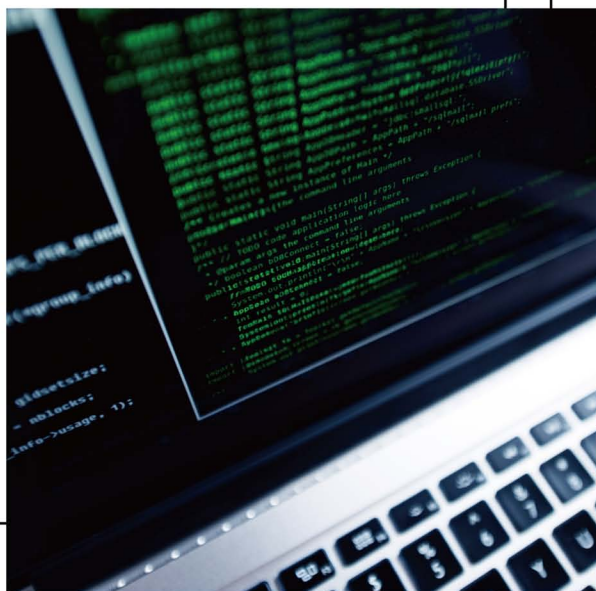




国家新闻出版改革发展项目库入库项目
高等院校计算机类规划教材
全国高等院校计算机基础教育研究会重点立项项目



C语言

程序设计教程

主 编 李艳玲
副主编 路 璐 马 强



北京邮电大学出版社
www.buptpress.com





国家新闻出版改革发展项目库入库项目
高等院校计算机类规划教材
全国高等院校计算机基础教育研究会重点立项项目

C 语言程序设计教程

主 编 李艳玲
副主编 路 璐 马 强



北京邮电大学出版社
www.buptpress.com

内 容 简 介

本书是高等院校计算机基础教育研究课题的成果之一。全书共分 11 章,结合计算思维培养,以一个完整的图书管理系统案例对每个知识点进行详细的分析,内容包括计算思维与 C 语言程序设计、C 语言的基础知识、顺序结构程序设计、选择结构程序设计、循环结构程序设计、函数、数组、指针、自定义数据类型、预处理、文件等。

本书选材先进,内容丰富,结构完整,理论联系实际,深入浅出,循序渐进,通俗易懂,注重培养读者的程序设计能力及良好的程序设计风格和习惯,并配有大量的实例以方便读者上机实践。

本书可作为高等院校各专业计算机公共基础课程程序设计的教学用书,也可作为计算机等级考试培训教材及自学人员用书。

图书在版编目(CIP)数据

C 语言程序设计教程 / 李艳玲主编. -- 北京:北京邮电大学出版社, 2020.6
ISBN 978-7-5635-6060-8

I. ①C… II. ①李… III. ①C 语言—程序设计—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2020) 第 081986 号

策划编辑:马晓仟 责任编辑:王晓丹 左佳灵 封面设计:七星博纳

出版发行:北京邮电大学出版社

社 址:北京市海淀区西土城路 10 号

邮政编码:100876

发行部:电话:010-62282185 传真:010-62283578

E-mail: publish@bupt.edu.cn

经 销:各地新华书店

印 刷:

开 本:787 mm×1 092 mm 1/16

印 张:14.5

字 数:358 千字

版 次:2020 年 6 月第 1 版

印 次:2020 年 6 月第 1 次印刷

ISBN 978-7-5635-6060-8

定价:40.00 元

· 如有印装质量问题,请与北京邮电大学出版社发行部联系 ·

前 言

本书是按照教育部高等院校计算机基础课程教指委制订的《大学计算机基础课程教学基本要求》中有关程序设计基础的教学要求编写的。程序设计基础是高校计算机基础教学的核心课程,本书以高级编程语言为平台,介绍计算机程序设计的思想和方法,既可为学生后继计算机相关课程的学习打下基础,也有利于帮助学生理解基本的计算思想和方法,培养学生应用计算机求解问题的基本能力。

本书充分考虑应用型本科院校“强应用”的特色,结合近几年的教学和开发实践经验,围绕项目案例——图书管理系统的开发,对各章知识点进行了详细的分析,深入浅出地阐述了 C 语言程序设计方法。

全书共分 11 章,具体内容如下。

第 1 章主要介绍计算思维与 C 语言程序设计,帮助学生初步认识 C 语言,了解计算思维与计算机的关系,为后面的程序开发奠定基础。

第 2~7 章主要介绍语言的基础知识,主要包括数据类型、结构化程序设计方法、函数、数组等,并提供了丰富的案例,使学生了解程序设计语言的基本结构和运用程序设计求解实际问题的基本过程。

