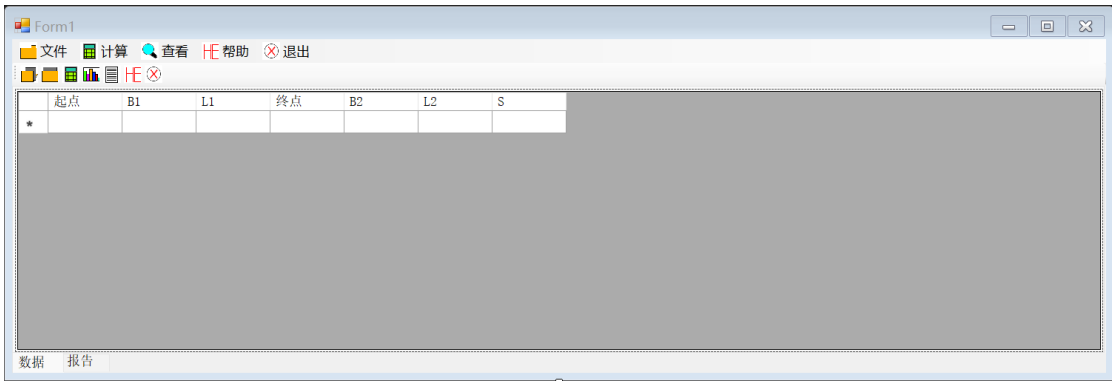
报告文档

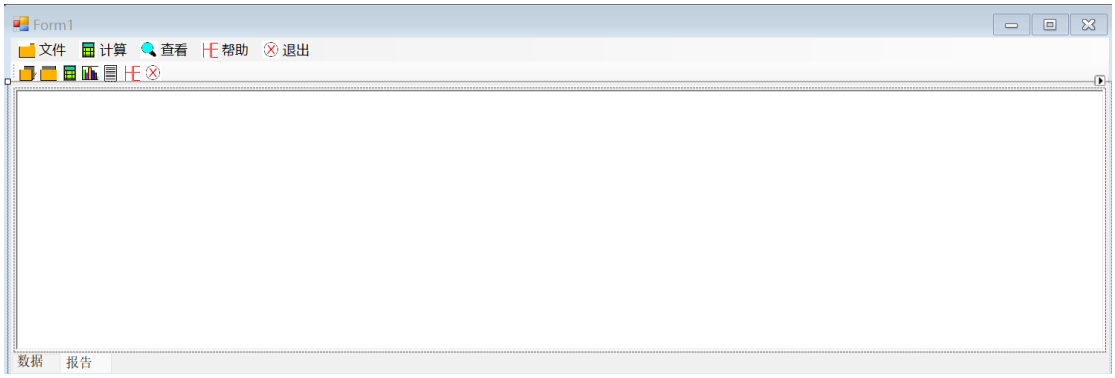
目录

一、程序优化性说明

1．用户交互界面说明

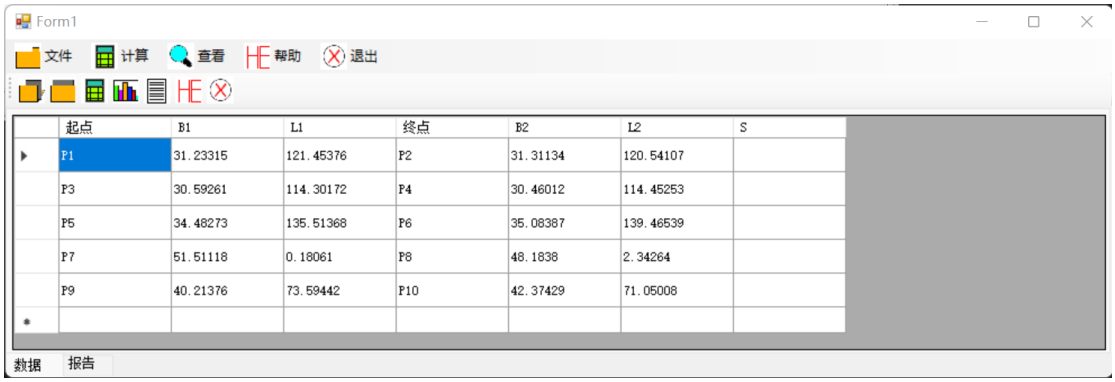
菜单、工具、数据、报告



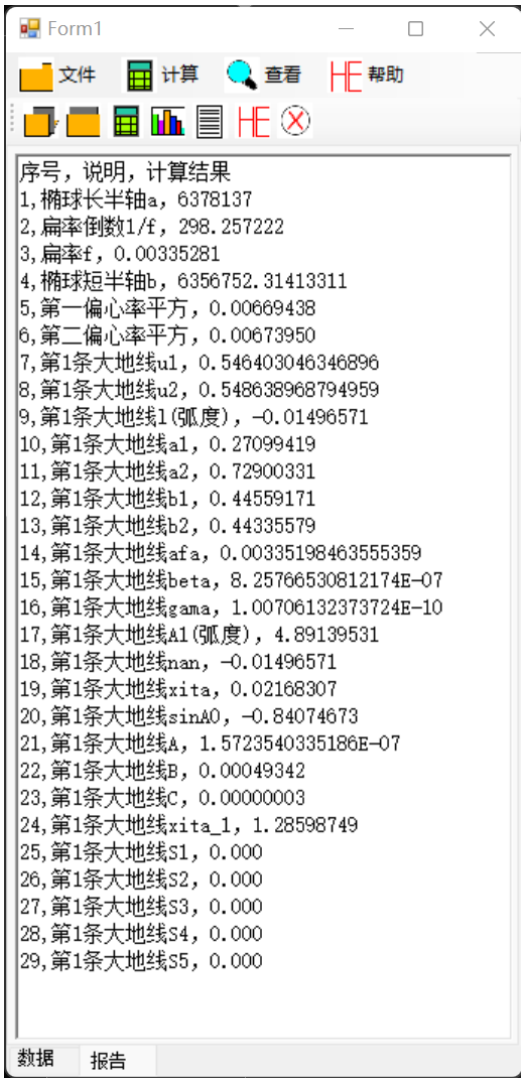


2．程序运行过程说明

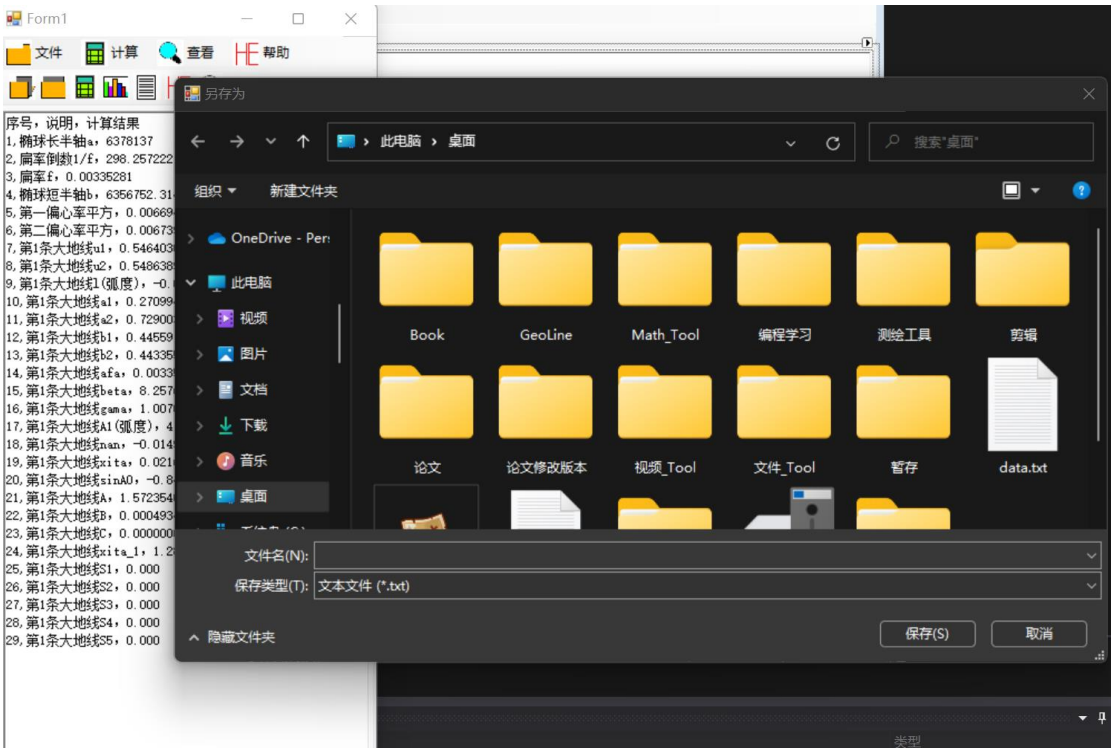
打开数据



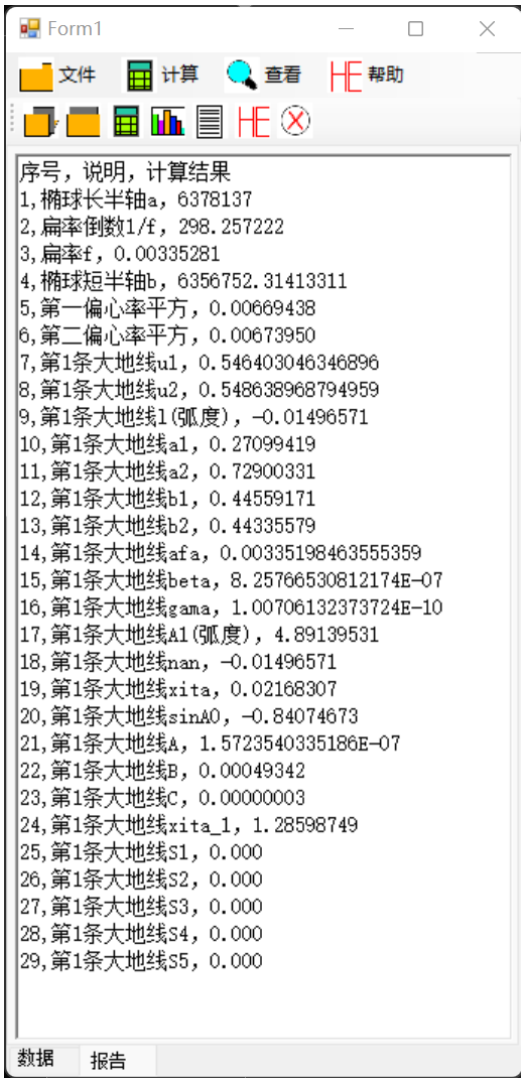
一键计算


Form1
文件
计算
查看
帮助
序号，说明，计算结果
1,椭球长半轴a，6378137
2,扁率倒数1/f，298.257222
3,扁率f，0.00335281
4,椭球短半轴b，6356752.31413311
5,第一偏心率平方，0.00669438
6,第二偏心率平方，0.00673950
7,第1条大地线u1，0.546403046346896
8,第1条大地线u2，0.548638968794959
9,第1条大地线l(弧度)，-0.01496571
10,第1条大地线a1，0.27099419
11,第1条大地线a2，0.72900331
12,第1条大地线b1，0.44559171
13,第1条大地线b2，0.44335579
14,第1条大地线afa，0.00335198463555359
15,第1条大地线beta，8.25766530812174E-07
16,第1条大地线gama，1.00706132373724E-10
17,第1条大地线A1(弧度)，4.89139531
18,第1条大地线nan，-0.01496571
19,第1条大地线xita，0.02168307
20,第1条大地线sinA0，-0.84074673
21,第1条大地线A，1.5723540335186E-07
22,第1条大地线B，0.00049342
