

华东高校计算机基础教育研究会推荐教材

# C/C++ 程序设计教程

主 编 唐 全  
副主编 海 滨  
胡建华

图书馆

东南大学出版社

中国水利水电出版社

上海交通大学出版社

华东高校计算机基础教育研究会推荐教材

# C/C++ 程序设计教程

(修 订)

主 编 唐 全

副主编 海 滨 胡建华

主 审 李秀山

参 编 (按姓氏策划)

王 锋 赵鸿萍 胡建华 徐圣林

唐 全 海 滨

东南大学出版社  
中国水利水电出版社  
上海交通大学出版社

## 内 容 提 要

C语言是一种通用的程序设计语言,它既可编写应用程序软件,也可编写系统程序软件。因此,C语言近年来受到越来越多读者的青睐。

本书首先从C语言的发展、特点和它的基本结构、基本运算讲起,然后详细地介绍了C语言的各类语句、数组、函数与变量、指针和文件,最后进一步讨论了C语言的发展,从C过渡到C++。本书的编写力求内容简明实用,概念完整清晰,文字通俗易懂。为了便于读者阅读和理解,书中还附有典型的例题和习题。

本书可作为大专院校计算机基础教育“C/C++程序设计”课程的教材和教学参考书,也可作为计算机培训教材和自学用书。

### 图书在版编目(CIP)数据

C/C++ 程序设计教程/唐全主编.—2版(修订本).

南京:东南大学出版社,2002.8

华东高校计算机基础教育研究会推荐教材

ISBN 7-81050-544-0

I. C... II. 唐... III. C语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2001)第 080013 号

东南大学出版社出版发行  
(南京四牌楼2号 邮编 210096)

出版人:宋增民

江苏省新华书店经销 溧阳市晨明印刷有限公司印刷  
开本:787mm×1092mm 1/16 印张:18 字数:450千字

2002年8月第2版 2002年8月第2次印刷

印数:4001—8000 定价:24元

(凡因印装质量问题,可直接向发行科调换。电话:025-3792327)

## 序

为了适应我国高校面向 21 世纪计算机基础教育的发展和需要,华东高校计算机基础教育研究会于 1998 年 11 月在浙江金华召开了理事扩大会,对高校计算机基础教育的教材建设问题进行了专题研讨。会议认为,华东地区经济发达、科教先进,高校多达 300 余所,而现有的计算机基础教育的教材建设与现有的地区优势极不相符。80 年代中期华东高校计算机基础教育研究会曾组织出版过一批深受读者欢迎的计算机教材。面对当前计算机科学与技术的飞速发展,计算机基础教育已成为理、工、农、医、商、经民政治、文化、艺术等各行各业的公共基础教育。培养大批掌握计算机科学知识与应用技能的跨世纪高级人才,已成为历史赋予高校的一项重要任务。为此,加强高校计算机基础教材建设已提到重要议事日程,学会决定组织力量,编写一套面向 21 世纪的、适应高校计算机基础教学需要的新教材,推动华东高校计算机教育事业的发展。

学会于 1999 年 1 月在南京召开了华东地区高校计算机基础教育教材编委会第一次会议,编委会由浙江大学、上海交通大学、东南大学、同济大学、华东理工大学等知名高校的专家学者及上海交通大学出版社、东南大学出版社、中国水利水电出版社的代表共同组成。学会特邀中国工程院院士、浙江大学校长潘云鹤教授和中国工程院院士、东南大学校长顾冠群教授担任编委会名誉主任;由学会会长张森教授任编委会主任,学会副会长李文忠教授任编委会副主任,学会秘书长赵民德兼编委会秘书长。编委会汇集了浙江大学、上海交通大学、东南大学、复旦大学、华东师范大学等数十所院校长期从事高校计算机基础教育、有丰富教学实践经验的资深教师共同研讨,确定编写“华东高校计算机基础教育教材”第一批教材计 21 种,由上海交通大学出版社、东南大学出版社、中国水利水电出版社分别负责出版发

行,并作为华东高校计算机基础教育研究会的推荐教材面向大专院校。

教材是教学过程中的“一剧之本”,是当前高校计算机教学的首要问题。在编委会的领导下,经过参编教师的辛勤劳动和三家出版社的共同努力,编写及出版工作进展顺利,预计 2000 年可全部推出。第二批教材的组织准备工作正在进行中。

三家出版社联合策划、分工协作、联合出版、联合发行,在华东乃至全国还是首创,得到了教师和同行们的赞赏。

教材建设是一项长期艰巨的系统工程,尤其是计算机科学技术发展迅速,更新快,因此,教学内容就要不断更新。为使教材更新跟上科学技术的发展,本会将密切注视计算机科学技术的发展新动向,使我们的教材编写不断推陈出新,逐步与国际接轨,不断提高教材质量,为华东高校计算机基础教育的教材建设作出应有的贡献。

