,int r): (2)
```

```
\{ (3) r=r; \}
   void draw()
   {cout<<"画圆:圆心是("<<x<<","<<y<<"),半径是"<<r<endl;}
};
void drawobj(Shape &s)
{s.draw();}
void main()
{
 Point *pp=___(4)__(10,20);
 Circle *pc= (5) (3,14,15);
 Shape *p;
 p=pp;p->draw();
 p=pc;p->draw();
  (6) ;
  (7);
 delete pp;
 delete pc;
}
```

🝺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

 2. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试, 直至正确。

- 4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。
- 5. 正式运行程序,记录数据输入和数据输出的情况。

6. 写出实验报告。

### 📌 五、注意事项

1. this 指针的使用。

2. 友元的定义、种类与使用时要注意的问题。

### 💊 六、拓展与思考

1. 如何理解继承是软件可重用的一种形式?

2. 派生类继承关系的特点有哪些?

- 3. 本实训内容 4 的参考解答:
  - (1) virtual
  - (2) Point(a,b)
  - (3) this->
  - (4) new Point
  - (5) new Circle
  - (6) drawobj(\*pp)
  - (7) drawobj(\*pc)

# 实训 11 C++的输入/输出流

## 🗓 一、实训目的与要求

- 1. 掌握标准设备的输入/输出方法。
- 2. 掌握 I/O 流类库中常用的输入/输出操作的方法。
- 3. 掌握磁盘文件的打开、关闭与读/写方法。
- 4. 理解文本流与二进制流在操作上的区别。

### 🧧 二、实训内容

1. 用编辑工具建立一个包含若干行字符串的文本文件,然后编程将该文件的内容按行读出,并加上行号后复制到另外一个文本文件中。

2. 从键盘输入一行字符,将其中的所有字母加密(方法是将每个字母加一序数 2,非字母字符不变)后写入到磁盘文件 data.txt 中。

3. 分析主教材第6章习题中的三、(15)程序,然后上机运行验证。

### 🛄 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

#### 📝 四、实训步骤

1. 根据题目,设计各题算法并画出结构化流程图。

2. 按照流程图编写各题的程序,并对程序认真进行语法检查与逻辑检查。

3. 上机调试程序。创建空白新工程,向新工程中添加空文件→编写代码→编译、调试,直至正确。

4. 试运行程序。按照各题中给定的已知条件计算出各题的结果,记录程序调试情况。

- 5. 正式运行程序,记录数据输入和数据输出的情况。
- 6. 写出实验报告。

### 📌 五、注意事项

1. 文件的存放位置与路径的使用。

2. 输入/输出语句格式。

## 💊 六、拓展与思考

1. 文件的操作步骤是怎样的?

2. 文件的类型有哪几种?

# 实训 12 对话框应用程序的设计

# 🗓 一、实训目的与要求

1. 理解 Windows 编程的特点及其方式。

- 2. 熟悉对话框应用程序的框架结构及其运行机制。
- 3. 掌握利用 MFC AppWizard(exe)创建对话框应用程序的方法、过程。

**4.** 掌握常用控件的重要属性、主要消息、常用成员函数,并能熟练地应用这些控件设计应用程序。

# 🧧 二、实训内容

1. 已知华氏温度与摄氏温度之间的转换公式为:  $C = \frac{5}{9}(F - 32)$ , 其中 C 为摄氏温度, F 为华氏温度。试设计一个如图 1.7 所示的转换程序。

2. 创建一个如图 1.8 所示的应用程序,以实现对编辑框中选定的内容进行复制、剪切、 粘贴和删除操作。



图 1.7 温度转换



图 1.8 文本编辑

3. 设计一个如图 1.9 所示的应用程序。当用户重新选择了字体和字型后,下面编辑框中的内容会相应改变。要求:初始时"黑体"单选按钮和"加粗"复选框处于选定状态。

4. 设计一个如图 1.10 所示的包含了两个列表框的应用程序。要求双击任一列表框中的项目时,该项目从所在的列表框中删除,并添加到另外一个列表框中。

图 1.10 列表框应用

5. 设计一个如图 1.11 所示的应用程序。当用户选定了计算机的配置并单击"确定"按钮 后,在右边的列表框中显示所选择的情况。



图 1.11 控件综合练习