23,第1条大地线C，0.00000003
24,第1条大地线xita_1，1.28598749
25,第1条大地线S1，0.000
26,第1条大地线S2，0.000
27,第1条大地线S3，0.000
28,第1条大地线S4，0.000
29,第1条大地线S5，0.000
数据
报告

保存结果


Form1
文件
计算
查看
帮助
序号，说明，计算结果
1,椭球长半轴a，6378137
2,扁率倒数1/f，298.257222
3,扁率f，0.00335281
4,椭球短半轴b，6356752.31
5,第一偏心率平方，0.00669
6,第二偏心率平方，0.00673
7,第1条大地线u1，0.546403
8,第1条大地线u2，0.548638
9,第1条大地线l(弧度)，-0.
10,第1条大地线a1，0.27099
11,第1条大地线a2，0.72900
12,第1条大地线b1，0.44559
13,第1条大地线b2，0.44335
14,第1条大地线afa，0.0033
15,第1条大地线beta，8.257
16,第1条大地线gama，1.007
17,第1条大地线A1(弧度)，4
18,第1条大地线nan，-0.014
19,第1条大地线xita，0.021
20,第1条大地线sinA0，-0.8
21,第1条大地线A，1.572354
22,第1条大地线B，0.000493
23,第1条大地线C，0.000000
24,第1条大地线xita_1，1.2
25,第1条大地线S1，0.000
26,第1条大地线S2，0.000
27,第1条大地线S3，0.000
28,第1条大地线S4，0.000
29,第1条大地线S5，0.000
另存为
此电脑 › 桌面
搜索"桌面"
组织
新建文件夹
OneDrive - Per
此电脑
视频
图片
文档
下载
音乐
桌面
Book
GeoLine
Math_Tool
编程学习
测绘工具
剪辑
论文
论文修改版本
视频_Tool
文件_Tool
暂存
data.txt
文件名(N):
保存类型(T):
文本文件 (*.txt)
隐藏文件夹
保存(S)
取消
类型

