



新世纪高职高专实用规划教材

● 计算机系列

C 语言程序设计

C YUYAN CHENGXU SHEJI

钟文峰 邱 力 万国平 主 编
管银枝 范宏斌 副主编



清华大学出版社

新世纪高职高专实用规划教材·计算机系列

C 语言程序设计

邱 力 万国平 主 编

钟文峰 管银枝 范宏斌 副主编

清华大学出版社

北 京

内 容 简 介

本书系统地介绍了C语言程序设计的基本理论与实用技术。全书共12章。主要内容包括：C语言的特点，基本数据类型，基本运算符和表达式，结构化程序设计方法；数组及使用；函数及函数之间的数据传递；编译预处理；指针的概念和使用；结构体、共用体和枚举类型及使用；位运算；文件操作；实验指导。每章都提供了大量的思考与练习题。为便于读者学习参考，附录中给出了每章习题的参考答案。

本书可作为高职高专相关专业的程序设计课程教材，也可作为电脑爱好者自学C语言程序设计的参考用书。

版权所有，翻印必究。举报电话：010-62782989 13901104297 13801310933
本书封面贴有清华大学出版社激光防伪标签，无标签者不得销售。

图书在版编目(CIP)数据

C语言程序设计/邱力，万国平主编；钟文峰，管银枝，范宏斌副主编.—北京：清华大学出版社，2004.8
(新世纪高职高专实用规划教材·计算机系列)

ISBN 7-302-09131-5

I. C… II. ①邱… ②万… ③钟… ④管… ⑤范… 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(2004)第075364号

出 版 者：清华大学出版社

<http://www.tup.com.cn>

社 总 机：010-62770175

地 址：北京清华大学学研大厦

邮 编：100084

客户服务：010-62776969

组稿编辑：刘天飞

文稿编辑：章忆文

封面设计：陈刘源

印 装 者：北京宏伟双华印刷有限公司

发 行 者：新华书店总店北京发行所

开 本：185×260 印张：19.5 字数：457千字

版 次：2004年8月第1版 2004年8月第1次印刷

书 号：ISBN 7-302-09131-5/TP·6439

印 数：1~5000

定 价：28.00元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题，请与清华大学出版社出版部联系调换。联系电话：(010)62770175-3103或(010)62795704

《新世纪高职高专实用规划教材》序

编写目的

目前,随着教育的不断深入,高等职业教育发展迅速,进入到一个新的历史阶段。学校规模之大,数量之众,专业设置之广,办学条件之好和招生人数之多,都大大超过了历史上任何一个时期。然而,作为高职院校核心建设项目之一的教材建设,却远远滞后于高等职业教育发展的步伐,以至于许多高职院校的学生缺乏适用的教材,这势必影响高职院校的教育质量,也不利于高职教育的进一步发展。

目前,高职教材建设面临着新的契机和挑战:

(1) 高等职业教育发展迅猛,相应教材在编写、出版等环节需要在保证质量之前提下加快步伐,跟上节奏。

(2) 新型人才的需求,对教材提出了更高要求,科学性、先进性和实用性要充分体现。

(3) 高职高专教育自身的特点是强调学生的实践能力和动手能力,教材的取材和内容设置必须满足不断发展的教学需求,突出理论和实践的紧密结合。

(4) 新教材应充分考虑一线教师的教学需要和教学安排。

有鉴于此,清华大学出版社在相关主管部门的大力支持下,组织近百所高等职业技术学院的优秀教师以及相关行业的工程师,推出了一系列切合当前教育改革需要的高质量的面向就业的职业技术实用型教材。

系列教材

本系列教材主要涵盖以下领域:

- 计算机基础及其应用
- 计算机网络
- 计算机图形图像处理与多媒体
- 电子商务
- 计算机编程
- 电子电工
- 机械
- 数控技术及模具设计
- 土木建筑
- 经济与管理
- 金融与保险

另外,系列教材还包括大学英语、大学语文、高等数学、大学物理、大学生心理健康等基础教材。所有教材都有相关的配套用书,如实训教材、辅导教材、习题集等。

教材特点

为了完善高等职业技术教育的教材体系,全面提高学生的动手能力、实践能力和职业技术素质,特意聘请有实践经验的高级工程师参与系列教材的编写,采用了一线工程技术人员与在校教师联合编写的模式,使课堂教学与实际操作紧密结合。本系列丛书的特点如下:

- (1) 打破以往教科书的编写套路,在兼顾基础知识的同时,强调实用性和可操作性。
- (2) 突出概念和应用,相关课程配有上机指导及习题,帮助读者对所学内容进行总结和提高。
- (3) 设计了“注意”、“提示”、“技巧”等带有醒目标记的特色段落,让读者更容易得到有益的提示与应用技巧。
- (4) 增加了全新的、实用的内容和知识点,并采取由浅入深、循序渐进、层次清楚、步骤详尽的写作方式,突出实践技能和动手能力。

读者定位

本系列教材针对职业教育,主要面向高职高专院校,同时也适用于同等学历的职业教育和继续教育。本丛书以三年高职为主,同时也适用于两年制高职。

本系列教材的编写和出版是高职教育办学体制和运作体制改革下的产物,在后期的推广使用过程中将紧紧跟随职业技术教育发展的步伐,不断吸取新型办学模式,课程改革的思路和方法,为促进职业培训和继续教育的社会需求奉献自己的一份力。

我们希望,通过本系列教材的编写和推广应用,不仅有利于提高职业技术教育的整体水平,而且有助于加快改进职业技术教育的办学模式、课程体系和教学培训方法,形成具有特色的职业技术教育的新体系。

教材编委会

《新世纪高职高专实用规划教材》

编委会

主 任	吴文虎			
副主任	边奠英			
委 员	(以姓氏笔画为序)			
	万国平	王洪发	王庆延	邓安远
	孙 辉	孙远光	朱华生	朱烈民
	李 萍	杨 龙	杨扶国	邱 力
	易镜荣	苑鸿骥	柏万里	胡剑锋
	黄 俭	黄学光	黄晓敏	曾 斌
	熊中侃	廖乔其	蔡泽光	魏 明

前 言

随着计算机技术突飞猛进的发展,计算机在人们工作、学习和社会生活的各个方面正在发挥着越来越重要的作用。社会对人才应该具有的计算机知识和应用能力的要求越来越高,计算机编程已不再是 IT 专业人员的专利。C 语言是目前世界上流行最广、使用最多的高级程序设计语言,国内外高等院校都把利用 C 语言进行程序设计,以培养学生的计算机软件开发能力作为培养目标之一。全国计算机等级考试、全国计算机应用技术证书考试(NIT)和全国各地组织的大学生计算机统一考试都将 C 语言列入了考试范围。

由于 C 语言涉及的概念多,语法规则复杂,运算符多样,数据类型丰富,使用灵活。对于初学者来说学习掌握 C 语言并不是一件容易的事。怎样使读者快捷地掌握 C 语言,并学以致用,这是我们常常考虑的问题。我们在深入研究教育部制定的《高职高专教育基础课程教学基本要求》和《高职高专教育专业人才培养目标》的基础上,根据多年高职高专教学的经验,参考了有关 C 程序设计的著作,组织编写了该教材。

本书是以 C 语言的新标准——ANSI C 为基础,以 Turbo C 为主线,全面系统地介绍了 C 语言。在教材编写的过程中,对教材的逻辑顺序作了精心安排以适应高职高专学生的需求,不仅强调 C 语言的基础知识,而且重视编程能力和调试能力的培养;对常用函数的使用进行说明。例题选取丰富、实用而又多样,有利于提高学生的学习兴趣。本书适合课堂教学和学生自学。

本书的第 1、2、7 章由钟文峰编写,第 3、4 章由管银枝编写,第 5、6、11 章由邱力编写,第 8、10 章由万国平编写,第 9 章由范宏斌和黄亚斌编写,第 12 章由邱力、范宏斌、万国平、钟文峰和管银枝编写。全书由邱力、万国平主编。

