

谭浩强 主编

全国计算机等级考试 名师名导

Visual Basic 程序设计应试辅导

(二级)

陈志泊 等编著

- ◎ 名师编著，紧扣最新大纲，精辟讲解
- ◎ 专家指导，令您事半功倍，轻松掌握
- ◎ 内容全面，教学自学培训，考生必备



清华大学出版社

内容商

谭浩强 主编

全国计算机等级考试 名师名导

Visual Basic 程序设计应试辅导

(二级)

陈志泊 等编著

清华大学出版社

内 容 简 介

本书是根据教育部考试中心颁布的“全国计算机等级考试二级考试最新大纲”中关于 Visual Basic 语言程序设计的考试要求而编写的。全书共分 16 章,包括:公共基础、Visual Basic 程序开发环境、Visual Basic 对象及其操作、可视化程序设计初步、运算符与表达式、数据输入与输出、选择程序结构、循环程序结构、数组、过程、多重窗体程序设计、菜单设计、对话框、键盘与鼠标事件、数据文件、上机考试辅导。

全书涵盖了考试大纲所要求的全部内容和知识点;每章都包括了考试知识点、要点和难点、典型例题及解析、自测训练题及其参考答案 5 部分,便于读者边学习边巩固所学内容;加深掌握程度。

本书可以作为参加计算机等级考试人员的自学、培训或辅导教材,也可以作为大专院校 Visual Basic 语言程序设计课程的教学用书和教学辅助参考书。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

Visual Basic 程序设计应试辅导(二级)/陈志泊等编著. —北京:清华大学出版社,2005.8

(全国计算机等级考试名师名导/谭浩强主编)

ISBN 7-302-11158-8

I. V… II. 陈… III. BASIC 语言-程序设计-水平考试-自学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 058943 号

出版者:清华大学出版社 地 址:北京清华大学学研大厦
<http://www.tup.com.cn> 邮 编:100084
社 总 机:010-62770175 客户服务:010-62776969

责任编辑:薛 阳

印 刷 者:北京市清华园胶印厂

装 订 者:三河市金元装订厂

发 行 者:新华书店总店北京发行所

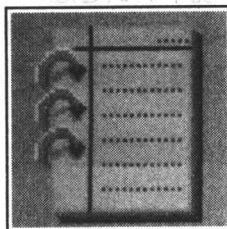
开 本:185×260 印张:26 字数:646 千字

版 次:2005 年 8 月第 1 版 2005 年 8 月第 1 次印刷

书 号:ISBN 7-302-11158-8/TP·7374

印 数:1~4000

定 价:35.00 元



前 言

Basic 语言自 1964 年问世以来,深受初学者欢迎,几十年来长盛不衰。微软公司从 1990 年起就推出了 Visual Basic 1.0 版来简化 Windows 应用程序的开发,可以说, Visual Basic 1.0 版的推出极大地改变了人们对 Microsoft Windows 的看法以及使用 Windows 的方式。Visual Basic 语言已经不是以前意义上的 Basic 语言了,它的功能已经变得十分强大,完全可以胜任任何大型应用程序的开发工作。

由于 Visual Basic 具有原 BASIC 语言简单、易学易用的优点,同时它又增加了结构化和可视化程序设计语言的功能,如:编程的可视化、引用了面向对象的程序设计方法、事件驱动的编程机制,并且具有快速方便地访问数据库等特点,所以深受广大计算机爱好者的欢迎。现在许多大学、中专、小学也相继开设了 Visual Basic 语言课程。教育部考试中心在 2001 年的全国计算机等级考试大纲中,把 Visual Basic 作为二级(程序设计)考试的可选语种之一。

为了帮助准备参加全国计算机等级考试的读者掌握 Visual Basic 语言程序设计,我们根据教育部考试中心颁布的《全国计算机等级考试二级考试最新大纲》关于 Visual Basic 语言程序设计的考试要求编写了本书。全书共分 16 章,包括:公共基础、Visual Basic 程序开发环境、Visual Basic 对象及其操作、可视化程序设计初步、运算符与表达式、数据输入与输出、选择程序结构、循环程序结构、数组、过程、多重窗体程序设计、菜单设计、对话框、键盘与鼠标事件、数据文件、上机考试辅导。