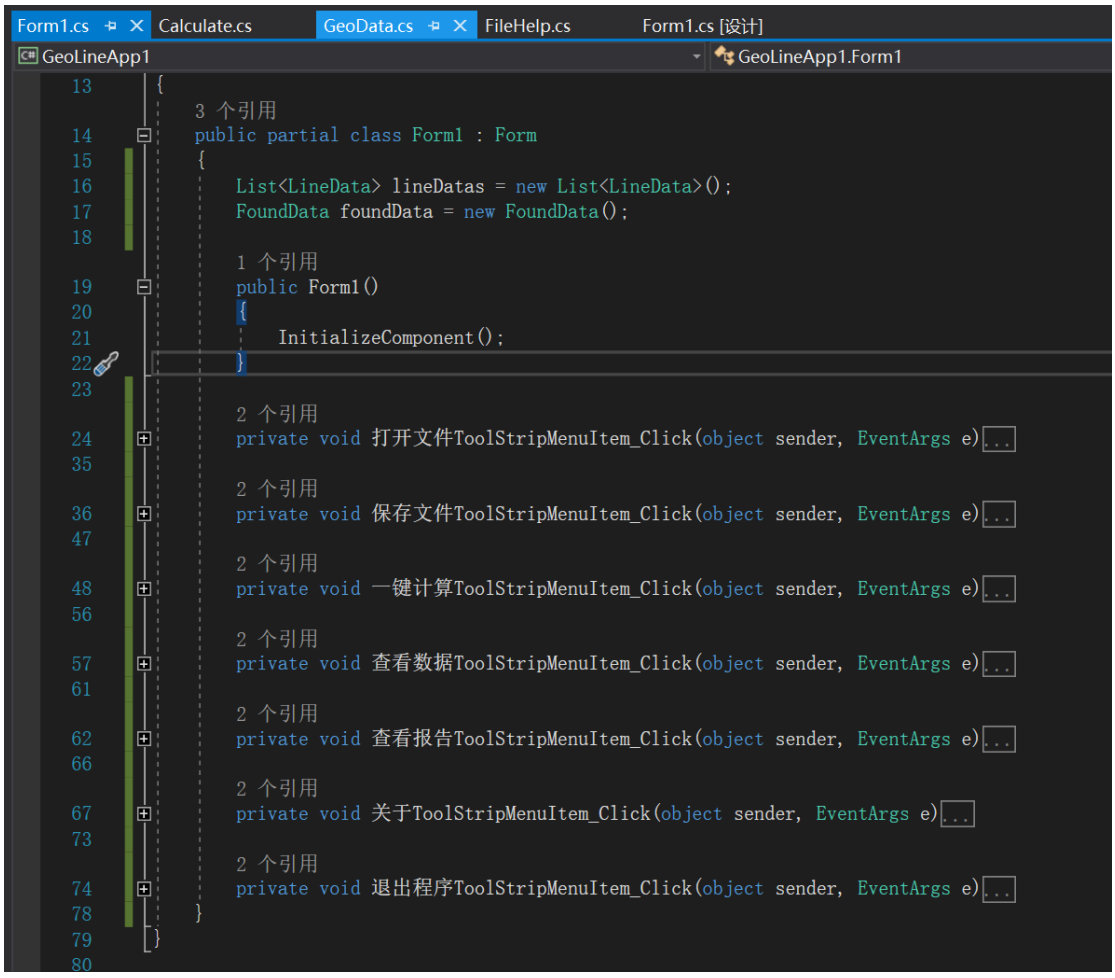
3．程序运行结果（给出程序运行结果）



二、程序规范性说明

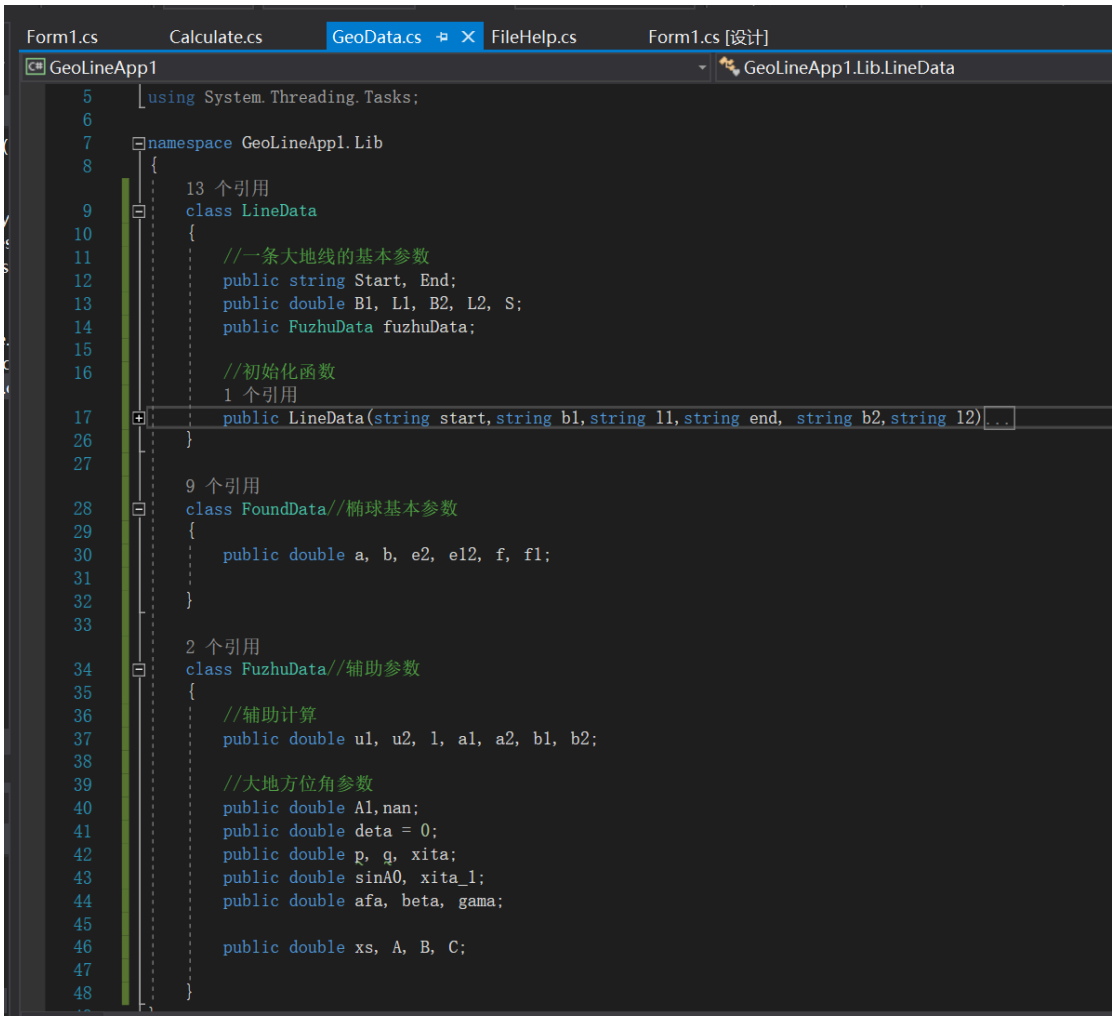
1．程序功能与结构设计说明

Form.cs



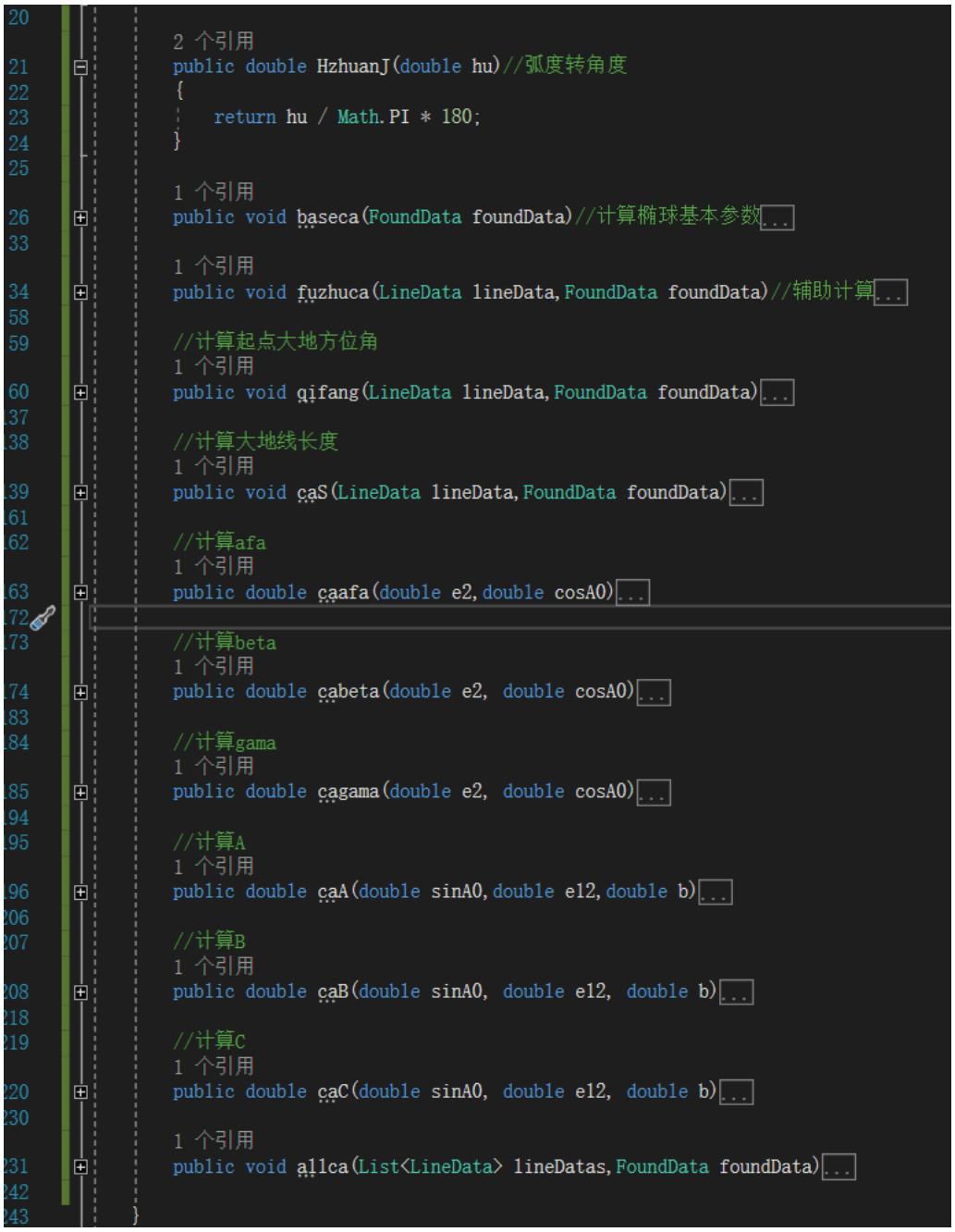
Lib

GeoData.cs



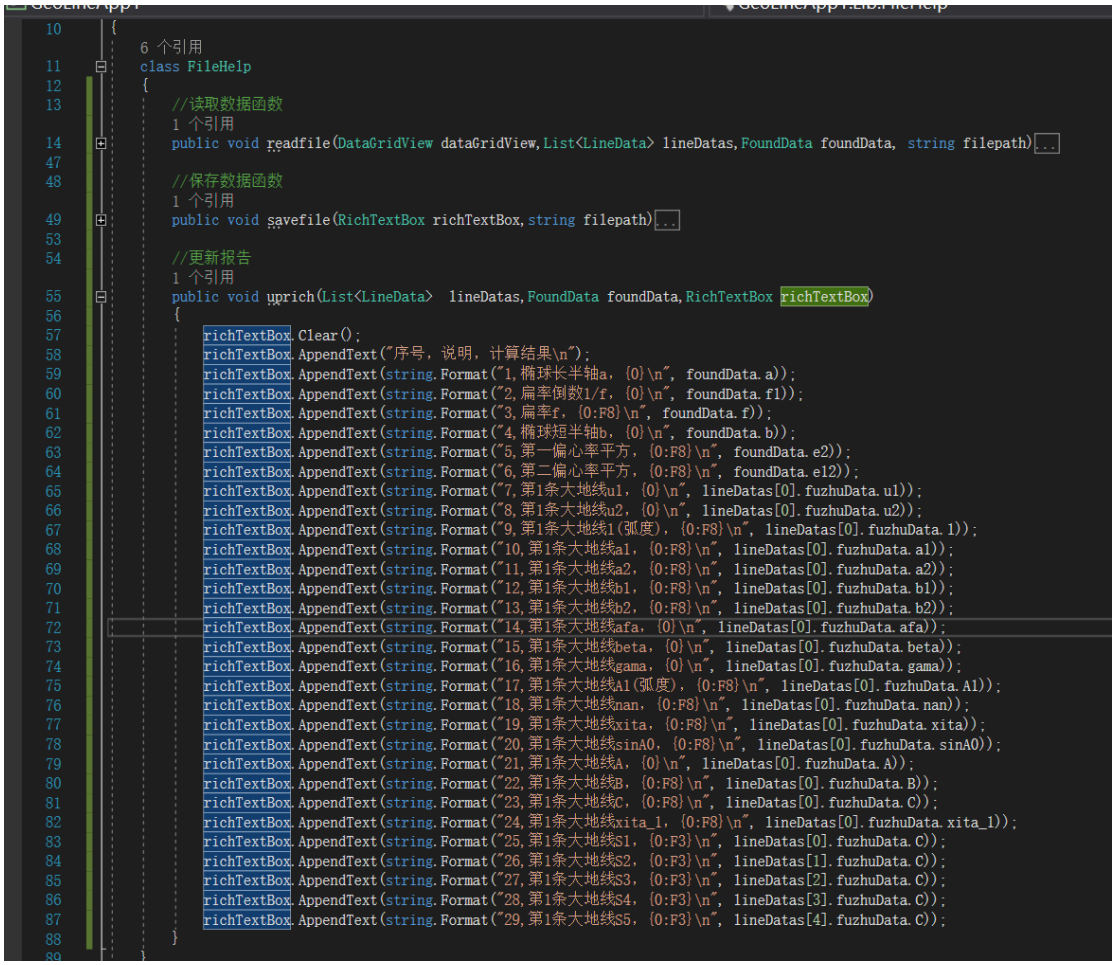

```
using System.Threading.Tasks;

namespace GeoLineApp1.Lib
{
    13 个引用
    class LineData
    {
        //一条大地线的基本参数
        public string Start, End;
        public double B1, L1, B2, L2, S;
        public FuzhuData fuzhuData;

        //初始化函数
        1 个引用
        public LineData(string start,string b1,string l1,string end, string b2,string l2)...
    }

    9 个引用
    class FoundData//椭球基本参数
    {
        public double a, b, e2, e12, f, f1;


    }

    2 个引用
    class FuzhuData//辅助参数
    {
        //辅助计算
        public double u1, u2, l, a1, a2, b1, b2;

        //大地方位角参数
        public double A1,nan;
        public double deta = 0;
        public double p, q, xita;
        public double sinA0, xita_1;
        public double afa, beta, gama;

        public double xs, A, B, C;

    }
```

