

利用构建不规则三角网（TIN）进行体积计算

不规则三角网（TIN）是由一系列不规则三角形组成的网络，本试题是通过读取数据文件，构建 TIN，进行体积计算。

一、数据文件读取

编程读取“正式数据.txt”文件。数据文件格式：第一行是“参考高程”，第三行开始为“点名， x 分量， y 分量， h 分量”，数据内容如表 1 所示：

表 1 数据内容和格式说明

参考高程,10.0	基准高程
Q01, 3778.594, 2885.732, 9.468	点名，坐标分量 x ,坐标分量 y , 高程
Q02, 3773.103, 2888.487, 9.533	
Q03, 3766.087, 2892.923, 9.669	
Q04, 3762.06, 2898.991, 9.996	
Q05, 3759.293, 2906.144, 10.081	
Q06, 3758.296, 2913.009, 10.138	
.....	

二、程序算法

1.凸包多边形的生成（快速凸包法）

1.1 查找四个顶点

遍历所有顶点构成的离散点集 O ，在散点中找到上下左右 4 个顶点（标记： x 最大 $P3$ ， x 最小 $P1$ ， y 最大 $P2$ ， y 最小 $P4$ ），连接 4 个顶点把散点分为 5 个区域，如图 1 所示。并在原散点集 O 中删除 4 个顶点。

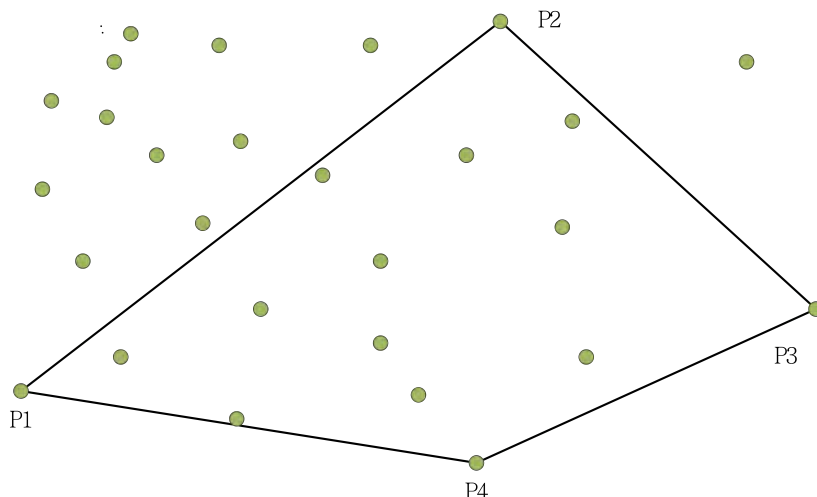


图 1 快速凸包示意图

说明: 找出 4 个顶点, P1 至 P4, 输出这 4 个点的点名和平面坐标, 小数点后保留 3 位数值。

(1) 结果保存“result.txt”中。

(2) 将 P2、P4 的平面坐标输出到“程序正确性.xls”中。输出格式为:

序号	输出格式要求	说明
1	*.***	P2 点的平面坐标 x
2	*.***	P2 点的平面坐标 y
3	*.***	P4 点的平面坐标 x
4	*.***	P4 点的平面坐标 y

1.2 利用迭代求出凸包点

创建栈 CH(存放凸包点) 把 P1 放入 CH 中。初始化 i=1。

(1) 取出 P(i), P(i+1), 遍历散点集把 P(i)-P(i+1)的左边点放入 LP 点集列表中。

判断 P(x,y)在 P1(x₁,y₁)-P2(x₂,y₂)左侧的计算公式为:

$$tem = x_1y_2 - x_2y_1 + x(y_1 - y_2) + y(x_2 - x_1) \quad (1)$$

如果 tem>0 则 P 在 P1-P2 的左侧; 如果 tem=0 则 P 在 P1-P2 的线上; 如果 tem<0 则 P 在 P1-P2 的右侧。

(2) 求出 LP 点集中距离 P1-P2 直线最远的点并记录为 F1, 并在 LP 点集中删除 F1 点。

- 如果 LP 点集为空, 把 P(i+1)放入 CH 中, 当 i= (i+1) %4 时, 返回到 (1) 继续。
- 遍历求出 LP 点集的每个点, 与 P1,P2 连成的三角形面积, 面积最大则点距离 P1-P2 最远。