华东高校计算机基础教育研究会

1999 年 10 月

# 华东高校计算机基础教育研究会

## 教材编委会名单

名誉主任：潘云鹤 顾冠群

主任：张 森

副主任：李文忠

秘书长：赵民德

编委(以姓氏笔画为序)：

石 冰	石文俊	宁正元	东鲁红
冉榴红	朱 敏	吕 刚	李正凡
张钧良	张 煦	陈 轩	陈凤兰
杭必政	周金辉	俞俊甫	徐安东
秦 军	詹国华	薛永生	

## 前 言

C语言是一种通用的程序设计语言,它在国内外已得到广泛使用,并受到一致推崇。C语言作为UNIX操作系统的主力语言,随着UNIX的发展而日趋普及和完善。使用C语言编制的程序具有简洁灵活、符合结构化程序设计的要求、可移植性好等特点。C语言既具有高级语言的一般功能,又有许多低级语言的特点,所以C语言应用面极广,可用来开发各种应用程序。同时,C语言与UNIX系统的Shell命令相结合,向用户提供了良好的软件开发环境。因此,目前全国高校许多专业,包括一些非计算机专业都开设了C语言程序设计这一课程。通过C语言的学习,我们可以掌握计算机解决实际问题的方法和手段,也可以为进一步学习计算机技术打下良好的基础。

本书由华东高校计算机基础教育研究会安排编写,是原版《C/C++程序设计》一书的修订本。本书编者在总结多年教学经验和教改实践的基础上,根据教育部关于普通高校计算机基础教育三个层次的指导性意见和“面向21世纪课程教材”的要求,并结合计算机科学中面向对象技术的发展,组织编写了这本《C/C++程序设计教程》。它可作为普通高校计算机基础教育“C/C++程序设计”课程的教材,也可作为计算机培训教材和自学用书。

C语言数据类型丰富,概念比较复杂,使用十分灵活,编程容易出错,特别是指针的概念,对于初学者来说,不容易理解。因此本书在内容安排上力求科学、全面、实用,强调语言基础,强化应用,遵循循序渐进的原则。首先介绍了C语言的基础内容(1~5章),包括C语言的发展、特点和基本数据类型,基本运算,顺序、选择、循环三种程序设计的基本结构;然后详细介绍了C语言的核心内容(6~11章),包括C语言中的数组、预编译技术、函数、结构体、指针和文件等内容;最后再用3章的篇幅(12~14章)简要介绍了面向对象的程序设计语言C++,包括C++的特点、C++与C语言的联系与主要区别、类和对象、继承和派生等内容。

值得一提的是,人们对C语言程序设计的学习往往是“入门容易得到难”。如果读者学习了一本C语言入门书,动手编程并上机解题并非难事,但要真正理解并掌握C语言的精髓,充分发挥C语言编程潜力,则有一定的难度。确实,C语言实在太灵活,功能太强,也很难全面地理解它、掌握它。但是,世上无难事,读者一定是个有心人,“入门并不难,进一步深入学好C语言也是办得到的”。C程序设计是一门强调实践的课程,希望读者通过编写程序和上机实践,取得融会贯通、熟能生巧的学习效果,培养良好的程序设计风格,真正具备使用

C/C++ 语言编写简洁高效程序的能力。让我们齐心协力、专心致志把 C/C++ 程序设计课程学好。

本书由唐全教授任主编,海滨、胡建华任副主编,北京科技大学计算机科学系李秀山教授主审。海滨编写了第 1、2、10 章,唐全编写了第 3、4、5、7 章,赵鸿萍编写了第 6、8 章,徐圣林编写了第 9 章,王锋编写了第 11 章,胡建华编写了第 12、13、14 章。本书的出版得到了华东高校计算机基础教育研究会和东南大学出版社的大力支持,另外中国药科大学李辉老师不辞劳苦地做了许多文字工作,编者在此一并致谢。

