

实验四 基于 MFC 编程（一）

一、实验目的：

- 1 掌握 MFC 类的组织结构和主要类
- 2 掌握 MFC 下标准控件的使用，每类控件的属性、事件及方法

二、实验要求：

- 1 预习本次实验的实验要求和任务
- 2 熟悉基于 MFC 的标准控件应用方法，并独立完成本次实验

三、实验环境：

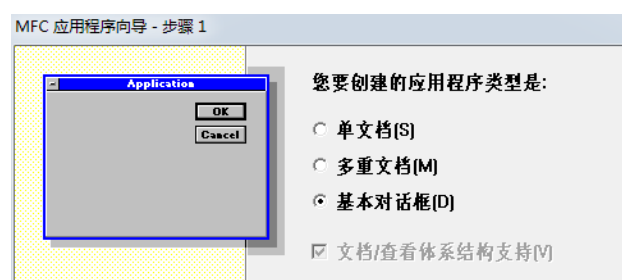
- 1 软件：VC++ 6.0 或 Visual Studio

四、实验内容：

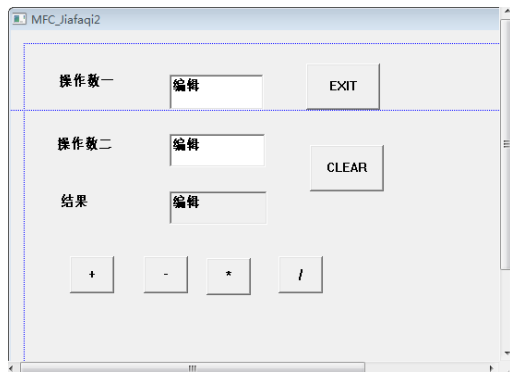
- 1 创建一个基于 MFC 的工程，利用 MFC 应用程序向导创建一个基本对话框
- 2 在对话框里添加加减乘除运算的相关控件，并设置各个控件的名称、属性等
- 3 实现加减乘除运算，并能给出正确运算结果

五、实验过程及结果描述

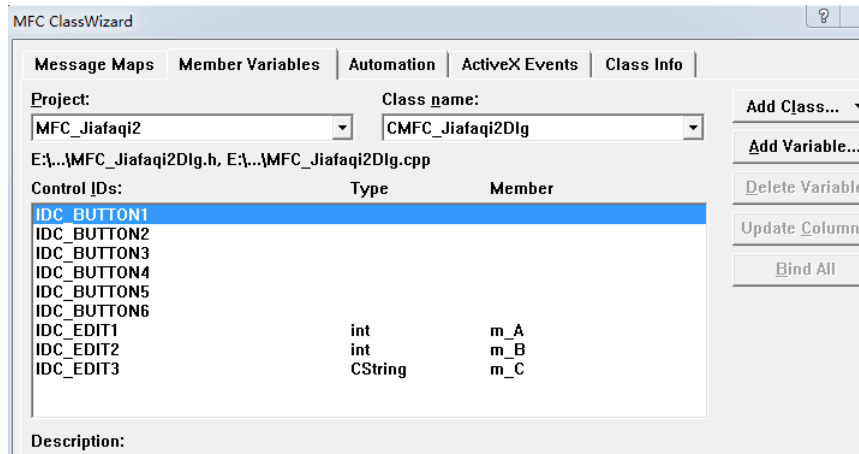
- 1 在 VC++ 6.0 环境下，选择“新建”，再选择“新建”下的“工程”中的“MFC Appwizard[exe]”，选择要创建的工程所在文件夹，输入要创建的工程文件名
- 2 在 MFC 应用程序向导第一步中选择“基本对话框”



- 3 依次按照 MFC 应用程序向导步骤创建一个包含对话框的 MFC 工程
- 4 在对话框资源中添加相应的静态文本框、编辑框与按钮控件，如下图所示：



5 同时按下 Ctrl+W 键，打开 MFC ClassWizard, 给相应控件对象添加相应的成员变量，设置各成员变量的类型，如下图所示：



6 对六个下压式按钮分别添加单击消息响应函数，并各自键入相应代码，代码如下：

```
(1) void CMFC_Jiafaqi2Dlg::OnButton1()
{UpdateData(TRUE);
    int i=m_A+m_B;
    m_C.Format("%d",i);
    UpdateData(FALSE);
    // TODO: Add your control notification handler code here
}
```

```
(2) void CMFC_Jiafaqi2Dlg::OnButton2()
{UpdateData(TRUE);
    int i=m_A-m_B;
    m_C.Format("%d",i);
    UpdateData(FALSE);
    // TODO: Add your control notification handler code here
}
```

```
(3) void CMFC_Jiafaqi2Dlg::OnButton3()
{UpdateData(TRUE);
```

```

        int i=m_A*m_B;
        m_C.Format("%d",i);
        UpdateData(FALSE);
        // TODO: Add your control notification handler code here
    }

