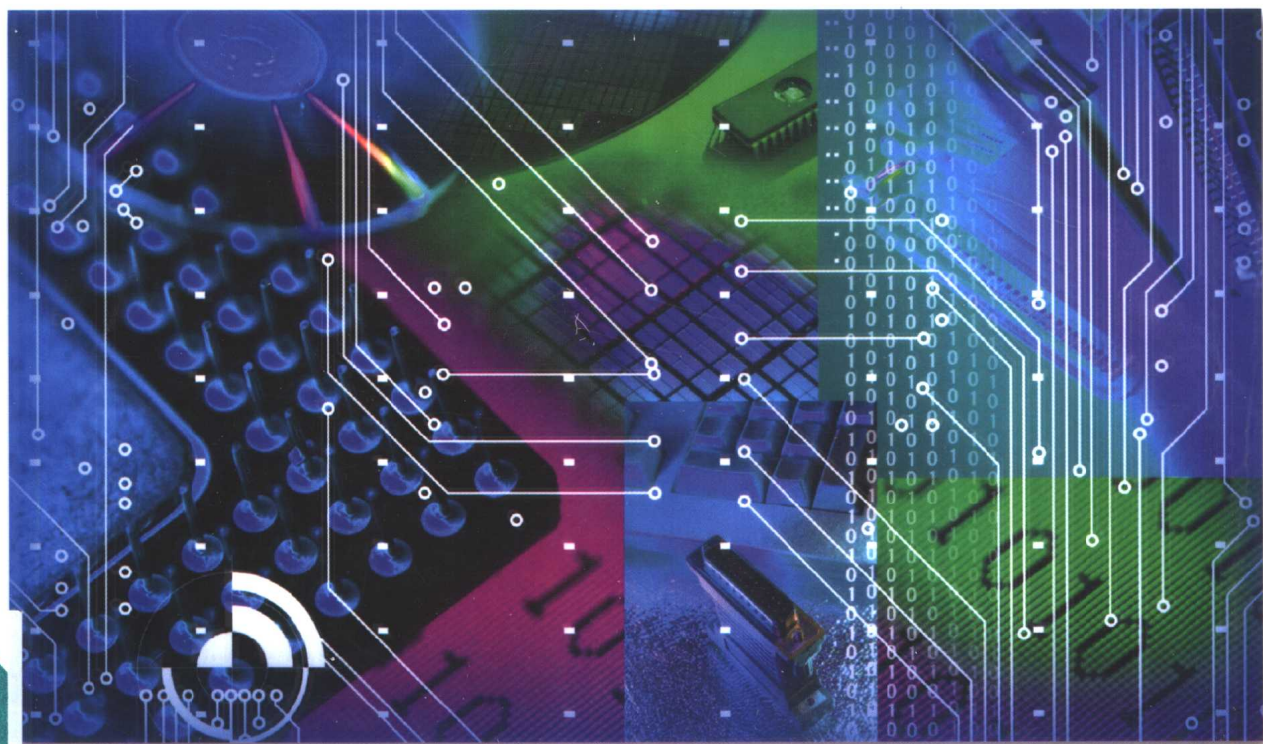


21 世纪高等院校计算机系列教材

Visual Basic 6.0 程序设计

贺世娟 陈冀川 主编 付灵丽 侯立坤 副主编



中国水利水电出版社
www.waterpub.com.cn

21 世纪高等院校计算机系列教材

Visual Basic 6.0 程序设计

贺世娟 陈冀川 主编

付灵丽 侯立坤 副主编

中国水利水电出版社

内 容 提 要

本书是为学习和使用 Visual Basic 进行程序设计的初、中级用户编写的,该书以 Visual Basic 6.0 为背景,介绍了在 Windows 环境下进行程序设计的基本思想和方法,主要内容包括: Visual Basic 语言程序设计基础,面向对象的概念和程序设计方法, Visual Basic 的窗体、控件、菜单等的设计与应用,程序调试和排错等。

本书内容丰富,例题、习题详尽,除注意培养基本的程序设计能力外,重点介绍了可视化编程技术、面向对象方法。不仅可以作为广大软件开发人员和自学者的参考用书,还可以作为高等院校非计算机专业或计算机应用专业的教材。

