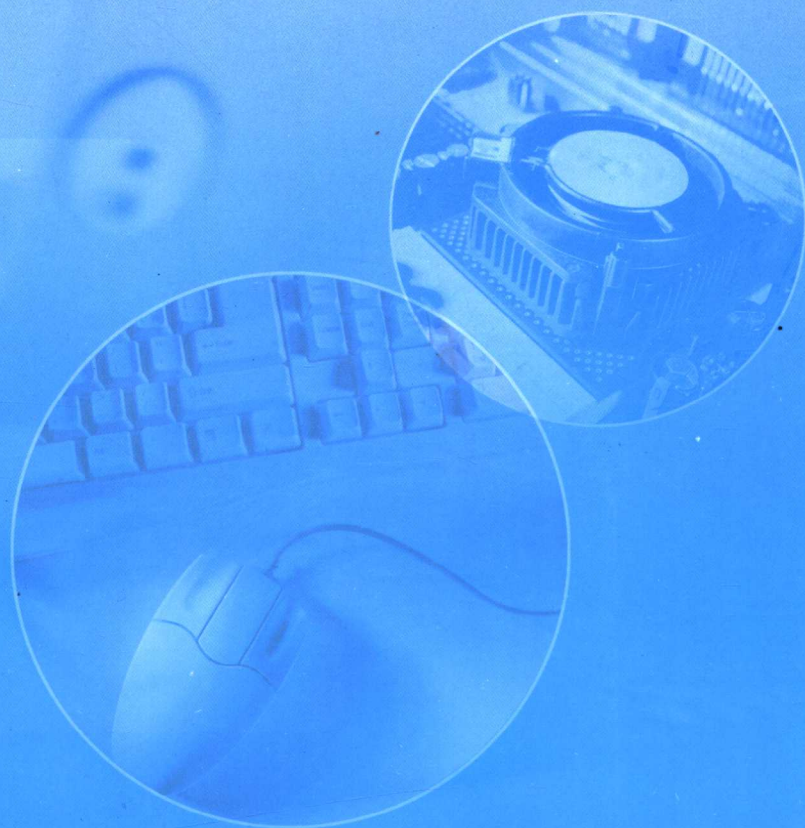


高等教育 **21**世纪 课程教材
计算机基础教育课程体系改革教材

Visual Basic 程序设计

上机指导与习题解析

吴文斗 周 兵 吴兴勇 编著



湖南教育出版社
www.hneph.com

高等教育 21 世纪 课程教材
计算机基础教育课程体系改革教材

Visual Basic 程序设计

上机指导与习题解析

吴文斗 周 兵 吴兴勇 编著

湖南教育出版社

内 容 简 介

本书是配合《Visual Basic程序设计教程》一书编写的参考书,由三部分组成。第一部分是重点难点解答;第二部分是上机实验指导,结合教材内容提供了16个实验,力求突出代表性、典型性和实用性,每个实验都作了详细的分析和提示,同时给出了详尽的解答和程序代码;第三部分为练习题和习题解析,以便是读者加深对所学知识的理解和掌握。

本书内容丰富、实用性强,是学习Visual Basic程序设计教程十分有用的一本参考书。本书既可作为高等院校Visual Basic课程的上机实践指导用书,也可以作为有一定Visual Basic语言基础知识的读者的自学指导书和全国计算机等级考试参考书。

图书在版编目(CIP)数据

Visual Basic程序设计上机指导与习题解析/吴文斗
著. —长沙:湖南教育出版社, 2005

I. V... II. 吴... III. BASIC 语言—程序设计—教
学参考资料 IV. TP312

中国版本图书馆CIP数据核字 (2005) 第043890号

Visual Basic程序设计上机指导与 习题解析

吴文斗 周 兵 吴兴勇 编著

责任编辑:龙育群

湖南教育出版社发行(长沙市韶山北路643号)

