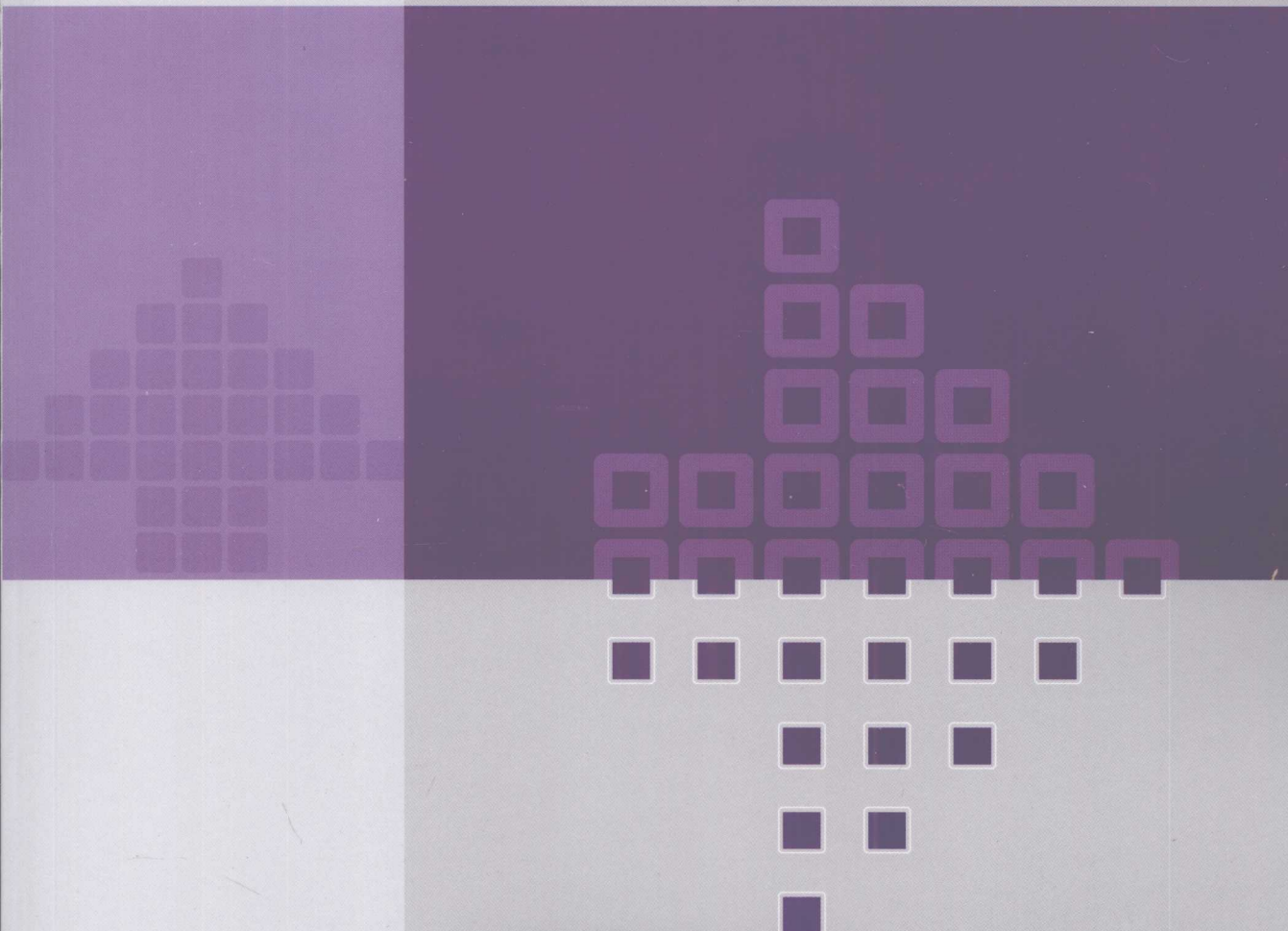


● ● ● ● 21 世纪高等院校计算机规划教材

高级语言程序设计基础

杨 明 主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

21世纪高等院校计算机专业系列教材

高级语言程序设计基础

第二版

陈 明 主编



21 世纪高等院校计算机规划教材

高级语言程序设计基础

主 编 杨 明

副主编 孙冀侠 王岚岚 崔再惠 李 季

参 编 王洪艳 崔旭东 许 岩

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书以 Visual Basic 6.0 为背景,在介绍面向对象可视化界面程序设计的同时,也介绍了结构化程序设计的语句、常用算法和高级语言编程思想,并通过大量丰富的实例给予说明。