第 8~11 章主要介绍 C 语言中的核心内容,主要包括指针、结构体、共用体、预处理、文件操作等。只有熟练掌握这些知识,才能真正掌握程序设计的基本思想和方法,才能初步具备利用程序设计语言和开发环境编程求解实际问题的能力。

本书内容编排循序渐进,通俗易懂,语言简练,将一些复杂问题简单化,以“零基础”为起点,让初学者能够轻松理解并快速掌握相关知识。全书案例由易到难,所有代码均已上机通过,可直接引用。本书适合作为高等院校各专业计算机公共基础课程程序设计方面的教材,还可作为计算机等级考试的培训教材及自学人员的用书。

本书是大家通力合作的成果,是集体智慧的结晶。本书由李艳玲组织编写,并担任主编,由路璐、马强担任副主编。本书各章分工如下:第 1 章、第 2 章由杜丽美编写;第 3 章、第

4 章、第 5 章由李艳玲编写；第 6 章、第 7 章由路璐编写；第 8 章、第 9 章由李慧玲编写；第 10 章、第 11 章由马强编写。

为了方便老师使用和学生学习,本教材配备了丰富的教学资源,除了精美的课件外,我们还提供课程大纲、电子教案、所有例题代码、操作录屏和习题答案。

由于作者水平有限,书中难免存在疏漏与不妥之处,敬请广大读者批评指正。作者邮箱:hhly11109@163.com。期待您的来信和指导。

编者



“北邮智信”App 使用说明

目 录

第 1 章 计算思维与 C 语言程序设计	1
1.1 什么是计算思维	1
1.1.1 计算思维概念	1
1.1.2 计算思维的特征	1
1.1.3 计算思维的本质	2
1.1.4 计算思维与计算机的关系	2
1.2 C 语言概述	3
1.2.1 计算机语言发展史	3
1.2.2 什么是 C 语言	3
1.2.3 C 语言的特点	4
1.2.4 C 语言的发展趋势	5
1.3 C 语言开发环境的搭建	5
1.3.1 Visual C++ 6.0 集成开发环境	6
1.3.2 利用 Visual C++ 6.0 开发环境新建文件	6
1.3.3 利用 Visual C++ 6.0 开发环境编写程序	7
1.4 C 语言代码风格	9
1.4.1 程序格式	9
1.4.2 程序注释	11
本章小结	11
习题 1	11
第 2 章 C 语言的基础知识	12
2.1 C 语言的基本数据类型	12
2.1.1 数据类型概述	12
2.1.2 整数类型	13
2.1.3 实数类型	13
2.1.4 字符类型	14
2.2 标识符与关键字	14
2.3 常量	15
2.3.1 数值常量	15

2.3.2	字符型常量	15
2.3.3	字符串常量	16
2.3.4	符号常量	16
2.4	变量	16
2.5	运算符与表达式	17
2.5.1	算术运算符	17
2.5.2	关系运算符	19
2.5.3	逻辑运算符	19
2.5.4	赋值运算符	20
2.5.5	条件运算符	21
2.5.6	逗号运算符	21
2.5.7	求字节运算符	21
2.5.8	各种运算符的优先级	22
2.6	数据类型转换	22
2.6.1	自动转换	22
2.6.2	赋值转换	23
2.6.3	强制类型转换	24
2.7	本章小结	24
	习题 2	24
第 3 章	顺序结构程序设计	27
3.1	C 语言的基本语句	27
3.1.1	表达式语句	27
3.1.2	函数调用语句	27
3.1.3	流程控制语句	28
3.1.4	空语句	28
3.1.5	复合语句	28
3.2	输入输出操作	29
3.2.1	格式化输入与输出	29
3.2.2	字符数据输入与输出	32
3.3	顺序程序设计	34
3.4	图书管理系统案例	35
	本章小结	36
	习题 3	36
第 4 章	选择结构程序设计	38
4.1	if 条件语句	38

