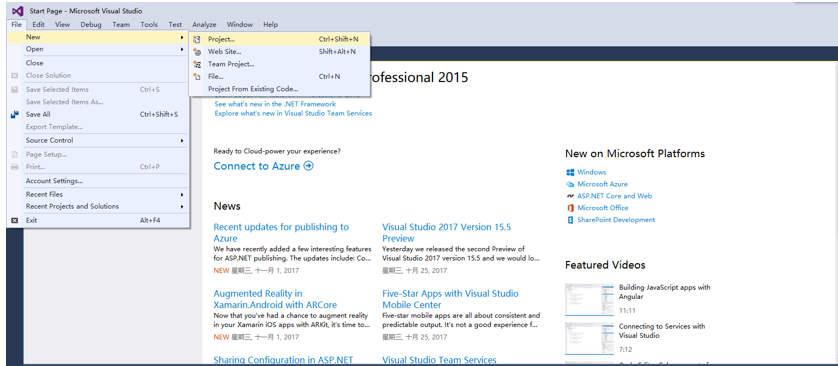
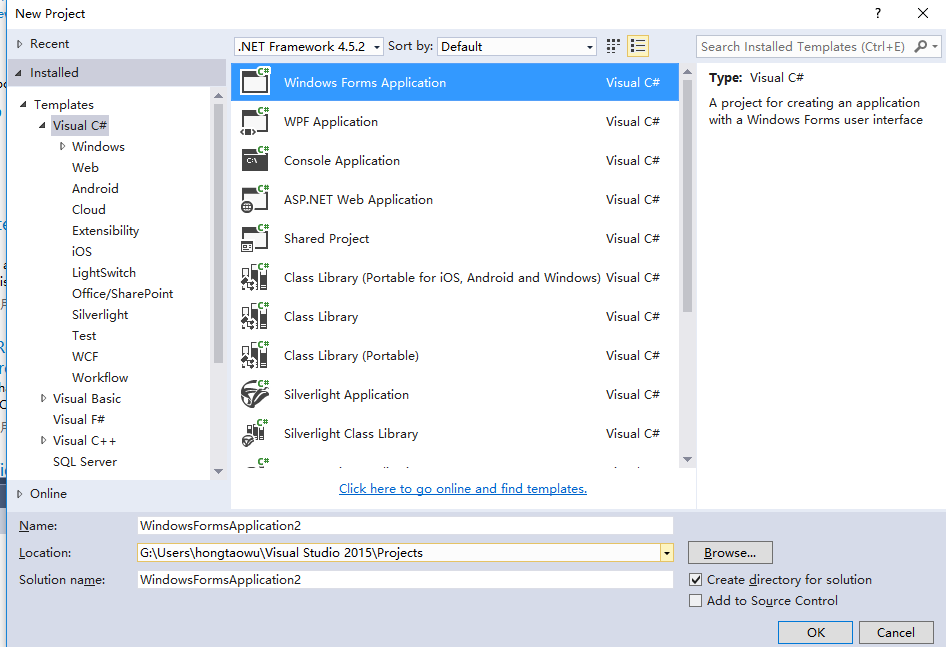
## 实验一 计算器