本书共分 9 章,主要内容包括: Visual Basic 概述、Visual Basic 简单的程序设计、Visual Basic 程序设计基础、Visual Basic 控制结构、数组、Visual Basic 常用标准控件、过程、可视界面程序设计和文件。

本书内容丰富、循序渐进,并配有配套的实验教材、电子教案和网络版考试系统,适合作为高等学校各专业教材,也可作为高职高专各专业教材或教学参考书。

图书在版编目(CIP)数据

高级语言程序设计基础/杨明主编. —北京:中国铁道出版社, 2008. 12

(21 世纪高等院校计算机规划教材)

ISBN 978-7-113-08859-0

I. 高… II. 杨… III. BASIC 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 209463 号

书 名: 高级语言程序设计基础
作 者: 杨 明 主编

策划编辑: 严晓舟 刘 亮

责任编辑: 李小军

编辑助理: 吴媛媛

责任印制: 李 佳

编辑部电话: (010) 63583215

封面设计: 付 巍

封面制作: 白 雪

出版发行: 中国铁道出版社(北京市宣武区右安门西街 8 号 邮政编码: 100054)

印 刷: 三河市华业印装厂

版 次: 2008 年 12 月第 1 版 2008 年 12 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 17 字数: 397 千

书 号: ISBN 978-7-113-08859-0/TP·2870

定 价: 30.00 元

版权所有 侵权必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

前言

计算机技术的飞速发展,促进了计算机基础教育的发展。教育部非常重视高等学校非计算机专业的计算机基础教学,先是提出了“计算机文化基础”、“计算机技术基础”和“计算机应用基础”三个层次的课程体系,而后又根据形势的发展提出了“模块化”的教学体系。根据国家教育部的指示精神,对于非计算机专业的学生来说,学习高级语言程序设计,掌握一门面向对象程序设计语言在新形势下非常必要。

Visual Basic 提供了开发 Windows 应用程序最迅速、最简洁的方法。它提供了一整套工具,供用户开发应用程序。Visual Basic 是国内外最流行的程序设计语言之一。目前,许多高校都开设了“Visual Basic 程序设计”课程,但通过我们多年的教学实践经验来看,真正适合高校非计算机专业教学的“高级语言程序设计”课程的教材并不多。2000 年鞍山师范学院就将“高级语言程序设计”列为学院的重点课程,2003 年该课被评为学院精品课程。为了配合精品课程建设,我们参照全国高校非计算机专业计算机基础教育改革方案,以 Visual Basic 程序设计为主编写了此教材,并配套了实验教材。同时,我们创作了与之配套的电子教案和开发了本课程的“无纸化”考试系统,读者可发送电子邮件到 ym@mail.asnc.edu.cn 索取。本书的作者都是多年从事高级语言程序设计教学的教师,具有丰富的教学经验和实践经验,在编写本教材时力求做到理论和实践相结合,强调教学实践环节和学生应用能力的培养。

本书共分为 9 章,主要内容包括 Visual Basic 概述、Visual Basic 简单的程序设计、Visual Basic 程序设计基础、Visual Basic 控制结构、数组、Visual Basic 常用标准控件、过程、可视界面程序设计和文件,每章均配有习题。为了兼顾文、理、艺术、体育学生存在的差异,本教材的内容介绍尽可能详细明了、深入浅出,各章内容相对独立,可以根据实际教学情况进行取舍。

本书的第 1 章和第 8 章 8.1 节由杨明编写,第 2 章由崔旭东编写,第 3 章由孙冀侠编写,第 4 章由李季编写,第 5 章由王岚岚编写,第 6 章由王洪艳编写,第 7 章由崔再惠编写,第 8 章 8.2 节及以后内容和第 9 章由许岩编写。全书由杨明统稿定稿。

在本书的编写过程中,得到了许多教师的大力支持和帮助,在此一并表示感谢。

由于编者水平有限,时间仓促,书中难免有不足之处,恳请读者批评指正!