网 址: <http://www.hneph.com>

电子邮箱: csgaojiao@163.com

湖南华商文化商务有限公司印刷

787×1092 16开 印张: 13.25 字数: 320000

2005年5月第1版

2006年1月第2次印刷

ISBN7-5355-4520-3/G·4515

定价: 18.00元

本书若有印刷、装订错误,可向承印厂调换

前 言

本书是配合《Visual Basic 程序设计教程》一书编写的教学参考书。全书分为三大部分：第一部分提出每章的基本要求，同时对每章的重点和难点进行解答；第二部分为实验上机指导，共包含 16 个实验，每个实验都具有很强的针对性和实用性，为帮助读者尽快熟悉 Visual Basic 的操作环境、掌握 Visual Basic 的编程思想和编程方法，每个实验都提供了详细的分析和提示，同时给出了详尽的解答和程序代码；第三部分为补充练习题和习题解析，在该部分中，根据各章的知识点，设计了若干典型练习题，并给出了详细解析，以帮助读者迅速巩固每章所学知识点。这里，我们要求读者学习完一章后，能独立认真地完成这些精心设计的练习。

本书由吴文斗、周兵、吴兴勇、赵红波、丁超、侯发忠等编写，由吴文斗修改和统稿。在编写过程中，吴凤状、唐红丽等参加了材料组织和校对工作，在此表示感谢，同时还要感谢湖南教育出版社对本书出版给予的鼓励和支持。

由于时间紧迫，加之作者水平有限，缺漏与错误在所难免，恳请读者批评指正。

编 者

2005 年 4 月

目 录

第一部分 重点难点解答

第一章 Visual Basic 程序设计概述.....	1
第二章 Visual Basic 简单应用程序设计.....	2
第三章 Visual Basic 程序设计基础.....	4
第四章 顺序程序设计结构.....	7
第五章 选择程序设计结构.....	8
第六章 循环结构.....	10
第七章 数组.....	13
第八章 过程及函数.....	16
第九章 常用控件.....	20
第十章 多重窗体程序设计.....	23
第十一章 对话框及菜单程序设计.....	25
第十二章 键盘鼠标事件.....	27
第十三章 数据文件.....	28
第十四章 图形操作.....	32
第十五章 Visual Basic 数据库技术.....	34

第二部分 实验指导

实验一 Visual Basic 集成环境和简单程序设计.....	39
实验二 Visual Basic 简单应用程序设计.....	43
实验三 Visual Basic 程序设计基础.....	46
实验四 顺序结构程序设计.....	49
实验五 选择结构程序设计.....	52
实验六 循环结构程序设计.....	57
实验七 数组的使用.....	60
实验八 过程与函数.....	68
实验九 VB 的常用控件.....	73

实验十 菜单设计与通用对话框·····	79
实验十一 键盘及鼠标事件·····	84
实验十二 文件的基本操作·····	89
实验十三 图形操作·····	97
实验十四 Visual Basic 数据库技术·····	101
实验十五 安装文件制作·····	105

第三部分 基础知识测试

第一章 Visual Basic 程序开发环境·····	110
第二章 Visual Basic 对象及其操作·····	115
第三章 Visual Basic 程序设计基础·····	121
第四章 顺序结构程序设计·····	129
第五章 选择结构程序设计·····	133
第六章 循环结构程序设计·····	138
第七章 数组·····	148
第八章 过程与函数·····	158
第九章 Visual Basic 常用标准控件·····	168
第十章 多重窗体程序设计·····	176
第十一章 对话框及菜单程序设计·····	179
第十二章 键盘及鼠标事件·····	184
第十三章 数据文件·····	190
第十四章 图形操作·····	197
第十五章 Visual Basic 数据库技术·····	202
参考文献·····	206

第一部分 重点难点解答

第一章 Visual Basic 程序设计概述

一. 基本要求

1. 掌握 Visual Basic 的特点。
2. 理解可视化程序设计和事件驱动编程的概念。
3. 掌握 Visual Basic 的集成开发环境。
4. 掌握 Visual Basic 的三种工作模式。