```

```

    (4) void CMFC_Jiafaqi2Dlg::OnButton4()
    {UpdateData(TRUE);//刷新控件的值到对应的变量。(外部输入值交给内部变量)
        float i=float(m_A)/float(m_B);
        m_C.Format("%6f",i);
        UpdateData(FALSE);//拷贝变量值到控件显示。(变量的最终运算结果值交给外部输出显示)
        // TODO: Add your control notification handler code here
    }

```

```

    (5) void CMFC_Jiafaqi2Dlg::OnButton5()
    {OnOK();
        // TODO: Add your control notification handler code here
    }

```

```

    (6) void CMFC_Jiafaqi2Dlg::OnButton6()
    {m_A=0;
        m_B=0;
        m_C.Format("%d",0);
        UpdateData(FALSE);
        // TODO: Add your control notification handler code here
    }

```

注意：实验报告只需附上程序运行截图即可，程序的标题显示自己的姓名和学号！

实验五 基于 MFC 编程（二）

一、实验目的：

- 1 掌握 MFC 下如何对标准控件设置添加关联的成员变量
- 2 掌握 MFC 下如何添加消息响应函数

二、实验要求：

- 1 预习本次实验的实验要求和任务
- 2 熟悉基于 MFC 的标准控件成员变量设置及消息函数创建，并独立完成本次实验

三、实验环境：

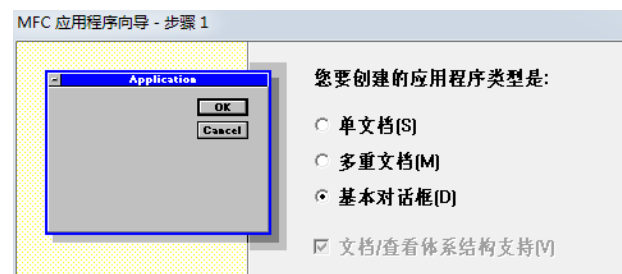
- 1 软件：VC++ 6.0 或 Visual Studio

四、实验内容：

- 1 创建一个基于 MFC 的工程，利用 MFC 应用程序向导创建一个基本对话框
- 2 在对话框里添加能够实现多组数据输入、数据修改、数据查询的相关控件，并为相应控件建立关联的成员变量
- 3 创建相应的消息函数，并且添加消息响应代码，正确实现多组数据的输入、修改以及查询的功能

五、实验过程及结果描述：

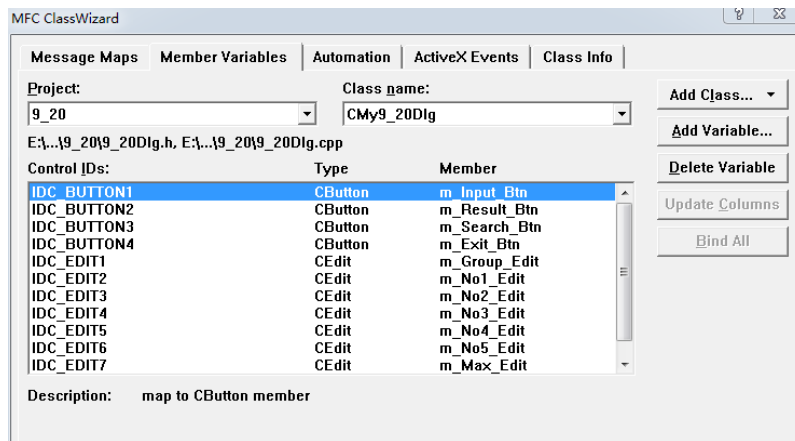
- 1 在 VC++ 6.0 环境下，选择“新建”，再选择“新建”下的“工程”中的“MFC Appwizard[exe]”，选择要创建的工程所在文件夹，输入要创建的工程文件名
- 2 在 MFC 应用程序向导第一步中选择“基本对话框”：