编 者

2008 年 10 月

目 录

第 1 章 Visual Basic 概述	1
1.1 程序设计语言与程序设计	1
1.1.1 程序设计语言的发展	1
1.1.2 程序设计语言的分类	3
1.1.3 常见的程序设计语言	3
1.1.4 程序设计语言的编译	5
1.2 关于 Visual Basic	5
1.2.1 Visual Basic 的发展	5
1.2.2 Visual Basic 的特点	6
1.3 安装和运行 Visual Basic 6.0	8
1.3.1 Visual Basic 6.0 运行环境	8
1.3.2 安装 Visual Basic 6.0	8
1.3.3 启动 Visual Basic 6.0	9
1.4 Visual Basic 集成开发环境	9
1.4.1 标题栏	10
1.4.2 菜单栏	10
1.4.3 标准工具栏	11
1.4.4 控件箱	12
1.4.5 窗口	13
1.5 Visual Basic 6.0 帮助系统的使用	16
1.5.1 使用 MSDN Library 查阅器	16
1.5.2 上下文帮助	17
1.5.3 运行所提供的样例	17
习题	18
第 2 章 Visual Basic 简单的程序设计	19
2.1 Visual Basic 中对象的概念	19
2.1.1 Visual Basic 的对象	19
2.1.2 对象属性设置	21
2.2 建立一个简单的应用程序	23
2.2.1 用 Visual Basic 开发应用程序的一般步骤	23
2.2.2 编写 Visual Basic 应用程序	24
2.2.3 “代码”窗口	29
2.3 窗体及三个常用控件的使用	30
2.3.1 窗体	31
2.3.2 控件	36

2.4	工程管理	46
2.4.1	程序的保存、加载和运行	46
2.4.2	Visual Basic 应用程序的结构与工作方式	50
	习题	51
第 3 章	Visual Basic 程序设计基础	53
3.1	Visual Basic 语言字符集及编码规则	53
3.1.1	Visual Basic 的字符集	53
3.1.2	源程序的书写规则与约定	54
3.2	数据类型	55
3.2.1	字符串类型	55
3.2.2	数值类型	55
3.2.3	其他类型	56
3.3	常量和变量	57
3.3.1	常量	57
3.3.2	变量	59
3.4	运算符与表达式	61
3.4.1	算术运算符与算术表达式	61
3.4.2	字符串运算符与字符串表达式	63
3.4.3	关系运算符与关系表达式	63
3.4.4	逻辑运算符与逻辑表达式	64
3.4.5	日期型表达式	65
3.4.6	各种运算符的执行顺序	66
3.5	常用内部函数	66
3.5.1	数学函数	66
3.5.2	字符串函数	68
3.5.3	转换函数	69
3.5.4	日期、时间函数	71
3.5.5	其他函数	71
3.5.6	立即窗口	75
3.6	数据输入——InputBox() 函数	75
3.7	MsgBox() 函数和 MsgBox 语句	78
3.7.1	MsgBox() 函数	78
3.7.2	MsgBox 语句	80
	习 题	80
第 4 章	Visual Basic 控制结构	84
4.1	顺序结构	84
4.2	选择结构	84
4.2.1	单行结构条件语句	84
4.2.2	块结构条件语句	85
4.2.3	Select Case 语句	88

4.2.4	IIf()函数	91
4.3	循环结构	91
4.3.1	For 循环控制结构	91
4.3.2	Do 循环控制结构	95
4.3.3	当循环控制结构	99
4.3.4	多重循环	102
4.4	其他辅助控制语句	104
4.4.1	GoTo 语句	104
4.4.2	结束语句 End	105
4.5	与控制结构相关的算法实例	105
4.5.1	算法知识	105
4.5.2	与控制结构相关的算法实例	105
	习题	109
第 5 章	数组	115
5.1	数组的概念	115
5.2	静态数组	115
5.2.1	静态数组的定义	116
5.2.2	默认数组	120
5.3	动态数组	121
5.4	数组的基本操作	122
5.4.1	数组的引用	122
5.4.2	数组元素的输入	123
5.4.3	数组元素的输出	124
5.4.4	数组元素的复制	125
5.4.5	数组的初始化	125
5.5	数组的应用举例	127
5.5.1	一维数组的应用	127
5.5.2	二维数组的应用	132
5.5.3	动态数组的应用	135
5.6	控件数组	135
5.6.1	控件数组的概念	136
5.6.2	创建控件数组	136
	习题	139
第 6 章	Visual Basic 常用标准控件	144
6.1	单选按钮和复选框	144
6.1.1	单选按钮	145
6.1.2	复选框	146
6.2	框架	149
6.3	滚动条	150
6.4	列表框、组合框	152