全书涵盖了考试大纲所要求的全部内容和知识点,每章都包括了考试知识点、要点和难点、典型例题及解析、自测训练题及其参考答案五部分,知识点部分简明地概括了考试大纲所要求的知识点,使内容更加精练,突出了重点,便于读者掌握;要点和难点部分概括和总结了读者在学习过程中易犯错误、较难理解的内容,突出了难点;典型例题及解析部分收集了历届等级考试试题及自编的部分例题,共包括各种题型的例题近 160 道,这些实例针对有关的知识点、重点和难点内容进行了详细分析,有助于读者加深对知识点的理解和掌握;习题部分包括了选择题约 330 道、填空题约 200 道,习题形式与考试真题相一致,内容紧扣知识点,并融合了程序设计中的常用基本算法,读者通过练习不仅可以巩固所学知识,而且对考试的规律和出题特点有准确的把握。

本书讲解深入浅出,例题和习题丰富,对知识点、考点力求举一反三,即使没有任何计算机基础的人,也能通过学习本书快速掌握 Visual Basic 程序设计的基本知识、方

法等。

书中所有的例题和习题中的程序都已在 Visual Basic 6.0 中文版环境中调试通过。

本书由谭浩强教授担任主编, 陈志泊副教授负责编写, 由谭浩强教授修改定稿。另外, 黄金燕、刘湛、张惠萍参加了全书的文字校对工作并提出了很多宝贵的修改建议, 在此表示衷心的感谢。

本书如有不当之处, 敬请专家和广大读者批评指正。

编 者

2004 年 10 月

目 录

第 1 章 二级公共基础	1
1.1 本章知识点	1
1.1.1 基本数据结构与算法	1
1.1.2 程序设计基础	16
1.1.3 软件工程基础	19
1.1.4 数据库设计基础	28
1.2 本章重点与难点	36
1.3 典型例题及解析	39
1.4 自测训练题	58
1.4.1 选择题	58
1.4.2 填空题	62
1.5 自测训练题参考答案	63
1.5.1 选择题	63
1.5.2 填空题	71
第 2 章 Visual Basic 程序开发环境	76
2.1 本章知识点	76
2.1.1 Visual Basic 的特点与版本	76
2.1.2 Visual Basic 集成开发环境的组成	77
2.2 本章重点与难点	81
2.3 典型例题及解析	82
2.4 自测训练题	84
2.4.1 选择题	84
2.4.2 填空题	86
2.5 自测训练题参考答案	86
2.5.1 选择题	86
2.5.2 填空题	87

第3章 Visual Basic 对象及其操作	88
3.1 本章知识点	88
3.1.1 对象及有关概念	88
3.1.2 窗体及窗体的常用属性与事件	90
3.1.3 控件及有关概念	91
3.1.4 命令按钮控件	92
3.1.5 文本类控件	93
3.1.6 图形类控件	94
3.1.7 选择类控件	97
3.1.8 滚动条控件	99
3.1.9 计时器控件	100
3.1.10 框架控件	100
3.1.11 文件系统控件	101
3.2 本章重点与难点	102
3.3 典型例题及解析	107
3.4 自测训练题	112
3.4.1 选择题	112
3.4.2 填空题	116
3.5 自测训练题参考答案	117
3.5.1 选择题	117
3.5.2 填空题	118
第4章 可视化程序设计初步	119
4.1 本章知识点	119
4.1.1 Visual Basic 下语句的书写格式	119
4.1.2 Visual Basic 下的几种简单语句介绍	119
4.1.3 事件驱动的编程机制与程序代码的编写	120
4.1.4 顺序程序结构	121
4.2 本章重点与难点	121
4.3 典型例题及解析	123
4.4 自测训练题	129
4.4.1 选择题	129
4.4.2 填空题	132
4.5 自测训练题参考答案	133
4.5.1 选择题	133
4.5.2 填空题	134

