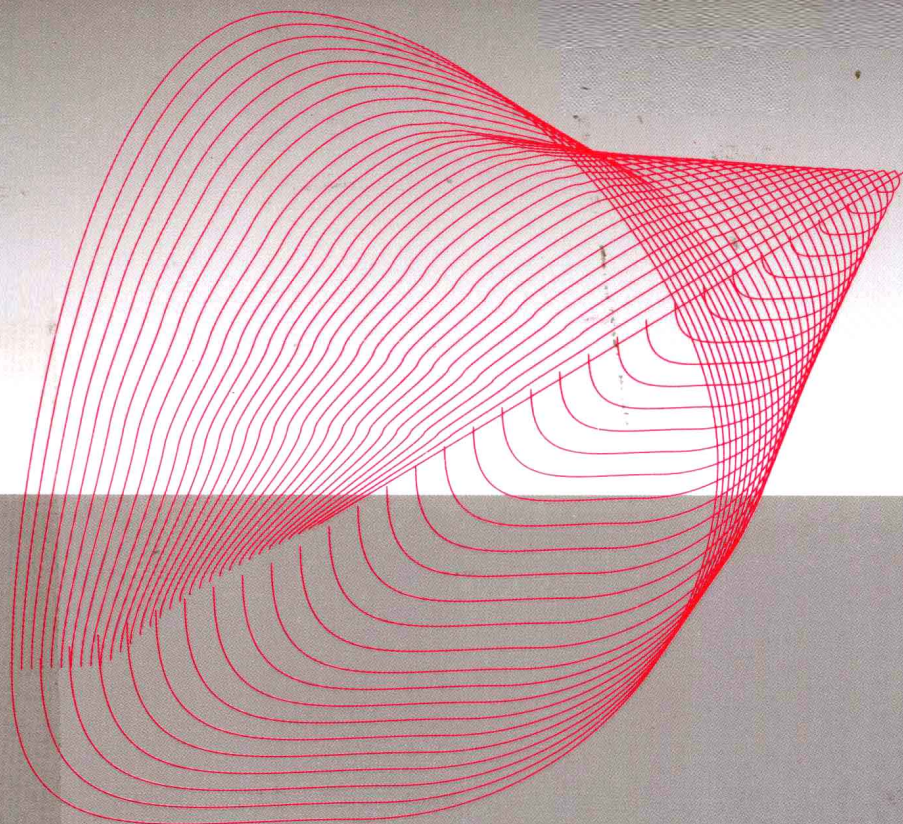


21 世纪高等学校计算机教育实用规划教材

# C语言程序设计基础教程

于延张军 主编  
么丽颖 张必英 常亮 副主编



清华大学出版社

21 世纪高等学校计算机教育实用规划教材

# C语言程序设计基础教程

于延 张军 主编  
么丽颖 张必英 常亮 副主编

清华大学出版社  
北京

## 内 容 简 介

本书从培养应用型人才的角度出发,系统地介绍了C语言编程的基本知识和程序设计的基本方法,内容包括C语言概述、基本数据类型、运算符和表达式、控制结构、函数、作用域和存储类别、数组、指针、字符串、结构体和共用体、编译预处理、文件。各章配有大量例题和练习。

本书可作为各类高等院校非计算机专业计算机公共基础课程的教学用书,也可作为计算机等级考试和自学参考用书。

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。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

## 图书在版编目(CIP)数据

C语言程序设计基础教程/于延,张军主编. —北京:清华大学出版社,2011.1  
(21世纪高等学校计算机教育实用规划教材)

ISBN 978-7-302-23507-1

I. ①C… II. ①于… ②张… III. ①C语言—程序设计—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2010)第157143号

责任编辑:付弘宇 张为民

责任校对:梁毅

责任印制:李红英

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954,jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015,zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260 印 张:19 字 数:470千字