6.4.1	列表框	152
6.4.2	组合框	158
6.5	计时器	160
6.6	图形控件	161
6.7	焦点与 Tab 顺序	164
6.7.1	设置焦点	164
6.7.2	Tab 顺序	165
6.8	多重窗体	165
6.8.1	建立多重窗体应用程序	165
6.8.2	多重窗体程序的执行与保存	169
6.9	鼠标与键盘事件过程	170
6.9.1	鼠标事件	170
6.9.2	键盘事件	171
习题		173
第 7 章	过程	176
7.1	函数过程的定义和调用	177
7.1.1	函数过程的定义	178
7.1.2	函数过程的调用	180
7.2	子过程的定义与调用	181
7.2.1	子过程的定义	182
7.2.2	子过程的调用	183
7.3	参数传递	185
7.4	过程、变量的作用域	191
7.4.1	过程的作用域	193
7.4.2	变量的作用域	199
7.4.3	静态变量	203
7.5	递归	205
7.6	常用算法	207
习题		212
第 8 章	可视界面程序设计	213
8.1	通用对话框	213
8.1.1	自定义对话框	214
8.1.2	通用对话框控件	216
8.1.3	文件对话框	217
8.1.4	颜色对话框	219
8.1.5	字体对话框	220
8.1.6	打印对话框	221
8.1.7	帮助对话框	222
8.2	菜单设计编辑器	222
8.2.1	菜单概述	223

8.2.2	菜单编辑器	224
8.2.3	利用菜单编辑器建立菜单	226
8.2.4	动态菜单	228
8.2.5	弹出式菜单	229
8.3	多文档窗体	229
8.3.1	多文档界面的结构	230
8.3.2	菜单设计	231
8.3.3	多窗口管理	233
8.4	工具栏	234
8.4.1	关于 ActiveX 控件	235
8.4.2	创建工具栏	236
8.5	状态栏	238
	习题	239
第 9 章	文件	242
9.1	驱动器、目录和文件列表框	242
9.1.1	驱动器列表框控件	243
9.1.2	目录列表框控件	243
9.1.3	文件列表框控件	244
9.1.4	三个控件的联动	244
9.2	文件及其结构	246
9.2.1	文件系统的基本概念	247
9.2.2	文件操作语句和函数	248
9.3	顺序文件	249
9.4	随机文件	253
9.5	二进制文件	256
9.6	文件的基本操作	256
	习题	258
	参考文献	261



第 1 章

Visual Basic 概述

在过去的几十年间，大量的程序设计语言被发明、被取代、被修改或组合在一起。尽管人们多次试图创造一种通用的程序设计语言，却没有一次尝试是成功的。直到高级程序设计语言（也称高级语言）的出现才使得计算机程序设计语言不再过度地依赖某种特定的机器或环境，这是因为高级语言在不同的平台上会被编译成不同的机器语言，而不是直接被机器执行。本章以 Visual Basic 为例介绍高级语言的相关知识。

1.1 程序设计语言与程序设计

“语言”是人与人之间用来表达意思、交流思想的工具，是由语音、词汇和语法构成的一定系统。“程序设计语言”（也称算法语言）是人与计算机交流的工具，是由字、词和语法规则构成的指令系统。它是一种被标准化的交流技巧，用来向计算机发出指令。一种计算机语言让程序员能够准确地定义计算机所需要使用的数据，并精确地定义在不同情况下所应当采取的行动。

程序设计就是利用计算机语言完成某项工作的过程。过程叫做设计，成果叫做程序。程序的思想就是用软件驱动硬件的思想。一般来说，程序设计分 3 步：

- ① 系统分析。
- ② 编写源程序。
- ③ 调试运行。系统分析和调试运行是程序设计的重中之重。

一台计算机是由硬件系统和软件系统两大部分构成的，硬件是物质基础，而软件可以说是计算机的灵魂。没有软件，计算机是一台“裸机”，是什么也不能干的；有了软件，才能灵动起来，成为一台真正的“电脑”。所有的软件，都是用计算机语言编写的。

1.1.1 程序设计语言的发展

计算机程序设计语言的发展，经历了从机器语言、汇编语言到高级语言的历程。

1. 机器语言

计算机所使用的是由“0”和“1”组成的二进制数，二进制是计算机语言的基础。计算机发

明之初,人们只能用计算机语言去命令计算机进行操作,一句话,就是写出一串串由“0”和“1”组成的指令序列交由计算机执行,这种语言,就是机器语言。使用机器语言是十分痛苦的,特别是在程序有错需要修改时,更是如此。而且,由于每台计算机的指令系统往往各不相同,所以在一台计算机上执行的程序,要想在另一台计算机上执行,必须另外编写程序,造成了重复工作。但由于使用的是针对特定型号计算机的语言,故而运算效率是所有语言中最高的。机器语言是第一代计算机语言。