6. 设计如图 1.12 所示的计算 0~10 之间某个数阶乘的程序。要求通过滚动条(最小值为 0,最大值为10,单击两端箭头时滑块移动的增量值为1,单击滑块与两端箭头之间的区域时 滑块移动的增量值为 2, 初值为 5) 与旋转按钮 (最小值为 0, 最大值为 10, 增减量为 1) 两 种方法实现。

7. 设计一个如图 1.13 所示的界面,用于实现将文本文件的内容读入编辑框或将编辑框的 内容写入指定的文本文件。

		💑 文件与控件的使用
<mark>鑫</mark> 滚动条与旋转按钮的使用	×	对文本文件进行读写
	F	功打开的检查,然后
	4	操作来完成读写操作 get(含getline)和 读写。完毕后对文件)
求阶乘: 4!=24		
	结束	

	4文作进行逐与操作之间,自充要 ,必要时边要对打开的文件进行是 开的检查,然后再对文件流按打开 行读或写的操作,可以通过提取。 来完成读写操作,也可以通过成员 (含getline)和put 实现对文本文	打开 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一	
读写	。完毕后对文件进行正常关闭。		读取文件
			写入文件
		-	结束运行

×

图 1.12 滚动条与旋转按钮的使用



#### 🝺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

### 📝 四、实训步骤

- 1. 利用 MFC AppWizard(exe)向导创建一个基于对话框的应用程序框架。
- 2. 删除现有控件,调整窗口大小,按照题目要求增加所需控件并设置控件属性。
- 3. 为控件连接变量并对感兴趣的消息编写代码。
- 4. 调试运行。

# 📌 五、注意事项

连接变量的类型与相关成员函数的使用。

#### 💊 六、拓展与思考

- 1. 使用 MFC 编程有什么优点?
- 2. 设计单选按钮时应注意什么问题?
- 3. 组合框有哪几种类型? 它们之间有什么区别?
- 4. 当对话框上有多个滚动条时,如何确定用户对滚动条的操作发生在哪一个滚动条上?

# 实训 13 菜单与工具栏的设计

## 💹 一、实训目的与要求

- 1. 熟悉菜单编辑器与工具栏编辑器的操作方法。
- 2. 掌握菜单项助记符与热键的使用。
- 3. 掌握通过菜单调用对话框的方法。
- 4. 学习工具按钮与菜单项命令相结合的方法。

## 🧧 二、实训内容

1. 设计一个具有单文档界面的简易学生成绩处 理系统,要求:

(1)处理前需要进行用户身份认证,如图 1.14 所示。

(2)通过主界面的菜单或工具栏调用成绩录入 和处理对话框,主界面如图 1.15 所示。

系统登录	×
用户名: wyg	
□ 令:  ***	
确认(11) 退出(21)	
图 1.14 登录界面	

无标题 - 简易 成绩处理(5) 关	学生成绩处理》 于( <u>A</u> )	系统		<u> </u>
16 🔋				
	成绩输入并计	<b>ţ</b>	×	
	数学:	80		
	语文:	90		
	英语:	87	计算	
	平均:	85.6666666	返回	
就绪				

图 1.15 主界面

菜单格式与功能如表 1.1 所示。

表 1.1 菜单格式与功能

成绩处理( <u>S</u> )		关于( <u>A</u> )
录入并计算( <u>O</u> )	Ctrl+O	系统信息( <u>S</u> )
退出( <u>S</u> )	Alt+F4	

成绩录入和处理界面如图 1.16 所示。

成绩输入并计	算		×
数学:	80		
语文:	70		
英语:	60	计算	
平均:	70	返回	

图 1.16 成绩输入与计算界面

系统信息界面如图 1.17 所示。

关于《简絜	易学生成绩处理系统》	×
	《简易学生成绩处理系统(1.0 版)》 版权所有 (C) 2008	铺定

图 1.17 系统信息界面

### 🝺 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。

4. 投影仪一台。

### 📝 四、实训步骤

1. 利用 MFC AppWizard(exe)创建一个工程名称为 SY\_Dlg 的单文档应用程序,并将标题 改为"简易学生成绩处理系统",单击"完成"按钮后再单击"确定"按钮即进入开发环境。

2. 单击 ResourceView 选项卡→SY\_Dlg resources→Menu→IDR\_MAINFRAME, 打开菜单 编辑器, 按实训要求修改现有菜单, 并将"录入并计算"菜单的 ID 设为 ID\_MENU\_IN;"退 出"菜单的 ID 设为 ID\_MENU\_EXIT。

注意:此时编译新增的菜单是灰色的。

3. 单击 ResourceView 选项卡→Accelerator→IDR\_MAINFRAME,为新增菜单添加快捷键, 如图 1.18 所示。

Accel 居性 ————————————————————————————————————	<u>×</u>
