

高等院校计算机系列教材

C 语言程序设计

主 编 杨克昌 羊四清 周克江
副主编 王岳斌 戴经国 朱承璋
王智河 龙 平

E m a i l



WUHAN UNIVERSITY PRESS

武汉大学出版社

高等院校计算机系列教材

C 语言程序设计

主 编 杨克昌 羊四清 周克江

副主编 王岳斌 戴经国 朱承璋

王智河 龙 平



WUHAN UNIVERSITY PRESS

武汉大学出版社

图书在版编目(CIP)数据

C 语言程序设计/杨克昌,羊四清,周克江主编. —武汉:武汉大学出版社,2007.1

(高等院校计算机系列教材)

ISBN 978-7-307-05409-7

I. C… II. ①杨… ②羊… ③周… III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 163712 号

责任编辑:林 莉

责任校对:黄添生

版式设计:支 笛

出版发行:武汉大学出版社 (430072 武昌 珞珈山)

(电子邮件:wdp4@whu.edu.cn 网址:www.wdp.com.cn)

印刷:湖北省新华印务有限公司

开本:787×1092 1/16 印张:20.625 字数:491千字

版次:2007年1月第1版 2007年1月第1次印刷

ISBN 978-7-307-05409-7/TP·234 定价:29.00元

版权所有,不得翻印;凡购买我社的图书,如有缺页、倒页、脱页等质量问题,请与当地图书销售部门联系调换。

高等院校计算机系列教材书目

计算机网络	李永帆等
数据库系统原理与应用	刘先锋等
C++ 程序设计教程	刘 宏等
计算机组成与系统结构	陈书开等
计算机操作系统	王志刚等
Delphi 2005 数据库基础教程	徐长梅等
网页制作与网站设计	阳西述等
C 语言程序设计(公共课)	杨克昌等

高等院校计算机系列教材

编 委 会

- 主 任：刘 宏，湖南师范大学数学与计算机科学学院计算机系主任，教授
编 委：（以姓氏笔画为序）
- 王 毅，湘潭大学信息工程学院副院长，副教授
乐晓波，长沙理工大学计算机与通信工程学院计算机科学与技术系主任，教授
许又泉，邵阳学院信息电气工程系
羊四清，湖南人文科技学院计算机系主任，副教授
刘先锋，湖南师范大学数学与计算机科学学院，教授
刘连浩，中南大学信息工程学院计算机系教授
全惠云，湖南师范大学数学与计算机科学学院信息与计算科学系主任，教授
沈 岳，湖南农业大学信息科学技术学院院长，副教授
张小梅，凯里学院数学与计算机科学系副主任，副教授
杨克昌，湖南理工学院计算机与信息工程系教授
何迎生，吉首大学数学与计算机科学学院信息与计算科学系副主任
邱建雄，长沙学院计算机科学与技术系副教授
李勇帆，湖南第一师范学院信息技术系主任，教授
周 昱，吉首大学师范学院数学与计算机科学系副主任
罗新密，湖南商学院计算机与电子工程系副教授
徐雨明，衡阳师范学院计算机科学系副主任，副教授
郭国强，湖南文理学院计算机科学与技术系主任，教授
晏峻峰，湖南中医药大学计算机系副教授
龚德良，湘南学院计算机科学系副主任，副教授

蒋伟进，湖南工业大学计算机科学与技术系副主任，副教授

熊江，重庆三峡学院数学与计算机科学学院副教授

谭敏生，南华大学计算机学院院长，副教授

戴祖雄，湖南科技大学计算机科学与工程学院

执行编委：黄金文，武汉大学出版社计算机图书事业部主任，副编审



内 容 提 要

C 语言是一种结构化的计算机程序设计语言,也是当前大学生学习程序设计的首选语言之一,在软件开发领域有着十分广泛的应用。

本书根据初学者的认知特点,循序渐进,紧贴教学,深入浅出,通俗易懂,讲述了 C 语言的基本概念、数据类型与语句的使用、模块化程序设计的方法以及文件的基本操作,注重培养读者良好的程序设计风格,提高程序设计能力。每章后附有适量练习题,题目类型与全国计算机等级考试接轨。根据教学进程,在实验部分安排了十个上机实验。

本书可作为大专院校非计算机专业的 C 语言程序设计教材,也可供各类计算机培训或其他自学者使用。



前言

进入 21 世纪,以计算机为核心的信息科学的迅猛发展与广泛应用,正在对人类社会的发展进程以至人们的工作方式与思维方式的改变产生深远的影响。随着计算机技术广泛应用于信息社会的各个领域,推动社会信息化进程的不断深入,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。

