第一图书网, tushu007.com

<<C语言程序设计>>

图书基本信息

书名：<<C语言程序设计>>

13位ISBN编号：9787030259578

10位ISBN编号：7030259572

出版时间：1970-1

出版时间：科学出版社

作者：李刚健 等 著

页数：270

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

前言

自20世纪80年代以来，高等学校计算机教育发展迅速，计算机教育的内容不断扩展、程度不断加深。
特别是近十余年来，计算机向高度集成化、网络化和多媒体化发展的速度一日千里。
社会信息化不断向纵深发展，各行各业的信息化进程不断加速。
计算机应用技术与其他专业的教学、科研工作的结合更加紧密。
各学科与以计算机技术为核心的信息技术的融合，促进了计算机学科的发展，各专业对学生的计算机应用能力也有更高和更加深入的要求。
基于近年来计算机科学的发展以及教育部关于计算机基础教学改革的指导思路，我们确立了本书的编写思想。
本书的所有编者均为一线教师，根据长期在C语言程序设计教学中所遇到的问题有针对性地编写了此书。
另外，我们在国内高校做了系统、详细的调研。
对教育部制订的教学计划做了认真的研究，还对国内外已出版的教材做了理性的分析，确立了依托国家教学计划、传播先进教学理念、为培养符合社会需要的高素质创新型、应用型人才服务的创作宗旨。
在本书的策划过程中，我们组织了多次研讨会，对现有比较出色的教材的特点及优点进行了分析，博采众长，力求实现教材权威性与实用性的完美结合。
与目前出版的C语言程序设计教材相比，本书的主要特点是：充分考虑到工科院校学生知识、能力、素质的特点和实际教学情况，增强了实用性。
采用案例式教学法，以编程应用为驱动，教材内容经过精心组织，体系合理、结构严谨，由浅入深、循序渐进地讲解C语言程序设计的思想和方法。
每章均精心设计了由浅入深的例题，能够让学生充分地理解相关知识点，并通过每章的课后习题巩固、加深对基本概念的理解和掌握，提高C语言程序设计的水平。
由于函数是C语言中统领全局的重要概念，我们将函数的内容提到数组和指针之前来讲解，并在后边的相关章节中反复强化函数的概念和使用。
经验证明，这将更有利于学生对C语言程序设计完整的理解和把握。
本书可作为工科院校相关专业本、专科学生程序设计课程教材，也可供程序设计爱好者自学和参考。
教育的改革不会停止，教材也将不断地推陈出新。
本书将接受广大教学第一线教师的检验。
由于我们的水平和经验有限，本书在编审、出版工作中还存在不少缺点和不足，欢迎使用本书的广大读者提出批评与建议，以便改进我们的工作，使教材质量不断提高。

内容概要

《C语言程序设计》全面介绍了C语言的体系、概念、语法和语义、特点及结构化程序设计方法。全书共13章，第1章介绍了C语言程序设计的基础知识；第2、3章介绍了C语言的基本数据类型、常量和变量以及表达式；第4～6章介绍了用C语言进行结构化程序设计的基本方法，包括结构化程序的顺序结构、选择结构、循环结构及其设计方法；第7、10章分别介绍了函数及编译预处理相关知识；第8、9章对C语言的数组、指针作了详尽的阐述；第11～13章分别介绍结构体与共用体、文件、位运算相关内容。
全书以编程应用为驱动，通过案例和问题引入内容。
每一章都有由浅入深的程序范例，以尽可能详尽地解释相关语法的概念、作用、含义和使用方法，强调对C语言基础知识的理解和掌握，注重培养学生的程序设计综合素质和解决实际问题的能力。
《C语言程序设计》可作为工科院校相关专业本科生、专科生的程序设计课程教材，也可供程序设计爱好者自学和参考。