2. 汇编语言

为了减轻使用机器语言编程的痛苦,人们进行了一种有益的改进:用一些简洁的英文字母、符号串来替代一个特定指令的二进制串,比如用 ADD 代表加法,用 MOVE 代表数据传递,等等。这样一来,人们很容易读懂并理解程序在干什么,纠错及维护都变得方便了,这种程序设计语言就称为汇编语言,即第二代计算机语言。然而,计算机是不认识这些符号的,这就需要一个专门的程序,专门负责将这些符号翻译成二进制数的机器语言,这种翻译程序称为汇编程序。

汇编语言同样十分依赖于机器硬件,移植性不好,但效率仍十分高,针对计算机特定硬件而编制的汇编程序,能准确发挥计算机硬件的功能和特长,程序精炼而质量高,所以至今仍是一种常用而强有力的软件开发工具。

3. 高级语言

从最初与计算机交流的痛苦经历中,人们意识到,应该设计一种这样的语言,这种语言接近于数学语言或人类的自然语言,同时又不依赖于计算机硬件,编出的程序能在所有机器上通用。经过努力,1954年,第一个完全脱离机器硬件的高级语言——FORTRAN 问世了,50多年来,共有几百种高级语言出现,有重要意义的有几十种,影响较大、使用较普遍的有 FORTRAN、ALGOL、COBOL、BASIC、LISP、SNOBOL、PL/1、Pascal、C、PROLOG、Ada、C++、Visual C++、Visual Basic、Delphi、Java 等。

高级语言的发展也经历了从早期语言到结构化程序设计语言,从面向过程语言到非过程化程序语言的过程。相应地,软件的开发也由最初的个体手工作坊式的封闭式生产,发展为产业化、流水线式的工业化生产。

20世纪60年代中后期,软件越来越多,规模越来越大,而软件的生产基本上是各自为战,缺乏科学规范的系统规划与测试、评估标准,其恶果是大批耗费巨资建立起来的软件系统,由于含有错误而无法使用,甚至带来巨大损失,软件给人的感觉越来越不可靠,以致几乎没有不出错的软件。这一切,极大地震动了计算机界,史称“软件危机”。人们认识到:大型程序的编制不同于编写小程序,它应该是一项新的技术,应该像处理工程一样处理软件研制的全过程。程序的设计应易于保证正确性,也便于验证正确性。1969年提出了结构化程序设计方法,1970年第一个结构化程序设计语言——Pascal 语言出现,这标志着结构化程序设计时代的开始。

20世纪80年代初开始,在软件设计思想上,又产生了一次革命,其成果就是面向对象的程序设计。在此之前的高级语言,几乎都是面向过程的,程序的执行是流水线似的,在一个模块被执行完成前,人们不能干别的事,也无法动态地改变程序的执行方向。这和人们日常处理事物的方式是不一致的,对我们而言是希望发生一件事就处理一件事,也就是说,不能面向过程,而应

是面向具体的应用功能,也就是对象(object)。其方法就是软件的集成化,如同硬件的集成电路一样,生产一些通用的、封装紧密的功能模块,称之为软件集成块,它与具体应用无关,但能相互组合,完成具体的应用功能,同时又能重复使用。对用户来说,只关心它的接口(输入量、输出量)及能实现的功能,至于如何实现的,那是它内部的事,用户完全不用关心,C++、Visual Basic、Delphi 就是典型代表。

高级语言的下一个发展目标是面向应用,也就是说:只需要告诉程序你要干什么,程序就能自动生成算法,自动进行处理,这就是非过程化的程序语言。

高级语言的发展趋势是模块化、简明性和形式化。

① 模块化:不仅语言具有模块成分,程序由模块组成,而且语言本身的结构也是模块化的。

② 简明性:涉及的基本概念不多,成分简单,结构清晰,易学易用。

③ 形式化:发展合适的形式体系,以描述语言的语法、语义、语用。另外,程序设计语言的并行化和可视化也是人们追求的目标。

1.1.2 程序设计语言的分类

程序设计语言的基本成分有:① 数据成分,用于描述程序所涉及的数据;② 运算成分,用以描述程序中所包含的运算;③ 控制成分,用以描述程序中所包含的控制;④ 传输成分,用以表达程序中数据的传输。

程序设计语言按照语言级别可以分为低级语言和高级语言。低级语言有机器语言和汇编语言。低级语言与特定的机器有关,功效高,但使用复杂、烦琐、费时、易出差错。机器语言是表示成数码形式的机器基本指令集,或者是操作码经过符号化的基本指令集。汇编语言是机器语言中地址部分符号化的结果,或进一步包括宏构造。高级语言的表示方法要比低级语言更接近于待解问题的表示方法,其特点是在一定程度上与具体机器无关,易学、易用、易维护。