图书在版编目(CIP)数据

Visual Basic 6.0 程序设计/贺世娟,陈冀川主编. —北京:中国水利水电出版社, 2002

(21 世纪高等院校计算机系列教材)

ISBN 7-5084-1098-X

I.V... II.①贺…②陈…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(2002)第 057061 号

书 名	Visual Basic 6.0 程序设计
主 编	贺世娟 陈冀川
副 主 编	付灵丽 侯立坤
出版、发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@public3.bta.net.cn (万水) sale@waterpub.com.cn 电话: (010) 68359286 (万水)、63202266 (总机)、68331835 (发行部)
经 售	全国各地新华书店
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787×1092 毫米 16 开本 15 印张 335 千字
版 次	2002 年 8 月第一版 2002 年 8 月北京第一次印刷
印 数	0001—5000 册
定 价	20.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

编 委 会

主 编 贺世娟 陈冀川

副主编 付灵丽 侯立坤

编 委 郝淑珍 李惠然 柴 欣 李 娟

 朱怀忠 宋 洁 史巧硕 武优西

 赵卫萍 吴焕丽

前 言

Visual Basic 是基于 Windows 环境应用系统的功能强大的开发工具，既继承了其先辈所具有的程序设计语言简单易学、易用的特点，又在编程系统中引入了面向对象的机制，用一种巧妙的方法把 Windows 编程的复杂性封装起来，提供了一种可视界面的设计方法，使用窗体和控件设计应用程序界面，从而极大地提高了用户开发应用程序的效率。

本书作者有着丰富的教学经验和极强的科研能力，曾利用 Visual Basic 及其他开发工具开发了“计算机基础教学、考试、考务集成系统”、“计算机职称考试系统”等软件，因此对 Visual Basic 有着较深入的理解。为了使初学程序设计的读者能快速地掌握开发基于图形界面的应用程序的能力和解决实际问题的能力，作者精选了 Visual Basic 的内容，本着加强基础、注重实践、勇于创新、突出应用的原则，力求使本书达到可读性、适用性和先进性。为了便于读者自学，在全书的体系结构和内容上注意了由浅入深、深入浅出、循序渐进的方针。为了提高读者的编程技巧，在大部分章节中都提供了典型例题。

本书共分十章，第一、二章介绍程序设计语言的发展及中文 Visual Basic 6.0 概述；第三章介绍 Visual Basic 程序设计基础；第四章介绍窗体及常用控件；第五章为对话框和菜单；第六章为过程与模块；第七章为常用算法；第八章为文件操作；第九章为图形操作；第十章为程序调试。

为了实现理论联系实际，达到良好的学习效果，配合本书我们还编写了《Visual Basic 6.0 程序设计实验教程》，“实验教程”与教材相呼应，在各章均相应地安排了若干个上机试验，以方便读者有计划有目的地进行上机操作，从而达到事半功倍的教学效果。

参加本书编写的有贺世娟、陈冀川、付灵丽、侯立坤、郝淑珍、李惠然、柴欣、李娟、朱怀忠、宋洁、史巧硕、武优西、赵卫萍、吴焕丽等。

本书写作时，参考了大量文献资料，在此向这些文献资料的作者深表感谢。

计算机的发展日新月异，由于编审者水平所限，加之时间十分紧迫，书中定有不妥之处，恳请读者及有关专家批评指正。

编者

2002 年 8 月

目 录

前言

第 1 章 计算机程序设计概述	1
1.1 计算机程序设计的发展.....	1
1.1.1 计算机程序设计语言.....	1
1.1.2 程序设计的发展历程.....	1
1.2 结构化程序设计.....	2
1.2.1 结构化程序设计的概念.....	2
1.2.2 结构化程序设计方法.....	4
1.3 面向对象程序设计概述.....	5
1.3.1 面向对象的程序设计概念.....	6
1.3.2 面向对象的程序设计语言.....	6
1.3.3 面向对象的程序设计方法与结构化程序设计方法的比较.....	7
习题 1	8
第 2 章 中文 Visual Basic 6.0 概述	9
2.1 Visual Basic 简介	9
2.1.1 Visual Basic 的发展过程	9
2.1.2 Visual Basic 的功能特点	10
2.2 Visual Basic 的运行环境、安装和启动	11
2.2.1 Visual Basic 的运行环境	11
2.2.2 Visual Basic 的安装	11
2.2.3 Visual Basic 的启动	12
2.3 Visual Basic 6.0 的集成开发环境	13
2.3.1 主窗口	13
2.3.2 窗体窗口	14
2.3.3 属性窗口	14
2.3.4 工程资源管理器窗口.....	15
2.3.5 代码窗口	16
2.3.6 窗体布局窗口.....	16
2.3.7 对象浏览器窗口.....	16
2.3.8 工具箱窗口.....	17
2.4 设计一个简单的 Visual Basic 应用程序	17
2.4.1 创建应用程序的界面.....	18