1、新建Windows Forms application 类型的project

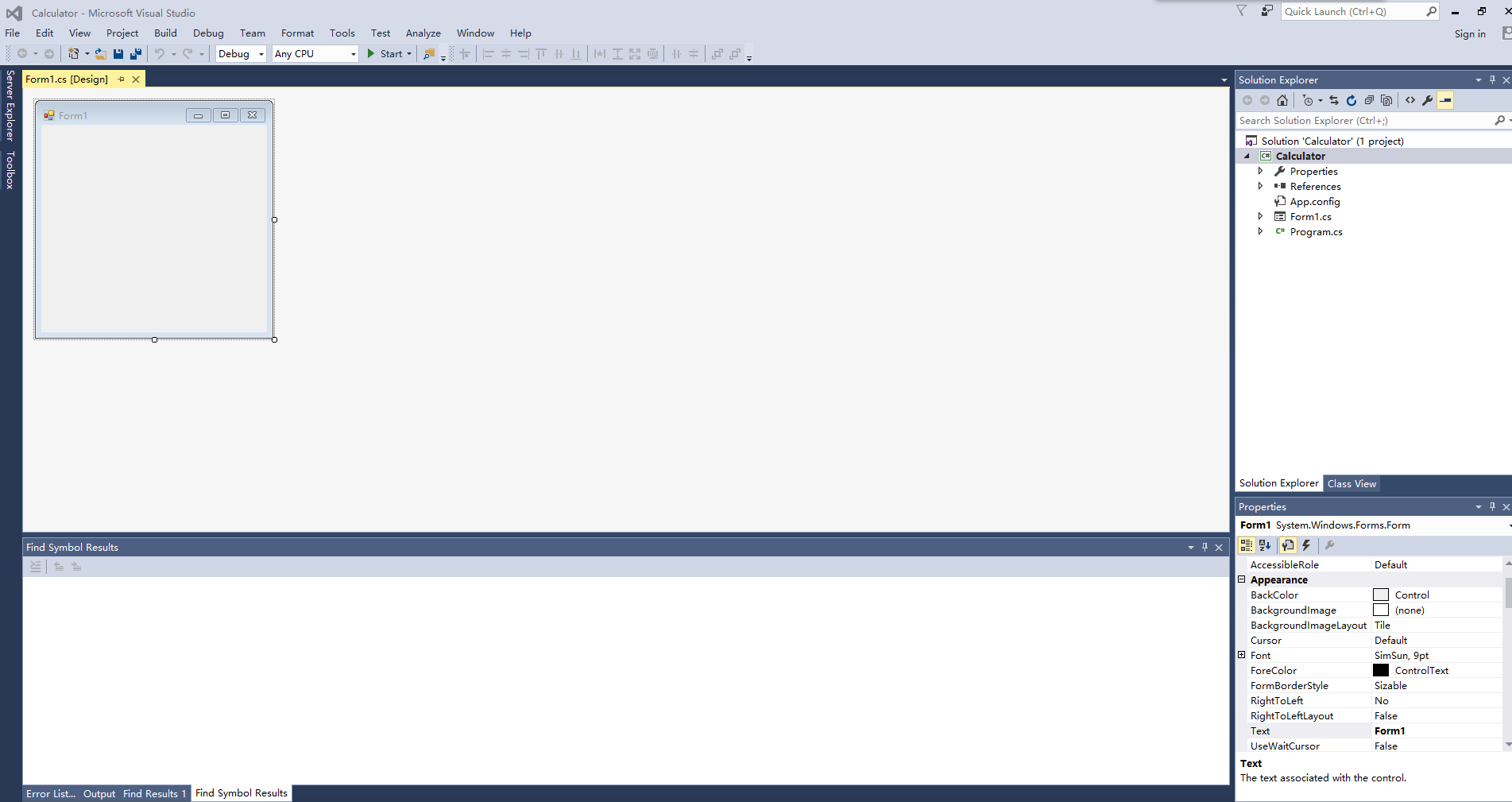
（1）File🡺new🡺project



（2）新建project（Windows Forms application）



（3）结果



左上窗口：当前，用于编辑前段的界面；后续，在该窗口相关成员函数中添加代码

右上窗口：clas view

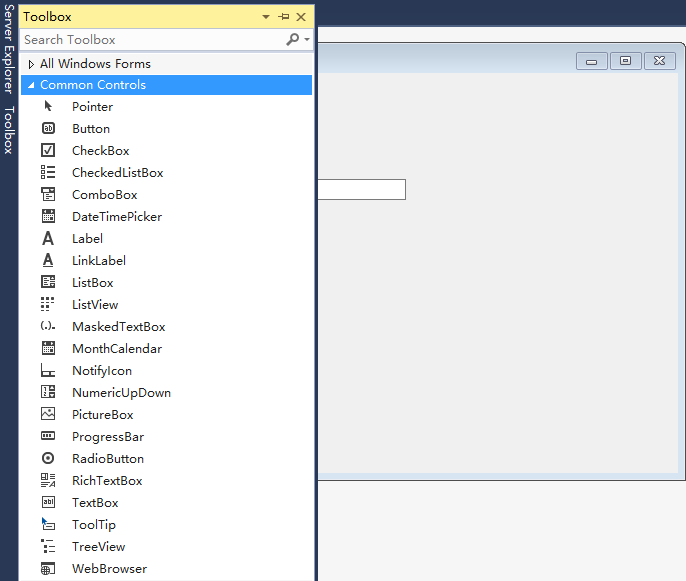
左下窗口：error list，局部变量的监控

右下窗口：修改控件的属性

Ps：可以用鼠标左键选中Form1的左下角，进行拖动，实现放大/缩小Form的目标

2、界面设计

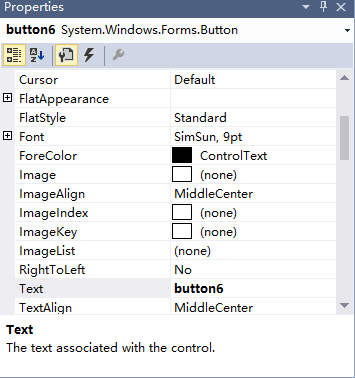
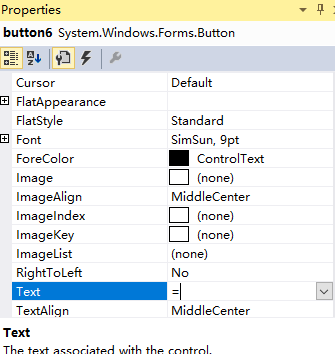
（1）添加控件：点击如下图所示最左侧的Toolbox，选择“Common controls”，把相应控件拖到Form1上（选中相应控件，按住鼠标左键，把选中的控件往Form1上拖）