版 次:2011年1月第1版 印 次:2011年1月第1次印刷

印 数:1~3000

定 价:29.50元

---

产品编号:037053-01

# 出版说明

---

随着我国高等教育规模的扩大以及产业结构调整的不断深入,社会对高层次应用型人才的需求将更加迫切。各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,合理调整和配置教育资源,在改革和改造传统学科专业的基础上,加强工程型和应用型学科专业建设,积极设置主要面向地方支柱产业、高新技术产业、服务业的工程型和应用型学科专业,积极为地方经济建设输送各类应用型人才。各高校加大了使用信息科学等现代科学技术提升、改造传统学科专业的力度,从而实现传统学科专业向工程型和应用型学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时,不断更新教学内容、改革课程体系,使工程型和应用型学科专业教育与经济建设相适应。计算机课程教学在从传统学科向工程型和应用型学科转变中起着至关重要的作用,工程型和应用型学科专业中的计算机课程设置、内容体系和教学手段及方法等也具有不同于传统学科的鲜明特点。

为了配合高校工程型和应用型学科专业的建设和发展,急需出版一批内容新、体系新、方法新、手段新的高水平计算机课程教材。目前,工程型和应用型学科专业计算机课程教材的建设工作仍滞后于教学改革的实践,如现有的计算机教材中有不少内容陈旧(依然用传统专业计算机教材代替工程型和应用型学科专业教材),重理论、轻实践,不能满足新的教学计划、课程设置的需要;一些课程的教材可供选择的品种太少;一些基础课的教材虽然品种较多,但低水平重复严重;有些教材内容庞杂,书越编越厚;专业课教材、教学辅助教材及教学参考书短缺,等等,都不利于学生能力的提高和素质的培养。为此,在教育部相关教学指导委员会专家的指导和帮助下,清华大学出版社组织出版本系列教材,以满足工程型和应用型学科专业计算机课程教学的需要。本系列教材在规划过程中体现了如下一些基本原则和特点。

(1) 面向工程型与应用型学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映基本理论和原理的综合应用,强调实践和应用环节。

(2) 反映教学需要,促进教学发展。教材规划以新的工程型和应用型专业目录为依据。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材建设仍然把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现工程型和应用型专业教学内容和课程体系改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材要配套,同一门课程可以有多种具有不同内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材,教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主编。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机教育实用规划教材编委会  
联系人:魏江江 weijj@tup.tsinghua.edu.cn

# 前 言

---

C 语言作为一种适用于开发系统软件及应用软件的计算机语言,已经成为计算机程序设计语言的主流语种。“C 语言程序设计”课程是高校计算机专业的专业基础课,也是很多非计算机专业理科学生的必修课。虽然目前功能最强、最受用户青睐的语言是 Java 和 C++,但学 Java 或 C++ 来入门是不妥的,因为 C++ 是在 C 语言的基础上开发的,且 Java 与 C 也有千丝万缕的联系,在 Java 环境下就可以直接用 C 语言程序,因而用 C 语言作为入门语言是最佳的选择。

编者多年从事程序设计课程的教学以及应用软件的开发,针对软件开发应用领域中程序设计的要求,在程序设计教学过程中,避免陷入程序设计语言繁杂的语法和格式,将主要精力集中在所要解决的实际问题上,从知识点以及具体问题出发,掌握如何通过程序设计来解决问题。

本书基本依据 ANSI C 标准编写,并参考教育部和一些高校计算机专业的“C 语言程序设计”教学大纲,对内容进行了精心的选择和组织,以满足各个专业学习及应用计算机的要求。例题部分强调对基本概念、原理和方法的运用能力,围绕基本算法,按照循序渐进、覆盖面广、重点突出的原则进行选题,目的是解难释疑、开阔思路。

本书努力体现以下特色:

(1) 本书主要是为大学计算机程序设计首选教学语言 C 编写的教材,同时兼顾广大计算机用户和自学爱好者,适合教学和自学。

(2) 既介绍 C 语言的使用,又介绍程序设计的基本方法和技巧。

(3) 重视良好的编程风格和习惯的培养。

(4) 力求做到科学性、实用性、通俗性三者的统一。编者希望本书通俗易懂的叙述方式能方便广大读者的学习。

(5) 在内容编排上不同于以往的其他教材,充分地考虑到初学者的实际情况,由浅入深,难点分散。

通过本书这种特色的教学模式,可以更好地实现高等教育人才培养的目标。不仅要让学生学习程序设计的基本概念和方法,掌握编程的技术,更重要的是培养学生针对生产实际分析问题和解决问题的能力,培养学生程序设计的能力和计算机操作能力。

本书由于延、张军任主编,么丽颖、张必英、常亮任副主编,各章编写分工如下:于延编写第 1、6、7、8、10 章和附录,张军编写第 3、12 章,么丽颖编写第 2、11 章,张必英编写第 5、9 章,

