

目录

开头	2
1.重复记录标上记号	2
2.计算 [B].max + [B].min + [B]	3
3.按照需求对字段[D]编号	6
4.将点 x y 坐标值写入字段[E]	7
5.对图形的操作	8
小结:	8
第一层 初出茅庐-工具的使用	10
1、属性表中的 Field Calculator	10
2、toolbox 中的 Calculate field	11
第二层 牛刀小试	11
1、字段批量赋值	11
2、字段之间赋值	12
第三层 潜心修炼-变量、语法使用	13
1、基本函数的应用	13
2、简单语法的应用	14
第四层 初露锋芒-利用复杂语句完成属性计算	15
1、复杂计算	15
第五层 日新月异-简单图形计算	18
1、简单图形计算	18
2、如何判断一个面是多部分要素、还是环状要素	19
第六层 大显身手-简单图形操作	22
1、批量改变线方向	22
2、在 ArcMap 中如何将线自动闭合	22
第七层 游刃有余-复杂图形操作	23
1、如何求多边形质心到多边形边界最远的距离	23
2、提取改字段的最大值，提取这个字段的最小值，然后进行该地段+最大值+最小值的操作	25
3、怎样求得一个不规则多边形的极点?	26



ESRI 中国社区学习心得

---之 Field Calculator

为什么要把 Field Calculator 专门作为一章呢，是因为 Field Calculator 着实很强大，利用 Field Calculator 总能达到事半功倍的效果，具体有多厉害，我们下面来一一介绍。

开头

以 **tj051181** 的帖子来做开头吧

[学习笔记]ArcMap 中的 VBA 之 Field Calculator

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=54293>

<学习笔记>

ArcMap 中，无论是在属性框中右击字段打开的 Field Calculator 还是在 Arc toolbox\ Data Management Tools\ Field\ 工具箱中的 Calculate field，都支持 vba 表达式。

字段计算器，用以计算字段值。这些字段值，可以是属性表里相关字段的综合，比如百分比，分解字段等；可以是和图形有关的值，比如点的 xy 坐标，面的中心点坐标，线的长度等等；也可以是为了满足某种特定需求而创造出来的值。

比如如下问题：

1. 要找出图层某字段[A]里的重复值，并标上记号；
2. 图层某字段[B]，现在要将字段[B]中的 { 最大值 + 最小值 + 该行值 } 写入一个新字段[C]中；
3. 字段[D]需要进行编号，号码格式为"000001","000002","000003","000004"..., "000202"..;
4. 将点层的 xy 坐标按 x,y 的格式输出到 [E] 字段

1.重复记录标上记号

要找出图层某字段的重复记录，对于存储在 Personal Geodatabase 中 feature 而言，可以通过 select by attribute 找出重复值的记录：

*Sql: select * from layer where:*

[A] in (select [A] from 层名 group by [A] having count([A]) >1)

这样执行下来的结果，就是该图层中，[A]字段值重复数大于等于 2 的所有记录，而对于不支持子查询的比



如 shapefile 格式文件，可以怎么办呢？在属性表中新建一个字段，取名 "test"，类型设置为 long integer，然后右击该字段，选择 Field Calculator，勾选 advanced，写入以下代码：

```
'-----  
' ESRI Support Home > Knowledge Base > Technical Articles > Article Detail  
' 代码参见 HowTo: Identify duplicate field values  
' @ Tsonghua 090924  
'-----  
  
Static d As Object  
Static i As Long  
Dim iDup As Integer  
Dim sField  
'-----  
'这里填写需要检查的字段名  
sField = [A]  
'-----  
  
If (i = 0) Then  
Set d = CreateObject("Scripting.Dictionary")  
End If  
If (d.Exists(CStr(sField))) Then  
iDup = 1  
Else  
d.Add CStr(sField), 1  
iDup = 0  
End If  
i = i + 1
```

在 Pre_Logic VBA Script Code 下方的 A= 窗口中填 iDup

这是一段在 ESRI Support(www.support.esri.com)上找到的代码，功能就是可以讲重复值做上记号，属于重复值的记录，[test]字段值计算结果为 1，否则为 0。就是说，如果有三条记录[A]字段值分别为"aa","aa","aa"，那么有一条被标上 0，剩余 2 条被标上 1。

对重复记录做了记号，如何处理，就可以随心所欲啦。这算是一个为了满足某种需求，而重新新建字段，并计算其值的例子

2.计算 [B].max + [B].min + [B]

这个问题，就是根据属性表里其他字段来重新提取所需要的信息进行组合。



这个例子，是在论坛上看到有人问的问题。同样，去 ESRI 的支持中心去搜了一下，找到了如下代码：

```
'-----  
' ESRI Support  
' Author Christian Degrassig  
' @ Tsonghua 090924  
'-----  
  
Dim pMxDoc As IMxDocument  
Dim pFLayer As IFeatureLayer  
Dim pData As IDataStatistics  
  
Dim pCursor As ICursor  
Dim pStatResults As IStatisticsResults  
  
Static lngFlag as long  
Dim lngLayerIndex as long  
Dim strFieldName as string  
  
'因为不清楚你的字段类型，所以这里设置三个变量类型为 Variant  
Static varZMax as Variant  
Static varZMin as Variant  
Dim varVal as Variant  
  
if lngFlag = 0 then  
    '1.注：这里填你需要取最大最小值的字段名  
    strFieldName = "B"  
  
    '2.这里设置你需要计算字段值的图层所在图层索引号  
    'TOC 中，从上至下，第一层 LayerIndex=0，第二层 为 1，以此类推。  
    '这里 LayerIndex 值是第二个需要设置的地方  
    lngLayerIndex = 0  
    Set pMxDoc = ThisDocument  
    Set pFLayer = pMxDoc.FocusMap.Layer(lngLayerIndex)  
    Set pCursor = pFLayer.Search(Nothing, False)  
    Set pData = New DataStatistics  
    pData.Field = strFieldName  
    Set pData.Cursor = pCursor  
    Set pStatResults = pData.Statistics  
    varZMax = pStatResults.Maximum  
    varZMin = pStatResults.Minimum  
    lngFlag = 1  
End If  
'3.这里，第三个数请双击上面 fields 框中的对应字段
```



$varVal = varZMax + varZMin + [B]$

同样，在 "C =" 的框框中填入 varVal

字段计算器里，不管是一般的赋值表达式抑或是 advanced 里写的较复杂的表达式，其实都是在每一行运行一次，对该记录进行赋值运算。因此，如果需要将某参数值传递下去的话，可以用 "static 参数 as 类型" 这样的参数声明方式，"Dim 参数 as 类型" 的声明方式，只是在进行某条记录赋值的时候有效，一旦结束，参数值被清空。

例如，如果想对某字段赋值为 1, 2, 3, 4... 可以用如下赋值表达式：

```
Static i as long
Dim a as string
a=i+1
i=i+1
__esri_field_calculator_splitter__
a
```

注：__esri_field_calculator_splitter__ 是分割标志，具体的可以在 Field Calculator 中写一个表达式，然后点击 save，保存的文件是 ".cal" 格式，可以用记事本打开编辑。对于一个经常用的表达式，可以写好之后保存("save")起来，下次使用的时候，只需要点 "Load" 加载过来即可

对比，

```
dim i as long
Dim a as string
a=i+1
i=i+1
__esri_field_calculator_splitter__
a
```

对比发现，用 static 方式声明的结果是 1, 2, 3, 4... 而用 dim 方式声明的结果是 1, 1, 1, 1, ... 也就是说，dim 声明的参数 i 在 Field Calculator 逐行运行的过程中，没有把参数值传递下去。

回过头来分析一下这段重新分析字段，获取最大值，最小值来组成新字段值的问题

```
if lngFlag = 0 then
    '1.注：这里填你需要取最大最小值的字段名
    strFieldName = "自定义的字段名"

    '2.这里设置你需要计算字段值的图层所在图层索引号
    'TOC 中，从上至下，第一层 LayerIndex=0，第二层 为 1，以此类推.
    '这里 LayerIndex 值是第二个需要设置的地方
```



```

    lngLayerIndex = 0
    Set pMxDoc = ThisDocument
    Set pFLayer = pMxDoc.FocusMap.Layer(lngLayerIndex)
    Set pCursor = pFLayer.Search(Nothing, False)
    Set pData = New DataStatistics
    pData.Field = strFieldName
    Set pData.Cursor = pCursor
    Set pStatResults = pData.Statistics
    varZMax = pStatResults.Maximum
    varZMin = pStatResults.Minimum
    lngFlag = 1
End If