静态文本框，显示文本，

文本框：Textbox ，用于输入/输出

按钮：，在右下属性窗口修改文本内容

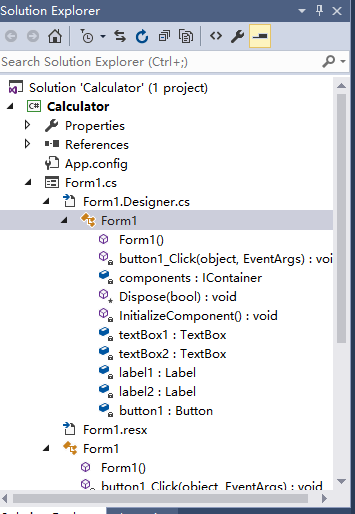
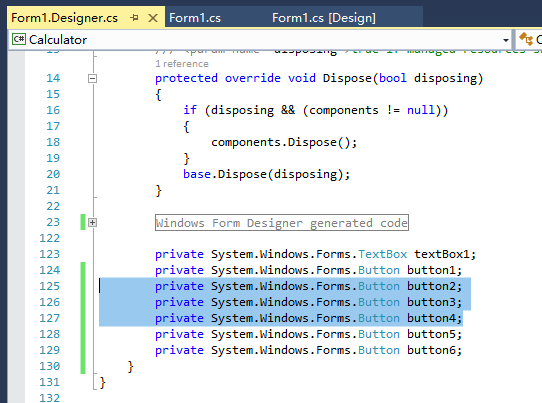
 

（2）前端结果

鼠标选中相应控件/Form后，在右下属性窗口修改文本内容。



（3）展开右上角窗口树状目录如下图左半部分所示，双击Form1，可得下图右半部分所示的class Form1相关数据成员

（4）鼠标左键，双击“+”按钮，添加相应成员函数

选中函数名“button1\_Click”，右键“rename”，给该成员函数一个有意义的函数名，例如Add

private void Add(object sender, EventArgs e)

（5）基本思路：参考手机上的计算器实现方式，把整个混合运算的输入看作是字符串（例如“11+22+33-44”，通过this.textBox1.Text进行输入/输出），然后在“=”对应的成员函数处进行相关处理：1）提取操作数；2）提取运算符；3）计算；

