

## 内容：

编写一个歌曲大奖赛评分计分系统。

程序功能：

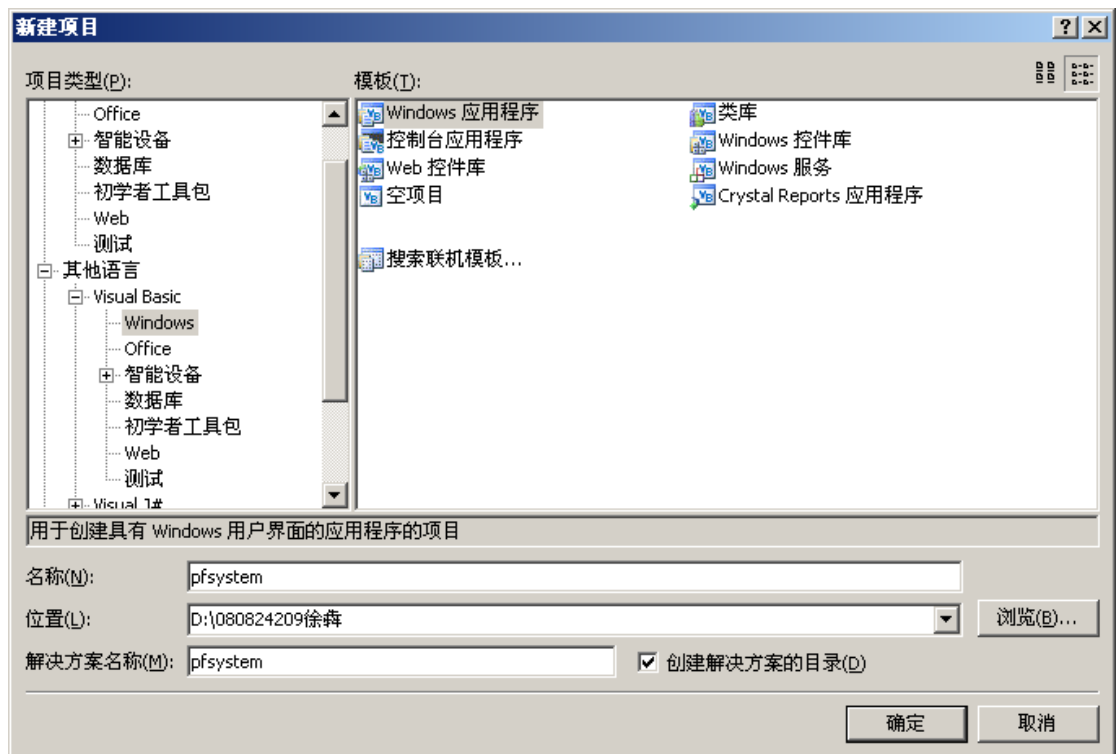
(1) 系统能首先进行有关参数的初始化工作，如输入评委的数量、参加大奖赛的选手数量、各选手编号等初始数据。

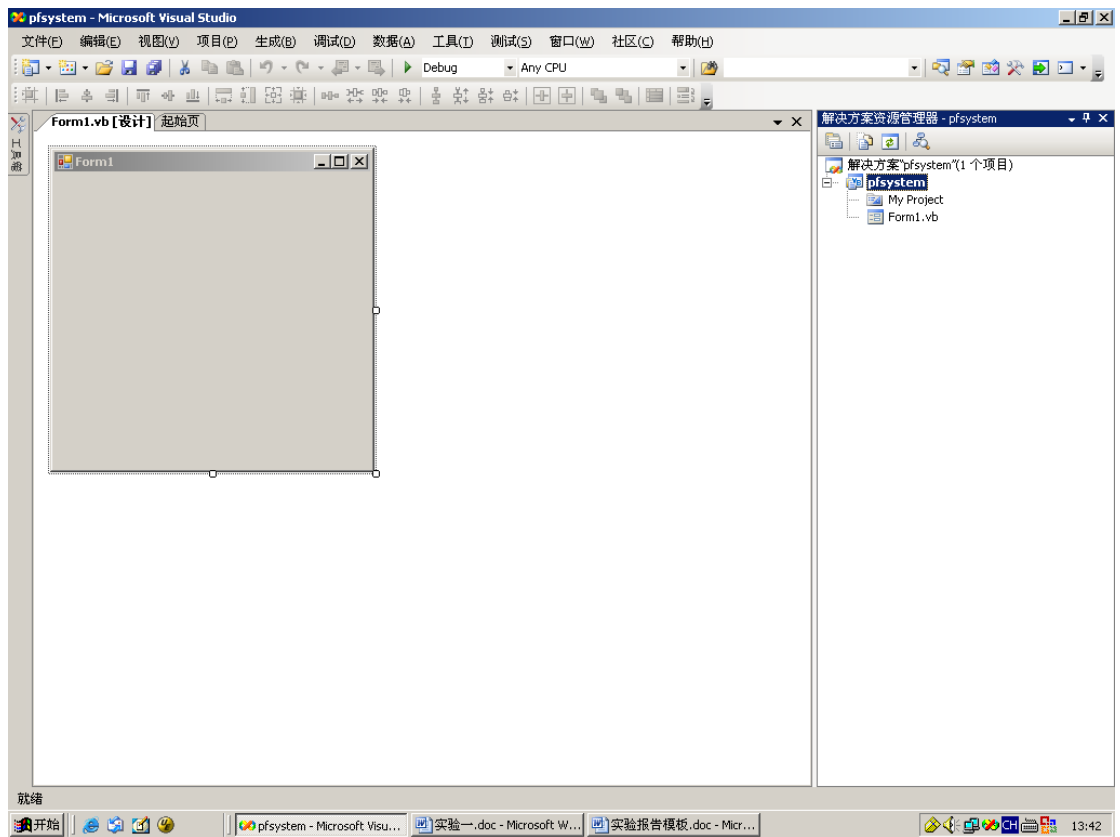
(2) 现场打分：根据主持人宣读的每一位评委的打分情况，输入各评委的打分，当输入完每个评委的打分后，去掉一个其中的最高分和其中一个最低分，显示所有评委的打分和其中的最低分、最高分以及选手的最后得分。