第 5 章 Visual Basic 数据类型、运算符和表达式	135
5.1 本章知识点	135
5.1.1 Visual Basic 应用程序代码的组织结构	135
5.1.2 基本数据类型	136
5.1.3 常量及其表示	137
5.1.4 变量的命名与定义	139
5.1.5 用户自定义数据类型	141
5.1.6 枚举数据类型	142
5.1.7 常用内部函数	143
5.1.8 运算符	143
5.1.9 表达式	144
5.2 本章重点与难点	145
5.3 典型例题及解析	147
5.4 自测训练题	157
5.4.1 选择题	157
5.4.2 填空题	159
5.5 自测训练题参考答案	161
5.5.1 选择题	161
5.5.2 填空题	161
第 6 章 数据输入/输出	163
6.1 本章知识点	163
6.1.1 Print 方法	163
6.1.2 与 print 方法有关的标准函数	164
6.1.3 与数据输出有关的属性和方法	164
6.1.4 InputBox 函数	165
6.1.5 MsgBox 函数	166
6.2 本章重点与难点	168
6.3 典型例题及解析	168
6.4 自测训练题	175
6.4.1 选择题	175
6.4.2 填空题	179
6.5 自测训练题参考答案	181
6.5.1 选择题	181
6.5.2 填空题	181

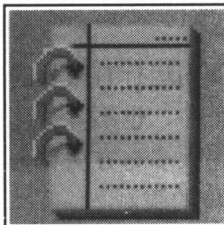
第 7 章 选择程序结构	182
7.1 本章知识点	182
7.1.1 单行结构条件语句	182
7.1.2 块结构条件语句	183
7.1.3 If 函数	183
7.1.4 多分支控制结构语句	184
7.2 本章重点与难点	185
7.3 典型例题及解析	187
7.4 自测训练题	193
7.4.1 选择题	193
7.4.2 填空题	196
7.5 自测训练题参考答案	199
7.5.1 选择题	199
7.5.2 填空题	200
第 8 章 循环程序结构	201
8.1 本章知识点	201
8.1.1 当循环结构	201
8.1.2 For 循环控制结构	202
8.1.3 Do 循环控制结构	203
8.2 本章重点与难点	204
8.3 典型例题及解析	207
8.4 自测训练题	216
8.4.1 选择题	216
8.4.2 填空题	224
8.5 自测训练题参考答案	230
8.4.1 选择题	230
8.4.2 填空题	231
第 9 章 数组	232
9.1 本章知识点	232
9.1.1 数组的概念	232
9.1.2 定义固定大小的数组	232
9.1.3 定义动态数组	234
9.1.4 数组的清除	234
9.1.5 数组的基本操作	235
9.1.6 For Each...Next 语句	236
9.1.7 控件数组	236

9.2	本章重点与难点	237
9.3	典型例题及解析	238
9.4	自测训练题	249
9.4.1	选择题	249
9.4.2	填空题	253
9.5	自测训练题参考答案	259
9.5.1	选择题	259
9.5.2	填空题	259
第 10 章	过程	260
10.1	本章知识点	260
10.1.1	Sub 过程	260
10.1.2	Function 过程	262
10.1.3	形式参数与实际参数	262
10.1.4	实际参数和形式参数的对应关系	263
10.1.5	按值传递与按地址传递	263
10.1.6	数组参数的传递	264
10.1.7	可选参数与可变参数	265
10.1.8	对象参数	265
10.1.9	Shell 函数	266
10.2	本章重点与难点	267
10.3	典型例题及解析	268
10.4	自测训练题	279
10.4.1	选择题	279
10.4.2	填空题	286
10.5	自测训练题参考答案	292
10.5.1	选择题	292
10.5.2	填空题	292
第 11 章	多重窗体程序设计	294
11.1	本章知识点	294
11.1.1	与多重窗体程序设计有关的语句和方法	294
11.1.2	多重窗体程序设计的方法和步骤	295
11.1.3	Sub main 过程	296
11.2	本章重点与难点	297
11.3	典型例题及解析	298
11.4	自测训练题	301
11.4.1	选择题	301