顺应信息技术不断发展与计算机教育不断深入的潮流,帮助引导包括各大中专在校学生在内的广大青少年逐步掌握计算机的基本理论与基本技能,在程序设计中熟悉基本算法,开拓求解思路,解决实际问题,培养创新意识,不断提高程序设计的水平与应用计算机解决实际问题的能力,是我们计算机教育工作者义不容辞的职责。

作为第三代计算机程序设计语言的代表,C 语言功能强大,表达能力丰富,使用简洁灵活,应用面广,目标程序效率高,可移植性好,既具有高级语言的优势,又具有汇编语言的特点,是当前大学生学习程序设计的首选语种,同时也是全国计算机等级考试、全国计算机应用证书考试与全国各类程序设计竞赛中使用最多的计算机程序设计语种之一。考虑到 C 语言规则繁多,概念复杂,使用灵活,容易出错,我们针对初学者的认知特点,在材料组织上力求做到:层次结构清晰,概念解析简洁,例题习题精选,重点难点突出,紧贴教学进程,简明方便实用。

本书简洁、全面介绍了 C 语言的概念、规则、特点与结构化程序设计的基本方法。全书共十二章,第一章介绍 C 语言与程序设计的基本知识。第二章讲述 C 语言的数据类型、常量、变量与表达式。第三、四、五章介绍了 C 语言结构化程序设计的基本方法,包括顺序结构、分支结构与循环结构程序及其设计方法。第六、七、八章分别对数组、函数与指针进行了充分阐述。第九章介绍了结构体与共用体的概念与应用。第十章介绍了 C 语言的文件操作。为了指导提高读者应用 C 程序设计解决实际问题的能力,在第十一章中选编了八个有一定难度的 C 程序设计综合应用实例,这些问题也可以作为 C 语言课程设计的题材。同时,为了读者对 C 语言的扩充有所了解,在第十二章简要介绍了 C++ 的概念与特点。

每章配置的例题为紧贴所说明的结构与概念的基础题,并具有一定的梯度。同时,每章后附有适量练习题,题目类型与全国计算机等级考试接轨。

C 程序设计是一门实践性很强的课程,既要掌握 C 语言的概念与语法规则,又要动手编程与上机调试。为了方便程序设计上机实验,根据教学进程,在实验部分设计编排了十个上机实验。

本书由杨克昌、羊四清、周克江主编。周克江编写第一、二章,朱承璋编写第三、四章,杨克昌编写第五章,杨克昌、龙平编写十一章,王岳斌编写第六、十二章,羊四清编写第七、八章,戴经国编写第九章,王智河编写第十章。

由于作者水平有限,书中错误与不当之处在所难免,敬请读者批评指正。

作者

2006 年 12 月

目 录

第一章 C 语言程序设计概述	1
1.1 程序与程序设计语言	1
1.1.1 程序的基本概念	1
1.1.2 程序设计语言	1
1.2 算法与算法描述	2
1.2.1 算法的概念	2
1.2.2 算法的描述	3
1.3 C 语言的发展与特点	5
1.3.1 C 语言的发展概况	5
1.3.2 C 语言程序的基本结构及其特点	6
1.4 C 语言字符集、标识符与关键字	8
1.4.1 C 语言字符集	9
1.4.2 标识符与关键字	9
1.4.3 C 程序的书写风格	10
1.5 C 程序的开发环境	11
1.5.1 C 程序的开发过程	11
1.5.2 TC 2.0 介绍	11
1.5.3 C 程序的上机步骤	14
习题一	15
第二章 基本数据类型与表达式	17
2.1 C 语言的基本数据类型	17
2.1.1 数据类型概述	18
2.1.2 整型	20
2.1.3 实型	21
2.1.4 字符型	21
2.2 常量与变量	22
2.2.1 常量	23
2.2.2 变量	23
2.3 运算符与表达式	25
2.3.1 算术运算符与算术表达式	25
2.3.2 赋值运算符与赋值表达式	27
2.3.3 逗号表达式	29
2.3.4 条件运算符	30