```

这段是解决问题的关键所在，它在 Field Calculator 对第一行记录赋值的时候，已经对整层进行统计并获取了最大值 `pStatResults.Maximum` 储存在参数 `varZMax` 中，同时把获取的最小值 `pStatResults.Minimum` 存在参数 `varZMin` 中。因为对整层该字段的赋值过程中，只需要获取一次最大值最小值即可，因此这里做了个标志 `lngFlag`。一旦最大最小值获取完成，就让 `lngFlag` 值为 1，以后的逐行赋值中，不再进行统计，而直接使用参数 `varZMax`, `varZMin` 的值。

最后一行 `varVal = varZMax + varZMin + [待计算字段名]`这就很简单了，直接进行加减运算。

3.按照需求对字段[D]编号

有了上面两则例子的提示，这里不难分析，其实只需要先按要求生成 1, 2, 3, 4... 然后对其格式标准化(也就是让字段长度一致，不足的补零)。

参照如下代码：

为了有更大的延展性，这段优秀的代码里将起始值、步长（就是编号每次增加的值）、字段规范长度等都设置为可以改变的参数，分别为：`lStart`，`lInterval`，`iStringLength`

自然地，可以有其他思路来解决问题：简单地生成 1, 2, 3, ..., 然后规范一下格式为"0000000" 这种。如下参考代码：

```

'-----
'rec_Number_ConstantLengthString2.cal
'Author: lanko Tchoukanski
'http://www.ian-ko.com
'@ Tsonghua

```



```

'-----
Dim sID As String
Dim sFormatString As String
sFormatString = "000000000"
sID= Format( [FID], sFormatString)

__esri_field_calculator_splitter__
sID

```

这段代码，同样地，也是我为了寻求某 GIS 交流群里群友问题答案的时候再 ESRI 支持中心上找到的，网址是 <http://www.ian-ko.com>，他们写了一批优秀的表达式，打包为"

EasyCalculate50.zip"，免费提供下载使用。同时还有个工具"ETGeoWizards99_92.exe"，一样地很优秀，作为 ArcGIS 的插件，有很多实用的功能，只是这个插件只有些基本的处理是共享，其他更高级一些处理过程还有搭建的模块就是需要注册了。

要研究 Field Calculator 中的 vba，认真学习 [ian-ko.com](http://www.ian-ko.com) 提供的免费的 EasyCalculate50，是个不错的选择。

4.将点 x y 坐标值写入字段[E]

前面提到的几个例子，都是纯粹的属性字段的操作，Field Calculator 里提供使用 vba 对图形的某些属性性质进行获取或者修改的方法。

本例中，需要获取点层数据个点的坐标值，示例代码如下：

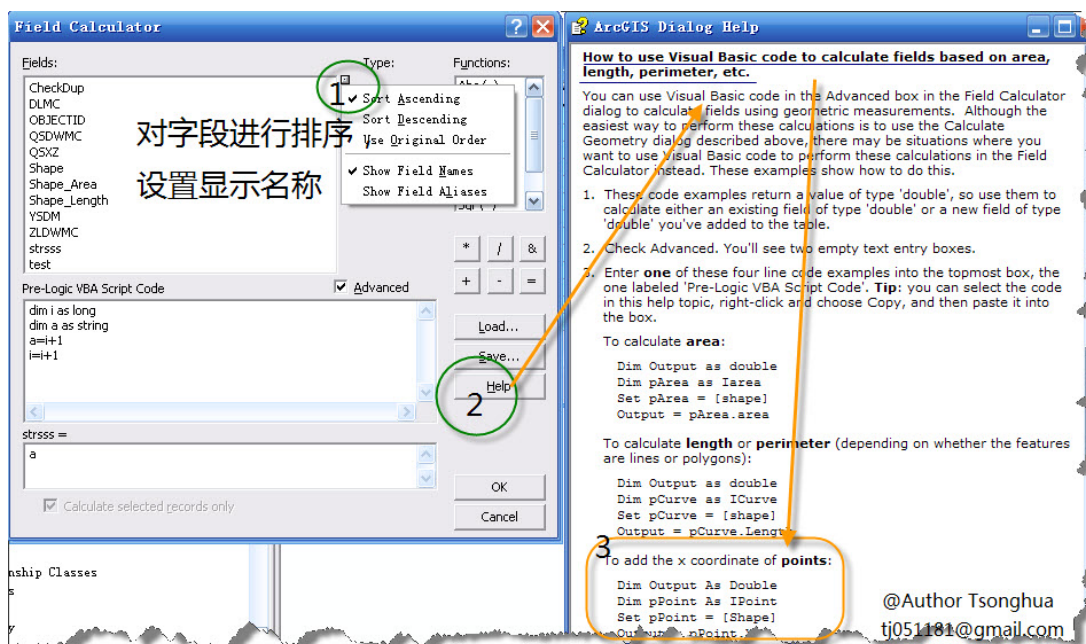
```

'-----
' @ Tsonghua 090924
'-----
Dim Output As string
Dim pPoint As IPoint
Set pPoint = [Shape]
Output = pPoint.X & "," & pPoint.Y
__esri_field_calculator_splitter__
Output

```

这一段的绝大部分是拷贝自 Field Calculator 的 Help 文档





5.对图形的操作

具体实例就不介绍了。

比如这里，来生缘老师讲 [距离小于一定数值的点之间连线](#) 这个问题里，来生缘老师在 Field Calculator 里用了一段 vba 代码实现了在线层生成由点层适合条件点连成的线的效果。

还有这里，lwf902，在 [如何求曲线上任意点到端点的距离](#) 问题中，用合适的 vba 实现求算曲线上任一点到线端点距离。

还有一个，lwf902，分享 [在 arcmap 中如何将线自动闭合](#)，直接编辑了线层，使得线终点与起点重合，达到闭合的效果。

小结：

脚本处理，在 ArcGIS desktop 随处可见。引用 [diligentpig](#) 的一段话：

Geoprocessing 是 ArcGIS 产品的三大核心（Geodatabase，Geovisualization，Geoprocessing）理念之一，也是 AGS 中 gp 服务的基石。Desktop 之所以强大，在很大程度上取决于 Geoprocessing Framework 的支撑。若要

深入掌握 Geoprocessing，除了反复操练 ArcToolbox，还得写的一手好的 Python Scripts 才行。

Geodatabase 是 ESRI 基于对地理空间数据的理解，构建合适的数据结构来存储空间数据；Geovisualization 是准确，漂亮地展现种类繁多，数量巨大，表达方式复杂的空间数据；Geoprocessing 在 ArcGIS 产品中的具体体现。脚本技术是 ArcGIS desktop 应用来进行数据分析处理的重要方式，熟练掌握一点脚本知识，是不无裨益的。

对于 vba 的学习，可以根据具体的需要，在各处查阅资料，

推荐 1: ArcGIS desktop Help

可以说，在学习的过程中需要的资料，最详实的，莫过于帮助了。体系完善，相关链接齐全，相关内容讲解十分透彻。只不过是 E 文的，读起来有点烦。

推荐 2: ESRI 中国社区

里面都是很熟悉的中文，活跃着许多热心博学的 GISer。遇到了问题，可以先点搜索，输入关键字，标题搜索、全文搜索，先自己搜一把，找找答案，如果没有解决办法，就描述好问题发帖提问。这里太感谢社区的筒子们了，比如博学睿智的来生缘老师，诲人不倦，循循善诱，跟着他学，获益良多。

推荐 3: ESRI Support

这里的积淀更深一些，基本上现在遇到的问题，都有人早就碰到过，解决过。有主题讨论、知识讲解、脚本下载、补丁下载、bug 信息等等内容。^^ 那里下载脚本资料等都是免费的。>_<论坛下载需要花积分，许多人灌水-_-..洪水泛滥了都..

ESRI 技术支持是 E 文的，不过都是些很简单，把大致意思看懂，是需要的东西，就仔细找个字典慢慢查查^^.

