

C语言数据结构程序设计

王开铸 俞经善 金虎 李秀坤 编著



哈尔滨工业大学出版社

C 语言数据结构程序设计

王开铸 俞经善 金虎 李秀坤 编著

哈尔滨工业大学出版社

·哈尔滨·

内容提要

本书是为高等学校计算机应用专业的学生编写的教材。全面介绍了程序设计的知识,数据结构的知识,本书的特点是用 C 语言把程序设计和数据结统一起来,做到了上下内容连贯,程序设计风格一致。既可将两门课合起来讲授,也可单独分开来讲授。本教材也适合非计算机专业的学生和一般科技人员阅读。

图书在版编目(CIP)数据

C 语言数据结构程序设计/王开铸等编著. —哈尔滨:哈尔滨工业大学出版社,2003.3

ISBN 7-5603-1820-7

I. C… II. 王… III. ①C 语言—程序设计—高等学校—教材
②数据结构—高等学校—教材 IV. ① TP312 ② TP311.12

中国版本图书馆 CIP 数据核字(2003)第 007663 号

出版发行 哈尔滨工业大学出版社

社 址 哈尔滨市南岗区教化街 21 号 邮编 150006

传 真 0451 - 6414749

印 刷 哈尔滨市工大节能印刷厂

开 本 787×1092 1/16 印张 25.75 字数 648 千字

版 次 2003 年 3 月第 1 版 2003 年 3 月第 1 次印刷

书 号 ISBN 7-5603-1820-7/TP·183

印 数 1~3 000

定 价 29.80 元

前 言

计算机高级语言是进行程序设计的先进工具。C 语言在 20 世纪 70 年代中期是一种很好的语言,不仅可以用它编写应用程序的程序,而且可以用它开发系统软件。著名的 UNIX 操作系统就是用 C 语言开发的。C 语言的编译系统也是用小 C 和汇编语言编写的。如今的软件工具先进了,用先进的软件工具开发应用程序,不仅缩短了软件的开发周期,而且还保证了应用程序的质量。如 Visual Basic 和 Visual C++ 这两种语言能让用户编写出极好的输入输出的窗口界面。但是,用户只能知道生成这个窗口的所以,确不能知道怎么生成这个窗口的所以然。用 C 语言进行程序设计,就能够使用户知道生成窗口的所以和所以然。也就是说,C 语言能使我们学到更多的程序设计的本领。

一般给大学本科生讲授 C 语言程序设计,用 40 个学时只能讲授简单的 C 语言知识,能编写出十几行的程序,对较为复杂的程序,不仅看不懂,而且也编写不出来。而对大专生讲授 C 语言,就是用了 60 学时,也不能达到这个要求。怎么解决这个问题呢,增加学时数?增加上机实习时间?进行强化训练?固然,这样做后,能取得一些效果,但也不尽人意。问题究竟出在哪里呢?

当计算机应用领域扩大后,遇到了非数值计算问题,也无法用数学方程加以描述,而必须分析要处理的对象的特性以及各处理对象之间存在的关系。数据结构是研究非数值计算问题中对象相互之间存在的一种或多种特定关系的数据元素的集合,或者说,数据结构是一门研究非数值计算的程序设计问题中计算机的操作对象以及它们之间的关系和操作的学科。

国内所有“数据结构”的书都认为它是一门学科,也都认为它还不完善。因此,都认为把它单独拉出来作为一门学科来研究、发展。在教学上作为单独的一门课进行讲授。这种做法无非是强调程序设计的发展进程,即从 60 年代初不认识“数据结构”的重要性,把程序设计作为一门艺术来看,到 60 年代末认识到“数据结构”的重要性,把程序设计作为一门技巧来看,到 70 年代中又认识到“数据结构”的重要性,不仅把程序设计作为一门学科来发展,而且,还把“数据结构”作为一门学科来发展。

我们把“数据结构”和“C 语言程序设计”捏到一起来讲,也许能弥补分开讲的缺点,即讲了“C 语言程序设计”不会用“数据结构”,而讲了“数据结构”又不会用“C 语言程序设计”。捏到一起来讲,也是遵循教学的本意:美国学者 D.E. Knuth 在他的学术巨著“计算机程序设计技巧”的第一卷就讲了数据结构;各个领域的应用程序的发展,才有了“数据结构”的今天;瑞士学者 N. Wirth 写的一本书书名就是“算法 + 数据结构 = 程序”;他进一步认为:“程序设计的方法学必然包含数据结构的一切方面”,“程序的构成与数据结构是两个不可分割地联系在一起的问题”。事实上,现今的高级语言设计中已包容了许多数据结构的内容。尽管两者都谈到了数据结构,但在内涵上是

有差别的。这就是把“数据结构”和“C 语言程序设计”捏到一起来讲的本意。

本书也可以作为“C 语言程序设计”教材来使用,或作为“数据结构”教材来使用。凡无星号的章节适用于“C 语言程序设计”教材,凡带两个星号的章节适用于“数据结构”教材。凡带单星号的章节,是这两本书的可选部分。

讲好一门课仅有教师讲和学生听还不够。传授知识得有本好的教材。如今,程序设计语言的教材体系,绝大多数是按照程序设计语言的知识体系的完整性来编写的,即按照语言本身的体系:从常量、变量、表达式、语句的语约定,到函数或过程的编写,类型是伴随着常量和变量来介绍的,实例也随着这些知识的介绍而引入。全书是以介绍语言的语句的语法和语义,且以语法为主,忽略了语用,忽略了语言的整体与部分之间的关系。因此,老师按照这类教材传授知识,多数学生感到学到一半时,还不知道有什么用。同理,“数据结构”教材理论部分讲的都不错,到上计算机实习时,都用一些类 C 语言或用类 pascal 语言编写程序,就很难通过。本书把两本教材合在一起,就解决了这一问题。本书由哈尔滨工业大学王开铸、李秀坤,哈尔滨工程大学俞经善,黑龙江大学金虎编著,由黑龙江东方学院常红对全书例题程序上机调试通过。

这本书是参阅了国内很多学者写的 C 语言和数据结构的书写成的。在这里对他们表示真诚的谢意。本书的全部手稿的内容,曾在黑龙江东方学院计算机系几届学生中进行讲授,加速了本书的最终形成。在此,对学生也表示谢意。最后,感谢哈尔滨工业大学出版社孙雅杰同志,得到她多次鼓励,才得以出版此书。

本书形成仓促,错误在所难免,敬请读者批评指正。

作 者

2003 年 1 月于哈尔滨工业大学

目 录

第一篇 初级篇

第1章 两种语言对比	(3)
1.1 程序结构分析	(3)
1.1.1 汉语文章结构分析	(3)
1.1.2 C语言程序结构分析	(3)
1.2 C语言成分分析	(4)
1.2.1 C程序成分分析	(4)
1.2.2 C语言的字	(5)
1.2.3 C语言的词	(6)
1.2.4 C语言的常数	(7)
1.2.5 C语言的表达式	(8)
1.3 C的语句	(8)
1.3.1 C的执行语句	(8)
1.3.2 C的说明语句	(10)
1.4 C语句的功能(语义、语用)	(11)
1.4.1 C的整型类型、整型运算符与整型表达式	(11)
1.4.2 C的实型类型、实型运算符与实型表达式	(12)
1.4.3 C的字符类型、字符运算符与字符表达式	(14)
1.4.4 C的执行语句	(15)
1.5 输入和输出函数	(22)
1.5.1 控制台输入输出一个字的函数	(23)
1.5.2 控制台输入输出一个字符串的函数	(24)
1.5.3 控制台格式化输入输出函数	(27)
小 结	(31)
习题一	(31)
第2章 C程序设计初步	(34)
2.1 程序设计的概念	(34)
2.1.1 程序与算法	(34)
2.1.2 程序设计	(36)
2.1.3 基本数据类型	(37)
2.2 算法表示与分析	(38)
2.2.1 算法的表示	(38)

2.2.2 算法分析	(41)
2.3 程序设计的方法	(43)
2.3.1 顺序程序设计	(43)
2.3.2 选择程序设计	(46)
2.3.3 循环程序设计	(49)
2.4 程序设计题解	(50)
2.4.1 科学计算题解	(51)
2.4.2 科学发现题解	(54)
小 结	(57)
习题二	(58)
第3章 函数初步	(61)
3.1 函数的概念和分类	(61)
3.1.1 函数引入	(61)
3.1.2 函数的概念	(63)
3.1.3 函数的分类	(63)
3.2 函数的定义与调用	(64)
3.2.1 函数的定义	(64)
3.2.2 函数的调用与返回	(65)
3.2.3 函数间的数据传递	(66)
3.3 变量的存储类型	(67)
3.3.1 什么是变量的存储类型	(67)
3.3.2 可控的存储类型	(69)
3.4 综合应用实例	(71)
3.4.1 问题分析	(71)
3.4.2 建立函数的认知模型算法	(72)
3.4.3 建立函数的行为模型算法	(74)
3.4.4 建立函数的程序模型的算法	(75)
3.4.5 调试程序	(79)
3.4.6 分析结果	(83)
小 结	(84)
习题三	(85)
第4章 阅读程序	(87)
4.1 阅读程序的目的	(87)
4.2 阅读程序的方法	(87)
4.3 阅读实例	(88)
4.3.1 简单程序分析	(88)
4.3.2 复杂程序分析	(94)

小 结	(100)
习题四	(100)

第二篇 高级篇

第5章 线性表与数组	(105)
5.1 数据结构	(105)
5.1.1 什么是数据结构	(105)
5.1.2 数据结构研究的内容	(106)
5.2 数据类型	(107)
5.3 线性表	(107)
5.3.1 线性表及其基本运算	(108)
5.3.2 线性表的存储结构	(108)
5.4 数组	(109)
5.4.1 数组的基本概念	(109)
5.4.2 一维数组定义与引用	(109)
5.4.3 一维数组的初始化	(110)
5.4.4 二维数组的定义和引用	(111)
5.4.5 二维数组的初始化	(112)
5.5 实例分析	(113)
5.5.1 数据结构分析	(113)
5.5.2 系统认知模型分析	(113)
5.6 堆栈与队列	(119)
5.6.1 堆栈及其操作	(119)
5.6.2 堆栈的存储结构及实现	(120)
5.6.3 栈的应用举例	(121)
5.6.4 队列及其应用	(126)
小 结	(129)
习题五	(130)
第6章 C语言的特色	(132)
6.1 C语言的运算符	(132)
6.1.1 运算符的种类	(132)
6.1.2 一目运算符	(132)
6.1.3 二目运算符	(134)
6.1.4 运算符的优先级	(137)
6.2 C语言的表达式	(138)

6.2.1 表达式的计算	(138)
6.2.2 逻辑类运算符与逻辑类表达式	(139)
6.3 再谈 for 循环语句	(141)
6.4 递归函数	(143)
6.4.1 递归调用	(143)
6.4.2 递归函数	(143)
6.5 数组作为函数参数	(148)
6.5.1 数组元素作为函数的实在参数	(148)
6.5.2 数组名可作函数参数	(150)
6.5.3 函数通用性	(151)
6.6 再论变量的存储类型	(153)
6.6.1 静态存储类型	(154)
6.6.2 外部存储类型	(155)
6.6.3 存储类别小结	(158)
6.7 内部函数和外部函数	(159)
6.7.1 内部函数	(159)
6.7.2 外部函数	(160)
6.8 如何运行一个多文件的程序	(160)
6.8.1 建立 Project 文件	(160)
6.8.2 用命令行编译、连接	(160)
小 结	(161)
习题六	(162)
第7章 串与串程序设计	(164)
7.1 串	(164)
7.1.1 串的概念	(164)
7.1.2 串抽象数据类型	(165)
7.2 串的机内表示	(166)
7.2.1 串的机内表示	(166)
7.2.2 串(字符数组)在 C 语言中表示	(166)
7.3 串操作的实现	(168)
7.3.1 字符串的输入和输出操作	(168)
7.3.2 串操作的算法	(168)
7.3.3 串操作的函数	(176)
7.4 综合应用实例	(177)
7.4.1 主函数 w-statistic 分析	(178)
7.4.2 函数 read-text 分析	(179)
7.4.3 函数 read-word 分析	(180)

7.4.4 函数 out-text 分析	(182)
7.4.5 函数 out-word 分析	(182)
小 结	(185)
习题七	(185)
第8章 指 针	(187)
8.1 地址和指针的基本概念	(187)
8.1.1 地址	(187)
8.1.2 指针	(188)
8.1.3 指针变量定义	(189)
8.1.4 指针变量定值	(190)
8.1.5 指针类型运算符和指针变量的运算	(190)
8.2 指针变量作为函数参数	(193)
8.3 建立一个指针	(195)
8.3.1 建立指向任意数据类型的指针变量	(196)
8.3.2 指针变量的运算	(197)
8.4 指针变量与数组	(200)
8.4.1 指针变量与一维数组	(200)
8.4.2 指针变量与二维数组	(201)
8.5 指针数组	(202)
8.5.1 指针数组概要	(202)
8.5.2 用指针数组处理高维数组数据	(202)
8.6 利用字符指针数组处理字符串	(203)
8.7 数组与函数参数	(204)
小 结	(206)
习题八	(206)
第9章 构造型数据类型——结构体与联合	(210)
9.1 记录数据结构	(210)
9.2 记录在 C 语言中的实现	(211)
9.2.1 结构体类型定义	(211)
9.2.2 结构体变量定义(命名)	(212)
9.2.3 结构体变量的定值与引用	(214)
9.3 结构体数组	(215)
9.3.1 结构体数组的定义	(215)
9.3.2 结构体数组定值	(217)
9.4 结构体变量与函数	(221)
9.4.1 结构体变量作为函数的参数	(221)
9.4.2 结构体类型的函数	(224)

9.5 指向结构体类型数据的指针变量	(229)
9.5.1 结构体指针变量的命名、定值、引用	(229)
9.5.2 指向结构体数组的指针变量	(232)
9.5.3 结构体指针数组的命名、定值、引用	(233)
9.6 结构体变量与函数参数的关系	(233)
9.7 特殊的数据类型——联合	(235)
9.7.1 联合体变量的定义、定值、引用	(235)
9.7.2 联合体变量的应用	(237)
9.8 枚举数据类型	(240)
9.8.1 枚举类型定义、定值、引用	(240)
9.8.2 枚举数据类型变量的应用	(242)
9.9 typedef 定义类型	(245)
小 结	(245)
习题九	(245)
第10章 文件	(249)
10.1 外存储设备	(249)
10.2 文件的基本概念	(249)
10.2.1 文件的种类	(250)
10.2.2 文件操作	(251)
10.3 C 语言文件操作的实现	(252)
10.3.1 C 的文件	(252)
10.3.2 文件缓冲区	(253)
10.3.3 数据文件的建立和使用	(253)
10.4 文件的输入输出	(255)
10.4.1 文件的字符输入输出函数	(256)
10.4.2 文件的字符串输入输出函数	(260)
10.4.3 格式化文件输入输出	(262)
10.4.4 二进制数据块输入输出函数	(264)
10.5 数据文件的随机输入输出	(267)
10.5.1 文件定位	(267)
10.5.2 数据文件的使用	(268)
小 结	(275)
习题十	(275)

第三篇 实用篇

第11章 链式结构与动态存储分配技术	(281)
11.1 链式结构与链表	(281)

11.2 链表的存储结构与操作	(283)
11.2.1 线性链表的操作	(283)
11.2.2 线性链表的插入操作	(283)
11.2.3 线性链表的删除操作	(285)
11.3 系统层—动态存储分配技术	(285)
11.3.1 操作系统层存储器管理	(286)
11.3.2 动态存储管理的分配方法	(286)
11.3.3 无用单元(垃圾)回收和存储紧缩	(288)
11.3.4 扩充内存管理	(289)
11.4 编译(语言)系统层存储管理	(290)
11.5 应用程序层存储管理	(292)
11.5.1 汇编语言的指针	(293)
11.5.2 Turbo C 存储模式	(294)
11.5.3 Turbo C 的 TC 环境	(294)
11.6 链表实例	(295)
小 结	(305)
习题十一	(305)
第12章 非线性数据结构	(307)
12.1 图的基本概念	(307)
12.2 树的基本概念	(309)
12.3 图与树的机内表示(存储结构)	(312)
12.3.1 图的机内表示(存储结构)	(312)
12.3.2 树的机内表示	(314)
12.4 二叉树与树、森林之间的转换	(318)
12.4.1 二叉树与树之间的转换	(318)
12.4.2 二叉树与森林之间的转换	(320)
12.5 遍历图与二叉树	(320)
12.5.1 图的深度优先搜索法	(321)
12.5.2 图的广度优先搜索法	(322)
12.5.3 遍历二叉树	(322)
12.5.4 几种常用的二叉树	(323)
12.6 图和树的实际应用	(329)
小 结	(341)
习题十二	(341)
第13章 文本窗口与菜单设计	(343)
13.1 显示器工作方式	(343)
13.2 有关文本窗口的输入输出函数	(345)
13.2.1 文本窗口及文本模式管理函数	(345)
13.2.2 光标设置、文本输出及管理函数	(346)

13.2.3 文本颜色的控制	(347)
13.2.4 文本窗口加边框的设计	(349)
13.2.5 文本窗口移动设计	(350)
13.3 bioskey 函数	(351)
13.4 图形显示方式	(355)
13.4.1 图形模式初始化	(355)
13.4.2 图形库函数	(356)
13.4.3 绘图函数	(356)
13.4.4 图形视口处理函数	(358)
13.4.5 图形模式下的文本输出函数	(359)
小 结	(361)
习题十三	(361)
第14章 查找与内部排序	(362)
14.1 线性表查找	(363)
14.1.1 顺序查找	(363)
14.1.2 折半查找	(364)
14.1.3 索引顺序查找(分块查找)	(366)
14.2 哈希查找	(367)
14.3 插入排序	(372)
14.3.1 直接插入排序	(372)
14.3.2 折半插入排序	(373)
14.3.3 希尔排序	(374)
14.4 选择排序	(375)
14.4.1 直接选择排序	(375)
14.4.2 堆排序	(377)
14.5 快速排序	(380)
14.6 归并排序	(382)
14.7 基数排序	(384)
小 结	(387)
习题十四	(387)
附录一 ASCII 码表	(388)
附录二 Turbo C 的库函数	(389)

第一篇 初级篇

初级篇主要介绍C语言的语法、语义和语用；并以语用为主来介绍C语言程序设计的四种方法；这里提出编写程序的认知模型和行为模型算法，介绍如何将行为模型算法翻译成C语言的程序的方法；最后，介绍阅读程序的方法。

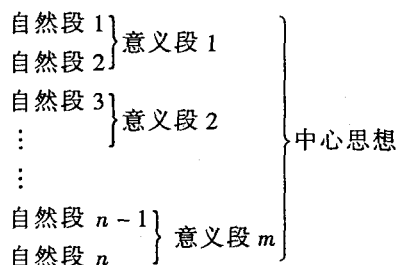
第 1 章 两种语言对比

自然语言(nature language)是人类用来交流思想和进行思维的工具。语言是一群人所认可的一组字、约定和规则的集合。世界上,现已查明有 5 761 种自然语言。自然语言具有两种表现形式:口语和书面语。由于自然语言的复杂性和庞大性,我们不能一下子把自然语言系统放在桌上或写在纸上。然而,人们面对这个复杂而庞大的自然语言系统,本族人学习本族语言,几乎是不学自通,不感到困难。本族人经过适当的学习和训练,也会用书面语来书写文章和表达思想,也不感到很难学。本章的目的就是要说明只有 24 个句子的 C 语言,是很好学的。

1.1 程序结构分析

1.1.1 汉语文章结构分析

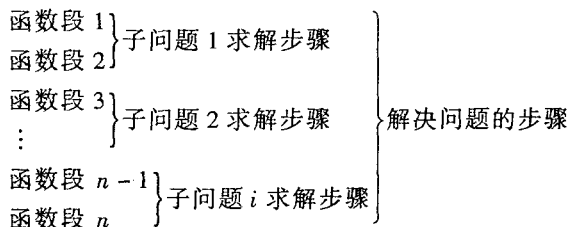
暂且回避汉语文章的内容,仅观察文章的外形。一篇文章的统一格式如下:



一篇文章由若干自然段组成,一个自然段由若干句子组成。在形式上,自然段是有明显标志的,它具有段头行、段体行和段尾行。段头行是指一行的开头缩进几格的行,段体行是指完整的若干行,而段尾行则是指其下一行是段头行或全文结束。有时一个自然段就是一行,它既是段头行又是段体行和段尾行。文章从内容上看,全文由若干个意义段组成,即每个意义段由一个或几个自然段组成。意义段的划分,通常是无严格的形式标志。意义段的内容,有时是自然段中的内容,有时又是言外之意。同样,一篇文章的中心思想,是从若干个意义段中提炼出来的,它表示这篇文章想要表达的中心思想。

1.1.2 C 语言程序结构分析

C 语言(C language)是一种人工语言。C 程序是程序员写出的 C 语言的书面文章,它是用来与计算机交流如何解决问题的方法和步骤的思想。计算机之所以懂得 C 语言,是因为软件人员还设计了一个 C 语言的编译程序(Compiler),装有 C 语言编译程序的计算机,才能懂得 C 语言的程序。同样,如果计算机装上了自然语言的编译程序,那么,计算机就能懂得自然语言了。遗憾的是,现在的软件人员还没有能力设计出自然语言的编译程序。C 程序的结构如下:



C 程序是告诉计算机如何解决问题的方法和步骤。程序结构类似于汉语文章的结构。C 程序由若干函数段(function)组成,函数段是组成 C 程序的基本构件。函数段类似于汉语文章的自然段。一个函数段是由若干个语句组成的,语句是组成函数段的基本构件。子问题求解步骤在形式上类似于汉语文章的意义段,它是由一个或几个函数段归纳出来的。整个问题的求解步骤是由若干子问题求解步骤组成的。函数段类似于自然段,它的外形具有明显的标志。函数段是由段头行、段体行和段尾行组成的。函数段的一般形式是:

```
[ <类型说明> ] <函数名> ( [ <参数表> ] )           /* 段头行 */
{
    [ <函数体语句组> ]                               /* 段体行 */
}                                                       /* 段尾行 */
```

段头行由三部分组成:第一部分是由方括号[]表示的可有可无的类型说明,第二部分是必有的函数名,第三部分是必有一对圆括号(),圆括号中是可有可无的参数表。注意,方括号和尖括号只是一种表示可选和必选的标志,本身在程序中并不出现(以后不再叙述)。

段体行由必有的一个开花括号{开始,后跟可选的函数体语句组组成。

段尾行用闭花括号}表示。但是,这个闭花括号}必须和段头行下面的开花括号{配对出现,缺一不可。

程序员想编程序解决问题,首先要将问题分解成若干子问题,对每一子问题再分解成多个具体步骤,其次用 C 语句描述每个具体步骤。也可以这样说,编写问题求解的程序,首先要把问题本身说清楚,即说清楚问题的初始状态和问题的终止状态,然后说清楚从初始状态到终止状态求解的路径(具体步骤)。或者说,C 程序就是告诉计算机“做什么”和“怎么做”的具体步骤。

1.2 C 语言成分分析

汉语文章的句子是无穷的,但句子的类型是有限的。按谓语句性质来划分可以分成主谓句,非主谓句,特殊句三类。每类句型又可分十几种。如特殊句就有:“是”字句,“有”字句,存在句,连动句,兼语句……等。汉语的词有十几类,细划分有几十类,具体的词就更多了,常用词就有八千多个。汉语的字是有限的,总共不过五万多个,常用字只有三四千个。C 语言与汉语相比真是小极了。句子不过 24 个,词不过 7 类,而字只有 89 个,应该说是很好学的。

1.2.1 C 程序成分分析

观察下面这个程序。

【例 1.1】 阅读下列程序

```
#define MAXLINE 1000                                /* maximum input line size */
#include <stdio.h>
```