（6）主要用到的类

String 类

<https://msdn.microsoft.com/zh-cn/library/system.string(VS.80).aspx>

String 成员

<https://msdn.microsoft.com/zh-cn/library/system.string_members(v=vs.80).aspx>

Regex 类

<https://msdn.microsoft.com/zh-cn/library/system.text.regularexpressions.regex(VS.80).aspx>

Convert 类

<https://msdn.microsoft.com/zh-cn/library/system.convert.aspx>

MessageBox 类

<https://msdn.microsoft.com/zh-cn/library/system.windows.messagebox.aspx>

Color 成员

<https://msdn.microsoft.com/zh-cn/library/system.drawing.color_members(v=vs.80).aspx>

Timer 类

https://msdn.microsoft.com/zh-cn/library/system.windows.forms.timer.aspx

（7）参考代码

private void On1(object sender, EventArgs e)

{

this.textBox1.Text += "1";

}

private void OnAdd(object sender, EventArgs e)

{

this.textBox1.Text += "+";

}

private double Caculate(string operators, string numbers)

{

double result = 0;

//把由空格间隔的数字组成的string，根据空格分割成全是数字的string数组

//例如“12 23 34”分割成string数组，包含“12”，“23”，“34”三个元素

//使用Regex.Split，需要手动添加using System.Text.RegularExpressions;

string[] numberarray = Regex.Split(numbers, " ");

string[] operatorarray = Regex.Split(operators, " ");

result = Convert.ToInt32(numberarray[0]);

int operatorindex = 0;

for (int numberindex = 1; numberindex < numberarray.Length; numberindex++)

{

switch (operatorarray[operatorindex])

{

case "+":

result += Convert.ToDouble(numberarray[numberindex]);

break;

case "-":

result -= Convert.ToDouble(numberarray[numberindex]);

break;

case "\*":

result \*= Convert.ToInt32(numberarray[numberindex]);

break;

case "/":

result /= Convert.ToInt32(numberarray[numberindex]);

break;

}

operatorindex++;

}

return result;

}

private void OnCaculate(object sender, EventArgs e)

{

string myoperator = string.Empty, mynumber = string.Empty;

char text;

//获取运算符

for(int index = 0; index < this.textBox1.Text.Length; index ++)

{

text = this.textBox1.Text[index];

//数字

if (text >= '0' && text <= '9')

mynumber += text.ToString();

else

{

//保存当前的运算符,运算符之间用空格分开

myoperator += text.ToString() + " ";

//操作数之间用空格分开

mynumber += " ";

}

}

//去掉末尾的空格

mynumber = mynumber.TrimEnd();

myoperator = myoperator.TrimEnd();

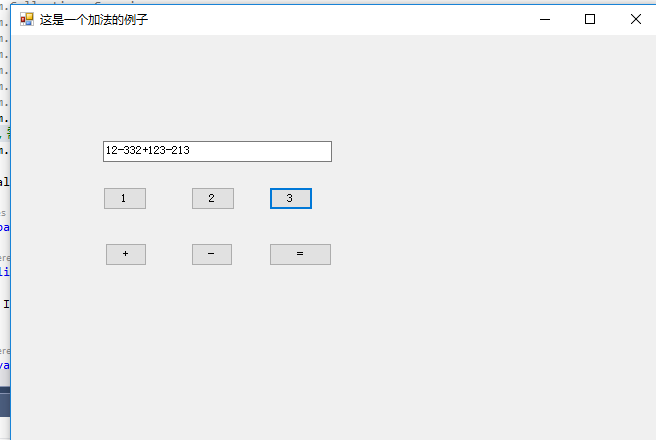
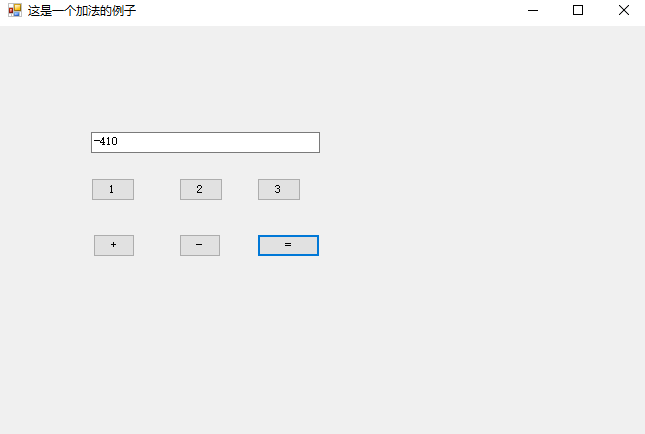
double result = Caculate(myoperator, mynumber);