由于时间仓促和编者水平所限,对于书中错误和不当之处,衷心希望广大读者不吝赐教。

编者

2002 年 4 月于南京



# 目 录

<b>1 C语言概述</b> .....	(1)
1.1 C语言的由来与发展 .....	(1)
1.2 C语言的特点 .....	(2)
1.3 C语言的基本结构 .....	(3)
1.3.1 字符集 .....	(3)
1.3.2 标识符 .....	(3)
1.3.3 关键字 .....	(4)
1.3.4 运算符 .....	(5)
1.3.5 C程序的结构 .....	(6)
1.3.6 C程序的书写 .....	(7)
1.4 C程序的上机方法 .....	(8)
1.4.1 C程序实现步骤 .....	(8)
1.4.2 程序的执行 .....	(9)
1.4.3 TURBO C上机操作 .....	(9)
1.4.4 UNIX操作系统中运行C程序 .....	(11)
习 题 .....	(11)
<b>2 数据类型与表达式</b> .....	(12)
2.1 C的数据类型 .....	(12)
2.1.1 整型 .....	(12)
2.1.2 浮点型 .....	(14)
2.1.3 字符型 .....	(15)
2.1.4 逻辑类型 .....	(17)
2.2 标准函数 .....	(18)
2.3 数据的定义方法 .....	(19)
2.3.1 常量定义 .....	(19)
2.3.2 变量定义 .....	(19)
2.4 表达式 .....	(20)
2.4.1 表达式的组成 .....	(20)
2.4.2 表达式的书写 .....	(21)
2.4.3 表达式的分类 .....	(21)
2.4.4 表达式的计算 .....	(22)
2.4.5 数据类型转换 .....	(22)
2.4.6 算术表达式 .....	(24)
2.4.7 赋值表达式 .....	(25)
2.4.8 逻辑表达式 .....	(28)

2.4.9 位运算表达式 .....	(28)
2.4.10 其他表达式 .....	(30)
习 题 .....	(32)

### 3 C 程序设计初步 .....

3.1 程序和程序设计 .....	(34)
3.1.1 程序和程序设计的概念 .....	(34)
3.1.2 评价一个程序系统的标准 .....	(35)
3.1.3 用先进的程序设计理论指导程序设计 .....	(37)
3.2 程序设计风格 .....	(38)
3.3 结构化程序设计方法 .....	(40)
3.3.1 限制使用 GOTO 语句 .....	(41)
3.3.2 结构化程序设计的三种基本结构 .....	(41)
3.3.3 结构化流程图 .....	(43)
3.3.4 自顶向下逐步求精 .....	(44)
3.4 说明语句 .....	(45)
3.5 赋值语句 .....	(46)
3.6 数据输入函数 .....	(47)
3.6.1 getchar( ) 函数 .....	(47)
3.6.2 scanf( ) 函数 .....	(47)
3.7 数据输出函数 .....	(49)
3.7.1 putchar( ) 函数 .....	(49)
3.7.2 printf( ) 函数 .....	(51)
3.8 简单的程序设计举例 .....	(55)
习 题 .....	(58)

### 4 分支结构程序 .....

4.1 条件语句 .....	(61)
4.1.1 简单的条件语句 .....	(61)
4.1.2 if-else 语句 .....	(62)
4.1.3 else-if 形式 .....	(64)
4.1.4 if 语句的嵌套 .....	(65)
4.1.5 条件运算符 .....	(67)
4.2 switch 语句 .....	(69)
4.3 复合语句和程序举例 .....	(71)
习 题 .....	(76)

### 5 循环控制结构 .....

5.1	while 语句	(78)
5.2	do-while 语句	(81)
5.3	for 语句	(82)
5.4	循环的嵌套	(86)
5.5	goto 语句 continue 语句 break 语句	(88)
5.5.1	goto 语句	(88)
5.5.2	continue 语句	(89)
5.5.3	break 语句	(90)
5.6	程序举例	(91)
	习 题	(98)

## 6 数组及其应用 (101)

6.1	数组概述	(101)
6.1.1	数组的概念	(101)
6.1.2	数组的分类	(101)
6.1.3	数组在内存中的存放	(101)
6.2	一维数组	(102)
6.2.1	一维数组的定义与引用	(102)
6.2.2	一维数组的初始化	(103)
6.2.3	一维数组程序举例	(104)
6.3	二维数组	(106)
6.3.1	二维数组的定义与引用	(106)
6.3.2	二维数组的初始化	(107)
6.3.3	二维数组程序举例	(108)
6.4	字符数组	(110)
6.4.1	字符数组的定义	(110)
6.4.2	字符数组的初始化	(111)
6.4.3	字符数组的输入输出	(114)
6.4.4	字符串处理函数	(115)
6.4.5	字符数组应用举例	(118)
	习 题	(120)

## 7 编译预处理 (121)

7.1	宏定义	(121)
7.1.1	不带参数的宏定义	(121)
7.1.2	带参数的宏定义	(124)
7.2	文件包含	(125)
7.3	条件编译	(125)
	习 题	(129)

