

高等学校计算机基础教育
改革与实践系列教材

Visual Basic 程序设计教程

主 编 亢临生
副主编 王金虹



高等教育出版社
HIGHER EDUCATION PRESS

高等学校计算机基础教育改革与实践系列教材

Visual Basic 程序设计教程

Visual Basic Chengxu Sheji Jiaocheng

主 编 亢临生

副主编 王金虹



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容提要

本书以 Visual Basic 6.0 为蓝本,介绍了程序设计基础、VB 语言基础、顺序结构、选择结构、循环结构、数组、过程与函数、鼠标与键盘事件、图形操作、界面设计和文件操作。

全书针对 VB 语言程序设计的初学者,通过精心设计的案例,结合实际,深入浅出地介绍应掌握的知识点,同时注重对初学者“计算思维”方式的培养,引导初学者如何从以计算机语言为核心的学习模式转变为以问题为核心的学习模式,为此,把标准函数和常用控件内容分散到了相关性较强的章节中。

本书有配套的实验教材。为适应不同的群体,该实验教材以从易到难、从模仿到自己动手的原则,把实验题目分为验证性实验、设计性试验和提高性试验三类。同时,为方便教学,本书配有教学课件和相关教学素材,可以到中国高校计算机课程网上下载,网址为 <http://computer.cncourse.com>。

本书适合作为高等学校计算机程序设计类课程的教材,也可作为 Visual Basic 程序设计培训及计算机等级考试的辅导用书。

图书在版编目(CIP)数据

Visual Basic 程序设计教程/亢临生主编. —北京:高等教育出版社,2010.8

(高等学校计算机基础教育改革与实践系列教材/陈立潮主编)

ISBN 978-7-04-030154-0

I. ① V… II. ① 亢… III. ① Basic 语言—程序设计—高等学校—教材 IV. ① TP312

中国版本图书馆 CIP 数据核字(2010)第 144423 号

策划编辑	饶卉萍	责任编辑	许 可	封面设计	张志奇	责任绘图	尹 莉
版式设计	张 岚	责任校对	杨雪莲	责任印制	陈伟光		

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100120

经 销 蓝色畅想图书发行有限公司
印 刷 涿州市京南印刷厂

开 本 787×1092 1/16
印 张 21.5
字 数 520 000

购书热线 010-58581118
咨询电话 400-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landaco.com>
<http://www.landaco.com.cn>
畅想教育 <http://www.widedu.com>

版 次 2010 年 8 月第 1 版
印 次 2010 年 8 月第 1 次印刷
定 价 29.30 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 30154-00

高等学校计算机基础教育改革与实践系列教材

编审委员会

主 任：陈立潮

副主任：段 富

委 员：

亢临生 赵山林 贾 伟 李祥生

王建国 朱红康 张 森 张英俊

序 言

教育部高等学校计算机基础课程教学指导委员会在《高等学校计算机基础教学发展战略研究报告暨计算机基础课程教学基本要求》(以下简称《基本要求》)中指出,党的十七大提出了我国要从改造传统工业入手,走工业与信息技术相融合的新型工业化道路。为此,需要培养大批新一代“专业+信息”的工程技术人才。作为信息技术的核心,计算机基础教育的重要性被提到了空前的高度,计算机基础课程在高等学校确立了公共基础课的地位。在实施高等学校本科教学质量与教学改革的过程中,计算机基础的教学改革朝着高水平、应用化、规范化方向推进。

在完成教育部高等学校计算机基础课程教学指导委员会课题的过程中,我们组织了十余所高等学校计算机基础教学的负责人和一线教师,对这些高等学校中几千名本科生以问卷的形式对计算机基础教育现状进行了抽样调查,内容涉及大学生起始计算机技能基本情况、目前课程安排和课程效果评价3个方面,着重了解当代大学生对计算机基础教育的新需求及对计算机基础课程的意见与建议等。通过对这些调查问卷进行科学的分析,我们得到一些计算机基础课程课程体系改革的启示。在此基础上,按照《基本要求》的精神,结合计算机技术发展和应用的实际,以“知识-技能-能力”培养为目标,对计算机基础课程体系进行了重新的设计和调整,构建了“大学计算机基础+X门计算机应用课程”和“程序设计基础+X门计算机应用课程”两种“1+X”课程体系模式,形成了新型的计算机基础课程教学方案。