11.4.2	填空题	303
11.5	自测训练题参考答案	305
11.5.1	选择题	305
11.5.2	填空题	305
第 12 章	菜单设计	307
12.1	本章知识点	307
12.1.1	有关菜单的基本概念	307
12.1.2	菜单编辑器	307
12.1.3	设计下拉式菜单	310
12.1.4	设计弹出式菜单	310
12.2	本章重点与难点	311
12.3	典型例题及解析	312
12.4	自测训练题	314
12.4.1	选择题	314
12.4.2	填空题	316
12.5	自测训练题参考答案	318
12.5.1	选择题	318
12.5.2	填空题	318
第 13 章	对话框	319
13.1	本章知识点	319
13.1.1	对话框的特点及分类	319
13.1.2	自定义对话框	320
13.1.3	通用对话框	320
13.2	本章重点与难点	323
13.3	典型例题及解析	327
13.4	自测训练题	331
13.4.1	选择题	331
13.4.2	填空题	333
13.5	自测训练题参考答案	334
13.5.1	选择题	334
13.5.2	填空题	335
第 14 章	键盘与鼠标事件过程	336
14.1	本章知识点	336
14.1.1	KeyPress 事件	336
14.1.2	KeyDown 和 KeyUp 事件	337

14.1.3	鼠标事件与事件过程	338
14.1.4	鼠标光标的形状	339
14.1.5	对象拖放	340
14.1.6	与拖放有关的对象的属性、事件和方法	340
14.1.7	自动拖放与手动拖放	341
14.2	本章重点与难点	342
14.3	典型例题及解析	343
14.4	自测训练题	350
14.4.1	选择题	350
14.4.2	填空题	355
14.5	自测训练题参考答案	357
14.5.1	选择题	357
14.5.2	填空题	357
第 15 章	数据文件	358
15.1	本章知识点	358
15.1.1	文件的结构和分类	358
15.1.2	文件的基本操作步骤	359
15.1.3	文件的打开与关闭	360
15.1.4	文件操作的通用语句和函数	361
15.1.5	顺序文件的读写	362
15.1.6	随机文件的读写	364
15.1.7	文件的基本操作	364
15.2	本章重点与难点	365
15.3	典型例题及解析	367
15.4	自测训练题	374
15.4.1	选择题	374
15.4.2	填空题	377
15.5	自测训练题参考答案	380
15.5.1	选择题	380
15.5.2	填空题	380
第 16 章	上机考试辅导	381
16.1	本章知识点	381
16.1.1	上机考试内容	381
16.1.2	登录考试系统	381
16.1.3	答题	382
16.1.4	交卷	384

16.2 本章重点与难点.....	384
16.3 典型例题及解析.....	385
16.4 自测训练题.....	397
16.5 自测训练题参考答案.....	399
参考文献.....	400



第 1 章

二级公共基础

1.1 本章知识点

二级公共基础的内容主要包括数据结构与算法、程序设计基础、软件工程基础、数据库设计基础 4 个部分。

1.1.1 基本数据结构与算法

本部分的考试要点如下。

- (1) 算法的基本概念, 算法复杂度的概念和意义(时间复杂度与空间复杂度)。
- (2) 数据结构的定义, 数据的逻辑结构与存储结构, 数据结构的图形表示, 线性结构与非线性结构的概念。
- (3) 线性表的定义, 线性表的顺序存储结构及其插入与删除运算。
- (4) 栈和队列的定义, 栈和队列的顺序存储结构及其基本运算。
- (5) 线性单链表、双向链表与循环链表的结构及其基本运算。
- (6) 树的基本概念, 二叉树的定义及其存储结构, 二叉树的前序、中序和后序遍历。
- (7) 顺序查找与二分查找算法, 基本排序算法(交换类排序、选择类排序、插入类排序)。