ID: ID_MENU_IN=32771 ▼ 键(K): 0 ▼ 下一个按下的键(N)	補助键
Accel 雇性	×
Accel 雇性 ────	×
Accel 服性 一型 ② 常规 D: ID_MENU_EXIT	× 補助键 □ <u>C</u> trl □ <u>AL</u> □ <u>S</u> hift

图 1.18 "Accel 属性"对话框

4. 单击 ResourceView 选项卡→Dialog→IDD\_ABOUTBOX,修改系统信息界面。

5. 单击 ResourceView 选项卡,右击 Dialog,在弹出的快捷菜单中选择"插入 Dialog"选项,为程序添加用户身份认证与成绩录入对话框,如图 1.19 所示。

Accelera     Accelera     Balang     Dialog     Di	tor MAINFRAME 保存(3) SY_DIg.rc 检查 插入(1) 插入(2) 5 入(M)	
	◎ 編音(D) 隐藏(H) 『 属性(O)	
■ <mark>:</mark> ClassView <b>#</b>	ResourceView 🗐	FileView

图 1.19 Dialog 的快捷菜单

修改标题,删除其中的"确定"和"取消"按钮,并按实训要求的界面图进行修改。 按 Ctrl+W 组合键打开 MFC ClassWizard 对话框,为登录界面对话框(IDD\_DIALOG1)

与成绩处理对话框(IDD\_DIALOG2)分别建立新类,类名为CDlgPw和CdlgIn,然后选择CDlgIn 类并切换到 Member Variables 选项卡,用前面所学知识为控件连接变量并为"计算"和"返回" 按钮编写相应的代码。

void CDlgIn::OnButton1()

// TODO: Add your control notification handler code here
UpdateData(true);
m\_pj=(m\_sx+m\_yw+m\_yy)/3.0;
UpdateData(false);

}

{

void CDlgIn::OnButton2()

{

// TODO: Add your control notification handler code here
OnOK();

```
}
```

为登录界面对话框(IDD\_DIALOG1)的编辑框分别连接 CString 类型的变量: m\_username 和 m password。

6. 建立对话框与菜单的联系。

按 Ctrl+W 组合键打开 MFC ClassWizard 对话框,选择类名 CMainFrame,选择 Object IDs 为 ID MENU IN,添加 COMMAND 消息,代码分别如下:

CDlgIn dlg; //创建对话框对象

dlg.DoModal(); //打开成绩输入窗口,注意要在首部添加#include "DlgIn.h"

选择 Object IDs 为 ID\_MENU\_EXIT, 添加 COMMAND 消息,代码如下: exit(0);

7. 单击 ResourceView 选项卡→Toolbar→IDR\_MAINFRAME,为新增菜单项设置工具图标; 然后双击图标,在弹出的属性窗口中选择图标对应的 ID 分别为 ID\_MENU\_IN 和 ID\_MENU\_EXIT,提示为: 请输入学生成绩\n 输入成绩并计算、结束系统一行,以与菜单同步,如图 1.20 所示。



图 1.20 工具栏图标编辑窗口

28

8. 单击 ClassView 选项卡→CSY\_DlgApp→InitInstance,在此启动主窗体前添加设置启动 界面的程序代码:

CDlgPw dlg; // 注意在程序首部加上#include "DlgPw.h" BOOL K=TRUE; while (K) //建立循环,以便当输入错误后重输 if (dlg.DoModal() == IDOK) //用户单击 "确定"按钮 if (dlg.m\_password!="123" || dlg.m\_username!="wyg") //设定用户名为 wyg, 密码为 123 AfxMessageBox("输入有误,请重输!",MB\_OK|MB\_ICONERROR); else K=false; //输入正确退出循环 else return FALSE; //用户单击 "退出"按钮 9. 编译、连接并运行。

### 📌 五、注意事项

1. 删除现有菜单,只需在菜单编辑器中右击该菜单,在弹出的快捷菜单中选择"剪切" 选项; 而删除工具栏中的按钮是将该按钮拖离工具栏,而不能按 Del 键。

2. 将编辑框属性对话框"样式"选项卡中的"密码"选中,则输入时用"\*"号代替输入内容,可避免密码泄露。

#### 😪 六、拓展与思考

1. 菜单有哪些常用的规则? 如何为菜单项设计助记符与快捷键?