(3) 待所有选手参赛完毕后，显示一个列表，该列表中显示每一位歌手的编号及最后得分，并按各选手的最后得分从高到低排序。

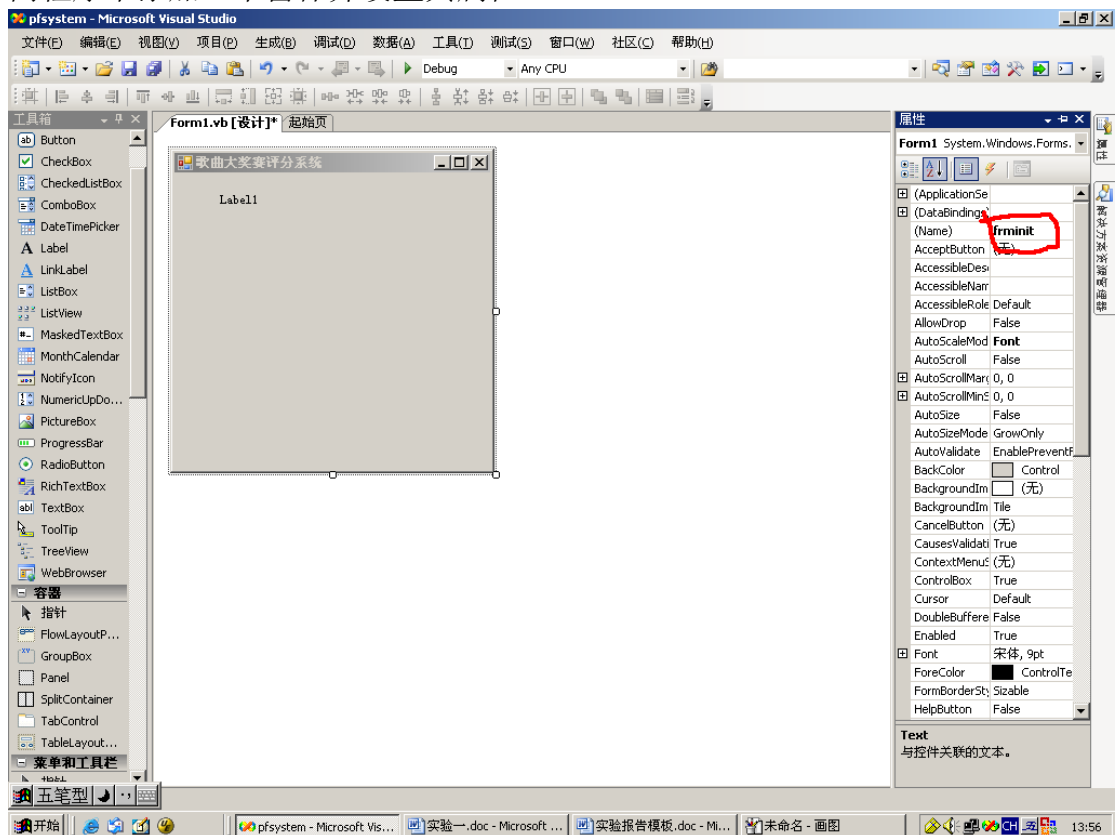
## 步骤：

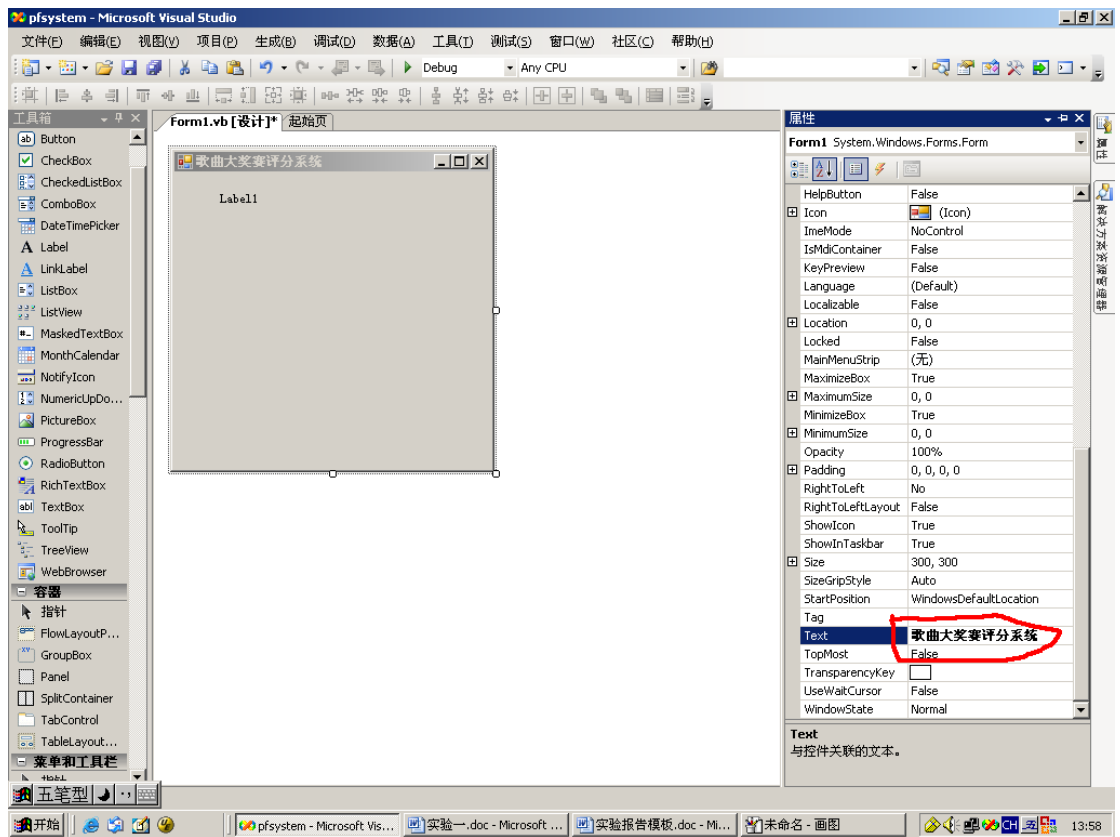
### 1. 新建一个 windows 项目



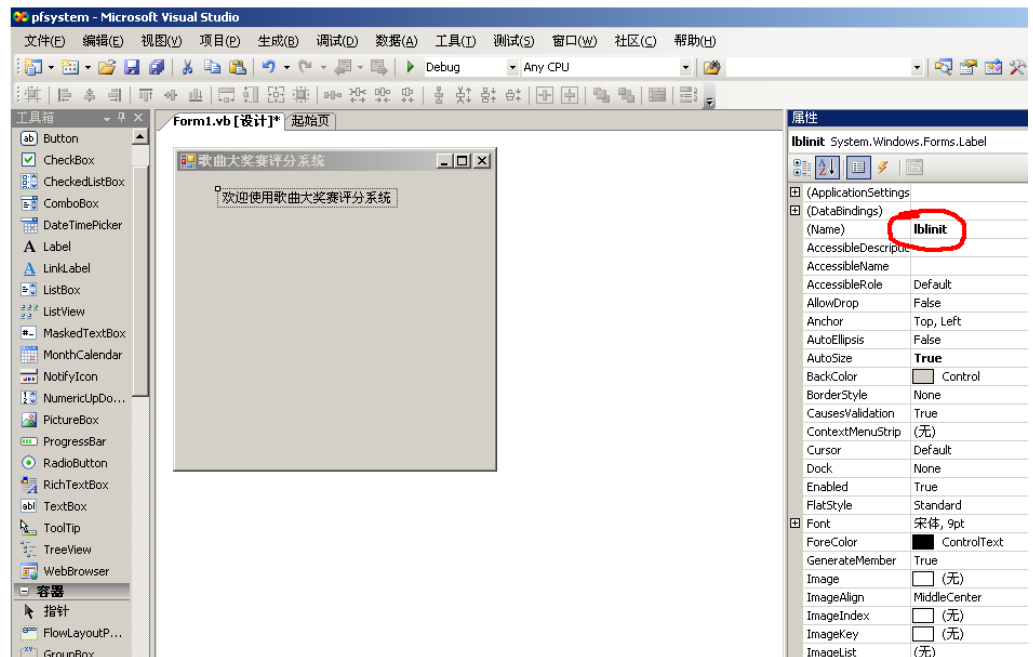