判断哪个点距离 P1-P2 最远可用面积最大判断。P1(x₁,y₁)、P2(x₂,y₂)、P1(x₃,y₃)三点求面

积计算公式为:

$$tem = \frac{1}{2} |x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)| \quad (2)$$

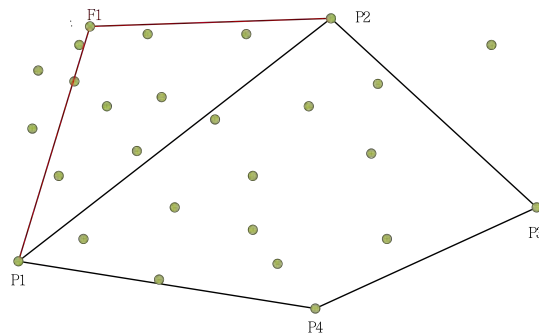


图 2 连接最远点

(3) 连接 $P(i)-F1, F1-P(i+1)$, 并求出 LP 点集中的点在线段 $P(i)-F1$ 左侧的点放入 LP1, 在线段 $F1-P(i+1)$ 左侧的点放入 LP2 中。

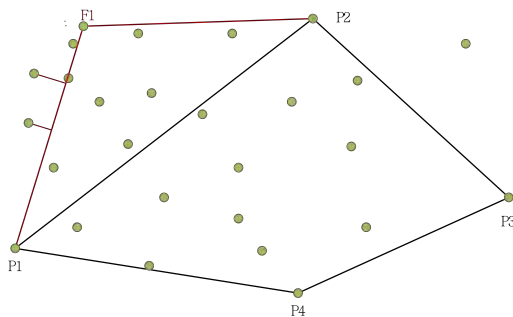


图 3 求 $P1-F1$ 最远点

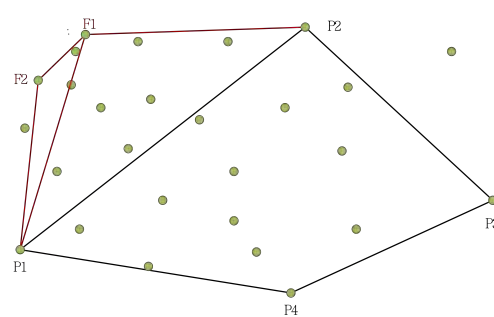


图 4 $P1-F1$ 边迭代

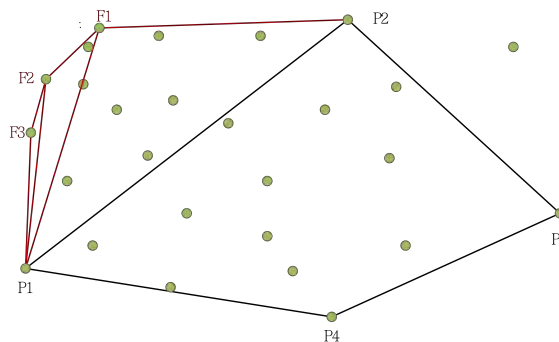


图 5 $P1-F2$ 迭代结束, 下一步迭代 $F1-P2$

- 如果 LP1 为空, 把这条线段的尾放入 CH 中。(如: 线段是 $P(i)-F1$, 则把 F1 放入 CH 中。线段是 $P(i)-F3$, 则把 F3 放入 CH 中)。
- 如果 LP1 不为空, 返回 (2), 把 LP1 当作 LP 运算。
- 如果 LP2 为空, 把这条线段的尾放入 CH 中。(如: 线段是 $F1-P(i+1)$, 则把 $P(i+1)$ 放入 CH 中。线段是 $F3-F2$, 则把 F2 放入 CH 中)。
- 如果 LP2 不为空, 返回 (2), 把 LP2 当作 LP, 运算。

