

21 世纪 计 算 机 科 学 与 技 术 实 践 型 教 程

丛书主编 陈明

刘艳军 主编
胡章平 季松华 万忠杰 副主编

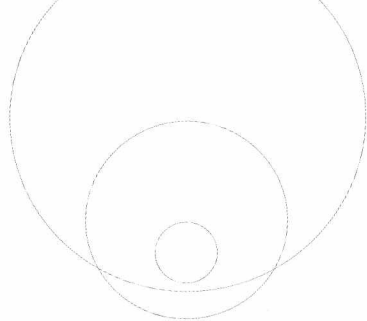
案例式C语言程序设计

清华大学出版社



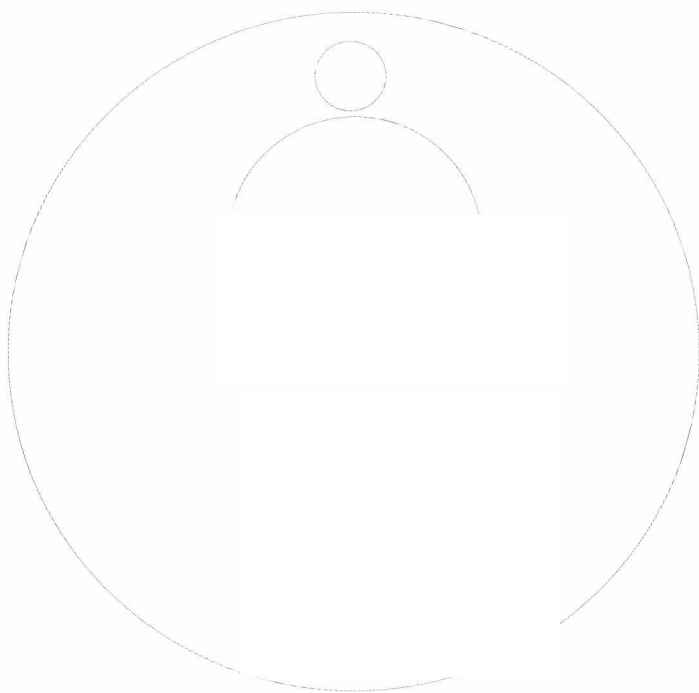
21世纪计算机科学与技术实践型教程

丛书主编 陈明



刘艳军 主编
胡章平 季松华 万忠杰 副主编

案例式C语言程序设计



清华大学出版社
北 京

内 容 简 介

本书引入“算法思想”和“案例教学”的理念,以“理论联系实际,算法制导程序,调试验证程序”的新模式,将理论知识、实践内容与调试运行融合在一起,以实用型的案例为导向,整合主干知识点,使学生能够快速入门,同时又扩展细节知识点,配合阶段性综合项目,从而实现高效、实用的教学目的。

全书共12章,内容包含C语言概述、基本数据类型、选择结构、循环结构、数组、函数、指针、字符串处理、自定义数据类型、文件、位运算和预处理等,是一本兼具趣味性和实用性的C语言程序设计教材。

本书以“学生好学,教师好用”为原则,将“教学—实验—练习”结合在一起形成一体化教程,同时配有多媒体课件、例题和习题源代码等教学资源。

本书可作为高等学校各专业的C语言程序设计课程教材和参考用书,也可作为对C语言程序设计感兴趣的读者的自学用书。

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

案例式C语言程序设计/刘艳军主编. —北京:清华大学出版社,2016

21世纪计算机科学与技术实践型教程

ISBN 978-7-302-44638-5

I. ①案… II. ①刘… III. ①C语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2016)第179377号

责任编辑:谢琛 薛阳

封面设计:常雪影

责任校对:梁毅

责任印制:杨艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印书者:北京富博印刷有限公司

装订者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:18.75

字 数:438千字

版 次:2016年9月第1版

印 次:2016年9月第1次印刷

印 数:1~2000

定 价:39.00元

产品编号:064612-01

《21 世纪计算机科学与技术实践型教程》

编辑委员会

主 任：陈 明

委 员：毛国君 白中英 叶新铭 刘淑芬 刘书家
汤 庸 何炎祥 陈永义 罗四维 段友祥
高维东 郭 禾 姚 琳 崔武子 曹元大
谢树煜 焦金生 韩江洪

策划编辑：谢 琛

《21 世纪计算机科学与技术实践型教程》

序

21 世纪影响世界的三大关键技术是：以计算机和网络为代表的信息技术；以基因工程为代表的生命科学和生物技术；以纳米技术为代表的新型材料技术。信息技术居三大关键技术之首。国民经济的发展采取信息化带动现代化的方针，要求在所有领域中迅速推广信息技术，导致需要大量的计算机科学与技术领域的优秀人才。