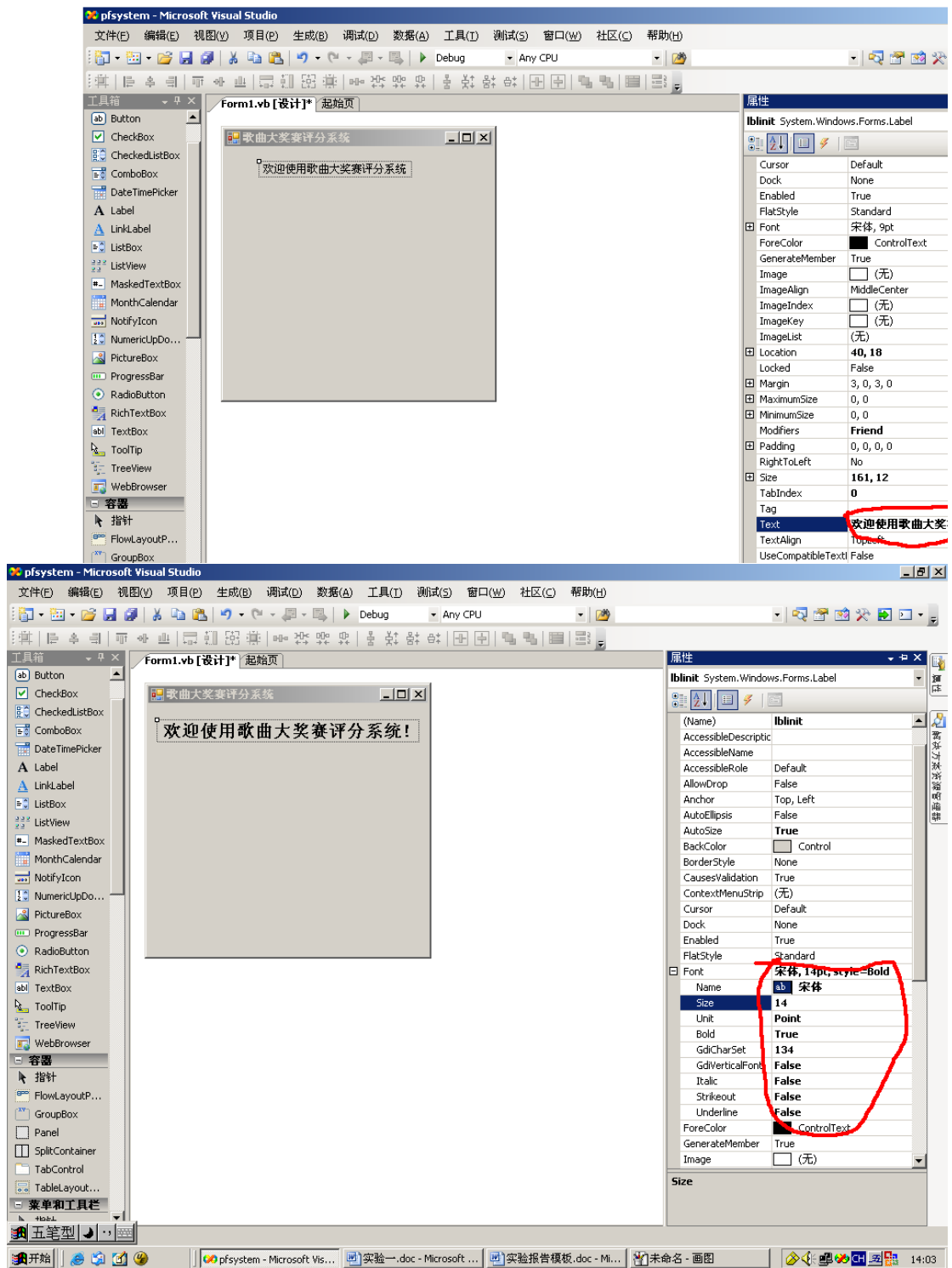
## 2. 向程序中添加一个窗体并设置其属性



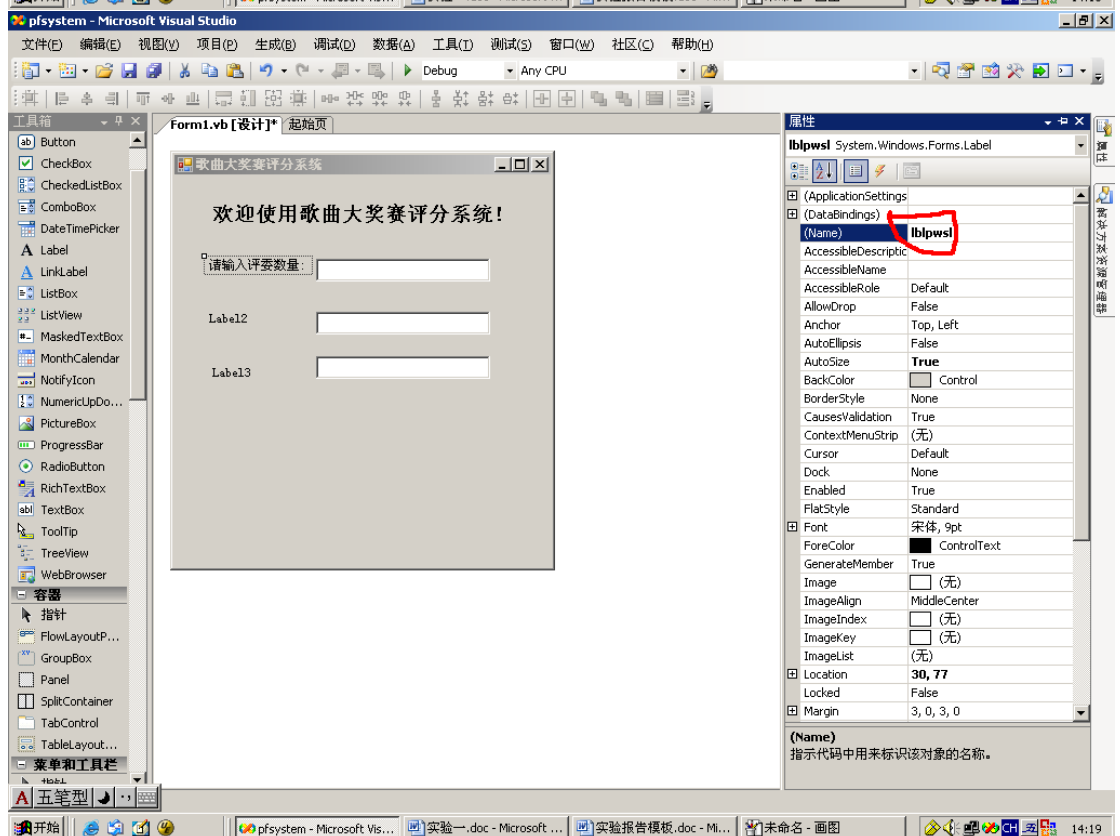
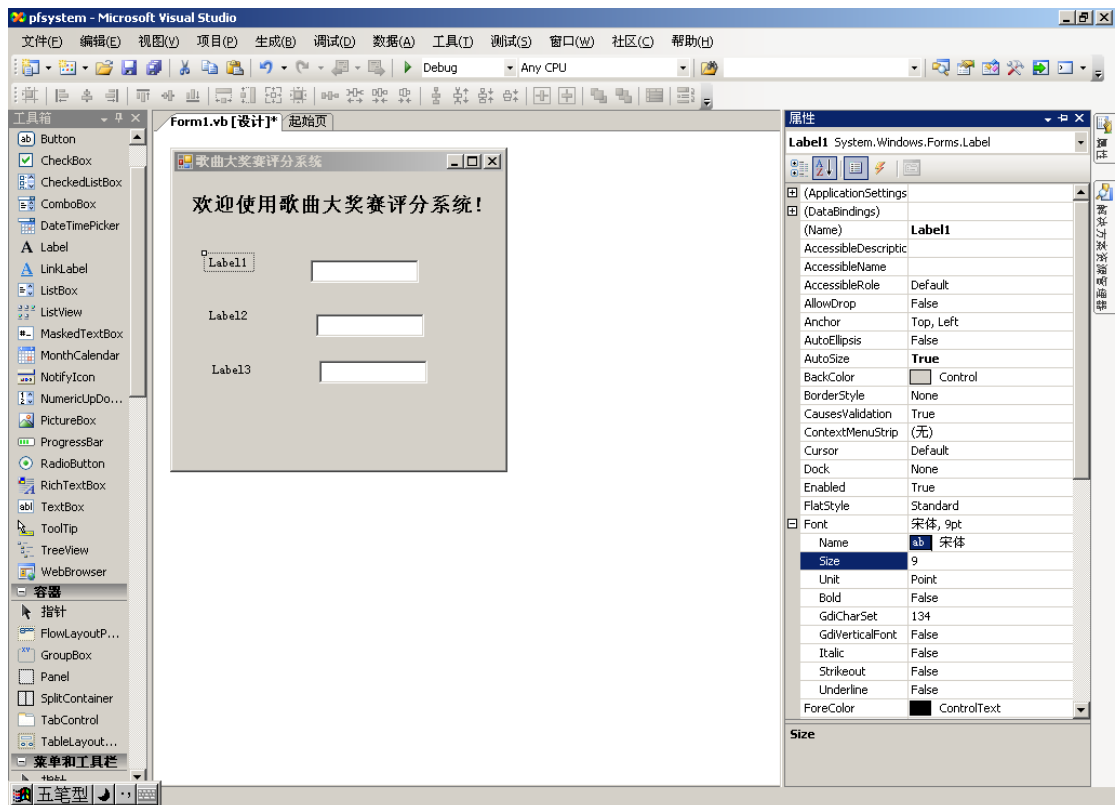


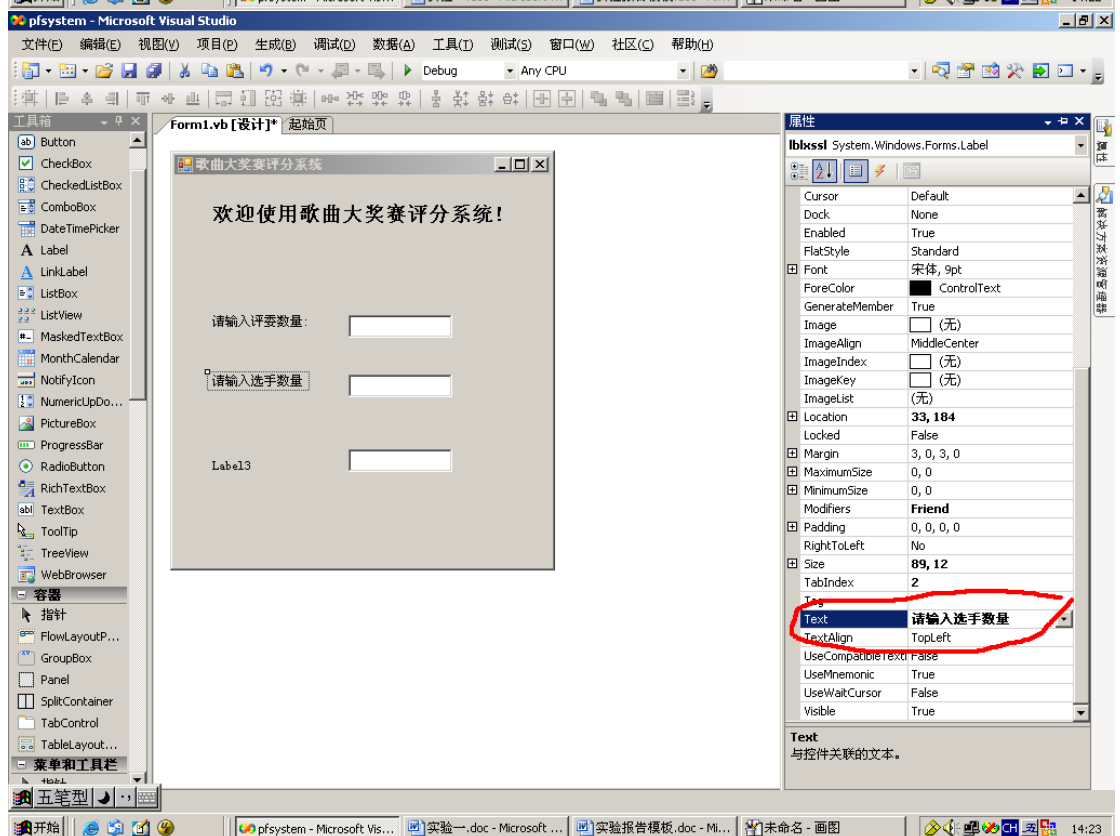
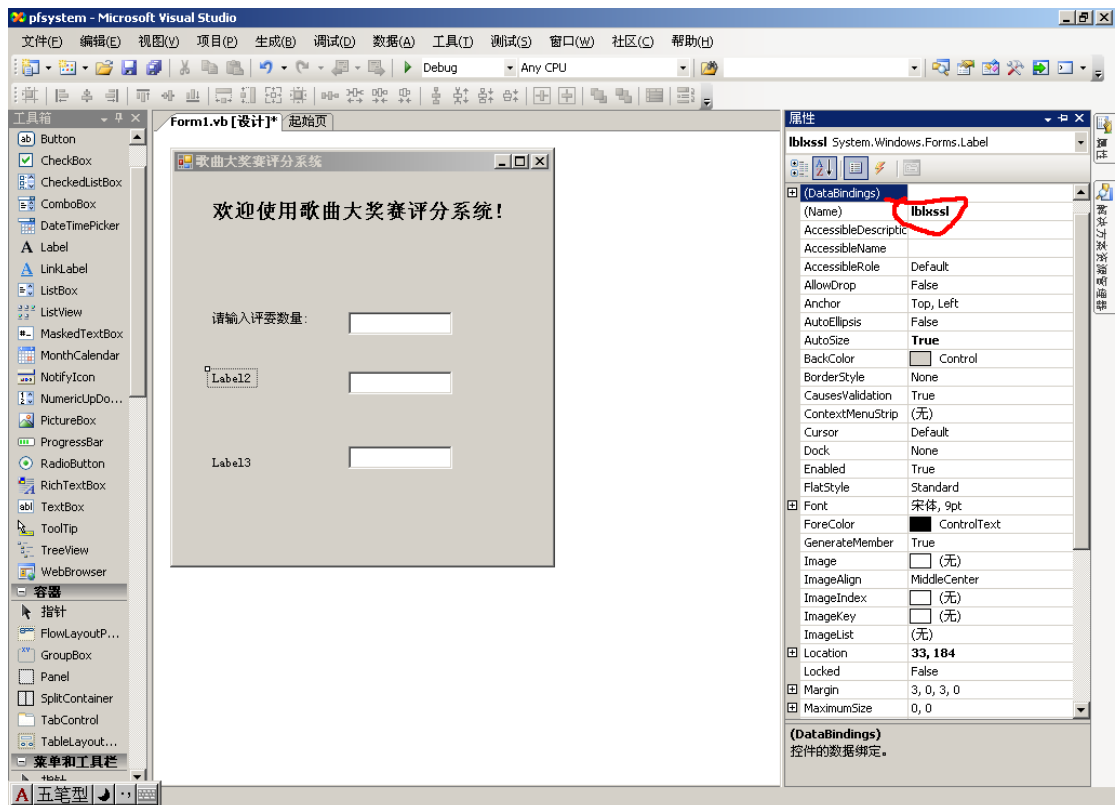
添加一个标签并设置属性，用于显示欢迎信息