二 重点难点解答

1. Visual Basic 的主要特点有哪些？

- (1) 具有面向对象的可视化设计工具 OOP
- (2) 事件驱动的编程机制
- (3) 提供了易学易用的应用程序集成开发环境
- (4) 结构化的程序设计语言
- (5) 支持多种数据库系统的访问
- (6) 支持动态数据交换(DDE)、动态链接库(DLL)和对象的链接与嵌入技术(OLE)
- (7) 联机帮助功能

其中最突出的两个特点是：可视化编程和事件驱动。

Visual Basic 的另一个显著特点是交互式，所谓交互式是指在编写代码过程中出现语法错误时系统会立即获得通知，并在开发过程中可运行程序进行调试。

2. 如何理解可视化编程和事件驱动的概念？

(1) 可视化编程

传统的编程方法使用的是面向过程、按顺序进行的机制，其缺点是程序员始终要关心什么时候发生什么事情，应用程序的界面都需要程序员编写语句来实现，对于图形界面的应用程序，只有在程序运行时才能看到效果，一旦不满意，还需要修改程序，因而使得开发工作非常烦琐。用 Visual Basic 进行应用程序开发主要有两部分工作：即设计界面和编写代码。在开发过程中所看到的界面，与程序运行时的界面基本相同，同时 Visual Basic 还向程序员提供了若干界面设计所需要的对象(称为控件)，在设计界面时，只需将所需要的控件放到窗口的指定位置即可，整个界面设计过程基本不需要编写代码。概括地说，可视化编程就是程序员在开发过程中能看到界面的实际效果。

(2) 事件驱动

用 Visual Basic 开发的应用程序，代码不是按照预定的路径执行，而是在响应不同的事件时执行不同的代码片段。事件可以由用户操作触发，如单击鼠标、键盘输入等事件。也可以由来自操作系统或其他应用程序的消息触发。这些事件的顺序决定了代码执行的顺

序。概括地说,事件驱动是指应用程序没有预定的执行路径,而是由程序运行过程中的事件决定。

3. Visual Basic 6.0 有哪三种工作模式?分别具有什么功能?

Visual Basic 6.0 的三种工作模式和功能是设计模式、运行模式和中断模式。

设计模式: 可以进行用户界面的设计和代码的编写。

运行模式: 运行应用程序,此时不可以编辑代码,也不可以编辑界面。

中断模式: 应用程序运行暂时中断,此时可以编写代码,但不可以编辑界面。

第二章 Visual Basic 简单应用程序设计

一. 基本要求

1. 理解对象、属性、事件和方法的概念及其关系。
2. 掌握 VB 编程的基本步骤,会设计简单的 VB 程序。
3. 掌握窗体、文本框、标签和命令按钮的主要属性、事件和方法。

二. 重点难点解答

1. 什么是对象、属性、事件和方法?

(1) 对象:从广义上来说,对象是客观事物的一个实体。在可视化编程语言中,对象分为两类,一类称为预定义对象,预定义对象由系统提供,可以直接使用或对其进行操作,如工具箱中的控件、窗体和菜单等,另一类称为自定义对象,由用户自行定义。

(2) 属性:属性是一个对象的特性,不同的对象有不同的属性。对象常见的属性有标题(Caption)、名称(Name)、颜色(Color)、字体(Font)、是否可见(Visible)等。通过修改对象的属性,可以改变对象的外观和功能。

属性可以在设计阶段通过属性窗口进行设置,也可以在运行阶段通过代码进行设置。其中在代码中设置的语法格式为:

对象名.属性名=属性值

例如:在窗体上加入一个文本框(TextBox)。名称为“Text1”,文本框内容是:“Good Morning”。则用下列语句进行设置:

Text1.Text=" Good Morning"

(3) 事件:所谓事件就是发生在对象上的事情。在 Visual Basic 中,系统为每个对象预先定义好了一系列的事件。例如单击(Click)事件、双击(Dbclick)事件、装入(Load)事件、改变(Change)事件、鼠标移动(MouseMove)事件、获取焦点(GotFocus)事件等。对象的事件是固定的,用户不能建立新的事件。当事件由用户触发(如单击)或由系统触发(如装入的),对象就会对该事件作出响应,响应某个事件后所执行的程序代码就是事件过程。