2.4 数据类型转换	31
2.4.1 类型自动转换	31
2.4.2 赋值转换	33
2.4.3 强制类型转换	33
习题二	34
第三章 输入输出和顺序结构程序设计	36
3.1 C 语言的基本语句	36
3.1.1 简单语句	36
3.1.2 复合语句	37
3.2 格式输入输出函数	38
3.2.1 格式输入函数 (scanf 函数)	38
3.2.2 格式输出函数 (printf 函数)	41
3.3 字符输入输出函数	46
3.3.1 单个字符输入函数 (getchar 函数)	46
3.3.2 单个字符输出函数 (putchar 函数)	47
3.4 顺序结构程序设计举例	48
习题三	49
第四章 选择结构程序设计	52
4.1 关系运算符和关系表达式	52
4.1.1 关系运算符及其优先次序	52
4.1.2 关系表达式	52
4.2 逻辑运算符和逻辑表达式	53
4.2.1 逻辑运算符及其优先次序	53
4.2.2 逻辑表达式	54
4.3 if 语句	56
4.3.1 if 语句的三种格式	56
4.3.2 if 语句的嵌套	58
4.3.3 条件运算符	59
4.4 switch 语句	60
4.5 选择结构程序设计举例	62
习题四	65
第五章 循环结构程序设计	67
5.1 while 循环	67
5.2 do-while 循环	68
5.3 for 循环	70
5.4 break、continue 与 goto 语句	72
5.4.1 break 语句	72
5.4.2 continue 语句	73
5.4.3 goto 语句	73
5.5 循环嵌套	74



5.6 几种循环结构比较	75
5.7 循环结构程序设计应用举例	75
习题五	79
第六章 数组	82
6.1 一维数组	82
6.1.1 一维数组的定义	82
6.1.2 一维数组元素的引用	83
6.1.3 一维数组的存储与初始化	84
6.2 二维数组	86
6.2.1 二维数组的定义	86
6.2.2 二维数组元素的引用	87
6.2.3 二维数组的存储与初始化	88
6.3 多维数组	90
6.3.1 多维数组的定义、引用与存储	90
6.3.2 多维数组的初始化	90
6.4 数组的应用	91
6.5 字符数组与字符串	98
6.5.1 字符数组的定义与初始化	99
6.5.2 字符串的概念及存储	99
6.5.3 字符串的输入与输出	100
6.5.4 字符串处理函数	104
6.6 典型例题分析	106
习题六	109
第七章 函数和编译预处理	114
7.1 程序结构与函数的定义	114
7.1.1 C语言程序的结构	114
7.1.2 函数的分类	115
7.1.3 函数的定义	116
7.2 函数的调用与执行	118
7.2.1 函数的参数与参数传递	118
7.2.2 函数的返回值	120
7.2.3 函数的调用方式	120
7.2.4 函数的执行	121
7.3 数组作为函数参数	122
7.3.1 数组元素作函数参数	122
7.3.2 数组名作为函数参数	123
7.4 函数的嵌套调用与递归	126
7.4.1 函数的嵌套调用	126
7.4.2 函数的递归调用	126
7.5 局部变量和全局变量	131



7.5.1 局部变量	131
7.5.2 全局变量	133
7.6 变量的存储	135
7.6.1 变量存储方式	135
7.6.2 变量的存储说明	135
7.7 编译预处理	137
7.7.1 预处理命令	138
7.7.2 宏定义	138
7.7.3 文件包含	142
7.7.4 条件编译	142
习题七	144
第八章 指针	146
8.1 指针的定义	146
8.1.1 地址	146
8.1.2 指针	147
8.2 指针变量	148
8.2.1 指针变量的定义	148
8.2.2 指针变量的赋值	149
8.2.3 指针变量的运算	151
8.2.4 指针变量的进一步说明	158
8.3 指针与数组	159
8.3.1 指向数组元素的指针	159
8.3.2 数组指针	160
8.3.3 指针数组	164
8.4 指针与字符串	165
8.4.1 字符串的表示	165
8.4.2 字符串指针变量与字符数组的区别	167
8.4.3 用指针数组表示字符串组	168
8.5 指针变量作为函数参数	169
8.5.1 形参是指针变量,实参是变量的地址	169
8.5.2 形参是指针变量,实参是指针变量	171
8.5.3 数组或指针变量作为函数的参数	172
8.5.4 指针数组作为函数的参数	177
8.6 函数指针变量与指针型函数	178
8.6.1 函数指针变量	178
8.6.2 指针型函数	179
8.7 二重指针与带参数的 main 函数	180
8.7.1 二重指针与指针数组的关系	180
8.7.2 main 函数的参数	182
习题八	183