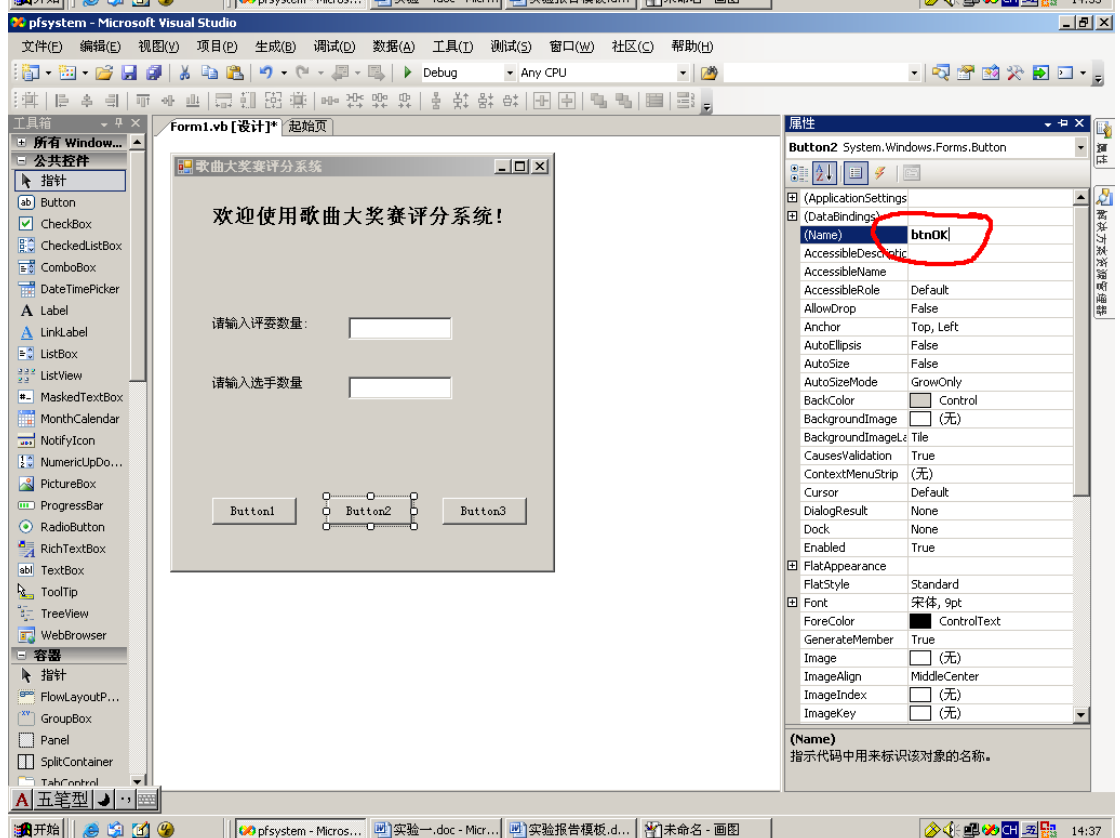
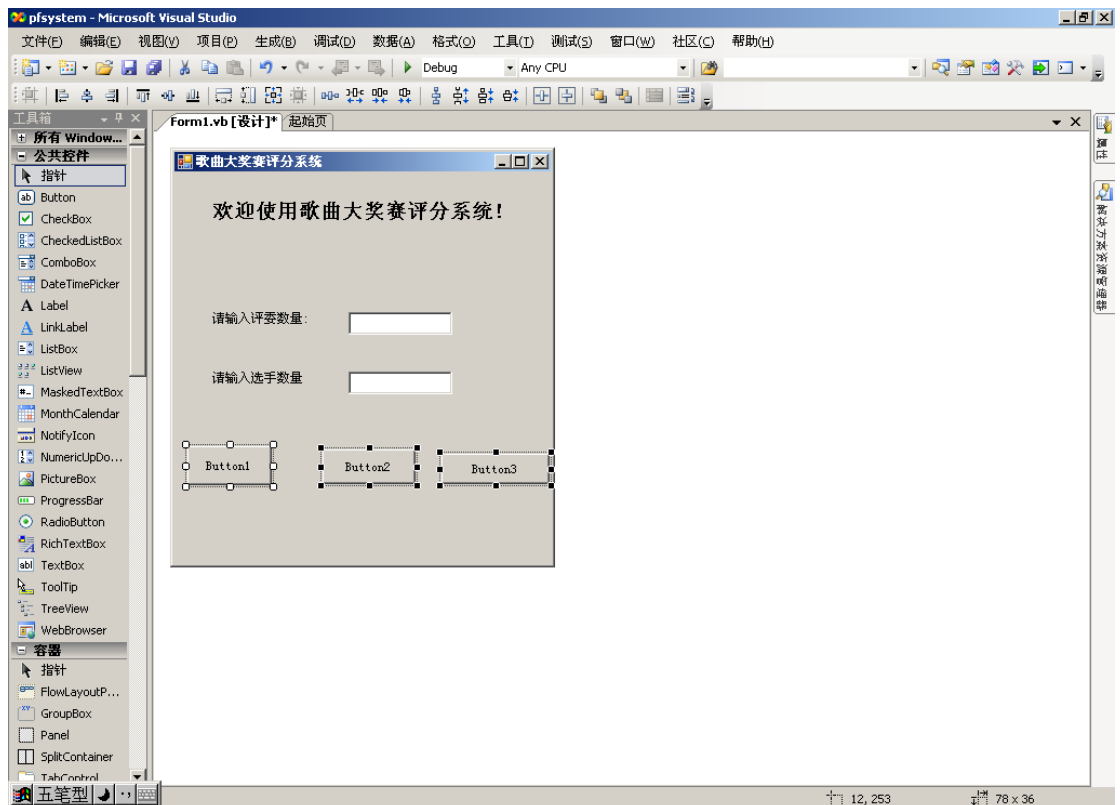


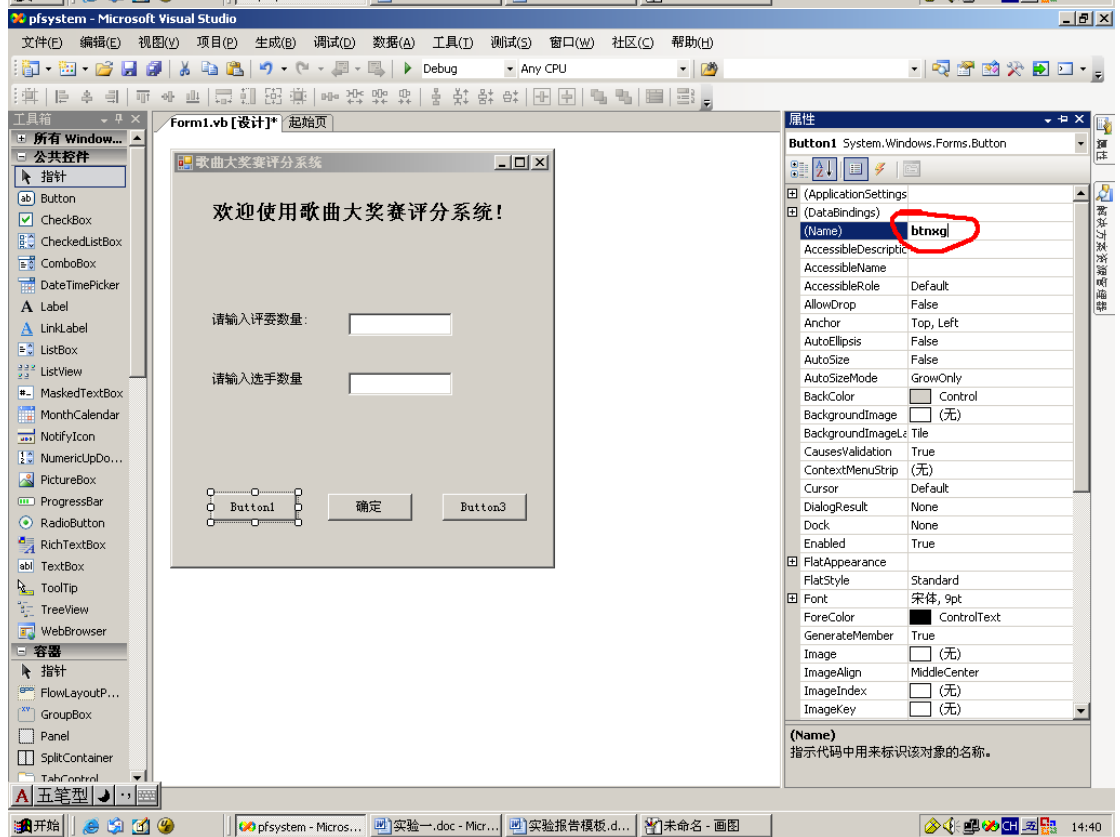
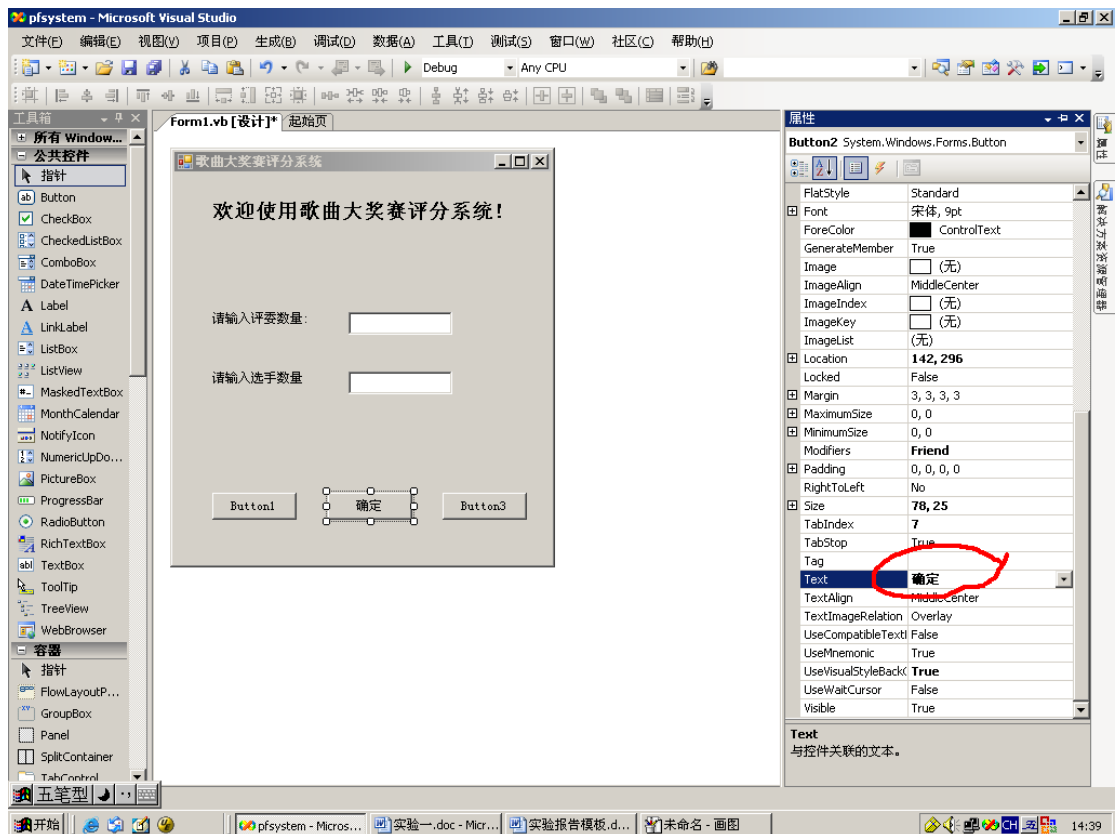
再添加几个标签和文本框控件用于提示并让用户输入初始数据，如下图