- 3 依次按照 MFC 应用程序向导步骤创建一个包含对话框的 MFC 工程
- 4 在对话框资源中添加相应的静态文本框、编辑框、组框与按钮控件，如下图所示：



5 同时按下 Ctrl+W 键，打开 MFC ClassWizard, 对相应控件对象添加相应的成员变量，如下图所示：



6 给“输入数据”、“输出结果”、“检索数据”、“退出”四个按钮添加单击消息响应函数，并键入各个函数的代码，代码如下：

(1) “输入数据”按钮单击消息函数代码：

```
void CMy9_20Dlg::OnInputBtn()
```

```
{
```

```
    //加入代码
```

```
    CString cstrGroup,cstrInputData[5];          //输入数据为字符串
```

```
    int iGroup;                                   //组号为整数
```

```
    m_Group_Edit.GetWindowText(cstrGroup);       //得到组号字符串
```

```
    m_No1_Edit.GetWindowText(cstrInputData[0]);  //得到输入数据
```

```
    m_No2_Edit.GetWindowText(cstrInputData[1]);
```

```
    m_No3_Edit.GetWindowText(cstrInputData[2]);
```

```
    m_No4_Edit.GetWindowText(cstrInputData[3]);
```

```
    m_No5_Edit.GetWindowText(cstrInputData[4]);
```

```
    iGroup=atoi((LPCTSTR)cstrGroup);           //将组号转化为整数形式
```

```
    for(int i=0;i<5;i++)                         //将输入数据转化为浮点形式
```

```
    {
```

```
        InputData[iGroup-1].dfData[i]=atof((LPCTSTR)cstrInputData[i]);
```

```
    }
```

```
}
```

(1) “输出结果” 按钮单击消息函数代码:

```
void CMy9_20Dlg::OnResultBtn()
{
    //加入代码
    CString cstrGroup;
    int iGroup;
    double dfMax=0,dfMin=0;
    char cBuffer[10];
    m_Group_Edit.GetWindowText(cstrGroup);           //得到组号
    iGroup=atoi((LPCTSTR)cstrGroup);
    dfMax=InputData[iGroup-1].dfData[0];           //初始化最大最小值
    dfMin=InputData[iGroup-1].dfData[0];
    for(int i=0;i<5;i++)                           //求最大最小值
    {
        if(dfMax<InputData[iGroup-1].dfData[i])
        {
            dfMax=InputData[iGroup-1].dfData[i];
        }
        if(dfMin>InputData[iGroup-1].dfData[i])
        {
            dfMin=InputData[iGroup-1].dfData[i];
        }
    }
    _gcvt(dfMax,10,cBuffer);                       //将最大值转化为字符串
    m_Max_Edit.SetWindowText((LPCTSTR)cBuffer);     //将最大值字符串输出
    _gcvt(dfMin,10,cBuffer);                       //将最小值转化为字符串
    m_Min_Edit.SetWindowText((LPCTSTR)cBuffer);     //将最小值字符串输出
}
```

(3) “检索数据” 按钮单击消息函数代码:

```
void CMy9_20Dlg::OnSearchBtn()
{
    //加入代码
    CString cstrGroup;
    int iGroup;
    char cBuffer[10];
    int i;
    m_Group_Edit.GetWindowText(cstrGroup);           //得到组号
    iGroup=atoi((LPCTSTR)cstrGroup);
    //输出检索的结果
    _gcvt(InputData[iGroup-1].dfData[0],10,cBuffer); //将数据转化为字符串
    m_No1_Edit.SetWindowText((LPCTSTR)cBuffer);     //输出
}
```

```
_gcvt(InputData[iGroup-1].dfData[1],10,cBuffer);
m_No2_Edit.SetWindowText((LPCTSTR)cBuffer);
_gcvt(InputData[iGroup-1].dfData[2],10,cBuffer);
m_No3_Edit.SetWindowText((LPCTSTR)cBuffer);
_gcvt(InputData[iGroup-1].dfData[3],10,cBuffer);
m_No4_Edit.SetWindowText((LPCTSTR)cBuffer);
_gcvt(InputData[iGroup-1].dfData[4],10,cBuffer);
m_No5_Edit.SetWindowText((LPCTSTR)cBuffer);
}
```

(4) “退出” 按钮单击消息函数代码:

```
void CMy9_20Dlg::OnExitBtn()
{
    //加入代码
    OnOK();    // 退出
}
```

注意: 实验报告只需附上程序运行截图即可, 程序的标题显示自己的姓名和学号!