2.4.2	编写应用程序的代码.....	19
2.4.3	运行应用程序.....	20
2.4.4	保存文件	20
2.5	Visual Basic 中的对象	21
2.5.1	Visual Basic 中对象的概念	21
2.5.2	对象的属性.....	22
2.5.3	对象的方法.....	23
2.5.4	对象的事件及事件过程.....	23
2.6	生成可执行文件和制作安装盘.....	25
2.6.1	生成可执行文件.....	25
2.6.2	制作安装盘发布应用程序.....	26
2.7	Visual Basic 6.0 的帮助功能	29
2.7.1	了解 MSDN Library 查阅器	30
2.7.2	在 MSDN Library 中定位信息	30
2.7.3	应用 MSDN Library 进行其他工作	32
2.7.4	从 Internet 上获得帮助	33
	习题 2	33
第 3 章	Visual Basic 语言程序设计	35
3.1	Visual Basic 编码规则	35
3.2	Visual Basic 的数制及数据类型	36
3.2.1	Visual Basic 的数制	36
3.2.2	Visual Basic 的数据类型	36
3.3	常量和变量.....	40
3.3.1	常量	40
3.3.2	变量	41
3.4	运算符和表达式.....	44
3.4.1	算术运算符与算术表达式.....	44
3.4.2	关系运算符与关系表达式.....	45
3.4.3	逻辑运算符与逻辑表达式.....	46
3.4.4	字符串运算符与字符串表达式.....	47
3.4.5	数据类型的转换.....	48
3.4.6	运算符的优先级别.....	48
3.5	常用内部函数.....	48
3.5.1	数学函数	48
3.5.2	字符串函数.....	49
3.5.3	转换函数	51
3.5.4	日期函数	52

3.5.5	Shell 函数	53
3.6	基本语句	53
3.6.1	顺序结构程序设计	53
3.6.2	选择结构程序设计	59
3.6.3	循环结构程序设计	69
3.7	数组及应用	78
3.7.1	静态数组	78
3.7.2	动态数组	80
3.7.3	数组的基本操作	81
习题 3		86
第 4 章	窗体设计和常用控件的使用	88
4.1	窗体的类型、使用、属性和方法	88
4.1.1	窗体类型以及如何添加窗体	88
4.1.2	设置启动窗体	89
4.1.3	窗体的属性、事件和方法	89
4.1.4	控件的概念和使用方法	91
4.1.5	窗体对象的生命过程	92
4.2	常用控件的基本属性	94
4.2.1	控件的种类	94
4.2.2	控件的常用属性	94
4.2.3	设置 Tab 键转移焦点的顺序	96
4.2.4	控件默认属性	98
4.3	基本控件的使用	98
4.3.1	命令按钮	98
4.3.2	使用标签控件显示文字	99
4.3.3	使用文本框控件显示和输入文本	100
4.3.4	使用单选按钮和检查框控件	102
4.3.5	使用框架控件	104
4.3.6	提供大量选择的列表框控件	106
4.3.7	提供大量选择的组合框控件	110
4.3.8	使用滚动条控件	112
4.3.9	使用时钟控件	113
4.3.10	驱动器、目录和文件列表框控件	115
4.3.11	图形控件	117
4.3.12	Image 控件和 PictureBox 控件	118
4.4	ActiveX 控件的使用	119
4.4.1	ActiveX 控件	119