4.1.1 if 语句	38
4.1.2 if... else 语句	40
4.1.3 if... else if 语句	41
4.1.4 if 条件语句的嵌套	43
4.2 switch 条件语句	43
4.3 图书管理系统案例	47
本章小结	49
习题 4	49
第 5 章 循环结构程序设计	51
5.1 for 循环语句	51
5.1.1 语句格式	51
5.1.2 for 语句的变形	53
5.2 while 循环语句	54
5.3 do... while 循环语句	55
5.4 循环结构的嵌套	57
5.5 break 和 continue 语句	58
5.5.1 break 语句	58
5.5.2 continue 语句	59
5.5.3 break 和 continue 的区别	60
5.6 图书管理系统案例	60
本章小结	62
习题 5	62
第 6 章 函数	65
6.1 函数概述	65
6.1.1 使用 include 命令行调用标准库函数	65
6.1.2 标准库函数的调用	66
6.2 函数定义和返回值	66
6.2.1 函数定义	66
6.2.2 函数的返回值	67
6.3 函数的调用	69
6.3.1 函数的调用方式	69
6.3.2 函数调用时的语法规定	69
6.4 函数的声明	71
6.4.1 函数声明的形式	71
6.4.2 函数声明的位置	72

6.5 参数传递	73
6.5.1 数据传递方式	73
6.5.2 函数调用的过程	74
6.6 函数的嵌套与递归	75
6.6.1 嵌套调用	75
6.6.2 递归调用	75
6.7 变量的作用域和存储类型	77
6.7.1 局部变量和全局变量	77
6.7.2 变量的存储类型	78
6.7.3 内部函数与外部函数	80
6.8 图书管理系统案例	80
本章小结	83
习题 6	83
第 7 章 数组	85
7.1 数组概述	85
7.2 一维数组	85
7.2.1 一维数组的定义	85
7.2.2 一维数组的初始化	86
7.2.3 一维数组元素的引用	87
7.3 二维数组	89
7.3.1 二维数组的定义	89
7.3.2 二维数组的初始化	90
7.3.3 二维数组元素的引用	90
7.4 多维数组	92
7.5 字符数组与字符串	93
7.5.1 字符数组的定义	93
7.5.2 字符数组的初始化	93
7.5.3 字符串处理的函数	95
7.6 数组作为函数参数	98
7.6.1 数组元素作为函数的实参	98
7.6.2 数组名作为函数的实参	98
7.7 图书管理系统案例	99
本章小结	101
习题 7	101
第 8 章 指针	104
8.1 指针的基本概念	104

8.2 变量与指针	104
8.2.1 指针变量的定义	105
8.2.2 指针变量的引用	105
8.2.3 指针变量作为函数参数	108
8.3 一维数组与指针	109
8.3.1 指向数组元素的指针变量	109
8.3.2 指向数组的指针的相关运算	110
8.3.3 通过指针引用数组元素	112
8.3.4 数组作函数的参数	115
8.4 二维数组与指针	119
8.4.1 二维数组的地址	119
8.4.2 指向二维数组元素的指针	121
8.4.3 行指针变量	122
8.5 字符串与指针	123
8.5.1 字符串的表示与引用	123
8.5.2 字符串指针作函数参数	124
8.6 返回指针值的函数	125
8.7 指针数组	126
8.7.1 指针数组概述	126
8.7.2 指向指针的指针	128
8.8 函数的指针和指向函数的指针变量	130
8.9 图书管理系统案例	131
本章小结	132
习题 8	132
第 9 章 自定义数据类型	134
9.1 结构体类型	134
9.1.1 结构体类型的定义	134
9.1.2 结构体变量	136
9.1.3 结构体数组	139
9.1.4 结构体指针	142
9.2 共用体数据类型	147
9.2.1 共用体类型的定义	147
9.2.2 共用体变量的引用	148
9.2.3 共用体的应用	149
9.3 枚举数据类型	150
9.4 自定义类型	152

9.5 图书管理系统案例	153
本章小结	155
习题 9	155
第 10 章 预处理	157
10.1 预处理器	157
10.2 宏定义	158
10.2.1 不带参数的宏	158
10.2.2 带参数的宏	160
10.3 文件包含	162
10.4 条件编译	163
10.4.1 #if 命令	163
10.4.2 #ifdef 命令	164
10.4.3 #ifndef 命令	165
本章小结	165
习题 10	165
第 11 章 文件	168
11.1 文件概述	168
11.1.1 文件的分类	168
11.1.2 文件指针	169
11.2 文件的打开与关闭	169
11.2.1 文件打开 fopen 函数()	170
11.2.2 文件关闭函数 fclose 函数()	171
11.3 文件的读写	172
11.3.1 字符读写函数 fgetc()和 fputc()	172
11.3.2 字符串读写函数 fgets()和 fputs()	176
11.3.3 数据块读写函数 fread()和 fwrite()	178
11.3.4 格式化读写函数 fscanf()和 fprintf()	180
11.4 文件的随机读写	182
11.4.1 文件定位	182
11.4.2 文件的随机读写	183
11.5 文件检测函数	184
11.5.1 文件结束检测函数 feof()	184
11.5.2 读写文件出错检测函数 ferror()	184
11.5.3 文件出错标志和文件结束标志置 0 函数 clearerr()	185
11.6 图书管理系统案例	185