通常,选了一个对象之后,Visual Basic 系统会自动确定程序代码(事件过程)的开始和结束语句,程序代码的一般形式是:

Private 对象名称_事件名称()

事件过程

End sub

其中事件过程由若干程序代码组成，需要用户根据需要进行编写。

例如：命令按钮 Command1 的 Click(单击)事件过程为：

```
Private Sub Command1_Click()  
    Form1.BackColor=vbRed  
End Sub
```

过程中的代码 Form1.BackColor=vbRed 表示将窗体的背景颜色设置成红色。运行时，用鼠标单击命令按钮 Command1，就会执行该事件过程，将窗体的背景颜色设置成红色。

(4) 方法：在面向对象的程序设计中，引入了称为方法(Method)的特殊过程和函数供用户直接调用。如对象打印(Print)方法、显示窗体(Show)方法、清除(Cls)方法、移动(Move)方法等。用户可直接调用，不能进行修改。在代码中使用方法的格式为：

[对象名.]方法名 [参数列表]

例如：Form1.Print " 欢迎使用 Visual Basic "

该语句使用 .print 方法在名称为 Form1 的窗体上输出字符串 " 欢迎使用 Visual Basic "。

2. 如何理解对象、事件和方法三者之间的关系？

在 VB 编程中，完成某种功能可通过触发对象事件和调用对象方法，而触发事件和调用对象方法都必须基于某个对象。触发某个事件实际上就是执行一个事件过程代码段，在该代码段中又可以随意调用对象方法，而调用对象方法不可能触发某个事件。

例如：在窗体上添加一个命令按钮，编写一个响应 Command1 的单击(Click)事件过程如下：

```
Private Sub Command1_Click()  
    Command1.Caption=" 确定 "  
    Form2.Show  
End Sub
```

上述代码段表明单击对象按钮 Command1 触发单击事件，即执行上述代码。在执行上述事件过程中又调用了显示窗体(Form2)的方法。

3. 如何理解属性和方法的区别和联系？

对象的属性是描述对象的一组特征。如设置对象的名称、标题、颜色、大小等，有的可直接在设计界面时通过属性窗口进行设置，有的可以在事件过程或通用过程代码中进行设置。对象的方法是在对象上实施某种操作的功能，不能在设计界面时调用对象方法，一般在过程代码中可随意调用，根据需要还可以添加适当的参数。

4. 简述 Visual Basic 应用程序设计的一般步骤。

分析问题，确定程序要完成什么任务，按下面的步骤创建应用程序。

- (1) 新建工程。创建一个应用程序首先要打开一个新的工程。
- (2) 创建应用程序界面。
- (3) 设置属性值。
- (4) 对象事件过程的编程。
- (5) 保存文件。
- (6) 程序运行与调试。再次保存修改后的程序。

第三章 Visual Basic 程序设计基础

一. 基本要求

1. 掌握 Visual Basic 提供的标准数据类型。
2. 掌握变量、常量的概念及其数据类型。掌握变量的命名规则,理解变量声明的含义。
3. 掌握各种运算符与表达式的含义和应用,掌握运算符的优先级。
4. 掌握部分常用函数。

二. 重点难点解答

1. Visual Basic 提供了哪些标准数据类型?

数据是程序的必要组成部分,也是程序处理的对象。在各种程序设计语言中,数据类型的规定和处理方法是各不相同的。Visual Basic 不但提供了系统定义的数据类型,而且还允许用户定义自己的数据类型。

Visual Basic 6.0 定义了 11 种数据类型:它们是整型(Integer)、长整型(Long)、单精度(Single)、双精度(Double)、货币型(Currency)、字节型(Byte)、变长字符串(String)、定长字符串(String* N)、布尔型(Boolean)、日期型(Date)、对象型(Object)和可变类型(Variant)。

2. 如何理解变量和声明变量?