4.4.2 向工具箱中添加 ActiveX 控件	119
习题 4	120
第 5 章 对话框和菜单	121
5.1 对话框的分类和使用	121
5.1.1 系统预定义对话框	121
5.1.2 通用对话框	123
5.1.3 自定义对话框	128
5.2 菜单的设计和使用	130
5.2.1 菜单设计	130
5.2.2 快捷菜单的设计	132
5.2.3 菜单设计举例	133
习题 5	135
第 6 章 模块与过程	136
6.1 Visual Basic 的代码模块	136
6.2 Visual Basic 的过程	136
6.2.1 Visual Basic 过程的分类	137
6.2.2 过程的创建与调用	139
6.2.3 参数传递	143
习题 6	146
第 7 章 常用算法	147
7.1 排序算法	147
7.1.1 比较互换法	147
7.1.2 选择法排序	148
7.1.3 冒泡法排序	148
7.2 查找	149
7.3 素数的求法	151
7.3.1 由素数的定义求素数	151
7.3.2 用筛选法求素数	152
7.4 解一元方程	153
7.4.1 用牛顿迭代法解一元方程	153
7.4.2 用二分法解一元方程	154
7.5 数值积分	155
7.6 多项式及其导数的值	157
7.7 数制转换	159
7.8 Visual Basic 的其他使用	161
习题 7	165
第 8 章 文件操作	166

8.1	文件的基本概念.....	166
8.1.1	数据文件的类型.....	167
8.1.2	文件号	167
8.2	顺序文件.....	167
8.2.1	顺序文件的打开.....	167
8.2.2	顺序文件的关闭.....	168
8.2.3	顺序文件的读操作.....	168
8.2.4	顺序文件的写操作.....	170
8.3	随机文件.....	171
8.3.1	随机文件的打开和关闭.....	171
8.3.2	定义记录类型.....	172
8.3.3	随机文件的读操作.....	172
8.3.4	随机文件的写操作.....	172
8.4	二进制文件.....	175
8.4.1	二进制文件的打开和关闭.....	175
8.4.2	二进制文件的读、写操作.....	175
8.5	文件与目录的操作语句和函数简介.....	177
8.6	文件与目录的操作控件简介.....	179
8.7	使用 File System Object 对象模型编程简介	179
8.7.1	File System Object 模型介绍	179
8.7.2	文件系统对象.....	179
8.8	使用文件系统对象编程.....	180
8.8.1	FileSystemObject 对象	180
8.8.2	管理驱动器.....	182
8.8.3	管理文件夹.....	185
8.8.4	管理文件	192
	习题 8	197
第 9 章	图形操作.....	199
9.1	坐标系统.....	199
9.1.1	使用默认刻度.....	200
9.1.2	选择标准刻度.....	200
9.1.3	创建自定义刻度.....	201
9.2	使用图形控件.....	202
9.2.1	添加图片	202
9.2.2	运行时删除图片.....	203
9.2.3	移动图片和改变图片尺寸.....	204
9.3	用图形方法创建图形.....	204

9.3.1	画点	205
9.3.2	画各种直线和形状.....	206
9.3.3	绘制方框	207
9.3.4	画圆	209
9.3.5	画圆弧	209
9.3.6	画椭圆	210
9.3.7	指定线宽	211
9.3.8	指定实线或虚线.....	212
9.3.9	窗体加载时创建图形.....	212
9.4	使用颜色.....	212
9.4.1	使用 RGB 函数	213
9.4.2	使用颜色属性.....	213
	习题 9	214
第 10 章	错误处理与程序调试.....	215
10.1	Visual Basic 的错误分类	215
10.1.1	语法错误.....	215
10.1.2	运行错误.....	216
10.1.3	逻辑错误 (logic error)	216
10.2	Visual Basic 的调试环境	216
10.2.1	调试工具栏.....	217
10.2.2	调试菜单.....	218
10.3	Visual Basic 应用程序所处的三种工作模式.....	219
10.4	Visual Basic 的调试技术	220
10.4.1	使用调试窗口.....	220
10.4.2	使用中断模式.....	222
10.5	Visual Basic 的错误处理	225

第 1 章 计算机程序设计概述

1.1 计算机程序设计的发展

1.1.1 计算机程序设计语言

计算机之所以能自动地进行工作，是因为它采用了程序存储的原理。计算机的工作体现为执行程序。程序是控制计算机完成某项功能的一组有序指令的集合，编写程序所使用的语言称为程序设计语言，它是人与计算机之间进行信息交流的工具。