在以上课题研究的基础上,我们成立了“高等学校计算机基础教育改革与实践系列教材”编审委员会,希望能编写出一套适合于此教学方案的教材并建设相应的课程教学资源。

该系列教材以“面向应用、强化基础、注重融合”为原则,从面向应用的计算机硬件基础和软件基础两个角度入手,从融合专业技术的发展、社会对现代人才知识结构的要求出发,按照两种“1+X”的课程设置方案,选择了5门比较基础且通用的计算机基础课程来组织编写。该系列教材具有以下特点。

1. 从实践中来,到实践中去。所有教学内容均从应用问题出发,以引例、实例和案例作为背景,提出每章的教学内容与教学目标,使学生对学习什么知识、为什么要学这些知识有一个概括的认识,并通过解决问题使所学基础知识得到强化。所有引例、实例和案例都具有代表性,能激发学习的积极性,达到学以致用目的。

2. 内容新颖,知识结构更加合理。所有教学内容进一步体现了新版《基本要求》的精神,并

在此基础上,结合多年来教学改革与实践经验及地方经济结构和行业的需要,融合相关专业知识,适当地增加了部分内容。同时突破了传统的知识结构与教学模式,对相关内容的知识结构顺序做了调整,更利于学生对计算机基础知识的理解和掌握。

3. 进一步体现了计算机基础教学的目的和意义。计算机基础课程到底应该学哪些知识?计算机技术的发展水平和社会对计算机知识的需求是什么?计算机基础课程教学的内容怎样适应这种要求?所有这些是衡量计算机基础课程教学成败的关键。要使我们的教学紧跟时代的步伐,就要在不同的时期调整我们的教学内容。本系列教材在这方面做了大量的调研,对教学内容进行了适当的选择,进一步体现了“大学计算机文化基础”之后的新的知识与内容,进一步满足了社会对现代人才面向应用的计算机技能的基本要求。

为了保证教材的编写质量,编委会对本系列教材的编写过程进行了全程把关,各书的主编和编委由来自各高等学校的计算机基础教学负责人或骨干教师担任,他们都有丰富的教学实践和教材编写经验。可以说本系列教材综合了各高等学校计算机基础教学改革与实践的经验和成果,是集体智慧的结晶。

希望各高等学校在使用该系列教材的过程中能够提供有益的帮助和意见,以促进相关课题研究工作的不断深入。

高等学校计算机基础教育改革与实践系列教材

编审委员会

2010年5月

前 言

目前, Visual Basic 是一种功能强、应用较广的编程语言, 尤其适合于初学者学习。

根据非计算机专业人员的特点和计算机基础教育的目的和要求, 计算机基础教育应该注重“计算思维”方式的培养, 提高利用计算机解决问题的能力, 因此本教材的主导思想是: 尽可能地把传统的以计算机语言为核心的教学模式逐步转变为以问题为核心的教学模式, 即把“语言有什么功能, 如何应用”转变为“问题有何特性, 如何用计算机解决”。

为达到上述之目的, 本教材注重“传统思维”向“计算思维”方式转变的“引导”作用, 主要体现在三个方面: 一是选择有多种数学解决方法的问题, 引导初学者从中选择更适合用计算机解决问题的方法; 二是选择只有一种数学方法, 但可以有多种计算机解决方案, 简要分析各种方案的优缺点, 从中选择最佳方案; 三是注重培养学生调试程序的能力, 当程序出现错误时, 大多数情况都与“传统思维”有关, 这时, 首先给出“就事论事”的解决方法, 然后逐步过渡到利用“计算思维”的解决方案上。