本章小结.....	186
习题 11	187
习题参考答案.....	189
参考文献.....	210
附录 1 运算符的优先级和结合性	211
附录 2 常用字符的 ASCII 码对照表	213
附录 3 C 语言常用库函数	215

第 1 章 计算思维与 C 语言程序设计

【学习目标】

- 理解计算思维概念、特征、本质
- 了解计算思维与计算机的关系
- 了解 C 语言的发展史
- 理解 C 语言的特点
- 了解 C 语言的发展趋势
- 学会 C 语言开发环境的搭建

1.1 什么是计算思维

1.1.1 计算思维概念

计算思维概念的提出是计算机学科发展的自然产物。第一次明确使用这一概念的是美国卡内基·梅隆大学的周以真(Jeannette M. Wing)教授。她认为:计算思维是运用计算机科学的基础概念去求解问题、设计系统和理解人类的行为;是人类求解问题的一条途径,但绝非要使人类像计算机那样地思考。计算思维是一种递归思维,其本质是抽象和自动化。

计算思维(computational thinking)又称构造思维,是指从具体的算法设计规范入手,通过算法过程的构造与实施来解决给定问题的一种思维方法。计算思维涵盖了计算机科学之广度的一系列思维活动。

计算思维是人类科学思维活动固有的组成部分。人类在认识世界、改造世界的过程中表现出了 3 种基本的思维特征:以观察和总结自然规律为特征的实证思维(以物理学科为代表)、以推理和演绎为特征的推理思维(以数学学科为代表)、以设计和构造为特征的计算思维(以计算机学科为代表)。计算机技术的出现及其广泛应用,更进一步强化了计算思维的意义和作用。

计算思维不仅反映了计算机学科最本质的特征和最核心的方法,也反映了计算机学科的 3 个不同领域(理论、设计、实现)。

1.1.2 计算思维的特征

计算思维有如下特点。

(1) 计算思维汲取了问题求解所用的一般数学思维方式,颠覆了现实世界中设计与评估巨大复杂系统的一般过程思维方法和理解心理以及人类行为的一般科学思维方法。

(2) 计算思维建立在计算过程的能力和限制之上,由人和机器执行;计算方法和模型可以处理那些原本无法由个人独立完成的问题。

(3) 计算思维最根本的内容是抽象,计算思维中的抽象完全超越物理中的时空观,以致完全用符号来表示。与数学和物理的抽象相比,计算机思维的抽象更为丰富和复杂。

1.1.3 计算思维的本质

计算思维是基于可计算的,以定量化方式求解问题的一种思维过程;是通过约简、嵌入、转化和仿真等方法,把一个困难的问题重新描述成一个成熟的解决方案;是一种选择合适的方式陈述一个问题,或对一个问题的相关方面建模使其易于处理的思维方法;是按照预防、保护及通过冗余、容错、纠错的方式,从最坏情况进行系统恢复的一种思维方法;是利用启发式推理寻求解答,即在不确定情况下规划、学习和调度的思维方法;是利用海量数据来加快计算,在时间和空间之间、在处理能力和存储容量之间进行折中的思维方法。

在理解计算思维时,要特别注意以下几个问题。

(1) 计算思维并不仅仅是像计算机科学家那样去为计算机编程,还要求能够在多个层次上进行抽象思维。

(2) 计算思维是一种根本技能,是每个人为了在现代社会中发挥职能所必须具有的思维。

(3) 计算思维是人类求解问题的一条途径,但绝非要使人类像计算机那样去思考。计算机给了人类强大的计算能力,人类应该好好利用这种能力去解决各种现实问题。

(4) 计算思维是思想,不是人造品。计算机科学不只是将软硬件等人造物呈现在人们的生活中,更重要的是计算的概念,它被人们用来求解问题、管理日常生活以及与他人进行交流和互动。

计算机科学在本质上源自数学思维,它的形式化基础建筑于数学之上。计算机科学又从本质上源自工程思维,因为人们建造的是能够与现实世界互动的系统,所以计算思维是数学与工程思维的互补与融合。

