测绘程序设计

————地图图幅编号计算

时

时间:2025年5月26日

目录

[一、程序优化性说明 3](#_Toc29386)

[1.用户交互界面说明 3](#_Toc31961)

[2. 程序运行过程说明 4](#_Toc18198)

[3. 程序运行结果 4](#_Toc8271)

[二、 程序规范说明 5](#_Toc22851)

[1. 程序功能与结构设计说明 5](#_Toc23354)

[2. 核心算法源码 5](#_Toc7885)

[2.1根据经纬度计算旧图幅 5](#_Toc18401)

[2.2根据新图幅计算左下角经纬度 7](#_Toc16557)

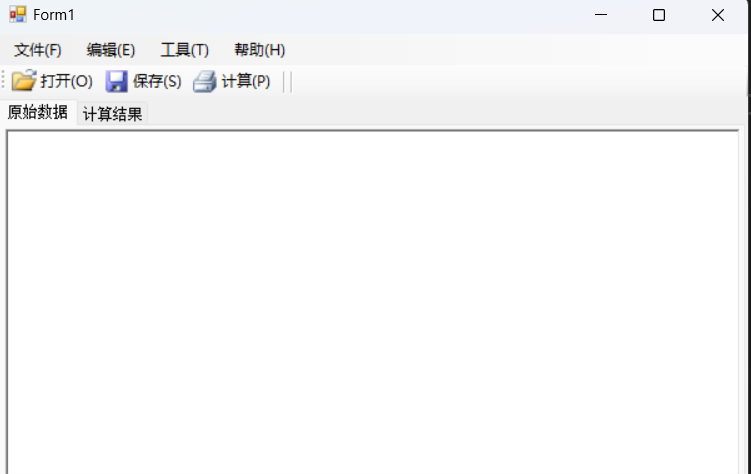
[2.3根据旧图幅计算新图幅 8](#_Toc19423)

[2.4定义变量 10](#_Toc5240)

# 一、程序优化性说明

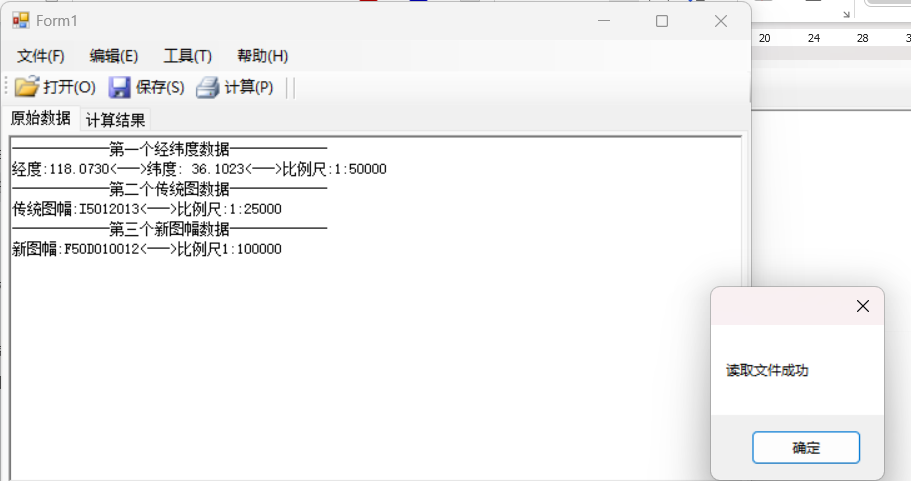
1.用户交互界面说明

首先该界面有打开文件，保存文件，计算按钮，其中有菜单栏，查看原始数据，和计算结果的界面，然后点击打开和保存按钮，会有提示如（打开文件成功）（保存文件成功），然后打开的文件中的数据会呈现在原始数据界面当中，点击计算按钮后会弹出计算成功，需点击计算结果界面才可看到计算结果