程序设计语言按照用户的要求有过程式语言和非过程式语言之分。过程式语言的主要特征是,用户可以指明一系列可顺序执行的运算,以表示相应的计算过程,如 FORTRAN、COBOL、Pascal 等。

按照应用范围有通用语言与专用语言之分,如 FORTRAN、COBOL、Pascal、C 等都是通用语言。目标单一的语言称为专用语言,如 APT 等。

按照使用方式有交互式语言和非交互式语言之分。具有反映人机交互作用的语言成分的语言称为交互式语言,如 BASIC 等。不反映人机交互作用的语言称为非交互式语言,如 FORTRAN、COBOL、ALGOL69、Pascal、C 等都是非交互式语言。

按照成分性质有顺序语言、并发语言和分布语言之分。只含顺序成分的语言称为顺序语言,如 FORTRAN、C 等。含有并发成分的语言称为并发语言,如 Pascal、Modula 和 Ada 等。考虑到分布计算要求的语言称为分布语言,如 Modula 便属于分布语言。

1.1.3 常见的程序设计语言

1. ActionScript

ActionScript 是 Flash 的脚本语言,是在 Flash 动画设计中使用的编程语言,是一种强大的、面向对象的编程语言,可以制作出极富感染力的 Web 内容。

2. Ada

Ada 是一种表现力很强的通用程序设计语言,它是美国国防部为克服软件开发危机,耗费巨资,历时近 20 年研制成功的。它被誉为第四代计算机语言的成功代表。

3. 汇编语言

汇编语言(assembly language)是面向机器的程序设计语言。在汇编语言中,用助记符(memoni)代替操作码,用地址符号(symbol)或标号(label)代替地址码。这样用符号代替机器语言的二进制码,就把机器语言变成了汇编语言,于是汇编语言亦称为符号语言。

4. BASIC

BASIC 是一种高级语言,它的英文含义是“初学者通用符号指令代码”,是在 1965 年 5 月由美国科学家托马斯·库尔兹研制出来的。10 多年后,比尔·盖茨(现微软公司总裁)把它移植到 PC 上。40 多年来,BASIC 语言一直是初学计算机语言者使用最广泛的一种高级语言。它能进行数值计算、画图、演奏音乐,功能十分强大,学习起来也非常容易。

5. FORTRAN

FORTRAN 是英文 FORmula TRANslator 的缩写,译为“公式翻译器”,它是世界上最早出现的计算机高级程序设计语言,广泛应用于科学和工程计算领域。

6. VBScript

VBScript 是 Visual Basic Script 的简称,即 Visual Basic 描述语言,有时也被缩写为 VBS。VBScript 是微软开发的一种脚本语言,可以看做是 Visual Basic 语言的简化版,与 VBA 的关系也非常密切。它具有源语言容易学习的特性。目前,这种语言广泛应用于网页制作和 ASP 程序中。

7. C

C 语言是目前世界上非常流行且使用广泛的高级程序设计语言之一。C 语言对操作系统和系统使用程序以及需要对硬件进行操作的场合明显优于其他高级语言,许多大型应用软件都是用 C 语言编写的。

8. COBOL

COBOL 是适合于商业及数据处理的类似英语的程序设计语言。这种语言可使商业数据处理过程精确表达。

9. HTML

HTML(HyperText Markup Language)即超文本置标语言,是 WWW 的描述语言。HTML 也是网络的通用语言,一种简单、通用的全置标记语言。

10. Java

Java 包含了一种计算机编程语言和一个平台。由 Sun 公司发布,并作为一种开放的标准进行提供。Java 平台包括了 Java 虚拟机和 Java 应用程序接口(API)。Java 非常适合于企业网络和 Internet 环境,现在已成为 Internet 中最受欢迎、最有影响的编程语言之一。

11. JavaScript

JavaScript 是一种由 Netscape 的 LiveScript 发展而来的脚本语言,它提高了与 Java 的兼容性。是适应动态网页制作的需要而诞生的一种新的编程语言。