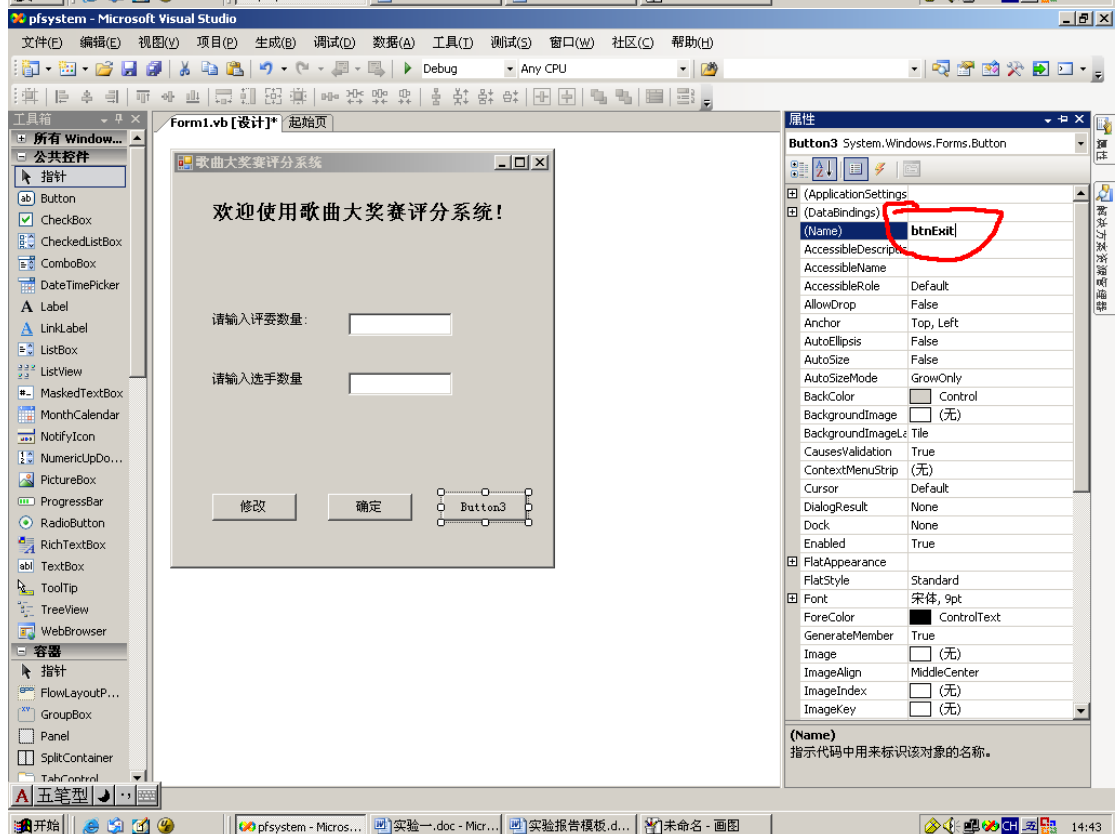
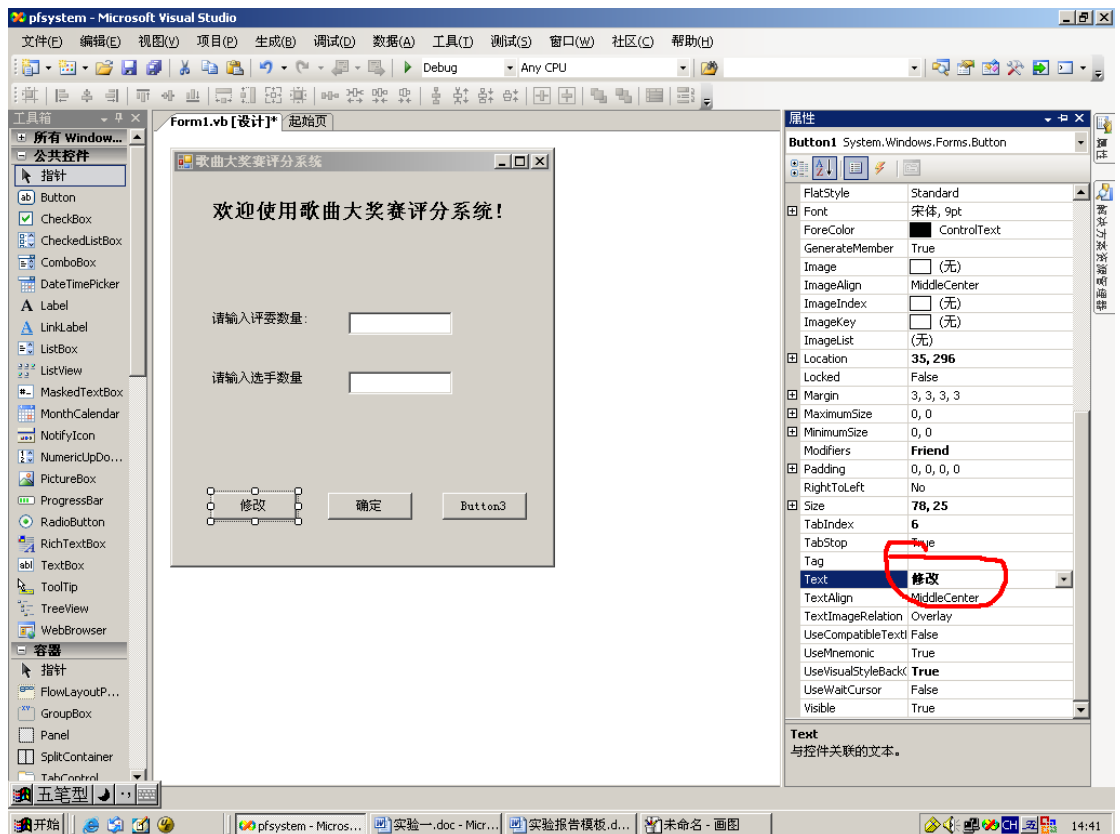


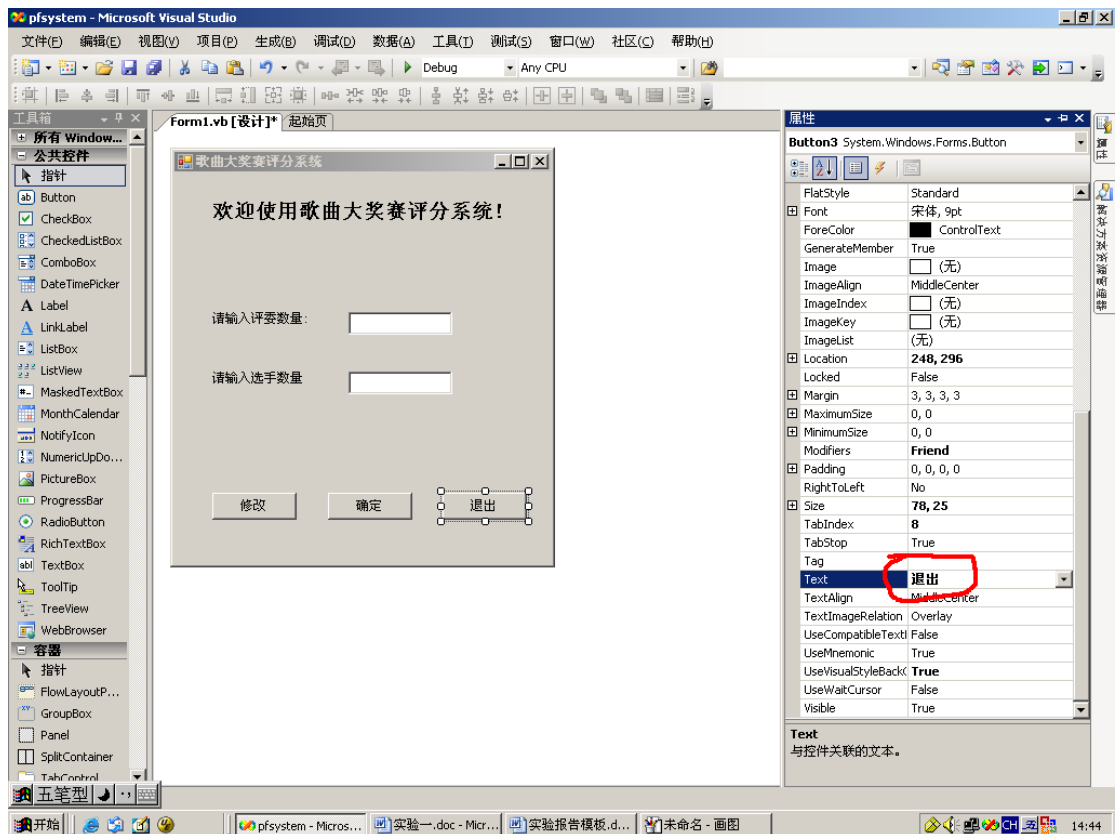
考虑到本实验初始数据只要评委数量和选手数量两个参数即可，故将下面剩余的一个标签和一个文本框删掉，改为三个按钮：“确定”，“修改”和“退出”