自世界上第一台计算机诞生起，至今仅有短短的 50 余年，计算机技术得到了迅速发展，程序设计语言也经历了三个发展阶段：机器语言、汇编语言和高级语言。目前世界上流行的计算机语言有上千种之多，被人们广泛使用的也有数十种。

计算机语言按其与硬件接近的程度，可以划分为低级语言和高级语言两大类。机器语言和汇编语言属于低级语言，分别称为第一代语言和第二代语言；高级语言包括过程式语言和非过程式语言，即称为第三代语言。

1.1.2 程序设计的发展历程

程序设计发展的历史，大体上可以划分为以下几个不同的时期。

在 50 年代，程序都是用指令代码或汇编语言来编写，这种程序的设计相当烦琐，编写、调试一个大的程序要花费很长的时间，培养一个熟练的程序员更需要经过长时间的训练和实习，在这种局面下，要想普及计算机的应用是非常困难的。

60 年代高级语言的出现，使程序设计变得简单，缩短了编程周期，显示出强大的生命力。编制程序已不再是软件专业人员才能做的事，一般工程技术人员经过短期的培训、学习，也可以使用计算机编程。这个时期，随着计算机的应用日益广泛地渗透到各学科领域，发展了一系列不同风格的、为不同对象服务的程序设计语言。其中较为著名的有 FORTRAN, COBOL, ALGOL, LISP, PL/1, PASCAL 等十几种语言。高级语言的蓬勃兴起，使得编译和形式语言理论日趋完善，这是该时期的主要特征。但就整个程序设计方法而言，并无实质性的改进。

60 年代末到 70 年代初，诸如操作系统、数据库等大型系统软件的相继出现，给程序设计带来了新的问题。大型系统软件的研制需要花费大量的资金和人力，而研制出来的产品却是可靠性差，错误多，且不易维护和修改。当时，人们称这种现象为“软件危机”。

“软件危机”震动了软件界，程序设计的传统习惯和工作方式导致了不清晰的程序结构，使得程序的可靠性难以保障；另一方面，程序设计工具的严重缺乏也使得大型系统的

开发陷入困境。此时人们开始重新审视程序设计中的一些最基本的问题。例如，程序的基本组成部分是什么？应该用什么样的方法来设计程序？如何保证程序设计正确？程序设计的主要方法和技术应如何规范等。

60 年代末，E.W.Dijkstra 首先提出了结构化程序设计的思想，强调从程序结构和风格上来研究程序设计。经过几年的探索和实践，结构化程序设计的应用确实取得了成效，用结构化程序设计方法编写出来的程序不仅结构良好，易写易读，而且易于证明其正确性。

到 20 世纪 70 年代末，结构化程序设计方法得到了长足的发展，Niklaus Wirth 又提出了“算法+数据结构=程序设计”的程序设计方法，他将软件划分成若干个能够单独命名和编址的部分，每个部分称为模块，模块化使软件的管理和维护更加方便、有效。在 80 年代，模块化程序设计方法普遍被人们所接受。

虽然几十年来结构化程序设计技术得到了广泛的使用，但有些问题并没有得到很好的解决。比如，结构化方法的软件开发是对问题的求解过程，从认识论角度看，软件开发过程应包括人们对要解决问题的认识及基于认识所进行的描述。而结构化设计方法不能直接反映出人类认识问题的过程。另外，在结构化设计方法中，程序模块和数据结构是松散地耦合在一起的，因此，当应用程序比较复杂时，也容易出错，并难以维护。随着计算机软件的发展，软件系统越来越复杂庞大，结构化程序设计方法已显得力不从心。

在上述环境下，软件开发中的各种概念和方法逐渐被积累，人们提出了如何超越程序的复杂性障碍，如何在计算机系统中自然地表示客观世界等问题；并且提出了面向对象的程序设计方法。面向对象的方法不再将问题分解为过程，而是将问题分解为对象，对象将自己的属性和方法封装成一个整体，供程序设计者使用。对象之间的相互作用则通过消息传递来实现。用面向对象的程序设计方法，可以使人们对复杂系统的认识过程、系统的程序设计过程和实现过程尽可能地一致。有人预测，这种“对象+消息”的面向对象的程序设计模式将逐渐取代“算法+数据结构”的面向过程的程序设计模式。