(4) 如果 $i=4$, 则结束, 否则 $i = (i+1) \% 4$, 返回 (1), 继续。

说明: 计算所有凸包点, 输出这些点的点名和平面坐标, 小数点后保留 3 位数值。

(1) 结果保存“result.txt”中。

(2) 将第 3、第 5 个凸包点的平面坐标输出到“程序正确性.xls”中。输出格式为：

序号	输出格式要求	说明
5	*.***	第 3 个凸包点的平面坐标 x
6	*.***	第 3 个凸包点的平面坐标 y
7	*.***	第 5 个凸包点的平面坐标 x
8	*.***	第 5 个凸包点的平面坐标 y

2.不规则三角网的构建

2.1 生成初始三角网

计算凸包多边形 $CH\{P_1, \dots, P_n\}$ 的平面几何中心：

$$x = \frac{\sum x_{P_i}}{n}, y = \frac{\sum y_{P_i}}{n} \quad (3)$$

找到离该几何中心平面距离最近的点，记为 P0。

首先删除离散点中的凸包点，再从离散点中取出 P0，与凸包多边形的每一条相连，构成多个三角形。把生成的三角形加入到三角形列表 T1 中。

说明：找出 P0 点，小数点后保留 3 位数值。

(1) 将 P0 平面坐标和初始三角形的个数结果保存“result.txt”中。

(2) 将 P0 点的平面坐标和初始三角形的个数输出到“程序正确性.xls”中。输出格式为：

序号	输出格式要求	说明
9	*.***	P0 点的平面坐标 x
10	*.***	P0 点的平面坐标 y
11	*	初始三角形的个数

2.2 通过遍历离散点，生成平面三角网

(1) 从离散点列表中取出一一点作为待插点 P；

(2) 按顺序从 T1 中取出一个三角形 ABC，设其顶点为 A (x1,y1), B (x2,y2), C (x3,y3) 并计算该三角形外接圆的圆心 O (x0,y0) 及半径 r，计算公式为：

$$\begin{cases} x_0 = \frac{(y_2 - y_1)(y_3^2 - y_1^2 + x_3^2 - x_1^2) - (y_3 - y_1)(y_2^2 - y_1^2 + x_2^2 - x_1^2)}{2(x_3 - x_1)(y_2 - y_1) - 2(x_2 - x_1)(y_3 - y_1)} \\ y_0 = \frac{(x_2 - x_1)(x_3^2 - x_1^2 + y_3^2 - y_1^2) - (x_3 - x_1)(x_2^2 - x_1^2 + y_2^2 - y_1^2)}{2(y_3 - y_1)(x_2 - x_1) - 2(y_2 - y_1)(x_3 - x_1)} \\ r = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} \end{cases} \quad (4)$$

判断 P 点是否在三角形 ABC 外接圆的内部，若是，将该三角形剪切到影响三角形列表 T2 中（即从 T1 移动到 T2）；

(3) 重复第二步，直到 T1 中全部三角形遍历完毕；

(4) 在 T2 的三角形中寻找所有公共边，并删除这些公共边，再将剩下的边加入到边列表 S 中，然后清空 T2；

(5) 将 S 中的每条边的端点与 P 点连接，得到多个新的三角形，并将它们添加到三角形列表 T1 中；清空 S。

(6) 重复 (1) - (5) 步，直至所有离散点遍历完成。

说明：通过遍历离散点，生成平面三角网。

(1) 将所有的三角形结果保存“result.txt”中。

(2) 进行三角形数目的统计，相关结果输出到“程序正确性.xls”中。输出格式为：

序号	输出格式要求	说明
12	*	包含第 1 个凸包点的三角形个数
13	*	包含第 3 个凸包点的三角形个数
14	*	包含第 5 个凸包点的三角形个数
15	*	总的三角形个数