12. SQL

SQL 全称是“结构化查询语言”(Structured Query Language)。目前,常用的数据库开发系统都支持 SQL 语言作为查询语言。

13. Visual Basic

Visual Basic (简称 VB)是一种由微软公司开发的包含协助开发环境的事件驱动编程语言,是一种通用性的高级程序设计语言。

1.1.4 程序设计语言的编译

任何高级语言编写的源程序都不能在计算机上直接执行。计算机不能直接理解高级语言,只能直接理解机器语言,所以必须把高级语言翻译成机器语言,计算机才能执行高级语言编写的程序。翻译的方式有两种,一种是编译,另一种是解释。两种方式只是翻译的时间不同。编译型语言写在程序执行之前,需要一个专门的编译过程,把程序编译成为机器语言的文件,比如.exe 文件,以后运行时就不用重新翻译了,直接使用编译的结果就行了(.exe 文件),因为翻译只做了一次,运行时不需要翻译,所以编译型语言的程序执行效率高。解释则不同,解释性语言的程序不需要编译,省了一道工序。解释性语言在运行程序的时候才翻译,比如解释性 BASIC 语言,专门有一个解释器能够直接执行 BASIC 程序,每条语句都是执行的时候才翻译。这样,解释性语言每执行一次就要翻译一次,效率比较低。Java 很特殊,Java 程序也需要编译,但是没有直接编译成机器语言,而是编译成字节码,然后用解释方式执行字节码。

1.2 关于 Visual Basic

Visual Basic 提供了开发 Windows 应用程序最迅速、最简洁的方法。它提供了一整套工具,供用户开发应用程序。

什么是 Visual Basic? Visual 指的是开发图形用户界面(GUI)的方法。在图形用户界面下,不需要编写大量代码去描述界面元素的外观和位置,而只要把预先建立的对象加到屏幕上的适当位置,再进行简单的设置即可。Basic 指的是 BASIC (Beginner's All-purpose Symbolic Instruction Code)语言,是一种应用十分广泛的计算机语言。Visual Basic 在原有 BASIC 语言的基础上进一步发展,至今包含了数百条语句、函数及关键词,其中很多和 Windows GUI 有直接关系。专业人员可以用 Visual Basic 实现其他任何 Windows 编程语言的功能,而初学者只要掌握几个关键词就可以建立简单的应用程序。

1.2.1 Visual Basic 的发展

1991 年,Microsoft 公司推出 Visual Basic 1.0 版后,虽然存在一些缺陷,但仍受到了广大程序员的青睐。随后,Microsoft 公司于 1992 年秋推出 2.0 版,1993 年 4 月推出 3.0 版,1995 年 10 月推出 4.0 版,1997 年推出 5.0 版,1998 年推出 6.0 版。随着版本的改进,Visual Basic 已逐渐成为简单易学、功能强大的编程工具。从 1.0 版到 4.0 版,Visual Basic 只有英文版;而 5.0 版以后的 Visual Basic 在推出英文版的同时,又推出了中文版,这大大方便了中国用户。

Visual Basic 6.0 包括 3 种版本,分别为学习版、专业版和企业版。这些版本是在相同的基础

上建立起来的,因此大多数应用程序可在3种版本中通用。3种版本适合于不同的用户层次。

学习版是 Visual Basic 的基础版本,可使编程人员轻松开发 Windows 应用程序。该版本包括所有的内部控件(标准控件)、网格(Grid)控件、Tab 对象以及数据绑定控件。

专业版为专业编程人员提供了一整套功能完备的软件开发工具。该版本包括学习版的全部功能,同时还包括 ActiveX 控件、Internet 控件、Crystal Report Writer 和报表控件。

企业版使得专业编程人员能够开发功能强大的组内分布式应用程序。该版本包括专业版的全部功能,同时具有自动化管理器、部件管理器、数据库管理工具、Microsoft Visual SourceSafe 面向工程版的控制系统等。

3种版本中,企业版功能最全,而专业版包括了学习版的功能,用户可以根据自己的需要选用不同的版本。对于大多数用户来说,专业版完全可以满足需要。

本书使用的是 Visual Basic 6.0 中文企业版,但其内容可用于专业版和学习版,所有程序可以在专业版和学习版中运行。

Visual Basic 6.0 是专门为 Microsoft 的 32 位操作系统设计的,可用来建立 32 位的应用程序。在 Windows 9x、Windows NT 或 Windows 2000 环境下,用 Visual Basic 6.0 的编译器可以自动生成 32 位应用程序。这样的应用程序在 32 位操作系统下运行,速度更快、更安全,并且更适合在多任务环境下运行。

1.2.2 Visual Basic 的特点

Visual Basic 有以下几个主要的功能特点。

1. 提供了可视化编程工具

在用传统程序设计语言设计程序时,都是通过编写程序代码来设计用户界面,在设计过程中看不到界面的实际效果,必须编译后运行程序才能观察。如果对界面的效果不满意,还要回到程序中去修改。有时候,这种编程—编译—修改的操作可能要反复多次,大大影响了软件开发效率。Visual Basic 提供了可视化设计工具,把 Windows 界面设计的复杂性“封装”起来,开发人员不必为界面设计而编写大量程序代码,只需要按设计要求的屏幕布局,用系统提供的工具在屏幕上画出各种“部件”,即图形对象,并设置这些图形对象的属性。Visual Basic 自动产生界面设计代码,程序设计人员只需要编写实现程序功能的那部分代码,从而可以大大提高程序设计效率。

