

国家自然科学基金重点资助项目(59634030)

矿区沉陷 预测预报系统

吴侃 周鸣 著

中国矿业大学出版社

前 言

矿区沉陷预测预报是矿山开采沉陷学的核心内容之一。它对开采沉陷的理论研究和生产实践都有重要的意义。利用预测的结果可以定量地研究受开采影响的岩层与地表移动在时间上和空间上的分布规律。预测所得的结果常被用来判别建筑物是否受开采影响和受开采影响的程度,作为受影响建筑物进行维修、加固或就地重建或采取地下开采措施的依据;可以根据预测所得的结果判断铁路下开采的可能性,估算铁路维修工作量和材料用量,安排维修计划;可以根据预测的结果全面掌握矿区土地的塌陷情况,包括塌陷面积、塌陷深度,以便开展矿区土地复垦,保护矿区生态环境等。过于保守的(偏大)预测结果将导致花费不必要的保护费用,造成浪费;过于低估影响的(偏小)预测结果可能导致保护措施不足,使保护对象受到破坏,造成不必要的经济损失。

本书所介绍的矿区沉陷预测预报系统是在作者先前开发的地表移动变形预计程序的基础上,经过不断吸收新的研究成果(“211”工程建设和国家自然科学基金重点资助项目成果),不断完善其使用功能而形成的。先前的版本已在淮南、兖州、平顶山矿区和煤田地质局等单位进行了大量的应用,获得了良好的效果。而新形成的预测预报系统也在许多科研课题的研究工作和工程设计中发挥了重要的作用。例如,“淮北石台矿铁路桥下采煤”、“淮河堤坝下采煤”、“河南台前县黄河堤坝下采煤可行性研究”、“资新矿务局唐洞矿八一井工业广场下采煤”及“郑煤集团米村矿村下采煤”等项目都应用了该系统。长期的实践证明,该系统的预测预报

结果是准确的和可靠的。

矿区沉陷预测预报系统包含了解决矿区开采沉陷及“三下”采煤问题所需要的大部分数据处理问题。主要内容有:地表观测、岩体内部监测数据处理,地表动态、稳态移动变形预测及绘图,动态、稳态预测参数的求取,矿区土地沉陷预报和地表移动变形动态演示等。这是一个非常实用的系统。该系统是在 WINDOWS 环境下,使用 VISUAL BASIC 开发而成的。在系统开发过程中,得到了中国矿业大学王悦汉教授、马伟民教授、何国清教授、邓喀中教授和开采损害及防护研究所全体同事的大力支持,研究生 98 靳建明同学和测量 94 本科生魏征军同学帮助调试了部分程序,在此一并表示衷心的感谢。书中引用了一些单位和个人发表的文献资料,在此对所引文献的作者表示衷心的感谢。

由于作者水平有限,书中缺点和错误在所难免,恳请读者批评指正。

作 者

1999 年 5 月

目 录

1	概述	1
1.1	矿区开采沉陷预测预报系统基本功能	1
1.2	系统开发环境	2
1.3	VB 基本概念和程序设计方法	4
1.4	软件、硬件需求	21
1.5	系统安装及运行	21
1.6	书写器使用方法简介	23
2	矿区沉陷数据采集方法	24
2.1	地表移动观测站	24
2.2	岩体内部监测基本原理	27
3	地表观测数据处理及实例	34
3.1	数据处理方法	34
3.2	地表观测数据处理子系统的可视化实现	35
3.3	使用指南	68
3.4	地表观测数据处理实例	69
4	岩体内部监测数据处理	72
4.1	数据处理方法	72
4.2	观测数据准备	75
4.3	计算生成的数据文件格式	76
4.4	使用指南	77
4.5	实例	80
5	地表沉陷动态、稳态预测及绘图	85