Calculate.cs



```
20
        2 个引用
21      public double HzhuanJ(double hu)//弧度转角度
22      {
23          return hu / Math.PI * 180;
24      }
25
        1 个引用
26      public void baseca(FoundData foundData)//计算椭球基本参数...
33
        1 个引用
34      public void fuzhuca(LineData lineData, FoundData foundData)//辅助计算...
58
59      //计算起点大地方位角
        1 个引用
60      public void qifang(LineData lineData, FoundData foundData)...
37
38      //计算大地线长度
        1 个引用
39      public void caS(LineData lineData, FoundData foundData)...
61
62      //计算afa
        1 个引用
63      public double caafa(double e2, double cosA0)...
72
73      //计算beta
        1 个引用
74      public double cabeta(double e2, double cosA0)...
83
84      //计算gama
        1 个引用
85      public double cagama(double e2, double cosA0)...
94
95      //计算A
        1 个引用
96      public double caA(double sinA0, double e12, double b)...
206
207     //计算B
        1 个引用
208     public double caB(double sinA0, double e12, double b)...
218
219     //计算C
        1 个引用
220     public double caC(double sinA0, double e12, double b)...
230
        1 个引用
231     public void allca(List<LineData> lineDatas, FoundData foundData)...
242
243 }
```

FileHelp.cs



```
{
    6 个引用
    class FileHelp
    {
        //读取数据函数
        1 个引用
        public void readfile(DataGridView dataGridView,List<LineData> lineDatas,FoundData foundData, string filepath)...

        //保存数据函数
        1 个引用
        public void savefile(RichTextBox richTextBox,string filepath)...

        //更新报告
        1 个引用
        public void uprich(List<LineData>  lineDatas,FoundData foundData,RichTextBox richTextBox)
        {
            richTextBox.Clear();
            richTextBox.AppendText("序号，说明，计算结果\n");
            richTextBox.AppendText(string.Format("1,椭球长半轴a，{0}\n", foundData.a));
            richTextBox.AppendText(string.Format("2,扁率倒数1/f，{0}\n", foundData.f1));
            richTextBox.AppendText(string.Format("3,扁率f，{0:F8}\n", foundData.f));
            richTextBox.AppendText(string.Format("4,椭球短半轴b，{0}\n", foundData.b));
            richTextBox.AppendText(string.Format("5,第一偏心率平方，{0:F8}\n", foundData.e2));
            richTextBox.AppendText(string.Format("6,第二偏心率平方，{0:F8}\n", foundData.e12));
            richTextBox.AppendText(string.Format("7,第1条大地线u1，{0}\n", lineDatas[0].fuzhuData.u1));
            richTextBox.AppendText(string.Format("8,第1条大地线u2，{0}\n", lineDatas[0].fuzhuData.u2));
            richTextBox.AppendText(string.Format("9,第1条大地线l(弧度)，{0:F8}\n", lineDatas[0].fuzhuData.l));
            richTextBox.AppendText(string.Format("10,第1条大地线a1，{0:F8}\n", lineDatas[0].fuzhuData.a1));
            richTextBox.AppendText(string.Format("11,第1条大地线a2，{0:F8}\n", lineDatas[0].fuzhuData.a2));
            richTextBox.AppendText(string.Format("12,第1条大地线b1，{0:F8}\n", lineDatas[0].fuzhuData.b1));
            richTextBox.AppendText(string.Format("13,第1条大地线b2，{0:F8}\n", lineDatas[0].fuzhuData.b2));
            richTextBox.AppendText(string.Format("14,第1条大地线afa，{0}\n", lineDatas[0].fuzhuData.afa));
            richTextBox.AppendText(string.Format("15,第1条大地线beta，{0}\n", lineDatas[0].fuzhuData.beta));
            richTextBox.AppendText(string.Format("16,第1条大地线gama，{0}\n", lineDatas[0].fuzhuData.gama));
            richTextBox.AppendText(string.Format("17,第1条大地线A1(弧度)，{0:F8}\n", lineDatas[0].fuzhuData.A1));
            richTextBox.AppendText(string.Format("18,第1条大地线nan，{0:F8}\n", lineDatas[0].fuzhuData.nan));
            richTextBox.AppendText(string.Format("19,第1条大地线xita，{0:F8}\n", lineDatas[0].fuzhuData.xita));
            richTextBox.AppendText(string.Format("20,第1条大地线sinA0，{0:F8}\n", lineDatas[0].fuzhuData.sinA0));
            richTextBox.AppendText(string.Format("21,第1条大地线A，{0}\n", lineDatas[0].fuzhuData.A));
            richTextBox.AppendText(string.Format("22,第1条大地线B，{0:F8}\n", lineDatas[0].fuzhuData.B));
            richTextBox.AppendText(string.Format("23,第1条大地线C，{0:F8}\n", lineDatas[0].fuzhuData.C));
            richTextBox.AppendText(string.Format("24,第1条大地线xita_1，{0:F8}\n", lineDatas[0].fuzhuData.xita_1));
            richTextBox.AppendText(string.Format("25,第1条大地线S1，{0:F3}\n", lineDatas[0].fuzhuData.C));
            richTextBox.AppendText(string.Format("26,第1条大地线S2，{0:F3}\n", lineDatas[1].fuzhuData.C));
            richTextBox.AppendText(string.Format("27,第1条大地线S3，{0:F3}\n", lineDatas[2].fuzhuData.C));
            richTextBox.AppendText(string.Format("28,第1条大地线S4，{0:F3}\n", lineDatas[3].fuzhuData.C));
            richTextBox.AppendText(string.Format("29,第1条大地线S5，{0:F3}\n", lineDatas[4].fuzhuData.C));
        }
    }
```