也有些筒子们上去用中式 E 文发帖，呵呵，看起来也很亲切。

推荐 4: 比较优秀的基于 ArcGIS 开发的一些工具

像上面推荐的 <http://www.ian-ko.com>，还有许多同样很优秀的 GIS 网站上有丰富的工具和学习资料。工欲善其事必先利其器，除了自己写工具之外，找到一些优秀的工具，可以事半功倍。

由于水平有限，文中所提到的不是之处，

tj051181 的这个帖子可谓经典中的经典。但对于有些出入江湖的新手来讲，直接看这个帖子还是有点吃力的，因为小雷已经算是武林中的风云人物，要看懂这些乃非一日之功，而且还需要有其他技艺傍身——“编程”，这里的编程不需要太高深，只要懂得基本的语法即可。

下面就吧论坛中涉及到 Field Calculator 的一些帖子整理一下，由简入深的把内容分为十层，一至三层

卢宇森(lucy11149@163.com)

雷中华(tj051181@gmail.com)

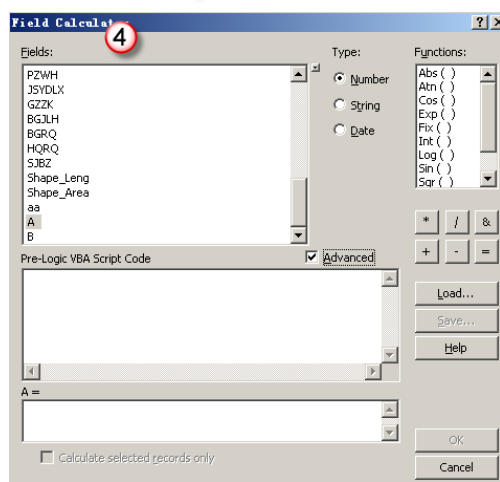
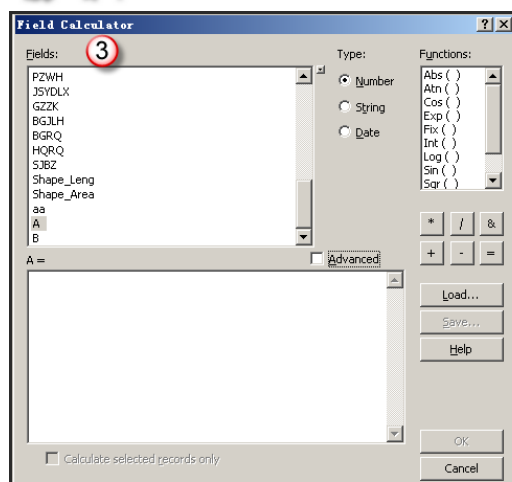
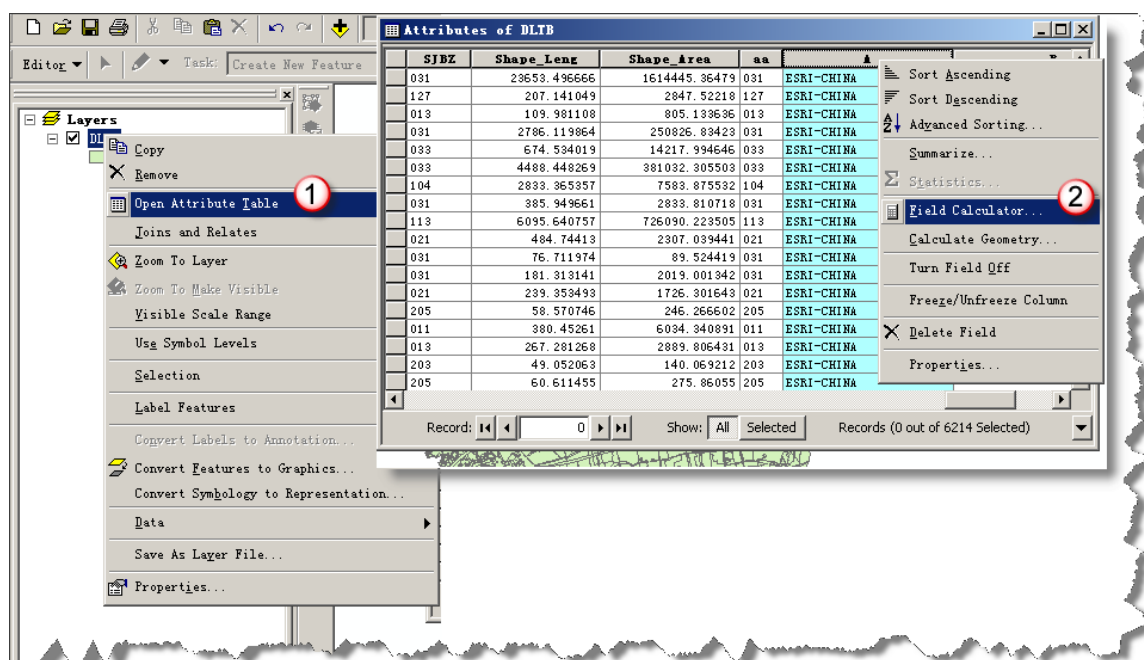


功夫是下乘,也就是入门功夫,有三层功夫可谓入门,不具三层功夫不能入门.四至六层功夫是中乘功夫,通常可以说是已登堂入室.七层功夫是中乘功夫与上乘功夫之过渡阶段,也是关键一步,常谓之炉火纯青阶段,也是中乘功夫的最后阶段.八至十层功夫为登峰造极上乘功夫,八层者登峰而未造极.凡具十层而造极者必是出类拔萃之代表人物,屈指可数,古今不多,非数年纯功不能达此境地。

第一层 初出茅庐-工具的使用

这一层来看看 Field Calculator 如何使用

1、属性表中的 Field Calculator

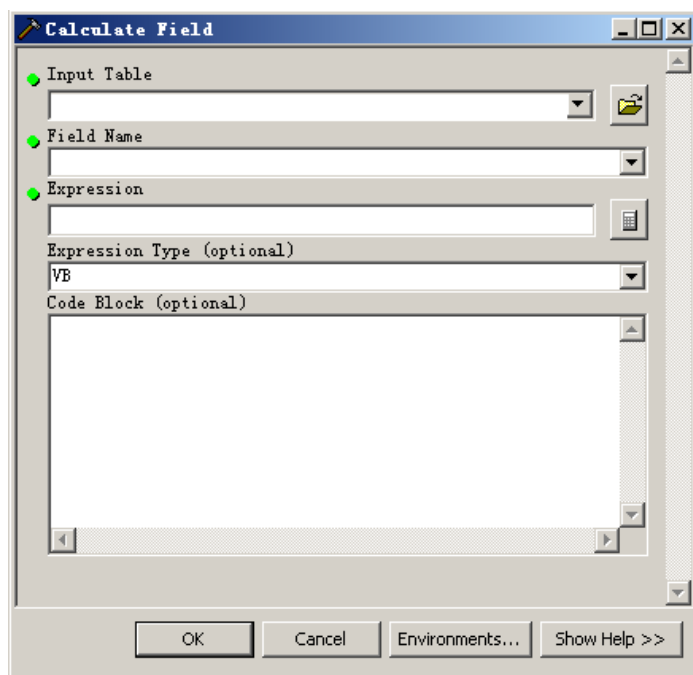


(1) 在图层上右键, 打开属性表

- (2) 在要计算的字段上右键，打开 Field Calculator
- (3)、(4) Field Calculator 窗口，(3) 简单 (simple) 模式，(4) 高级 (advanced) 模式

2、toolbox 中的 Calculate field

Data Management Tools>>fields>>Calculate Field



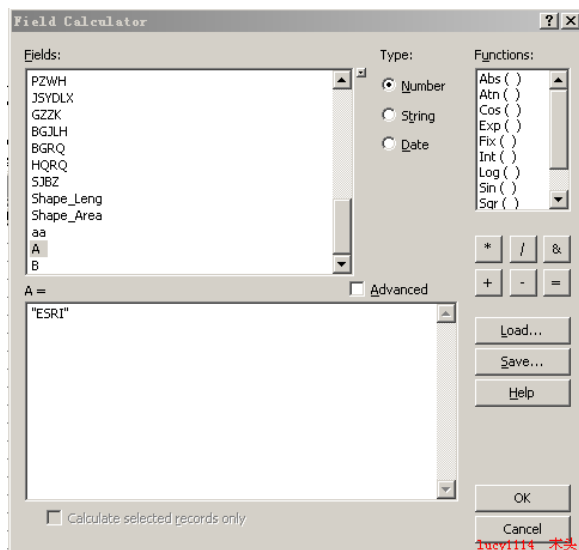
第二层 牛刀小试

知道 Field Calculator 在哪了, 下来就要看 Field Calculator 怎么用了。从最基本的开始

1、字段批量赋值

给字段 A 批量赋值 “ESRI”



