

江苏地区专供



○ 计算机基础教学专家组 编

计算机程序设计强化复习

(Visual Basic 篇)

东南大学出版社
SOUTHEAST UNIVERSITY PRESS

最新大纲版

计算机等级考试强化复习系列用书

计算机程序设计强化复习 (Visual Basic 篇)

计算机基础教学专家组 编

东南大学出版社
· 南京 ·

内 容 简 介

本书是江苏省计算机等级考试强化复习系列用书之一。本书的使用对象是具有初步程序设计知识、准备参加江苏省计算机等级考试的在校大学生或对程序设计有兴趣的各类人员。本书的结构新颖,内容丰富,通篇划分为五个大的篇章来将初学编程者必须要掌握的知识点一一给出。有别于程序设计教程,本书从程序设计的角度出发,将重要的概念通过大量的例题加以深入细致地剖析。本书是欲参加江苏省计算机等级考试的学生不可多得的好助手。

图书在版编目(CIP)数据

计算机程序设计强化复习. Visual Basic 篇 / 计算机基础
教学专家组编. —南京:东南大学出版社,2006.9

ISBN 7-5641-0558-5

I. 计... II. 计... III. BASIC 语言—程序设计—水
平考试—自学参考资料 IV. TP311.1

中国版本图书馆 CIP 数据核字(2006)第 106057 号

计算机程序设计强化复习(Visual Basic 篇)

主 编	计算机基础教学专家组
责任编辑	张 煦
装帧设计	康 靖
出版发行	东南大学出版社
社 址	南京市四牌楼 2 号(210096)
经 销	江苏省新华书店
印 刷	南京京新印刷厂
版 次	2006 年 9 月第 1 版 2006 年 9 月第 1 次印刷
开 本	787mm×1092mm 1/16
印 张	14.75
字 数	368 千字
印 数	1—5000 册
定 价	22.00 元

凡因印装质量问题,请直接向东南大学出版社电子分社调换。电话:025—83793865

前 言

本书是针对江苏省计算机等级考试(VB)编写的复习用书。全书分为四篇,分别从概念、算法和上机实验环节对 VB 涉及的各知识点进行了总结和阐述。

第一篇“VB 语法知识基础”,从 VB 的程序框架入手,对最基本的概念进行了归纳和分析,帮助学生掌握、领会和吃透最基础的语法知识。

第二篇“VB 语法知识进阶”,对 VB 中比较复杂的、不易理解的过程、参数传递等概念进行了总结和阐述,结合第一篇的基础语法知识分析,帮助读者将零散和难懂的知识点组织串联成相互关联的记忆网络。

第三篇“常用算法及编程技巧”,通过对基本算法技巧的归纳,对具体应用问题的分类分析和讲解,帮助读者在众多的应用程序中发现和总结算法规律,掌握解题工具。

第四篇“界面元素与程序调试”,主要针对上机考试环节。向读者介绍常见错误及常用调试方法,从而引导读者找出改错题的解题方法。通过对编程模式的介绍,帮助读者在编程题环节获取更多的分数。

第五篇“模拟试题”,精选了三套模拟试题,并附以答案,旨在为读者营造逼真考试氛围,通过自测来巩固强化复习的效果。

本书是 Visual Basic 程序设计复习用书,读者为完成了“VB 程序设计”学习经历的学生,全书对概念和知识点主要进行概括、总结和比较,并通过例题分析来强化概念。在适当的章节还使用了图表来帮助读者记忆。

