

高职高专

现代信息技术系列教材

# C语言 程序设计

张强华 吕新平 编著

Information



Technology

人民邮电出版社  
www.pptph.com.cn

高职高专现代信息技术系列教材

# C 语言程序设计

张强华 吕新平 编著

人民邮电出版社

## 图书在版编目 (CIP) 数据

C 语言程序设计/张强华、吕新平编著. 北京: 人民邮电出版社, 2001.8  
高职高专现代信息技术系列教材

ISBN 7-115-09189-7

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 (2001) 第 048035 号

高职高专现代信息技术系列教材

### C 语言程序设计

- 
- ◆ 编 著 张强华 吕新平  
责任编辑 潘春燕
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@pptph.com.cn  
网址 <http://www.pptph.com.cn>  
读者热线:010-67129212 010-67129211(传真)  
北京汉魂图文设计有限公司制作  
北京朝阳隆昌印刷厂印刷  
新华书店总店北京发行所经销
  - ◆ 开本:787×1092 1/16  
印张:16.75  
字数:385 千字 2001 年 8 月第 1 版  
印数:1-6 000 册 2001 年 8 月北京第 1 次印刷

ISBN 7-115-09189-7/TP·2137

---

定价:22.00 元

本书如有印装质量问题,请与本社联系 电话:(010)67129223

## 内 容 提 要

本书主要讲述了 C 语言程序设计的基本理论、基础知识及使用方法。主要内容包括 C 语言的特点、上机编程方法、常量与变量、表达式、逻辑运算、数组、函数与指针、结构与联合、I/O 函数与文件及预处理等。

本书对基本知识讲述详细清晰,编排上有意把难点分开,避免读者在某一阶段过于吃力;示例丰富,通过例子来启发读者理解概念;在每章后面还附一些启发式思考题,以帮助读者掌握 C 语言的基本编程技巧。附录收入了库文件、Turbo C 编译信息英汉对照和库函数总汇(包括版本、语法格式、功能简述、返回值以及示例程序),并在最后建立了库函数的快速索引,可供读者在编程中长期查阅、参考。

为了方便高职高专的学生参加全国计算机等级考试,本书内容覆盖了 C 语言等级考试的大纲,并在书的每一章后面附有针对计算机等级考试的辅导,包括“出题思路”、“知识要点”及“典型试题举例”,课后的作业也有类似于等级考试的模拟题。

本书适合作为高职高专 C 语言课教材,也可作为计算机等级考试的辅导教材。

# 高职高专现代信息技术系列教材

## 编委会名单

主 编 高 林

副 主 编 张强华

委 员 （以姓氏笔画为序）

吕新平 林全新 郭力平 程时兴

## 丛书前言

江泽民总书记在十五大报告中提出了培养数以亿计高素质的劳动者和数以千万计专门人才的要求,指明了高等教育的发展方向。只有培养出大量高素质的劳动者,才能把我国的人数优势转化为人力优势,提高全民族的竞争力。因此,我国近年来十分重视高等职业教育,把高等职业教育作为高等教育的重要组成部分,并以法律形式加以约束与保证。高等职业教育由此进入了蓬勃发展时期,驶入了高速发展的快车道。

高等职业教育有其自身的特点。正如教育部“面向 21 世纪教育振兴行动计划”所指出的那样,“高等职业教育必须面向地区经济建设和社会发展,适应就业市场的实际需要,培养生产、管理、服务第一线需要的实用人才,真正办出特色。”因此,不能以本科压缩和变形的形式组织高等职业教育,必须按照高等职业教育的自身规律组织教学体系。为此,我们根据高等职业教育的特点及社会对教材的普遍需求,组织高等职业学校有丰富教学经验的老师,编写了这套《高职高专现代信息技术系列教材》。

本套教材充分考虑了高等职业教育的培养目标、教学现状和发展方向,在编写中突出了实用性。本套教材重点讲述目前在信息技术行业实践中不可缺少的、广泛使用的、从业人员必须掌握的实用技术。即便是必要的理论基础,也从实用的角度、结合具体实践加以讲述。大量具体操作步骤、许多实践应用技巧、接近实际的实训材料保证了本套教材的实用性。

在本套丛书编写大纲的制定过程中,广泛收集了高等职业学院的教学计划,调研了多个省市高等职业教育的实际,反复讨论和修改,使得编写大纲能最大限度地符合我国高等职业教育的要求,切合高等职业教育实际。

在选择作者时,我们特意挑选了在高等职业教育一线的优秀骨干教师。他们熟悉高等职业教育的教学实际,并有多年的教学经验;其中许多是“双师型”教师,既是教授、副教授,同时又是高级工程师、认证高级设计师;他们既有坚实的理论知识,很强的实践能力,又有较多的写作经验及较好的文字水平。

目前我国许多行业开始实行劳动准入制度和职业资格制度,为此,本套教材也兼顾了一些证书考试(如计算机等级考试),并提供了一些具有较强针对性的训练题目。

对于本套教材我们将提供教学支持(如提供电子教案),同时注意收集本套教材的使用情况,不断修改和完善。

本套教材是高等职业学院、高等技术学院、高等专科学院教材。适用于信息技术的相关专业,如计算机应用、计算机网络、信息管理、电子商务、计算机科学技术、会计电算化等。也可供优秀职高学校选作教材。对于那些要提高自己的应用技能或参加一些证书考试的读者,本套教材也不失为一套较好的参考书。

最后,恳请广大读者将本套教材的使用情况及各种意见、建议及时反馈给我们,以便我们在今后的工作中,不断改进和完善。

# 关于本书

---

---

## 1. 本书概览

C 语言是一种被广泛学习、普遍使用的计算机程序设计语言。它的高级语言形式、低级语言功能具有特殊的魅力。由于 C 语言具有完整的编程语言特点，因而被大多数高等院校作为典型的计算机教学语言。在计算机等级考试、全国计算机应用证书考试等多种计算机知识考试中都有 C 语言。此外，C 语言作为一门实用的、功能强大的程序设计语言，被程序设计人员广泛使用。因此，C 语言是一门十分重要的程序设计语言。

## 2. 学生如何使用本书

学生在学习 C 语言时，往往感到有一定的困难。在教学实践中发现，有些学生不去记忆 C 语言的基本知识，有些学生只是死记硬背并不理解所记忆的内容，这样都不会得到好的学习效果。因此，学生在学习中应首先理解基本概念，掌握基本理论。每章后的知识要点对于学生复习本章的知识十分有益。同时，要认真研究示例程序，掌握编写程序的思路。写程序与写文章有相似之处，应该注重学习范例甚至记忆典型范例（如关于链表的例子）。学习编程的最好方法就是读别人编写的程序，而后编写自己的程序，在编程中学习编程，在上机调试程序中得到巩固。概言之，学好 C 语言的重要途径是：理解基本概念、牢记基础知识、熟读典型示例、勤于上机实践。

## 3. 教师如何使用本书

本书是高职高专的教材，因此，在教学内容和难易程度上与本科教学有所区别。本书把 C 语言主要知识、核心内容讲解清楚，而对不常用、等级考试中又不要求的内容作了删减。对于要参加等级考试的学生，建议教师把每章后关于等级考试的辅导加以必要的讲解。同时配合作业中的内容，可有效提高等级考试成绩。在本书的后面还安排了实训内容，可作为上机练习题，这将有助于学生理解例题、提高学生编写程序的能力。本书作业的答案没有附在书后，需要答案和参考试卷的教师可通过因特网获取，地址是：

[zqh3882355@sina.com](mailto:zqh3882355@sina.com)

[zqh3882355@163.net](mailto:zqh3882355@163.net)

# 目 录

<b>第1章 C语言初步</b> .....	1
1.1 C语言概览 .....	1
1.1.1 C语言发展历程简述 .....	1
1.1.2 C语言的优点 .....	1
1.1.3 C语言的缺点 .....	2
1.1.4 C语言源程序的结构特征 .....	2
1.1.5 C语言源程序的书写要求 .....	4
1.1.6 最小的C函数 .....	5
1.1.7 函数的组成部分 .....	5
1.1.8 C语言的字符集 .....	5
1.1.9 C语言字符 .....	5
1.2 简述两个最常用的输入和输出函数 .....	7
等级考试指导 .....	8
练习题 .....	10
<b>第2章 常量</b> .....	12
2.1 整型常量 .....	12
2.1.1 八进制整常数 .....	12
2.1.2 十六进制整常数 .....	12
2.1.3 十进制整常数 .....	12
2.1.4 整型常数的后缀 .....	12
2.2 实型常量 .....	13
2.2.1 十进制数形式 .....	13
2.2.2 指数形式 .....	13
2.3 字符常量 .....	13
2.3.1 字符常量的构成 .....	13
2.3.2 转义字符 .....	13
2.3.3 字符串常量 .....	14
2.4 符号常量 .....	14
等级考试指导 .....	15
练习题 .....	17



<b>第 3 章 变量</b> .....	19
3.1 变量的基本知识 .....	19
3.1.1 变量的含义 .....	19
3.1.2 变量的命名 .....	19
3.1.3 变量的存储 .....	19
3.2 整型变量 .....	19
3.2.1 什么是整型变量 .....	19
3.2.2 整型变量的分类 .....	20
3.2.3 整型变量的说明 .....	20
3.3 实型变量 .....	21
3.4 字符变量 .....	22
等级考试指导 .....	24
练习题 .....	25
<b>第 4 章 基本运算符和表达式</b> .....	26
4.1 运算符的种类 .....	26
4.2 算术运算符 .....	27
4.3 自增自减运算符 .....	28
4.4 表达式 .....	30
4.4.1 强制转换与优先级 .....	30
4.4.2 算术表达式 .....	31
4.4.3 赋值运算符和表达式 .....	31
4.4.4 逗号运算符和逗号表达式 .....	32
4.5 逻辑运算 .....	33
4.5.1 关系运算 .....	33
4.5.2 关系表达式 .....	33
4.5.3 逻辑运算符 .....	34
4.5.4 逻辑表达式 .....	35
4.6 条件运算符和条件表达式 .....	35
等级考试指导 .....	36
练习题 .....	39
<b>第 5 章 输入输出语句</b> .....	42
5.1 printf 函数 .....	42
5.1.1 printf 函数的最简单形式 .....	42
5.1.2 转义字符的使用 .....	42
5.1.3 printf 函数调用的一般形式 .....	44
5.1.4 格式字符串 .....	45

5.2 scanf 函数 .....	48
5.2.1 scanf 函数的一般形式 .....	48
5.2.2 scanf 函数的格式说明符 .....	48
5.2.3 使用 scanf 函数的注意事项 .....	49
5.3 其他输入输出函数 .....	49
5.3.1 getch 与 getchar 函数 .....	49
5.3.2 gets 函数 .....	50
5.3.3 putchar 与 putch 函数 .....	50
5.3.4 puts 函数 .....	51
等级考试指导 .....	51
练习题 .....	53
<b>第 6 章 算法初步 .....</b>	<b>54</b>
6.1 算法的概念 .....	54
6.1.1 什么是算法 .....	54
6.1.2 算法的特性 .....	54
6.1.3 算法设计的要求 .....	55
6.2 算法的表示方法 .....	56
6.2.1 自然语言表示算法 .....	56
6.2.2 传统流程图表示法 .....	56
6.2.3 用 N-S 流程图表示算法 .....	57
6.2.4 用伪码表示算法 .....	58
6.3 结构化程序设计 .....	58
6.3.1 三种基本结构 .....	58
6.3.2 结构化程序设计 .....	59
6.3.3 结构化程序设计中的注意事项 .....	59
6.3.4 算法的合理性与优化 .....	61
等级考试指导 .....	63
练习题 .....	64
<b>第 7 章 程序设计基础 .....</b>	<b>65</b>
7.1 语句与流程 .....	65
7.1.1 表达式语句 .....	65
7.1.2 函数调用语句 .....	65
7.1.3 控制语句 .....	65
7.1.4 复合语句 .....	66
7.1.5 空语句 .....	66
7.1.6 程序的流程 .....	66
7.2 if 语句 .....	66

7.2.1 基本形式 .....	66
7.2.2 if-else 形式 .....	67
7.2.3 if-else-if 形式 .....	68
7.2.4 if 语句中应注意的问题 .....	69
7.2.5 if 语句的嵌套 .....	70
7.3 switch-case 语句 .....	72
7.4 循环结构程序 .....	74
7.4.1 while 语句 .....	74
7.4.2 do-while 语句 .....	75
7.4.3 for 语句 .....	77
7.4.4 break 语句 .....	79
7.4.5 continue 语句 .....	80
7.4.6 goto 语句 .....	80
等级考试指导 .....	81
练习题 .....	84
<b>第 8 章 数组</b> .....	<b>87</b>
8.1 数组基础 .....	87
8.1.1 数组类型说明 .....	87
8.1.2 数组元素的表示方法 .....	88
8.1.3 数组的赋值 .....	89
8.2 二维数组 .....	91
8.2.1 二维数组的定义 .....	91
8.2.2 二维数组元素的表示方法 .....	91
8.2.3 二维数组的初始化 .....	92
8.2.4 动态赋值 .....	93
8.3 字符数组 .....	93
8.3.1 字符数组的一般情况 .....	93
8.3.2 字符串 .....	93
8.3.3 字符串常用函数 .....	95
等级考试指导 .....	98
练习题 .....	100
<b>第 9 章 函数与变量</b> .....	<b>102</b>
9.1 函数定义 .....	102
9.1.1 根据来源定义 .....	102
9.1.2 根据返回情况定义 .....	102
9.1.3 根据参数传递定义 .....	103
9.2 函数定义与使用 .....	103

9.2.1 无参函数定义的一般形式	103
9.2.2 有参函数定义的一般形式	103
9.2.3 函数声明	105
9.2.4 函数调用	106
9.2.5 函数调用的类型	107
9.3 变量的作用域	109
9.3.1 局部变量	109
9.3.2 全局变量	110
9.4 变量的存储类型	112
9.4.1 概说	112
9.4.2 动态变量	113
9.4.3 静态变量	115
9.5 内部函数和外部函数	117
9.5.1 内部函数	117
9.5.2 外部函数	117
等级考试指导	118
练习题	120
<b>第 10 章 编译预处理</b>	<b>123</b>
10.1 宏	123
10.1.1 无参宏定义	123
10.1.2 带参数的宏	125
10.2 文件包含	127
10.2.1 文件包含的概念	127
10.2.2 文件包含的注意事项	128
10.3 条件编译	128
10.3.1 第一种形式	128
10.3.2 第二种形式	129
10.3.3 第三种形式	129
等级考试指导	130
练习题	131
<b>第 11 章 指针</b>	<b>133</b>
11.1 指针的基本概念	133
11.1.1 了解指针	133
11.1.2 指针变量的类型说明	133
11.1.3 指针变量的赋值	134
11.2 指针变量的运算	134
11.2.1 指针运算符	135

11.2.2 指针变量的运算 .....	135
11.3 数组指针变量的说明和使用 .....	137
11.3.1 数组的存储方式 .....	137
11.3.2 数组指针变量 .....	137
11.4 字符串指针变量的说明和使用 .....	138
11.5 函数指针变量 .....	139
11.5.1 函数指针变量的定义 .....	139
11.5.2 函数指针变量的调用 .....	140
11.5.3 函数指针变量的赋值 .....	141
11.5.4 函数指针变量的无参调用 .....	141
11.6 指针型函数 .....	141
11.6.1 指针型函数的定义 .....	141
11.6.2 指针型函数必须在主函数中说明 .....	141
11.6.3 局部变量不可返回 .....	141
11.6.4 一致性 .....	142
11.6.5 区别指针型函数与函数指针 .....	142
等级考试指导 .....	142
练习题 .....	144
<b>第 12 章 结构体、共用体和位运算 .....</b>	<b>147</b>
12.1 结构变量 .....	147
12.1.1 结构体的定义 .....	147
12.1.2 结构类型变量的说明 .....	148
12.1.3 结构变量成员的代表方法 .....	148
12.1.4 结构变量的赋值 .....	149
12.1.5 结构变量的初始化 .....	149
12.2 结构数组 .....	150
12.2.1 结构数组概述 .....	150
12.2.2 结构数组的初始化 .....	151
12.3 结构指针变量 .....	152
12.3.1 结构指针变量概述 .....	152
12.3.2 结构变量的使用 .....	152
12.4 联合类型的定义和联合变量的说明 .....	154
12.4.1 联合的定义 .....	154
12.4.2 联合变量的说明 .....	155
12.5 联合变量的赋值和使用 .....	156
12.5.1 赋值 .....	156
12.5.2 使用联合应注意的问题 .....	157
12.6 位运算符 .....	157

12.6.1 几个基本概念 .....	158
12.6.2 位运算符 .....	159
12.7 链表 .....	160
12.7.1 链表概述 .....	160
12.7.2 链表的建立 .....	161
12.7.3 链表的遍历 .....	163
12.7.4 删除结点 .....	164
12.7.5 插入结点 .....	164
等级考试指导 .....	166
练习题 .....	169
<b>第 13 章 文件与 I/O 函数 .....</b>	<b>172</b>
13.1 文件的基本概念 .....	172
13.1.1 磁盘文件 .....	172
13.1.2 设备文件 .....	172
13.1.3 标准设备文件 .....	172
13.1.4 两种文件 I/O 系统 .....	173
13.1.5 文件控制结构 .....	173
13.1.6 文件型指针 .....	173
13.2 文件的打开与关闭函数 .....	173
13.2.1 打开文件 .....	173
13.2.2 关闭文件 .....	175
13.3 文件的一般读写 .....	176
13.3.1 字符读写函数 .....	176
13.3.2 字符串读写函数 .....	179
13.3.3 数据块读写函数 .....	180
13.3.4 格式化读写函数 .....	181
13.4 文件定位与随机读写 .....	182
13.4.1 回拨指针函数 rewind() .....	182
13.4.2 移动指针函数 fseek() .....	182
13.4.3 取当前指针位置函数 .....	183
13.4.4 文件的随机读写 .....	183
13.5 文件检测函数 .....	184
13.5.1 文件结束检测函数 feof .....	184
13.5.2 读写文件出错检测函数 ferror .....	184
13.5.3 文件出错标志和文件结束标志置 0 函数 clearerr .....	185
等级考试指导 .....	185
练习题 .....	187

<b>第 14 章 C 语言实验指导</b> .....	189
14.1 实验概述 .....	189
14.1.1 实验的重要性 .....	189
14.1.2 实验的目标 .....	189
14.1.3 实验步骤 .....	190
14.2 C 语言程序的调试过程 .....	190
14.2.1 编辑源程序 .....	191
14.2.2 编译源程序 .....	191
14.2.3 连接目标程序 .....	192
14.2.4 运行可执行程序 .....	192
14.3 Turbo C 集成开发环境 .....	192
14.3.1 启动 TC .....	192
14.3.2 主菜单项的意义 .....	192
14.3.3 选择菜单项 .....	193
14.3.4 快速参考行 .....	193
14.3.5 TC 的编辑器 .....	194
14.3.6 调试源程序的步骤 .....	194
14.3.7 调试程序的高级技巧 .....	198
14.4 MS C 上机过程 .....	200
14.4.1 MS C 编译系统的组成 .....	200
14.4.2 MS C 上机步骤 .....	200
<b>实训</b> .....	201
实训 1 C 语言运行环境 .....	201
实训 2 常量的作用 .....	202
实训 3 变量的作用 .....	205
实训 4 运算符与表达式 .....	206
实训 5 程序设计方法 .....	206
实训 6 数组的使用及字符操作 .....	207
实训 7 函数及其使用 .....	208
实训 8 变量的作用范围及其存储 .....	208
实训 9 指针的使用 .....	209
实训 10 结构体和联合体的使用 .....	210
实训 11 文件操作 .....	211

附录 1 库文件 .....	212
附录 2 Turbo C 编译信息英汉对照 .....	214
附录 3 库函数总汇 .....	225



# 第 1 章 C 语言初步

---

---

## 1.1 C 语言概览

### 1.1.1 C 语言发展历程简述

C 语言是在 1978 年由美国电话电报公司(AT&T)贝尔实验室正式发表的。同时由 B.W.Kernighan 和 D.M.Ritchie 合著的《The C Programming Language》一书对 C 语言作了详细的描述。在此之后，美国国家标准学会(ANSI)制定了 C 语言标准。该标准于 1983 年发表，通常称之为 ANSI C。以后，在此基础上扩充形成了多种版本，目前常用的有以下几种：

- Microsoft C 或称 MS C；
- Borland Turbo C 或称 Turbo C；
- AT&T C。

这些不同的版本之间有一定差别。对初学者来说，不必过多地了解它们的差别。无论使用何种版本，重在理解 C 语言的特点与编程方法。

在 C 语言的基础上，1983 年又由贝尔实验室的 Bjarne Stroustrup 推出了 C++。C++ 进一步扩充和完善了 C 语言，成为一种面向对象的程序设计语言。

### 1.1.2 C 语言的优点

C 语言的优点主要可以概括为以下几个方面：

- 既有高级语言的程序思想与设计方法，又有低级语言的操作能力，所以它也被称为“中级语言”；
- 结构化的体系结构。层次清晰，便于按模块化方式组织程序，易于调试和维护；
- 非常强的处理能力。不仅具有丰富的运算符和数据类型，便于实现各类复杂的数据结构，还可以直接访问内存的物理地址。
- 广泛的可移植性。可以方便地移植到不同的软、硬件环境；
- 代码效率高。