<<C语言程序设计>>

书籍目录

前言第1章 C语言概述11.1 C语言的发展与特点11.1.1 C语言的发展11.1.2 C语言的特点21.2 程序设计的基本概念41.2.1 程序41.2.2 程序设计41.2.3 算法51.2.4 数据结构61.3 C语言的字符集与标识符61.4 C程序举例71.5 本章小结11习题11第2章 数据类型122.1 C的数据类型122.2 常量122.2.1 整型常量122.2.2 实型常量132.2.3 字符常量132.2.4 字符串常量152.2.5 符号常量152.3 变量162.3.1 整型变量172.3.2 实型变量202.3.3 字符变量212.3.4 变量赋初值222.4 库函数的使用232.5 本章小结24习题24第3章 C语言的运算符和表达式253.1 C语言运算符简介253.1.1 C语言运算符的种类及功能253.1.2 C语言运算符的优先级及结合性273.2 C语言的运算符283.2.1 算术运算符283.2.2 自增与自减运算符293.3 算术表达式313.3.1 算术表达式的运算313.3.2 算术表达式的书写规则323.4 表达式中数据间的混合运算与类型转换323.5 赋值运算符及复合赋值运算符353.5.1 赋值运算符353.5.2 复合赋值运算符363.6 逗号运算符及逗号表达式373.7 程序举例373.8 本章小结38习题39第4章 顺序结构程序设计404.1 结构化程序设计简介404.2 求华氏100°F对应的摄氏温度程序的实现414.2.1 程序解析414.2.2 C程序中的语句424.3 赋值语句444.4 数据的输入/输出454.4.1 字符输入/输出函数454.4.2 格式输出函数474.4.3 格式输入函数524.5 程序举例544.6 本章小结58习题58第5章 分支结构程序设计595.1 实数排序程序的实现595.1.1 程序解析595.1.2 if语句595.1.3 if…else语句615.1.4 if语句的嵌套625.2 简单英文星期转换程序的实现675.2.1 程序解析675.2.2 switch语句685.2.3 break语句在switch语句中的作用705.3 程序举例735.4 本章小结81习题81第6章 循环结构程序设计836.1 自然数1到100求和程序的实现836.1.1 程序解析836.1.2 while语句836.2 do…while语句856.3 for语句876.4 break和continue语句916.4.1 break语句916.4.2 continue语句926.5 循环的嵌套926.6 本章小结94习题94第7章 函数957.1 自然数1到100求和程序的实现957.1.1 程序解析957.1.2 函数的定义967.2 函数参数与函数值987.2.1 形式参数和实际参数987.2.2 函数的返回值1007.3 函数的调用1017.3.1 函数调用的方式1017.3.2 对被调用函数的声明1027.3.3 函数的嵌套调用1067.3.4 函数的递归调用1077.4 变量的作用域1117.4.1 局部变量1117.4.2 全局变量1137.5 变量的存储类型1147.5.1 变量的动态与静态存储方式1157.5.2 局部变量的存储类型1157.5.3 全局变量的存储类型1177.6 内部函数与外部函数1187.7 本章小结119习题119第8章 数组1208.1 输出fibonacci数列的前20项程序的实现1208.1.1 程序解析1208.1.2 一维数组的定义及初始化1218.1.3 一维数组的使用1228.2 找出矩阵中最大值所在的位置1258.2.1 程序解析1258.2.2 二维数组的概念1268.2.3 二维数组的定义1278.2.4 多维数组的定义1288.2.5 二维数组及多维数组的初始化1308.3 字符数组与字符串1338.3.1 程序解析1338.3.2 字符数组及其初始化1338.3.3 字符串的输入1368.3.4 字符串的输出1378.3.5 二维字符数组1378.4 数组作为函数的参数1438.4.1 数组元素作为函数的参数1438.4.2 数组名作为函数的参数1448.5 程序举例1458.6 本章小结150习题150第9章 指针1519.1 寻找保险箱密码程序的实现1519.1.1 程序解析1519.1.2 指针的概念1529.1.3 指针变量的定义与初始化1549.1.4 指针运算1569.1.5 多级指针1599.2 指针与函数1609.2.1 指针作为函数参数1609.2.2 指针函数1639.2.3 指向函数的指针1649.3 指针与数组1679.3.1 指向一维数组的指针1689.3.2 二维数组与多维数组的指针表示法1709.4 指针与字符串1769.4.1 字符串的指针表示法1769.4.2 字符串数组1789.5 指针数组与命令行参数1789.5.1 指针数组178……第10章 编译预处理命令188第11章 结构体与共用体196第12章 文件223第13章 位运算244附录A ASCII码表259附录B C语言常用库函数260附录C C语言的运行环境和运行过程265参考文献271

章节摘录

机器语言是计算机真正“理解”并识别的唯一语言。
而汇编语言是符号化的机器语言，它是用符号来表示每一条指令和地址，和机器语言相比，汇编语言指令的含义比较直观，也易于阅读和理解。
机器语言和汇编语言都是面向机器的，都与具体机器的硬件系统相关，因此又称为“低级语言”。
低级语言编写的程序可移植性差，抽象水平低，较难编写和理解，于是后来又出现了高级语言。
高级语言是面向问题的语言，独立于具体的机器，比较接近于人类的语言习惯和数学表达形式，如目前绝大多数高级语言都是用简单的英语表达。
高级语言与计算机结构无关，便于学习和使用，具有更强大的表达能力，高级语言写成的程序可移植性强，便于推广。
目前绝大多数程序设计语言如C语言、Pascal语言都是高级语言，绝大多数程序员也是使用高级语言。
有了程序设计语言后，程序设计就有了工具，程序设计才成为可能，但是程序设计语言并不能保证高质量的程序或者软件，程序设计需要方法学和理论上的指导。
在程序设计的早期，对一个可解的问题（当时是较简单的问题），常常由一两个人包揽，因此程序设计技术不免被看成是一种与个人思想、经验和技术相联系的技巧。
然而，随着软件的规模越来越大，越来越复杂，这种手工式的程序设计方法越来越不能满足要求，20世纪60年代末的软件危机是这种矛盾的集中爆发。
所谓“软件危机”，是指当时一方面需要大量的软件系统，如操作系统、数据库管理系统；另一方面，软件研制周期长，可靠性差，维护困难。
在这种背景下，1968年，北大西洋公约组织（NATO）在前联邦德国召开了第一次软件工程会议，分析了危机的局面，研究了问题的根源，第一次提出了用工程学的办法解决软件研制和生产的问题，本次会议可以算作是软件发展史上的一个重要的里程碑。
1969年，国际信息处理协会（IFIP）成立了“程序设计方法学工作组”，专门研究程序设计方法学，程序设计从手工艺式向工程化的方法迈进。
从那时开始，人们开始意识到程序设计是一门技术科学，从那以后，程序设计方法与程序设计技术取得了很大的进展，开始出现了结构化的方法、面向对象的方法等程序设计方法。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com