(1) 变量的概念

变量是指在程序运行过程中其值可以改变的量,是用于存储数据的内存单元。根据存储的数据不同,变量可以分为不同的数据类型。

(2) 变量的命名规则

- ①第一个字符必须是字母或汉字。
- ②长度不超过 255 个字符。
- ③不可以包含小数点或者内嵌的类型声明字符。
- ④不能使用关键字。

(3) 变量的声明

①显示声明

格式: **Dim|Private|Static|Public<变量名> [As <类型>]**

Dim: 在窗体模块、标准模块或过程中声明变量。

Private: 在窗体模块或过程中声明变量,使变量仅在该模块或过程中有效。

Static: 在过程中定义静态变量,即使该过程结束,也仍然保留变量的值。

Public: 在标准模块中声明全局变量,使变量在整个应用程序中有效。

如: **Dim n As Single** '声明变量 n 为单精度

②隐式声明

变量可以不经声明而直接使用,此时该变量为可变类型变量。在可变类型变量中,可以存放任何类型的数据,如数值、字符串、日期和时间。虽然这种方法很方便,但常会导致难以查找的错误,所以一般建议采用显示声明方式定义变量,即遵循“先定义,后使用”

③强制声明语句

3. 如何理解常量和声明常量?

直接常量是指在程序代码中，以直接明显的形式给出的数：

#11/10/2002# '日期型常量'

自定义常量是为方便用户开发而引入的符号常量。自定义常量使用关键字 **Const** 进行声明。

例如：

Const Pi = 3.14159 '声明符号常量 Pi 为 3.14159，单精度型。

4. Visual Basic 有哪些运算符及表达式?

(1) 算术运算符和算术表达式(见表 3-1)

表 3-1 算术运算符和算术表达式

顺序	符号	名称	用途	例子
1	^	指数	乘幂运算	x^y , $a^{(1/3)}$,
2	-	负号	取负数	-A, -234.
3	*	乘号	乘法运算	$(a1+a2)*(b1+b2)$
4	/	除号	除法运算	$(a1+a2)/(b1+b2)$
5	\	整除	整除运算	$(a1+a2)\backslash(b1+b2)$
6	Mod	取模	求余数	25.68 Mod 6.78 的结果: 5
7	+	加号	加法运算	$x+y$, $a1+a2$
8	-	减号	减法运算	$x-y$, $a1-a2$

(2) 字符串运算符(连接运算符)和字符串表达式

说明：“&”会自动将非字符串类型的数据转换成字符串后再进行连接，而“+”则不

能自动转换。例如：

" Hello " & " World " 结果为 " Hello World "

" abc " &123 结果为 " abc123 "

" abc " +123 错误

" 123 " +234=357

(3) 关系运算符和关系表达式

关系运算符用于对两个表达式的值进行比较，比较的结果为布尔值 True (真)或 False(假)，具体内容见表 3-2 所示。

表 3-2 关系运算符和表达式

运算符	运算	关系表达式例子	结果
=	等于	2=3	False
<>或><	不等于	2<>3	True
>	大于	2>3	False
<	小于	2<3	True
>=	大于等于	2>=3	False
<=	小于等于	2<=3	False

(4) 逻辑运算符和逻辑表达式

逻辑运算符两边的表达式要求为逻辑值。逻辑表达式的结果值仍为逻辑值，见表 3-3 所示。

表 3-3 逻辑运算符和逻辑表达式

优先级	运算符	运算	说明	例子	结果
1	Not	非	当表达式为假时，结果为真。	Not(3>8)	True
2	And	与	当两个表达式均为真时，结果才为真，否则为假。	(3>8)And(5<6)	False
3	Or	或	当两个表达式均为假时，结果才为假，否则为真。	(3>8)Or(5<6)	True
4	Xor	异或	当两个表达式同时为真或同时为假时，结果为假，否则为真。	(3<8)Xor(5<6)	False
				(3>8)Xor(5<6)	True
5	Eqv	等价	当两个表达式同时为真或同时为假时，结果为真，否则为假。	(3<8)Xor(5<6)	True
				(3>8)Xor(5<6)	False
6	Imp	蕴含	当第一个表达式为真，且第二个表达式为假时，结果为假，否则为真。	(3<8) Imp(5>6)	False

