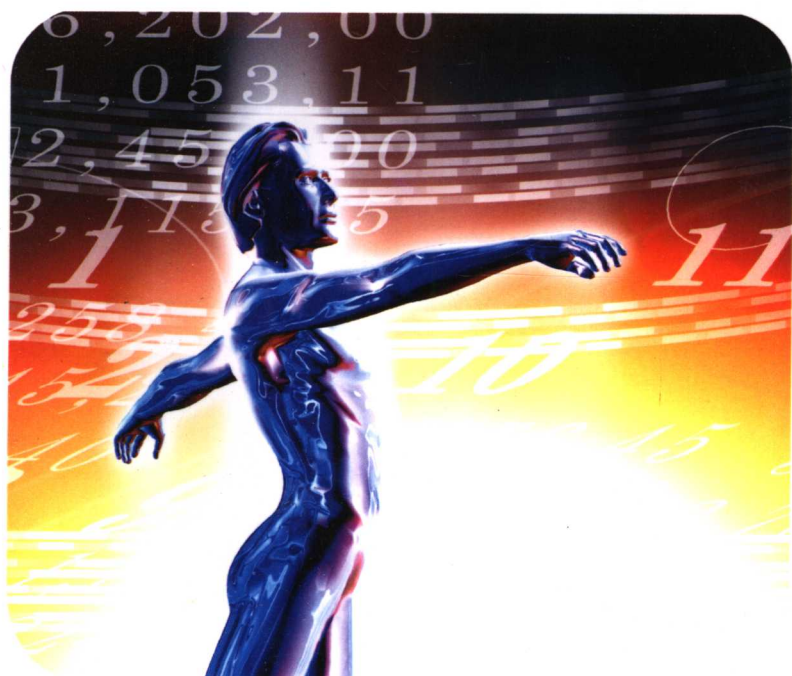


河南省高职高专计算机规划教材

C 语言程序设计

主 编 张 萍



西北大学出版社
NORTHWEST UNIVERSITY PRESS

C 语言程序设计

第二版



清华大学出版社

河南省高职高专计算机规划教材

C 语言程序设计

主 编 张 萍
副主编 刘其群 吕春峰 申志军
编 者 (以姓氏笔画为序)
申志军 刘玉昕 刘其群
吕春峰 张 萍 周国征
徐光联 秦晓梅 黄振中
韩哲欣 樊 继 王彩玲

西北大学出版社

图书在版编目(CIP)数据

C 语言程序设计 / 张萍主编. — 西安: 西北大学出版社, 2006. 2

高职高专计算机规划教材

ISBN 7-5604-2091-5

I. C... II. 张... III. C 语言 - 程序设计 - 高等学校: 技术学校 - 教材 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 140265 号

C 语言程序设计

主 编 张萍

出版发行 西北大学出版社

通讯地址 西安市太白北路 229 号 邮编 710069

经 销 新华书店经销

印 刷 河南新华印务有限公司

开 本 787mm × 1092mm 1/16

印 张 18

字 数 410 千字

版 次 2006 年 2 月第 1 版 2006 年 2 月第 1 次印刷

书 号 ISBN 7-5604-2091-5/TP · 41

定 价 27.00 元

出版说明

当前,我国正在走新型工业化道路,坚持以信息化带动工业化,以工业化促进信息化,加快发展现代服务业,全面建设小康社会。走新型工业化道路,既需要一大批拔尖创新人才,也需要数以千万计的专门人才和数以亿计的高素质劳动者。根据劳动力市场技能型人才短缺的状况,国家决定实施技能型紧缺人才培养培训工程,其中计算机人才的培养培训是其重要的组成部分。

为适应高职高专计算机教育发展的需要,促进教育教学改革和教材建设,满足经济和社会发展对计算机人才的需求,根据教育部等六部门印发的《关于实施职业院校制造业和现代服务业技能型紧缺人才培养培训工程的通知》精神,按照教育部《关于制定〈2004~2007年职业教育教材开发编写计划〉的通知》要求,在教育部有关部门的支持和指导下,经河南省教育厅批准,我们组织有关专家,对计算机高职高专教育的培养目标和模式、课程体系、教学内容、教学方法和手段、教学实践等方面,进行了广泛而深入的调研。

在充分调研的基础上,在教育部有关部门和河南省教育厅的大力支持下,我们组织有关专家召开了计算机高职高专教育教学研讨会、教学大纲审定会和主编人会议,确定了教材编写的指导思想、原则和要求,组织全省近40所院校的一线教师,吸取了最新的计算机高职高专教育教学经验和成果,编写了这套教材。本套教材充分体现了以就业为导向,以职业技能培养为根本的编写指导思想,突出了思想性、科学性、先进性、可读性和适用性的编写原则,较好地处理了“三基”(基本理论、基本知识、基本技能)关系,学历教育与职业认证、职业准入的关系。

这套教材虽经广泛调研与精心编撰,但一定还会存在这样或那样的不足,我们诚挚欢迎广大读者,尤其是选用该教材的教师和学生对其中的不足之处给予批评指正,以便我们在重印和修订时采纳有益的建议和意见,使之不断完善。

河南省高等学校计算机教育研究会

2006年1月

河南省高职高专计算机规划教材

参加编写学校

(以汉语拼音排序)

安阳师范学院

华北水利水电学院水利职业学院

黄河水利职业技术学院

河南财政税务高等专科学校

河南纺织高等专科学校

河南公安高等专科学校

河南工业大学

河南工业大学化学职业学院

河南经贸职业学院

河南农业职业学院

河南司法警官职业学院

河南商业高等专科学校

河南质量工程职业学院

济源职业技术学院

焦作大学

开封教育学院

开封人民警察学校

漯河职业技术学院

洛阳师范学院

平顶山工业职业技术学院

商丘师范学院

铁道警官高等专科学校

许昌职业技术学院

信阳农业高等专科学校

中原工学院广播影视职业学院

中州大学

郑州航空工业管理学院信息统计职业学院

郑州经济管理干部学院

郑州交通职业学院

郑州牧业工程高等专科学校

郑州轻工业学院

郑州轻工业学院轻工职业学院

郑州师范高等专科学校

郑州铁路职业技术学院

前 言

在各类程序设计语言中,C 语言应用得最为迅速和广泛。C 语言表达能力强、目标代码效率高、可移植性强,既具有高级语言的特点,又具有低级语言的特点,因此,它不仅适合编写系统软件,也适合编写应用软件。

C 语言作为计算机及其相关专业学生掌握程序设计这一必备技能之一的基础语言,无论是从实用性还是现实性考虑,都是十分必要和有意义的。

本书是作者根据多年来的教学经验和实际应用体会编写而成的,其主要特点如下:

1. 按照循序渐进、由浅入深的原则,所有章节内容都通过“实例引入→掌握基本知识→进行程序设计”这一讲述方式,以达到培养学生掌握基本程序设计方法和技能的目的。

2. 精选例题和习题,注重其典型性和实用性,既要达到对程序设计基本算法掌握的目的,又能起到学以致用效果。

3. 采用规范化文档,注重培养学生良好的程序设计风格。

本书由郑州经济管理干部学院的张萍担任主编,郑州铁路职业技术学院的吕春峰、刘其群、申志军为副主编。其中:第1章由刘其群编写,第2章由申志军编写,第3、4、5章由张萍、刘玉昕编写,第6、7、8章由吕春峰、韩哲欣编写,第9、10章由周国征、黄振中编写,第11章由秦晓梅编写,第12章由樊继编写,第13章由徐光联编写,第14章由王彩玲编写。全书由张萍组织并统稿,刘玉昕、吕春峰参加了全书的审校。在编写过程中,河南省高等学校计算机教育研究会和西北大学出版社给予了大力支持,在此表示诚挚的感谢。

限于编者的水平,本书难免有不妥之处,敬请各位同行和读者给予批评指正。

编 者

2005 年 7 月

目 录

第1章 程序设计与算法	(1)
1.1 程序设计的基本概念	(1)
1.1.1 程序	(1)
1.1.2 算法与数据结构	(1)
1.1.3 程序设计语言	(3)
1.2 程序设计方法	(4)
1.2.1 程序设计的一般步骤	(4)
1.2.2 结构化程序设计方法	(4)
1.3 程序设计风格	(6)
小结	(7)
习题	(8)
第2章 C语言程序设计初步	(9)
2.1 C语言简介	(9)
2.1.1 C语言的特点	(9)
2.1.2 C程序的结构	(9)
2.1.3 C语句概述	(12)
2.2 标识符、常量和变量	(13)
2.2.1 标识符	(13)
2.2.2 常量	(13)
2.2.3 变量	(14)
2.3 简单的数据类型	(14)
2.3.1 整型数据	(15)
2.3.2 实型数据	(17)
2.3.3 字符型数据	(18)
2.4 运算符及表达式	(21)
2.4.1 算术运算符和算术表达式	(21)
2.4.2 赋值运算符和赋值表达式	(22)
2.4.3 逗号运算符和逗号表达式	(23)
2.5 类型转换	(23)
2.5.1 自动类型转换	(23)
2.5.2 强制类型转换	(26)
小结	(27)
习题	(27)

第3章 顺序结构程序设计	(29)
3.1 顺序结构程序设计的思想	(29)
3.2 实现顺序结构程序设计的基本语句	(29)
3.2.1 赋值语句	(30)
3.2.2 标准输入/输出	(30)
3.3 顺序结构程序设计举例	(37)
小结	(38)
习题	(38)
第4章 分支结构程序设计	(40)
4.1 分支结构程序设计的思想	(40)
4.2 实现分支结构判断条件的构成	(41)
4.2.1 关系运算符与关系表达式	(41)
4.2.2 逻辑运算符与逻辑表达式	(41)
4.3 实现分支结构程序设计的语句	(43)
4.3.1 if 语句	(43)
4.3.2 switch 语句	(47)
4.4 分支结构程序设计举例	(49)
小结	(52)
习题	(53)
第5章 循环结构程序设计	(56)
5.1 循环结构程序设计的思想	(56)
5.2 实现循环结构的语句	(57)
5.2.1 while 语句	(57)
5.2.2 for 语句	(58)
5.2.3 do - while 语句	(60)
5.3 循环嵌套的概念及实现	(61)
5.4 循环结构程序设计举例	(65)
5.5 其他语句	(70)
5.5.1 break(间断语句)	(70)
5.5.2 continue(接续语句)	(71)
5.5.3 goto(转向语句)	(72)
小结	(73)
习题	(73)
第6章 函数	(78)
6.1 模块化程序设计的思想	(78)

6.1.1	什么是模块化程序设计	(78)
6.1.2	C 如何实现模块化程序设计	(78)
6.1.3	函数的分类	(79)
6.2	函数的定义	(80)
6.3	函数的调用	(82)
6.3.1	函数调用的一般形式	(82)
6.3.2	函数的参数和函数的值	(83)
6.3.3	对被调函数的声明和函数原型	(84)
6.4	函数的嵌套调用	(86)
6.5	函数的递归调用	(87)
6.6	变量的作用域	(90)
6.7	变量的存储类别	(95)
6.8	内部函数和外部函数	(100)
小结		(101)
习题		(102)
第7章	指针	(104)
7.1	概述	(104)
7.2	指针变量	(105)
7.2.1	指针的概念	(105)
7.2.2	指针变量的定义与赋值	(106)
7.2.3	指针变量的引用	(108)
7.3	指针和函数	(109)
7.3.1	指针变量作函数参数	(109)
7.3.2	返回指针值的函数	(112)
7.4	指向函数的指针变量	(113)
小结		(114)
习题		(114)
第8章	数组	(117)
8.1	一维数组	(117)
8.1.1	一维数组的定义	(117)
8.1.2	数组元素的访问	(118)
8.1.3	一维数组的初始化	(119)
8.1.4	一维数组程序举例	(120)
8.2	二维数组	(122)
8.2.1	二维数组的定义	(123)
8.2.2	二维数组元素的表示方法	(124)
8.2.3	二维数组的初始化	(125)

8.2.4	二维数组程序举例	(126)
8.3	字符数组	(128)
8.3.1	字符数组的定义	(128)
8.3.2	字符数组的初始化	(128)
8.3.3	字符数组的引用	(130)
8.3.4	字符串常用函数	(131)
8.3.5	字符数组应用举例	(134)
8.4	一维数组和指针	(136)
8.4.1	指向数组元素的指针	(136)
8.4.2	通过指针引用数组元素	(136)
8.4.3	指针变量的运算	(138)
8.5	二维数组和指针	(139)
8.5.1	二维数组的地址	(139)
8.5.2	通过指针引用二维数组元素	(140)
8.6	字符串和指针	(142)
8.6.1	指向字符串的指针	(142)
8.6.2	字符串指针作函数参数	(145)
8.7	向函数传递数组指针	(146)
8.8	指针数组及带参 main 函数	(149)
8.8.1	指针数组	(149)
8.8.2	指针数组作函数参数	(150)
小结		(153)
习题		(153)
第9章	结构体、共用体与枚举类型	(158)
9.1	结构体类型与结构体变量	(158)
9.1.1	结构体类型定义	(159)
9.1.2	结构体变量	(160)
9.1.3	结构体变量的引用	(161)
9.1.4	结构体变量的初始化	(162)
9.2	结构体数组	(163)
9.2.1	结构体数组的定义	(163)
9.2.2	结构体数组的初始化	(164)
9.2.3	应用举例	(165)
9.3	向函数传递结构体数据	(168)
9.3.1	结构体变量作函数参数	(168)
9.3.2	结构体指针作函数参数	(168)
9.3.3	返回结构体类型数据的函数	(169)
9.4	链表	(169)

9.4.1	链表概述	(169)
9.4.2	建立链表	(170)
9.4.3	对链表的插入操作	(171)
9.5	共用体	(172)
9.5.1	共用体定义	(172)
9.5.2	共用体变量	(173)
9.6	枚举型	(174)
9.6.1	枚举类型的定义	(174)
9.6.2	枚举类型应用举例	(174)
9.7	用户自定义类型	(175)
9.8	应用举例	(175)
小结		(178)
习题		(178)
第10章	编译预处理	(181)
10.1	宏定义	(181)
10.1.1	无参数宏定义	(181)
10.1.2	带参数宏定义	(183)
10.2	文件包含	(184)
10.2.1	文件包含的命令格式	(184)
10.2.2	文件包含命令的应用	(185)
10.3	条件编译	(185)
小结		(186)
习题		(187)
第11章	位运算	(189)
11.1	位运算和位运算符	(189)
11.1.1	“按位取反”运算符(~)	(189)
11.1.2	“左移”运算符 (<<)	(190)
11.1.3	“右移”运算符 (>>)	(190)
11.1.4	“按位与”运算符(&)	(191)
11.1.5	“按位异或”运算符(^)	(192)
11.1.6	“按位或”运算符()	(193)
11.1.7	位运算赋值运算符	(193)
11.2	位段	(194)
11.3	应用举例	(196)
小结		(198)
习题		(198)

第12章 文件	(199)
12.1 C 文件的基础知识	(199)
12.1.1 C 语言中的文件与流	(199)
12.1.2 文件缓冲区	(201)
12.2 文件类型指针	(201)
12.3 文件的打开与关闭	(202)
12.3.1 文件的打开(fopen 函数)	(202)
12.3.2 文件的关闭(fclose 函数)	(203)
12.4 文件的读写	(204)
12.4.1 文件的字符输入与输出函数(fgetc 与 fputc)	(204)
12.4.2 文件结束测试函数(feof)	(205)
12.4.3 文件的字符串输入与输出函数(fgets 与 fputs)	(206)
12.4.4 文件的格式输入与输出函数(fscanf 与 fprintf)	(207)
12.4.5 文件的数据块的输入与输出函数(fread 与 fwrite)	(208)
12.5 文件的定位和随机文件的读写	(209)
12.5.1 rewind 函数(又称“反绕”函数)	(209)
12.5.2 fseek 函数	(209)
12.5.3 ftell 函数	(211)
12.6 检测函数	(211)
12.6.1 ferror 函数	(211)
12.6.2 clearerr 函数	(212)
12.7 应用举例	(212)
小结	(213)
习题	(214)
第13章 综合程序设计案例	(215)
13.1 软件开发流程简介与举例	(215)
13.1.1 软件开发流程	(215)
13.1.2 软件开发各阶段任务	(216)
13.1.3 软件组成与编写进度	(217)
13.2 详细程序设计及模块编码技术	(218)
13.2.1 菜单技术	(218)
13.2.2 文件输入、输出技术	(220)
13.2.3 学生信息管理系统制作技术	(221)
13.3 程序组合	(226)
13.3.1 程序组合	(226)
13.3.2 组装测试与文档编写	(226)
小结	(228)

第14章 Turbo C 2.0 的使用及调试	(229)
14.1 Turbo C 2.0 的使用	(229)
14.1.1 Turbo C 2.0 集成开发环境简介	(229)
14.1.2 C 程序的编辑、运行	(230)
14.2 C 程序的简单调试	(232)
14.2.1 C 程序的简单调试	(232)
14.2.2 Turbo C 2.0 编译错误信息	(234)
附录 A ASCII 码表	(238)
附录 B C 语言中的关键字	(239)
附录 C 运算符和结合性	(240)
附录 D C 库函数	(242)
习题参考答案	(250)
参考文献	(274)

第 1 章 程序设计与算法

自 1973 年 Dennis M. Ritchie 设计并实现 C 语言以来,计算机程序设计领域发生了巨大的变化。以 C 语言为根基的 C++、VC++、Java 和 C#等面向对象编程的语言已经广泛应用于各种应用程序的开发。本章将介绍程序设计的基本知识、算法的初步知识、程序设计的方法和风格。

1.1 程序设计的基本概念

1.1.1 程序

自从 1946 年第一台计算机诞生以来,计算机已经应用于生活的各个领域,和人们的生活息息相关,成为生活和工作中不可缺少的工具。计算机是一种具有内部存储能力的自动、高效的电子设备,它最本质的使命就是执行指令所规定的操作。如果需要计算机完成什么工作,只要将其步骤用诸条指令的形式描述出来,并把这些指令存放在计算机的内部存储器中,需要结果时就向计算机发出一个简单的命令,计算机就会自动逐条顺序执行操作,全部指令执行完就得到了预期的结果。这种可以被连续执行的一条条指令的集合称为计算机的程序,即程序是计算机指令的序列。也可以这样说,程序就是人机交互的语言,是用于描述计算机解题方法和方式的代码,这些代码可以直接或间接地被计算机理解并执行。

1.1.2 算法与数据结构

一个程序应包括两部分内容:数据结构和算法。

数据结构指的是程序中要用到的数据和数据的组织形式;而算法指的是对数据结构的操作过程,是对解题方案准确而完整的描述。因此:

$$\text{程序} = \text{数据结构} + \text{算法}$$

算法和数据结构之间是密切相关的,算法是建立在数据结构基础上的;而明确了问题求解的算法,就能较好地设计出数据结构。数据结构是操作对象,算法是灵魂,算法是问题求解过程的精确描述。

例 1.1 求 $1+2+3+4+5$ 。

按传统的计算方法进行如下:

第1步:先求 $1+2$, 得到结果 3;

第2步:将第1步得到的和 3 再加上 3, 得到结果 6;

第3步:将第2步得到的结果再加上 4, 得到结果 10;

第4步:将 10 再加上 5, 得到 15。

上述算法虽然是正确的,但显得过于繁琐。如果求 $1+2+3+\cdots+100$ 的和,按这种方法,就需要 99 步,显然是不可取的。可采用通用的表示方法如下:

设两个变量,一个代表加数,一个代表被加数。将每一步的和放在被加数中。现设 s 为被加数, i 为加数,用循环算法求其结果如下:

step1:使 $s=1$

step2:使 $i=2$

step3:使 $s+i$, 结果放在变量 s 中,可表示为 $s=i+s$

step4:使 i 的值加 1,可表示为 $i=i+1$

step5:如果 i 不大于 5, 返回 step3, 重新执行 step3、step4 和 step5。否则算法结束,最后得到的 s 值就是 $1+2+3+4+5$ 算式的值。

如果求 $1+2+3+\cdots+100$ 和的话,只需将上述 step5 中的“如果 i 不大于 5”改为“如果 i 不大于 100”即可。由此可以看出,一个好的算法对解题具有重要意义。

从上例中可以看出一个算法应具有以下几个方面的特性:

1. 有穷性。

一个算法应在有限的步骤和有限的时间内完成。

2. 确定性。

算法中每一步的含义必须是确切的,不可出现任何二义性;对于相同的输入必须产生相同的输出。

3. 可行性。

算法中的每一步操作都应该能有效地执行,一个不可执行的操作是无效的。例如,一个数被 0 除的操作就是无效的,我们应当避免这种操作。

4. 有零个或多个输入。

这里的输入是指在算法开始之前所需要的初始数据,这些输入的多少取决于特定的问题,有些特殊算法也可以没有输入。

5. 有一个或多个输出。

所谓输出是指与输入有某种特定关系的量,因为算法的设计目的是为了求“解”,所以在—个完整的算法中至少应有一个输出。

算法设计是一件非常困难的工作,为了表示算法,可以采用不同的表示方法,常用的有自然语言、传统流程图、结构化流程图和伪代码等。流程图是描述算法常用的工具,传统的流程图由图 1.1 所示的几种基本框组成。

常用的算法设计技术主要有穷举法、迭代法、递推法、递归法、回溯法、分治法和动态规划法等。

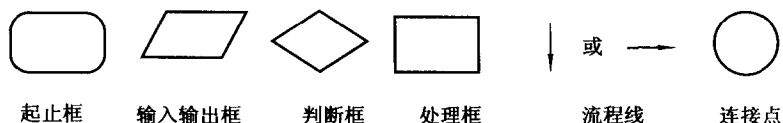


图 1.1 流程图的基本符号

1.1.3 程序设计语言

程序设计语言是指用于书写计算机程序的语言。

1. 分类

计算机语言一般分为低级语言和高级语言。

低级语言主要包括机器语言和汇编语言,是与计算机体系结构密切相关的程序设计语言。它依赖于具体的计算机,又称为“面向机器的语言”。

高级语言是用接近人们习惯的自然语言和数学语言作为语言的表示形式,其表示方法要比低级语言更接近于待解问题的表示方法,特点是在一定程度上与具体机器无关,易学、易用、易维护。但是用高级语言编写的程序,计算机是无法直接识别和执行的,要想被计算机识别和执行,必须通过相应的编译程序翻译成二进制的机器指令。

高级语言又可分为面向过程的语言和面向对象的语言。

面向过程的语言又称为结构化语言,用这类语言设计程序时,必须用语言的语句、函数、命令描述一步一步地解决问题的步骤。如:Pascal 语言、C 语言等。

面向对象的语言适合于解决大型的任务。用这类语言设计程序时,只需告诉计算机“做什么”,而不用说明“怎么做”。如:C++ 语言、Java 语言等。

2. 发展趋势

程序设计语言是软件的重要方面。它的发展趋势是模块化、简明性和形式化。

(1) 模块化。程序由模块组成,语言本身的结构也是模块化的。

(2) 简明性。涉及的基本概念不多,成分简单,结构清晰,易学易用。

(3) 形式化。

3. 部分程序设计语言的介绍

ALGOL 语言是早期的高级语言,是分程序结构语言,主要优点是可以有效地利用存储器。

FORTRAN 语言是第一个被广泛用于科学计算的高级语言。

COBOL 是一种面向商业的通用语言,在事务处理领域应用广泛。

PASCAL 语言是在 ALGOL60 的基础上发展起来的重要语言,其最大特点是简明性与结构化,语言本身具有很强的表达能力,而且其数据结构功能,在高级语言领域有很大的吸引力。

C 语言是 20 世纪 70 年代发展起来的,最初是为 UNIX 操作系统服务的,现在的 C 语言已成为世界上应用最广泛的计算机语言之一。

C++ 是面向对象的程序设计语言,它是在 C 语言基础上发展起来的,是对 C 语言的扩充,在 C++ 中最主要的是增加了类的功能。

Java 语言是一种通过解释方式来执行,面向对象的跨平台式的程序设计语言,语法规则