1. 算法的基本概念

(1) 算法与数据结构的关系

程序设计主要包括两个方面, 一是行为特性的设计, 二是结构特性的设计。

行为特性的设计一般是指将解决问题过程中的每一个细节准确地加以定义, 并将全部的解题过程用某种工具完整地描述出来。这一过程也称为算法的设计。

结构特性的设计是指为问题的解决确定合适的数据结构。

数据结构与算法之间有着密切的关系。特别是对于数据处理问题, 算法的效率通常



与数据结构在计算机中的表示有着直接的关系。

(2) 算法的基本特征

所谓算法是指对解题方案准确而完整的描述。

对于一个问题,如果可以通过一个计算机程序,在有限的存储空间内运行有限长的时间而得到正确的结果,则称这个问题是算法可解的。但算法不等于程序,也不等于计算方法。当然,程序也可以作为算法的一种描述,但程序通常还需考虑很多与方法和分析无关的细节问题,这是因为在编写程序时要受到计算机系统运行环境的限制。

算法实际上是一种抽象的解题方法,它具有动态性。作为一个算法,一般应具有以下几个基本特征。

① 能行性 (Effectiveness)

算法的能行性主要包括两个方面。一是算法中的每一个步骤必须是能实现的,二是算法执行的结果要能达到预期的目的。

② 确定性 (Definiteness)

算法的确定性,是指算法中的每一个步骤都必须是有明确定义的,不允许有模棱两可的解释,也不允许有多义性。这一性质也反映了算法与数学公式的明显差别。

③ 有穷性 (Finiteness)

算法的有穷性是指算法必须能在有限的时间内做完,即算法必须能在执行有限个步骤之后终止。

算法的有穷性还应包括合理的执行时间的含义。因为,如果一个算法需要执行千万年,也就失去了实用价值。

④ 拥有足够的情报

一个算法是否有效,还取决于为算法所提供的情报是否足够。通常,算法中的各种运算总是要施加到各个运算对象上,而这些运算对象又可能具有某种初始状态,这是算法执行的起点或是依据。因此,一个算法执行的结果总是与输入的初始数据有关,不同的输入将会有不同的结果输出。当输入不够或输入错误时,算法本身也就无法执行或导致执行有错。一般来说,当算法拥有足够的情报时,此算法才是有效的,而当提供的情报不够时,算法并不有效。

综上所述,所谓算法,是一组严谨地定义运算顺序的规则,并且每一个规则都是有效的,且是明确的,此顺序将在有限的次数下终止。

(3) 算法的基本要素

一个算法通常由两个基本要素组成:一是对数据对象的运算和操作,二是算法的控制结构。

① 算法中对数据的运算和操作

每个算法实际上是按解题要求从环境能进行的所有操作中选择合适的操作所组成的一组指令序列。因此,计算机算法就是计算机能处理的操作所组成的指令序列。

通常计算机可以执行的基本操作是以指令的形式描述的。一个计算机系统能执行的所有指令的集合,称为该计算机系统的指令系统。计算机程序就是按解题要求从计算机指令系统中选择合适的指令所组成的指令序列。



② 算法的控制结构

一个算法的功能不仅取决于所选用的操作,而且还与各操作之间的执行顺序有关。算法中各操作之间的执行顺序称为算法的控制结构。

算法的控制结构给出了算法的基本框架,它不仅决定了算法中各操作的执行顺序,而且也直接反映了算法的设计是否符合结构化原则。描述算法的工具通常有传统流程图、N-S 结构化流程图、算法描述语言等。一个算法一般都可以用顺序、选择、循环 3 种基本控制结构组合而成。

(4) 算法设计的基本方法

计算机解题的过程实际上是在实施某种算法,这种算法称为计算机算法。

常用的算法设计方法有如下几种。

① 列举法

列举法的基本思想是根据提出的问题,列举所有可能的情况,并用问题中给定的条件检验哪些是需要的,哪些是不需要的。因此,列举法常用于解决“是否存在”或“有多少种可能”等类型的问题,例如求解不定方程的问题。

列举法的特点是算法比较简单,但当列举的可能情况较多时,执行列举算法的工作量将会很大。通常,在设计列举算法时,要对实际问题进行详细的分析,将与问题有关的知识条理化、完备化、系统化,从中找出规律;或对所有可能的情况进行分类,引出一些有用的信息,就可以大大减少列举量。