3.利用不规则三角网进行体积计算

3.1 计算平衡高程

设平衡高程为 H_e

$$H_e = \frac{\sum_{i=1}^n \bar{h}_i * S_i}{\sum_{i=1}^n S_i} \quad (5)$$

其中，n 为三角形个数， $\bar{h}$ 为三角形三点的平均高度，S 为三角形投影底面的面积。设三

角形由 $P_1(x_1, y_1, h_1)$ 、 $P_2(x_2, y_2, h_2)$ 和 $P_3(x_3, y_3, h_3)$ 组成，则 S_i 和 $\bar{h}$ 的计算公式为：

$$\begin{cases} S_i = \frac{|(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)|}{2} \\ \bar{h} = \frac{h_1 + h_2 + h_3}{3} \end{cases} \quad (6)$$

说明：计算平衡高程，小数点后保留 3 位数值。

(1) 将三角形投影底面面积之和、平衡高程，保存到“result.txt”中。

(2) 将三角形投影底面面积之和、平衡高程输出到“程序正确性.xls”中。输出格式为：

序号	输出格式要求	说明
16	*.***	三角形投影底面面积之和
17	*.***	平衡高程

3.2 三角形的挖填方体积计算

设参考高程为 h_0 ，从 T1 中取一个三角形 ABC，设其顶点为 $P_1(x_1, y_1, h_1)$ 、 $P_2(x_2, y_2, h_2)$ 和 $P_3(x_3, y_3, h_3)$ 。

- ① 当三角形 3 个顶点高程均小于参考高程时，为全填方，当三角形 3 个顶点高程均大于参考高程时，为全挖方，全挖方和全填方体积用下式计算：

$$V_{\text{cut}} = S_i \bar{h}_i \quad V_{\text{fill}} = S_i \bar{h}_i \quad (7)$$

$$\begin{cases} S_i = \frac{|(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)|}{2} \\ \bar{h}_i = \frac{h_1 + h_2 + h_3}{3} - h_0 \end{cases} \quad (8)$$

S_i 为三角形 ABC 的投影底面面积， $\bar{h}_i$ 为平均高程与参考高程的高差

- ② 当三角形顶点中 2 个顶点高程小于参考高程，1 个顶点高程大于参考高程时，如图 6 所示，则三角形 $P_1I_1I_2$ 为挖方区域，四边形 $I_1I_2P_3P_2$ 为填方区域。

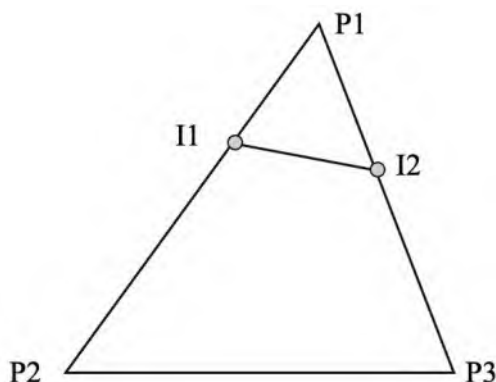


图 6 挖填方区域示意图

内插计算出三角形中 P_1P_2 边和 P_1P_3 边上高程为参考高程 h_0 的通过点 I_1 、 I_2 。设 I_1 点坐标为 (x, y) ，则有：

$$X = X_{p1} + \left| \frac{h_0 - h_{p1}}{h_{p2} - h_{p1}} \right| * (X_{p2} - X_{p1}) \quad (9)$$

$$y = y_{p1} + \left| \frac{h_0 - h_{p1}}{h_{p2} - h_{p1}} \right| * (y_{p2} - y_{p1}) \quad (10)$$