//返回运算结果到前端界面

this.textBox1.Text = result.ToString();

}

（8）运行结果

（9）需要完善的问题：

1）混合运算的优先级处理；

2）小数的输入；

3）计算器上一些函数的处理；

4）异常处理，如0作除数；

………

## 实验二 成绩管理程序设计与实现

1、前端界面设计，增加控件listview



2、插入标题行

public Form1()

{

InitializeComponent();

this.listView1.Columns.Add("学号", 100, HorizontalAlignment.Center);

this.listView1.Columns.Add("姓名", 100, HorizontalAlignment.Center);

this.listView1.Columns.Add("数学", 100, HorizontalAlignment.Center);

this.listView1.Columns.Add("英语", 100, HorizontalAlignment.Center);

this.listView1.Columns.Add("政治", 100, HorizontalAlignment.Center);

this.listView1.Columns.Add("总分", 100, HorizontalAlignment.Center);

this.listView1.Columns.Add("平均分", 100, HorizontalAlignment.Center);

this.listView1.Columns.Add("名次", 100, HorizontalAlignment.Center);

this.listView1.View = System.Windows.Forms.View.Details;

}



3、读写文件

File 方法（using [System.IO](https://msdn.microsoft.com/zh-cn/library/system.io(v=vs.110).aspx)）

<https://msdn.microsoft.com/zh-cn/library/system.io.file_methods(v=vs.110).aspx>

//把txt文件内容读入到file中，然后对string对象file进行相关处理

string file = File.ReadAllText("G:\\scoremanagenment.txt", UTF8Encoding.Default);

//写文件，把string对象file中的内容写入txt文件中

File.WriteAllText("g:\\scoremanagenment.txt",file, UTF8Encoding.Default);

//追加记录到txt文件中

File.AppendAllText("G:\\scoremanagenment.txt",newrecord,UTF8Encoding.Default);

4、listview视图显示

string file = File.ReadAllText("G:\\scoremanagenment.txt", UTF8Encoding.Default);

//把txt文件中按行存储的信息利用regex.Split按行存入string数组中

string[] records = Regex.Split(file, "\r\n");

//开始更新视图

this.listView1.BeginUpdate();

//清空原有视图

this.listView1.Items.Clear();

// records.Length为数组的元素个数

for ( int index = 0; index < records.Length; index++)

{ //分割每行记录的各个字段

string[] components = Regex.Split(records[index], " ");

//生成listview的一行

ListViewItem lvi = new ListViewItem(components);

//添加背景色

lvi.SubItems[0].BackColor = Color.Red;