计算思维无处不在,当计算思维真正融入人类活动的整体时,它作为一个解决问题的有效工具,人人都应掌握它,处处都会用到它。它应当有效地融入学生们每一堂课之中。

1.1.4 计算思维与计算机的关系

计算思维虽然具有计算机科学的许多特征,但是计算思维本身并不是计算机科学的专属。实际上,即使没有计算机,计算思维也会逐步发展,甚至有些内容与计算机没有关系,但是,正是由于计算机的出现,给计算思维的研究和发展带来了根本性的变化。计算机对于信息和符号的快速处理能力,使得许多原本只是从理论上可以实现的过程变成了实际中也可以实现的过程。

什么是计算?什么是可计算?什么是可行计算?计算思维的这些性质得到了前所未有的深入研究。由此不仅推进了计算机的发展,也推进了计算思维本身的发展。在这个过程中,一些属于计算思维的特点被逐步揭示出来,计算思维与逻辑思维、实证思维的差别越来越清晰。计算思维的概念、结构、格式等变得越来越明确。计算思维的内容得到不断的丰富

和发展。

计算机的出现丰富了人类改造世界的手段,同时也强化了原本存在于人类思维中的计算思维的意义和作用。从思维的角度来看,计算机科学主要研究计算思维的概念、方法和内容,并发展成为解决问题的一种思维方式,这极大地推动了计算思维的发展。

1.2 C语言概述

1.2.1 计算机语言发展史

在揭开C语言的神秘面纱之前,我们先来认识一下什么是计算机语言。计算机语言(computer language)是人与计算机之间通信的语言,它主要由一些指令组成,这些指令包括数字、符号和语法等,编程人员可以通过这些指令来指挥计算机进行各种工作。

计算机语言有很多种类,根据功能和实现方式的不同大致可分为三大类,即机器语言、汇编语言和高级语言,下面我们针对这3类语言的特点进行简单介绍。

1. 机器语言

计算机不需要翻译就能直接识别的语言被称为机器语言(又被称为二进制代码语言),该语言是由二进制数0或1组成的一串指令,对于编程人员来说,机器语言不便于记忆和识别。

2. 汇编语言

人们很早就认识到这样一个事实,尽管机器语言对计算机来说很好懂也很好用,但是对于编程人员来说记住由0和1组成的指令是件艰难的事。为了解决这个问题,汇编语言诞生了。汇编语言用英文字母或符号串来替代机器语言,把不易理解和记忆的机器语言按照对应关系转换成汇编指令,因此,汇编语言比机器语言更加便于阅读和理解。编译器可以把写好的汇编语言程序翻译成机器语言程序,以实现和计算机的沟通。

3. 高级语言

由于汇编语言依赖于硬件,程序的可移植性差,而且编程人员在使用新的计算机时还需要学习新的汇编指令,大大增加了编程人员的工作量,为此计算机高级语言诞生了。高级语言不是一种语言,而是一类语言的统称,它比汇编语言更贴近于人类使用的语言,易于理解、记忆和使用。由于高级语言和计算机的架构、指令集无关,因此它具有良好的可移植性。

高级语言的应用非常广泛,世界上绝大多数编程人员都在使用高级语言进行程序开发。常见的高级语言包括C、C++、Java、VB、C#、Python等。本书讲解的C语言就是目前最流行、应用最广泛的高级语言之一,也是计算机高级编程语言的元老。

1.2.2 什么是C语言

C语言是种高级程序设计语言,具有简洁、紧凑、高效等特点。它既可以用于编写应用程序,也可以用于编写系统软件。自1973年问世以来,C语言迅速发展并成为最受欢迎的编程语言之一,下面我们将针对C语言的发展史和C语言标准分别进行讲解。

1. C语言的发展史

早期的系统软件设计均采用汇编语言,例如大家熟知的 UNIX 操作系统。尽管汇编语言在可移植性、可维护性等方面远不及高级语言,但是一般的高级语言有时难以实现汇编语言的某些功能。那么,能否设计出一种集汇编语言和高级语言优点于一身的语言呢?于是,C语言就应运而生了。

C语言的发展颇为有趣,它的原型是 ALGOL 60 语言(也称 A 语言)。

1963年,剑桥大学将 ALGOL 60 语言发展成为 CPL,CPL 的全称为 Combined Programming Language。

1967年,剑桥大学的马丁·理查兹(Martin Richards)对 CPL 进行了简化,于是产生了 BCPL。

1970年,美国贝尔实验室的肯·汤普森(Ken Thompson)将 BCPL 进行了修改,并为它起了一个有趣的名字“B语言”,其含义是将 CPL“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。