同理可计算出 I_2 点的坐标。则三角形中挖方和填方体积用下式计算：

$$\begin{cases} S_{\Delta} = \frac{|(x_{11} - x_1)(y_{12} - y_1) - (x_{12} - x_1)(y_{11} - y_1)|}{2} \\ \bar{h}_i = \frac{h_1 + h_0 + h_2}{3} - h_0 \\ V_{cut} = S_{\Delta} * \left(\frac{h_1 + h_0 + h_2}{3} - h_0 \right) \\ V_{fill} = (S_i - S_{\Delta}) * \left(\frac{h_0 + h_0 + h_2 + h_3}{4} - h_0 \right) \end{cases} \quad (11)$$

③ 当三角形顶点中 2 个顶点高程大于参考高程，1 个顶点高程小于参考高程时，

将 V_{cut} 与 V_{fill} 计算公式交换即可。

说明：计算三角形的挖填方体积，小数点后保留 3 位数。

(1) 输出所有三角形的挖填方体积，并标注挖填方类型（全挖方、全填方、既有挖方又有填方），保存到“result.txt”中。

(2) 给出不同类型的填挖方三角形个数及体积之和，输出到“程序正确性.xls”中。输出格式为：

序号	输出格式要求	说明
18	*.***	全挖方三角形的个数
19	*.***	全挖方三角形的挖方体积之和
20	*.***	全填方三角形的个数
21	*.***	全填方三角形的填方体积之和
22	*.***	有 2 个顶点低于参考高程的三角形个数
23	*.***	有 2 个顶点低于参考高程的三角形的填方体积之和
24	*.***	有 1 个顶点低于参考高程的三角形个数
25	*.***	有 1 个顶点低于参考高程的三角形的挖方体积之和

3.3 计算所有三角形中挖方和填方体积

重复 (3.1)，计算 T1 中所有三角形对应的挖方与填方体积。

说明：输出挖方总体积和填方总体积，小数点后保留 3 位数值。

(1) 保存到“result.txt”中。

(2) 输出到“程序正确性.xls”中。输出格式为：

序号	输出格式要求	说明
26	*.***	挖方总体积
27	*.***	填方总体积

3.4 计算总体积

计算总挖方体积与总填方体积，挖方体积与填方体积之和即为总体积。

说明：输出总体积，小数点后保留 3 位数值。

(1) 保存到“result.txt”中。

(2) 输出到“程序正确性.xls”中。输出格式为：

序号	输出格式要求	说明
28	*.***	总体积

三、成果要求

1. 程序正确性评价

请根据“二、程序算法”中红色文字要求，利用“正式数据.txt”进行计算，生成成果文件“程序正确性.xls”，如表 2 所示，该文件将用于程序正确性评分。

说明：该文件编程实现。

表 2 “程序正确性.xls”的文件内容与分值

序号	输出格式要求	说明	分值
1	*.***	P2 点的平面坐标 x	1 分
2	*.***	P2 点的平面坐标 y	1 分
3	*.***	P4 点的平面坐标 x	1 分
4	*.***	P4 点的平面坐标 y	1 分
5	*.***	第 3 个凸包点的平面坐标 x	1 分
6	*.***	第 3 个凸包点的平面坐标 y	1 分
7	*.***	第 5 个凸包点的平面坐标 x	1 分

8	*.***	第 5 个凸包点的平面坐标 y	1 分
9	*.***	P0 点的平面坐标 x	1 分
10	*.***	P0 点的平面坐标 y	1 分
11	*	初始三角形的个数	1 分
12	*	包含第 1 个凸包点的三角形个数	1 分
13	*	包含第 3 个凸包点的三角形个数	1 分
14	*	包含第 5 个凸包点的三角形个数	1 分
15	*	总的三角形个数	2 分
16	*.***	三角形投影底面面积之和	1 分
17	*.***	平衡高程	2 分
18	*.***	全挖方三角形的个数	1 分
19	*.***	全挖方三角形的挖方体积之和	1 分
20	*.***	全填方三角形的个数	1 分
21	*.***	全填方三角形的填方体积之和	1 分
22	*.***	有 2 个顶点低于参考高程的三角形个数	1 分
23	*.***	有 2 个顶点低于参考高程的三角形的填方体积之和	1 分
24	*.***	有 1 个顶点低于参考高程的三角形个数	1 分
25	*.***	有 1 个顶点低于参考高程的三角形的挖方体积之和	1 分
26	*.***	挖方总体积	1 分
27	*.***	填方总体积	1 分
28	*.***	总体积	1 分