1.2 结构化程序设计

1.2.1 结构化程序设计的概念

1.2.1.1 结构化程序的基本结构

结构化程序包含三种基本结构：顺序结构、选择结构和循环结构。程序员可以用这三种基本结构来编写程序，表示一个好的算法，从而使程序的结构清晰、易读易懂且质量好、效益高。

1. 顺序结构

图 1.1 表示了一个顺序结构形式，顺序结构是一种最简单、最基本的结构，在顺序结构内，各块是按照它们出现的先后顺序依次执行。从图中可以看出它有一个入口 a 点，一个出口 b 点，在结构内 A 框和 B 框都是顺序执行的处理框。

2. 选择结构

图 1.2 所示是一个选择结构。选择结构中包含一个判断框，根据给定的条件 P 是否成立，决定是执行 A 框还是 B 框。当条件成立时，执行 A ，否则执行 B 。 A 框或 B 框可以是空框，即不执行任何操作，但判断框中的两个分支，执行完 A 或 B 后都必须汇合在一起，从出口 b 退出，然后接着执行其后的过程。在选择结构中，虽然程序产生了分支，但是对于整个虚线框而言，它仍然是只具有一个入口 a 和一个出口 b 的结构。

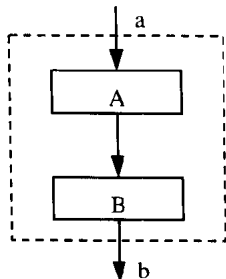


图 1.1 顺序结构示意图

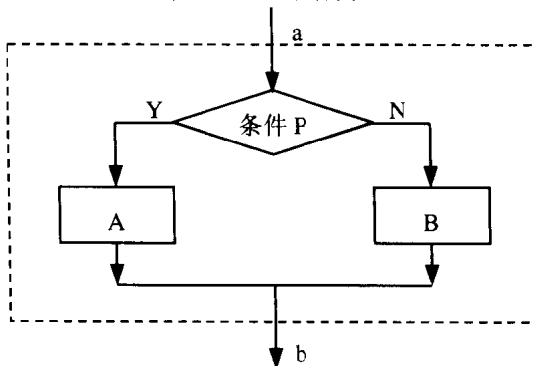


图 1.2 选择结构流程图

3. 循环结构

循环结构是指在一定条件下反复执行一个程序块的结构。循环结构也是只有一个入口，一个出口。根据循环条件的不同，循环结构分为两种：当型循环结构和直到型循环结构。

(1) 当型循环结构如图 1.3 所示，其功能是，当条件 P 成立时，执行 A 框操作，执行完 A 操作后，再判断 P 条件是否成立，如果成立，再次执行 A 操作，如此重复执行 A 操作，直到判断 P 条件不成立时才停止循环。此时不执行 A 操作，而从出口 b 脱离循环结构。

(2) 直到型循环结构如图 1.4 所示，其功能是，先执行 A 框操作，然后判断给定条件 P 是否成立，如果不成立，再次执行 A 操作；然后再对 P 进行判断，如此反复，直到给定的 P 条件成立为止。此时不再执行 A 框，从出口 b 直接脱离循环。