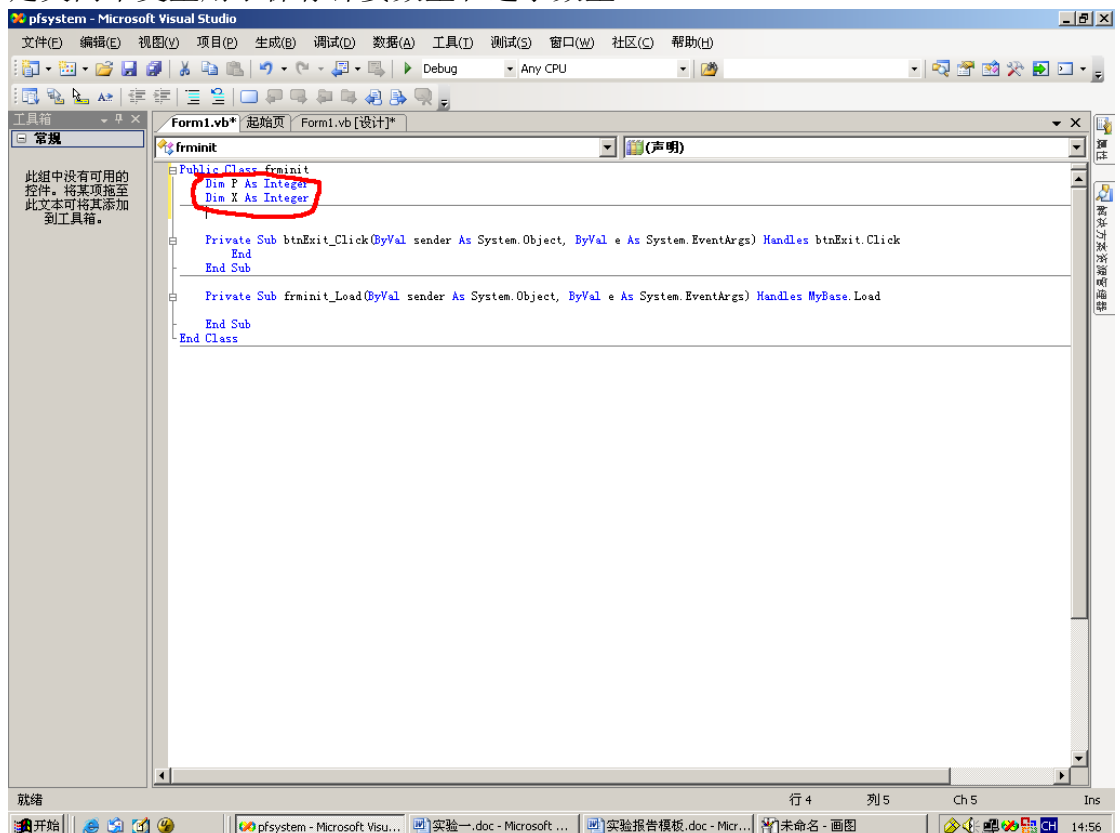








3. 编写相应代码使窗体中的按钮具有相应的功能:  
定义两个变量用于保存评委数量和选手数量



写代码实现当用户点击确定按钮时将用户在文本框中输入的评委数量和选手数量赋给相应的变量保存起来, 当用户点击修改按钮时将两个文本框

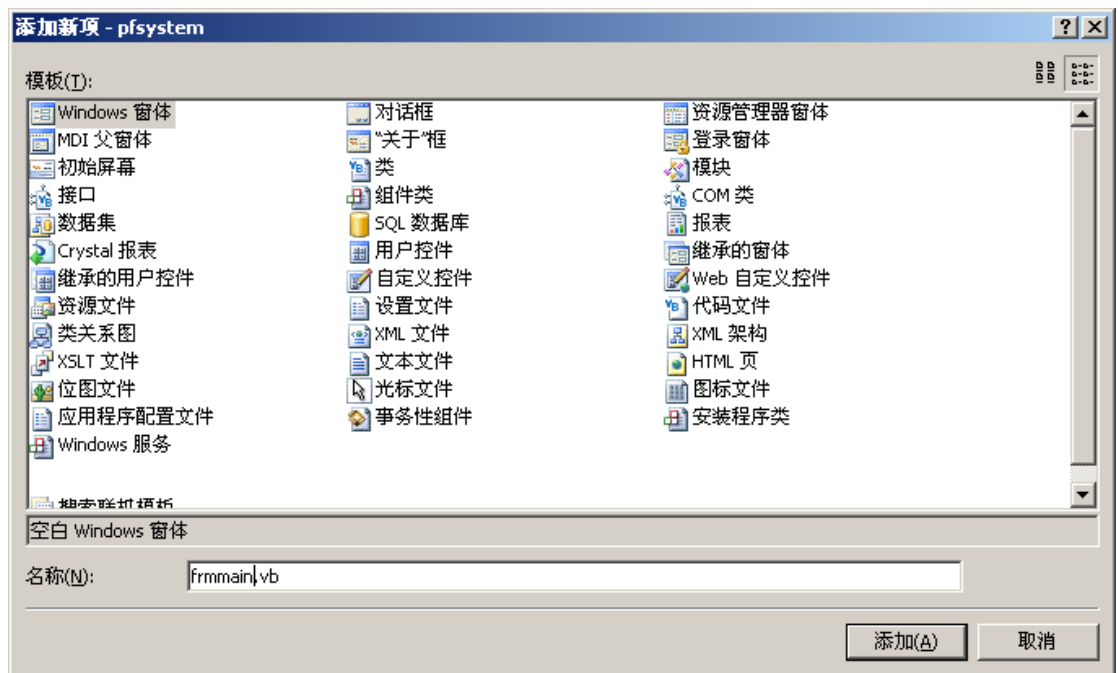
中的内容清空，并将光标停到第一个文本框中，当用户点击退出按钮时退出程序，相应代码如下

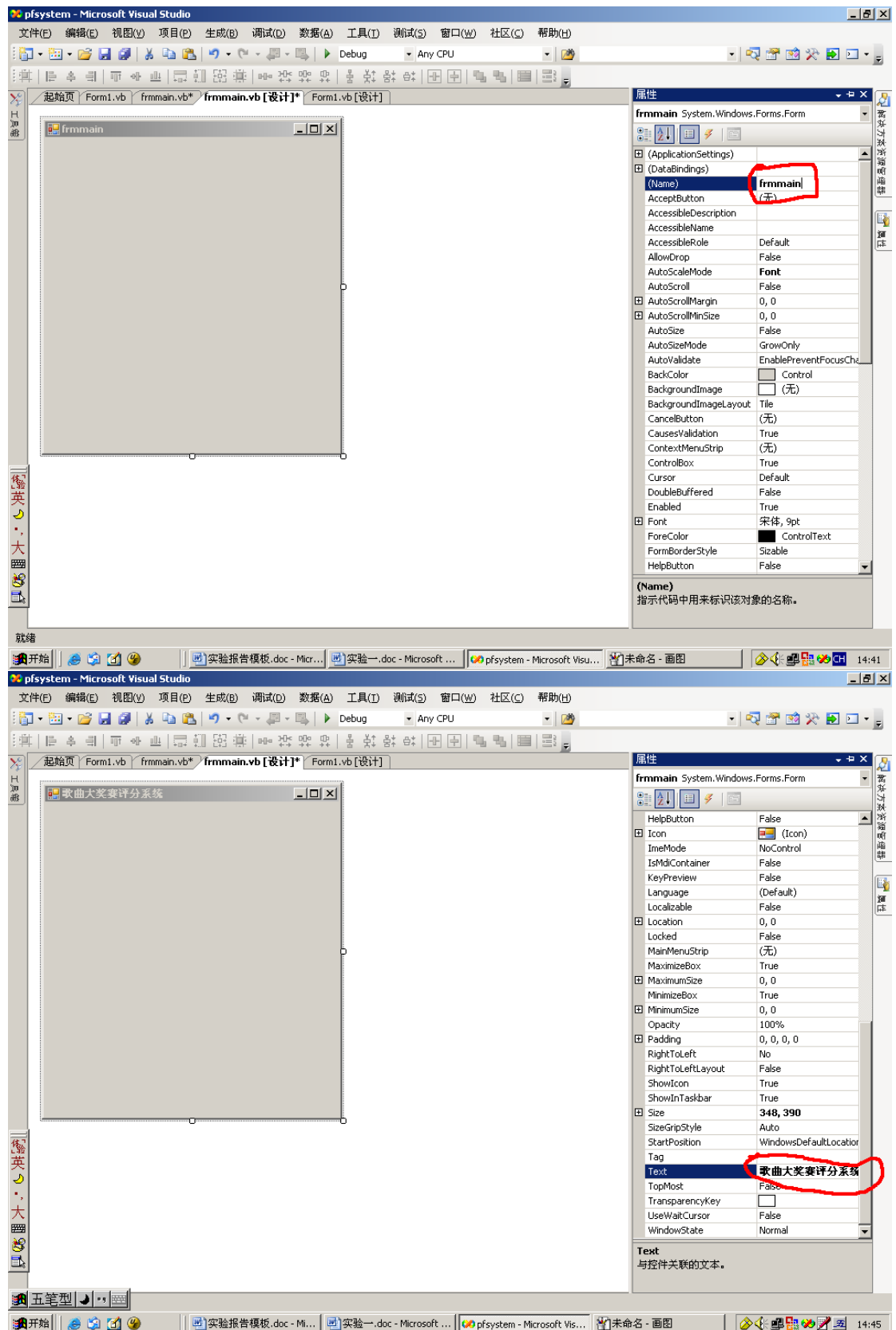
```
Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles btnExit.Click  
    End  
End Sub
```

```
Private Sub btnOK_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles btnOK.Click  
    P = Val(txtPW.Text)  
    X = Val(txtXS.Text)  
End Sub
```

```
Private Sub btnxg_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles btnxg.Click  
    txtPW.Text = ""  
    txtXS.Text = ""  
    txtPW.Focus()  
End Sub
```

4. 向窗体中再添加一个窗体，用于打分和显示结果



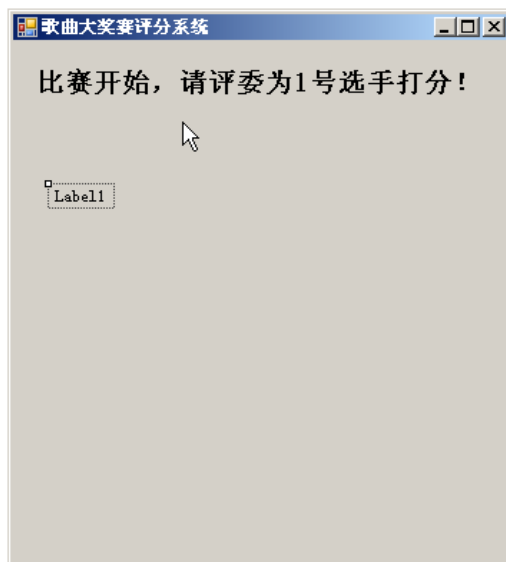
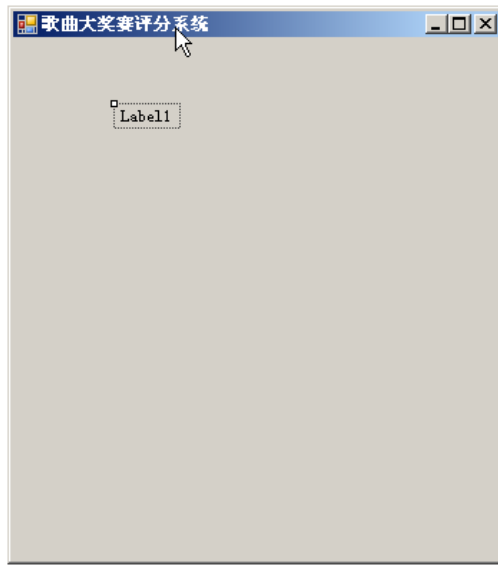