2.用户界面设计

2.1. 人机交互界面设计与实现

- (1) 包括菜单、工具条、表格、图形和文本等功能；
- (2) 功能完善、可正常运行，布局合理，直观美观、人性化；

2.2. 计算报告显示功能

在用户界面提供计算报告显示功能，显示内容为：“二、程序算法”中绿底文字内容，即利用“正式数据.txt”的保存“result.txt”文件中的内容。

2.3 图形显示功能

在程序用户界面中，编程显示以下内容的图形：

- (1) 绘制凸多边形（以红色实线表示）；
- (2) 绘三角形边线（以灰色虚线表示）；

- (3) 显示“正式数据.txt”中离散点（用黑色○标记）。
- (4) 显示凸多边形点（用红色□标记）。

2.4 表格显示功能

在程序用户界面中，编程实现数据表格显示功能，在表格中显示的内容如表 3 所示

表 3 表格显示内容

序号	点名 1	点名 2	点名 3	挖方体积	填方体积	总体积
*	*	*.***	*.***	*.***	*.***	*.***

3. 计算成果输出

3.1 计算报告保存

请根据“二、程序算法”中绿底文字内容，利用“正式数据.txt”进行计算，生成结果文件“result.txt”。

3.2 计算成果保存

编程实现 Access 格式（.mdb）的计算成果保存，保存内容如表 3 所示，保存文件名称为：计算结果.mdb。

3.3 图形文件保存

编程将“2.3 图形显示功能”的图形保存为：成果图形.jpg

3.4 程序正确性保存

利用“正式数据.txt”进行计算，编程生成成果文件“程序正确性.xls”。

4. 开发文档

针对程序开发过程，撰写编程开发技术文档，并保存为“开发文档.pdf”文件。

内容包括:

- (1) 程序功能简介;
- (2) 算法设计与流程图;
- (3) 主要函数和变量说明;
- (4) 主要程序运行界面;
- (5) 使用说明。

四、评分规则

评测内容	评分细则说明
程序正确性 (30 分)	1.本部分评分根据成果文件“程序正确性.csv”输出结果进行评分，该结果文件是用“正式数据.txt”计算生成的结果。“正式数据.txt”会在考试开始 3 小时左右分发。 2.如果本项成绩低于 15 分，不能参评特等奖和一等奖（该参赛队如果是第一个提交成果，其时间不作为最短时间基准，其时间得分为最高和最低分的平均值）。
程序完整 与规范性 (15 分)	数据读取正确(读“正式数据.txt”文件) (2 分) 数据库文件保存(输出“计算结果.mdb”文件) (2 分) 文本文件保存(输出“result.txt”文件) (3 分) 图形保存（输出“成果图.jpg”文件） (2 分) 程序结构完整、函数与类结构设计清晰 (2 分) 注释规范 (2 分) 类、函数和变量命名规范 (2 分)
程序优化性 (15 分)	人机交互界面设计良好 (4 分) 表格显示符合要求 (2 分) 报告显示符合要求 (3 分) 图形显示美观 (3 分) 容错性、鲁棒性好 (3 分)
开发文档 (10 分)	程序功能简介 (2 分) 算法设计与流程图 (2 分) 主要函数和变量说明 (2 分) 主要程序运行界面 (2 分) 使用说明 (2 分)
完成时间 (30 分)	$S = \left(1 - \frac{T_i - T_1}{T_n - T_1} \times 40\%\right) \times 30$ （其中 T_1，T_i，T_n 分别表示第一组，第 i 组和最后一组提交的时间）