2. 如何使一个工具按钮与菜单项命令相结合?

# 实训 14 文档视图结构应用程序的设计

### 💹 一、实训目的与要求

- 1. 了解文档/视图应用程序的结构及其运行机制。
- 2. 了解 SDI 应用程序和菜单设计的技术。

3. 了解 SDI 和 MDI 在编程、功能等方面的异同。

### 🧧 二、实训内容

实现教材【例 7.10】利用 AppWizard 创建一个类似于 Windows 记事本功能的文本编辑器 (默认文件扩展名为.txt),并要求在"编辑"菜单中添加"全选"、"查找"、"查找下一个"和"替换"功能。

#### 💼 三、实验设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。
- 4. 投影仪一台。

#### 📝 四、实训步骤

1. 利用 MFC AppWizard(exe)向导创建一个基于 SDI 的应用程序框架。

- 2. 对菜单按要求进行改造。
- 3. 修改标题与默认打开的文件类型。
- 4. 调试运行。

## 📌 五、注意事项

操作步骤、事件代码的编写。



1. SDI 程序框架由哪几部分组成? 它们的作用是什么?

2. 菜单设计的过程是怎样的?

# 上机测试

길 一、测试目的

通过本测试使学生掌握使用 Visual C++设计应用程序的基本步骤,能够编写、调试和运行 实用、规范、可读性好的 Visual C++程序。

### 🝺 二、测试要求

1. 时间 60 分钟。

2. 按照各题中的要求编写程序,上机调试并计算出结果。然后将整个工程以 Exam+你的 学号名压缩上交到服务器中你的目录下。

### 📚 三、测试内容(由教师指定选做一题)

1. 编写一个学生和教师数据输入和显示程序,学生数据要求有编号、姓名、班号和成绩, 教师数据有编号、姓名、职称和部门。要求将编号、姓名的输入和显示设计成一个类 person, 并作为学生数据操作类 student 和教师数据操作类 teacher 的基类,学生数据中的班号和成绩的 输入与显示在 student 类中实现,教师数据中的职称和部门的输入与显示在 teacher 类中实现。 最后在主函数中进行该类的测试。

下面给出了基类 person 的主要成员:

(1) 私有成员:

int no;	//编号
char name[10];	//姓名

- (2)公有成员:
   void input(); //编号和姓名的输入
   void display(); //编号和姓名的显示
- 2. 编写如图 1.21 所示的求两个正整数的最大公约数的程序(要求程序有一定的容错能力)。



图 1.21 运行界面

### 📌 四、测试设备及环境

- 1. 每人一台 PC 机器。
- 2. 每台机器上安装 Visual C++开发环境、Office 软件、WinRAR 或 WinZIP。
- 3. 网络畅通。

## 📝 五、评分依据

- 1. 程序编写是否规范?
- 2. 程序是否通过编译?
- 3. 程序运行结果是否正确?