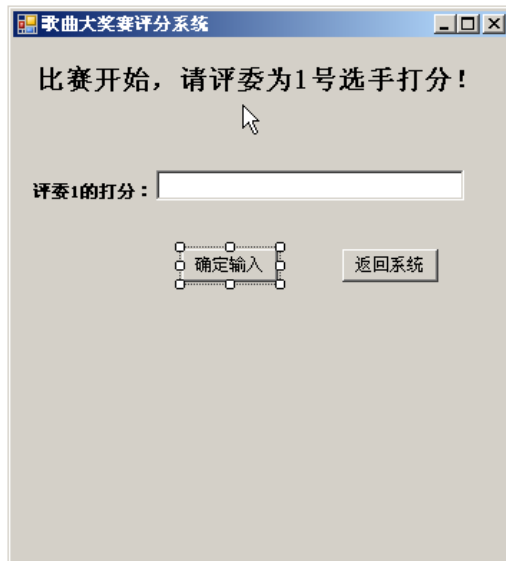
5. 在第一个窗体的确定按钮的消息处理函数中添加代码使用户点击确定时将第一个窗体隐藏，将第二个窗体显示出来  
代码如下：

```
frmmain.Show()
```

Me.Hide()

6. 设计第二个窗体的界面，使其能提示并接收用户输入的数据，过程如下





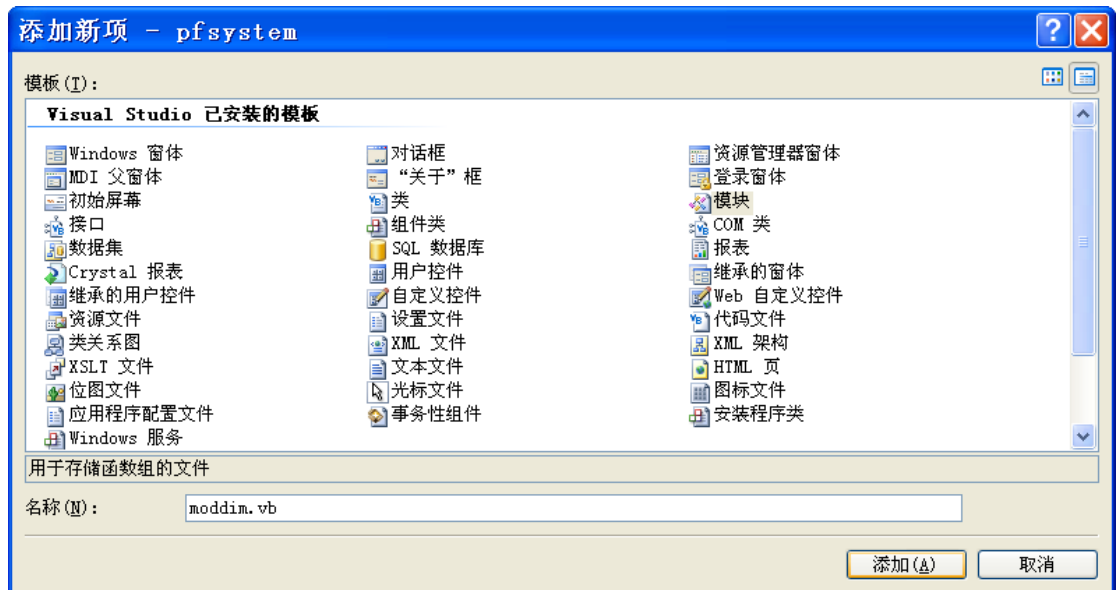
7. 编写代码使用户点击返回系统按钮时关闭评分窗体，重新显示初始化窗体，代码如下

```
Private Sub btnExit2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnExit2.Click
    frminit.Show()
    Me.Close()
End Sub
```

8. 考虑到此程序中已经有一个以上窗体，故将原来定义的两个用于存放初始化数据的变量重新定义成全局型变量，并定义一个全局型数组用于存放评委打分，代码如下

```
Public PW As Integer
Public XS As Integer
Public dafen(,) As Integer
```

9. 在第二个窗体的 load 事件中添加代码重新定义全型局数组的大小可是这时候出现错误，虽然上面三个变量已经定义成全局型，但在重新定义数组大小时，系统提示上述三个变量未声明，所以只好为程序添加一个模块用于全局变量的声明



将声明变量的代码拷到模块中

```
Module moddim
    Public PW As Integer
    Public XS As Integer
    Public dafen( ) As Integer
End Module
```

结果问题解决。

10. 接着试图编写代码将用户输入的分数的赋给数组元素，根据程序要求进行计算，并输出结果。但发现如果只用一个按钮控制用户输入很难做到，为此，又向窗体中添加一个按钮。然后又在模块中定义一些变量，相应代码如下

```
Module moddim
    Public PW As Integer
    Public XS As Integer
    Public dafen(.) As Integer
    Public a As Integer
    Public b As Integer
    Public max As Integer
    Public min As Integer
    Public defen() As Single
    Public i As Integer
    Public j As Integer
    Public t As Single
    Public u As Integer

End Module

Public Class frmmain
    Private Sub frmmain_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        ReDim dafen(XS - 1, PW - 1)
        ReDim defen(XS - 1)
        a = 0
        b = 0
    End Sub

    Private Sub btnExit2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit2.Click
        frminit.Show()
        Me.Close()
    End Sub

    Private Sub btnOK2_Click ...
    End Sub

    Private Sub btnContinue_Click ...
    End Sub
End Class
```

其中算法的关键在两个按钮 btnOK2 和 btnContinue 的消息处理函数中分别为：

```
Private Sub btnOK2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnOK2.Click
```

```
    If a < XS Then
```

```
        If b > PW - 1 Then
```

```
            b = b - 1
```

```
            max = dafen(a, 0)
```

```
            min = dafen(a, 0)
```

```
            For i = 1 To b
```

```
                If dafen(a, i) > max Then
```

```
                    max = dafen(a, i)
```

```
                End If
```

```
                If dafen(a, i) < min Then
```

```
                    min = dafen(a, i)
```

```
                End If
```

```
            Next i
```

```
            For i = 0 To b
```

```
                defen(a) = defen(a) + dafen(a, i)
```



```

        Next i
        defen(a) = (defen(a) - max - min) / (PW - 2)
        lblJG.Text = "评委给" & a + 1 & "号选手的打分分别为:" &
vbCrLf
        For i = 0 To b
            lblJG.Text = lblJG.Text & dafen(a, i) & "  "
        Next i
        lblJG.Text = lblJG.Text & vbCrLf & "最低分为:" & vbCrLf &
min & vbCrLf & "最高分为:" & vbCrLf & max & vbCrLf & "最后得分为:" &
defen(a) & vbCrLf
        a = a + 1
        b = 0
        txtscore.Text = ""
        txtscore.Focus()

    Else
        dafen(a, b) = Val(txtscore.Text)
        b = b + 1
        txtscore.Text = ""
        txtscore.Focus()
        If b < PW Then
            lblPW.Text = "评委" & b + 1 & "的打分:"
        Else
            lblPW.Text = "          "
        End If
    End If
End If
End If
End Sub
和
Private Sub btnContinue_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnContinue.Click
    If a > XS - 1 Then
        Dim temp(XS - 1) As Integer
        For i = 0 To XS - 1
            temp(i) = i
        Next i
        For i = 1 To XS
            For j = 0 To XS - 1 - i
                If defen(j) < defen(j + 1) Then
                    t = defen(j + 1)
                    defen(j + 1) = defen(j)
                    defen(j) = t
                    u = temp(j + 1)

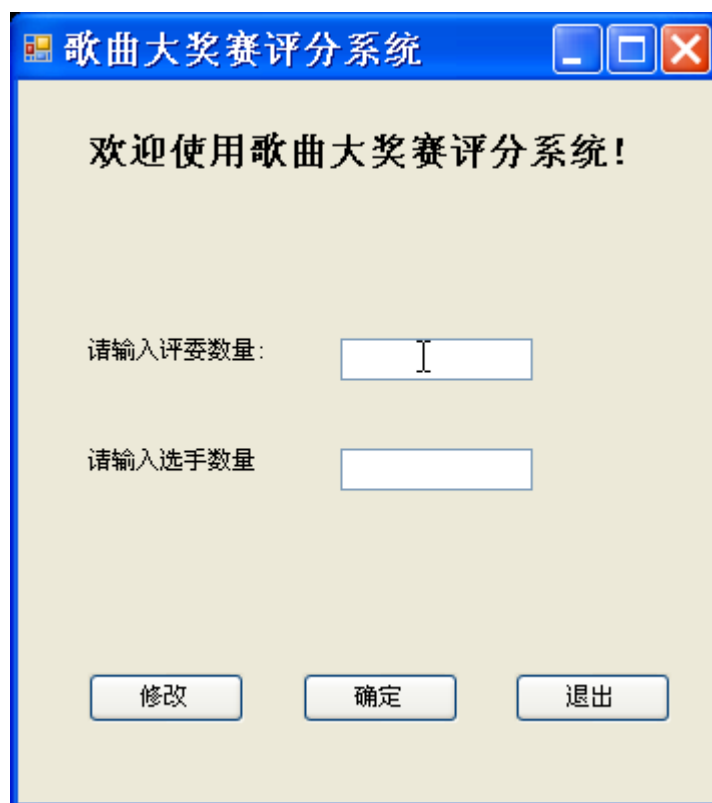
```





```

temp(j + 1) = temp(j)
temp(j) = u
End If
Next j
Next i
lblJG.Text = "各选手的得分分别为:" & vbCrLf
For i = 0 To XS - 1
    lblJG.Text = lblJG.Text & temp(i) + 1 & "号选手得分为: " &
defen(i) & vbCrLf
Next i
Else
    lblJG.Text = ""
    lblTS.Text = "请评委为" & a + 1 & "号选手打分!"
    lblPW.Text = "评委" & b + 1 & "的打分:"
    txtscore.Focus()
End If
End Sub

```

11. 完成后编译运行程序，可以完成程序要求的基本功能，运行过程如下列截图：







 歌曲大奖赛评分系统   

欢迎使用歌曲大奖赛评分系统！

请输入评委数量：

请输入选手数量

 歌曲大奖赛评分系统   

请评委为1号选手打分！

评委1的打分：

歌曲大奖赛评分系统

请评委为1号选手打分！

评委1的打分：

歌曲大奖赛评分系统

请评委为1号选手打分！

评委2的打分：


歌曲大奖赛评分系统

请评委为1号选手打分！

评委4的打分：

歌曲大奖赛评分系统

请评委为1号选手打分！

 歌曲大奖赛评分系统


请评委为1号选手打分！

输入

继续

返回

评委给1号选手的打分分别为：  
5 9 7 3  
最低分为：  
3  
最高分为：  
9  
最后得分为:6

 歌曲大奖赛评分系统

请评委为2号选手打分！

评委1的打分:

输入

继续

返回

歌曲大奖赛评分系统

请评委为2号选手打分!