本书以 Visual Basic 6.0 为蓝本, 介绍了 Visual Basic 主要内容: 程序设计基础、VB 语言基础、顺序结构、选择结构、循环结构、数组、过程与函数、鼠标与键盘事件、图形操作、界面设计和文件操作。为适应上述的教学目的, 本书对传统的内容体系做了一定的调整, 例如标准函数和常用控件分散到了与之关联比较密切的相关章节中。

本书有与之配套的实验教材和教学课件。为适应不同的群体, 实验教材以从易到难、从模仿到自己动手的原则, 把实验题目分为验证性实验、设计性试验和提高性试验三类。

本书由亢临生主编, 王金虹副主编, 亢临生、王金虹、萧宝玮、王雪婴、庄靖、李丹、马斌、黄煜、焦莉娟也参与了本书的编写工作。

本教材、实验教材和教学课件在山西省计算机学会和高等教育出版社的指导和帮助下完成, 在此对有关专家和老

编者
2010年4月

目 录

第1章 程序设计基础	1
1.1 程序设计语言概述	1
1.1.1 程序设计语言及发展	1
1.1.2 程序设计语言运行方式	2
1.1.3 程序设计方法发展历程	2
1.1.4 面向对象的程序设计语言概述	3
1.1.5 算法及其描述	4
1.2 Visual Basic 程序设计语言简介	5
1.2.1 Visual Basic 的发展及特点	5
1.2.2 VB 集成开发环境	7
1.2.3 VB 语句及书写规则	11
1.3 创建 VB 应用程序过程	12
1.3.1 引例 两位数加、减法自动 出题与测试	12
1.3.2 VB 中的对象及属性、事件 与方法	15
1.3.3 基本控件与界面设计:窗体、标签、 文本框、按钮	16
1.3.4 程序设计基本步骤	25
1.3.5 工程组成与管理	29
1.3.6 程序的调试	32
小结	36
习题	36
第2章 VB 语言基础	38
2.1 引例	38
2.2 数据及类型	39
2.2.1 标准数据类型	39
2.2.2 自定义数据类型	41
2.3 常量和变量	41
2.3.1 常量	42
2.3.2 变量	43
2.4 运算符与表达式	44
2.4.1 算术运算符及其优先级	44
2.4.2 字符串运算符	45
2.4.3 关系运算符	45
2.4.4 逻辑运算符及其优先级	46
2.4.5 表达式	46
2.5 标准函数及应用	47
2.5.1 函数	47
2.5.2 函数的使用	51
2.6 引例解决方案	51
2.7 案例(汉字编码问题)	52
2.7.1 问题描述	52
2.7.2 解题思路	52
2.7.3 程序设计	52
小结	53
习题	53
第3章 顺序结构	56
3.1 引例	56
3.2 赋值语句	56
3.2.1 格式	57
3.2.2 功能	57
3.2.3 说明	57
3.3 数据的输入输出	58
3.3.1 InputBox 输入函数	58
3.3.2 Print 方法	58
3.3.3 MsgBox 函数	59
3.3.4 用控件实现数据的输入与 输出	61
3.4 引例 3.1 解决方案	62