计算机科学与技术的广泛应用是计算机学科发展的原动力，计算机科学是一门应用科学。因此，计算机学科的优秀人才不仅应具有坚实的科学理论基础，而且更重要的是能将理论与实践相结合，并具有解决实际问题的能力。培养计算机科学与技术的优秀人才是社会的需要、国民经济发展的需要。

制定科学的教学计划对于培养计算机科学与技术人才十分重要，而教材的选择是实施教学计划的一个重要组成部分，《21 世纪计算机科学与技术实践型教程》主要考虑了下述两方面。

一方面，高等学校的计算机科学与技术专业的学生，在学习了基本的必修课和部分选修课程之后，立刻进行计算机应用系统的软件和硬件开发与应用尚存在一些困难，而《21 世纪计算机科学与技术实践型教程》就是为了填补这部分鸿沟。将理论与实际联系起来，结合起来，使学生不仅学会了计算机科学理论，而且也学会应用这些理论解决实际问题。

另一方面，计算机科学与技术专业的课程内容需要经过实践练习，才能深刻理解和掌握。因此，本套教材增强了实践性、应用性和可理解性，并在体例上做了改进——使用案例说明。

实践型教学占有重要的位置，不仅体现了理论和实践紧密结合的学科特征，而且对于提高学生的综合素质，培养学生的创新精神与实践能力有特殊的作用。因此，研究和撰写实践型教材是必须的，也是十分重要的任务。优秀的教材是保证高水平教学的重要因素，选择水平高、内容新、实践性强的教材可以促进课堂教学质量的快速提升。在教学中，应用实践型教材可以增强学生的认知能力、创新能力、实践能力以及团队协作和交流表达能力。

实践型教材应由教学经验丰富、实际应用经验丰富的教师撰写。此系列教材的作者不但从事多年的计算机教学，而且参加并完成了多项计算机类的科研项目，把他们积累的经验、知识、智慧、素质融合于教材中，奉献给计算机科学与技术的教学。

我们在组织本系列教材过程中，虽然经过了详细地思考和讨论，但毕竟是初步的尝试，不完善甚至缺陷不可避免，敬请读者指正。

本系列教材主编 陈明

2005 年 1 月于北京

前 言

程序设计是高等学校重要的计算机基础课程,它以编程语言为平台,介绍程序设计的思想和方法。学习程序设计本身是一件充满挑战和乐趣的事情,然而同学们在学习初期常常有“入门难”,“看得懂,但写不出”,“没思路”的感觉。这是由于传统教材强调理论的系统性和完整性,轻知识的应用性。本书引入“算法思想”和“案例教学”的理念,以“理论联系实际,算法制导程序,调试验证程序”的新模式,将理论知识、实践内容与调试运行融合在一起,以实用型的案例为导向,整合主干知识点,使学生能够快速入门,同时又扩展细节知识点,配合综合项目,从而实现高效、实用的教学目的。

本书是一本兼具趣味性和实用性的 C 语言程序设计教材,书中的案例都是编者参考大量教材并结合教学实践精心挑选的,既贴近生活便于理解,又尽量涵盖课程知识点。

全书共 12 章,中间穿插 4 个“我厉害了”的阶段项目和一个贯穿项目。各章按照“典型程序样例”→“语法、实践知识点说明”→“模仿编程”→“实践提高”→“综合应用”的顺序组织内容,先分析问题,用流程图列出解题思想,再用代码实现对应功能,最后在调试中观察程序的运行,掌握问题求解实现的全过程,带给读者以全新的体验。

本书由刘艳军主编并统稿,副主编为胡章平、季松华、万忠杰,罗代忠、李瑞、王俊祥、高峰也参与了教材的编写,参编人员都是长期从事“C 语言程序设计”课程教学的教师,他们在长期教学中积累了丰富的经验。这里尤其要感谢罗代忠教授,他在本书的策划和出版过程中给予了很多建议和帮助。本书在编写过程中参考了国内外许多相关著作,在此也向这些著作的作者们致以衷心的感谢。

由于水平有限,时间仓促,疏漏和不妥之处在所难免,敬请读者批评指正。

编 者

2016 年 3 月

目 录

第 1 章 概述	1
1.1 编程的概念	1
1.1.1 计算机如何工作	1
1.1.2 计算机语言	1
1.1.3 C 语言简介	2
1.2 如何学习 C 语言程序设计	4
1.2.1 C 语言编程需要什么	4
1.2.2 学习 C 语言程序设计的有效方法	4
1.2.3 创建 C 语言程序的一般步骤	5
1.3 C 语言程序的基本结构及书写规范	5
1.3.1 C 语言程序的基本结构	5
1.3.2 简单的 C 程序	6
1.3.3 C 语言的“字母”和“单词”	7
1.3.4 C 语言的程序书写规范	7
1.4 Visual C++ 集成编译环境介绍	8
1.4.1 Visual C++ 集成编译环境的安装	8
1.4.2 Visual C++ 集成编译环境的简单使用	10
1.5 算法及算法表示	15
1.5.1 算法与程序设计	15
1.5.2 算法的表示	16
练习与自测	18
实验指导	19
第 2 章 基本数据类型	21
2.1 计算购买实验指导书的总价	21
2.1.1 程序分析	21
2.1.2 常量和变量	22
2.1.3 赋值运算和算术运算	23

2.1.4	整型	25
2.1.5	输入输出函数	26
2.1.6	程序的排错与调试	29
2.2	计算银行存款的本息和	33
2.2.1	程序分析	33
2.2.2	库函数	34
2.2.3	表达式	35
2.2.4	实型	35
2.2.5	各类数值型数据之间的混合运算	37
2.2.6	程序的排错与调试	38
2.3	顺序结构	41
2.3.1	顺序结构	41
2.3.2	顺序结构典型问题——单位及货币换算	42
	编程习惯	46
	练习与自测	46
	实验指导	48

第3章 会思考的程序——选择结构

3.1	求整数绝对值(if语句——双分支和条件执行)	50
3.1.1	程序分析	50
3.1.2	关系运算符与关系表达式	51
3.1.3	逻辑运算符与逻辑表达式	52
3.1.4	if语句	54
3.1.5	程序的排错与调试	55
3.2	学生成绩评价(if——多分支结构)	56
3.2.1	程序分析	57
3.2.2	if语句多分支结构	58
3.3	if语句的嵌套	59
3.3.1	程序分析	59
3.3.2	if语句的嵌套	61
3.4	打印英文星期名(switch语句)	62
3.4.1	程序分析	62
3.4.2	switch语句	63
3.5	条件运算符	66
	本章知识点总结	67
	问与答	67
	编程习惯	68
	练习与自测	68

实验指导	71
第 4 章 反复执行,周而复始——循环结构	73
4.1 求 1 到 100 的累加和	73
4.1.1 程序分析	73
4.1.2 复合赋值运算符和自增自减运算符	76
4.1.3 if-goto 语句构成的循环	77
4.1.4 while 语句构成的循环	77
4.1.5 for 语句构成的循环	78
4.1.6 do-while 语句构成的循环	80
4.1.7 4 种循环结构的比较	80
4.1.8 程序的调试与运行	82
4.2 计算 $a+aa+aaa+aaaa+\cdots+aa\cdots a$ 的累加和	84
4.2.1 程序分析	84
4.2.2 程序的调试与运行	85
4.3 利用泰勒公式求圆周率	86
4.3.1 程序分析	86
4.3.2 程序的调试与运行	88
4.4 求学生平均成绩	89
4.4.1 程序分析	89
4.4.2 程序的调试与运行	90
4.5 打印九九乘法表	91
4.5.1 程序分析	91
4.5.2 循环嵌套	93
4.6 输出 100~200 之间的素数	94
4.6.1 程序分析	94
4.6.2 break 语句	96
4.7 计算 2~100 以内的素数和	97
4.7.1 程序分析	97
4.7.2 continue 语句	98
4.8 拓展应用	99
4.8.1 穷举法	99
4.8.2 迭代法	99
本章知识点总结	100
问与答	100
编程习惯	101
练习与自测	102
实验指导	104

案例：我厉害了——简易自动售货机	105
第 5 章 批量数据的处理——数组	108
5.1 统计成绩	108
5.1.1 程序分析	108
5.1.2 一维数组	110
5.1.3 程序的调试与运行	115
5.2 打印杨辉三角	116
5.2.1 程序分析	116
5.2.2 二维数组	118
5.2.3 程序的调试与运行	122
本章知识点总结	122
问与答	122
练习与自测	123
实验指导	126
案例：我厉害了——打印日历、学生成绩管理 v1.0	127
第 6 章 模块化与协作开发——函数	131
6.1 计算圆柱体积	132
6.1.1 程序分析	132
6.1.2 函数的定义	133
6.1.3 程序的调试与运行	136
6.2 函数的嵌套调用	139
6.3 5 个人求年龄问题	140
6.3.1 程序分析	140
6.3.2 递归	141
6.4 查找最高分和最低分	143
6.4.1 程序分析	143
6.4.2 全局变量和局部变量	144
6.4.3 变量的存储类别	146
本章知识点总结	147
问与答	147
练习与自测	148
实验指导	151
案例：我厉害了——自动售货机 v1.0	152
贯穿项目——学生成绩管理系统 v2.0	157

第 7 章 间接访问——指针	166
7.1 获取自动售货机密码(指针的概念)	166
7.2 偷梁换柱——改密码(指针作函数参数)	170
7.3 对学生成绩进行排序(数组名作函数参数)	173
7.3.1 使用指针访问数组	173
7.3.2 用数组名和指针作参数	175
7.3.3 程序的调试与运行	177
7.4 二维数组中学生成绩的输出(指向多维数组的指针)	178
本章知识点总结	182
问与答	182
练习与自测	184
实验指导	185
第 8 章 字符串处理	187
8.1 将验证码全部改为大写输出	187
8.1.1 程序分析	187
8.1.2 字符型	188
8.1.3 字符型的输入输出	190
8.2 名片排序	191
8.2.1 程序分析	191
8.2.2 字符串常量	193
8.2.3 字符数组	193
8.2.4 字符数组的输入输出	194
8.2.5 字符串处理函数	197
8.3 字符指针	199
8.3.1 使用字符指针访问字符串	199
8.3.2 字符指针作参数	200
本章知识点总结	202
问与答	202
练习与自测	202
实验指导	203
第 9 章 自定义数据类型	205
9.1 构建学生信息结构体	205
9.2 结构体类型的定义	206
9.2.1 结构体 3 种方法	206
9.2.2 typedef 的使用	207

9.3 结构体变量的使用	208
9.3.1 结构体变量的初始化	208
9.3.2 结构体变量的赋值	208
9.3.3 应用举例	209
9.4 结构体数组	210
9.5 结构体指针	212
9.5.1 结构体指针	212
9.5.2 结构体指针与链表	214
9.6 其他自定义类型	220
9.6.1 共用体	220
9.6.2 枚举类型	222
本章知识点总结	224
问与答	224
练习与自测	225
实验指导	226
第10章 数据持久化——文件	229
10.1 将短句“Hello World!”写入到文件	229
10.1.1 程序解析	230
10.1.2 文件概述	230
10.1.3 文件操作步骤	231
10.1.4 文件打开、关闭函数	232
10.2 将学生信息保存到文件(fprintf)	234
10.3 从文件读取学生信息(fscanf)	237
10.4 文件的复制(fgetc 和 fputc)	238
10.5 二进制文件(数据块读写 fread 和 fwrite)	241
10.6 文件定位函数	245
本章知识点总结	248
练习与自测	248
实验指导	249
第11章 位运算	251
11.1 IP 地址转换器	251
11.2 位运算的概念	252
11.3 位运算符	253
11.4 位域	255
本章知识点总结	258
练习与自测	258

实验指导	259
第 12 章 预处理	260
12.1 宏定义	260
12.1.1 不带参数的宏	260
12.1.2 带参数的宏	263
12.2 文件包含	264
12.3 条件编译	265
本章知识点总结	268
问与答	268
练习与自测	268
实验指导	270
附录 A C 语言中的关键字	272
附录 B 运算符和结合性	273
附录 C C 库函数	274
附录 D ASCII 表	280
参考文献	281

第 1 章 概 述

学习目标

- 理解编程的目的；
- 了解学习 C 语言的方法；
- 掌握 C 语言编写规范与运行环境；
- 掌握什么是算法及算法的表示。

1.1 编程的概念

大家都知道计算机本身是一台机器,为了使计算机能够理解人的意图,人类就必须将解决问题的思路、方法和手段通过计算机能够理解的形式告诉计算机,使得计算机能够根据人的指令一步一步去工作,进而完成某种特定的任务。这些指令序列的集合称为程序,而这种人和计算机之间交流的过程就是编程。

编程就是编写程序的中文简称,就是让计算机为解决某个问题而使用某种程序设计语言编写程序代码,并最终得到相应结果的过程。例如,Office Word 就是为了解决文档编辑所编写的程序,是将很多指令代码集合起来形成的应用软件。

1.1.1 计算机如何工作

计算机是由很多电子元件组装而成的,这些电子元件构成计算机的硬件系统,大致由 5 个基本部分组成:运算器、控制器、存储器、输入设备和输出设备。这 5 个部分通过系统总线完成指令所传达的操作,计算机在接受指令后,由控制器指挥,将数据从输入设备传送到存储器存放,再由控制器将需要参加运算的数据传送到运算器,由运算器进行处理,处理后的结果由输出设备输出。其工作流程如图 1-1 所示。

1.1.2 计算机语言

1. 机器语言

计算机应用的初期,一个系列的计算机有自身的指令系统,程序员使用机器本身的指令系统来编写计算机应用程序,这种程序称为机器语言程序。使用机器语言编写的程序,由于每条指令都对应计算机一个特定的基本动作,所以程序占用内存少、执行效率高。缺

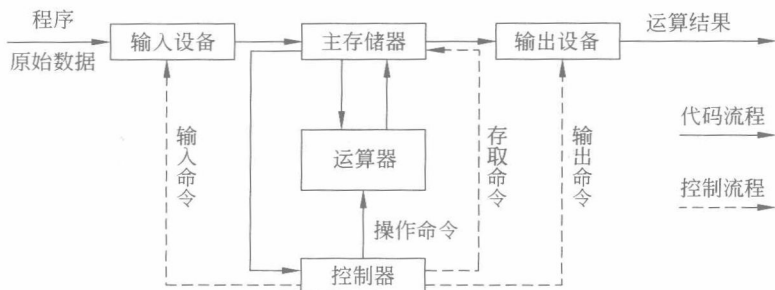


图 1-1 计算机工作流程

点也很明显,如编程工作量大,容易出错;依赖具体的计算机体系,因而程序的通用性、移植性都很差。

2. 汇编语言

为了解决使用机器语言编写应用程序所带来的一系列问题,人们首先想到使用助记符号来代替不容易记忆的机器指令。这种利用助记符号来表示计算机指令的语言称为符号语言,也称为汇编语言。在汇编语言中,每一条用符号来表示的汇编指令与计算机机器指令一一对应;记忆难度大大减小了,不仅易于检查和修改程序错误,而且指令、数据的存放位置可以由计算机自动分配。用汇编语言编写的程序称为源程序,计算机不能直接识别和处理源程序,必须通过某种方法将它翻译成为计算机能够理解并执行的机器语言,执行这个翻译工作的程序称为汇编程序。现在尽管很少使用汇编语言来编写程序,但它有不可替代的特性,例如一些单片机或者一些直接控制硬件的程序就一定要用汇编语言。

3. 高级语言

由于汇编语言与计算机硬件系统关系密切,程序员需要十分熟悉计算机系统的硬件结构。为了让更多人了解计算机,也能进行简单的程序编写,一类接近于人类的自然语言和数学语言的程序设计语言就出现了,通称高级语言。高级语言分为面向过程的语言和面向对象的语言,如 Fortran 语言、C 语言等都是面向过程的语言;而 C++、C#、Java 等是面向对象的语言。

1.1.3 C 语言简介

C 语言是一门通用的计算机编程语言,应用广泛。C 语言的设计目标是提供一种能以简易的方式编译、处理低级存储器、产生少量的机器码以及不需要任何运行环境支持便能运行的编程语言。

1. C 语言的发展过程

(1) C 语言的原型是 ALGOL 60 语言(也称为 A 语言),是面向对象的语言,离硬件比较远,属于逻辑语言。

(2) 1963 年,剑桥大学将 ALGOL 60 语言发展成为 CPL(Combined Programming Language)语言。

(3) 1967 年,剑桥大学的 Martin Richards 对 CPL 语言进行了简化,于是产生了 BCPL 语言。

(4) 1970 年,美国贝尔实验室的 Ken Thompson 将 BCPL 进行了修改,并为它起了一个有趣的名字“B 语言”。意思是将 CPL 语言煮干,提炼出它的精华。并且他用 B 语言写了第一个 UNIX 操作系统。

(5) 1973 年,B 语言也给人“煮”了一下,美国贝尔实验室的 D. M. RITCHIE 在 B 语言的基础上最终设计出了一种新的语言,他取了 BCPL 的第二个字母作为这种语言的名字,这就是 C 语言。

(6) 为了使 UNIX 操作系统获得推广,1977 年,Dennis M. Ritchie 发表了不依赖于具体机器系统的 C 语言编译文本《可移植的 C 语言编译程序》。

(7) 1978 年 Brian W. Kernighan 和 Dennis M. Ritchie 出版了名著 *The C Programming Language*,从而使 C 语言成为目前世界上流行最广泛的高级程序设计语言。

(8) 1987 年,随着微型计算机的日益普及,出现了许多 C 语言版本。由于没有统一的标准,使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况,美国国家标准研究所(ANSI)为 C 语言制定了一套 ANSI 标准,成为现行的 C 语言标准 3。C 语言的主要特点,即经典的 87 ANSI C。C 语言发展迅速,而且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。

(9) 1990 年,国际化标准组织 ISO(International Standard Organization)接受了 87 ANSI C 为 ISO C 的标准(ISO 9899—1990)。1994 年,ISO 修订了 C 语言的标准。目前流行的 C 语言编译系统大多是以 ANSI C 为基础进行开发的,但不同版本的 C 编译系统所实现的语言功能和语法规则略有差别。

2. C 语言的特点

C 语言是一种结构化语言。它层次清晰,便于按模块化方式组织程序,易于调试和维护。C 语言的表现能力和处理能力极强。它不仅具有丰富的运算符和数据类型,便于实现各类复杂的数据结构。它还可以直接访问内存的物理地址,进行位(bit)一级的操作。具体来讲,C 语言的特点有以下几点:

(1) 简洁紧凑、灵活方便。C 语言一共只有 32 个关键字,9 种控制语句,程序书写自由,主要用小写字母表示。它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) 运算符丰富。C 的运算符包含的范围很广泛,共有 34 个运算符。C 语言把括号、赋值、强制类型转换等都作为运算符来处理。从而使 C 的运算类型极其丰富、表达式类型多样化,灵活使用各种运算符可以实现在其他高级语言中难以实现的运算。

(3) 数据结构丰富。C 的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等。能用来实现各种复杂的数据类型的运算。并引入了指针概念,使

程序效率更高。另外 C 语言具有强大的图形功能,支持多种显示器和驱动器,且计算功能、逻辑判断功能强大。

(4) C 是结构式语言。结构式语言的显著特点是代码及数据的分隔化,即程序的各个部分除了必要的信息交流外彼此独立。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。

(5) C 语法限制不太严格、程序设计自由度大。一般的高级语言语法检查比较严,能够检查出几乎所有的语法错误。而 C 语言允许程序编写者有较大的自由度。

(6) C 语言允许直接访问物理地址,可以直接对硬件进行操作。因此 C 语言既具有高级语言的功能,又具有低级语言的许多功能,能够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,可以用来写系统软件。

(7) C 语言程序生成代码质量高,程序执行效率高,一般只比汇编程序生成的目标代码效率低 10%~20%。

(8) C 语言适用范围广,可移植性好。

(9) C 语言有一个突出的优点就是适合于多种操作系统,如 DOS、UNIX,也适用于多种机型。

常用的 C 语言 IDE(集成开发环境)有 Microsoft Visual C++, Borland C++, Watcom C++, Borland C++ Builder, Borland C++ 3.1 for DOS, Watcom C++ 11.0 for DOS, GNU DJGPP C++, Lccwin32 C Compiler 3.1, Microsoft C, High C, Turbo C, Dev-C++, C-Free, WIN-TC 等。

当然,C 语言也有自身的不足,例如 C 语言的语法限制不太严格,对变量的类型约束不严格,影响程序的安全性,对数组下标越界不作检查等。从应用的角度来看,C 语言比其他高级语言较难掌握。

总之,C 语言既有高级语言的特点,又具有汇编语言的特点;既是一种成功的系统设计语言,又是一种实用的程序设计语言;既能用来编写不依赖计算机硬件的应用程序,又能用来编写各种系统程序;是一种受欢迎、应用广泛的程序设计语言。

1.2 如何学习 C 语言程序设计

1.2.1 C 语言编程需要什么

环境需求:

- (1) 一台计算机,现在的计算机配置一般都能满足 C 语言编程要求;
- (2) 安装开发环境进行代码编写与调试,常用环境有 Turbo C 2.0, Turbo C++ 3.0, Microsoft Visual C++ 等。

1.2.2 学习 C 语言程序设计的有效方法

C 语言是“人与计算机”沟通的工具,就像英语是中国人与英国人沟通的一种工具—