导线测量程序设计报告

## 题目要求

设计程序进行附合导线计算，数据从文件中读入，计算角度闭合差、全长闭合差和各坐标的平差值。

## 需求分析

输入：./data.txt文件

格式：（角度使用 度,分,秒 表示；坐标使用 X,Y 表示。同一行的各数据间用空格隔开）

第一行：测站数n

第二行：已知方位角（2）

第三行：观测角（左角）（n）

第四行：边长（n-1）

第五行：已知坐标（2）

格式范例：

6

237,59,30 46,45,24

99,01,00 167,45,36 123,11,24 189,20,36 179,59,18 129,27,24

225.85 139.03 172.57 100.07 102.48

2507.69,1215.63 2166.74,1757.27

输出：./out.txt

内容：输出内容有角度闭合差、导线相对闭合差和各导线坐标。

范例：

角度闭合差=-0.000175

导线相对闭合差K=1/3523.809524

导线坐标：

(2507.69,1215.63)

(2299.83,1303.80)

(2186.29,1383.97)

(2192.45,1556.40)

(2179.74,1655.64)

(2166.74,1757.27)

数学模型：

先通过已知方位角和观测角计算角度闭合差，然后计算改正数和改正角以计得各坐标方位角。

通过坐标方位角与距离计算出坐标增量和相对闭合差，并计算改正。最后计算改正后坐标值。

## 方案设计

数据结构：使用数组表示各数据

角度使用弧度以double记录，当输入输出时转化成度分秒形式使用int[3]记录。

坐标使用double[2]记录。

边长使用double记录。

主要数据结构如下：

double \*gcj;//观测角

double \*jdgz;//角度改正值

double \*gzj;//改正后角度

double \*fwj;//方位角

double (\*zb)[2];//坐标

double (\*zbzl)[2];//坐标增量

double (\*zbgz)[2];//坐标改正

double (\*gzzl)[2];//改正后增量

double \*bc;//边长

主要算法：

程序中先将文件内容导入，在导入过程中先根据读到的测站数动态分配储存数据的内存。随后将其他数据读入内存并转换成相应的形式保存在变量中。

计算角度闭合差并与允许值进行对比。

进行角度改正。

计算方位角。

计算全长闭合差。

进行坐标改正。

计算改正后坐标。

将所需内容输出到文件。

模块分解：

程序中分为将主要算法和主要流程。主要流程在count()函数中。

函数设计：

int Round(double num) //四舍五入到整数

double Round2(double num) //四舍五入到小数点后两位

double DMSToRed(int angle[])//度分秒转弧度

void RedToDMS(double red, int angle[])//弧度转度分秒

void printDMS(char word[], int angle[])//输出度分秒

void printRedToDMS(char word[], double red) //输出度分秒

void refreshDegWithDMS(double &red) //刷新使用角度刷新弧度,防止产生累积差

void ZBZS(double zb[2], double bc, double fwj) //坐标正算

void strSplit(char str[], const char spl[], char \*\*ret) //字符分割

void StrToDoubles(char str[], const char spl[], double num[10]) //字符串转doubles

void StrToReds(char str[], const char spl[], double num[10]) //字符串转Reds

void StrToXYs(char str[], const char spl[], double num[11][2]) //字符串转x,y

void printXY(char str[], double zb[2])//输出坐标

double Modulo(double zb[2])//取模

程序实现：