

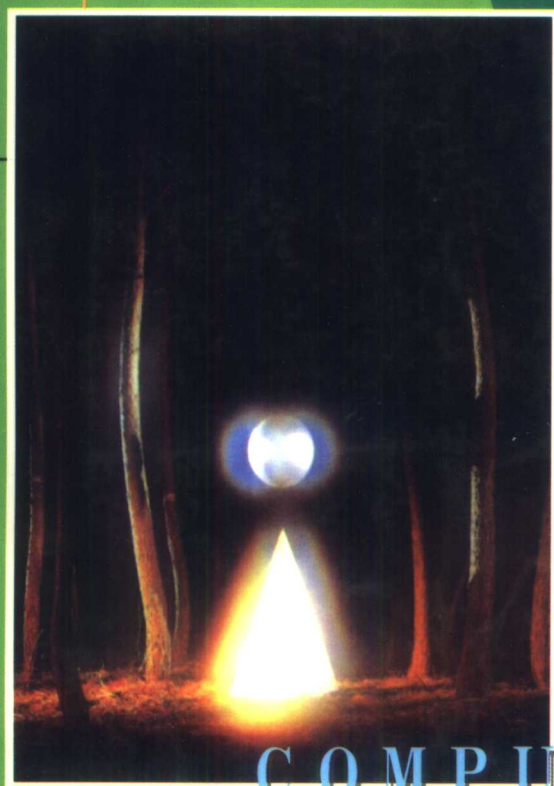
新编计算机应用基础系列教材

C 语言

C O M P U T E R

李 刚 主编

C
O
M
P
U
T
E
R



C O M P U T E R

C O M P U T E R

华 中 理 工 大 学 出 版 社

新编计算机应用基础系列教材

C 语 言

主 编	李 刚	
副主编	许中元	刘定富
编 者	金国芳	张瑞军
	朱功杰	顾守安

华中理工大学出版社

图书在版编目(CIP)数据

C 语言/李刚主编

武汉:华中理工大学出版社, 1998年11月

ISBN 7-5609-1848-4

I. C...

Ⅱ. ①李... ②许... ③刘...

Ⅲ. 计算机语言-C语言设计-教材

Ⅳ. TP363

C 语言

主 编:李 刚

责任编辑:周芬娜

*

华中理工大学出版社出版发行

(武昌喻家山 邮编:430074)

新华书店湖北发行所经销

华中理工大学出版社照排室排版

湖北省新生报印刷厂印刷

*

开本:787×1092 1/16 印张:12.25 字数:296 000

1998年11月第1版 1998年11月第1次印刷

印数:1—5 000

ISBN 7-5609-1848-4/TP·303

定价:12.50元

(本书若有印装质量问题,请向出版社发行部调换)

前 言

从世界上第一台计算机诞生至今 50 多年的时间里,产生了数百种高级语言。但使用最为广泛的高级语言不过 10 多种,C 语言就是其中之一。经过 30 余年的发展,C 语言及它的后继者 C++ 已成为当前世界上最为流行的几种计算机语言之一。众多的系统软件与应用软件都是用 C 或 C++ 写成。

C 语言众多的优点,使得越来越多的人希望学习 C 语言。甚至在许多高等学校与中等学校 C 语言也成为首选的教学用计算机语言。但是与 BASIC 等高级语言相比,学习 C 语言还是要困难一些。为了使读者减少学习中的困难,尽快掌握 C 语言的基础知识,本书重点讲授 C 语言的基本部分,并努力通过大量例题的讲解,使读者能够较快地掌握 C 语言的使用方法,分清 C 语言中容易混淆和出错的问题。对于 C 语言中初学者一时难于掌握的技巧、复杂的算法与综合应用,本书着笔较少,这样处理将使读者能集中精力学习 C 语言最关键、最主要的知识。为了提高初学者的兴趣,也为了使读者了解一些 C 语言的基本应用,在本书的最后一章介绍了 Turbo C 2.0 版的图形功能函数。另外在附录中我们附上了本书习题的参考答案,以减少读者学习中的困难。对于希望学习 C 语言中更深入部分或继续学习诸如 Visual C++、Borland C++ 的读者,本书也为他们的后续学习奠定了较好的基础。