第九章 结构体与共用体	184
9.1 结构体	184
9.1.1 结构体的声明	184
9.1.2 结构体变量的引用及初始化赋值	186
9.1.3 嵌套结构体	186
9.2 结构体数组	188
9.2.1 结构体数组的定义和初始化	188
9.2.2 结构体数组成员的引用	189
9.3 结构体指针	189
9.3.1 指向结构体变量的指针	189
9.3.2 指向结构体数组的指针	191
9.3.3 用结构体变量和指向结构体变量的指针作为函数参数	192
9.4 链表的基本操作	193
9.4.1 链表基本知识	193
9.4.2 内存动态管理函数	194
9.4.3 建立链表	195
9.4.4 输出链表	197
9.4.5 对链表的删除操作	197
9.4.6 对链表的插入操作	199
9.4.7 对链表的综合操作	201
9.5 共用体	202
9.5.1 共用体的概念	202
9.5.2 共用型变量的定义	203
9.5.3 共用型变量的引用	204
9.5.4 共用体类型数据的特点	204
9.5.5 共用体变量的应用	205
9.6 枚举类型	206
9.7 用 typedef 定义	209
9.8 典型例题分析	210
习题九	210
第十章 文件和位运算	213
10.1 C 文件的概念	213
10.2 文件类型指针	214
10.3 文件操作函数	214
10.3.1 文件的打开与关闭	214
10.3.2 读写字符函数	216
10.3.3 读写字符串函数	217
10.3.4 读写数据块函数	219
10.3.5 格式化读写函数 (fprintf 函数和 fscanf 函数)	221
10.4 文件的定位	222



10.4.1	移动位置指针函数 (fseek)	222
10.4.2	确定当前位置函数 (ftell)	223
10.4.3	位置复位函数 (rewind)	223
10.5	文件检测函数	223
10.6	位运算和位段结构	224
10.6.1	位运算	224
10.6.2	位段结构	226
10.7	典型例题分析	228
	习题十	230
第十一章	程序设计综合应用	232
11.1	勾股数	232
11.2	高斯八皇后问题	233
11.3	质因数分解	235
11.3.1	乘积形式分解	235
11.3.2	质因数指数形式分解	236
11.4	素数探求与合数世纪	237
11.4.1	探求区间素数	238
11.4.2	合数世纪	239
11.4.3	最小的连续 n 个合数	240
11.5	优美等式	241
11.5.1	优美乘积	241
11.5.2	优美和	242
11.5.3	桥本分式	243
11.6	泊松分酒	245
11.7	外索夫游戏	247
11.8	流水演示	250
	习题十一	253
第十二章	C++ 简介	254
12.1	C++ 概述	254
12.2	C++ 对 C 的扩充	256
12.2.1	C++ 的输入输出	256
12.2.2	C++ 的行注释	260
12.2.3	Const 常量	260
12.2.4	变量的引用	262
12.2.5	new 与 delete 算符	266
12.3	C++ 面向对象程序设计简介	268
12.3.1	类与对象概述	269
12.3.2	类的使用	271
	习题十二	272
	实验	275

实验一 熟悉 C 程序开发环境	275
实验二 数据描述	276
实验三 顺序结构程序设计	278
实验四 选择结构程序设计	280
实验五 循环结构程序设计	282
实验六 数组	286
实验七 函数	289
实验八 指针	291
实验九 结构体与共用体	294
实验十 文件与位运算	297
附录	300
附录一 ASCII 码表	300
附录二 C 语言运算符和结合性一览表	301
附录三 C 语言常用语法提要	302
附录四 常用 C 库函数	305
附录五 编译、连接与运行时出错提示信息	308
主要参考文献	312



第一章 C语言程序设计概述

C语言是一种面向过程的通用程序设计语言。它以表达简明、使用灵活、结构化的流程控制、丰富的数据结构和操作符集合、良好的程序可移植性和高效率的目标代码为特征。C语言不仅具有高级语言的要素,还兼有低级语言的功能,因此既可用于编写系统程序,也可用于编写不同领域的应用程序。本章主要介绍程序设计语言、算法的基本概念以及C程序的基本结构和书写风格、简单C程序的编制、Turbo C 2.0(以下简称TC)集成环境的使用方法。

1.1 程序与程序设计语言

根据要求编写出正确的程序,是我们学习一种程序设计语言的目的。这里我们先简单介绍一下这两个概念。

1.1.1 程序的基本概念

程序(Program)是指用某种计算机语言编写的能完成特定功能的指令模块。程序的产生分成五个阶段:

(1)需求(Requirements):充分了解所提供的信息(输入),以及将要产生的结果(输出),对所有输入及输出进行严密的描述。

(2)设计(Design):根据需求所得的数据集(如迷宫、多项式或者是姓名的字符串等),编写算法解决问题。算法不必拘泥于形式,可以用所熟悉的伪语言、图或文字来表示即可。

(3)分析(Analysis):针对问题提出其他解决方案,最后在所有的算法中挑选最佳者。

(4)细化与编码(Refinement and Coding):细化、修改所选择的算法,配合使用的程序语言特性,编写出程序的初稿。

(5)验证(Verification):验证的工作包含程序验证(Program Proving)、测试(Testing)及调试(Debugging)三部分。

1.1.2 程序设计语言

程序设计语言(Programming Language)是用于编写计算机程序的语言。语言的基础是一组记号和一组规则。根据规则由记号构成的记号串的总体就是语言。在程序设计语言中,这些记号串就是程序。程序设计语言包含三个方面,即语法、语义和语用。语法表示程序的结构或形式,亦即表示构成程序的各个记号之间的组合规则,但不涉及这些记号的特定含义,也不涉及使用者。语义表示程序的含义,亦即表示按照各种方法所表示的各个记号的特定含义,但也不涉及使用者。语用表示程序与使用的关系。

程序设计语言的基本成分有:

(1)数据成分:用于描述程序所涉及的数据;

- (2) 运算成分:用以描述程序中所包含的运算;
- (3) 控制成分:用以描述程序中所包含的控制;
- (4) 传输成分:用以表达程序中数据的传输。

程序设计语言按照语言级别可以分为低级语言和高级语言。低级语言有机器语言和汇编语言。低级语言与特定的机器有关、功效高,但使用复杂、繁琐、费时、易出差错。机器语言是表示成数码形式的机器基本指令集,或者是操作码经过符号化的基本指令集。汇编语言是机器语言中地址部分符号化的结果,或进一步包括宏构造。高级语言的表示方法要比低级语言更接近于待解问题的表示方法,其特点是在一定程度上与具体机器无关,易学、易用、易维护。

程序设计语言按照用户的要求有过程式语言和非过程式语言之分。过程式语言的主要特征是,用户可以指明一系列可顺序执行的运算,以表示相应的计算过程,如 FORTRAN、COBOL、PASCAL 等。

1.2 算法与算法描述

当程序设计人员对问题的描述清楚后,想要利用计算机来解决问题时,就得编写程序,但如果直接开始写程序可能有些复杂,所以一般都是先由算法(Algorithm)着手,再配合适当的数据结构,就能很容易地写出程序。所谓算法是指在有限的时间范围内,解决某一问题的一系列逻辑步骤。所以,当问题被确定后,将问题分解成几个小部分,针对每个小部分先写出伪代码(Pseudo Code),再将代表每部分的伪代码逐一转换成程序。这里的每部分都可视为子程序,而子程序间的调用可由主程序或其子程序来完成。所有的算法都必须满足下列几个条件:

- (1) 输入(Input):可以有一个以上的输入数据,也可以不需由外界提供输入数据。
- (2) 输出(Output):至少须产生一个结果,此结果与输入的数据构成某种特定的关系。
- (3) 确定性(Definiteness):每条指令皆须十分明确,有确切的含义,并且在任何情况下,对于相同的输入只能得出相同的输出。
- (4) 有限性(Finiteness):若依逻辑指令执行,则一定可在有限的步骤内终止,不会造成死机。
- (5) 有效性(Effectiveness):对于指令的执行,可用笔纸来模拟,确定指令是有效可行的。

1.2.1 算法的概念

算法(Algorithm)是在有限步骤内求解某一问题所使用的一组定义明确的规则。通俗点说,就是计算机解题的过程。在这个过程中,无论是形成解题思路还是编写程序,都是在实施某种算法。前者是推理实现的算法,后者是操作实现的算法。

一个算法应该具有以下五个重要的特征:

- 1. 有穷性:一个算法必须保证执行有限步之后结束;
- 2. 确切性:算法的每一步骤必须有确切的定义;
- 3. 输入:一个算法有零个或多个输入,以刻画运算对象的初始情况,所谓零个输入是指算法本身定了初始条件;
- 4. 输出:一个算法有一个或多个输出,以反映对输入数据加工后的结果,没有输出的算法是毫无意义的;
- 5. 可行性:算法原则上能够精确地运行,而且人们用笔和纸做有限次运算后即可完成。