该书是经验丰富的一线教师和教学专家在多年教学实践总结的基础上撰写而成,相信它一定会给广大的读者一些有益的帮助。

限于作者的水平与时间的窘迫,书中难免有许多不当之处,敬请读者不吝赐教。

编 者

2006 年 8 月

目 录

第一篇 VB 语法知识基础	(1)
第 1 章 程序代码的组织方式	(3)
1.1 程序代码的基本构成	(3)
1.1.1 字符集	(3)
1.1.2 关键字	(3)
1.1.3 标识符	(3)
1.1.4 运算符	(4)
1.1.5 表达式	(4)
1.1.6 语句及语法规则	(4)
1.1.7 语义	(5)
1.2 VB 中的过程	(5)
1.2.1 用过程编程的两大好处	(5)
1.2.2 VB 中过程的种类	(5)
1.3 VB 中的模块	(6)
1.3.1 窗体模块	(6)
1.3.2 标准模块	(6)
1.3.3 类模块	(6)
第 2 章 数据类型、常量与变量	(7)
2.1 基本数据类型	(7)
2.1.1 数值型	(7)
2.1.2 字符串型	(8)
2.1.3 逻辑型	(8)
2.1.4 日期型	(8)
2.1.5 变体型	(8)
2.1.6 数据类型转换	(9)
2.2 构造数据类型	(9)
2.2.1 数组	(9)
2.2.2 自定义类型	(10)
2.3 常量与变量	(10)
2.3.1 字面常量	(10)
2.3.2 符号常量	(10)
2.3.3 变量	(11)
第 3 章 运算符、表达式与公共函数	(12)
3.1 VB 中的运算符与表达式	(12)
3.1.1 算术运算符与算术表达式	(12)

3.1.2	字符串连接符	(12)
3.1.3	关系运算符与关系表达式	(13)
3.1.4	逻辑运算符与逻辑表达式	(14)
3.1.5	运算符的优先级	(14)
3.2	VB 中的公共函数	(14)
3.2.1	数学函数	(14)
3.2.2	字符串函数	(15)
3.2.3	类型转换函数	(17)
3.2.4	随机函数	(19)
3.2.5	输入与输出函数	(20)
第 4 章	数组	(22)
4.1	数组的命名及说明	(22)
4.1.1	数组的命名	(22)
4.1.2	数组的说明	(22)
4.2	数组结构与数组元素	(22)
4.2.1	数组的逻辑结构与存储结构	(22)
4.2.2	数组元素	(23)
4.3	固定大小数组与动态数组	(23)
4.3.1	固定大小数组	(23)
4.3.2	动态数组	(23)
4.3.3	Erase 语句	(24)
第二篇	VB 语法知识进阶	(27)
第 5 章	顺序结构	(29)
5.1	赋值语句	(29)
5.1.1	赋值语句的格式	(29)
5.1.2	不同类型数据之间的赋值	(29)
5.1.3	赋值号与等号的区别	(31)
5.2	Print 方法、Cls 方法	(31)
5.2.1	Print 方法	(31)
5.2.2	Cls 方法	(32)
第 6 章	选择结构	(33)
6.1	IF...Then 结构语句	(33)
6.1.1	标准分支选择结构	(33)
6.1.2	单分支结构	(33)
6.1.3	多分支结构	(34)
6.2	Select Case 结构语句	(35)
6.2.1	一般形式	(35)
6.2.2	测试项的形式	(36)

第 7 章 循环结构	(37)
7.1 Do...Loop 结构语句	(37)
7.1.1 Do-While 循环结构	(37)
7.1.2 Do-Until 循环结构	(37)
7.2 For...Next 结构语句	(39)
7.2.1 一般形式	(39)
7.2.2 For 循环与 Do 循环的比较	(45)
7.2.3 循环嵌套	(45)
第 8 章 过程设计	(48)
8.1 Sub 过程与 Function 过程的定义与调用	(48)
8.1.1 Sub 过程的定义	(48)
8.1.2 Function 过程的定义	(50)
8.1.3 Function 过程与 Sub 过程的区别	(50)
8.1.4 Sub 过程与 Function 过程的调用	(51)
8.2 过程调用时的参数传递	(53)
8.2.1 参数传递的限制	(53)
8.2.2 按值传递	(54)
8.2.3 按地址传递	(55)
8.3 递归过程	(59)
8.3.1 递归过程的定义	(59)
8.3.2 递归过程的构成条件	(61)
8.4 变量的作用域与生命期	(62)
8.4.1 变量的作用域	(62)
8.4.2 变量的生命期	(63)
8.4.3 同名变量	(66)
第 9 章 文件操作	(68)
9.1 文件的基本概念	(68)
9.1.1 文件的概念	(68)
9.1.2 文件类型	(68)
9.1.3 文件操作步骤	(69)
9.2 基本文件操作控件	(69)
9.2.1 驱动器列表框(DriveListBox)	(69)
9.2.2 目录列表框(DirListBox)	(69)
9.2.3 文件列表框(FileListBox)	(70)
9.3 常用文件操作语句与文件操作函数	(71)
9.3.1 Open 语句	(71)
9.3.2 Close 与 Reset 语句	(72)
9.3.3 Lock 和 Unlock 语句	(73)
9.3.4 Seek 语句	(73)
9.3.5 Print# 和 Write# 语句	(73)