本书由李刚主编,许中元、刘定富任副主编。参加编写的还有金国芳、张瑞军、朱功杰、顾守安。本书第一章、第七章、第十一章由李刚编写,第八章、第九章由许中元编写,第五章、第六章由刘定富编写,第四章由金国芳编写,第十章由张瑞军编写,第三章由朱功杰编写,第二章由顾守安编写。全书由李刚统稿并负责除第八章、第九章以外的习题解答与程序调试工作,许中元负责第八章、第九章的习题解答与程序调试工作。

本书全部例题和习题都在 Turbo C 2.0 版环境下运行调试通过。为了向读者提供方便,这些程序都集中在一张 3.5 英寸软盘上,可供读者选用。

本书在编写中参阅了不少著作与文献,在本书末尾列出了书目。这里仅向这些书籍的作者以及《新编计算机应用基础系列教材》编委会和为本书出版作了大量工作的华中理工大学出版社的同志们表示深切的谢意。限于我们的水平与经验,本书难免有诸多不足之处,欢迎读者批评指正。

编者

1998 年 5 月

内 容 简 介

本书以《87ANSI C》为基础,以 Turbo C 2.0 为工作平台,介绍了 C 语言的基础知识和程序设计方法,且理论与实践相结合。书中程序实例丰富,难易适中,通过实例由浅入深,富有启发性,并进一步阐述了 C 语言的基本概念和难点,文中语言简明扼要,条理清楚。此外每章均配有习题并在书后附有参考答案,便于教学和自学。

本书主要内容包括:C 语言的基本概念,C 语言的各种数据类型和运算,C 语言的模块化程序设计,C 语言的指针概念及使用,文件的基本操作及规则,C 语言的绘图功能等。

本书适用于大、中专学生与广大计算机爱好者阅读参考,也可作为相应的计算机培训班教材。

目 录

第一章 概述.....	(1)
第一节 C 语言发展概况.....	(1)
第二节 C 语言的特点与 C 语言程序设计的支持环境	(1)
第三节 Turbo C 2.0 使用简介	(3)
习题一.....	(7)
第二章 数据类型与运算.....	(8)
第一节 标识符、常量、变量与数据类型.....	(8)
第二节 基本数据类型.....	(9)
第三节 运算及运算符	(12)
习题二	(19)
第三章 结构化程序设计方法与数据的输入、输出.....	(21)
第一节 结构化程序设计方法	(21)
第二节 数据的输入和输出	(23)
第三节 C 语言的语句	(27)
习题三	(29)
第四章 选择结构与循环结构	(31)
第一节 选择结构	(31)
第二节 循环结构	(39)
习题四	(48)
第五章 数组	(53)
第一节 一维数组	(53)
第二节 二维数组	(56)
第三节 字符串和字符数组	(59)
习题五	(63)
第六章 函数与编译预处理	(66)
第一节 函数	(66)
第二节 函数调用	(69)
第三节 局部变量与全局变量	(72)
第四节 静态存储变量与动态存储变量	(73)
第五节 编译预处理	(76)
习题六	(80)
第七章 指针	(83)
第一节 指针的概念	(83)
第二节 指针变量	(84)
第三节 指针与数组	(87)

第四节	指针与字符串	(93)
第五节	函数参数中的指针	(96)
第六节	指针与函数	(101)
第七节	指针数组	(106)
第八节	指向指针的指针	(108)
第九节	main 函数中的参数	(109)
第十节	指针数据与指针运算小结	(110)
习题七		(111)
第八章	结构体与共用体	(114)
第一节	结构体变量的定义与引用	(114)
第二节	结构体数组	(119)
第三节	指向结构体类型数据的指针	(122)
第四节	共用体类型	(125)
习题八		(128)
第九章	位运算	(130)
第一节	按位运算	(130)
第二节	移位运算	(134)
习题九		(135)
第十章	文件	(136)
第一节	C 文件概述	(136)
第二节	C 文件的打开与关闭	(137)
第三节	文件的读写函数	(139)
习题十		(146)
第十一章	绘图	(148)
第一节	图形绘制的硬件基础	(148)
第二节	Turbo C 2.0 的图形函数	(149)
第三节	绘图程序举例	(153)
习题十一		(157)
附录一	ASCII 字符编码一览表	(158)
附录二	Turbo C 编译错误信息表	(159)
附录三	习题参考答案	(164)
参考文献		(188)

第一章 概 述

C 语言因具有严谨的结构化程序设计方法、良好的可移植性、丰富的数据结构、简捷的语言风格及具有高低级语言功能,而成为颇受欢迎的计算机高级语言之一。本章对 C 语言的发展概况、语言特点、运行环境及 Turbo C 2.0 集成软件给以综述。