## 程序运行过程说明

当程序运行时会弹出提示如（读取文件成功）等



## 程序运行结果

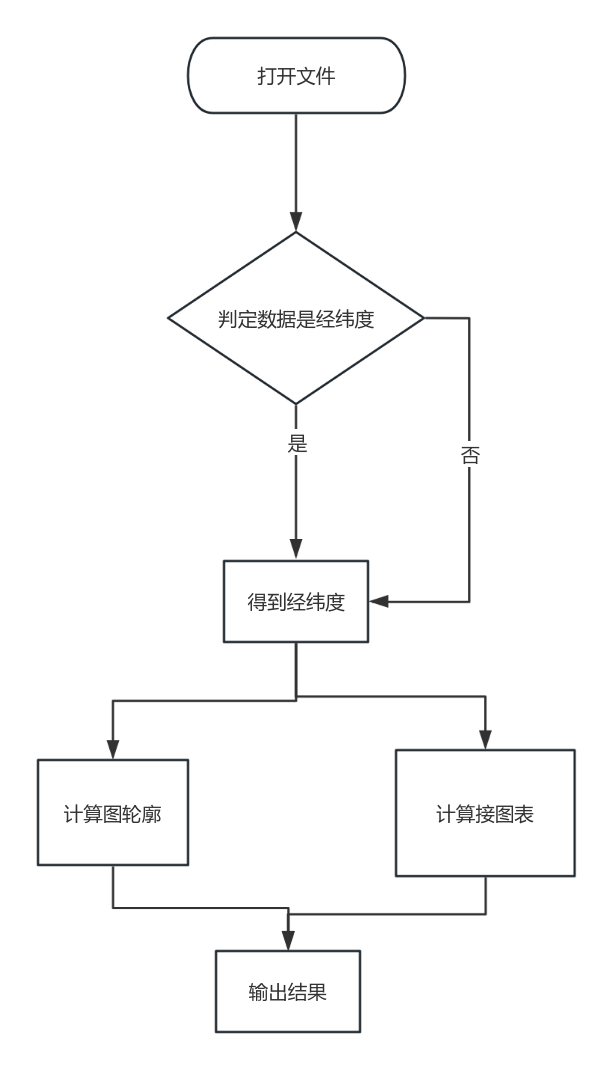
程序运行结果如下图所示



# 程序规范说明

## 程序功能与结构设计说明

该程序主要按照以下流程图设计



## 核心算法源码

### 2.1根据经纬度计算旧图幅

public string BLToOld(double B, double L, string scal)

{

//这部分先通过计算1:10w的号码，然后再依次计算，

string res = "";

// 计算1:100万图幅编号

int H100 = (int)B / 4 + 1;

Char H1001 = (char)('A' + H100 - 1);

int L100 = (int)L / 6 + 31;

res += $"{H1001}{L100}";

// 计算100万的

double Lc\_100 = L - ((int)L / 6) \* 6;

double Bc\_100 = B - ((int)B / 4) \* 4;

//计算10w的

double Bc\_10 = B - (int)(B / BC10) \* BC10;

double Lc\_10 = L - (int)(L / LC10) \* LC10;

double N1 = 0, N2 = 0, N3 = 0;

int row = 0, col = 0;

//计算5w的

double Bc\_5 = Bc\_10 - (int)(Bc\_10 / BC5) \* BC5;

double Lc\_5 = Lc\_10 - (int)(Lc\_10 / LC5) \* LC5;

if (scal == "1:100000") // 1:10万

{

row = 12 - (int)(Bc\_100 / BC10);

col = (int)(Lc\_100 / LC10);

N1 = (row - 1) \* 12 + col;

}

else if (scal == "1:50000") // 1:5万

{

row = 12 - (int)(Bc\_100 / BC10);

col = (int)(Lc\_100 / LC10)+1;

N1 = (row - 1) \* 12 + col;

double row1 = 2 - (int)(Bc\_10 / BC5);

double col1 = (int)(Lc\_10 / LC5)+1;

N2 = (row1 - 1) \* 2 + col1;

res += $"{N1}{N2}";

}

else if (scal == "1:25000") // 1:2.5万

{

row = 12 - (int)(Bc\_100 / BC10);

col = (int)(Lc\_100 / LC10) + 1;

N1 = (row - 1) \* 12 + col;

double row1 = 2 - (int)(Bc\_10 / BC5);

double col1 = (int)(Lc\_10 / LC5) + 1;

N2 = (row1 - 1) \* 2 + col1;

double row2 = 2 - (int)(Bc\_5 / BC25);

double col2 = (int)(Lc\_5 / LC5) + 1;

N2 = (row2 - 1) \* 2 + col2;

res += $"{N1}{N2}{N3}";

}//记得优化下这部分

return res;

}

### 2.2根据新图幅计算左下角经纬度

/// <summary>

/// 计算左下角经纬度

/// </summary>

/// <param name="New1"></param>

/// <returns></returns>

public (double B,double L)NewToBLZX(string New1)

{

int Row100 = (int)(New1[0]-'A'+1);

int Lie100 = int.Parse(New1.Substring(1, 2));

int Row = int.Parse(New1.Substring(4, 3));

int Lie = int.Parse(New1.Substring(7, 3));

double B=0.00000 ,L=0.0000;

if (New1.Substring(3,1)=="D")

{

B = (Row100 - 1) \* BC100 + (BC100/BC10-Row)\*BC10;

L = (Lie100 - 31) \* LC100 + (Lie - 1) \* LC10;

}

else if (New1.Substring(3,1) == "E")

{

B = (Row100 - 1) \* BC100 + (BC100 / BC5 - Row) \* BC5;

L = (Lie100 - 31) \* LC100 + (Lie - 1) \* LC5;

}

else if (New1.Substring(3,1) == "F")

{

B = (Row100 - 1) \* BC100 + (BC100 / BC25 - Row) \* BC25;

L = (Lie100 - 31) \* LC100 + (Lie - 1) \* LC25;

}

return (B, L);

}

### 2.3根据旧图幅计算新图幅

private string OldToNew(string oldMapNumber)

{

//48个格，120 1,3，144，12,10行，20,40

//这部分就是先计算10w的，10w有144个格子先算出来，然后再计算5w的在10w基础上

string res = $"{oldMapNumber.Substring(0, 3)}";

if(oldMapNumber.Length==6)//根据长度来判断这个是10w的

{

res += "D";

int H10 = int.Parse(oldMapNumber.Substring(3, 3));

if(H10%12==0)

{

int h\_10 = H10 / 12;

res += $"{h\_10:000}{012}";

}

else

{

int st= (int)H10 /12;

int l\_10 = H10 - st \* 12;

int h\_10 = st + 1;

res += $"{h\_10:000}{l\_10:000}";

}

}

else if(oldMapNumber.Length==7)//5w的

{

res += "E";

int H10 = int.Parse(oldMapNumber.Substring(3, 3));

int H5 = int.Parse(oldMapNumber.Substring(6, 1));

if (H10 % 12 == 0)

{

int h\_10 = H10 / 12;

int l\_10 = 12;

int h\_5 = h\_10 \* 2;

int l\_5 = l\_10 \* 2;

if(H5==1)

{

res += $"{h\_5 - 1:000}{l\_5 - 1:000}";

}

else if(H5==2)

{

res += $"{h\_5 - 1:000}{l\_5:000}";

}

else if(H5==3)

{

res += $"{h\_5:000}{l\_5 - 1:000}";

}

else if(H5==4)

{

res += $"{h\_5:000}{l\_5:000}";

}

}

else

{

int st = (int)H10 / 12;

int l\_10 = H10 - st \* 12;

int h\_10 = st + 1;

int h\_5 = h\_10 \* 2;

int l\_5 = l\_10 \* 2;

if (H5 == 1)

{

res += $"{h\_5 - 1:000}{l\_5 - 1:000}";

}

else if (H5 == 2)

{

res += $"{h\_5 - 1:000}{l\_5:000}";

}

else if (H5 == 3)

{

res += $"{h\_5:000}{l\_5 - 1:000}";

}

else if (H5 == 4)

{

res += $"{h\_5:000}{l\_5:000}";

}

}

}

else if (oldMapNumber.Length == 8)//2.5w的这部分可以优化下，就是反复叠加，写个函数，因为时间关系我只根据题目的特例来写

{

res += "F";

int H10 = int.Parse(oldMapNumber.Substring(3, 3));

int H5 = int.Parse(oldMapNumber.Substring(6, 1));

int H25 = int.Parse(oldMapNumber.Substring(7, 1));

if (H10 % 12 == 0)

{

int h\_10 = H10 / 12;

int l\_10 = 12;

int h\_5 = h\_10 \* 2;

int l\_5 = l\_10 \* 2;

if (H5 == 1)

{

h\_5 = h\_5 - 1;

l\_5 = l\_5 - 1;

int h\_25 = h\_5 \* 2;

int l\_25 = l\_5 \* 2;

if (H25 == 3)

{

res += $"{h\_25:000}{l\_25 - 1:000}";

}

}

}

}

return res;

}

### 2.4定义变量

//原始数据

public string B;

public string L;

public string scal1;

public string scal2;

public string scal3;

public string NewTf;

public string OldTf;

//定义比例尺

public string D = "1:100000";

public string E = "1:50000";

public string F = "1:25000";

//将经纬差转化为十进制

public double BC100 = DDMMSSToS("4.0000");

public double LC100 = DDMMSSToS("6.0000");

public double BC10 = DDMMSSToS("0.2000");

public double LC10 = DDMMSSToS("0.3000");

public double BC5 = DDMMSSToS("0.1000");

public double LC5 = DDMMSSToS("0.1500");

public double BC25 = DDMMSSToS("0.0500");

public double LC25 = DDMMSSToS("0.0730");