常亮编写第 4 章。全书由于延和张军统稿。

本书不但适合高等院校应用型本科专业使用,而且适合高职高专各类学校,还可作为计算机岗位培训的教学用书,或作为程序设计爱好者的学习参考书。与本书配套的习题集以及上机指导也将随后出版。本书的课件可以从清华大学出版社网站([www.tup.tsinghua.edu.cn](http://www.tup.tsinghua.edu.cn))下载,使用中如果出现问题请联系 [fuhy@tup.tsinghua.edu.cn](mailto:fuhy@tup.tsinghua.edu.cn)。

由于编者水平有限,书中不妥之处在所难免,敬请广大读者批评指正。

编 者

2010 年 5 月

# 目 录

---

<b>第 1 章 计算机程序设计导论</b> .....	1
1.1 计算机程序设计语言概述 .....	1
1.2 关于 C 语言 .....	3
1.2.1 C 语言的诞生和发展 .....	3
1.2.2 C 语言的特点 .....	3
1.2.3 怎样学好用好本书 .....	4
1.3 算法和程序 .....	6
1.3.1 算法举例 .....	6
1.3.2 算法的特性 .....	12
1.3.3 算法的表示 .....	13
1.4 结构化程序设计 .....	14
1.4.1 程序设计的三种基本结构 .....	14
1.4.2 结构化程序设计步骤 .....	16
习题 .....	16
<b>第 2 章 认识 C 语言</b> .....	18
2.1 一个最简单的 C 语言程序 .....	18
2.2 对 C 语言程序的进一步了解 .....	19
2.3 关键字和标识符 .....	25
2.3.1 关键字 .....	25
2.3.2 标识符 .....	26
习题 .....	26
<b>第 3 章 数据类型、运算符与表达式</b> .....	28
3.1 C 语言的基本数据类型 .....	28
3.2 常量 .....	29
3.2.1 整型常量 .....	29
3.2.2 实型常量 .....	30
3.2.3 字符型常量 .....	31
3.2.4 字符串常量 .....	33



3.2.5 符号常量 .....	33
3.3 变量 .....	34
3.3.1 变量的定义 .....	34
3.3.2 变量赋初值 .....	36
3.4 C 语言运算符 .....	37
3.4.1 运算符和表达式简介 .....	37
3.4.2 算术运算符和算术表达式 .....	41
3.4.3 赋值运算符和赋值表达式 .....	44
3.4.4 关系运算符和关系表达式 .....	46
3.4.5 逻辑运算符和逻辑表达式 .....	47
3.4.6 逗号运算符和逗号表达式 .....	49
3.4.7 条件运算符 .....	49
3.4.8 常用数学标准函数 .....	51
习题 .....	54
<b>第 4 章 顺序结构程序设计 .....</b>	<b>58</b>
4.1 C 语句介绍 .....	58
4.2 基本输入输出功能的实现 .....	59
4.2.1 字符输出函数 putchar() .....	60
4.2.2 字符输入函数 getchar() .....	61
4.2.3 字符输入函数 getche() 和 getch() .....	63
4.2.4 标准格式输出函数 printf() .....	64
4.2.5 格式输入函数 scanf() .....	68
4.3 顺序结构程序设计举例 .....	72
习题 .....	76
<b>第 5 章 选择结构程序设计 .....</b>	<b>81</b>
5.1 if 语句 .....	81
5.1.1 单分支 if 选择结构 .....	81
5.1.2 双分支 if 选择结构 .....	82
5.1.3 if 语句的嵌套 .....	89
5.1.4 多分支 if 选择结构 .....	92
5.2 switch 语句 .....	93
5.3 选择结构程序举例 .....	97
习题 .....	100
<b>第 6 章 循环结构程序设计 .....</b>	<b>104</b>
6.1 goto 语句及 goto 循环 .....	104
6.2 while 语句 .....	107