9.3.6	Input# 和 Line Input# 语句	(74)
9.3.7	Put# 和 Get# 语句	(74)
9.3.8	常用文件操作函数	(74)
第三篇 常用算法及编程技巧		(77)
第10章 算法中使用的基本技巧		(79)
10.1	交换、累加及累乘方法	(79)
10.1.1	交换	(79)
10.1.2	累加、累乘	(79)
10.2	最大、最小值问题	(80)
10.3	穷举与迭代	(81)
10.3.1	穷举	(81)
10.3.2	迭代	(82)
10.4	递推与递归方法	(83)
10.4.1	递推	(83)
10.4.2	递归	(84)
第11章 数列、级数求和及矩阵运算问题		(86)
11.1	数列问题	(86)
11.2	级数求和与连乘积问题	(87)
11.2.1	级数求和	(87)
11.2.2	连乘积	(89)
11.3	矩阵相关的计算	(90)
11.3.1	矩阵的生成和打印	(90)
11.3.2	矩阵的行列交换	(91)
11.3.3	矩阵的变换	(92)
11.3.4	寻找某特殊点	(94)
11.3.5	特殊矩阵	(95)
11.3.6	综合例题	(97)
第12章 数组的移动、排序及查找		(101)
12.1	移动	(101)
12.2	排序	(104)
12.2.1	选择法排序	(104)
12.2.2	冒泡法排序	(105)
12.2.3	插入法排序	(106)
12.2.4	合并排序	(107)
12.2.5	综合例题	(108)
12.3	查找	(110)
12.3.1	顺序查找	(110)
12.3.2	二分法查找	(111)
第13章 因数、最大公约数与素数等问题		(113)

13.1	整数的因数计算问题	(113)
13.1.1	问题的提出	(113)
13.1.2	如何保存求得的因数	(113)
13.1.3	进一步的讨论	(114)
13.2	最大公约数与最小公倍数问题	(114)
13.2.1	求最大公约数的一个简单方法	(114)
13.2.2	求最大公约数的欧几里得算法	(115)
13.2.3	求最大公约数的递归算法	(116)
13.2.4	求最小公倍数的一个简单方法	(117)
13.2.5	利用最大公约数求最小公倍数	(118)
13.3	素数问题	(119)
13.3.1	素数判断的基本方法	(119)
13.3.2	找出给定范围内的素数	(120)
13.3.3	素数问题的扩展	(121)
第 14 章	整数拆分、进制转换	(126)
14.1	整数拆分	(126)
14.1.1	转换成字符型处理	(126)
14.1.2	数值法处理	(126)
14.1.3	整数拆分的应用	(127)
14.2	进制转换	(129)
14.2.1	十进制转换为二进制	(129)
14.2.2	十进制转换为十六进制	(130)
14.2.3	二进制转换为十进制	(130)
14.2.4	十六进制转换为十进制	(131)
14.2.5	二进制转换为八进制	(132)
第 15 章	字符运算及圆圈数问题	(134)
15.1	字符运算	(134)
15.1.1	字符的替换	(134)
15.1.2	字符串的替换	(135)
15.1.3	逆序、回文串	(136)
15.1.4	统计	(137)
15.2	圆圈数问题	(139)
15.2.1	报数问题	(139)
15.2.2	圆圈中的特殊点	(140)
15.2.3	翻转问题	(141)
第四篇	界面元素与程序调试	(143)
第 16 章	常用控件的通用属性、事件及方法	(145)
16.1	常用控件的通用属性	(145)
16.1.1	控件的通用属性	(145)