3.4.1 需要考虑的问题	62	5.3.1 While 循环语句	94
3.4.2 解决方案	62	5.3.2 引例的解决方案	95
3.5 案例	65	5.4 Do 循环	96
小结	65	5.4.1 Do 语句	96
习题	66	5.4.2 引例 Do 解决方案	97
第4章 选择结构	70	5.5 循环的嵌套	98
4.1 引例	70	5.5.1 嵌套循环注意事项	98
4.2 If 语句	71	5.5.2 嵌套循环举例	100
4.2.1 单分支 If 语句及引例的 解决方案	71	5.6 案例	102
4.2.2 双分支 If 语句及引例的 解决方案	73	5.6.1 密码问题	102
4.2.3 多分支 If 语句及引例的 解决方案	74	5.6.2 猴子吃桃子问题	103
4.3 Select Case 语句及引例的 解决方案	75	小结	103
4.3.1 Select Case 语句	75	习题	104
4.3.2 引例的解决方案	77	第6章 数组	109
4.4 条件函数	77	6.1 一维数组	109
4.4.1 IIF 函数	77	6.1.1 引例 6-1	109
4.4.2 Choose 函数	78	6.1.2 一维数组的声明	110
4.5 相关控件	78	6.1.3 引例 6-1 的计算机方案	110
4.5.1 单选按钮	79	6.1.4 一维数组的其他应用	112
4.5.2 框架	81	6.1.5 应用案例	114
4.5.3 复选框	82	6.2 二维数组	115
4.5.4 滚动条	84	6.2.1 引例 6-2	115
4.5.5 定时器	85	6.2.2 二维数组的声明	115
4.6 案例	86	6.2.3 二维数组的使用	116
小结	88	6.2.4 引例 6-2 的静态数组 解决方案	117
习题	88	6.3 动态数组	118
第5章 循环结构	91	6.3.1 引例 6-3	118
5.1 引例	91	6.3.2 动态数组的声明	118
5.2 For 循环	92	6.3.3 动态数组的使用	120
5.2.1 For 循环语句	92	6.3.4 引例 6-3 的动态数组 解决方案	123
5.2.2 引例的解决方案	93	6.4 使用控件(列表框、组合框)	124
5.3 While 循环	94	6.4.1 列表框	124
		6.4.2 组合框	129
		6.5 控件数组	131

6.5.1 控件数组的基本概念	131	8.1.4 案例2:移动控件位置	173
6.5.2 控件数组的建立	132	8.2 鼠标的指针形状	173
6.5.3 控件数组的使用	133	8.2.1 MousePointer 属性	173
6.6 案例	135	8.2.2 自定义鼠标指针	174
小结	140	8.2.3 案例3:等待时鼠标指针的 图标变化	175
习题	140	8.3 键盘事件	175
第7章 过程与函数	144	8.3.1 KeyPress 事件	176
7.1 引例 7-1	144	8.3.2 KeyDown 和 KeyUp 事件	177
7.2 过程与函数概述	144	8.4 案例	178
7.3 过程的定义与调用	145	小结	181
7.3.1 过程的定义	145	习题	181
7.3.2 过程的调用	148	第9章 图形操作	183
7.3.3 引例 7-1 的解决方案	151	9.1 引例	183
7.4 函数	152	9.2 图形操作基础	184
7.4.1 引例 7-2	152	9.2.1 坐标系统	184
7.4.2 函数的定义	152	9.2.2 自定义坐标系统	184
7.4.3 函数的调用	154	9.2.3 图形层	185
7.4.4 引例 7-2 的解决方案	155	9.3 绘图属性	185
7.5 参数传递	155	9.3.1 当前坐标	185
7.5.1 形式参数与实际参数	155	9.3.2 线宽与线型	186
7.5.2 传址与传值	156	9.3.3 填充与色彩	186
7.5.3 数组参数的传递	159	9.4 图形控件	190
7.6 过程的嵌套调用	160	9.4.1 图形框	190
7.7 变量的作用域	162	9.4.2 图像框	191
7.7.1 局部变量	162	9.4.3 Line 控件	191
7.7.2 模块级变量	162	9.4.4 Shape 控件	192
7.7.3 全局变量	163	9.5 图形方法	193
7.7.4 静态变量	164	9.5.1 Line 方法	193
7.8 案例	164	9.5.2 Circle 方法	194
小结	166	9.5.3 Pset 方法	194
习题	167	9.5.4 Point 方法	195
第8章 鼠标与键盘事件	171	9.5.5 几何图形的绘制	195
8.1 鼠标事件	171	9.6 引例的解决方案	197
8.1.1 鼠标过程事件的格式	171	9.6.1 系统坐标解决方案	197
8.1.2 参数说明	171	9.6.2 自定义坐标解决方案	198
8.1.3 案例1:画圆	172		