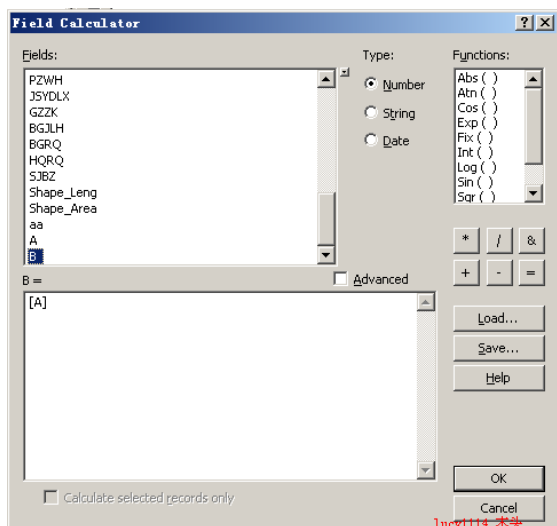


注意:

- 1.如果是字符串的话双引号一定要用英文的双引号，中文下的双引号在这里显示差不多，但点击 OK 时会报错。告诉大家个诀窍，这里英文双引号比中文双引号站的直。不同的字段类型表达式里写法不一样，这里需要注意。
- 2.如果有选中要素，默认是仅计算选中部分，也就是 Field Calculator 左下角 Calculate selected records only; 如果不想计算选中部分，则需要去掉对勾

2、字段之间赋值

把字段 A 的值赋批量赋给字段 B



注意：字段间赋值时要考虑到字段类型以及字段里填写的内容，例如文本型的字段再赋给数字型的字段时直接这么写会出错。需要通过函数进行转换，这个后面说明。



第三层 潜心修炼-变量、语法使用

1、基本函数的应用

基本函数说明:

Number 下:

Abs: 求绝对值

Atn: 求反正切值

Cos: 求余弦值

Exp: 求反对数值

Fix: 取整数部分, 与 Int 函数有区别的

Int: 取整数部分

Int 和 Fix 函数的区别在于如果 number 参数为负数时, Int 函数返回小于或等于 number 的第一个负整数, 而 Fix 函数返回大于或等于 number 参数的第一个负整数。

MyNumber = Int(99.8) ' 返回 99。

MyNumber = Fix(99.2) ' 返回 99。

MyNumber = Int(-99.8) ' 返回 -100。

MyNumber = Fix(-99.8) ' 返回-99。

MyNumber = Int(-99.2) ' 返回 -100。

MyNumber = Fix(-99.2) ' 返回 -99。

Log: 求对数值

Sin: 求正弦值

Sqr: 开方

Tan: 求正切

String 下:

Asc: 返回与字符串的第一个字母对应的 ANSI 字符代码

Chr: 将一个 ASCII 码转为相应的字符, 与它对应的是 ASC () 函数

Format: 返回根据格式 String 表达式中包含的指令设置格式的字符串, 例如 Format(13.3,"0.00")=13.30

Instr: 返回某字符串在另一字符串中第一次出现的位置

LCase: 返回字符串的小写格式, 例如 LCase("ARCGIS")="arcgis"

Left: 返回字符串左边的内容, 例如 Left("arcgis",2)="ar"

Len: 返回字符串的长度, 例如 Len("arcgis")=6

LTrim: 去掉字符串左边的空格, 例如 LTrim(" arcgis")="arcgis"

Mid: 取出字符串中间的内容, 例如 Mid("arcgis",2,1)="r"

QBColor: 返回一个 Integer 值, 该值表示对应于指定的颜色编号的 RGB 颜色代码

Right: 返回字符串右边的内容, 例如 Right("arcgis",2)="is"

RTrim: 去掉字符串右边的空格, 例如 RTrim("arcgis ")="arcgis"

Space: 返回由指定数量空格组成的字符串, 例如 MyString = "Hello" & Space(10) & "World" ' 在两个字符串之间插入 10 个空格。

StrConv: 返回按照指定方式转换的字符串。

String: 将对象转换为字符串。



Trim: 去掉字符串前后的空格, 例如 Trim(" arcgis ")="arcgis"

UCase: 返回字符串的大写格式, 例如 UCase("arcgis")="ARCGIS"

Date 下:

Date: 获取日期

DateAdd: 返回一个 Date 值, 其中包含已添加指定时间间隔的日期和时间值。

DateDiff: 两个日期之间存在的指定时间间隔的数目

DatePart: 用于计算日期并返回指定的时间间隔

Now: 获取日期+时间

例如: 要把字段 A 中的前四个字符 “ESRI” 赋值给字段 B

A	B
ESRI-CHINA	
ESRI-CHINA	
ESRI-CHINA	
ESRI-CHINA	
ESRI-CHINA	
ESRI-CHINA	

Left([A],4)

论坛中类似的帖子

在字段计算器中怎么样将字段中的某个字符去掉?

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=62782>

有没有办法在 name 字段的前四个字符后边统一加一个空格啊?

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=63993>

2、简单语法的应用

(1) 在论坛中出现最多的为根据条件赋值

参考这个帖子: 如何批量替换属性列表在的值

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=64969>

问题: 在要素类中如何将原有属性 1、2、3 三种, 批量替换成 A、B、C, 1-A, 2-B, 3-C。

解决方法:

Dim sResult as string

Dim sField as string

sField = [DLBM]

If (sField="1") Then

sResult ="A"

Elseif (sField="2") Then



```
sResult ="B"
ElseIf (sField="3") Then
sResult ="C"
End If
```

修炼到这一层，恭喜你已经学会了入门功夫。要想再上一个台阶，就需要再进行修炼。往后就需要用到基本的编程技能了，所以看看 VBA 是必要的。

参考资料：

- 1.Tools\Marcos\ Visual Basic Editor，在 VBE 环境下，点 Help 打开 Visual Basic 参考，官方资料
- 2.互联网这个宝箱中有关 VBA 的秘籍多不胜数，用 Google、百度这两把钥匙随便找些或翔实或具体的资料看看吧

第四层 初露锋芒-利用复杂语句完成属性计算

1、复杂计算

（1）唯一值编号

从 1 开始编号，一般采用[FID]+1,也可以采用全局变量

```
Static i as long
i=i+1
__esri_field_calculator_splitter__
i
```

（2）唯一值编号需要一定位数，位数不足时补 0

可参照下面帖子中的内容

[学习笔记]ArcMap 中的属性编辑

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=54290>

```
'=====
'rec_Number_ConstantLengthString.cal
'Author: lanko Tchoukanski
'http://www.ian-ko.com
'=====
Static rec As Long
Static j As Long
Dim lStart As Long
Dim lInterval As Long
Dim i As Integer
Dim iStringLength As Integer
Dim sID As String
'=====
```



```

'set the variables below
iStringLength = 7
lStart = 0
lInterval = 1
'=====
sID = ""
If (j = 0) Then
rec = lStart
Else
rec = rec + lInterval
End If
For i = 1 To iStringLength - Len(CStr(rec))
sID = sID & "0"
Next i
sID = sID & CStr(rec)
j = j + 1
__esri_field_calculator_splitter__
sID

```

也可以写成这样

```

Static i as long
Dim a as string
i=i+1
a= Format( i, "000000")
__esri_field_calculator_splitter__
a

```

(2) 怎样将图斑按不同的字段值进行自动编号

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=67620>

问题:

在属性表里面如何实现~~~根据某个字段的值进行自动编号。

比如说一个图层，这个图层里面有很多个村，一个村里有很多图斑，对这些图斑进行编号。比如 A 村有 10 个图斑，那就从 1，2，3，.....10.

B 村有 30 个图斑，那就从 1，2，3.....30

C 村有 60 个图斑，那就从 1，2，3.....60.

也就是说按村对图斑进行编号。请各位高手指教！谢谢~~

看看缘大叔是怎么解决的

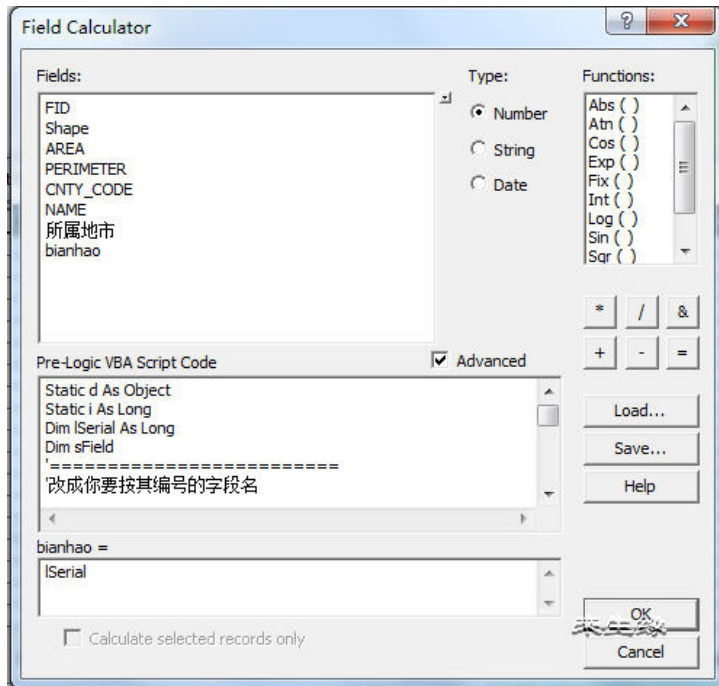
解决方法:

在属性表里新建一长整形字段，比如 bianhao，在其字段名上点右键，选 field calculator，将下列代码复制到 pro-logic vba script code 窗口内，把“所属地市”修改成你的字段名，在下边 bianhao=窗口内填入 ISerial，如附

卢宇森(lucy11149@163.com)
雷中华(tj051181@gmail.com)



图所示:



Static d As Object

Static i As Long

Dim ISerial As Long

Dim sField

'=====

'改成你要按其编号的字段名

sField = [所属地市]

'=====

If (i = 0) Then

Set d = CreateObject("Scripting.Dictionary")

End If

If (d.Exists(CStr(sField))) Then

ISerial = d.Item(CStr(sField)) + 1

d.Item(CStr(sField)) = ISerial

Else

ISerial = 1

d.Add CStr(sField), ISerial

End If

i = i + 1

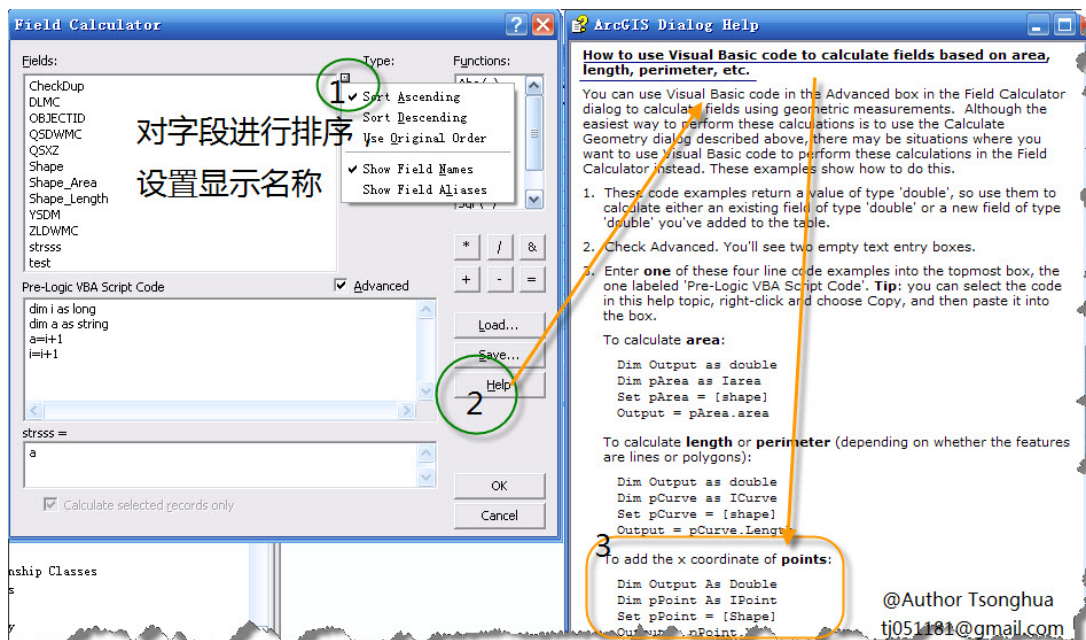
想一想，是不是之前在哪里用过这个呢？



第五层 日新月异-简单图形计算

1、简单图形计算

简单的图形计算可以参考帮助中的内容



计算面积

Dim Output as double

Dim pArea as IArea

Set pArea = [shape]

Output = pArea.area

__esri_field_calculator_splitter__

Output

再以计算线的终点 X 坐标值为例

```
'=====
'polyline_Get_X_ToPoint.cal
'Author: lanko Tchoukanski
'http://www.ian-ko.com
'=====
```

On Error Resume Next

Dim pMxDoc As IMxDocument

Dim pMap As IMap

Dim pCurve As ICurve

Dim pToPoint As IPoint

```

Dim dXTo As Double
Dim bSrefFromMap As Boolean
'=====
'Adjust the parameter below
'bSrefFromMap = True ==> the coordinates will be calculated in the projection of the Map
'bSrefFromMap = False ==> the coordinates will be calculated in the projection of the data
bSrefFromMap = True
'=====
If (Not IsNull([Shape])) Then
    Set pCurve = [Shape]
    If (Not pCurve.IsEmpty) Then
        If (bSrefFromMap) Then
            Set pMxDoc = ThisDocument
            Set pMap = pMxDoc.FocusMap
            pCurve.Project pMap.SpatialReference
        End If
        Set pToPoint = pCurve.ToPoint
        dXTo = pToPoint.X
    End If
End If
__esri_field_calculator_splitter__
dXTo

```

2、如何判断一个面是多部分要素、还是环状要素

作者: tj051181

如何判断一个面是多部分要素、还是环状要素

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=68137>

中空的或者是 multipart 的图形，可以用下面方法来挑选

(1) 在属性表新建字段

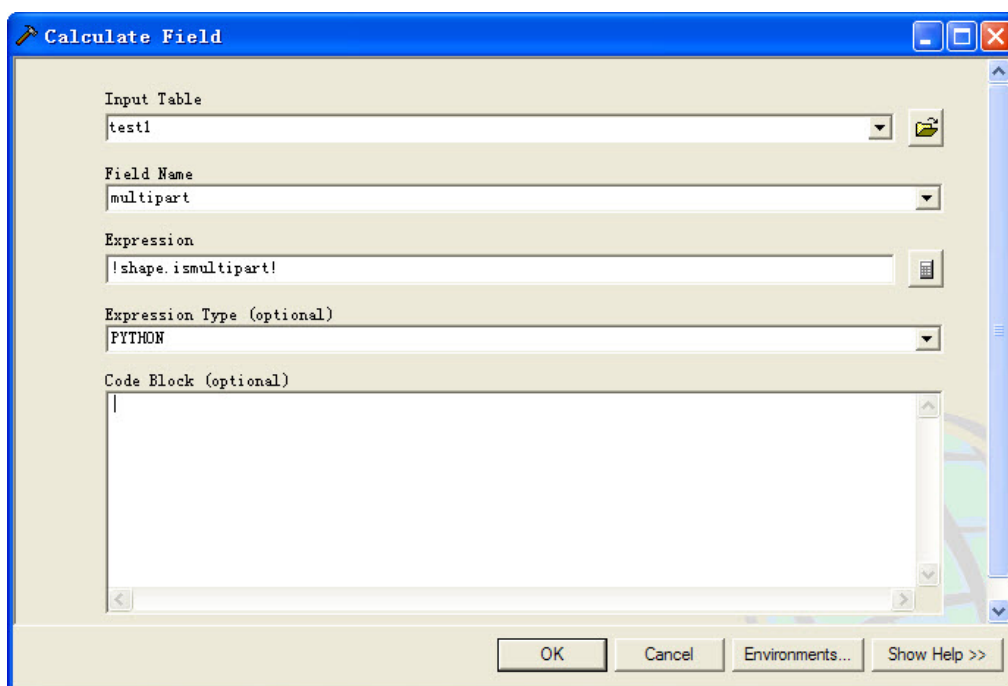
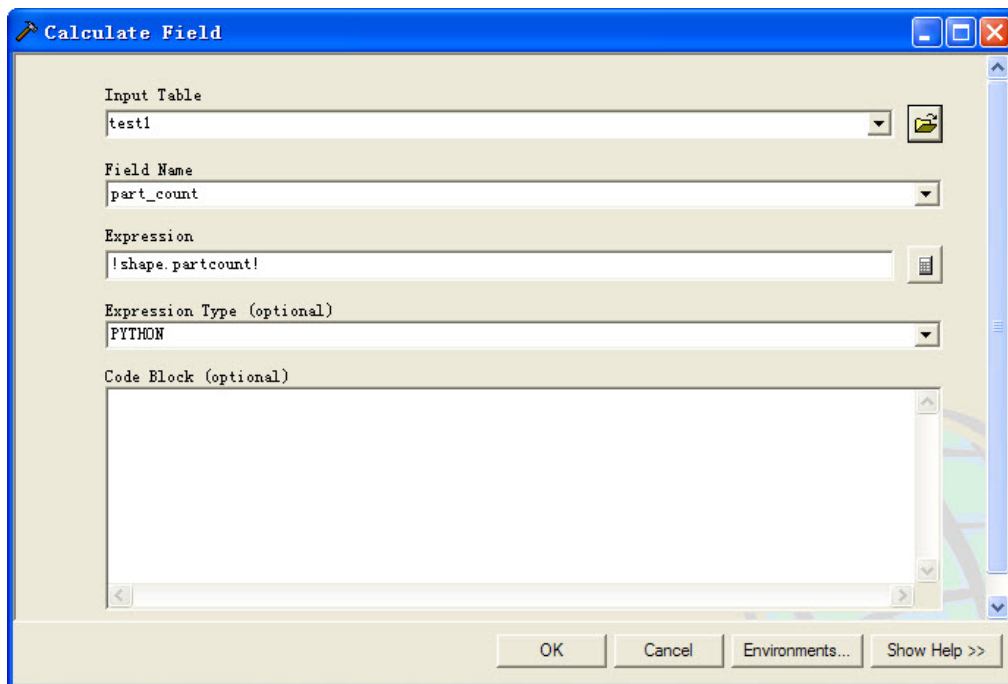
Part_Count , integer

IsMultipart , text

(2) 用 Arc toolbox\data Management tools\fields\calculate Field。

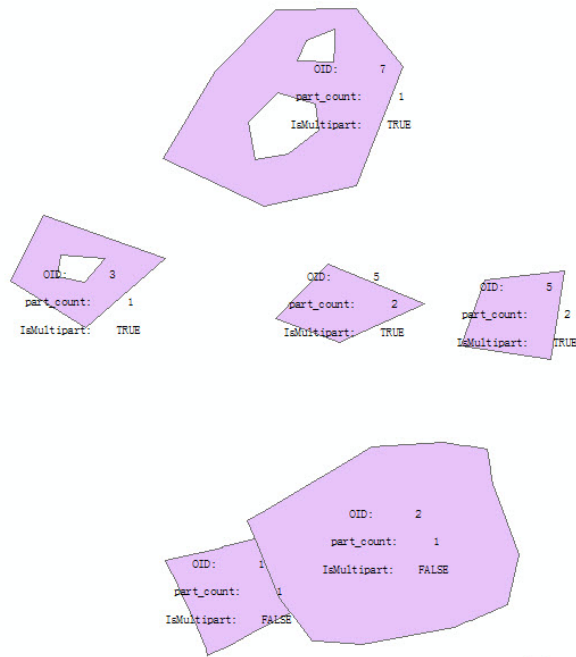
分别对上述两字段计算。设置如下图





(3) 结果如下:





Tsonghua

分析:

普通的面: PartCount =1, IsMultipart="False"

中空的面: PartCount =1, IsMultipart="True"

Multipart: PartCount >1, IsMultipart="True"

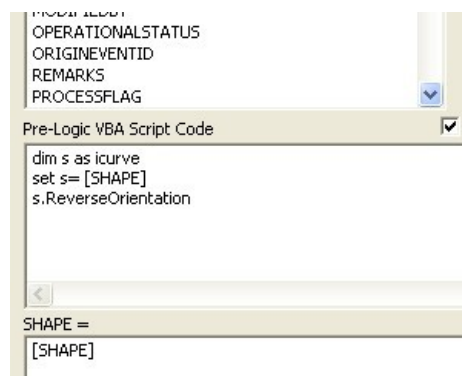
注意, 这里采用的 expression type 选择的是 python

练到这里, 就可以回头看看 [tj051181](#) 开始的帖子, 估计对于你来说就是小菜一碟了。Field Calculator 博大精深, 接着修炼吧。



第六层 大显身手-简单图形操作

1、批量改变线方向



```
Dim s as icurve
Set s=[shape]
s.ReverseOrientation
__esri_field_calculator_splitter__
[shape]
```

2、在 ArcMap 中如何将线自动闭合

可以看看 **lw902** 的这个帖子：

在 ARCMAP 中如何将线自动闭合

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=52551>

该代码在 shape 字段上计算

```
dim pCur as IPolyCurve
set pCur = [shape]
dim pPnt as IPoint
set pPnt=pCur.FromPoint
pCur.ToPoint=pPnt
'pCur.Smooth(20) '该行为平滑曲线，可选择使用
dim pLine as IPolyline
set pLine=pCur
```

如果要在其他字段进行计算，可参照



第七层 游刃有余-复杂图形操作

1、如何求多边形质心到多边形边界最远的距离

思路（将问题分解为小问题）：

- 1.取得面的质心点 pPointCenter
- 2.遍历面的节点（pVerticePoint），求每个节点到质心点距离，求得最长距离 dblMaxDis
- 3.把质心点和最长距离点生成线，写在线层中

分析（将问题关键点用接口表示）：

1. 质心点可以用 IArea Interface 的 Centroid 或者 QueryCentroid，所以令 pArea=[shape]，再 pArea.QueryCentroid pCenterPoint 可以得到其质心点为 pCenterPoint
- 2.面的节点，可以将面转为 IPointCollection，用 i = 0 to pPointCol.PointCount - 1 方式对其遍历，用每个点和面质心点求距离
3. esriGeometry.ILine.PutCoords 将质心点和最长距离节点作为起止点组成线，添加到线层

代码（具体编写、调试代码）：

```
'-----  
'@Tsonghua 20100515  
'-----  
  
Dim dblMaxDis As Double  
dblMaxDis = 0  
Dim intIdx As Integer  
intIdx = 0  
Dim pArea As IArea  
Set pArea = [Shape]  
Dim pCenterPoint As IPoint  
Set pCenterPoint = New Point  
pArea.QueryCentroid pCenterPoint  
Dim pPointCol As IPointCollection  
Set pPointCol = [Shape]  
Dim Dis As Double  
Dim pVerticePoint As IPoint  
Set pVerticePoint = New Point  
For i = 0 To pPointCol.PointCount - 1  
pPointCol.QueryPoint i, pVerticePoint
```



```

Dis = Sqr((pVerticePoint.X - pCenterPoint.X) ^ 2 + (pVerticePoint.Y - pCenterPoint.Y) ^ 2)
If Dis > dblMaxDis Then
dblMaxDis = Dis
intIdx = i
End If
Next i

```

```

Dim pMxDoc As IMxDocument
Dim pMap As IMap

```

```

Dim pSegColl As ISegmentCollection
Dim pFeatcls As IFeatureClass
Dim pFeatLayer As IFeatureLayer
Dim pFBuffer As IFeatureBuffer
Dim pFCursor As IFeatureCursor
Dim pLine As esriGeometry.ILine
Set pLine = New esriGeometry.Line
dim iLayerOrder as Integer 'iLayerOrder 是线层在 toc 中的次序 -1 的值
iLayerOrder = 0

```

```

Set pMxDoc = ThisDocument
Set pMap = pMxDoc.FocusMap
Set pFeatLayer = pMap.Layer(iLayerOrder)
Set pFeatcls = pFeatLayer.FeatureClass
Set pFBuffer = pFeatcls.CreateFeatureBuffer
Set pFCursor = pFeatcls.Insert(True)
pPointCol.QueryPoint intIdx, pVerticePoint
pLine.PutCoords pCenterPoint, pVerticePoint
Set pSegColl = New Polyline
pSegColl.AddSegment pLine
Set pFBuffer.Shape = pSegColl
pFCursor.InsertFeature pFBuffer

```

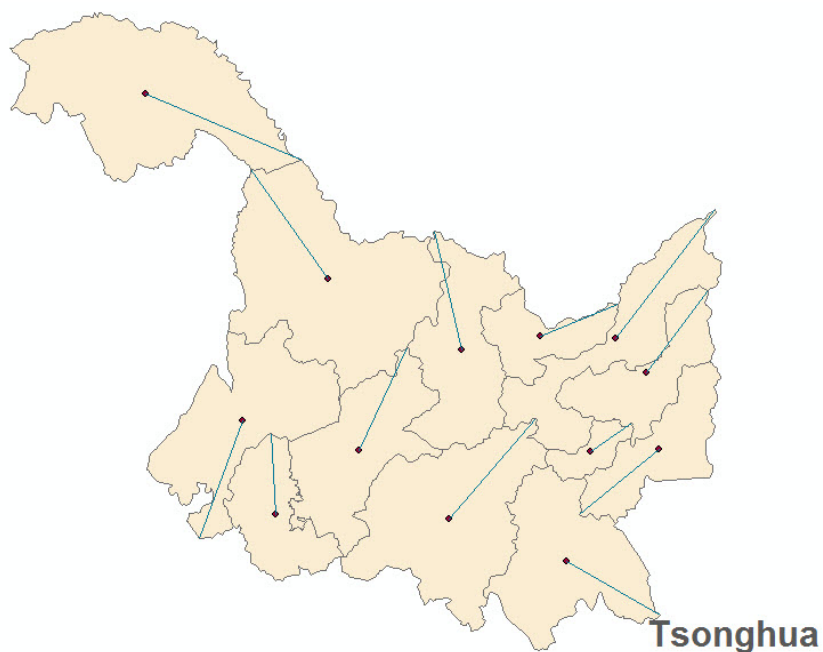
代码使用：

- 1.在面层新建字段 **dis2center**，类型设为 **double**，用于存放 最长距离
- 2.在 **arcmap** 左侧 **toc** 中，第一层放一个线层，用于存放生成的质心点到最远距离点的连线
- 3.打开面属性表，右击 **dis2center** 字段，选 **field calculator**，勾选 **advanced**，贴入代码，下面填 **dblMaxDis**，确定即可。

相似问题参考：要求得多边形各个节点到最近的服务点的距离，最终求出这些距离中最大的距离

效果图：





2、提取改字段的最大值，提取这个字段的最小值，然后进行该地段+最大值+最小值的操作

思路分析（略）

代码：

```
Dim pMxdDoc As IMxDocument
```

```
Dim pFLayer As IFeatureLayer
```

```
Dim pData As IDataStatistics
```

```
Dim pCursor As ICursor
```

```
Dim pStatResults As IStatisticsResults
```

```
Static lngFlag as long '关键点，定义 lngFlag 为全局变量
```

```
Dim lngLayerIndex as long
```

```
Dim strFieldName as string
```

```
Static varZMax as Variant
```

```
Static varZMin as Variant
```

```
Dim varVal as Variant
```

```
if lngFlag = 0 then '指定，当 lngFlag=0 才执行
```



```
strFieldName = [自定义的字段名]
```

```
lngLayerIndex = 0
```

```
Set pMxDoc = ThisDocument
```

```
Set pFLayer = pMxDoc.FocusMap.Layer(lngLayerIndex)
```

```
Set pCursor = pFLayer.Search(Nothing, False)
```

```
Set pData = New DataStatistics
```

```
pData.Field = strFieldName
```

```
Set pData.Cursor = pCursor
```

```
Set pStatResults = pData.Statistics
```

```
varZMax = pStatResults.Maximum
```

```
varZMin = pStatResults.Minimum
```

```
lngFlag = 1 '已经执行一次，也只需执行一次，因此 lngFlag=1，不在做多余的执行处理
```

```
End If
```

```
varVal= varZMax + varZMin + [待计算字段名]
```

3、怎样求得一个不规则多边形的极点？

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=61362>

思路：

通过帖子 3#介绍不难发现，问题可以通过 Feature Envelope To Polygon 得到的面 和 原面做 Intersect, output 设置为 point 求的，问题是，当有多个面，想分别求每个面的极点，这种方法在计算 EnvelopePolygon intersect 原面的时候会产生，用细颗粒实现分别求每个面的外包矩形和自身作叠置分析求的极点可以避免问题得到较好效果。

1.得到图形的 Envelope Polygon

2. Envelope Polygon （Intersect）原面 ，输出 0 维点

分析：

1. IFeature -> IEnvelope -> IPolygon 即得到 Envelope Polygon

2.set IPointCollection = ITopologicalOperator. Intersect(.., esriGeometry0Dimension)

3.所得 PointCollection 逐个输出到点层，PolygonOID 字段记上属于那个面的极点

代码：

```
Attribute VB_Name = "modmmPoints.bas"
```

```
'-----
```

```
' Author Tsonghua
```

```
' Date 20100522
```

```
' eMail tj051181@gmail.com
```

```
'-----
```



Public Sub mmPoints()

'获取当前被激活的 Data Frame

Dim pMxDoc As IMxDocument

Dim pMap As IMap

Set pMxDoc = ThisDocument

Set pMap = pMxDoc.FocusMap

'获取点层、面层

Dim pFeatLayer_polygon As IFeatureLayer

Dim pFeatLayer_point As IFeatureLayer

Set pFeatLayer_point = pMap.Layer(0)

Set pFeatLayer_polygon = pMap.Layer(2)

'获取图层对应的要素集

Dim pFeatClass_polygon As IFeatureClass

Dim pFeatClass_point As IFeatureClass

Set pFeatClass_polygon = pFeatLayer_polygon.FeatureClass

Set pFeatClass_point = pFeatLayer_point.FeatureClass

Call addPolygonOID(pFeatClass_point) '判断有没有 PolygonOID 字段，没有就加上

Dim pFeatCursor As IFeatureCursor

Dim pFeature_polygon As IFeature

Set pFeatCursor = pFeatClass_polygon.Search(Nothing, True)

Set pFeature_polygon = pFeatCursor.NextFeature

Do Until pFeature_polygon Is Nothing

Dim lngPolygonOID As Long

lngPolygonOID = pFeature_polygon.OID

Dim pEnvelope As IEnvelope

Set pEnvelope = pFeature_polygon.Extent

Dim pt1 As IPoint

Set pt1 = New Point

pt1.PutCoords pEnvelope.XMin, pEnvelope.YMin

Dim pt2 As IPoint

Set pt2 = New Point

pt2.PutCoords pEnvelope.XMax, pEnvelope.YMin

Dim pt3 As IPoint



```

Set pt3 = New Point
pt3.PutCoords pEnvelope.XMax, pEnvelope.YMax

Dim pt4 As IPoint
Set pt4 = New Point
pt4.PutCoords pEnvelope.XMin, pEnvelope.YMax

Dim pPointCol_polygon As IPointCollection
Dim pPolygon_Env As IPolygon
Set pPolygon_Env = New Polygon
Set pPointCol_polygon = pPolygon_Env
pPointCol_polygon.AddPoint pt1
pPointCol_polygon.AddPoint pt2
pPointCol_polygon.AddPoint pt3
pPointCol_polygon.AddPoint pt4
pPointCol_polygon.AddPoint pt1

Set pPolygon_Env = pPointCol_polygon
Set pPolygon_Env.SpatialReference = pMap.SpatialReference

Dim pTopo_Env As ITopologicalOperator
Set pTopo_Env = pPolygon_Env

Dim pTopo_Polygon As ITopologicalOperator
Set pTopo_Polygon = pFeature_polygon.ShapeCopy

Dim pPointCol As IPointCollection
Set pPointCol = pTopo_Polygon.Intersect(pTopo_Env.Boundary, esriGeometry0Dimension)

If pPointCol.PointCount > 0 Then
    For i = 0 To pPointCol.PointCount - 1
        Dim pFeature_point As IFeature
        Set pFeature_point = pFeatClass_point.CreateFeature
        pFeature_point.Value(pFeatClass_point.Fields.FindField("shape")) = pPointCol.Point(i)
        pFeature_point.Value(pFeatClass_point.Fields.FindField("PolygonOID")) = lngPolygonOID
        pFeature_point.Shape = pPointCol.Point(i)
        pFeature_point.Store
    Next i
End If

Set pFeature_polygon = pFeatCursor.NextFeature
Loop

```



```
Dim pAV As IActiveView
```

```
Set pAV = pMap
```

```
pAV.Refresh
```

```
MsgBox "处理完毕", vbInformation, "Tsonghua"
```

```
End Sub
```

'这部分是预处理，清除点层要素；

'判断该层是否有 PolygonOID 字段，没有就加上

```
Public Sub delPoints()
```

```
Dim pMxDoc As IMxDocument
```

```
Dim pMap As IMap
```

```
Set pMxDoc = ThisDocument
```

```
Set pMap = pMxDoc.FocusMap
```

```
Dim pFeatLayer As IFeatureLayer
```

```
Dim pFeatClass As IFeatureClass
```

```
Set pFeatLayer = pMap.Layer(0)
```

```
Set pFeatClass = pFeatLayer.FeatureClass
```

```
Dim pFeatCursor As IFeatureCursor
```

'这里就不用 Update 了

'经测试，此处是 true 的时候，delete 会出错

```
Set pFeatCursor = pFeatClass.Search(Nothing, False)
```

```
Dim pFeature As IFeature
```

```
Set pFeature = pFeatCursor.NextFeature
```

```
Do Until pFeature Is Nothing
```

```
    pFeature.Delete
```

```
    Set pFeature = pFeatCursor.NextFeature
```

```
Loop
```

```
Dim pAV As IActiveView
```

```
Set pAV = pMap
```

```
pAV.Refresh
```

```
MsgBox "点层要素清除完毕", vbInformation, "Tsonghua"
```

```
End Sub
```

```
Public Sub addPolygonOID(pFeatClass As IFeatureClass)
```

```
Dim blnField As Boolean
```

```
blnField = False
```



```

For i = 0 To pFeatClass.Fields.FieldCount - 1
    If UCase(pFeatClass.Fields.Field(i).Name) = UCase("PolygonOID") Then
        blnField = True
    End If
Next i

If blnField = False Then
    Dim pField As IField
    Set pField = New esriGeoDatabase.Field
    Dim pFieldEdit As IFieldEdit
    Set pFieldEdit = pField

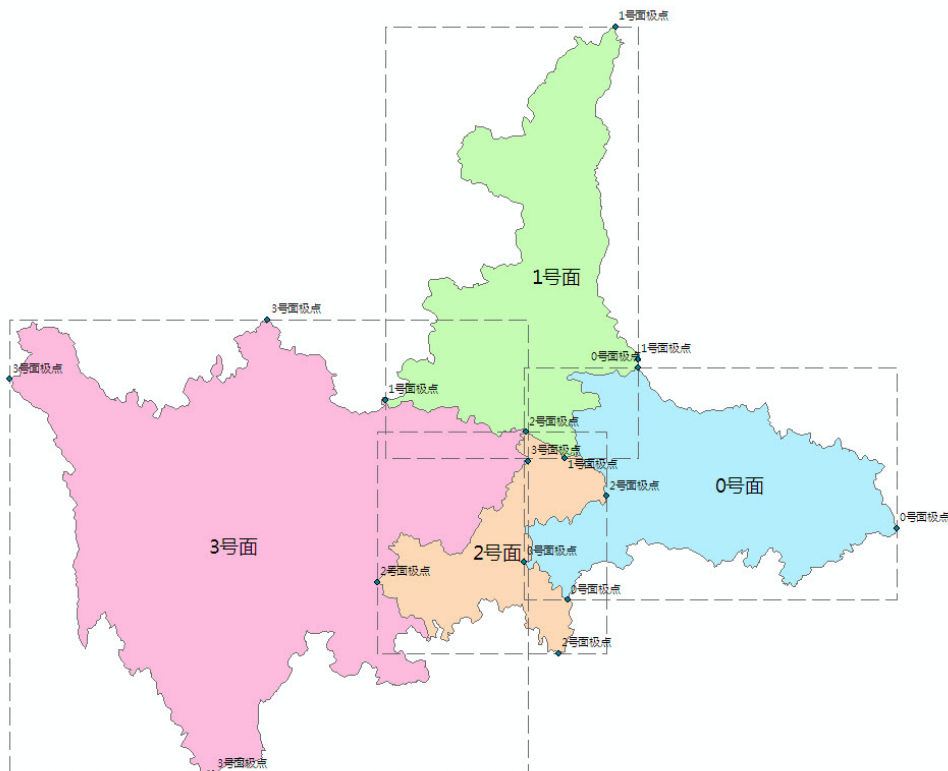
    pFieldEdit.Name = "PolygonOID"
    pFieldEdit.AliasName = "PolygonOID"
    pFieldEdit.Type = esriFieldType.esriFieldTypeInteger

    pFeatClass.AddField pField

End If
End Sub

```

效果图:



对此类应用可以到 ESRI 论坛中进行详细查看

依据已知点及该点某属性，做线

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=54587>

求一段 field calculator 的 vb 代码

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=53966>

如何绘制一点至平面内任意点之间的距离图层

<http://bbs.esrichina-bj.cn/ESRI/thread-69076-1-1.html>

在 ESRI 用 AO 框架搭建起应用程序范例-- ArcGIS Desktop 中可以用许多编程语言对其进行二次开发，相应的资料可以参考安装 Desktop 安装盘里的 SDK For .. (VBA、Java、.net..) 后 Developer Help 里的 .. Developer Help。其中在提供了完善的代码编写调试运行环境 VBE (Visual Basic Editor) 里可以用细颗粒搭建适合自己需求的应用工具，参考资料有上面提到的 VBA Developer Help，还有 [ArcMap.VBA ArcGISBook.doc \(中文\)](#) 等经典范例。

除了 VBE，在 Field Calculator 中同样可以使用 AO 底层细颗粒来实现属性、图形的操作。但是 Field Calculator 中没有调试环境，所以建议借用 VBE 环境来编写调试所写代码 (脚本)，以提高代码效率与正确率。

我们知道，Field Calculator 在执行的时候，会自动遍历要素集的每一个要素进行运算处理，所以在 VBE 环境下写的处理过程需要遍历要素集的，转为 Field Calculator 脚本时取循环里面的部分即可，循环体外部省略；如果需要多次遍历要素集，可以用 IFeatureClass.Search() 实现；如果不需循环只需要执行一次，在 Field Calculator 中加上一个标志 (比如 Flag)，在执行一次后让 flag 值变化达到不浪费地每次执行的目的

eg1. Field Calculator 中的遍历 (执行自动遍历每个要素)

Dim pGeoCol As IGeometryCollection

Set pGeoCol = [Shape] '这是获取每个要素的图形，在 Field Calculator 执行时，自动遍历

eg2. 在 Field Calculator 中再次使用遍历 (同 VBE 中写法)

Dim pMxdoc As IMxDocument

Dim pMap As IMap

Set pMxdoc = ThisDocument

Set pMap = pMxdoc.FocusMap

Dim pFeatLayer As IFeatureLayer

Set pFeatLayer = pMap.Layer(0)

Dim pFeatClass As IFeatureClass

Set pFeatClass = pFeatLayer.FeatureClass

Dim pFeatCursor As IFeatureCursor



```

Set pFeatCursor = pFeatClass.Search(pSF, True) '用 IFeatureClass.Search()来遍历(可以设置图属过滤条件)
Dim pFeatureSelected As IFeature
Set pFeatureSelected = pFeatCursor.NextFeature
Do Until pFeatureSelected Is Nothing

```

...

```

Set pFeatureSelected = pFeatCursor.NextFeature

```

Loop

eg3.加标志使得 Field Calculator 中不做不必要的多次执行

```

static flag as Boolean '这一步是关键，定义的 flag 需要为全局变量

```

```

if flag = false then '限制 flag 为 false 才执行下面的代码

```

```

... ' (处理过程)

```

```

flag = true '这里在处理完后标志变为 true，使得 field calculator 不再执行 if 中过程

```

```

end if

```

VBA 语言写 AO，除了 VBA Developer Help，ArcMap.VBA_ArcGISBook.doc（中文）也是不错的资料；再具体到 Field Calculator 中，<http://www.ian-ko.com/> 的 Easy Calculate 5.0 是极佳的参考。

修炼到这里你就已经达到炉火纯青的地步了，要想登峰造极就得继续潜心修炼。

关于属性计算的功用可以到下面帖子讨论

<http://bbs.esrichina-bj.cn/ESRI/viewthread.php?tid=55774>

@Shanghai

6/22/2010 12:17:28 AM

卢宇森(lucy11149@163.com)
雷中华(tj051181@gmail.com)