# 综合案例——简易图书管理系统

本案例取材于我院 2008 届信息与计算科学专业陈俊俊同学的毕业设计,相关的代码已包含在配套的光盘中。我们希望通过该案例的学习,使大家了解 Visual C++的强大开发功能以及程序开发中的一些技巧,激励大家的学习兴趣。

图书管理系统是一个典型的管理信息系统。该案例综合运用了开发管理信息系统的相关 知识,从中小型图书馆的实际需求出发,进行设计和具体实现。主要功能有:管理员权限验证; 用户信息管理,包括操作人员的增删和密码修改;图书借还流通管理;读者信息维护管理;图 书信息维护管理,包括图书信息的增删和检索;查看历史记录。

1. 简易图书管理系统的模块划分如图 1.22 所示。



图 1.22 简易图书管理系统模块划分图

2. 简易图书管理系统的业务处理流程如图 1.23 所示。



图 1.23 简易图书管理系统业务处理流程图

3. 数据库设计。

根据系统功能的设计及功能模块的划分,并结合最低数据冗余度的原则,本系统的数据 库结构设计如图 1.24 所示,并用 Access 2003 构建数据库 db\_lib.mdb。



图 1.24 简易图书管理系统 db\_lib.mdb 数据库结构图

4. 系统开发。

主控制平台使用简单和直观的布局设计,界面简洁友好、易于操作。

主体框架:使用 MFC AppWizard 创建一个单文档结构的应用程序工程 Library,建立起系 统主体框架,生成应用程序类(CLibraryApp)、文档类(CLibraryDoc)、视图类(CLibraryView) 和主框架类(CMainFram)。

菜单:使用资源编辑器创建编辑系统菜单为:借还管理、书库管理、读者管理、系统管理(含二级菜单:注销/重新登录、关于本系统、操作员管理、历史查看等)。

主控制台界面设计如图 1.25 所示。



图 1.25 简易图书管理系统主控制台界面

另外,为了减少等待程序加载过程中的枯燥感本系统使用了启动画面,在程序启动时自动加载显示欢迎画面登录窗。启动画面可以利用组件库中的 Splash Screen 组件实现,但本系统经过一次修改,为增强交互性与安全性,采用了无标题栏的对话框(DIALOG)实现并扩展了该功能。在应用程序类(CLibraryApp)的 InitInstance()函数中加入如下代码,在主框架创建之前显示此对话框:

BOOL CLibraryApp::InitInstance()

{
...
CWelLoginDlg loginDlg;
if(loginDlg.DoModal()!=IDOK)
 return FALSE;
m\_bUserName = loginDlg.m\_strName; // 记录登录者姓名,以便传给主框架
...
}

(1) 使用 MFC ADO 连接数据库。

使用 ADO 前在工程的 stdafx.h 头文件里用直接引入符号#import 引入 ADO 库文件,以使 编译器能正确编译。代码如下:

#import "c:\program files\common files\system\ado\msado15.dll" no\_namespaces rename("EOF" adoEOF") //将常数 EOF 重命名为 adoEOF, 以避免与文件结尾常数冲突

ADO 库包含 3 个基本接口:\_ConnectionPtr 接口、\_CommandPtr 接口和\_RecordsetPtr 接口。 \_ConnectionPtr 接口主要是一个连接接口,取得与数据库的连接,返回一个记录集或一个空指 针,通常使用它来创建一个数据连接或执行一条不返回任何结果的 SQL 语句,如插入、删除

```
记录等; _CommandPtr 接口返回一个记录集, 它提供了一种简单的方法来执行返回记录集的
存储过程和 SQL 语句; _RecordsetPtr 是一个记录集对象, 与以上两种对象相比, 它对记录集
提供了更多的控制功能,如记录锁定、游标控制等。
   在具体使用时,按照如下流程来对数据库进行操作:
   1)用 ConnectionPtr 建立连接指针。
   2) 执行 SQL, 按需要使用 CommandPtr 返回一个记录集。
   3)调用 close()函数关闭记录集,释放记录集指针。
   4)调用 close()函数关闭连接,释放连接指针。
   代码如下:
      CoInitialize(NULL); //初始化 OLE/COM 库环境
      _ConnectionPtr m_pConnection;
                                             //建立数据库连接指针
      m_pConnection.CreateInstance(__uuidof(Connection));
      try
      {
       // 打开数据库 db_lib.mdb
       m_pConnection -> Open( "Provider = Microsoft.Jet.OLEDB.4.0; Persist Security Info=False;
       Jet OLEDB: Database Password = 123456; Data Source = db_lib.mdb", "", "", adModeUnknown);
      }
      catch(_com_error e) //异常捕捉
      {
       AfxMessageBox("数据库连接失败!");
       return;
      }
      CString strSQL;
                     //构建 SQL 查询字符串
       strSQL = "•••••";
      _variant_t RecordsAffected;
      _RecordsetPtr m_pRecordset;
      m_pRecordset.CreateInstance(__uuidof(Recordset)); //建立记录集指针
      //执行 SQL 语句,并返回记录集
      m_pRecordset = m_pConnection -> Execute( ( _bstr_t )strSQL, &RecordsAffected, adCmdText );
      ...