1973年,美国贝尔实验室的丹尼斯·里奇(Dennis M. Ritchie)在 B语言的基础上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,即 C语言。

1978年,布赖恩·凯尼汉(Brian W. Kernighan)和丹尼斯·里奇(Dennis M. Ritchie)出版了 *THE C PROGRAMMING LANGUAGE*,从而使 C语言成为目前世界上广泛应用的高级程序设计语言。

2. C语言标准

随着微型计算机的日益普及,市面上出现了许多 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,美国国家标准学会(ANSI)为 C语言制定了一套 ANSI 标准,即 C语言标准。

人们将 1989年美国国家标准学会(ANSI)通过的 C语言标准 ANSI X3.159—1989 称为 C89。之后在 1990年,国际标准化组织(ISO)也通过了同样的标准 ISO 9899—1990,该标准被称为 C90。这两个标准只有细微的差别,因此,通常来讲 C89 和 C90 指的是同一个版本。

后来随着时代的发展,1999年 ANSI 又通过了 C99 标准。C99 标准相对 C89 做了很多修改,例如,变量声明可以不放在函数开头,支持变长数组等,但由于很多编译器仍然没有对 C99 提供完整的支持,因此本书将按照 C89 标准来进行讲解,在适当时会补充 C99 标准的规定和用法。

1.2.3 C语言的特点

C语言是一种通用的、面向过程的程序语言。它的诸多特点使它的应用面很广,下面我们简单学习一下 C语言的特点。

1. 语言简洁,使用方便灵活

C语言是现有程序设计语言中规模最小的语言之一,它仅有 32 个关键字,9 种控制语句,压缩了一切不必要的成分。其 32 个关键字与 9 种控制语句在后续章节中我们会陆续学习。

2. 结构化程序设计

C语言是面向过程的语言,它以函数作为程序设计的基本单位,具有自定义函数的功能,因此使用C语言可以很容易地进行结构化程序设计。

3. 能进行硬件操作

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,C语言的这种双重性使它既是成功的系统描述语言,又是通用的程序设计语言。

4. 执行速度快

众所周知,汇编语言程序目标代码是效率最高的,而C语言的目标代码效率仅比汇编语言低10%~20%。

尽管C语言具有很多的优点,但它和其他任何一种程序设计语言一样,也有其自身的缺点,如编写代码实现周期长、可移植性较差、过于自由、易出错、对平台库依赖性较大等。但总的来说,C语言的优点远远超过了它的缺点。

1.2.4 C语言的发展趋势

从20世纪70年代起,C语言通过UNIX操作系统迅速发展起来,逐渐在大型机、中型机、小型机,以及微型机中得到应用,成为风靡世界的计算机语言。大多数软件开发商都优先选择C语言来开发系统软件、应用程序、编译器和其他产品。

这样的现象一直保持了很多年,直到一种代表着先进思想的语言问世。C++语言的诞生解决了C语言不能解决的诸多难题,许多开发商开始使用C++来开发一些复杂的、规模较大的项目,因此,C语言进入一个冷落时期。

这个冷落时期并没有持续太长时间,随着嵌入式产品的增多,C语言简洁高效的特点又重新被重视起来,其强大的功能被广泛应用于各领域。

(1) C语言可以写网站后台程序,诸如百度、腾讯后台等。

(2) C语言可以写出绚丽的GUI界面。

(3) C语言可以专门针对某个主题写出功能强大的程序库,然后供其他程序使用,从而让其他程序节省开发时间。

(4) C语言可以写出大型游戏的引擎。

(5) C语言可以写操作系统和驱动程序,并且这两者只能用C语言编写。例如,用C语言编写的Linux操作系统的全部源代码都可以从网上得到,而要深入了解操作系统运行的秘密,只要懂得C语言即可。

(6) 任何设备只要配置了微处理器,就都支持C语言。

1.3 C语言开发环境的搭建

在使用C语言开发程序之前,首先要在操作系统上搭建开发环境。目前程序员大多使用集成开发工具来开发C语言程序,适合C语言的集成开发工具有很多,常用的有Turbo C、Microsoft C、Visual C++、Borland C++、C++Builder等。本书主要介绍在Visual C++ 6.0环境下的程序开发,书中的程序都在Visual C++ 6.0环境下调试通过。