核心计算源码:

```
class Calculate
    {
        public double JzhuanH(double jiao)//角度转弧度
        {
            //12.34566
            double du = (int)jiao;//12
            double fen = (int)((jiao - du) * 100);//34
            double miao = (((jiao - du) * 100) - fen) * 100;//56.6
            double zong = du + fen / 60 + miao / 3600;
            return zong / 180 * Math.PI;
        }

        public double HzhuanJ(double hu)//弧度转角度
        {
            return hu / Math.PI * 180;
        }
```

```
        public void baseca(FoundData foundData)//计算椭球基本参数
        {
            foundData.f = 1 / foundData.f1;
            foundData.b = foundData.a * (1 - foundData.f);
            foundData.e2 = (Math.Pow(foundData.a, 2) - Math.Pow(foundData.b, 2)) /
(Math.Pow(foundData.a, 2));
            foundData.e12 = foundData.e2 / (1 - foundData.e2);
        }

        public void fuzhuca(LineData lineData,FoundData foundData)//辅助计算
        {
            double B1 = JzhuanH(lineData.B1);double L1 = JzhuanH(lineData.L1);
            double B2 = JzhuanH(lineData.B2);double L2 = JzhuanH(lineData.L2);
            double e2 = foundData.e2;

            double u1 = Math.Atan((Math.Sqrt(1 - e2)) * (Math.Tan(B1)));
            double u2 = Math.Atan((Math.Sqrt(1 - e2)) * (Math.Tan(B2)));

            double l = L2 - L1;

            double a1 = Math.Sin(u1) * Math.Sin(u2);
            double a2 = Math.Cos(u1) * Math.Cos(u2);
            double b1 = Math.Cos(u1) * Math.Sin(u2);
            double b2 = Math.Sin(u1) * Math.Cos(u2);

            lineData.fuzhuData.u1 = u1;
            lineData.fuzhuData.u2 = u2;
            lineData.fuzhuData.l = l;
            lineData.fuzhuData.a1 = a1;
            lineData.fuzhuData.a2 = a2;
            lineData.fuzhuData.b1 = b1;
            lineData.fuzhuData.b2 = b2;
        }

        //计算起点大地方位角
        public void qifang(LineData lineData,FoundData foundData)
        {

            double l = lineData.fuzhuData.l;
            double u1 = lineData.fuzhuData.u1;
            double u2 = lineData.fuzhuData.u2;
            double a1 = lineData.fuzhuData.a1;
            double a2 = lineData.fuzhuData.a2;
            double b1 = lineData.fuzhuData.b1;
```

```
double b2 = lineData.fuzhuData.b2;

double afa, beta, gama, A1, nan, xita, sinA0;


double deta = 0;
double detaqian = 0;
while(deta==detaqian)
{
    detaqian = deta;
    nan = l + deta;
    nan = l + deta;
    double p = Math.Cos(u2) * Math.Sin(nan);
    double q = b1 - b2 * Math.Cos(nan);
    A1 = Math.Atan(p / q);
    A1 = HzhuanJ(A1);

    //判断pq符号
    if (p > 0)
    {
        if (q > 0) A1 = Math.Abs(A1);
        else A1 = 180 - Math.Abs(A1);
    }
    else
    {
        if (q < 0) A1 = 180 + Math.Abs(A1);
        else A1 = 360 - Math.Abs(A1);
    }

    if (A1 < 0) A1 = A1 + 360;
    else if (A1 > 360) A1 = A1 - 360;
    A1 = JzhuanH(A1);

    double sinxita = p * Math.Sin(A1) + q * Math.Cos(A1);
    double cosxita = a1 + a2 * Math.Cos(nan);
    xita = Math.Atan2(sinxita, cosxita);
    xita = HzhuanJ(xita);

    if (cosxita > 0) xita = Math.Abs(xita);
    else xita = 180 - Math.Abs(xita);
    xita = JzhuanH(xita);

    sinA0 = Math.Cos(u1) * Math.Sin(A1);
    double xita_1 = Math.Atan((Math.Tan(u1)) / (Math.Cos(A1)));
```

```
                double A0 = Math.Asin(sinA0);
                double cosA0 = Math.Cos(A0);

                afa = caafa(foundData.e2, cosA0);
                beta = cabeta(foundData.e2, cosA0);
                gama = cagama(foundData.e2, cosA0);

                deta =
                    (afa * xita +
                    beta * Math.Cos(2 * xita_1 + xita) * Math.Sin(xita) +