输入

继续

返回

评委给2号选手的打分分别为:  
9 9 8 6  
最低分为:  
6  
最高分为:  
9  
最后得分为:8.5

歌曲大奖赛评分系统

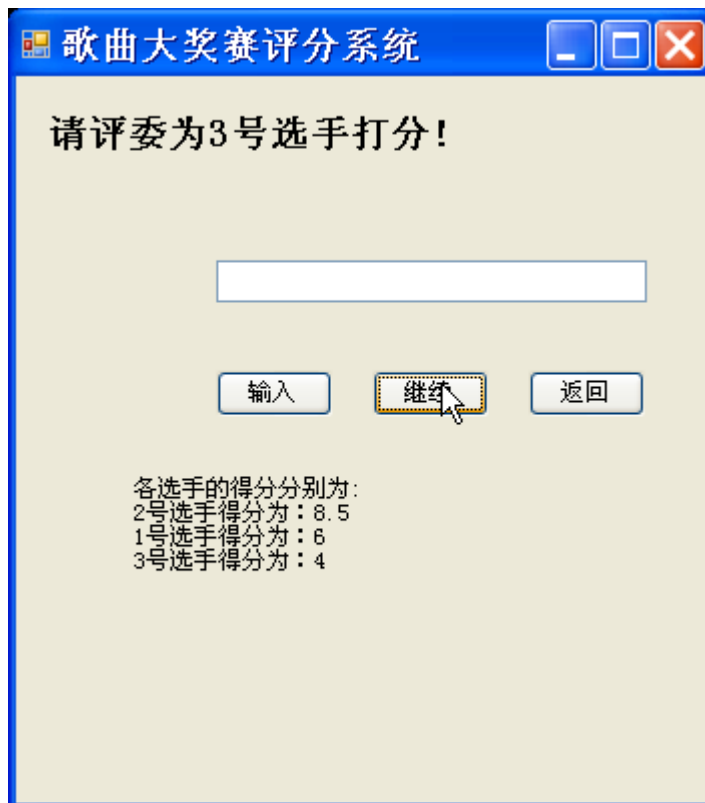
请评委为3号选手打分!

输入

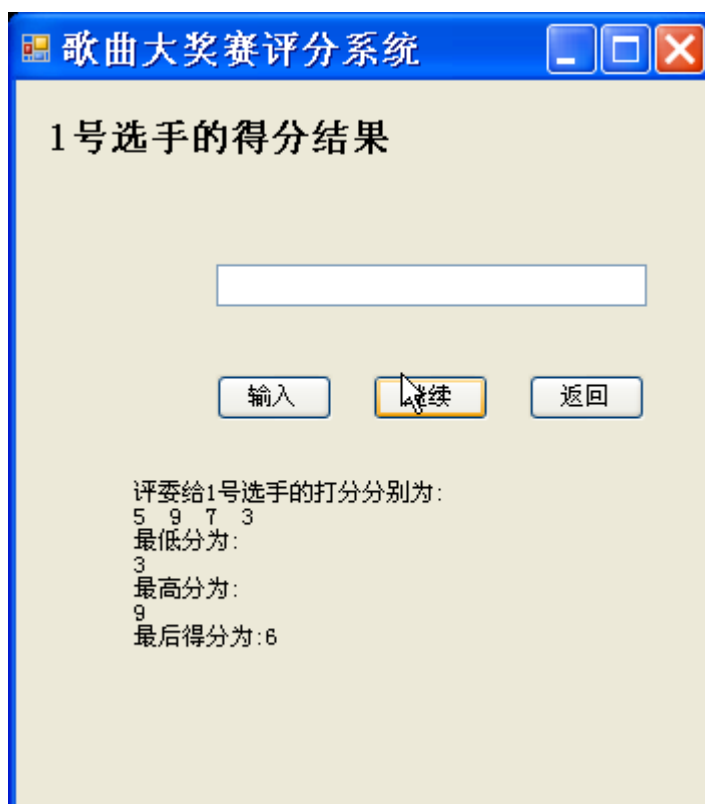
继续

返回

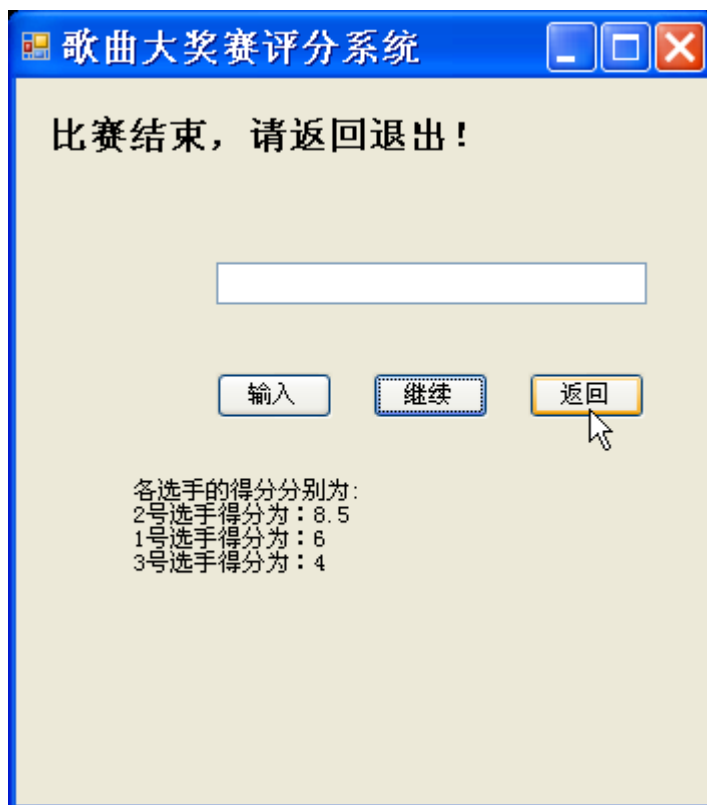
评委给3号选手的打分分别为:  
2 3 5 6  
最低分为:  
2  
最高分为:  
6  
最后得分为:4



12. 完成后再继续完善程序, 修改代码使输入完所有打分后在窗体中显示提示信息, 修改后程序运行结果如下







13. 再继续调试，发现程序还有一点小毛病，就是当用户输入完评委给一个选手的所有打分后应该点击“继续”按钮，如果误点了“确认”就会发生错误，故修改程序代码，使其在该情况下不产生响应，即在“确认”按钮的消息处理函数中修改代码，当用户输入完评委给某一位选手的所有打分并输出结果后，将控制变量  $b$  的值置为  $-1$ ，而在“继续”按钮处将  $b$  重新置  $0$  并在前面的判断处加上对  $b$  的符号的判断，完成后运行结果令人满意
14. 继续调试，发现程序还有毛病，就是如果用户没有在文本框中输入数据，即文本框为空时，如果用户点击输入按钮，程序就会默认将  $0$  存入打分数组中，这不符合要求，故修改代码，使文本框为空时输入按钮不响应，即在确定按钮消息处理函数的开始加一个判断语句
15. 同样道理，在第一个窗体的确定按钮中也添加这样一个判断语句避免用户未输入初始化数据就运行程序
16. 再调试，发现当用户输入完所有数据并正确显示结果以后，如果用户仍点击继续按钮，则会修改正确的显示结果，这不符合要求，于是修改继续按钮的代码使其不对这种情况下的点击事件发生响应
17. 完成后进一步完善程序界面，使第二个窗体初始位置在屏幕居中，将窗体的最大化、最小化按钮设置为无效，并将窗体背景设置成红色，文字设为黄色，相应设置如下列截图

AutoValidat	EnablePrever
BackColor	<span style="background-color: red;"> </span> <b>Red</b>
BackgroundI	<span style="background-color: white;"> </span> (无)
BackgroundImageLayout	
CancelButton	(无)
CausesValid	True
ContextMenu	(无)
ControlBox	True
Cursor	Default
DoubleBuffe	False
Enabled	True
<b>Font</b>	<b>黑体, 12</b> <span style="border: 1px solid black; padding: 0 2px;">...</span>
ForeColor	<span style="background-color: yellow;"> </span> <b>Yellow</b>
FormBorderS	Sizable
HelpButton	False
<b>Icon</b>	<span style="background-color: #cccccc; border: 1px solid black; display: inline-block; width: 15px; height: 15px;"></span> (Icon)
ImeMode	NoControl
IsMdiContai	False
KeyPreview	False
Language	(Default)
Localizable	False
<b>Location</b>	0, 0
Locked	False
MainMenuStr	(无)
MaximizeBox	<b>False</b>

字体
?
✕

字体 (F):

宋体  
宋体  
宋体-PUA  
微软雅黑  
新宋体  
中國能海報體  
中國能流書體  
中國能行書體

字形 (Y):

粗体  
常规  
斜体  
粗体  
粗斜体

大小 (S):

四号  
小三  
四号  
小四号  
小五号  
小六号  
小六

确定

取消

效果

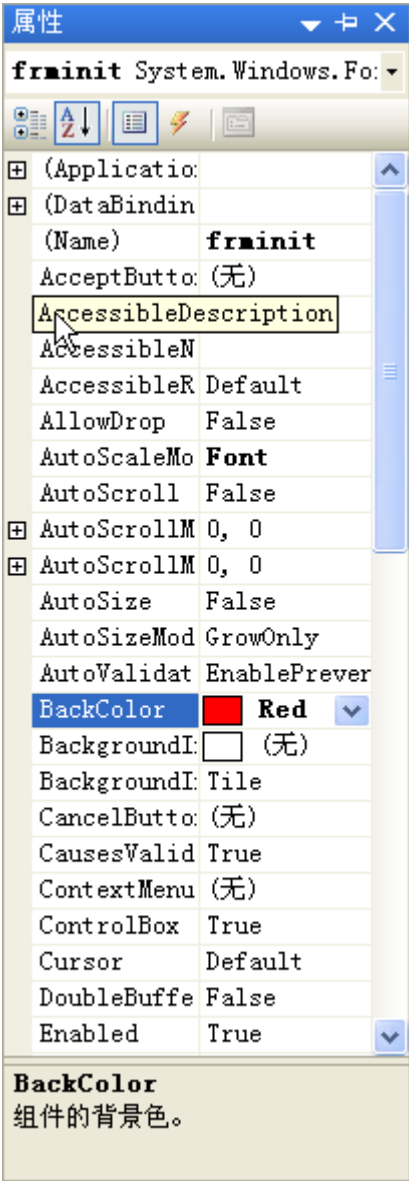
☐ 删除线 (X)  
☐ 下划线 (U)

示例

微软中文软件

字符集 (C):

CHINESE\_GB2312





18. 至此，程序编写完成，最后，为了方便他人和自己以后阅读代码，在代码中添加适当的注释。
19. 完成。