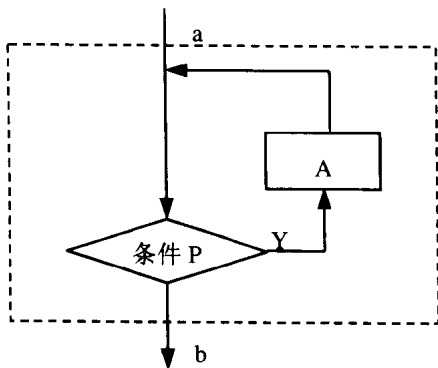


图 1.3 当型循环结构流程图

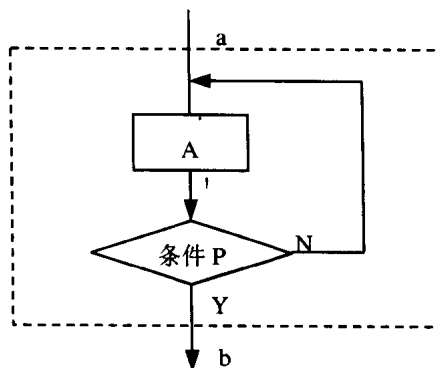


图 1.4 直到型循环结构流程图

由以上三种基本结构构成的程序，称为结构化程序。一个结构化程序应具有以下特点：

- 有一个入口。
- 一个出口。
- 没有死语句，即每一个语句都应该有一条从入口到出口的路径通过它（至少通过一次）。
- 没有死循环（无限制的循环）。

实践证明，任何满足以上四个条件的程序，都可以表示为由以上三种基本结构所构成的结构化程序；反之，任何一个结构化程序都可以分解为上述三种基本结构。

1.2.1.2 结构化程序的设计方法

结构化程序主要采用自上而下、逐步细化的设计方法，即先全局后局部、先整体后细节、先抽象后具体的设计方法。由于实际问题往往比较复杂，为了提高效率，在进行结构化程序设计时，常常使用关键部分优先考虑的设计方法。

1.2.1.3 结构化程序的组织结构

在结构化程序中常常用模块化结构来组织程序，图 1.5 给出了用模块化结构组织程序的示意图。

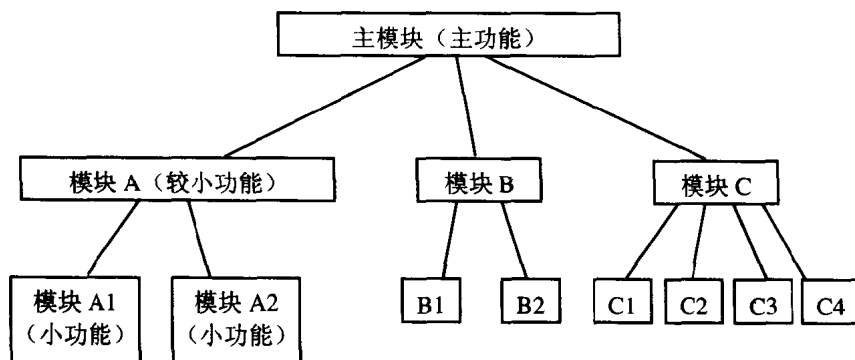


图 1.5 程序模块化分图

1.2.2 结构化程序设计方法

1.2.2.1 逐步细化法

将一个完整的问题分解成若干相对独立的子问题，只要这些子问题能分别得到正确的解决，整个问题也就解决了。子问题又可进一步分解为若干子问题，这样可以一直重复下去，直到每个问题都已简单到令人满意的程度为止。对每步分解，都要作出分解方法的决策，不同的决策会导致不同的解法。这种程序设计方法称之为逐步细化法，也就是在编写程序时一步一步地不断细化的过程。

细化过程可以自顶开始向下进行，也可以从底端开始向上进行。根据经验，程序自顶向下设计，再不断细化这种处理方法效果较好。细化过程的技术有三种，分别称为“划分和解决”的分割技术、“作出有限进展”的递推技术和“分析情况”的分析技术。

1. 分割技术

分割技术大致可以分为以下几个步骤：

(1) 把问题划分成不相交的一些部分, 直到可用复合语句为止。

(2) 依次解决划分后的每一部分问题。

2. 递推技术

对问题作出有限进展, 直到用循环语句实现为止。如果我们对问题找到一个解的方向, 并作出了有限进展, 这时再重复递推, 直到最终达到完全解。这是我们在程序设计时常用的方法。

3. 分析技术

对问题用情况分析来精细化, 直到可用条件语句为止。

1.2.2.2 模块法

模块化设计技术与方法, 是程序设计中应用最早的一种重要方法, 它早在使用低级语言时期就已出现, 但是, 只有在结构化程序设计中, 这种技术与方法才得以发展、充实、提高与完善。

模块化程序设计方法是指在程序设计中, 将一个复杂的算法(或程序)分解成若干个相对独立、功能单一的模块, 利用这些模块即可适当地组合成所需的全局算法(或程序)。这里所说的模块, 是一个可供调用(即让其他模块调去使用)的相对独立的操作块(或程序段), 每个模块都是由三种基本结构组成的结构化模块。在模块化结构中, 整个系统犹如搭积木一样, 由各模块适当组合而成。模块之间的相对独立性, 使每个模块均可独立地分别进行分析、设计、编写、调试、修改和扩充, 并且不会影响到其他模块和全局算法(或程序), 这表明模块化结构不仅使复杂的软件研制工作得以简化、缩短开发周期、节省开发费用、提高软件质量, 而且还可以有效地防止模块间错误的扩张, 增加整个系统的稳定性与可靠性。同时还可使软件具备结构灵活便于组装、层次分明利于维护、条理清晰容易理解的优点。因此, 模块化程序设计方法是结构化程序设计的重要方法之一, 主要用于算法设计阶段。