                    gama * Math.Sin(2 * xita) * Math.Cos(4 * xita_1 + 2 * xita)) *
sinA0;

                //赋值afa, beta, gama, A1, nan, xita, sinA0
                lineData.fuzhuData.afa = afa;
                lineData.fuzhuData.beta = beta;
                lineData.fuzhuData.gama = gama;
                lineData.fuzhuData.A1 = A1;
                lineData.fuzhuData.nan = nan;
                lineData.fuzhuData.xita = xita;
                lineData.fuzhuData.sinA0 = sinA0;

            }

        }

        //计算大地线长度
        public void caS(LineData lineData,FoundData foundData)
        {
            double sinA0 = lineData.fuzhuData.sinA0;


            double u1 = lineData.fuzhuData.u1;
            double A1 = lineData.fuzhuData.A1;
            double xita = lineData.fuzhuData.xita;
            double xita_1 = Math.Atan((Math.Tan(u1)) / (Math.Cos(A1)));
            double C = caC(sinA0, foundData.e12, foundData.b);
            double xs = C * Math.Sin(2 * xita) * Math.Cos(4 * xita_1 + 2 * xita);
            double B = caB(sinA0, foundData.e12, foundData.b);
            double A = caA(sinA0, foundData.e12, foundData.b);
            double S = (xita - B * Math.Sin(xita) * Math.Cos(2 * xita_1 + xita) - xs) /
A1;

            lineData.fuzhuData.A = A;
```

```
            lineData.fuzhuData.B = B;
            lineData.fuzhuData.C = C;
            lineData.fuzhuData.xita_1 = xita_1;
            lineData.fuzhuData.xs = xs;
            lineData.S = S;
        }

        //计算afa
        public double caafa(double e2,double cosA0)
        {
            double e4 = Math.Pow(e2, 2);
            double e6 = Math.Pow(e2, 3);
            double cos2A0 = Math.Pow(cosA0, 2);
            double cos4A0 = Math.Pow(cosA0, 4);

            return (e2 / 2 + e4 / 8 + e6 / 16) - ((e4 / 16 + e6 / 16) * cos2A0) + ((3 *
e6 / 128) * cos4A0);
        }

        //计算beta
        public double cabeta(double e2, double cosA0)
        {
            double e4 = Math.Pow(e2, 2);
            double e6 = Math.Pow(e2, 3);
            double cos2A0 = Math.Pow(cosA0, 2);
            double cos4A0 = Math.Pow(cosA0, 4);

            return ((e4 / 16 + e6 / 16) * cos2A0) - ((e6 / 32) * cos4A0);
        }

        //计算gama
        public double cagama(double e2, double cosA0)
        {
            double e4 = Math.Pow(e2, 2);
            double e6 = Math.Pow(e2, 3);
            double cos2A0 = Math.Pow(cosA0, 2);
            double cos4A0 = Math.Pow(cosA0, 4);

            return (e6 / 256) * cos4A0;
        }

        //计算A
        public double caA(double sinA0,double e12,double b)
        {
```

```
    double sin2A0 = Math.Pow(sinA0, 2);
    double cos2A0 = 1 - sin2A0;
    double k2 = e12 * cos2A0;
    double k4 = Math.Pow(k2, 2);
    double k6 = Math.Pow(k2, 3);

    return (1 - k2 / 4 + 7 * k4 / 64 - 15 * k6 / 256) / b;
}

//计算B
public double caB(double sinA0, double e12, double b)
{
    double sin2A0 = Math.Pow(sinA0, 2);
    double cos2A0 = 1 - sin2A0;
    double k2 = e12 * cos2A0;
    double k4 = Math.Pow(k2, 2);
    double k6 = Math.Pow(k2, 3);

    return k2 / 4 - k4 / 8 + 37 * k6 / 512;
}

//计算C
public double caC(double sinA0, double e12, double b)
{
    double sin2A0 = Math.Pow(sinA0, 2);
    double cos2A0 = 1 - sin2A0;
    double k2 = e12 * cos2A0;
    double k4 = Math.Pow(k2, 2);
    double k6 = Math.Pow(k2, 3);

    return k4 / 128 - k6 / 128;
}

public void allca(List<LineData> lineDatas,FoundData foundData)
{
    double test= JzhuanH(30);
    foreach(LineData lineData in lineDatas)
    {
        baseca(foundData);
        fuzhuca(lineData, foundData);
        qifang(lineData, foundData);
        caS(lineData, foundData);
    }
}
```

```
}
```