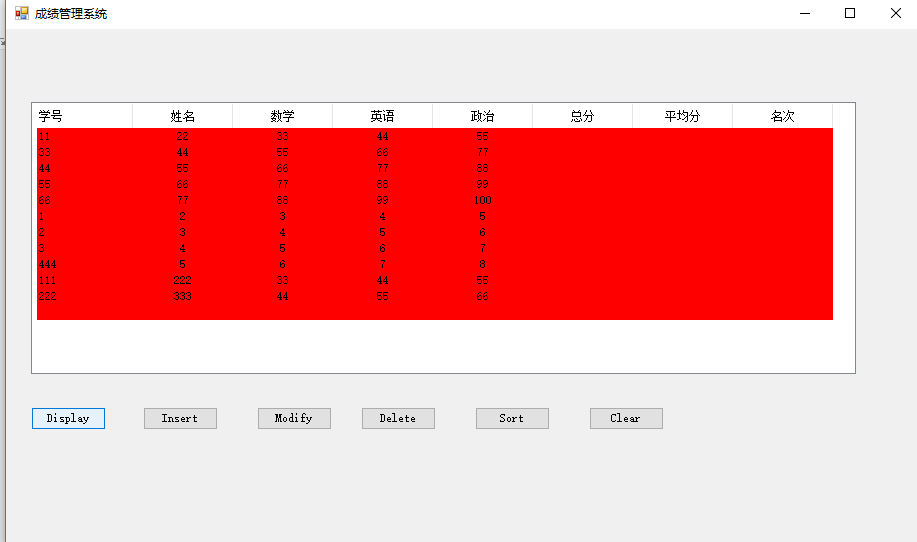
//把新生成的行加入到listview中

this.listView1.Items.Add(lvi);

}

//视图更新结束

this.listView1.EndUpdate();



5、弹出新form，进行插入操作

用于插入新纪录的Form2界面设计，然后在form1（成绩管理系统）的Insert按钮对应的成员函数中添加对Form2 对象的调用。在Form2对应的cs文件中，设计插入函数。



private void Insert(object sender, EventArgs e)

{

Form2 form2 = new Form2();

form2.Show();

}

private void Insert(object sender, EventArgs e)

{

//利用string的加法操作构造新纪录，作为一行，写入txt文件中

string newrecord = this.textBox1.Text + " " + this.textBox2.Text + " " + this.textBox3.Text + " " + this.textBox4.Text + " " + this.textBox5.Text + "\r\n";

File.AppendAllText("G:\\scoremanagenment.txt",newrecord,UTF8Encoding.Default);

//结束form2的调用

this.Dispose(false);

}

6、选中某一行

//选中的行号this.listView1.SelectedItems[0].Index

ListViewItem item = this.listView1.Items[this.listView1.SelectedItems[0].Index];

//用户没有选中任一行

this.listView1.SelectedItems.Count == 0

//删除选中的行

this.listView1.Items.RemoveAt(this.listView1.SelectedItems[0].Index);

//选中行的第i+1个字段的文本

item.SubItems[i].Text

//选中行的字段数

item.SubItems.Count;

7、可能用到的string成员函数

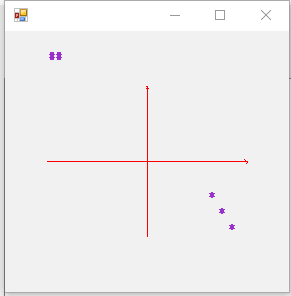
TrimEnd()

Replace(旧文本，新文本)

## 实验三超市

[http://blog.csdn.net/hbxtlhx/article/details/1650892](qq://txfile/)  
在WinForm里绘图  
当我们需要使用Window的窗体进行绘图的时候可能会对绘制图的进行放大或缩小的操作,利用PictureBox控件默认是实现不了这个功能的,但是我们可以通过对一个一般的UserControl操作来达到这个目的,而不一定使用PictureBox,这里,我使用了Form或UserControl的基类ScrollableControl进行了继承并在其上绘图,因为这个控件足够使用了.  
    我这里提到了如下的几个绘图的问题:  
    1:当我们在Windows里绘图的时候所用的坐标系统和我们在数学里使用的坐标系是不一样的,我们需要对其进行转换以适合我们在数学坐标系里的习惯.  
    2:当我们所需的画布大于当前窗口大小的时候需要滚动条的支持.  
    3:我们可能随时对所绘的图进行放大或缩小.  
    4:我们在绘图的时候不能出现闪动而使使用者感到讨厌,我们需要良好的显示效果,给人以快速的感觉  
    5:我们可能需要绘制出数学函数图像.或是给了一堆的要绘图的坐标绘制出图像.

那个链接里面的代码需要增添  
using System.Windows.Forms;  
using System.Drawing;  
using System.Drawing.Drawing2D;



随便画了四个坐标，两种符号显示