(5) 运算符的优先级

在 Visual Basic 中，运算符的优先级为：

数值运算符和字符串运算符→关系运算符→布尔运算符

5. Visual Basic 常用内部函数有哪些？

函数是一段完成特定功能的独立程序段。也是一种可以参加运算的量。一般说来，函数提供某一种特定的功能。Visual Basic 系统提供了约一百多个十余种类型的内部函数供

用户编写程序时调用，表 3-4 列出了常用的一些内部函数。

表 3-4 常用内部函数

函数分类	函数名
数学函数	Int, Fix, Sqr, Abs, Rnd, Sgn, Hex, Oct, Exp, Sin, Cos, Tan, Atn
字符串函数	Chr, Asc, String, Len, Left, Right, Mid, Lcase, Ucase, Ltrim, Rtrim, Trim, Str, Val, InStr, StrComp
日期时间函数	Now, Date, Time, Year, Month, Weekday, Day, Hour, Minute, Second
格式化函数	Format

第四章 顺序程序设计结构

一. 基本要求

1. 掌握赋值语句及其含义。
2. 掌握 Print 方法以及与 Print 方法有关的 Tab 函数和 Spc 函数。
3. 理解 InputBox 函数、MsgBox 函数和语句的使用方法。

二. 重点难点解答

1. 赋值语句

语法格式：

[Let] 变量名 = 表达式

[Let] 对象名.属性名 = 表达式

功能：将表达式的值赋值给变量名或指定对象的属性。一般用于给变量赋值或对控件设定属性值。

举例：
Form1.width = 300
Form1.caption = "输入框"
Temp = 20
Text1.Text = "欢迎使用 Visual Basic 6.0"

2. Print 方法及其使用

(1) Print 方法

Print 方法用于在窗体、图片框、打印机和立即窗口等对象上输出数据。

Print 方法的格式为：

[对象名.]Print [<表达式表>] [{, !; }]

对象可以是窗体、图片框、打印机等或立即窗口，如果省略“对象名”，则在当前窗体上输出。例如：

Picture1.Print " Visual Basic" ' 在图片框 Picture1 中显示： Visual Basic
Print " Visual Basic" ' 在当前窗体中显示： Visual Basic
Debug.Print " Visual Basic" ' 在立即窗口中显示： Visual Basic

表达式可以是算术表达式、字符串表达式、关系表达式或布尔表达式，多个表达式之

间的分隔符为逗号或分号。

(2) 与 Print 方法有关的函数

①Tab 函数

格式: Tab(n)

表示在指定的第 n 个位置上输出数据。

例如:

Print "Hello"; Tab(10); "World" ' 表示在第 10 列输出 World

②Spc 函数

格式: Spc (n)

功能: 产生 n 个空格。

例如:

Print "Hello"; Spc (4); "World"

输出结果: Hello World

3. InputBox 函数

InputBox 函数是提供从键盘输入数据的函数。利用它可产生一个对话框, 提示用户输入数据, 或单击命令按钮。该函数返回一个字符串。

语法格式:

变量名=InputBox(\$)(<提示信息>[, <标题>][, <缺省>][, <x 坐标>][, <y 坐标>])

说明: InputBox 返回的数据类型为字符串。

4. MsgBox 函数和 MsgBox 过程

①MsgBox 函数

语法格式:

变量[%] = MsgBox(提示信息[,按钮[+图标]+ [缺省按钮]+[模式]][, 标题])

说明: MsgBox 函数返回一个 1-7 之间的数值型数据。

②MsgBox 语句

语法格式:

MsgBox 提示[,按钮[+图标]+[缺省按钮]+[模式]][,标题]

第五章 选择程序设计结构

一. 基本要求

1. 掌握选择程序设计结构的基本特点。
2. 掌握 If 语句的三种结构。
3. 掌握 Select Case 语句结构。
4. 理解常用的几种嵌套结构。

二. 重点难点解答

1. Visual Basic 常用选择结构

计算机智能化的标志之一就是具有逻辑判断的能力，即针对不同的情况作出不同的选择。这一功能的实现是借助相应的选择结构语句来完成的。Visual Basic 中提供的实现选择结构的语句有：IF 语句和 Select Case 语句。