6.3	do-while 语句 .....	108
6.4	for 语句 .....	110
6.5	break 语句 .....	112
6.6	continue 语句 .....	113
6.7	循环结构的嵌套 .....	114
6.8	循环结构程序举例 .....	115
	习题 .....	127
<b>第 7 章</b>	<b>数组 .....</b>	<b>133</b>
7.1	认识数组 .....	133
7.2	一维数组 .....	134
7.2.1	一维数组的定义 .....	135
7.2.2	一维数组元素的引用 .....	136
7.2.3	一维数组的初始化 .....	136
7.2.4	一维数组程序举例 .....	137
7.3	二维数组 .....	139
7.3.1	二维数组的定义 .....	139
7.3.2	二维数组元素的引用 .....	139
7.3.3	二维数组元素的初始化 .....	139
7.3.4	二维数组程序举例 .....	140
7.4	字符数组 .....	142
7.4.1	字符串及字符数组 .....	142
7.4.2	字符数组的定义 .....	142
7.4.3	字符数组的初始化 .....	143
7.4.4	字符数组元素的输入 .....	143
7.4.5	字符数组元素的输出 .....	144
7.4.6	字符数组程序举例 .....	144
7.4.7	字符串函数 .....	148
	习题 .....	151
<b>第 8 章</b>	<b>函数 .....</b>	<b>156</b>
8.1	认识函数 .....	156
8.2	函数的定义和说明 .....	158
8.3	函数的调用 .....	162
8.3.1	函数调用的一般形式 .....	162
8.3.2	函数调用的方式 .....	164
8.3.3	return 语句与函数的返回值 .....	164
8.4	函数参数的传递方式 .....	168
8.4.1	值传递方式 .....	168

8.4.2 地址传递方式	169
8.5 数组作为函数的参数	170
8.5.1 数组元素作为函数参数	170
8.5.2 数组名作为函数参数	170
8.5.3 二维数组名作为函数参数	172
8.6 函数的嵌套调用	173
8.7 函数的递归调用	174
8.8 变量的作用域	178
8.8.1 局部变量	179
8.8.2 全局变量	180
8.9 变量的存储类型(生存期)	181
8.9.1 自动变量	182
8.9.2 静态变量	183
8.9.3 外部变量	185
8.9.4 寄存器变量	185
8.10 内部函数和外部函数	186
8.11 函数程序设计举例	187
习题	191
<b>第 9 章 编译预处理</b>	<b>197</b>
9.1 概述	197
9.2 宏定义	197
9.2.1 无参宏定义	197
9.2.2 带参宏定义	201
9.3 文件包含	204
9.4 条件编译	205
习题	207
<b>第 10 章 指针</b>	<b>210</b>
10.1 指针简介	210
10.2 指针变量的定义及使用	211
10.2.1 指针变量的定义	211
10.2.2 指针变量的赋值	211
10.2.3 指针运算符 & 和 *	212
10.2.4 指针变量的运算	213
10.3 指针与数组	217
10.3.1 指向一维数组的指针	217
10.3.2 指向二维数组的指针变量	219
10.3.3 指向字符串的指针	222

10.4	指针变量作函数参数 .....	224
10.5	使用字符串指针变量与字符数组的区别 .....	227
10.6	函数指针变量 .....	227
10.7	指针型函数 .....	229
10.8	指针数组 .....	230
10.9	指向指针的指针变量 .....	233
10.10	动态内存管理 .....	234
10.11	小结 .....	236
	习题 .....	237
<b>第 11 章 结构体与共用体 .....</b>		<b>243</b>
11.1	结构体 .....	243
11.1.1	结构体类型的定义 .....	243
11.1.2	结构体类型变量的说明 .....	244
11.1.3	结构体变量成员的引用 .....	245
11.1.4	结构体变量的赋值 .....	245
11.1.5	结构体变量的初始化 .....	246
11.2	结构体数组 .....	247
11.3	结构体指针变量 .....	249
11.3.1	结构体指针变量的说明和使用 .....	249
11.3.2	结构体数组指针变量 .....	250
11.3.3	结构体指针变量作函数参数 .....	251
11.4	共用体 .....	252
11.4.1	共用体类型的定义和共用体变量的说明 .....	253
11.4.2	共用体变量的说明 .....	253
11.4.3	共用体变量的赋值和使用 .....	254
11.5	小结 .....	255
	习题 .....	255
<b>第 12 章 文件 .....</b>		<b>261</b>
12.1	文件概述 .....	261
12.2	文件指针 .....	262
12.2.1	文件指针的定义 .....	263
12.2.2	文件打开函数 .....	263
12.2.3	文件关闭函数 .....	265
12.2.4	标准设备文件的打开与关闭 .....	266
12.3	文件的读写 .....	267
12.3.1	文件尾测试函数 .....	267
12.3.2	读/写字符函数 .....	268

12.3.3 字符串读/写函数.....	272
12.3.4 格式化读写函数 .....	275
12.3.5 数据块读写函数 .....	277
12.4 文件的随机读写 .....	279
12.5 文件操作出错检测函数 .....	280
12.6 C 语言库文件 .....	281
习题 .....	282
<b>附录 A ASCII 码 .....</b>	<b>286</b>
<b>附录 B 主函数的参数 .....</b>	<b>287</b>

随着计算机的不断普及和计算机应用的不断扩展,软件开发在当今是一个非常热门的专业。在目前以及未来,软件人才将是世界上缺口最大也是最抢手的人才。计算机技术已渗透到各个行业、各个角落,计算机软件在每个行业、每个领域和每个部门中都发挥重要的作用。而目前在我国,计算机软件的应用还仅仅局限在“使用软件”的范围内,在很长一个时期内对计算机软件产品的需求和计算机软件人才的需求仍是非常大的。特别需要指出的是,在计算机软件人才中,复合型、交叉型的软件人才奇缺。事实证明,一个纯计算机专业的毕业生,对其他专业往往涉猎很少,这样他对于一些专业性较强的软件往往不得要领,需要和其他专业人才合作才可以完成软件设计,而且如果双方的沟通和理解不是很好,设计出的软件在功能设计和可扩展方面都会有很大局限。所以,任何一个专业的人才群体中都应该而且也需要有一定比例的、掌握计算机软件设计技术的复合型人才,这样才能更好地利用计算机技术为本专业的研究服务。

本章介绍程序设计、算法、流程图等有关知识。

## 1.1 计算机程序设计语言概述

在软件开发的过程中,编程语言的选择是很关键的。编程语言的优良特性加上良好的编程风格,极大地影响着软件开发的进程,对确保软件的可靠性、可读性、可测试性、可维护性以及可重用性等起着重大的作用。

计算机程序设计语言的发展大致经历了机器语言、汇编语言、高级语言(面向过程的程序设计语言)以及面向对象的程序设计语言四个阶段。

### (1) 机器语言

机器语言是最底层的计算机语言,其指令和数据都是由二进制代码(由 0 和 1 组成)直接组合而成。用机器语言编写的程序,计算机硬件可以直接识别。对于不同的计算机硬件(主要是微处理器),其机器语言是不同的。因此,针对一种计算机所编写的机器语言程序不能在另一种计算机上运行。由于机器语言程序是直接针对计算机硬件的,因此它的执行效率比较高,能充分发挥计算机的速度性能。但是,用机器语言编写程序的难度比较大,容易出错,而且程序的直观性比较差,也不容易移植。

### (2) 汇编语言

为了便于理解与记忆,人们采用能帮助记忆的英文缩写符号(称为指令助记符)来代替机器语言指令代码中的操作码,用地址符号来代替地址码。用指令助记符及地址符号书写的指令称为汇编指令(也称符号指令),而用汇编指令编写的程序称为汇编语言源程序。汇

编语言又称符号语言。

汇编语言、机器语言与机器(计算机硬件系统)一般是一一对应的。因此,汇编语言也是与具体使用的计算机有关的。由于汇编语言采用了助记符,因此它比机器语言直观,容易理解和记忆。用汇编语言编写的程序也比机器语言程序易读、易检查、易修改。但是,计算机不能直接识别用汇编语言编写的程序,必须由一种专门的翻译程序将汇编语言源程序翻译成机器语言程序后,计算机才能识别并执行。这种翻译的过程称为“汇编”,负责翻译的程序称为汇编程序。

### (3) 高级语言(面向过程的程序设计语言)

机器语言和汇编语言都是面向机器的语言,一般称为低级语言。低级语言对机器的依赖性大,用它们开发的程序通用性很差。

随着计算机技术的发展以及计算机应用领域的不断扩大,计算机用户的队伍也在不断壮大。为了使广大计算机用户也能胜任程序的开发工作,从 20 世纪 50 年代中期开始逐步发展了面向问题的程序设计语言,称为高级语言。高级语言与具体的计算机硬件无关,其表达方式接近于被描述的问题,易于接受和掌握。用高级语言编写程序要比低级语言容易得多,并大大简化了程序的编制和调试过程,使编程效率得到大幅度的提高。高级语言的显著特点是不依赖于计算机硬件,通用性和可移植性好。

目前,计算机高级语言已有上百种之多,得到广泛应用的有十几种,如 BASIC、FORTRAN、Pascal、C、COBOL、dBASE、FoxBASE 等。这些高级语言也可以称为面向过程的程序设计语言。它的主要特征是程序由过程定义和过程调用组成,即:程序=过程定义+过程调用。

BASIC(Beginner's All-purpose Symbolic Instruction Code,初学者通用符号指令代码)语言是国际上广泛使用的一种计算机高级语言,它是一种解释执行的交互式会话语言。BASIC 语言简单、易学,目前仍是计算机入门的主要学习语言之一。

FORTRAN(FORmula TRANslation,公式翻译)语言是目前国际上广泛流行的一种高级语言,适用于科学计算,是专门为科学、工程问题中的那些能够用数学公式表达的问题而设计的语言,主要用于数值计算。这种语言简单易学,因为可以像抄写数学教科书里的公式一样书写数学公式,它比英文书写的自然语言更接近数学语言。

COBOL(Common Business Oriented Language,面向商业的通用语言)是商业数据处理中广泛使用的语言。

Pascal 语言是 20 世纪 70 年代初发展起来的一种结构化程序设计语言,它具有丰富的数据结构类型和优良的结构化特性,最初是作为一种教学语言推出的,后来得到广泛应用。

C 语言最初作为 UNIX 操作系统所使用的主要语言,它具有很强的功能以及高度的灵活性,和其他结构化语言一样,能够提供丰富的数据类型以及功能丰富而强大的运算符。

dBASE 和 FoxBASE 语言是用 C 语言编写的、专门用来处理数据库的编程语言。

### (4) 面向对象的程序设计语言

面向对象的程序设计语言是一种新的程序设计范型,其主要特征是:程序=对象+消息。面向对象程序的基本元素是对象,其主要特点是:程序一般由类的定义和类的使用两部分组成;在主程序中定义各对象并规定它们之间传递消息的规律,程序中的一切操作都是通过向对象发送消息来实现的;对象收到消息后,启动有关方法来完成相应的操作。面

向对象的程序设计语言也有很多,如 C++, C#, Visual Basic, Delphi, LISP, PROLOG, Java 等。

关于面向对象的程序设计语言,不在本书的讨论范围,建议读者在学完本书的内容以后,再继续学习 C++ 语言。

## 1.2 关于 C 语言

### 1.2.1 C 语言的诞生和发展

早期的系统软件几乎都是由汇编语言编写的。汇编语言过分地依赖硬件,可移植性很差。在这台计算机上编写的软件移植到另一台计算机上很可能无法运行。一般高级语言又难于实现汇编语言的某些功能,不能很方便地对底层硬件进行灵活的控制和操作。所以人们急于寻找一种既有高级语言特点又有低级语言功能的中间语言。

在 1960 年出现了 ALGOL(Algorithmic Language,算法语言),它是所有结构化语言的先驱,它有丰富的过程和数据结构,语法严谨。由于 ALGOL 本身及历史的原因,虽然它在欧洲广泛使用,但在国际上并未被广泛采用。它是面向问题的语言,不宜用来编写系统软件。

1963 年,剑桥大学推出了 CPL 语言,比其他高级语言稍接近硬件,但规模较大、不易实现;1967 年,剑桥大学对 CPL 语言做了适当简化后又推出了 BCPL 语言;1970 年,美国贝尔实验室的 Ken Thompson 以 BCPL 为基础做了进一步简化,设计了 B 语言。这种语言更加简单,更加接近硬件。开发者用 B 语言编写了最初的 UNIX 操作系统,尽管它过于简单且功能不全。

1972—1973 年间,美国贝尔实验室的 D. M. Ritchie 在 B 语言基础上设计出了 C 语言。1973 年,K. Thompson & D. M. Ritchie 合作将 UNIX 90% 以上的代码用 C 语言进行了改写。

C 语言是在 20 世纪 70 年代初问世的。1978 年由美国电话电报公司(AT&T)贝尔实验室正式发表了 C 语言。同时,由 B. W. Kernigh 和 D. M. Ritchie 合著了著名的《THE C PROGRAMMING LANGUAGE》一书,通常简称为《K&C》,也有人称之为《K&C 标准》。但是,在《K&C》中并没有定义一个完整的 C 语言标准,后来由美国国家标准协会(ANSI)委任一个委员会对 C 语言进行标准化。美国国家标准学会在此基础上制定了一个 C 语言标准,于 1983 年发表,通常称为 ANSI C。

早期的编程语言大多是用于 UNIX 系统。由于 C 语言的强大功能和各方面的优点,它逐渐为人们所认识,到了 20 世纪 80 年代,C 语言开始进入其他操作系统,并很快在各类大、中、小和微型计算机上得到了广泛的使用,成为当代最优秀的程序设计语言之一。C 语言的产生震撼了整个计算机界,它从根本上改变了编程的方法和思路。C 语言的产生是人们追求结构化、高效率、高级语言的直接结果,可用它替代汇编语言开发系统程序。

### 1.2.2 C 语言的特点

C 语言发展如此迅速,成为最受欢迎的语言之一,主要因为它具有强大的功能。许多著



名的系统软件如 dBASE III PLUS、dBASE IV 都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序,就更能显示 C 语言的优势了,像 PC-DOS、WORDSTAR 等就是用这种方法编写的。归纳起来,C 语言具有下列特点。

(1) C 语言是中级语言。

中级语言并没有贬义,不意味着它功能差,难以使用,或者比 BASIC、Pascal 那样的高级语言原始,也不意味着它与汇编语言相似,会给使用者带来类似的麻烦。C 语言之所以被称为中级语言,是因为它把高级语言的成分同汇编语言的功能结合起来,把高级语言的基本结构和语句与低级语言的实用性结合起来。C 语言可以像汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元。

(2) C 语言是结构化语言。

结构化语言的显著特征是代码和数据的分离。这种语言能够把执行某个特殊任务的指令和数据从程序的其余部分分离出去,并隐藏起来。这种结构化方式可使程序层次清晰,便于使用、维护以及调试。C 语言是以函数形式提供给用户的,这些函数可方便地调用,并具有多种循环、条件语句控制程序流向,从而使程序完全结构化。

(3) C 语言功能齐全。

C 语言具有各种各样的数据类型,并引入了指针概念,可使程序效率更高。另外,C 语言也具有强大的图形功能,支持多种显示器和驱动器,而且计算功能、逻辑判断功能也比较强大,可以实现决策目的。

(4) C 语言适用范围广。

C 语言还有一个突出的优点就是适合多种操作系统,如 DOS、UNIX、Linux 等,也适用于多种机型。

### 1.2.3 怎样学好用好本书

面对种类繁多的编程语言,初学者往往会在究竟选择学习哪门计算机语言作为突破口而犹豫不定。有的人认为语言越新越好,传统语言已经过时,应该学习新出现的计算机语言。笔者认为,虽然“跑”比“走”要快很多,但是不学会走是不可能跑的。对于一个计算机程序设计的初学者来说,不用说“跑”,甚至连想马上“走”好都是很困难的,所以他们必须先从“走”开始。只有认识到了这一点,我们才能不被那些目前流行的、新奇的语言所迷惑。不管一种计算机语言的功能有多强大、应用多广泛,其基本的编程思想和传统语言都是一样的,正所谓“万变不离其宗”。所以笔者认为,要想学习计算机程序设计,还是应该先从这些传统语言学起,因为它们比那些时下流行的语言更简单、简洁。

大学低年级是很重要的打基础阶段,其所选的计算机语言应该能充分适宜大学低年级加强基础、培养与训练计算机语言编程及应用能力、拓宽知识面的特色。C 语言能最好地满足这些要求,这是由 C 语言自身的众多良好特性所决定的。学好 C 语言并不是目的,掌握计算机程序设计的思想,为以后更好地应用计算机技术、学习后续编程语言打下良好的基础才是真正目的。本书是专为大学生入学后以 C 语言作为计算机第一教学语言而编写的,本书通过介绍和剖析大量的程序示例,系统地论述了 C 语言的各种数据类型和语句特性,以及用 C 语言进行程序设计的基本方法、技巧和应用,努力使读者既能熟悉一种优良的语言工具,又能掌握如何发挥这一工具效能的方法与技巧,并着力于良好的编程风格与习惯的养