5.1	基本计算方法	85
5.2	剖面曲线绘制计算方法	86
5.3	等值线图绘制计算方法	96
5.4	使用指南	113
5.5	动态、稳态预计实例	128
5.6	用 Winsurfer 绘图	134
6	开采沉陷参数求取及分析	137
6.1	求参数基本原理	137
6.2	模矢法求参数的计算机实施模型	138
6.3	使用指南	140
6.4	求参数效果分析及讨论	148
6.5	实例	155
7	开采沉陷动态预计参数与实时预计	157
7.1	开采沉陷动态预计参数经验公式	157
7.2	应用时序分析法求动态预计参数基本理论	159
7.3	时间序列分析实例	175
7.4	使用指南	185
7.5	实时预计及算例	186
8	矿区土地沉陷预报	188
8.1	基本算法	188
8.2	数据组织	190
8.3	使用指南	191
8.4	实例	192
9	地表移动变形动态演示	194
9.1	动态演示的基本原理	194
9.2	动态演示可视化实行	194
9.3	使用指南	195
	参考文献	198

1 概述

矿区开采沉陷预测预报系统,简称 MSPS 系统,包含了解决矿区开采沉陷及“三下”采煤问题所需要的大部分数据处理问题。这是一个非常实用的系统。该系统是在 WINDOWS 环境下,使用 VISUAL BASIC 开发而成的。

1.1 矿区开采沉陷预测预报系统基本功能

MSPS 系统的基本功能为(见图 1-1, MSPS 系统主窗体):

① 地表沉陷数据处理:该项功能完成地表移动观测站观测成果的计算及成果整理工作。

② 岩体内部数据处理:完成岩体内部钻孔监测成果的计算及整理。

③ 地表沉陷预计:对任意开采引起的地表沉陷问题作动态和最终稳定态的预计,并绘制有关图形。

④ 求开采沉陷预计参数:采用模矢法,用全盆地中任意实测点寻求开采沉陷最优参数,并提供反映拟合效果的拟合曲线图和拟合三维柱状图。

⑤ 地表沉陷实时预计:在开采过程中,用新采集的观测数据求出预计参数,用该参数预计随后开采可能出现的地表移动变形。由于随时利用已获得的实测数据,不断地修正预计结果,因此,预计结果可以达到很高的精度。

⑥ 矿区土地沉陷预报:分析矿区已塌陷的土地的范围、面积和塌陷的程度,预报矿区在未来任一时刻土地塌陷的范围、面积和塌陷的程度。

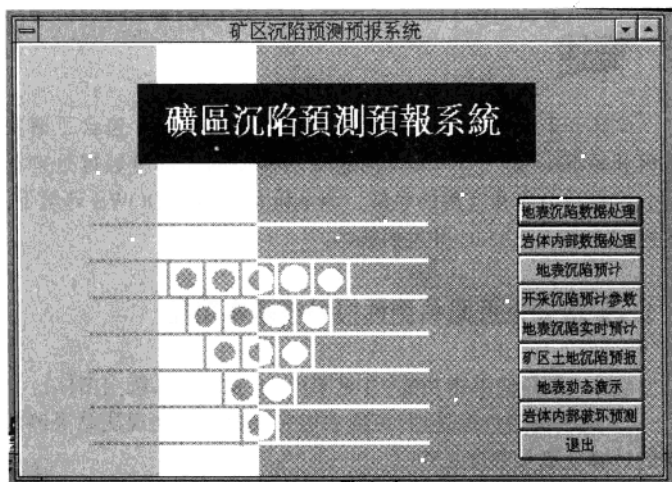


图 1-1

⑦ 地表移动变形动态模拟:模拟开采过程中主断面上地表移动变形的形成过程,动画显示开采过程中下沉盆地的形成过程。

⑧ 岩体内部破坏预测:用动态力学模型模拟岩体内部的破坏过程,计算岩体内部受开采影响的动态位移场(本书不详细介绍)。

1.2 系统开发环境

BASIC 作为一种程序设计语言从其创始人发明它开始,对计算机的普及推广起到了不可估量的作用。

BASIC 语言从产生到现在,也经历了一个逐渐演化的过程。

早期的 BASIC 语言功能较为简单,提供解释程序,只能满足简单程序设计的要求,为开发者提供的开发工具也只有简单的行编辑。这一时期的 BASIC 语言可以称为第一代 BASIC 语言。

到 80 年代,软件公司为软件开发者提供的 BASIC 语言功能大为增强,在保留原有解释功能的基础上,提供了与其他高级语言一样的编译功能和其他功能。主要产品有微软公司的 QUICK BASIC, BORLAND 公司的 TURBO BASIC 以及 TRUE BASIC 等。这一时期的 BASIC 语言可以称为第二代 BASIC 语言。

随着微电子技术的飞速发展,计算机操作系统得到不断发展和完善。WINDOWS 在微型计算机上为用户提供了一个具有多任务环境、图形用户界面、动态数据交换、对象链接与嵌入等功能的操作环境,成为当今非常受欢迎的软件开发环境。众多的用户已经从 DOS 软件的开发转向 WINDOWS。一方面 WINDOWS 为用户提供了前所未有的先进功能;另一方面也使得 WINDOWS 的软件开发增加了难度。在众多的开发工具中,选择 VISUAL BASIC(以下简称 VB)是比较理想的。使用 VB 不但可以感受到 WINDOWS 带来的新概念、新技术以及新的开发方法,而且开发效率也较高。

VB 与传统 DOS 下的 BASIC 或 QBASIC 最大的差别在于 VB 运用面向对象(Object Oriented)的概念建立一个事件驱动的环境,供用户直接调用和使用,程序设计者只要专心数据的运算处理,其余诸如 WINDOWS 的应用程序下所见的滚动条、按钮、下拉式菜单、核对框、列表框和存取文件对话框等等,都已有写好的子程序(WINDOWS 称为对象)供用户调用,而每一个对象都有许多事件、属性方法,供用户填入适当值或程序码组成一个应用程序。

MSPS 系统即是在 WINDOWS 环境下,采用 VB 开发而成的。

1.3 VB 基本概念和程序设计方法^[1]

开发完成后的 MSPS 系统是一个 WINDOWS 应用程序, 直接脱离 VB 环境在 WINDOWS 下就可以执行。用户可以不掌握 VB 的有关知识(用户可以不阅读这一节内容), 但了解一点 VB 的基本知识对 MSPS 系统的灵活应用和进一步开发是有益的。

1.3.1 VB 基本概念

1.3.1.1 WINDOWS 应用程序

在微型计算机操作系统 MS-DOS 下运行的 MICROSOFT WINDOWS 是一个基于图形用户界面的多任务、多窗口操作环境。MICROSOFT WINDOWS 特别友好的用户界面和与设备无关的图形设备接口, 已经被程序员和用户所接受, 成为事实上的图形操作环境标准。

WINDOWS 应用程序, 是在 WINDOWS 环境下运行的扩展名为 .EXE 的可执行文件。

WINDOWS 应用程序带给用户许多方便:

① 操作界面是标准的, 并且模拟现实生活, 所以特别简单直观, 只需用键盘和鼠标对图标、对话框、菜单和按钮等标准界面元素进行选择 and 操作就可完成程序对话。用户只要学会使用一种软件, 就能很轻松地学会使用所有的软件。

② 不必为每一个应用程序建立专门的外设驱动程序。

③ 多任务可以使用户不必考虑将一些附加程序常驻内存, 而且随时可以进行程序间的数据互访和通信。

④ 不必再担心内存不够的问题。

⑤ 使用上比较自由和随意。

对编程人员来说, 则有如下好处:

① 一般不需要为外部设备的配置及外设驱动程序的编写伤

脑筋。

② 使用标准的图形操作界面,减轻了为设计界面及用户操作方法所带来的沉重负担。

③ WINDOWS 的 API(应用过程接口)有大量丰富的标准化了的图形例程可供编程使用。

④ 编程不像 DOS 下那么拘泥。

1.3.1.2 WINDOWS 环境下的程序运行机制

从编写 DOS 应用程序转到 WINDOWS 应用程序,需要一些观念上的转变。其中最重要的一个转变是:从编写 DOS 下顺序的、过程驱动的程序,转变到编写 WINDOWS 下随机的、事件驱动(Event-Driven)的程序。

WINDOWS 下的应用程序是事件驱动的:由一些不可能知道先后顺序的随机发生的事件来控制与程序的对话。这些事件的识别和处理由 WINDOWS 的管理系统来安排。

WINDOWS 管理系统本身的任务就是不断对事件进行等待、监视、接收、排队、分析和响应。在事件驱动程序的编写过程中,编程人员的主要任务是对每一可能发生的事件编写出处理这一事件的程序代码,即编写事件过程。

1.3.1.3 面向对象编程

用传统的方法编写 WINDOWS 应用程序,并没有给编程人员带来多大益处。沿用传统的方法编写 WINDOWS 应用程序,想要提高效率,几乎是行不通的。编写这种复杂庞大的程序,至少要用面向对象(OOP)的编程方法。

用面向对象的编程方法编写 WINDOWS 应用程序的主要思想是,有关窗口、对话框、命令按钮、组合框和菜单等这些在每个 WINDOWS 应用程序中都要用到的标准界面元素的代码编写,已经标准化,形成了“标准件”。界面元素的功能、特征、响应的事件及具体用法等,都被编写成代码放在同一个模块中被“封装”起来、

成为一个对象,作为一个编程语言的基本元素包含在开发工具中。在开发人员编程时,需要考虑的只是如何组织安排这些对象,让它们形成一个用户界面,去完成特定的功能。

在使用这些对象时,要考虑对象的属性(Properties)和方法(Method)。

(1) 对象的属性

一个对象不仅有确定的形状,还有区别于其他对象的一些本质的特性。一个对象的属性决定了这个对象的类型。同一类型的对象,也有一个个相互区别的具体的对象实体,即同一类型的对象中,各个对象实体的属性值还是有区别的。对象的每一个属性值都是可以度量的。同一类型的对象中,对象的属性值与对象属性的关系是个别与一般的关系。

在设计用户界面时,要设计一个具体的、实实在在的对象实体,即给某对象类型赋予“个性”。例如,“窗口”是一个对象类型,它有一些固定的属性(大小、颜色和标题等)。在给定具体的属性值前,它是一个“抽象”的窗口,无法知道它到底是一个什么样的具体窗口,无法将它在屏幕上显示出来,因为它的属性值还是不确定的。如果给定:宽300点、高200点、颜色是白色、标题是‘XXX’,此时,这一窗口才是一个具有确定的属性值的对象实体,可以把它确确实实地在屏幕上显示出来。

(2) 对象的方法

对象的另一个特征就是它的方法。为了赋予对象能作相应的动作,通常预先把动作的程序代码编好封装在对象里。这种对象经常要用到的程序代码叫对象的方法。

既有属性又有方法的对象,就是有外形、能动作的“活的”对象。当给出对象时,它的方法代码已经被封装在其中了。有了方法,只要在编程过程中使用它就行了。

1.3.1.4 VB 中的面向对象的编程方法

VB 本身也是一个 WINDOWS 应用程序。使用 VB 是为了编写 WINDOWS 应用程序。

(1) 可视的程序设计环境

在 VB 中,更进一步把一些对象变成可以看得见的(即可视的)具体图形。所有的界面元素(即对象)是一些现成的图形块,显示在屏幕上。使用者在设计用户界面时,只要用鼠标从现成的对象中选择一个或几个,然后再填少量的表格就可以完成设计。在设计过程中,屏幕上就可以显示出在运行中即将显现的用户界面。如果感觉不合适,可以修改它,直至满意为止。这种可视的程序设计环境,给使用者带来了极大的方便。这使界面设计非常简单,主要工作不是设计用户界面,而是编写对象所响应的事件过程。

(2) VB 所能识别的对象

VB 环境下提供的对象是有限的,它们的类型在 VB 中有明确的规定。对象的类型名是 VB 的关键字。下面分别列出了 VB 所提供的对象名字。

全局使用对象:

App(应用程序), Clipboard(剪贴板), Debug(调试窗口), Printer(打印机)和 Screen(屏幕)。

用户界面对象:

Form(窗体), MDI Form(多文档窗体)和 Control(控件)。

其中,控件是窗体内布置的许多小的标准件的总称,它包括:

Check box(复选框), Combo box(组合框), Command button(命令按钮), Data(数据), Directory list box(目录列表框), Drive list box(驱动器列表框), File list box(文件列表框), Frame box(分类框), Grid(网格), Image(图像), Label(标签), Line(直线), List box(列表框), Menu(菜单), OLE(对象链接与嵌入), Option Button(单选钮), Picture box(图片框), Shape(外形), Scroll bar(滚动

条)和 Timer(计时器)。

数据库管理对象:

Database(数据库), Dynaset(动态记录集), Field(字段), Fields(字段集), Index(索引), QueryDef(查询定义), Snapshot(静止记录集), Table(表), TableDef(表定义)和 TableDefs(表定义集)。

(3) VB 对象的属性、方法和事件

在 VB 中,每一对象类型都有若干种属性,这是明确规定好的,它是 VB 的关键字。

在 VB 中,每一对象类型都有若干种方法,这也是明确规定好的,它是 VB 的关键字。

用 VB 编写的 WINDOWS 应用程序所能感知的事件是有限的。每一对象所能响应的事件(即对象的事件)种类有明确的规定,而且对每一事件的确切含义也有界定。每一个事件都有一个名字,它是 VB 的关键字。

1.3.1.5 语句、函数和过程

语句是 VB 语言可以识别的最小的完整语言单位。函数是有返回值的语句。过程是语句的扩充,即使用者自己用 VB 语句和函数编造的新语句。

(1) 基本语句

注释语句:为程序加注解;

赋值语句:用于对变量的赋值;

多重语句:当多条语句排在一行时,用“:”将各语句分开。

(2) 控制结构

控制结构:顺序结构、分支结构和循环结构,分别对应顺序语句、分支语句和循环语句。

(3) 程序过程

VB 有两类过程:事件过程和通用过程。事件过程是使用者为每一对象事件所编写的响应事件的程序代码;通用过程是使用

者自己建立的、供其他程序(包括事件过程)调用的通用程序代码。

事件过程:一个对象能响应的每个事件都可以有一个程序过程与之关联,构成一个事件过程。当用户或系统在某一对象上触发某种事件时,就会引发去执行相应的事件过程,完成特定的功能。事件过程是 WINDOWS 应用程序,也是 VB 应用程序的主体。WINDOWS 管理系统会将属于 VB 应用程序的事件通知 VB 应用程序。VB 应用程序根据事件的性质去执行相应的事件过程。事件过程是在窗体所对应的编码窗口中建立的。事件过程全部是窗体级的,即只能在窗体下创建。这些事件过程和窗体界面一起存于窗体文件(.frm)中。

通用过程:许多事件过程中,都需要调用完成某种公共功能的程序模块,这时使用者可以自己编写这些公共代码模块作为通用过程来使用。通用过程一般分为可被一个窗体调用的窗体级通用过程和可被所有窗体调用的模块级通用过程。窗体级通用过程随窗体和事件过程一起存入窗体文件中。模块级通用过程可以被整个应用程序的所有事件过程调用。它是通过生成一个模块文件(.bas)来创建的。

(4) 自定义函数

自定义函数可以看成有返回值的过程。它的用途和建立方法都类似于通用过程。但是,通用过程只有参数传递给过程,而没有返回值。调用自定义函数时有参数传入函数,函数执行完毕还有函数值返回。所以自定义函数像变量一样有自己的类型,它决定于函数返回值的数据类型。

(5) 引用算子

在 VB 中引用自定义类型的元素,或要引用和修改某个对象的一个属性值,或要给某个对象指定一种方法时,使用引用算子“.”。例如:

对象名.属性名 = 属性值

对象名.方法名 参数 1, 参数 2, ...

对某一对象触发的一个事件,在编写对象的事件过程时,使用引用算子“-”。例如:

Sub 对象名-事件名(参数 1, 参数 2, ...)

...程序语句

End Sub

1.3.1.6 基本数据类型和运算符

(1) 基本数据类型

VB 提供了以下几种基本数据类型:

整型(Integer)、长整型(Long)、单精度型(Single)、双精度型(Double)、货币型(Currency)、字符串型(String)、变体型(Variant)和自定义类型(Type)。

(2) 运算符

运算符:算术运算符、比较运算符、逻辑运算符、引用对象比较运算符(Is)和匹配比较运算符(Like)。

(3) 常量

常量是在程序过程中始终不变的常数值。编程中,要进行常量声明。格式为:Const 常量名=表达式。

(4) 变量

在程序中可以改变的量为变量。编程中,用声明语句创建一个变量,一般格式为:Dim 变量名 As 变量类型。

(5) 数组

数组是用来存放一组相互间有联系的类型完全相同的变量。数组的三个要素是:数组名、数组类型和数组下标(数组维数)。程序中创建数组,要用数组声明语句。声明数组的语句与声明变量的语句完全相同,只要将变量名用数组名代替即可:Dim 数组名(下标) As 类型名。

(6) 动态数组

一般的数组在创建时就规定了大小,但有时应用程序中需要的数组的大小并不知道,VB 提供了一种方法,可以为数组动态地分配内存空间。这样定义的数组可重复使用,还可以在程序运行中随时改变数组的大小(不能改变其类型和维数),这种数组称为动态数组。

(7) 控件数组

控件数组是用来存放窗体中一系列在结构上相关联,共用同一控件名的一组控件。创建控件数组不是通过声明语句,而是在 VB 可视环境下直接随控件一起创建。

1.3.1.7 项目

在 VB 环境下编程,编程人员首先要设计用户界面,然后编写事件过程、通用过程和自定义函数,还可能要用到自己或别人用其他开发工具设计的一些对象。这些作为设计结果都要保存起来,以供运行前的编译。设计好的用户界面(扩展名为 .frm 的文件)、程序模块(扩展名为 .bas 的文件)及新的对象类型(扩展名为 .vbx 的文件)都先以文件的形式保存起来,然后共同装入一个叫做项目(Project)的总文件中,以扩展名 .mak 形式的文件保存。由此可见,编写一个应用程序,就是建立一个项目。

项目用来保存一个应用程序所用到的所有编程资源。应用程序越大,它的优越性也越大。VB 的项目有如下几个特点:

- ① 当修改其中个别文件后再编译时,只对修改过的部分进行编译,节省了编译时间。
- ② 便于保存程序。
- ③ 项目中的一些文件在必要时可供其他应用程序使用,即可以加入其他项目中。

1.3.2 VB 的编程环境

1.3.2.1 安装并进入 VB 环境

在 WINDOWS 下运行 VB 软件第一张盘上的安装程序 SET-

UP.EXE, 按照提示选择有关选项。安装完成后, WINDOWS 程序管理器窗口中出现 VB 程序组。

启动 VB, 屏幕上出现图 1-2 所示窗体, 即进入了 VB 环境。

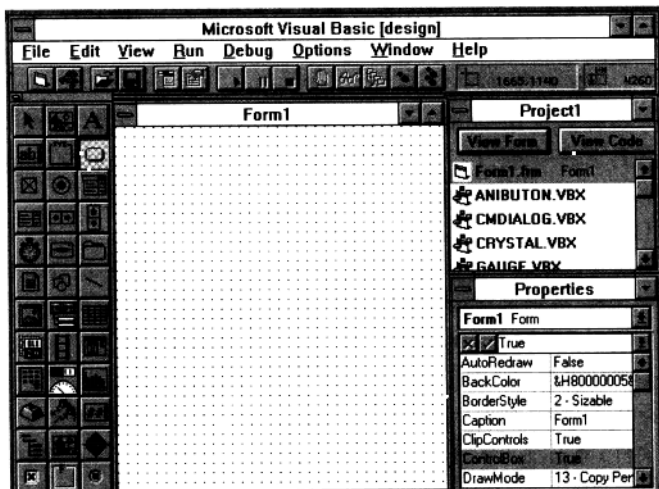


图 1-2

1.3.2.2 主窗口

VB 主窗口由标题、主菜单和工具栏组成。

(1) 标题

显示窗口标题, 标题后的括号中显示 VB 的运行状态, 分为设计阶段(Design)、执行阶段(Run)和中断阶段(Break)。

(2) 主菜单

主菜单中共有八个菜单项, 依次为: 文件(File)、编辑(Edit)、查看(View)、运行(Run)、调试(Debug)、选项(Options)、窗口(Window)和帮助(Help)菜单项。每个菜单项又有许多子菜单, 分