2. 面向对象的程序设计

4.0 版特别是 5.0 版以后的 Visual Basic 支持面向对象的程序设计,但它与一般的面向对象的程序设计语言(如 C++)不完全相同。在一般的面向对象程序设计语言中,对象由程序代码和数据组成,是抽象的概念;而 Visual Basic 则是应用面向对象的程序设计(OOP)方法,把程序和数据封装起来作为一个对象,并为每个对象赋予应有的属性,使对象成为实在的东西。在设计对象时,不必编写建立和描述每个对象的程序代码,而是用工具画在界面上,Visual Basic 自动生成对象的程序代码并封装起来。每个对象以图形方式显示在界面上,都是可见的。

3. 事件驱动的编程机制

Visual Basic 通过事件来执行对象的操作。一个对象可能会产生多个事件,每个事件都可以通

过一段程序来响应。例如,命令按钮是一个对象,当用户单击该按钮时,将产生一个“单击”(Click)事件,而在产生该事件时将执行一段程序,用来实现指定的操作。

在用 Visual Basic 设计大型应用软件时,不必建立具有明显开始和结束的程序,而是编写若干个微小的子程序,即过程,这些过程分别面向不同的对象,由用户操作引发某个事件来驱动执行某种特定的功能,或者由事件驱动程序调用通用过程来执行指定的操作。这样可以方便编程人员,提高效率。

4. 结构化的程序设计语言

Visual Basic 是在 BASIC 和 Quick BASIC 语言的基础上发展起来的,具有高级程序设计语言的语句结构,接近于自然语言和人类的逻辑思维方式,其语句简单易懂,其编辑器支持彩色代码,可自动进行语法错误检查,同时具有功能强且使用灵活的调试器和编译器。

Visual Basic 是解释型语言,在输入代码的同时,解释系统将高级语言分解翻译成计算机可以识别的机器指令,并判断每个语句的语法错误。在设计 Visual Basic 程序的过程中,随时可以运行程序,而在整个应用程序设计好之后,可以编译生成可执行文件(.exe),脱离 Visual Basic 环境,直接在 Windows 环境下运行。

5. 提供了易学易用的应用程序集成开发环境

在 Visual Basic 的集成开发环境中,用户可以设计界面、编写代码、调试程序,直至将应用程序编译成可执行文件在 Windows 上运行,使用户在友好的开发环境中工作。

6. 支持多种数据库系统的访问

Visual Basic 系统具有很强的数据库管理功能。利用数据控件和数据库管理窗口,可以直接建立或处理 Microsoft Access 格式的数据库,并提供了强大的数据存储和检索功能。同时,Visual Basic 还能直接编辑和访问其他外部数据库,如 Btrieve、dBase、FoxPro、Paradox 等,这些数据库格式都可以用 Visual Basic 编辑和处理。

Visual Basic 提供开放式数据连接(Open DataBase Connectivity),即 ODBC 功能,它可通过直接访问或建立连接的方式使用并操作后台大型网络数据库,如 SQL Server、Oracle 等。在应用程序中,可以使用结构化查询语言(SQL)数据标准,直接访问服务器上的数据库,并提供了简单的面向对象的库操作指令、多用户数据库访问的加锁机制和网络数据库的 SQL 编程技术,为单机上运行的数据库提供了 SQL 网络接口,以便在分布式环境中快速而有效地实现客户/服务器(Client/Server)方案。

7. 支持动态数据交换(DDE)、动态链接库(DLL)和对象的链接与嵌入(OLE)

动态数据交换是 Microsoft Windows 除了剪贴板和动态链接函数以外,在 Windows 内部交换数据的第三种方式。利用这项技术可使 Visual Basic 开发的应用程序与其他 Windows 应用程序之间建立数据通信。

动态链接库中存放了所有 Windows 应用程序可以共享的代码和资源,这些代码或函数可以用多种语言写成。Visual Basic 利用这项技术可以调用任何语言产生的 DLL。

对象的链接与嵌入是 Visual Basic 访问所有对象的一种方法。利用 OLE 技术,Visual Basic 可以将其他应用软件作为一个对象嵌入到应用程序中进行各种操作。