9.7 应用案例	198	11.3.5 引例的顺序文件解决方案	237
小结	201	11.4 随机文件	238
习题	202	11.4.1 打开随机文件	238
第10章 界面设计	204	11.4.2 随机文件的写操作	238
10.1 菜单设计	204	11.4.3 随机文件的读操作	239
10.1.1 打开菜单编辑器	205	11.4.4 案例的随机文件解决 方案	240
10.1.2 创建菜单	206	11.5 文件系统控件	241
10.1.3 案例1:下拉式菜单	207	11.5.1 驱动器列表框	241
10.1.4 案例2:弹出式菜单	211	11.5.2 目录列表框	241
10.2 通用对话框设计	212	11.5.3 文件列表框	241
10.2.1 设置颜色案例	212	11.6 案例	243
10.2.2 通用对话框的 Action 属性 和 Show 方法	213	小结	247
10.2.3 “打开”/“另存为”对话框	215	习题	248
10.2.4 “颜色”对话框	217	第12章 数据库操作	250
10.2.5 “字体”对话框	218	12.1 引例	250
10.2.6 “打印”对话框	220	12.2 数据库概述	251
10.2.7 “帮助”对话框	221	12.2.1 关系型数据库	252
10.3 多文档界面设计	223	12.2.2 基本的 SQL 查询	253
10.3.1 案例1:建立简单的多文档 窗口界面	223	12.3 数据库操作及引例的 实现方案	254
10.3.2 案例2:创建 MDI 多文档 应用程序	224	12.3.1 引例的数据管理器(菜单)的 操作方案	254
小结	230	12.3.2 引例 ADO 方式的实现	260
习题	231	小结	271
第11章 文件操作	233	习题	272
11.1 引例	233	第13章 综合应用	275
11.2 文件概述	234	13.1 四则算术运算自动练习	275
11.2.1 文件结构	234	13.1.1 问题	275
11.2.2 文件分类	234	13.1.2 程序设计	276
11.2.3 文件的读写过程	235	13.2 二分法查找	284
11.3 顺序文件	235	13.2.1 问题	284
11.3.1 顺序文件的打开和关闭	235	13.2.2 算法流程图	284
11.3.2 顺序文件的写入操作	235	13.2.3 程序设计	285
11.3.3 顺序文件的读操作	236	13.3 学籍管理	289
11.3.4 文件操作函数和命令	236	13.3.1 系统功能	289

13.3.2	数据库字典	289
13.3.3	开发说明	291
13.3.4	界面及对应的程序模块	291

附录 VB 常用标准函数	323
--------------------	-----

参考文献	326
------------	-----

程序设计基础

第 1 章

1.1 程序设计语言概述

从 1946 年世界上第一台计算机问世至今短短的 60 多年中,计算机硬件技术获得了飞速发展。与此相适应,软件开发工具经历了机器语言、汇编语言到高级语言的发展;程序设计方法也经历了早期面向计算机的程序设计、面向过程的程序设计到面向对象的程序设计等几个发展阶段。由此,在短短几十年的时间内,人类创造了数以百计的各种各样的程序设计语言,加速了计算机在各个领域更为广泛的应用。

1.1.1 程序设计语言及发展

计算机之所以能自动进行计算,是因为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特定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。

在短短的 60 多年间,程序设计语言的发展从低级到高级,经历了机器语言、汇编语言、高级语言三个阶段。

1. 机器语言

机器语言是计算机系统能够直接识别和执行的二进制指令(也称机器指令)的集合,早期的计算机程序都是直接使用机器语言编写的,这种语言使用 0、1 代码,指令难以记忆,人们编写或阅读程序都十分困难,并且容易出错,目前已不被人们使用。