2. If 语句

(1) 单分支结构

语法格式：

If <表达式> Then

语句块

End If

或

If <表达式> Then <语句>

功能：当条件表达式的值为“真”时，则执行 Then 后面的语句或语句块；否则不执行任何操作。

(2) 双分支结构

语法格式 1：

If < 表达式 > Then

< 语句块 1>

Else

< 语句块 2>

End If

语法格式 2：

If < 表达式 > Then < 语句 1> Else < 语句 2>

功能：当表达式的值为非 0(True)时，执行 Then 后面的语句块 1，否则执行 Else 后面的语句块 2。

(3) 多分支结构

语法格式：

If <表达式 1> Then

<语句块 1>

ElseIf <表达式 2>Then

<语句块 2>

 ...

[Else

语句块 n+1]

End If

3. Select Case 语句

语法格式：

Select Case 变量或表达式

Case 表达式列表 1

语句块 1

Case 表达式列表 2

语句块 2

...

[Case Else

语句块 n+1]

End Select

功能：根据 Select Case <变量或表达式>中的结果与各 Case 子句中的值比较，决定执行哪一组语句块。如果有多个 Case 短语中的值与测试值匹配，则根据自上而下判断原则，只执行第一个与之匹配的语句块。

4. 选择结构的嵌套

在 IF 语句的 Then 分支和 Else 分支中可以完整地嵌套另一 IF 语句或 Select Case 语句，同样 Select Case 语句的每一个 Case 分支中都可嵌套另一 IF 语句或另一 Select Case 语句，这种结构称为嵌套结构。例如：

IF <条件 1> Then

.....

IF <条件 2> Then

.....

Else

.....

End If

....

Else

....

Select Case...

Case ...

.....

End Select

.....

End If

第六章 循环结构

一. 基本要求

1. 理解循环结构及其特点。
2. 掌握常用的几种循环结构。
3. 理解嵌套的构成及其执行过程。

二. 重点难点解答

1. 循环结构及其分类

循环结构是一种重复执行的程序结构,循环的本质是不断地重复某种动作。重复执行的语句(块)称为循环体。对计算机程序而言,循环必须具备两个重要因素:

- (1) 在一定的条件下,重复执行一组语句。
- (2) 必然出现不满足条件的情况,使循环终止。

Visual Basic 中提供了两种类型的循环语句:一种是计数循环语句;另一种是条件循环语句。

2. Do...Loop 循环结构

Do...Loop 循环一般用于事先不知道循环次数的循环结构,此种语句有两种形式:

(1) 当型循环

语法格式:

Do While(Until) 〈条件表达式〉

 〈循环体〉

[Exit Do]

 〈循环体〉

Loop

执行过程:先判断条件,后执行循环,循环体至少执行 0 次。

(2) 直到型循环

语法格式:

Do

 〈循环体〉

[Exit Do]

 〈循环体〉

Loop While(Until) 〈条件表达式〉

执行过程:先执行循环,后判断条件,循环体至少执行 1 次。

说明:

- ①条件表达式可以是一个逻辑表达式,也可以是一个关系表达式,其值应是逻辑型。
- ②Do 和 Loop 应成对出现。
- ③循环体中要有控制循环的语句,避免出现死循环。
- ④使用 While 条件构成的循环时,当条件为 True 时,重复执行循环体;当条件为 False 时,则退出循环体。
- ⑤使用 Until 条件构成的循环时,当条件为 False 时,重复执行循环体;当条件为 True 时,则退出循环体。
- ⑥Exit Do 表示当遇到该语句时,强制退出循环,执行 Loop 后的下一条语句,Exit Do 语句经常配合 If 语句使用。

例:用辗转相除法求两自然数 m , n 的最大公约数和最小公倍数。

分析:求最大公约数的算法思想: