



21st CENTURY

实用规划教材

21世纪全国高职高专
计算机系列实用规划教材

C语言程序设计

主 编 刘迎春 王 磊
副主编 陈 静 陈庆惠 马玉凤
主 审 徐 峰

内容特点:

- 本书是2005年山东省精品课程，并有配套用书《C语言程序设计上机指导与同步训练》。
- 本书以职业技术能力培养为主线，采用问题驱动式教学，创设问题情景，将所有必讲内容分为32讲。每讲先提出问题，带着问题学习相关知识，然后解决问题，最后进行问题的深化。
- 本书提供的教学资源有：配套课件、Flash算法动画、源代码、C语言精品课程论坛和在线测试系统等。



北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国高职高专计算机系列实用规划教材

C 语言程序设计

主 编	刘迎春	王 磊	
副主编	陈 静	陈庆惠	马玉凤
参 编	胡晓芳	李宪玲	魏 慧
主 审	徐 峰		



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

该书是 2005 年山东省精品课程。

全书共分为 12 章, 内容包括: C 语言概述, 数据类型、运算符与表达式, 顺序结构程序设计, 选择结构程序设计, 循环结构程序设计, 数组, 函数, 指针, 结构体与共用体, 位运算, 文件和 C++ 面向对象理论基础。

全书采用问题驱动式教学, 创设问题情景, 将所有必讲内容分为 32 讲。每讲先提出问题, 带着问题学习相关知识点, 然后解决问题, 最后进行问题的深化。在教学内容的组织上, 基本语句、基本语法够用即可, 重视算法思想的讲解, 侧重培养学生的逻辑思维能力和编程解决实际问题的能力。

本书可作为高职高专计算机专业或非计算机专业开设的 C 语言程序设计课程教材, 也可作为学习 C 语言程序设计的参考书。

图书在版编目(CIP)数据

C 语言程序设计/刘迎春, 王磊主编. —北京: 北京大学出版社, 2006.8

(21 世纪全国高职高专计算机系列实用规划教材)

ISBN 7-301-10883-4

I. C… II. ①刘… ②王… 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(2006)第 077847 号

书 名: C 语言程序设计

著作责任者: 刘迎春 王 磊 主编

责任编辑: 李彦红

标准书号: ISBN 7-301-10883-4/TP · 0886

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://www.pup.cn> <http://www.pup6.com>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750667 出版部 62754962

电子信箱: pup_6@163.com

印 刷 者: 北京宏伟双华印刷有限公司

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 19.5 印张 442 千字

2006 年 8 月第 1 版 2006 年 8 月第 1 次印刷

定 价: 26.00 元

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究

举报电话: 010-62752024

电子邮箱: fd@pup.pku.edu.cn

21 世纪全国高职高专计算机系列实用规划教材

专家编审委员会

主 任 刘瑞挺

副主任 (按拼音顺序排名)

陈玉国 崔锁镇 高文志 韩希义

黄晓敏 魏 峥 谢一风 张文学

委 员 (按拼音顺序排名)

安志远 丁亚明 杜兆将 高爱国 高春玲 郭鲜凤

韩最蛟 郝金镇 黄贻彬 季昌武 姜 力 李晓桓

连卫民 刘德军 刘德仁 刘辉珞 栾昌海 罗 毅

慕东周 彭 勇 齐彦力 沈凤池 陶 洪 王春红

闻红军 武凤翔 武俊生 徐 红 徐洪祥 徐受容

许文宪 严仲兴 杨 武 易永红 于巧娥 袁体芳

张 昕 赵 敬 赵润林 周朋红 訾 波

信息技术的职业化教育

(代丛书序)

刘瑞挺/文

北京大学出版社第六事业部组编了一套《21 世纪全国高职高专计算机系列实用规划教材》。为此,制订了详细的编写目的、丛书特色、内容要求和风格规范。在内容上强调面向职业、项目驱动、注重实例、培养能力;在风格上力求文字精练、图表丰富、脉络清晰、版式明快。

一、组编过程

2004 年 10 月,第六事业部开始策划这套丛书,分派编辑深入各地职业院校,了解教学第一线的情况,物色经验丰富的作者。2005 年 1 月 15 日在济南召开了“北大出版社高职高专计算机规划教材研讨会”。来自 13 个省、41 所院校的 70 多位教师汇聚一堂,共同商讨未来高职高专计算机教材建设的思路和方法,并对规划教材进行了讨论与分工。2005 年 6 月 13 日在苏州又召开了“高职高专计算机教材大纲和初稿审定会”。编审委员会委员和 45 个选题的主、参编,共 52 位教师参加了会议。审稿会分为公共基础课、计算机软件技术专业、计算机网络技术专业、计算机应用技术专业 4 个小组对稿件逐一进行审核。力争编写出一套高质量的、符合职业教育特点的精品教材。

二、知识结构

职业生涯的成功与人们的知识结构有关。以著名侦探福尔摩斯为例,作家柯南道尔在“血字的研究”中,对其知识结构描述如下:

- ◆ 文学知识——无;
- ◆ 哲学知识——无;
- ◆ 政治学知识——浅薄;
- ◆ 植物学知识——不全面。对于药物制剂和鸦片却知之甚详。对毒剂有一般了解,而对于实用园艺却一无所知;
- ◆ 化学知识——精深;
- ◆ 地质学知识——偏于应用,但也有限。他一眼就能分辨出不同的土质。根据裤子上泥点的颜色和坚实程度就能说明是在伦敦什么地方溅上的;
- ◆ 解剖学知识——准确,却不系统;
- ◆ 惊险小说知识——很渊博。似乎对近一个世纪发生的一切恐怖事件都深知底细;
- ◆ 法律知识——熟悉英国法律,并能充分实用;
- ◆ 其他——提琴拉得很好,精于拳术、剑术。

事实上,我国唐朝名臣狄仁杰,大宋提刑官宋慈,都有类似的知识结构。审视我们自己,每人的知识结构都是按自己的职业而建构的。因此,我们必须面向职场需要来设计教材。

三、职业门类

我国的职业门类分为 18 个大类：农林牧渔、交通运输、生化与制药、地矿与测绘、材料与能源、土建水利、制造、电气信息、环保与安全、轻纺与食品、财经、医药卫生、旅游、公共事业、文化教育、艺术设计传媒、公安、法律。

每个职业大类又分为二级类，例如电气信息大类又分为 5 个二级类：计算机、电子信息、通信、智能控制、电气技术。因此，18 个大类共有 75 个二级类。

在二级类的下面，又有不同的专业：75 个二级类共有 590 种专业。俗话说：“三百六十行，行行出状元”，现代职业仍在不断涌现。

四、IT 能力领域

通常信息技术分为 11 个能力领域：规划的能力、分析与设计 IT 解决方案的能力、构建 IT 方案的能力、测试 IT 方案的能力、实施 IT 方案的能力、支持 IT 方案的能力、应用 IT 方案的能力、团队合作能力、文档编写能力、项目管理能力以及其他能力。

每个能力领域下面又包含若干个能力单元，11 个能力领域共有 328 个能力单元。例如，应用 IT 方案能力领域就包括 12 个能力单元。它们是操作计算机硬件的能力、操作计算机软件包的能力、维护设备与耗材的能力、使用计算机软件包设计机构文档的能力、集成商务计算机软件包的能力、操作文字处理软件的能力、操作电子表格应用软件的能力、操作数据库应用软件的能力、连接到互联网的能力、制作多媒体网页的能力、应用基本的计算机技术处理数据的能力、使用特定的企业系统以满足用户需求的能力。

显然，不同的职业对 IT 能力有不同的要求。

五、规划梦想

于是我们建立了一个职业门类与信息技术的平面图，以职业门类为横坐标、以信息技术为纵坐标。每个点都是一个函数，即 $IT(Professional)$ ，而不是 $IT+Professional$ 单纯的相加。针对不同的职业，编写它所需要的信息技术教材，这是我们永恒的主题。

这样组合起来，就会有 $IT((328)*(Pro(590)))$ ，这将是一个非常庞大的数字。组织这么多的特色教材，真的只能是一个梦想，而且过犹不及。能做到 $IT((11)*(Pro(75)))$ 也就很不容易了。

因此，我们既要在宏观上把握职业门类的大而全，也要在微观上选择信息技术的少而精。

六、精选内容

在计算机科学中，有一个统计规律，称为 90/10 局部性原理(Locality Rule)：即程序执行的 90% 代码，只用了 10% 的指令。这就是说，频繁使用的指令只有 10%，它们足以完成 90% 的日常任务。

事实上，我们经常使用的语言文字也只有总量的 10%，却可以完成 90% 的交流任务。同理，我们只要掌握了信息技术中 10% 频繁使用的内容，就能处理 90% 的职业化任务。

有人把它改为 80/20 局部性原理，似乎适应的范围更广些。这个规律为编写符合职业教育需要的精品教材指明了方向：坚持少而精，反对多而杂。

七、职业本领

以计算机为核心、贴近职场需要的信息技术已经成为大多数人就业的关键本领。职业教育的目标之一就是培养学生过硬的 IT 从业本领，而且这个本领必须上升到职业化的高度。

职场需要的信息技术不仅是会使用键盘、录入汉字，而且还要提高效率、改善质量、降低成本。例如，两位学生都会用 Office 软件，但他们的工作效率、完成质量、消耗成本可能有天壤之别。领导喜欢谁？这是不言而喻的。因此，除了道德品质、工作态度外，必须通过严格的行业规范和个人行为规范，进行职业化训练才能养成正确的职业习惯。

我们肩负着艰巨的历史使命。我国人口众多，劳动力供大于求的矛盾将长期存在。发展和改革职业教育，是我国全面建设小康社会进程中一项艰巨而光荣的任务，关系到千家万户人民群众的切身利益。职业教育和高技能人才在社会主义现代化建设中有特殊的作用。我们一定要兢兢业业、不辱使命，把这套高职高专教材编写好，为我国职业教育的发展贡献一份力量。

刘瑞挺教授 曾任中国计算机学会教育培训委员会副主任、教育部理科计算机科学教学指导委员会委员、全国计算机等级考试委员会委员。目前担任的社会职务有：全国高等院校计算机基础教育研究会副会长、全国计算机应用技术证书考试委员会副主任、北京市计算机教育培训中心副理事长。

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

刘瑞挺教授

(moo.dquq.www\k.qun) 刘瑞挺教授

本系列教材编写目的和教学服务

本系列教材在遍布全国的各位编写老师的共同辛勤努力下,在编委会主任刘瑞挺教授和其他编审委员会成员的指导下,在北京大学出版社第六事业部的各位编辑刻苦努力下,本系列教材终于与广大师生们见面了。

教材编写目的

近几年来,职业技术教育事业得以蓬勃的发展,全国各地的高等职业院校以及高等专科学校无论是从招生人数还是学校的软、硬件设施上都达到了相当规模。随着我国经济的高速发展,尽快提高职业技术教育的水平显得越来越重要。教育部提出:职业教育就是就业教育,也就是说教学要直接面对就业,强调实践。不但要介绍技术,更要介绍具体应用,注重技术与应用的结合。本套教材的主要编写思想如下。

1. 与发达国家相比,我国职业技术教育教材的发展比较缓慢并且滞后,远远跟不上职业技术教育发展的需求。我们常常提倡职业教育的实用性,但在课堂教学中仍然使用理论性和技术性教材进行职业实践教学。针对这种现状,急需推出一系列切合当前教育改革需要的高质量的优秀职业技术实训型教材。

2. 本套教材总结了目前优秀计算机职业教育专家的教学思想与经验,与广大职业教育一线老师共同探讨,最终落实到本套教材中,开发出一套适合于我国职业教育教学目标和教学要求的教材,它是一套能切实提高学生专业动手实践能力和职业技术素质的教材。

3. 社会对学生的职业能力的要求不断提高,从而催化出了许多新型的课程结构和教学模式。新型教学模式是必须以工作为基础的模仿学习,它是将学生置于一种逼真的模拟环境中,呈现给学生的是具有挑战性、真实性和复杂性的问题,使学生得到较真实的锻炼。

4. 教材的结构必须按照职业能力的要求创建并组织实施新的教学模式。教学以专项能力的培养展开,以综合能力的形成为目标。能力的培养既是教学目标,又是评估的依据和标准。

5. 本套的重点是先让学生实践,从实践中领悟、总结理论,然后再学习必要的理论,用理论指导实践。从这一个循环的教学过程中,学生的职业能力将得到极大的提高。

教学服务

1. 提供电子教案

本系列教材绝大多数都是教程与实训二合一,每一本书都有配套的电子教案,以降低任课老师的备课强度,此课件可以在我们网站上随时下载。

2. 提供教学资源下载

本系列教材中涉及到的实例(习题)的原始图片和其他素材或者是源代码、原始数据等文件,都可以在我们网站上下载。

3. 提供多媒体课件和教师培训

针对某些重点课程,我们配套有相应的多媒体课件。对大批量使用本套教材的学校,我们会免费提供多媒体课件,另外还将免费提供教师培训名额,组织使用本套教材的教师进行相应的培训。

前 言

C 语言程序设计是 2005 年山东省精品课程。作者结合多年的 C 语言程序设计教学及改革试验编写而成,以职业技术能力培养为主线,采用问题驱动式教学,创设问题情景,将所有必讲内容分为 32 讲。每讲先提出问题,带着问题学习相关知识点,然后解决问题,最后进行问题的深化。在教学实施中,一改以往 C 语言教学忙于讲解语法、代码的现象,以问题为向导来学习知识、分析问题、设计方法和解决问题,围绕问题组织课堂教学。本书可作为高职高专计算机专业或非计算机专业开设的 C 语言程序设计课程教材。

在教学内容的组织上,基本语句、语法够用即可,重视算法思想的讲解,培养学生的逻辑思维能力和编程解决实际问题的能力。

在 C 语言程序设计课程中,对带参宏、条件编译等很少使用的内容不作讲解,重点强化选择结构、循环结构、数组以及函数等编程中经常使用的知识点。

精选适合高职高专学生、与知识点结合密切和具有一定趣味性的 C 程序项目约 100 个,作为提出和深化的问题。每个问题的解决都配有算法思想的描述,以及必要的图示,并且开发了 Flash 算法动画,帮助学生充分理解,在算法基础上逐步写出源代码。

本书提供以下教学资源:

- (1) 配套课件。
- (2) Flash 算法动画。
- (3) 源代码。
- (4) 课程设计选题及设计。
- (5) 《C 语言程序设计上机指导与同步训练》(已经出版)。
- (6) C 语言精品课程论坛(<http://bluestar.yoeoo.com/php/clanguage>)。
- (7) 在线测试(<http://bluestar.yoeoo.com/TCClass/login.jsp>)。

本书由济宁职业技术学院的老师编写。其中第 1~3 章由王磊编写,第 4、10、11、12 章由刘迎春编写,第 6、7 章由陈静编写,第 5、9 章由陈庆惠编写,第 8 章由马玉凤编写,附录由李宪玲制作,本书的全部 Flash 算法动画由胡晓芳设计,魏慧做了部分校对工作。全书由刘迎春统稿、徐峰主审。

在编写本书过程中,虽然经过多位老师的反复审改并多次校对,但由于内容较多,再加上作者水平、时间、精力有限,难免有不妥和错误之处,敬请广大读者批评指正,不胜感激。衷心感谢所有关心本书编写的老师和朋友。

编者 E-mail: yashengli@sina.com 或 sdjnw@126.com。

编 者

2006 年 2 月

第 1 章 C 语言概述	1
1.1 C 语言出现的历史背景	1
1.2 C 语言的特点	2
1.3 简单的 C 程序介绍	3
1.3.1 三个简单的 C 程序实例	3
1.3.2 C 程序的基本组成	5
1.4 C 程序的上机步骤	7
1.5 算法	8
1.5.1 算法的概念	8
1.5.2 简单算法举例	9
1.5.3 算法的特性	11
1.5.4 算法的表示	12
1.5.5 结构化程序设计方法	17
1.6 本章小结	20
第 2 章 数据类型、运算符与表达式	21
2.1 C 语言的数据类型	21
2.2 常量与变量	23
2.2.1 常量与符号常量	23
2.2.2 变量	23
2.3 整型数据	25
2.3.1 整型常量的表示方法	25
2.3.2 整型变量	26
2.4 实型数据	29
2.4.1 实型常量的表示方法	29
2.4.2 实型变量	30
2.4.3 实型常量的类型	32
2.5 字符型数据	33
2.5.1 字符常量	33
2.5.2 字符变量	34
2.5.3 字符数据在内存中的存储形式及其使用	34
2.5.4 字符串常量	36
2.6 变量赋初值	37

2.7 各类数值型数据之间的混合运算	37
2.7.1 整型、实型、字符型数据之间可以混合运算	37
2.7.2 自动转换	37
2.8 算术运算符和算术表达式	39
2.8.1 C 运算符简介	39
2.8.2 算术运算符和算术表达式	40
2.9 赋值运算符和赋值表达式	45
2.9.1 赋值运算符与赋值表达式	45
2.9.2 复合的赋值运算符	47
2.10 逗号运算符和逗号表达式	48
2.11 本章小结	50
第 3 章 顺序结构程序设计	51
3.1 C 语句概述	51
3.1.1 C 程序结构	51
3.1.2 C 语句分类	52
3.2 赋值语句	53
3.3 输入/输出的概念及其 C 语言的实现	54
3.4 字符数据的输入/输出	55
3.4.1 putchar 函数——字符输出函数	55
3.4.2 getchar 函数——字符输入函数	55
3.5 格式输入与输出	55
3.5.1 printf 函数——格式输出函数	55
3.5.2 scanf() 函数——格式输入函数	59
3.6 顺序结构程序设计举例	63
3.7 预处理命令	66
3.7.1 宏定义	67
3.7.2 文件包含	69
3.8 本章小结	71

第 4 章 选择结构程序设计.....	73	6.2 二维数组的定义和引用.....	129
4.1 关系运算符和关系表达式.....	73	6.2.1 二维数组的定义.....	129
4.1.1 关系运算符及其优先级.....	73	6.2.2 二维数组的引用.....	130
4.1.2 关系表达式.....	74	6.2.3 二维数组的初始化.....	131
4.2 逻辑运算符和逻辑表达式.....	75	6.3 字符数组.....	135
4.2.1 逻辑运算符及其优先级.....	75	6.3.1 字符数组的定义.....	135
4.2.2 逻辑表达式.....	76	6.3.2 字符数组的初始化.....	135
4.3 单分支和双分支选择语句.....	77	6.3.3 字符数组的引用.....	136
4.3.1 单分支选择语句.....	77	6.3.4 字符串和字符串结束标志.....	136
4.3.2 双分支选择语句.....	78	6.3.5 字符数组的输入/输出.....	137
4.4 多分支选择语句.....	83	6.4 字符串处理函数.....	143
4.4.1 if...else...多分支选择语句.....	83	6.5 本章小结.....	155
4.4.2 switch 开关语句.....	84	第 7 章 函数.....	157
4.5 选择语句的嵌套与条件运算符.....	85	7.1 函数的定义、函数参数和函数值.....	157
4.5.1 选择语句的嵌套.....	86	7.1.1 C 语言对函数的规定.....	157
4.5.2 条件运算符.....	86	7.1.2 函数的定义.....	158
4.6 本章小结.....	91	7.2 函数的调用.....	160
第 5 章 循环结构程序设计.....	93	7.3 函数的嵌套调用.....	165
5.1 概述.....	93	7.4 函数的递归调用.....	166
5.1.1 基本概述.....	93	7.5 数组作为函数参数.....	171
5.1.2 goto 语句.....	94	7.5.1 数组元素作函数实参.....	171
5.2 while 语句.....	94	7.5.2 数组名作函数实参.....	172
5.3 do...while 语句.....	96	7.6 局部变量和全局变量.....	176
5.4 for 语句.....	101	7.6.1 局部变量.....	177
5.5 几种循环的比较.....	106	7.6.2 全局变量.....	178
5.5.1 循环结构的基本组成部分.....	106	7.7 动态存储变量和静态存储变量.....	180
5.5.2 几种循环的比较.....	106	7.7.1 变量的存储类别.....	180
5.6 循环的嵌套.....	108	7.7.2 局部变量的存储方式.....	180
5.7 break 语句和 continue 语句.....	112	7.7.3 全局变量的存储方式.....	182
5.7.1 break 语句.....	112	7.8 内部函数和外部函数.....	183
5.7.2 continue 语句.....	114	7.8.1 内部函数.....	183
5.8 本章小结.....	119	7.8.2 外部函数.....	183
第 6 章 数组.....	121	7.9 本章小结.....	186
6.1 一维数组的定义和引用.....	121	第 8 章 指针.....	189
6.1.1 一维数组的定义.....	121	8.1 地址的概念.....	189
6.1.2 一维数组元素的引用.....	122	8.2 变量与指针变量.....	190
6.1.3 一维数组的初始化.....	123	8.2.1 指针变量的定义.....	190

8.2.2 指针变量的基本操作.....	191	9.8.2 共用体变量的引用.....	244
8.2.3 指针变量的使用.....	194	9.8.3 共用体类型数据的特点.....	244
8.3 数组与指针变量.....	196	9.9 枚举类型.....	246
8.3.1 指向一维数组和数组元素的 指针变量.....	196	9.10 用 typedef 定义类型.....	248
8.3.2 指针与二维数组.....	199	9.11 本章小结.....	250
8.4 指针变量作函数参数.....	204	第 10 章 位运算	252
8.4.1 指针变量作函数参数.....	204	10.1 位运算符与位运算.....	252
8.4.2 用数组名作函数参数.....	207	10.1.1 按位与(&).....	253
8.5 字符串与指针.....	212	10.1.2 按位或().....	254
8.5.1 字符串的表示形式.....	212	10.1.3 按位异或(^).....	254
8.5.2 指向字符串的指针作函数 参数.....	215	10.1.4 按位取反(~).....	255
8.6 指针函数.....	219	10.1.5 左移位(<<).....	255
8.7 指针数组和指向指针的指针.....	220	10.1.6 右移位(>>).....	256
8.7.1 指针数组的概念.....	220	10.2 本章小结.....	260
8.7.2 指向指针的指针.....	221	第 11 章 文件	261
8.8 本章小结.....	225	11.1 C 语言文件概述.....	261
第 9 章 结构体与共用体	226	11.2 文件的打开与关闭.....	262
9.1 定义结构体类型.....	227	11.2.1 FILE 类型.....	262
9.2 定义结构体类型变量的方法.....	228	11.2.2 文件的打开.....	263
9.3 结构体变量的引用.....	229	11.2.3 文件的关闭.....	264
9.4 结构体变量的初始化.....	230	11.3 文件的读/写.....	265
9.5 结构体数组.....	230	11.3.1 fputc 函数和 fgetc 函数.....	265
9.6 指向结构体类型数据的指针.....	231	11.3.2 fread 函数和 fwrite 函数.....	268
9.6.1 结构体指针变量.....	231	11.4 其他的文件读/写函数.....	272
9.6.2 结构体变量作函数参数.....	232	11.5 文件的定位.....	272
9.7 用指针处理链表.....	236	11.5.1 feof 函数.....	273
9.7.1 链表概述.....	236	11.5.2 rewind 函数.....	273
9.7.2 简单链表的建立.....	236	11.5.3 fseek 函数.....	274
9.7.3 处理动态链表所需的函数.....	238	11.6 本章小结.....	274
9.7.4 建立动态链表.....	238	第 12 章 C++面向对象理论基础	276
9.7.5 输出链表.....	240	12.1 C++面向对象理论基本概念.....	276
9.7.6 链表的插入操作.....	241	12.1.1 面向过程与面向对象程序 设计语言的区别.....	276
9.7.7 链表的删除操作.....	242	12.1.2 类和对象的概念.....	277
9.7.8 链表的综合操作.....	243	12.2 类和对象的定义.....	277
9.8 共用体.....	243	12.2.1 类的定义.....	277
9.8.1 共用体的概念.....	243	12.2.2 对象的定义.....	279

12.3 对象的初始化.....	280	附录 I 常用字符与 ASCII 代码.....	287
12.4 继承性和派生类.....	282	附录 II 关键字及其用途.....	288
12.4.1 基类和派生类.....	282	附录 III 运算符的优先级和结合性.....	289
12.4.2 派生类的定义.....	283	附录 IV Turbo C 2.0 常用库函数.....	290
12.5 本章小结.....	285	参考文献.....	295
附录.....	287		
1.1 C 语言的发展.....	1		
1.2 C 语言的特点.....	1		
1.3 C 语言的应用.....	1		
1.4 C 语言的学习方法.....	1		
1.5 C 语言的发展.....	1		
1.6 C 语言的特点.....	1		
1.7 C 语言的应用.....	1		
1.8 C 语言的学习方法.....	1		
1.9 C 语言的发展.....	1		
1.10 C 语言的特点.....	1		
1.11 C 语言的应用.....	1		
1.12 C 语言的学习方法.....	1		
1.13 C 语言的发展.....	1		
1.14 C 语言的特点.....	1		
1.15 C 语言的应用.....	1		
1.16 C 语言的学习方法.....	1		
1.17 C 语言的发展.....	1		
1.18 C 语言的特点.....	1		
1.19 C 语言的应用.....	1		
1.20 C 语言的学习方法.....	1		
1.21 C 语言的发展.....	1		
1.22 C 语言的特点.....	1		
1.23 C 语言的应用.....	1		
1.24 C 语言的学习方法.....	1		
1.25 C 语言的发展.....	1		
1.26 C 语言的特点.....	1		
1.27 C 语言的应用.....	1		
1.28 C 语言的学习方法.....	1		
1.29 C 语言的发展.....	1		
1.30 C 语言的特点.....	1		
1.31 C 语言的应用.....	1		
1.32 C 语言的学习方法.....	1		
1.33 C 语言的发展.....	1		
1.34 C 语言的特点.....	1		
1.35 C 语言的应用.....	1		
1.36 C 语言的学习方法.....	1		
1.37 C 语言的发展.....	1		
1.38 C 语言的特点.....	1		
1.39 C 语言的应用.....	1		
1.40 C 语言的学习方法.....	1		
1.41 C 语言的发展.....	1		
1.42 C 语言的特点.....	1		
1.43 C 语言的应用.....	1		
1.44 C 语言的学习方法.....	1		
1.45 C 语言的发展.....	1		
1.46 C 语言的特点.....	1		
1.47 C 语言的应用.....	1		
1.48 C 语言的学习方法.....	1		
1.49 C 语言的发展.....	1		
1.50 C 语言的特点.....	1		
1.51 C 语言的应用.....	1		
1.52 C 语言的学习方法.....	1		
1.53 C 语言的发展.....	1		
1.54 C 语言的特点.....	1		
1.55 C 语言的应用.....	1		
1.56 C 语言的学习方法.....	1		
1.57 C 语言的发展.....	1		
1.58 C 语言的特点.....	1		
1.59 C 语言的应用.....	1		
1.60 C 语言的学习方法.....	1		
1.61 C 语言的发展.....	1		
1.62 C 语言的特点.....	1		
1.63 C 语言的应用.....	1		
1.64 C 语言的学习方法.....	1		
1.65 C 语言的发展.....	1		
1.66 C 语言的特点.....	1		
1.67 C 语言的应用.....	1		
1.68 C 语言的学习方法.....	1		
1.69 C 语言的发展.....	1		
1.70 C 语言的特点.....	1		
1.71 C 语言的应用.....	1		
1.72 C 语言的学习方法.....	1		
1.73 C 语言的发展.....	1		
1.74 C 语言的特点.....	1		
1.75 C 语言的应用.....	1		
1.76 C 语言的学习方法.....	1		
1.77 C 语言的发展.....	1		
1.78 C 语言的特点.....	1		
1.79 C 语言的应用.....	1		
1.80 C 语言的学习方法.....	1		
1.81 C 语言的发展.....	1		
1.82 C 语言的特点.....	1		
1.83 C 语言的应用.....	1		
1.84 C 语言的学习方法.....	1		
1.85 C 语言的发展.....	1		
1.86 C 语言的特点.....	1		
1.87 C 语言的应用.....	1		
1.88 C 语言的学习方法.....	1		
1.89 C 语言的发展.....	1		
1.90 C 语言的特点.....	1		
1.91 C 语言的应用.....	1		
1.92 C 语言的学习方法.....	1		
1.93 C 语言的发展.....	1		
1.94 C 语言的特点.....	1		
1.95 C 语言的应用.....	1		
1.96 C 语言的学习方法.....	1		
1.97 C 语言的发展.....	1		
1.98 C 语言的特点.....	1		
1.99 C 语言的应用.....	1		
2.00 C 语言的学习方法.....	1		

第 1 章 C 语言概述

C 语言是国际上广泛流行的计算机高级程序设计语言,它集高级语言和低级语言的功能于一体,既可用于系统软件的开发,也适合于应用软件的开发。同时它还具有效率高和可移植性强等特点,因此被称为当代最优秀的程序设计语言。本章详细介绍 C 语言的发展、特点、C 程序的结构、C 程序的上机执行过程以及程序的灵魂——算法。

本章内容

- (1) C 语言出现的历史背景。
- (2) C 语言的特点。
- (3) C 程序的结构。
- (4) C 程序的上机执行过程。
- (5) 算法的概念及特点。
- (6) 算法的表示方法。
- (7) 结构化程序设计方法。

相关知识点

1.1 C 语言出现的历史背景

1. C 语言的发展历史

对 C 语言的研究起源于系统程序设计的深入研究和发 展。1967 年,英国剑桥大学的 M. Richards 在 CPL(Combined Programming Language)语言的基础上,实现并推出了 BCPL(Basic Combined Programming Language)语言。

1970 年,美国贝尔实验室的 K.Thompson 以 BCPL 语言为基础,设计了一种类似于 BCPL 的语言,称为 B 语言。他用 B 语言在 PDP-7 机上实现了第一个实验性的 UNIX 操作系统。

1972 年,贝尔实验室的 Dennis M. Ritchie 为克服 B 语言的诸多不足,在 B 语言的基础上重新设计了一种语言,由于是 B 语言的后继,故称为 C 语言。1973 年,贝尔实验室的 K. Thompson 和 Dennis M. Ritchie 合作,首先用 C 语言重新改写了 UNIX 操作系统,在当时的 PDP-11 计算机上运行。此后,C 语言作为 UNIX 操作系统上标准的系统开发语言,伴随着 UNIX 操作系统的发展,越来越广泛地被人们接受和应用并被移植到其他计算机系统。

1978 年,Brian W. Kernighan 和 Dennis M. Ritchie(K&R)正式出版了著名的《The C

Programming Language》一书，此书中介绍的C语言成为后来广泛使用的C语言版本基础，它被称为标准C语言。

C语言的标准化工作是从20世纪80年代初期开始的。1983年，美国国家标准化协会(ANSI, American National Standard Institute)根据各种C语言版本对C语言扩充和发展，颁布了C语言的新标准ANSI C。ANSI C比标准C有了很大的扩充和发展。

由于C语言的不断发展，1987年，美国国家标准化协会在综合各种C语言版本的基础上，又颁布新标准，为了与标准ANSI C区别，称为87 ANSI C。1990年，国际标准化组织ISO接受了87 ANSI C作为ISO C的标准。这是目前功能最完善、性能最优良的C语言新版本。目前流行的C语言编译系统都是以它为基础的。本书讲述的内容基本上是以ANSI C为基础，并参考87 ANSI C。

2. C语言是当代最优秀的程序设计语言

早期的C语言主要是用于UNIX系统。由于C语言的强大功能和各方面的优点逐渐为人们认识，到了20世纪80年代，C语言开始进入其他操作系统，并很快在各类大、中、小和微型计算机上得到广泛的使用，成为当代最优秀的程序设计语言之一。

从C语言的发展历史可以看出，C语言是一种既具有一般高级语言特性(ALGOL 60带来的高级语言特性)，又具有低级语言特性(BCPL带来的接近硬件的低级语言特性)的程序设计语言。C语言从一开始就是用于编写大型、复杂系统软件的，当然C语言也可以用来编写一般的应用程序。也就是说：C语言是程序员的语言！

3. 目前我国PC系列兼容机上常用的C语言版本

(1) Borland公司：Turbo C(V2.0, V3.0)、Turbo C++、Borland C++和C++Builder(Windows版本)。

(2) Microsoft公司：Microsoft C(V5.0, V6.0, V7.0)和Visual C++(Windows版本)。

(3) Computer Innovasions公司：C 86(V2.3)。

(4) Lattice公司：Lattice C(V4.0)等。

1.2 C语言的特点

C语言之所以能被世界计算机界广泛接受，正是由于自身具备的突出特点。从语言体系和结构上讲，与PASCAL、ALGOL 60等语言相类似，是结构化程序设计语言。但从用户应用、实现难易程度、程序设计风格等角度来看，C语言的特点又是多方面的。

(1) 适应性强。能适应从8位微型机到巨型机的所有机种。

(2) 应用范围广。可用于系统软件到涉及各个领域的应用软件。

(3) 语言本身简洁，使用灵活，便于学习和应用。在源程序表示方法上，与其他语言相比，一般功能上等价的语句，C语言的书写形式更为直观、精练。

(4) 语言的表达能力强。C语言是面向结构化程序设计的语言，通用直观；运算符达30种，涉及的范围广，功能强。可直接处理字符，访问内存物理地址，进行位操作，可以直接对计算机硬件进行操作，反映了计算机的自身性能，足以取代汇编语言来编写各种系

统软件和应用软件。鉴于C语言兼有高级语言和汇编语言的特点,也可称其为“中级语言”。

(5) 数据结构系统化。C语言具有现代化语言的各种数据结构,且具有数据类型的构造能力,因此,便于实现各种复杂的数据结构的运算。

(6) 控制流程结构化。C语言提供了功能很强的各种控制语句(如 if、while、for、switch 等语句),并以函数作为主要结构成分,便于程序模块化,符合现代程序设计风格。

(7) 运行程序质量高,程序运行效率高。试验表明,C源程序生成的运行程序的效率仅比汇编程序的低 10%~20%,但C语言编程速度快,程序可读性好,易于调试、修改和移植,这些优点是汇编语言所无法比拟的。

(8) 可移植性好。统计资料表明,C编译程序 80%以上的代码是公共的,因此稍加修改就能移植到各种不同型号的计算机上。

(9) C语言存在的不足之处是:运算符和运算优先级过多,不便于记忆;语法定义不严格,编程自由度大,编译程序查错纠错能力所限,对不熟练的程序员带来一定困难;C语言的理论研究及标准化工作也有待推进和完善。为此,C语言对程序设计人员的素质要求相对要高。

1.3 简单的C程序介绍

1.3.1 三个简单的C程序实例

用C语言语句编写的程序称为C程序或C源程序。本节从两个简单的C程序中分析C程序的特性。

【例 1.1】 在窗体上显示“Welcome to the C world!”语句,并且显示一个由“*”号组成的大写的C字母。

实现此功能的C语言程序代码如下:

```
#include <stdio.h> /* 文件包含 */
main() /* 定义主函数 */
{
    printf("Welcome to the C world!\n"); /* 输出欢迎语句 */
    printf("*****\n"); /* 以下几行输出大写字母 C */
    printf("*\n");
    printf("*\n");
    printf("*****\n");
}
```

说明:

(1) main 表示“主函数”。每个C语言程序都必须有一个 main 函数,它是每一个C语言程序的执行起始点(入口点)。main()表示“主函数”main 的函数头。

(2) 用{ }括起来的是“主函数”main 的函数体。main 函数中的所有操作(语句)都在这一对{ }之间。即 main 函数的所有操作都在 main 函数体中。

(3) printf 是C语言的输出函数,功能是用于程序的输出(显示在屏幕上),双引号内的字符串原样输出。“\n”是换行符,即在输出完“Welcome to the C world!”或“*”后回车换行。

(4) 每条语句用“;”号结束。

(5) `/*.....*/`括起来的部分是一段注释, 注释只是为了改善程序的可读性, 在编译、运行时不起作用(事实上编译时会跳过注释, 目标代码中不会包含注释)。注释可以放在程序任何位置, 并允许占用多行, 只是需要注意“`/*`”、“`*/`”匹配。

思考: 根据此例仿写程序, 在屏幕上输出如下内容。

```
*****
How are you!
*****
```

【例 1.2】计算两数之乘积, 并输出结果。

实现此功能的 C 语言程序代码如下:

```
main() /* 计算两数之积 */
{
    int a,b,p; /* 这是定义变量 */
    a=12;b=25; /* 以下 3 行为 C 语句 */
    p=a*b;
    printf("p=%d\n",p);
}
```

说明:

(1) 同样此程序也必须包含一个 `main` 函数作为程序执行的起点。`{ }`之间为 `main` 函数的函数体, `main` 函数所有操作均在 `main` 函数体中。

(2) “`int a,b,p;`”是变量声明。声明了三个具有整数类型的变量 `a,b,p`。C 语言的变量必须先声明再使用。

(3) “`a=12;b=25;`”是两条赋值语句。将整数 12 赋给整型变量 `a`, 将整数 25 赋给整型变量 `b`。`a,b` 两个变量分别为 12,25。注意这是两条赋值语句, 每条语句均用“;”结束。

也可以将两条语句写成两行, 即:

```
a=12;
b=25;
```

由此可见 C 语言程序的书写可以相当随意, 但是为了保证容易阅读要遵循一定的规范。

(4) “`p=a*b;`”是将 `a,b` 两变量内容相乘, 然后将结果赋值给整型变量 `p`。

(5) “`printf("p=%d\n",p);`”是调用库函数 `printf()` 输出 `p` 的结果。“`%d`”为格式控制, 表示 `p` 的值以十进制整数形式输出。

程序运行后, 输出:

```
p=300
```

思考: 根据此例仿写程序, 计算小明本学期四门功课的平均分。

【例 1.3】输入两个整数, 找出二者中较大的数, 并输出。

实现此功能的 C 语言程序代码如下:

```
int max(int x,int y) /* 计算两数中较大的数 */
{ /* max 函数体开始 */
    int z; /* 声明部分, 定义变量 */
    if(x>y) z=x;
    else z=y;
```