2. 汇编语言

为了解决机器语言的困难,人们采用了以助记符代替 0、1 代码的汇编语言编写程序,然后通过一个预先设计好的“汇编程序”,将汇编语言程序一对一地翻译成机器语言程序,再让计算机系统执行。汇编语言的执行效率高,目前主要应用于计算机自动控制领域,但它同时具有机器语言的缺陷,仍然不易理解,也不便于交流,因此,在其他领域使用较少。

3. 高级语言

机器语言和汇编语言是面向机器的语言,高级语言则更接近于人类的自然语言和数学语言。使用高级语言编程,一般不必了解计算机的指令系统和硬件结构,只需掌握解题方法和高级语言的语法规则,就可以编写应用程序。所以,对非计算机专业人员来说更容易学习和掌握。由于高级语言克服了机器语言的缺点,所以程序易读、易维护。高级语言种类繁多,不同的语言各有不同的特点,适合的应用领域也不同,因而开发应用程序时,首先要选择恰当的语言。常见的几种高级语言有 FORTRAN 语言、BASIC 语言、Pascal 语言、C 语言、Java 语言等。高级语言普遍具有容易学习、使用方便、通用性好、移植性好等特点,便于各类人员学习、掌握和应用。

1.1.2 程序设计语言运行方式

高级语言编写的源程序必须经过“翻译程序”翻译成计算机所能识别的 0、1 代码程序才能让计算机执行。这个“翻译程序”一般称为语言处理程序。其翻译方式一般有编译方式和解释方式两种。

1. 编译方式

编译方式是将源程序先由编译器整体翻译为功能等价的机器语言目标程序,然后由连接程序将目标程序和系统资源(如函数库、系统过程等)连接在一起,形成一个可执行程序。这个过程类似于外文文献的书面翻译过程。通过编译方式得到的可执行程序可以重复执行,效率很高。一般系统程序、用户已调试好的程序大多采用编译方式生成可执行程序,供用户使用。

2. 解释方式

解释方式是将源程序通过“解释程序”逐句翻译成可执行的指令代码,翻译一句执行一句,如果句子有错,则会立即中断程序的运行并提示出错信息,这时用户可以观察一些中间计算结果,分析错误原因并立即对程序进行修改。这个过程类似于同声翻译。解释方式不产生目标代码程序,每次执行都要重复翻译,因此程序运行速度较慢,效率较低,但解释方式对于调试程序很方便。Visual Basic 语言既保留了 BASIC 语言的解释方式的特点,同时也可以使用编译方式运行。

1.1.3 程序设计方法发展历程

伴随着计算机系统的发展,程序设计方法也在不断发展前进,其主要经过了三个发展阶段:面向计算机的程序设计、面向过程的程序设计和面向对象的程序设计。

1. 面向计算机的程序设计

20 世纪 50 年代,程序设计人员关注的是如何使程序尽可能地被计算机接受并按指令正确地执行,所以采用由计算机可以直接识别的二进制指令组成的机器语言或使用了助记符的汇编语言来编写程序,但这两种程序设计语言与人类的自然语言之间存在着较大的鸿沟,与人类的思想也相差甚远,而且它的抽象层次低,设计人员需要考虑大量的机器细节,因此软件开发难度大、周期长,而且开发出的软件功能也相对简单。在这一时期,计算机的应用仅限于

科学计算。

2. 面向过程的程序设计

面向过程的程序设计也称为结构化程序设计,是由荷兰的计算机科学家 Dijkstra 于 1969 年首先提出的,其基本思想是:自顶向下、逐步求精。其程序结构按功能划分为若干个基本模块(基本程序),这些模块形成一个树状结构,各模块之间的关系尽可能简单,在功能上相对独立:每一个模块内部均由顺序、条件、循环三种基本结构组成,其模块化实现的具体方法是使用子程序。结构化程序设计由于采用了模块分化与功能分解,自顶向下、分而治之的方法,因而可将一个较复杂的问题分解为若干个子问题,各子问题分别由不同的人员解决,从而提高了应用程序开发的效率,并且便于程序的调试,有利于软件的开发和维护。

3. 面向对象的程序设计

面向对象的程序设计是 20 世纪 80 年代初提出的,旨在使计算机问题的求解过程更接近于人的思维活动,其基本思想是:注重对象、抽象成类。在程序设计中,将客观世界中的事物看成对象。对象是由数据及对数据的操作构成的一个不可分离的整体。对同类型的对象抽象出其共性,形成类。类中大多数数据,只能用本类的方法进行处理。类通过一个简单的外部接口与外界发生联系,对象与对象之间通过消息传递信息。面向对象的程序设计由于数据与操作封装在对象这个统一体中,使得设计人员在编程过程中能够将数据与操作联系在一起,从而便于程序的修改和调试。并且,由于类的继承性使得编程人员可以在可视化的环境中进行组件化的编程,从而将设计人员从单调、重复的编程过程中解放出来,去进行创造性的总体设计工作。

面向过程的程序设计突出过程,即如何做(How to do),它强调代码的功能是如何得以完成的。面向对象的程序设计突出真实世界及其组成对象,即做什么(What to do),它将大量的工作交由相应的对象完成,设计人员在应用程序中只需说明对象所需完成的任务。

1.1.4 面向对象的程序设计语言概述

面向对象的程序设计方法认为现实世界是由对象组成的,解决问题时,首先要确定问题是由哪些对象组成的。所以面向对象的程序是由一个个对象组成,对象之间通过消息相互作用。对象代表着正在创建的系统中的—个实体,每个对象都包含两个因素:数据和与数据有关的操作,其中数据描述对象的属性特征,操作是附属于对象的行为的,因此对象是一个属性和操作(方法)的封装体。

在面向对象的程序设计语言中,每个对象都属于一个类。类代表了某一批对象的共性和特征。类是对象的抽象,是用来定义对象的一种抽象数据类型,或者说它是产生对象的模型,而对象是类的具体实例。正如蛋糕模型与蛋糕之间,蛋糕模型就是一个类,它确定了每一块蛋糕的特征,如大小和形状,而用此模型制作的蛋糕就是一个对象。

面向对象程序设计具有以下优点:

- ① 使软件开发过程符合人们的思维方法,便于分析和解决复杂而多变的问题。
- ② 使软件的维护和功能的增减易于实现。
- ③ 使程序代码可重用,从而提高了软件的开发效率。
- ④ 与可视化技术相结合,使用户界面图形化、更美观、更友好。

目前常用的面向对象程序设计语言有:VC++、C#、Visual Basic(简称 VB)、VB.NET、Java、

VJ++、PowerBuilder、Delphi、C++ Builder 等。

1.1.5 算法及其描述

1. 算法及其特征

所谓算法,简单地讲,就是计算机解决问题的具体步骤。

例如:求 $1+2+3+\cdots+N=?$,其算法可描述如下。

步骤1:给出 N 的值。

步骤2:设两个变量 R 、 I , I 为大于等于零的整数, R 表示 $1\sim I$ 的所有整数和,令 $I=0$, $R=0$ 。

步骤3: $I=I+1$, $R=R+I$ 。

步骤4:判断 $I\leq N$ 是否成立。如果成立返回步骤3,如果不成立则执行下一个步骤。

步骤5:输出 R ,即 $1\sim N$ 的所有整数和。

一个正确的算法,应具备的基本特征有:

- ① 算法的有穷性。应在有限步骤内结束。
- ② 算法的确定性。只要初始条件相同,就能得到相同的、确定的结果。
- ③ 算法的有效性。算法中每一步操作必须是可执行的。
- ④ 有零个或多个输入。一个算法既可以有输入数据,也可以没有输入数据。
- ⑤ 至少有一个输出。算法的目的就是求解问题,求解的结果必须向用户输出。

对于同一个问题,可能有多种不同的算法,也就是存在一题多解的情况,这就要求在众多的算法中选择较好的一个算法。

2. 算法的描述

要想对算法进行描述,必须使用相应的工具。目前,计算机程序设计中常用的描述工具有自然语言、流程图、N-S图等。

(1) 自然语言

自然语言是人类在日常生活中进行交流的语言,也可用于描述问题求解的算法。但用自然语言描述算法,存在文字冗长、有二义性、表达不确切等缺点,所以仅限于一些简单问题的处理过程的描述。

(2) 流程图

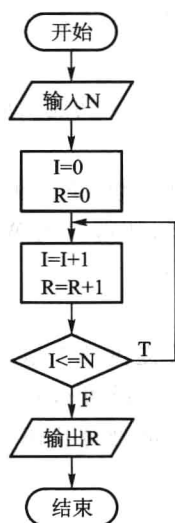
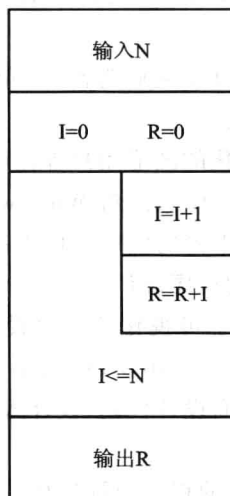
流程图是一种借助于图形符号(如图1-1所示)来描述算法过程的图形方法,具有直观、形象、易于理解等特点,应用广泛。求 $1+2+\cdots+N=?$ 的流程图如图1-2所示。



图 1-1 常用流程图符号

(3) N-S图

N-S图是流程图的发展,它去掉了流程图中的流程线,全部算法都表示在一个矩形框内,图1-3所示为求 $1+2+\cdots+N=?$ 的N-S图。

图 1-2 求 $1 + 2 + \dots + N = ?$ 的流程图图 1-3 求 $1 + 2 + \dots + N = ?$ 的 N-S 图

1.2 Visual Basic 程序设计语言简介

Visual Basic 是 Microsoft 公司为开发 Windows 应用程序而推出的一种可视化的快速编程工具, 它以其简单易学、语法简洁、功能强大的特点而深受广大编程人员的青睐。

1.2.1 Visual Basic 的发展及特点

1. Visual Basic 的发展

Visual Basic 是 Microsoft 公司于 1991 年在原 DOS 操作系统平台上广为流行的 BASIC 语言的基础上开发推出的新一代面向对象的程序设计语言, 版本号为 1.0, 并获得巨大成功。接着分别于 1992 年秋天、1993 年 4 月推出 2.0 版和 3.0 版。为适应 Windows 95 操作系统 32 位环境需要, Microsoft 公司于 1995 年 8 月推出了 4.0 版本。随着 Internet 的迅速发展, 为适应基于 Internet 的应用开发的强劲需要, 增强 VB 对 Internet 的支持和开发能力, 1997 年 Microsoft 公司推出了 5.0 版本, 该版本在 4.0 的基础上有了质的飞跃, 首先引入了本机代码编译器, 使用户开发的应用程序能真正编译成标准的 EXE 文件, 大大提高了 VB 应用程序的运行速度。1998 年秋季, 随着 Windows 98 的发行, Microsoft 又推出了功能更强、更完善的 VB 6.0, 该版本在创建自定义控件、对数据库的访问以及对 Internet 的访问等方面得到了进一步的增强、完善和提高。Windows 2000 推出后, Microsoft 公司又成功推出了功能更强大的 VB.NET。从 5.0 版本开始, VB 根据用户对象的不同, 分学习版、专业版和企业版进行销售。

① 学习版: 是 VB 的基础版本, 主要针对初学者学习和使用。包括所有的内部控件以及网络、数据绑定等控件。

② 专业版: 为专业编程人员提供了一整套用于软件开发的功能完备的工具。除包括学习版的全部功能外, 还包括 ActiveX、Internet 等控件开发工具及动态 HTML 页面设计等高级特性。