1.3 面向对象程序设计概述

结构化程序设计技术虽已使用了几十年, 但以下问题仍未得到很好地解决:

(1) 这种面向过程的设计方法与人们习惯的思维方法仍然存在一定的差距, 所以很难自然、准确地反映真实世界。因而用此方法开发出来的软件, 有时很难保证其质量, 甚至需要进行重新开发。

(2) 该方法在实现中只突出了实现功能的操作方法(模块), 而被操作的数据(变量)处于实现功能的从属地位, 即程序模块和数据结构是松散地耦合在一起。因此, 当应用程序比较复杂时, 容易出错、难以维护。

由于上述缺陷, 结构化程序设计方法已不能满足现代化软件开发的要求, 一种全新的软件开发技术应运而生, 这就是面向对象的程序设计(Object Oriented Programming, 简称OOP)。

1.3.1 面向对象的程序设计概念

面向对象的程序设计在 20 世纪 80 年代初就提出了,它起源于 Smalltalk 语言。用面向对象的方法来解决实际问题,不再将问题分解为过程,而是将问题分解为对象。对象是现实世界中可以独立存在、可以被区分的一些实体,也可以是一些概念上的实体。世界是由许多对象组成的。对象有自己的数据(属性),也包括作用于数据的操作(方法)。对象将自己的属性和方法封装成一个整体,供程序设计者使用。对象之间的相互作用通过消息传递来实现。下面介绍一些面向对象的程序设计中经常用到的术语。

1. 对象

对象是属性和服务的封装体。对象的属性用于描述对象的静态数据特征,对象的属性可用系统的或用户自定义的数据类型来表示,也可用抽象的数据类型表示。对象属性值的集合称为对象的状态(state)。

对象的服务用于描述对象的动态特征,也称之为行为或功能,它是定义在对象属性基础上的一组操作方法(method)的集合。

2. 类

类是对象的抽象及定义,是具有共同属性和操作的多个对象的相似特征的统一描述。是具有共同特征的对象集合,而对象只是类的一个实例。

3. 消息

消息是面向对象系统中实现对象间的通信和请求任务的操作。消息传递是程序运行的基本处理活动。

当然,面向对象的程序设计并不是要抛弃结构化程序设计方法,而是站在比结构化程序更高、更抽象的层次上去解决问题。当它分解为低级代码模块时,仍需要结构化编程技巧,但是,它分解一个大问题为小问题时采取的思路与结构化方法是不同的。

1.3.2 面向对象的程序设计语言

早在 20 世纪 60 年代,就出现了最早的面向对象的程序设计语言 Simula67 语言,具有了类和对象的概念,随后又推出了纯面向对象的程序设计语言。如 80 年代美国 Xerox Palo Alto 研究中心推出的 Smalltalk,它完整地体现并进一步丰富了面向对象的概念,它还开发了配套的工具环境,为其最终实用化奠定了基础。由于当时人们已经接受并广泛应用了结构化设计理论,不能一下子完全接受面向对象程序设计的所有思想等原因,这些语言没有能够广泛地流行起来。后来人们开始对流行的语言进行面向对象的扩充,曾经推出过许多版本的版本,而主要成功的代表是在当时流行的程序设计语言 C 语言的基础上开发的 C++ 语言。C++ 是一门高效实用的混合型程序设计语言,它是由 AT&T 公司的贝尔实验室的 Bjarne Stoustrup 博士开发的,它最初的设计目标是,支持面向对象编程技术;支持抽象形态的类;更完善的 C 语言。C++ 语言包括两部分:一是 C++ 基础部分,以 C 语言为核心;另一部分是 C++ 面向对象特性部分,是 C++ 对 C 语言的扩充部分。这样它既支持面向对象的程序设计方法,又支持结构化程序设计方法。由于广泛的应用基础和丰富的开发环境的