本书可作为高等专科学校、高等职业技术学院相关专业的程序设计课程的教材,也可作为全国计算机等级考试的参考书以及电脑爱好者自学参考用书。

由于作者水平有限,书中难免有疏漏之处,敬请广大读者予以指正。

编者

2004 年 5 月 8 日

目 录

第1章 C语言概述..... 1	
1.1 C语言的发展历史及其基本特性..... 1	
1.1.1 C语言的发展历史..... 1	
1.1.2 C语言的基本特性..... 1	
1.2 简单的C程序介绍..... 2	
1.3 C程序的上机步骤..... 4	
1.4 小结..... 6	
1.5 思考与练习..... 8	
第2章 基本数据类型、运算符和表达式..... 9	
2.1 C语言的基本数据类型..... 9	
2.1.1 常量和变量..... 9	
2.1.2 整型数据..... 11	
2.1.3 实型数据..... 14	
2.1.4 字符型数据和字符串常量..... 15	
2.2 变量赋初值..... 18	
2.3 各类数值型数据间的混合运算..... 20	
2.4 C语言的运算符和表达式..... 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..... 25	
2.4.2 逗号运算符和逗号表达式..... 28	
2.5 小结..... 29	
2.6 思考与练习..... 29	
第3章 基本输入输出和顺序结构程序设计..... 31	
3.1 数据输入输出的概念..... 31	
3.2 字符数据的输入输出..... 31	
3.2.1 putchar函数 (字符输出函数)..... 32	
3.2.2 getchar函数 (字符输入函数)..... 33	
3.3 格式输入与输出..... 34	
3.3.1 printf()函数 (格式输出函数)..... 34	
3.3.2 scanf函数(格式输入函数)..... 37	
3.4 常用函数的使用..... 40	
3.4.1 数学函数..... 40	
3.4.2 字符处理函数..... 41	
3.5 顺序结构程序设计举例..... 42	
3.6 屏幕定位及彩色输出..... 44	
3.6.1 文本窗口的定义..... 44	
3.6.2 文本窗口内的输入 输出函数..... 45	
3.6.3 文本窗口颜色的设置..... 46	
3.6.4 有关屏幕定位的操作..... 48	
3.6.5 基本图形函数..... 49	
3.7 小结..... 50	
3.8 思考与练习..... 51	
第4章 选择结构、循环结构程序设计..... 54	
4.1 C语句概述..... 54	
4.1.1 C语句概述..... 54	
4.1.2 结构化程序设计..... 55	
4.2 关系运算符和关系表达式..... 59	
4.2.1 关系运算符..... 59	
4.2.2 关系表达式..... 59	
4.2.3 关系表达式的值..... 60	
4.3 逻辑运算符和逻辑表达式..... 60	
4.3.1 逻辑运算符..... 60	
4.3.2 逻辑表达式..... 61	
4.3.3 逻辑表达式的值..... 61	
4.4 选择结构程序设计..... 62	
4.4.1 if语句..... 63	
4.4.2 条件运算符..... 67	
4.4.3 switch语句..... 68	

4.5 循环结构的程序设计.....	72	第 7 章 编译预处理.....	129
4.5.1 while 语句.....	72	7.1 宏定义.....	129
4.5.2 do-while 语句.....	74	7.1.1 不带参数的宏.....	129
4.5.3 for 语句.....	75	7.1.2 带参数的宏.....	131
4.5.4 循环的嵌套.....	78	7.2 文件包含处理.....	133
4.5.5 break 语句和 continue 语句.....	79	7.3 条件编译.....	135
4.6 程序设计.....	80	7.4 小结.....	137
4.7 小结.....	84	7.5 思考与练习.....	137
4.8 思考与练习.....	85	第 8 章 指针.....	139
第 5 章 数组.....	89	8.1 指针的概念.....	139
5.1 一维数组.....	89	8.1.1 变量的地址与 变量的内容.....	139
5.1.1 一维数组的定义和引用.....	89	8.1.2 直接访问与间接访问.....	140
5.1.2 一维数组的初始化.....	91	8.1.3 指针与指针变量.....	140
5.2 二维数组.....	93	8.2 指针变量的定义与引用.....	141
5.2.1 二维数组的定义与引用.....	93	8.2.1 指针变量的定义.....	141
5.2.2 二维数组初始化.....	94	8.2.2 指针变量的引用.....	142
5.2.3 二维数组程序举例.....	95	8.3 指针运算.....	144
5.3 字符串与数组.....	97	8.3.1 指针的算术运算.....	144
5.3.1 字符串.....	97	8.3.2 指针的关系运算.....	145
5.3.2 字符串的输入和输出.....	98	8.4 指针和数组.....	145
5.3.3 字符串处理函数.....	100	8.4.1 指针与一维数组.....	146
5.3.4 字符串数组.....	104	8.4.2 指针与二维数组.....	148
5.4 小结.....	105	8.5 指针与字符串.....	152
5.5 思考与练习.....	106	8.5.1 字符串的表示与引用.....	152
第 6 章 函数.....	109	8.6 指针数组和指向指针的指针.....	155
6.1 函数的定义和调用.....	110	8.6.1 指针数组.....	155
6.1.1 函数定义的一般形式.....	110	8.6.2 指向指针的指针.....	157
6.1.2 函数的调用.....	111	8.7 指针与内存的动态分配.....	159
6.1.3 函数的参数和函数的值.....	113	8.7 指针与内存的动态分配.....	160
6.2 函数的嵌套调用和递归调用.....	115	8.8 指针与数组作为函数的参数.....	162
6.2.1 函数的嵌套调用.....	116	8.8.1 指针变量作为函数的参数.....	163
6.2.2 递归调用.....	117	8.8.2 数组名作为函数的参数.....	164
6.3 变量的作用域和存储类别.....	119	8.9 带参数的 main 函数.....	167
6.3.1 局部变量和全局变量.....	119	8.10 返回指针值的函数.....	169
6.3.2 数据的存储类别.....	121	8.11 函数指针的定义与引用.....	172
6.3.3 内部函数与外部函数.....	123	8.11.1 函数指针的定义.....	172
6.4 小结.....	124	8.11.2 函数指针变量的赋值.....	172
6.5 思考与练习.....	125	8.11.3 函数指针变量的引用.....	172

8.11.4 函数指针变量作为 函数的参数	173	10.3 小结	215
8.12 本章小结	176	10.4 思考与练习	216
8.13 思考与练习	178	第 11 章 文件	219
第 9 章 结构体与共用体	182	11.1 文件概述	219
9.1 概述	182	11.2 文件类型指针	220
9.2 结构体类型与结构体 变量的定义	183	11.3 文件的打开和关闭	220
9.2.1 结构体类型定义	183	11.3.1 文件的打开	220
9.2.2 结构体变量的定义 和引用	184	11.3.2 文件的关闭	222
9.2.3 简化结构体类型名	187	11.4 文件读写	222
9.2.4 结构体数组	188	11.4.1 单个字符的读写	222
9.3 指向结构体类型数据的指针	190	11.4.2 字符串的读写	224
9.3.1 指向结构体变量的指针	190	11.4.3 数据块读写	225
9.3.2 指向结构体数组的指针	191	11.4.4 格式化数据读写	227
9.3.3 用结构体变量和指向 结构体数据的指针作 函数参数	192	11.5 文件定位函数 rewind, fseek, ftell	228
9.4 链表	194	11.6 小结	230
9.4.1 概述	194	11.7 思考与练习	231
9.4.2 创建一个新链表	196	第 12 章 上机实验	234
9.4.3 对链表的插入操作	197	实验一 C 程序的运行环境	234
9.4.4 对链表的删除操作	197	实验二 简单的 C 程序设计	238
9.5 共用体	198	实验三 顺序结构程序设计	240
9.5.1 共用体类型的定义	198	实验四 选择结构程序设计	242
9.5.2 共用体变量的定义和使用	199	实验五 循环结构程序设计	244
9.6 枚举型	201	实验六 数组	247
9.7 小结	202	实验七 函数	250
9.8 思考与练习	204	实验八 函数及编译预处理	252
第 10 章 位运算	207	实验九 指针	253
10.1 位运算符和位运算	207	实验十 位运算	256
10.1.1 按位与运算符(&)	208	实验十一 结构体和共用体	258
10.1.2 按位或运算符()	209	实验十二 文件	258
10.1.3 按位异或运算符(^)	209	附录一 常用字符与 ASCII 代码 对照表	260
10.1.4 按位取反运算符(~)	210	附录二 C 语言中的关键字及其用途	261
10.1.5 左移运算符(<<)	211	附录三 运算符的优先级和结合性	263
10.1.6 右移运算符(>>)	211	附录四 Turbo C 2.0 常用库函数	264
10.1.7 位运算赋值运算符	212	附录五 思考与练习答案	268
10.1.8 位运算程序举例	212	参考文献	293
10.2 位段	214		

第 1 章 C 语言概述

C 语言是近年来在国内外得到迅速推广应用的一种计算机语言,它之所以得到迅速推广,是与它的特点分不开的。本章学习的目的是让初学者在深入学习 C 语言之前,首先对 C 语言有个初步的了解,让他们知道学习 C 语言的重要性及学习方法。本章学习要点有:了解 C 语言程序的基本结构,掌握上机步骤;掌握 Turbo C 的启动方法以及 Turbo C 编译器的使用方法。

1.1 C 语言的发展历史及其基本特性

1.1.1 C 语言的发展历史

C 语言作为一种计算机高级语言,它既有一般高级语言的特性,又具有一定的低级语言特殊性,所以它既适合编写系统程序又适合编写应用程序,已在国际上广泛流行。

C 语言是 1972 年由美国的 Dennis Ritchie 设计开发,并首次在 UNIX 操作系统的 DECPDP-11 计算机上使用的。C 语言的前身是 ALGOL 语言(算法语言),1963 年英国剑桥大学在 ALGOL 语言的基础上增添了处理硬件的能力,并命名为"CPL(Combined Programming Language,复合程序设计语言)".CPL 由于规模大,学习和掌握困难,所以没有流行起来。1967 年剑桥大学的 Martin Richards 对 CPL 语言进行简化,推出了"BCPL(Basic Combined Programming Language,基本复合程序设计语言)"语言。1970 年美国贝尔实验室的 K.Thompson 对 BCPL 进行了进一步简化,突出了硬件处理能力,并取了"BCPL"的第一个字母"B"作为新语言的名称。1972 年贝尔实验室的 Brian W.Kernighan 和 Dennis M.Richie 对 B 语言进行了完善和扩充,并取了"BCPL"的第二个字母"C"作为新语言的名称,此时 C 语言就问世了。

随着微型计算机的日益普及,出现了许多 C 语言版本,为了统一标准,美国国家标准化协会(ANSI)于 1987 年制定了 C 语言的标准,称为"ANSI C",成为现行的 C 语言标准。

1.1.2 C 语言的基本特性

C 语言发展如此迅速,而且成为最受欢迎的语言之一,主要是由于它具有强大的功能。现在它已经成为在微型、小型、大型和巨型计算机上,从系统程序设计到工程应用程序设计都能使用的一种高级程序设计语言。归纳起来,C 语言具有以下特点:

- (1) C 语言是高、低级兼容语言。比其他高级语言更接近硬件,比低级语言更接近算法,程序易编、易读、易查错、易修改。

- (2) C 语言是一种结构化程序设计语言。它提供了完整的程序控制语句,可使程序层次清晰,便于使用、维护及调试。C 语言是以函数形式提供给用户的,这些函数可方便地调用,并具有多种循环、选择语句来控制程序的流向,从而使程序完全结构化。
- (3) C 语言的数据类型和运算符十分丰富。程序设计和算法描述十分简单和方便。
- (4) C 语言功能齐全。因为引入了指针,所以程序运行效率更高。另外,C 语言也具有强大的图形功能,支持多种显示硬件和驱动程序,而且计算机的逻辑判断能力也比较强大。
- (5) C 语言是一种模块化程序设计语言。适合大型软件的研制和调试。
- (6) C 语言程序的可移植性好。C 语言本身不依赖于机器硬件系统,从而便于在硬件结构不同的机器间和各种操作系统间实现程序的移植。

C 语言的优点很多,但也有些不足之处。C 语言语法限制不太严格,程序设计时自由度大。C 语言对变量的类型使用比较灵活,例如,整型和字符型数据可以通用。C 语言允许程序设计者有较大的自由度,放宽了对语法的检查。因此,程序员应认真仔细地检查程序,确保其正确性,而不要过分依赖于 C 语言编译程序去查错。不过,程序员使用 C 语言编写程序时会感到限制少、灵活性大、功能强,可以编写出任何类型的程序。

关于以上特点,作为初学者可以先简单了解一下,当学完本书内容时,一定会有比较深刻的体会。

1.2 简单的 C 程序介绍

C 程序由 C 语言的语句序列组成,通常一个 C 程序包括一个或多个函数,其中必定有一个函数称为主函数,其函数名为 `main`。下面来看几个简单的 C 程序例子,以便对 C 语言有一个初步的认识,进而了解编写 C 程序的一些基本问题。

C 语言像其他语言一样,按其规定的格式和提供的语句,由用户编写程序。请看下面几段 C 源程序。

【例 1.1】输出一行信息的 C 程序。

```
main()          /*定义主函数*/
{
    printf("How are you?\n");
}
```

该程序只由一个主函数构成,程序的第一行 `main` 为主函数名,主函数名后的一对圆括弧 `()` 是函数的组成部分。C 程序中的所有函数的函数名后都必须带一对圆括弧。程序的第二至第四行为函数体部分,函数体要用一对大括号 `{}` 括住。程序的第三行为一个函数调用语句,用 `printf` 函数来输出数据,该函数的使用将在第 3 章中详细介绍。程序中的分号是 C 语句的结束标志,规定 C 语句都必须以分号结尾。

编译并运行该程序得出结果:

```
How are you?
```

【例 1.2】求两个整数之和，两个整数由键盘输入。

```
main()
{
    int x,y,z;
    scanf("%d,%d",&x,&y); /*输入两值，分别放入变量x、y中*/
    z=x+y;
    printf("sum=%d\n",z);
}
```

这个 C 程序只由一个 main 函数组成。程序的第三行为定义变量部分，在这里定义了三个整型变量 x、y 和 z，其中的 int 为数据类型标识符，表示整型；x、y、z 为三个变量的名称，定义变量时变量名之间用逗号隔开。程序的第四行为函数调用语句，通过 scanf 函数给变量 x 和 y 输入数据，该函数的具体使用将在第 3 章中详细介绍。程序的第五行为赋值语句，将变量 x 与 y 两数之和赋予变量 z，即用变量 z 存放两变量 x、y 的和。程序的第六行用于输出变量 z 的值。

该程序的运行结果如下：

```
3,5↵
sum=8
```

(注：本书中所有的"↵"均表示回车符，不再另行说明)

【例 1.3】求三个整数中的最小值，三个整数由键盘输入。

```
/*min函数用于求x1, x2, x3三个数中的最小值*/
int min(int x1,int x2,int x3) /*函数说明部分，int为数据类型标识符，min*/
/*为函数名，x1, x2, x3为三个形参*/
{
    int x; /*定义整型变量x*/
    x=x1; /*先把x1作为最小值并存放于x中*/
    if(x>x2) x=x2; /*如果x2小于x，则把x2放于x中*/
    if(x>x3) x=x3; /*如果x3小于x，则把x3放于x中*/
    return(x); /*将所求的最小值x返回到主调函数中*/
}

/*main函数完成三个整数的输入，并输出这三个数中的最小值*/
main()
{
    int x,y,z,k;
    scanf("%d,%d,%d",&x,&y,&z);
    k=min(x,y,z); /*调用min函数求x, y, z三个数中的最小值*/
    printf("max=%d\n",k);
}
```

该 C 程序由 min，main 这两个函数构成，它们各有一定的功能。

该程序运行结果如下：

```
15,3,57↵
min=3
```

从上述三个 C 语言程序例子，可以看出：

(1) C 程序是由函数构成的。

每个 C 源程序必须有一个而且只能有一个称作主函数的 main 函数，除主函数外，可以没有其他函数(如例 1.1，例 1.2)，也可以有一个或多个其他函数(如例 1.3)。

因此函数是构成 C 程序的基本单位。

- (2) 每个函数(包括主函数)的定义分为两部分: 函数说明部分和函数体。

函数说明部分的格式如下:

返回值的类型 函数名(数据类型标识符 形式参数1, 数据类型标识符 形式参数2, ...)

函数体的格式如下:

```
{变量定义部分  
 实现函数功能的语句串  
}
```

当然, 函数体中也可以没有任何内容, 只有一对 {}, 这在一些特殊场合可能会遇到。

- (3) C 源程序中的所有语句都必须用一个分号作为终止符, 否则 C 语言认为该语句没有结束。但在预处理命令、函数体中右花括号之后不能加分号。
- (4) C 语言中的字母要区分大小写, 习惯上用小写字母。只有常量、宏定义等多用大写, 所有的关键字必须小写, 如 `int`、`if`、`return` 等只能小写。同样, 使用变量时也必须注意变量名的大小写, 相同字母的大、小写代表不同的变量, 如 `a` 与 `A` 表示两个不同的变量。
- (5) 可以用 `/*.....*/` 对 C 程序中的任何部分作注释, 一个好的有使用价值的源程序都应当加上必要的注释, 以增强程序的可读性。注释通常放在一段程序的开始, 用以说明该段程序的功能; 或者放在某个语句的后面, 对该语句进行说明。源程序编译时, 注释部分将不参加编译, 也不会出现在目标程序中。
- (6) C 程序的书写格式很灵活, 一个语句可以写在多行上, 在一行上也可以书写多个语句。在程序清单中的任何一处都可以插入空格符或回车符。通常书写程序时采用缩进并对齐的书写方式。
- (7) C 程序的执行总是从主函数开始, 并在主函数中结束。主函数的位置是任意的, 可以在程序的开头、两个函数间或程序的结尾。
- (8) 主函数可以调用任何非主函数, 任何非主函数都可以相互调用, 但不能调用主函数。
- (9) C 语言本身没有输入、输出语句, 输入、输出的操作是由库函数 `scanf` 和 `printf` 等函数来完成的。

1.3 C 程序的上机步骤

本书以 Turbo C 2.0 为例, 介绍 Turbo C 2.0 集成开发环境的使用, 它可运行于 PC 系列微机, 并且可在 DOS 环境中使用。本节中假定 Turbo C 2.0 已经安装在 C 盘的 C:\tc 目录中。如果在实际操作中, 安装的位置不是在 C:\tc, 操作时就应该按 Turbo C 实际安装的位置使用正确的路径。

1. 编辑源程序

有两种进入编辑状态的方法:

- (1) 设编辑的源文件名为 file.c, 可输入命令:

```
TC file✓
```

如果它是一个新文件, 则 Turbo C 屏幕窗口为空白, 等待用户输入源文件; 如果它是一个已存在的文件, 则系统将其调入内存并在屏幕窗口显示其内容, 用户可以根据需要进行修改。

- (2) 输入命令:

```
TC✓
```

这时, 屏幕出现一个窗口, 窗口示意图如图 1.1 所示。

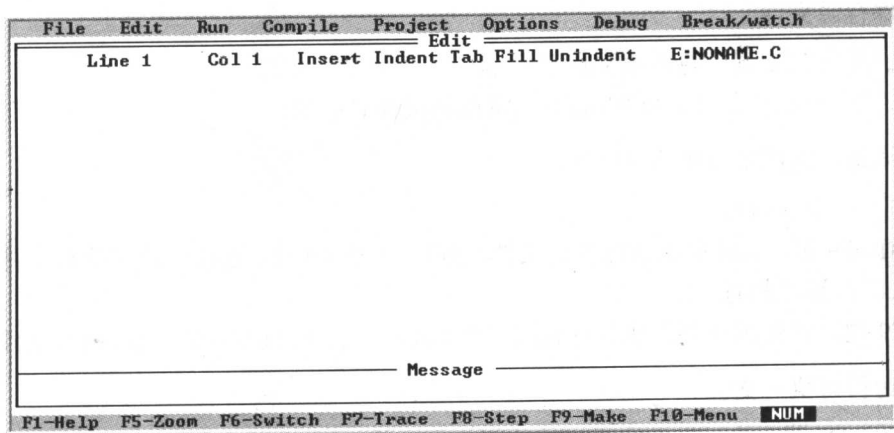


图 1.1 Turbo C 2.0 主屏幕

图 1.1 中, 最顶上一行是 Turbo C 2.0 的主菜单, 中间窗口为编辑窗口, 接下来是信息窗口, 最底下一行为参考行。以后的代码编辑、编译、调试以及运行都将在这个主窗口中进行。下面将详细介绍主菜单的内容。

在主菜单中, 除 Edit 外, 其他各项均有子菜单, 只要用 Alt 键加上菜单上某项中的第一个字母(即大写的那个字母), 就可以进入该项的子菜单中。如 Alt+F, 就进入 File 菜单项。对各主菜单及其子菜单的操作相似, 下面仅以 File 菜单为例来讲解操作方法。

按 Alt+F 组合键, 或按 F10 键, 激活主菜单, 然后按←键或→键移动屏幕上的光标到 File 上后, 再按回车键, 就可以进入 File 菜单, 该菜单的子菜单如图 1.2 所示。用↑键或↓键将屏幕上的光标移到某项后按回车键, 即可执行该项命令。

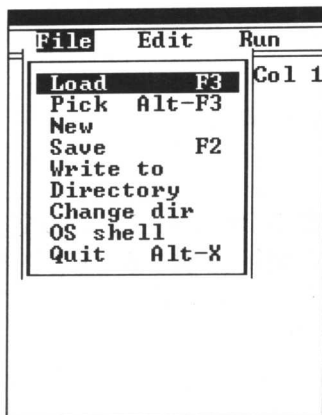


图 1.2 File 菜单

2. 保存源文件

方法一：按 F2 键，将以原文件名存盘。

方法二：选取菜单项 File/Write to，将以新文件名存盘。

3. 编译、连接生成可执行文件

方法一：分步执行。

按 Alt+F9 键，编译生成目标代码文件(.obj)；再按 F9 键，连接生成可执行文件(.exe)。

方法二：连续执行。

按 F9 键，首先编译并生成目标代码文件(.obj)；接着自动连接生成可执行文件(.exe)。

4. 运行可执行文件

按 Ctrl+F9 键，将运行刚编译连接的可执行文件。

5. 显示运行结果

按 Alt+F5 键，可以暂时退出 Turbo C，进入 DOS 环境，用户在屏幕上可以看到程序运行的结果。按任意键返回 Turbo C 状态。

 **注意：** 如果程序运行结果有错，可以转入前面第三步，重新修改源文件，然后编译、连接并运行。C 语言程序的开发过程如图 1.3 所示。

1.4 小 结

本章主要介绍了 C 语言的来源及特性，C 语言程序的基本结构，并且还详细介绍了 Turbo C 集成环境的应用，新建源程序的方法，输入、编译、调试和运行 C 语言程序的基本方法和步骤。

在 C 语言编写和开发过程中要注意以下几个问题：

- (1) 编写程序要规范，培养良好的程序设计风格，最好采用缩进并对齐的书写方式。
- (2) C 程序由一个或多个函数构成，有且只有一个主函数(main 函数)。

- (3) C 程序中的所有语句都必须以分号作为结束符，通常每一行为一条基本语句，复合语句必须用 {} 括起来。
- (4) C 语言中的所有关键字都必须小写，不能另作他用，例如想把 `int` 标识符作为变量名使用是不允许的。
- (5) 要熟练掌握源程序的编辑、保存、编译、连接及运行的操作方法，了解程序调试的基本概念。
- (6) C 程序的执行总是从主函数 `main` 开始，并在主函数中结束。

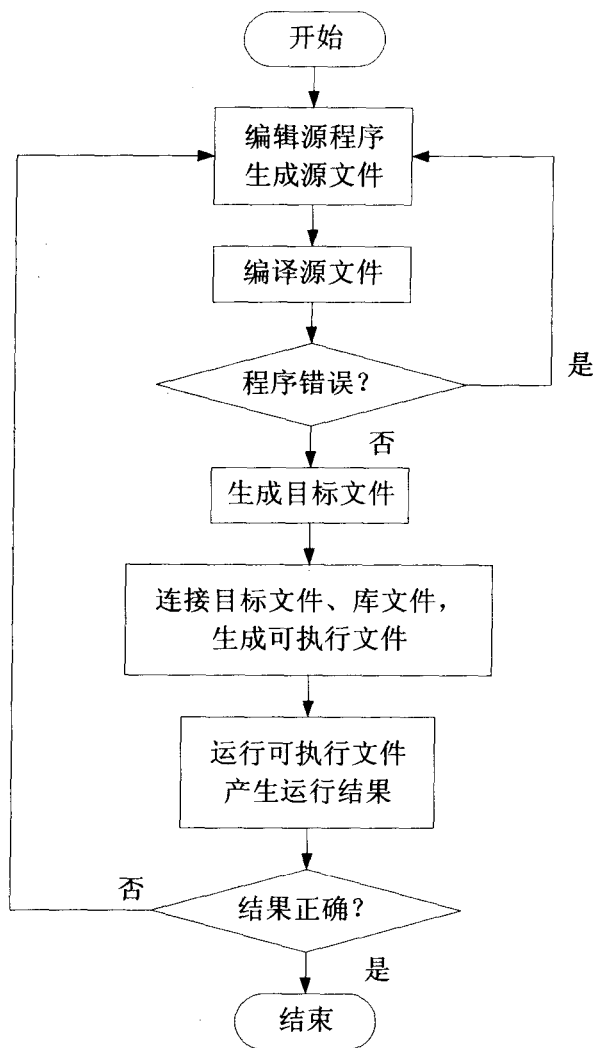


图 1.3 C 语言程序的开发过程