16.1.2	属性值的设置方法	(146)
16.1.3	运行时属性	(146)
16.1.4	控件的默认属性和默认属性值	(146)
16.2	常用控件的通用事件	(146)
16.2.1	鼠标事件:Click 事件、DbClick 事件	(147)
16.2.2	键盘事件:KeyPress 事件	(147)
16.2.3	光标事件:LostFocus 事件	(148)
16.3	常用控件的通用方法	(148)
16.3.1	Move 方法	(148)
16.3.2	Refresh 方法	(149)
16.3.3	SetFocus 方法	(149)
第 17 章	窗体的常用属性、事件及方法	(152)
17.1	窗体的常用属性	(152)
17.1.1	ForeColor 属性	(152)
17.1.2	BackColor 属性	(152)
17.1.3	BorderStyle 属性	(152)
17.1.4	Icon 属性	(152)
17.1.5	Picture 属性	(152)
17.2	窗体的常用事件与方法	(153)
17.2.1	窗体的常用事件	(153)
17.2.2	窗体的常用方法	(154)
17.3	多窗体工程的设计	(154)
17.3.1	在工程中添加窗体	(155)
17.3.2	保存窗体	(155)
17.3.3	从工程中删除窗体	(155)
17.3.4	设置工程的启动窗体	(155)
17.3.5	窗体语句	(155)
17.3.6	不同窗体间数据的存取	(155)
第 18 章	重要控件	(158)
18.1	文本框、标签与命令按钮	(158)
18.1.1	文本框	(158)
18.1.2	标签框	(159)
18.1.3	命令按钮	(159)
18.2	单选按钮、复选框及框架	(159)
18.2.1	单选按钮	(159)
18.2.2	复选框	(160)
18.2.3	框架	(160)
18.3	列表框与组合框	(160)
18.3.1	列表框	(160)
18.3.2	组合框	(162)

18.4	图片框与图像控件	(162)
18.4.1	图片框	(162)
18.4.2	图像框	(164)
18.5	滚动条、定时器、菜单	(164)
18.5.1	滚动条	(164)
18.5.2	定时器控件	(164)
18.5.3	菜单	(165)
第 19 章	程序调试和程序设计	(169)
19.1	程序调试的基本方法	(169)
19.1.1	Visual Basic 的调试工具	(169)
19.1.2	Visual Basic 的工作模式	(170)
19.1.3	调试窗口	(171)
19.1.4	调试手段	(172)
19.2	程序中常见的错误和调试步骤	(173)
19.2.1	编辑错误	(173)
19.2.2	编译错误	(174)
19.2.3	运行时错误	(177)
19.2.4	逻辑错误	(180)
19.2.5	程序调试步骤	(180)
19.3	编程题的设计思路	(181)
19.3.1	题目分析	(181)
19.3.2	界面设置	(182)
19.3.3	程序设计	(182)
19.3.4	运行和调试程序	(185)
19.3.5	保存程序	(185)
19.4	实例	(185)
第五篇	模拟试题	(201)
	Visual Basic 模拟试题一	(203)
	Visual Basic 模拟试题二	(209)
	Visual Basic 模拟试题三	(215)
	参考文献	(224)

第一篇

VB 语法知识基础

本篇从最基本的 VB 字符开始,一直到 VB 的多窗体结构,将 VB 语法的最基本知识做个回顾。本篇的材料犹如砖块对于一座大厦的作用一样,是最基础的也是最重要的。通过对 VB 程序代码的组织方式的复习,进一步加深对 VB 代码框架的理解。通过对数据类型、常量及变量的复习,进一步理解 VB 中数据的表现形式,深刻领会无处不在的数据类型问题。任何一段程序都离不开“计算”,而“计算”是多种多样的,计算可以是解决数值问题也可以是解决非数值问题。有计算就有运算符、表达式以及函数。通过第 3 章的复习就是要熟练掌握这些对数据加工的工具。大量数据处理离不开数组这一数据结构,大部分程序都要使用数组。通过第 4 章的复习,就是要掌握数组定义及使用中的各个要点,为正确、灵活地使用数组打下坚实基础。

第1章 程序代码的组织方式

VB 的代码组织结构按照语言成分从小到大如图 1-1 所示：

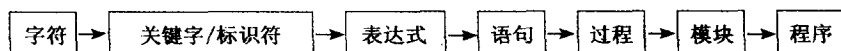


图 1-1 VB 的代码组织结构

1.1 程序代码的基本构成

1.1.1 字符集

犹如自然语言是人们交流的工具一样,程序设计语言是程序设计的工具。每种语言都有自己的符号集,程序设计语言的编译系统只接纳能识别的字符。字符集的要点是:

- 1) 字符是构成程序设计语言的最小单位;
- 2) 每一种程序设计语言都有一个自己的字符集;
- 3) VB 的基本字符集包括:

i. 数字(0~9)

ii. 英文字母(26 个,不区分大小写)

iii. 特殊字符(Space ! " # \$ % & ' () * + - , . / : ; < = > ? @ [\] ^ _ { | } ~)

- 4) 注意,全角或中文符号不能作为语言成分符号。

[提示] 对于程序中会出现的 π 、 α 等非 VB 字符,一般用英文字母拼出,如用 pi 来代替 π ,用 alpha 来代替 α 。

1.1.2 关键字

每一种程序设计语言都会使用一些特殊的字符串,这些字符串往往是常用的英文单词,从字面上就可理解它的一般意义。由于这些单词在语法上的特殊地位,所以称为关键字。关键字的要点是:

- 1) 关键字又称为保留字,它们在语法上有固定的意义,是语言的组成部分,往往表示系统提供的标准过程、运算符、常量等等;
- 2) VB 中约定关键字的首字母为大写字母,但输入时不区分大小写,系统能够识别并自动转换为标准形式。

[提示] 我们常常利用 VB 系统的这一特点在输入关键字时不区分大小写,让系统自动转换来检查输入的关键字的正确性。

1.1.3 标识符

标识符也是由若干字符组成的字符串,标识符的要点是:

- 1) 用来标记用户自己定义的常量、类型、变量、控件、过程或函数等的名字。
- 2) 在 VB 中,标识符的命名必须遵循如下规则:
 - i. 首字符必须是字母;
 - ii. 长度不超过 255 个字符,用于控件、窗体和模块的名字不能超过 40 个字符;
 - iii. 不能和关键字同名;
 - iv. 作为变量名的字符串内不得包括点号和用于类型说明的字符 %、&、!、#、@、\$。

[提示] 类似于关键字,我们在命名标识符时,尽量以大写字母开头且大小写混用,这样就可以利用 VB 系统的自动转换功能来检查输入的标识符的正确性。

1.1.4 运算符

与数学中的运算符不同的是,程序设计语言中的运算符必须考虑计算机的特点。程序设计语言中的运算符往往有特殊的意义,并体现出计算机运算的特点。每一种程序设计语言都有自己的运算符,运算符数量丰富与否反映了程序设计语言能力的大小。运算符的要点是:

- 1) 运算是数据的加工,最基本的运算形式用一些简洁的符号来描述,这些符号称为运算符;
- 2) 不同类型的数据可以参与不同的运算,换句话说,不同类型的数据使用不同的运算符操作。

[提示] VB 中一般要求运算符要与其操作对象(数据)类型相匹配。如果不匹配,系统就要对操作对象进行类型的自动转换,然而这种转换不是总能进行的,这是需要我们关注的。

1.1.5 表达式

程序中的各种运算是通过表达式实现的,表达式的要点是:

- 1) 为求得一个新的结果,由运算符和运算量构成的一个式子称为表达式;
- 2) 表达式通常由常量、变量、函数、运算符及括号等构成;
- 3) 因为每一个表达式有一个结果值,所以表达式也有类型,它表示了运算结果的类型。

[提示] 表达式的类型往往不是直观的,且表达式的结果常常要进一步被引用,如果对表达式的类型理解不正确,这种引用就会造成程序错误。

1.1.6 语句及语法规则

任何一种语言,不论是自然语言还是程序设计语言,都有自己的语法规则。程序设计者通过正确应用语法规则实现设计目标。和自然语言不一样的是,程序设计语言中的语法规则是及其严格的,不容有丝毫错误。语句及语法规则要点是:

- 1) 具有特定含义的符号串或句子称为语句。
- 2) VB 中的语句分为三类:
 - i. 说明语句(如:Dim 语句、Option 语句等),作用是向编译系统传达一些必要的信息;
 - ii. 执行语句(如:赋值语句、For 语句等),作用是完成一些特定的操作;
 - iii. 过程、方法调用语句(Call 语句、Form1、Print 等),作用是引用系统提供的或用户自定义的一些功能程序段。