列举算法是计算机算法中的一个基础算法。

② 归纳法

归纳法的基本思想是:通过列举少量的特殊情况,经过分析,最后找出一般的关系。显然,归纳法要比列举法更能反映问题的本质,并且可以解决列举量为无限的问题。但是,从一个实际问题中总结归纳出一般的关系,并不是一件容易的事情,尤其是要归纳出一个数学模型更为困难。从本质上讲,归纳就是通过观察一些简单而特殊的情况,最后总结出有用的结论或解决问题的有效途径。

③ 递推

所谓递推是指从已知的初始条件出发,逐次推出所要求的各中间结果和最后结果。其中初始条件或是问题本身已经给定,或是通过对问题的分析与化简而得到确定。递推本质上也属于归纳法,工程上许多递推关系式实际上是通过在实际问题的分析与归纳而得到的,因此,递推关系式往往是归纳的结果。

④ 递归

递归的基本思想是将一个复杂的问题归结为若干个较简单的问题,然后将这些较简单的每一个问题再归结为更简单的问题,这个过程可以一直做下去,直到最简单的问题为止。递归是一种很重要的算法设计方法。

递归分为直接递归与间接递归两种。

⑤ 减半递推技术

解决实际问题的复杂程度往往与问题的规模有着密切的关系。因此,利用分治法解决这类实际问题是有效的。所谓分治法就是对问题分而治之。工程上常用的分治法是减



半递推技术。

所谓“减半”是指将问题的规模减半,而问题的性质不变。所谓“递推”是指重复“减半”的过程。

⑥ 回溯法

在工程上,有些实际问题却很难归纳出一组简单的递推公式或直观的求解步骤,并且也不能进行无限的列举。对于这类问题,一种有效的方法是“试”。通过对问题的分析,找出一个解决问题的线索,然后沿着这个线索逐步试探,对于每一步的试探,若试探成功,就得到问题的解,若试探失败,就逐步回退,换别的路线再进行试探。这种方法称为回溯法。回溯法在处理复杂数据结构方面有着广泛的应用。

(5) 算法复杂度

算法的复杂度主要包括时间复杂度和空间复杂度。

① 算法的时间复杂度

所谓算法的时间复杂度,是指执行算法所需要的计算工作量。

算法的工作量用算法所执行的基本运算次数来度量,而算法所执行的基本运算次数是问题规模的函数,即

$$\text{算法的工作量} = f(n)$$

其中 n 是问题的规模。

在同一问题规模下,如果算法执行所需的基本运算次数取决于某一特定输入时,可以用以下两种方法来分析算法的工作量。

平均性态 (Average Behavior)

所谓平均性态分析是指用各种特定输入下的基本运算次数的带权平均值来度量算法的工作量。算法的平均性态定义为

$$A(n) = \sum_{x \in D_n} p(x)t(x)$$

其中, x 是所有可能输入中的某个特定输入, $p(x)$ 是 x 出现的概率(即输入为 x 的概率), $t(x)$ 是算法在输入为 x 时所执行的基本运算次数, D_n 表示当规模为 n 时,算法执行时所有可能输入的集合。

最坏情况复杂性 (Worst-Case Complexity)

所谓最坏情况复杂性,是指在规模为 n 时,算法所执行的基本运算的最大次数。它定义为

$$W(n) = \max_{x \in D_n} \{t(x)\}$$

显然, $W(n)$ 的计算要比 $A(n)$ 的计算方便得多。由于 $W(n)$ 实际上是给出了算法工作量的一个上界,因此,它比 $A(n)$ 更具有实用价值。

② 算法的空间复杂度

一个算法的空间复杂度,一般是指执行这个算法所需要的内存空间。

一个算法所占用的存储空间包括算法程序所占的空间、输入的初始数据所占的存储空间以及算法执行过程中所需要的额外空间。其中额外空间包括算法程序执行过程中的工作单元以及某种数据结构所需要的附加存储空间。如果额外空间量相对于问题规模来