      m_pRecordset->Close();
                          //关闭记录集
      m_pRecordset.Release();
                          //释放指针
      m_pConnection->Close(); //关闭连接
      m_pConnection.Release();
                         //释放连接指针
      CoUninitialize();
   为数据库数据的安全及数据同步着想,本系统在每个功能模块中一旦结束该模块的操作
```

```
就关闭数据库。
```

(2) 权限验证模块和用户信息管理模块开发。

根据需求分析,权限验证模块主要是根据输入的用户名和密码验证用户身份,决定其是 否有操作权限。本系统中该模块分为欢迎界面登录模块和注销再次登录模块,这里以注销再次 登录模块为例,具体界面如图 1.26 所示。



图 1.26 权限验证模块

在本系统中使用 BOOL 型全局变量 m\_bLogin 来判断用户权限,登录成功则为 TRUE,开 启功能模块,并在主控制台的状态栏中显示当前操作人员,否则为 FALSE。在登录失败或未 登录的情况下,所有的功能模块都无法使用。代码如下:

CLoginDlg dlg; if (dlg.DoModal() == IDOK){ m\_bLogin = TRUE; m\_strUserName = dlg.m\_strName ; // 设置状态栏 CString strUser; strUser = "当前操作员: " + m\_strUserName; m\_wndStatusBar.SetPaneText(0,strUser); } else{ CString strUser; strUser = "当前操作员: " + m\_strUserName; m\_wndStatusBar.SetPaneText(0,strUser); }

同时,针对操作人员基本信息的建立与维护,开发用户管理信息管理模块,以实现对管理人员进行增加、删除及密码修改。该模块主要划分为3个子模块:新增用户、修改密码、删除用户,具体界面如图1.27所示。



图 1.27 用户管理模块

根据用户的选择,使用控件的 EnableWindow()函数实现右侧所需编辑框的开启与关闭, 以引导用户正确完成操作。如单击"修改密码"按钮时,则开启旧密码、新密码、确认密码 3 个编辑框,当旧密码通过认证,并且新密码与确认密码一致时单击"保存"按钮,将会弹出消 息框提示修改密码成功。

在这个界面中,新增用户和修改密码两个按钮的响应函数只负责实现右侧编辑框的开启, 具体功能在"保存"按钮中实现,以下是"保存"按钮实现修改密码的部分代码:

```
void COManageDlg::OnOmgSave()
{
 UpdateData();
 // 用户填写信息判断
 if (m_ctrUserID.IsWindowEnabled()){ //新增用户
      ...
 }
 else { //修改用户密码
      if(m_strUserID == ""){
          MessageBox("请先选择一个用户!");
          return;
      }
      if(m_strOldPwd == ""){
          MessageBox("请输入密码!");
          m_ctrOldPwd.SetFocus();
          return;
      }
 }
 if(m_strPwd == "")
      MessageBox("密码不能为空!");
      m_ctrPwd.SetFocus();
      return;
 }
 if(m_strPwd != m_strRePwd){
      MessageBox("密码不匹配,请重新输入!");
      m_strPwd = "";
      m_strRePwd = "";
      UpdateData(FALSE);
      m_ctrPwd.SetFocus();
      return;
 }
 //判断完毕,下面更新数据库
 //链接数据库
 ...
 CString strSQL;
                  //创建 SQL 查询字符串
 if (m_ctrUserID.IsWindowEnabled()){ //新增用户
      ...
 }
 else{ //修改用户密码
      _RecordsetPtr m_pRecordset;
                                  //创建记录集指针
      m_pRecordset.CreateInstance(__uuidof(Recordset));
      strSQL.Format("select * from operators where operator_ID='%s' and passwd='%s'", m_strUserID,
      m_strOldPwd );
      m_pRecordset = m_pConnection -> Execute( ( _bstr_t )strSQL, &RecordsAffected, adCmdText );
      if(m_pRecordset->adoEOF) { //对旧密码进行认证
```

```
36
```

```
MessageBox("旧密码不正确,请重新输入!");
            ...
            return;
       }
       m_pRecordset->Close(); // 关闭记录集
       m_pRecordset.Release();
       // 更新记录, 完成密码修改
       strSQL.Format("update operators set passwd='%s' where operator_ID=%s"", m_strPwd, m_strUserID);
       m_pConnection->Execute((_bstr_t)strSQL,&RecordsAffected,adCmdText);
       MessageBox("密码修改成功,请记住你的新密码!");
   }
   //更新用户列表
   RefreshData();
   ...