第一节 C 语言发展概况

1967 年英国剑桥大学 Martin Richards 在 CPL 语言的基础上开发了 BCPL (Basic Combined Programming Language) 语言。1970 年美国贝尔实验室的 K. Thompson 对 BCPL 语言作了改进和简化,推出了 B 语言(“B”字取自 BCPL 的第一个字母)。B 语言能对计算机的某些硬件进行操作,语言简练。K. Thompson 用 B 语言重写了 UNIX 操作系统,实现了长期以来未能实现的用高级语言编写系统软件梦想。1973 年前后,贝尔实验室的 D. M. Ritchie 在 B 语言的基础上设计了 C 语言(“C”字取自 BCPL 的第二个字母)。D. M. Ritchie 与 K. Thompson 合作,用 C 语言将 UNIX 操作系统的核心程序与外层软件又重写了一篇,即 UNIX 系统第 5 版。1977 年《可移植 C 语言编译程序》的出现,使得 C 语言易于移植到各种类型的计算机上。也使得各种类型计算机都能使用 UNIX 操作系统。在这以后又出现了 C 语言的各种版本。1983 年,美国国家标准局成立的专门委员会对 C 语言的各种不同版本加以整理、补充和修改,确定了 C 语言的新标准,称为《ANSI C》。以后的 C 语言的各种编译系统都向《ANSI C》靠拢。1987 年美国国家标准局又公布了《87 ANSI C》。

经过 30 余年的发展,C 语言及它的后继者 C++ 已成为当前世界上最为流行的几种计算机语言之一。众多的系统软件(如关系数据库 FoxBASE+ 等)与应用软件都是用 C 或 C++ 写成。现在各种版本的 C 语言编译系统虽各有不同,但其基本部分是相同的。本书以《87 ANSI C》为基础,以 Turbo C 2.0 为工作平台,讲授 C 语言的基础知识和程序设计方法。

第二节 C 语言的特点与 C 语言程序设计的支持环境

一、C 语言的特点

C 语言之所以能成为当前世界上少数几种广为流行的计算机语言,并有可能取代 PASCAL 语言成为主要的教学语言是与 C 语言的特点分不开的。C 语言的特点可归纳如下:

① C 语言是面向过程的高级语言。机器语言和汇编语言是面向机器的,各种不同类型的计算机所使用的机器语言和汇编语言也是不相同的。C 语言与其它高级语言不是面向具体的机器,而是面向解题的过程。用 C 语言编制程序,可以不必熟悉计算机内部的具体构造和机器指令。

② C 语言是一种优秀的结构化语言。C 语言提供了编写现代程序所需要的各种数据结构

和控制结构。以函数为主的程序设计风格,使 C 语言能够方便地实现结构化的程序设计方法。

③ C 语言组成精练、简洁,使用方便、灵活。C 语言只有 32 个关键字,45 种运算符,9 种控制语句。C 语言程序书写形式自由,压缩了一切不必要的成分。与其它高级语言(如 PASCAL 语言)相比,C 语言显得更为简洁、精练。

④ C 语言具有丰富的数据类型与运算符。C 语言具有现代编程语言所应有的各种数据类型,能够实现各种复杂的数据结构的运算。C 语言的运算符十分丰富(有 45 种),这些运算符能构成各种各样的表达式。灵活运用这些运算符和表达式,使得 C 语言能够实现其它高级语言所无法实现的运算。

⑤ C 语言既具有高级语言又具有低级语言的功能。C 语言具有一般高级语言(如 BASIC 语言、PASCAL 语言)易于使用的特点,又有一些其它高级语言所不具有的功能。这些功能与汇编语言(低级语言)的功能接近,能实现对计算机硬件的控制。但掌握 C 语言比掌握汇编语言容易,现在很多系统软件都用 C 或 C++ 编写,编制的难度要比只用汇编语言小得多。

⑥ C 语言有良好的可移植性。C 语言的可移植性好,在一种机型上编制的 C 语言源程序可以不加修改或只作少量修改便可在另一种机型上运行。这是因为 C 语言所提供的语句并不直接依赖于硬件,有关硬件的操作,例如输入、输出等都是通过调用系统提供的库函数来实现的,而库函数本身并不是 C 语言的组成部分。

⑦ C 语言程序执行速度快。C 语言程序所生成的目标代码质量高,程序执行速度比其它高级语言快。

⑧ C 语言程序书写自由。C 语言程序书写灵活,一行可以书写多个语句,一个语句也可以分开写成几行。但其编译程序对 C 语言源程序中的语法错误的检查不够严格,这增加了 C 语言源程序调试的难度。数据类型、运算符的多样性使得学习 C 语言比一般高级语言困难,这就要求 C 语言的学习者和使用者应较为熟练地掌握 C 语言的语法规则,并通过较多的上机操作掌握 C 语言程序的调试技巧。

C 语言的突出优点使得大量的系统软件与应用软件可使用 C 或 C++ 编写。C 语言得到越来越广泛地运用。

二、C 语言程序设计的支持环境

现代的计算机都配备有操作系统。操作系统的作用是有效地组织和利用计算机的硬、软件资源,是计算机最重要的系统软件。用 C 语言进行程序设计,程序人员应了解自己所使用的计算机的类型及所使用的操作系统。因为在不同的机型、不同的操作系统之下,运行 C 语言程序的方式和命令是不同的。本书以 IBM-PC 微型计算机、PC DOS 操作系统为硬、软件支持环境,讲授 C 语言程序设计的基本方法。

IBM-PC 微型计算机及其兼容机系列是当今世界上使用最为广泛的计算机类型。全世界数亿台微型计算机中的 80% 是 IBM-PC 微型计算机及其兼容机。我国 95% 以上的微型计算机也是这种机型。PC DOS(或 MS DOS)是运行在 IBM-PC 系列微机上的一种最简单、最常用的操作系统。

以 IBM-PC 机与 PC DOS 操作系统为支持环境也还有数种不同的 C 语言编译系统,如 MSC、Turbo C 等。本书以 Turbo C 2.0 版为 C 语言程序的开发环境。

第三节 Turbo C 2.0 使用简介

Turbo C 是美国 Borland 公司在采用了美国国家标准局(ANSI)建议的 C 语言标准的基础上而推出的集编辑、编译、调试、运行为一体的集成软件,当前使用最为广泛的是 Turbo C 2.0 版。

Turbo C 提供了两种开发 C 语言程序的方式。

一、命令行方式

使用命令行方式时,需要使用 Turbo C 中的 TCC.EXE 文件。在使用 TCC.EXE 之前,用户可用某一种文本编辑程序(如 WORDSTAR、EDIT、WPS 等)编辑自己的 C 语言源程序,源程序扩展名为 .C。源程序编辑完毕存盘后,在 DOS 中使用改变当前目录命令(即 CD 命令)将 TCC.EXE 文件所在的目录作为当前目录。在操作系统提示符下键入 TCC 及源程序文件名即可对源程序进行编译、连接并生成可执行文件(.EXE 文件)。然后在操作系统提示符下直接运行可执行程序文件即可得到结果。

例 1-1 在字处理软件 WPS 中用 N 命令方式(或用 DOS 中的 EDIT 命令)在 C 盘的根目录下建立 example1.c 文件,文件内容是下面的 C 语言源程序。

```
main( )
{ printf("This is a C program.");
}
```

说明:

① C 语言程序要求用小写字母书写(只在某些情况下可以用大写,如字符串中的大写字母;符号常量等)。

② main 是 C 语言中的主函数名,每一个 C 语言程序文件都由一个或多个函数组成,但主函数 main() 是必不可少的。

③ 这里的花括号 { } 及其中的语句是函数体。每一个语句都必须用分号“;”结尾。

④ printf() 是标准输出函数。

⑤ 这个 C 语言程序的功能是在屏幕上输出英文字符串 This is a C program. 。

用命令行方式编译、连接该程序的方式如下:

若 Turbo C 2.0 版装在 C 盘一级子目录 TC 下,当前目录是 TC,屏幕显示 DOS 提示符为

```
C:\TC>
```

因为源程序在 C 盘根目录下,文件名为 example1.c,在提示符下键入 TCC\example1.c 并回车。Turbo C 编译并连接源程序文件生成目标文件 example1.obj 与可执行文件 example1.exe。这两个文件都在当前目录即 TC 一级子目录之下。

在 C:\TC> 提示符下键入 example1 并回车,运行可执行文件,屏幕显示运行结果: This is a C program. 。

上面例题表明开发一个 C 语言程序的基本过程是编辑、编译、连接、执行。

1. 编辑

选择文本编辑程序(如 EDIT、WPS 等)将 C 语言源程序输入到计算机中,并以文件形式

存盘。在 DOS 操作系统下,源程序文件名均以 .C 为文件扩展名。

2. 编译

C 语言源程序文件需使用 C 语言编译程序对其进行编译(如 Turbo C 中的 TCC.EXE),如果在源程序编译过程中发现语法错误,应回到编辑阶段对源程序文件进行修改,然后再重新进行编译。在 DOS 操作系统下,源程序文件经过编译后得到目标文件,目标文件的扩展名为 .OBJ。

3. 连接

源程序文件经过编译后得到的目标文件还不能直接执行,需要经过连接程序的连接才能生成可执行文件(TCC 既有编译又有连接功能)。在 DOS 操作系统下,目标文件连接后得到可执行文件,可执行文件的扩展名为 .EXE 。

4. 运行

在 DOS 操作系统提示符下,输入可执行文件的路径及文件名即可运行可执行文件,得到 C 语言程序最后运行的结果。

二、利用 Turbo C 的集成开发环境方式

利用 Turbo C 的集成开发环境可以将 C 语言源程序的编辑、编译、连接、运行 4 个步骤在一个窗口中进行,用户使用十分方便。

若 Turbo C 2.0 版在 C 盘 TC 一级子目录下,集成环境程序文件名为 TC.EXE。在提示符 C:\TC> 下,键入 TC 再回车即可进入 Turbo C 集成环境。按 F10 键后再利用左、右、上、下光标键可以看见命令菜单。在图 1-1 中给出了最常用的命令菜单栏目,即 File (文件)与 Run(运行)。

File	Edit	Run	Compile	Project	Options	Debug	Break/watch
Load		F3		Run		Ctrl-F9	C:\NONAME.C
Pick		Alt-F3		Program reset		Ctrl-F2	
New				Go to cursor		F4	
Save				Trace into		F7	
Write to				Step over		F8	
Directory				User screen		Alt-F5	
Change Dir							
OS shell							
Quit		Alt-X					
Message							
F1-Help	F5-Zoom	F6-Switch	F7-Trace	F8-Step	F9-Make	F10-Menu	

图 1-1

Turbo C 2.0 版集成环境功能繁多,下面将常用的命令与功能键概述如下:

1. File (文件)

File 栏目下有 9 个子命令,主要解决文件的命名、存取等问题,现将常用的 7 个子命令介绍如下:

(1) Load (装载命令)

该命令装载一个在磁盘上已存在的源程序文件。用户输入磁盘上文件的路径与文件名后,再回车,即可装载所需源程序文件。集成系统也能够给出已存在文件的名称,多次回车出现当前目录下的文件列表时,用户可以选择已存在文件。

(2) Pick (选择命令)

该命令选择最近已编辑过的源程序文件。

(3) New (新文件命令)

该命令编辑一个新的源程序文件,文件名先由系统给出,名为 NONAME.C。当使用 Save 命令存盘时,用户必须给出正式文件名。当编辑一个新文件时,建议先用 Save 命令存盘,定好源程序的正式名称,尽量避免使用 NONAME.C 文件名。

(4) Save (存盘命令)

该命令将当前窗口中的源程序文件存盘。

(5) Write to (存盘命令)

该命令将当前窗口中的源程序文件写到给出文件名的文件中,若给出的文件名文件已经存在,则会改写这个文件。

(6) OS shell (暂时退出命令)

该命令暂时退出 Turbo C,转到 DOS 提示符下,可用 EXIT 命令返回到 Turbo C。

(7) Quit (退出命令)

该命令退出 Turbo C,返回到 DOS 提示符下。

2. Edit (编辑)

Edit 栏目下没有子命令。当装载一个文件或建立一个新文件后,集成环境自动转为 Edit 状态。在 Edit 状态下,常用的编辑键有:

- ① ← 左删键。删除光标左边的字符。
- ② 删除键。删除光标所在的字符。
- ③ <Ins> 键插入、改写转换键。
- ④ <Home> 键。光标到行首。
- ⑤ <End> 键。光标到行尾。
- ⑥ <PgUp> 键。向上翻页。
- ⑦ <PgDn> 键。向下翻页。
- ⑧ <Ctrl>+N。插入一行。
- ⑨ <Ctrl>+Y。删除光标所在行。
- ⑩ <Ctrl>+KB。设置块首。
- ⑪ <Ctrl>+KK。设置块尾。
- ⑫ <Ctrl>+KV。块移动。
- ⑬ <Ctrl>+KC。块复制。
- ⑭ <Ctrl>+KY。块删除。
- ⑮ <Ctrl>+KH。块取消。
- ⑯ <Esc> 回退键。取消命令菜单。
- ⑰ <Ctrl>+<Break>。中断程序运行。

3. Run (运行)

Run 栏目下有 6 个子命令。常用的有 Run 命令、Program reset 命令与 User screen 命令

(1) Run (运行命令)

源程序编辑完毕,即可使用该命令。这个命令集编译、连接、运行于一体,如果编译过程中没有发现语法错误,则可生成可执行文件(.EXE),并可用 User screen 命令或<Alt>+F5 键查看运行结果。如果编译过程中发现语法错误,则会出现错误信息。该命令的快速键是<Ctrl>+F9。

(2) Program reset (程序重启命令)

该命令使程序重新启动,中止当前的调试,返回编辑状态。

(3) User screen (使用屏幕命令)

可以使用该命令查看程序最后运行的结果。该命令的快速键是<Alt>+F5。

在图 1-1 中已列出各种常用的功能键与快速命令键。使用特别频繁的是<Ctrl>+F9、<Alt>+F5、F6 与 F10。功能键 F6 的作用是激活编辑窗口,当希望回到编辑窗口修改源程序时,常使用 F6 键。功能键 F10 的作用是从活动窗口转到主菜单,或从任一菜单转到活动窗口。

一般情况下,一个 C 语言源程序不经过调试,一次运行成功的可能性是很小的。这可能是程序复杂,用户在编制程序时对问题的考虑不够周密,也可能是对 C 语言的语法结构未能熟练掌握,还有可能是出现字符输入错误。正因为如此,C 语言的学习者应多上机运行、调试程序,积累调试程序的经验,增强调试程序的能力。

如果一个 C 语言源程序出现语法错误,在编译过程中 Turbo C 会给出警告或错误信息,以帮助程序编制人员排出错误。

例 1-2 给出下列源程序

```
main( )
{ int a,b,c;          /* 定义 a,b,c 为整数类型变量 */
  a=3;                /* 将整数 3 赋值给变量 a */
  b=4;                /* 将整数 4 赋值给变量 b */
  c=a+b;              /* 将 a 与 b 的和赋值给变量 c */
  printf("%d\n",c);    /* 用整数输出 c 的值 */
}
```

说明:

- ① 符号 /* 与 */ 之间的内容为注释。
- ② 格式符 %d 表示输出十进制整数。
- ③ 控制符 \n 表示换行。

用 Turbo C 集成环境调试、运行该程序的操作过程如下:

在 C:\TC> 提示符下键入 TC 并回车,进入 Turbo C 集成环境编辑窗口。按 F10 键出现命令行菜单,用下光标键选择 Save 命令,回车后出现一个弹出式菜单,但文件名为 NONAME.C,更改文件名为 EXAMPLE2 (扩展名 .C 可由系统给出),再回车。

输入源程序内容,编辑窗口中显示源程序。用 Run 命令或<Ctrl>+F9 键运行该程序,若程序输入无误,编译、连接很快进行完毕,可用 User screen 命令或<Alt>+F5 键查看运行结果(屏幕显示结果为 7,按任意键又回到编辑窗口)。若源程序输入有错误,则在屏幕中间出现编译信息,在屏幕下方出现警告与错误信息。按任意键后,编译信息消失,出现第一次语法错误的行(或下一行)呈现反象显示,提示错误位置。在分析错误与警告信息后,按 F6 键回到编辑

窗口,修改源程序,再运行、调试直至正确为止。

在编译和调试源程序时,Turbo C 通过信息窗口(Message)给出错误信息与警告信息,同时在编辑窗口(Edit)中指出错误在源程序中的相应位置。

警告信息并不阻止编译的进行过程,但指出源程序中值得怀疑的情况,由程序编制者自己决定是否应该修改源程序。

错误信息分为致命错误与一般错误,致命错误极为罕见,但一般错误却经常出现。Turbo C 2.0 版列举了 139 条一般错误与 30 条警告(详见附录二)。

如果编译程序发现多处错误,关键是修改第一处错误,有时第一处错误经过重新编辑修改再运行,错误信息会大大减少。

现将 Turbo C 2.0 中最常见的一般错误信息及基本处理办法列举如下:

(1) Declaration syntax error

说明中出现语法错误。一般是在源文件中某一个说明中丢失了某些符号或有多余的符号。

(2) Expression syntax

表达式语法错误。出现这个错误,通常是由于出现两个连续的操作符,括号不匹配或缺少括号,或者前一句遗漏了分号“;”所引起的。

(3) Statement missing ;

语句缺少分号“;”。出现这个错误是因为编译程序发现某一表达式语句后没有分号“;”,当这个错误信息是第一条错误信息时,往往是光标所在位置的前面一条语句缺少分号。

(4) Undefined symbol 'xxxxxxx' in function main

在主函数中没有定义符号 'xxxxxxx'。一般是光标所在位置的标识符出现拼写错误,或这个标识符没有定义就使用了。

说明:

用来标识变量名、符号常量名、函数名、数组名、类型名、文件名的有效字符序列称为标识符(identifier)。也就是说,标识符就是一个名字。C 语言规定标识符只能由字母、数字和下划线 3 种字符组成,且第一个字符必须为字母或下划线。

习 题 一

1-1 C 语言的主要特点是什么?

1-2 熟悉 Turbo C 2.0 的基本使用方法。

1-3 上机运行本章的两个例题。

1-4 编写一个 C 语言程序,输出以下信息:

```
* * * * *
```

```
How do you do !
```

```
* * * * *
```

1-5 编写一个 C 语言程序,求 3 个整数的和并输出结果。

第二章 数据类型与运算

C 语言程序中所用到的每一个常量、变量及函数都是 C 语言的基本操作对象,它们都与一种数据类型相联系,即每一个常量、变量及函数都有其数据类型。每种数据类型又都有一定的取值范围及能够在这个取值范围内进行的运算。

第一节 标识符、常量、变量与数据类型

一、标识符

所谓标识符,是指用来标识程序中用到的变量名、函数名、类型名、数组名、文件名以及符号名等各类名称的有效字符序列。

标识符的命名规则是:由字母(大、小写均可)、数字及下划线组成,且第一个字符必须是字母或下划线。

如:sum-1、TEST_one、year_x1 等是有效标识符,而 4th、#123、“x”、.com、1-2-3、1997y、order-no 等是不合法的标识符。

对于大部分 C 语言编译系统,标识符的前面 8 个字符是有效字符,而后面字符无效。如变量名 studentname 与 studentnumber 被认为是同一变量。当一个命名较长时,可采用缩写方式,如 stu_name 与 stu_num 就认为是不同的标识符。

注意:

C 语言中大、小写字母是有区别的,如果是标识符,就会作为不同标识符来看待,而且 C 语言中主要用小写字母,如 C 语言中的关键字 int、float、if 等就要求用小写字母书写。

二、常量

常量是指在程序运行过程中,其值不变的量。如整型常量 5,实型常量 6.4,字符型常量 'A' 等。

C 语言中还有符号常量,即用一个名字(字符序列)来代表一个常量,符号常量一般使用大写。如命令行 #define PI 3.14159265 定义了符号常量 PI,PI 代表常量 3.14159265。

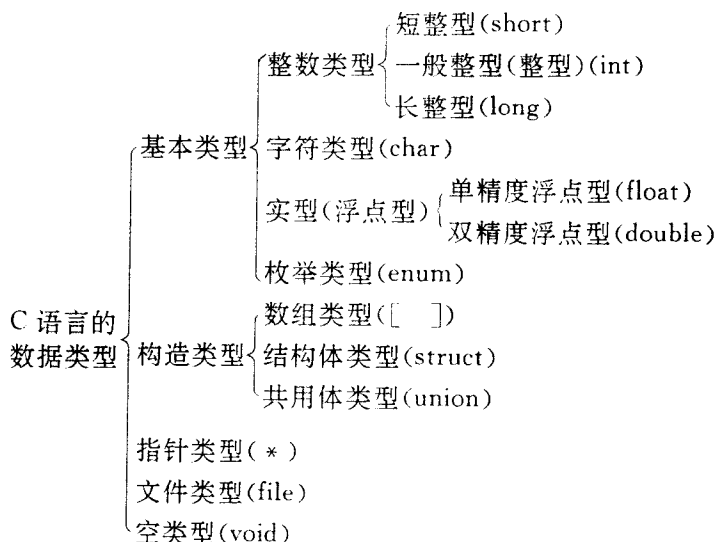
三、变量

变量是指在程序运行过程中,其值可以改变的量。不同类型的变量占据大小不同的内存空间,它用来存储相应变量的值。

变量名是一种标识符,所以其命名规则应与标识符命名规则相同。变量也有其数据类型,如整型变量,实型变量,字符型变量等。语句 “int a,b,c;” 定义了 3 个整数类型变量 a、b、c。

四、数据类型

C 语言中的数据类型可作如下划分(括号中的内容为类型说明符):



在基本类型中,还有一些修饰词,如 short、long、unsigned、signed 等。

在第二节中,将讲述基本数据类型,其它数据类型将在以后各章中讲述。

第二节 基本数据类型

一、整数类型(整型)

C 语言中的整数类型分为一般整型(int)(有时也将一般整型 int 简称为整型)、长整型(long int)、短整型(short int)和无符号整型。而无符号整型又有无符号一般整型(unsigned int)、无符号长整型(unsigned long)、无符号短整型(unsigned short)。

整型常量是不带小数点的整数。但整型常量可以有 3 种表示方式,即十进制、八进制与十六进制。12、-34、0 是十进制整型常量。0101 是以 0 (零)开头的整数,表示八进制整型常量,即十进制整型常量 65。0x41 是以 0x (零 x)开头的整数,表示十六进制整型常量,也即十进制整型常量 65。

整型变量可用来存放整型数据,如语句“int a,b;”定义 a、b 为一般整型变量。可以将整型常量赋值给整型变量,如语句“a=5;”与语句“b=-3;”就将 5 赋值给 a,将 -3 赋值给 b。

在定义整型变量时,也可同时给变量赋值。如语句“int a=5,b=-3;”既定义了整型变量,同时也给变量赋值。下面对整数类型分类讲述。

1. 一般整型

语句“int x;”定义 x 为一般整型。一般整型(int 型)数据在内存中占 2 个字节(16 位),值域为 -32768 至 32767 (对于不同的机型,同一种数据类型的值域是不同的,本书只以目前常用的 IBM-PC 微型计算机为例)。

2. 长整型

语句“long int x;”(或“long x;”)定义 x 为长整型变量。long int 型数据在内存中占 4

个字节(32 位),其值域为 -2147483648 至 2147483647 。

如果在整型常量的末尾添上字母 L 或 l,就构成了长整型常量,如语句“ $x=23456L$;”将长整型常量 23456 赋给长整型变量 x。

3. 短整型

语句“`short int x;`”(或“`short x;`”)定义 x 为短整型变量。短整型变量用于存放比较小的整型数,目的是节省计算机内存。对于 IBM-PC 机,short int 型数据和 int 型数据均占用 2 个字节的内存,它们之间无明显区别,其值域均为 -32768 至 32767 之间。

4. 无符号一般整型

语句“`unsigned int x;`”(或“`unsigned x;`”)定义 x 为无符号一般整型。无符号一般整型变量用于存放大于或等于 0 的整数,也是占 2 个字节,故其值域为 0 至 65535 之间。

5. 无符号短整型

语句“`unsigned short x;`”定义 x 为无符号短整型变量,无符号短整型与无符号一般整型之间也无明显区别。

6. 无符号长整型

语句“`unsigned long x;`”定义 x 为无符号长整型变量,也是用于存放大于或等于 0 的整数,但占 4 个字节,其值域为 0 至 4294967295。

二、实型

C 语言中的实型数据也称为浮点型数据。

1. 浮点型常量数据的两种表示形式

(1) 十进制形式

十进制形式由数字和小数点组成。

如 5.6, -7.2 , 3.1415927。

(2) 指数形式

指数形式即科学记数法。

如 102300.0 可表示为 $1.023e5$; -0.00256 可表示为 $-2.56e-3$ 。

2. 浮点型变量的分类

浮点型变量分为单精度浮点型(float)和双精度浮点型(double)。

(1) 单精度浮点型

语句“`float a,b;`”定义 a、b 为单精度浮点型变量。单精度浮点型数据在内存中占 4 个字节,其值域约为 10^{-38} 至 10^{38} 。

如语句“`float a=3.65,b=-6.42;`”定义了单精度浮点型变量 a、b,并且 a 得到赋值 3.65, b 得到赋值 -6.42 。

(2) 双精度浮点型

语句“`double a,b;`”定义 a、b 为双精度浮点型变量。双精度浮点型数据在内存中占 8 个字节。其值域约为 10^{-308} 至 10^{308} 。

三、字符类型

1. 字符类型常量

字符类型常量数据是用单引号括起来的单一字符,如 'p'、'*'、'#'、'4' 等,它在内存中占 10。