3) 构成合法语句的规则称为语法。语法通常采用严格的形式化描述,也有采用语法图描述的。程序设计语言规定了一组描述语法的规则,称为语法规则。如:

- i. Print sum, ave 是符合语法规则的合法语句;
- ii. Print sum : ave 是不合法语句。

1.1.7 语义

合法语句的实际含义称为该语句的语义。语义通常采用自然语言描述。程序设计语言规定了每一个语句的语义。对语义的理解是否准确,直接影响程序的可靠性和正确性。

如以下三个 Print 方法调用语句对下一个输出项具有不同的影响

“Print a”、“Print a,”、“Print a;”

“Print a”的下一个输出项将另起一行;“Print a,”的下一个输出项将输出到同行的下一个输出区;而“Print a;”的下一个输出项将紧跟该输出项。

[提示] 阅读程序、编写程序时对语义的理解是至关重要的。

1.2 VB 中的过程

在设计一个规模较大、较复杂的程序时,需要按功能将程序分解成若干个相对独立的部分,即程序段。这些程序段称为程序的逻辑部件,它可以简化程序设计任务。这种逻辑部件称为过程。

1.2.1 用过程编程的两大好处

1) 过程可将程序划分成离散的逻辑单元,每个单元都比无过程的整个程序容易理解、容易修改、容易调试。

2) 一个程序中的过程,往往不必修改或只需稍作改动,便可以成为另一个程序的构件,这就是代码的重用性。

1.2.2 VB 中过程的种类

1) 系统提供的过程(事件过程)

当 Visual Basic 中的对象对一个事件的发生作出认定时,便自动用相应事件的名字调用该事件的过程。因为名字在对象和代码之间建立了联系,所以说事件过程是附加在窗体和控件上的。事件过程通常总是处于空闲状态,如果用户引发了事件或系统引发了事件,则相应的事件过程响应发生的事件。

事件过程的框架由系统提供,框架确定了该过程的访问权属性(Private 或 Public)、过程名、参数的个数及类型等,用户不能对它们进行修改,只能在此框架内填写代码。

2) 用户自定义过程(通用过程)

通用过程告诉应用程序如何完成一项指定的任务。一旦确定了通用过程,就必须由应用程序来调用。事件不能触发通用过程。

通用过程必须由用户自行定义,用户必须首先确定通用过程的框架,再进一步完成框架内的代码。

1.3 VB 中的模块

模块是 Visual Basic 用于将不同类型过程代码组织到一起而提供的一种结构。在 VB 中具有三种类型的模块,即窗体模块、标准模块和类模块。

简单的应用程序可以只有一个窗体,应用程序的所有代码都驻留在窗体模块中。而当应用程序庞大、复杂时,就要另加窗体。有时在几个窗体中都有要执行的公共代码,因为不希望两个窗体中有重复代码,所以要创建一个独立模块,它包含实现公共代码的过程。独立模块应为标准模块。

1.3.1 窗体模块

窗体模块(文件扩展名为 .FRM)是大多数 Visual Basic 应用程序的基础。窗体模块可以包含事件过程、通用过程以及变量、常数、类型和外部过程的窗体级声明。如果在文本编辑器(如 Windows 操作系统自带的写字板、记事本)中观察窗体模块,则还会看到窗体及其控件的描述,包括它们的属性设置值。写入窗体模块的代码是该窗体所属的具体应用程序专用的;它也可以引用该应用程序内的其他窗体或对象。

1.3.2 标准模块

标准模块(文件扩展名为 .BAS)是应用程序内其他模块访问的过程和声明的容器,我们一般把各个模块共享的东西放到标准模块中,这样做不仅提高了代码的利用率而且也给代码维护提供了方便。它们可以包含变量、常数、类型、外部过程和全局过程的全局(在整个应用程序范围内有效)声明或模块级声明。

1.3.3 类模块

在 Visual Basic 中,类模块(文件扩展名为 .CLS)是面向对象编程的基础。可在类模块中编写代码、建立新对象。这些新对象可以包含自定义的属性和方法。实际上,窗体正是这样一种类模块,在其上可安放控件、可显示窗体窗口。