   //关闭链接
  }
(3) 书库管理模块和读者管理模块开发。
```

书库管理模块主要实现对所有图书信息的管理功能,包括新书的入库登记、图书信息检 索和浏览等。读者管理模块主要实现对所有读者信息的管理功能,包括新读者的登记、读者信 息检索和浏览、读者注销等,同书库管理模块在实现上极为相似,因此这里只以书库管理模块 为例。

书库管理模块主要划分为4个子模块:新书登记、显示所有图书信息、据书名检索图书、 删除图书信息。同样也是根据用户的不同选择开启不同的编辑框让用户操作,如单击"书名检 索"按钮时,仅开启"图书名称"编辑框,单击"提交"按钮后,将在列表框中显示出检索到 的图书信息,其界面设计如图 1.28 所示。

1. 书库管理						
图书信息		_				
图书编号:						
图书名称:	C程序设计语	言		作	者:	
出版社:				图书	价格:	
图书摘要:						
						提交
图书编号 图 2008001 C科	书名称 呈序设计语言	作者徐宝文	出版社 机械工业出	图书价格 30.00	借出状态  已借出	摘要 本书是由Ci
2008010 C	坐序设计语言 [HE END!	徐宝文 *****	机械工业出	25.00	在馆	本书主要由
<b>e</b>						
新书登记	. 显示	全部	书名检索	删除		退出

图 1.28 书库管理模块

以"删除"按钮为例,以下是部分代码:

void CBManageDlg::OnBmgDelete()
UpdateData();
if( m\_strBookID == "" ){
 MessageBox("请先选择一本图书!");

return;
}
int checkdel = MessageBox("确认删除该图书吗? ","警告",MB_ICONWARNING   MB_YESNO);
if( checkdel != 6 )
{
return;
}
//链接数据库
CString strSQL; //构建 SQL 查询字符串
strSQL.Format("delete from books where book_ID='%s'",m_strBookID);
m_pConnection->Execute((_bstr_t)strSQL,&RecordsAffected,adCmdText);
//关闭链接
MessageBox("已成功删除指定图书! 请刷新图书列表! ");
//清空用户列表
m_ctrList.DeleteAllItems();

(4) 借阅图书模块和还书模块。

这两个模块应该是图书管理系统的核心模块,但它们是以前面几个模块为基础建立的。 借阅图书模块要求完成读者信息认证和图书信息认证,若读者录入信息不正确或图书已被借 出,则立即中止借书,防止数据错误;否则正常借出,并生成借书记录备档。还书模块相对简 单,更新借书读者的借书状态和图书借出状态,并生成历史记录备档即可。此处仅以借阅图书 模块为例。

借阅图书模块,要求在输入完读者编号后,立即显示用户信息、图书信息及已借图书情况。同样,在输入图书编号后,立即调出该书信息,以便确认。这里采用了 OnKillfocus\*\*\*() 功能响应函数来完成这一功能,效果十分理想。借书模块的界面设计如图 1.29 所示。

… 借书							.:. 8
读者信息							
读者编号:	2004001			姓:	名: 🗄	E小明	
单位:	数学科学学院			性:	别: 月	夷	
电 话:	1234567			2 ·	借:	2	册
图书信息							
图书编号:	2008003						
图书名称:	C++编程你也行			作	者:	荣耀	
出版社:	机械工业出版社			价	格: 🤇	45	
摘 要:	本书是一本优秀的	的C++教材,	内容包括:	基础类型、	操作	符和简单变	<b>置,</b> 1
↓ 该读者已借图=	书 <b>:</b>						
流水编号 图 27 计 26 C	书名称 ├算机专业英语 程序设计语言	借出日 2008 2008	<b>1期</b> -05-09 -05-09		操作 adi adi	乍员 min min	
				确定借	出	退	出

图 1.29 借阅图书模块

以下为"确定借出"按钮响应函数的部分代码: void CBorrowDlg::OnBowOk()

```
{
        // 连接数据库
        UpdateData();
        if (m_strReaderID.IsEmpty())
        {
             AfxMessageBox("读者编号不能为空!");
             m_ctrReaderID.SetFocus();
             return;
        }
        if (m_strBookID.IsEmpty())
        {
             AfxMessageBox("图书编号不能为空!");
             m_ctrBookID.SetFocus();
             return:
        }
        // 创建记录集指针
                        //构建 SQL 查询字符串,判断图书在馆情况
        CString strSQL;
        strSQL.Format("select isborrowed from books where book_ID='%s'",m_strBookID);
        m_pRecordset = m_pConnection -> Execute( ( _bstr_t )strSQL, &RecordsAffected, adCmdText);
        _variant_t visBow;
        visBow = m_pRecordset->GetCollect(_variant_t((long)0));
        if( visBow.lVal >= 1 )
        {
             AfxMessageBox("该图书已被借出,请查找其他副本");
             return;
        }
        //关闭记录集
        //确定借出,完成数据库的更新
        strSQL.Format( "insert into borrow( reader_ID,book_ID,operator_ID ) values( '%s', '%s', '%s')",
        m_strReaderID,m_strBookID,m_strOperatorID );
        m_pConnection->Execute((_bstr_t)strSQL,&RecordsAffected,adCmdText);
        strSQL.Format("update readers set bbcount=bbcount+1 where reader_ID='%s'", m_strReaderID );
        m_pConnection->Execute((_bstr_t)strSQL,&RecordsAffected,adCmdText);
        strSQL.Format("update books set isborrowed=1 where book_ID='%s'", m_strBookID);
        m_pConnection->Execute((_bstr_t)strSQL,&RecordsAffected,adCmdText);
        //关闭链接
        MessageBox("图书借出成功!");
        RefreshData();
       }
     (5) 历史查看模块。
    历史查看模块是一个附加模块,并不是该体系的必需模块。但其功能却不小,主要用来
查看图书流通的历史记录信息,方便图书流通过程中对图书流向的把握,以及图书馆整体状态、
```

历史查看模块主要分为两个子模块:借出历史查看和归还历史查看,借出历史显示所有借出未还图书的历史;归还历史则显示已归还的图书流通记录。

历史查看模块界面设计如图 1.30 所示。代码相对简单,仅仅是一个简单的数据库查询, 在前面几个模块的代码中也有体现,这里不再重复。

	+ 1.1		Att a set the	- Intert Ale	
344137 1 2 5 5 6 3 7 4	*王王王王王张张陈小样	四10日476 计算机考生工英语 C++编程你也行 计算机考生及语 WC++信息系统开发实例 蚁群算法原理及其应用 SQL语言艺术 C程序设计语言	2008-05-04 2008-05-04 2008-05-04 2008-05-04 2008-05-04 2008-05-08 2008-05-08	2008-05-04 2008-05-08 2008-05-08 2008-05-09 2008-05-09 2008-05-09 2008-05-09 2008-05-09	main admin admin 陈俊俊 admin admin admin 陈俊俊

图 1.30 历史查看模块

(6) 系统界面美化。

为增强系统界面的友好,本系统在功能全部实现后,替换精致的图标,并采用 SkinMagic 对本系统的界面进行了皮肤加载,全面美化。在应用程序类(CLibraryApp)的 InitInstance() 函数中加入如下代码:

BOOL CLibraryApp::InitInstance()

{

```
VERIFY(1 == InitSkinMagicLib(AfxGetInstanceHandle(), _T("ReadUI"), NULL, NULL));
VERIFY(1 == LoadSkinFile("Skins\\xplus.smf")); //加载皮肤文件 xplus.smf
VERIFY(1 == SetDialogSkin(_T("Dialog")));
```

美化效果对比如图 1.31 和图 1.32 所示。



图 1.31 界面美化后

图 1.32 界面美化前

- 5. 系统的主要优点。
  - (1) 采用 Visual C++ 6.0 开发,运行效率高。
  - (2) 采用 Access 2003 作为后台数据库,小巧灵活。
  - (3) 界面友好美观,操作简单。
  - (4) 采用 SkinMagic 对本系统界面进行美化,简洁大方。