<b>8</b>	<b>函数与变量</b> .....	(132)
8.1	函数概述 .....	(132)
8.1.1	C函数间的调用关系 .....	(132)
8.1.2	C函数的分类 .....	(133)
8.1.3	C语言源程序的编译、连接 .....	(134)
8.2	库函数 .....	(134)
8.2.1	库函数概述 .....	(134)
8.2.2	库函数的分类 .....	(134)
8.2.3	库函数的调用方法 .....	(134)
8.3	函数的定义 .....	(135)
8.3.1	函数的定义 .....	(135)
8.3.2	空函数 .....	(136)
8.4	函数参数与函数的返回值 .....	(137)
8.4.1	形式参数和实际参数 .....	(137)
8.4.2	函数的返回值 .....	(138)
8.5	函数的调用 .....	(141)
8.5.1	函数调用的过程 .....	(141)
8.5.2	函数调用的一般形式 .....	(141)
8.5.3	函数调用的方式 .....	(142)
8.5.4	数据复制方式与地址传送方式传递数据 .....	(142)
8.5.5	对被调用函数的说明 .....	(145)
8.6	函数的嵌套与递归 .....	(147)
8.6.1	函数的嵌套调用 .....	(147)
8.6.2	函数的递归调用 .....	(150)
8.7	局部变量与全局变量 .....	(153)
8.7.1	局部变量 .....	(153)
8.7.2	全局变量 .....	(154)
8.8	变量的存储类型 .....	(156)
8.8.1	存储器类型与变量的生存期 .....	(156)
8.8.2	变量的存储类型 .....	(156)
8.8.3	变量存储类型小结 .....	(161)
	习 题 .....	(161)
<b>9</b>	<b>指针</b> .....	(162)
9.1	指针的概念与其变量的定义 .....	(162)
9.1.1	指针的概念 .....	(162)
9.1.2	指针变量的定义 .....	(163)
9.1.3	指针的运算 .....	(164)

9.2 指针与数组 .....	(166)
9.2.1 指向一维数组的指针变量 .....	(166)
9.2.2 数组名作函数参数 .....	(169)
9.2.3 指向二维数组的指针变量 .....	(171)
9.2.4 指向由 n 个整数组成一维数组的指针变量 .....	(174)
9.3 指针与字符串 .....	(175)
9.3.1 字符串指针的表示 .....	(175)
9.3.2 字符串(指针)作函数参数 .....	(177)
9.3.3 字符指针变量与字符数组 .....	(179)
9.4 指针与函数 .....	(179)
9.4.1 指向函数的指针变量 .....	(179)
9.4.2 指针的函数 .....	(182)
9.5 指针的其他应用 .....	(183)
9.5.1 指针数组 .....	(183)
9.5.2 指向指针的指针 .....	(184)
9.5.3 数组作 main 函数的形参 .....	(185)
9.6 程序设计举例 .....	(186)
习 题 .....	(190)
<b>10 结构体 共用体 链表 .....</b>	<b>(193)</b>
10.1 结构体 .....	(193)
10.1.1 结构体类型的定义 .....	(193)
10.1.2 结构体变量的定义 .....	(194)
10.1.3 结构体变量的引用 .....	(196)
10.1.4 结构体变量的初始化 .....	(196)
10.1.5 结构体作函数参数 .....	(198)
10.1.6 嵌套结构体 .....	(199)
10.2 结构体数组 .....	(200)
10.3 共用体类型 .....	(202)
10.3.1 共用体类型的定义 .....	(202)
10.3.2 共用体变量的定义 .....	(202)
10.3.3 共用体变量的引用 .....	(204)
10.4 枚举类型 .....	(205)
10.4.1 枚举类型的定义 .....	(205)
10.4.2 枚举类型变量的定义 .....	(206)
10.5 链表 .....	(208)
10.5.1 单链表的数据描述 .....	(208)
10.5.2 单链表的建立 .....	(209)
10.5.3 单链表的基本操作 .....	(211)
习 题 .....	(213)

<b>11 文件</b> .....	(215)
11.1 概述 .....	(215)
11.2 文件指针 .....	(215)
11.3 文件的打开与关闭 .....	(215)
11.3.1 文件的打开( <code>fopen</code> 函数) .....	(216)
11.3.2 文件关闭函数( <code>fclose</code> 函数) .....	(218)
11.4 文件的读写 .....	(218)
11.4.1 字符读写函数 <code>fgetc</code> 和 <code>fputc</code> .....	(218)
11.4.2 数据块读写函数 <code>fread</code> 和 <code>fwrite</code> .....	(221)
11.4.3 格式化读写函数 <code>fscanf</code> 和 <code>fprintf</code> .....	(223)
11.4.4 字符串读写函数 <code>fgets</code> 和 <code>fputs</code> .....	(223)
11.5 文件的随机读写 .....	(225)
11.5.1 文件定位 .....	(225)
11.5.2 文件的随机读写 .....	(226)
11.5.3 文件检测函数 .....	(227)
习 题 .....	(227)

<b>12 C++ 语言概述</b> .....	(229)
12.1 面向对象程序设计方法 .....	(229)
12.1.1 什么是面向对象的程序设计方法 .....	(229)
12.2 C++ 与 C 的关系 .....	(229)
12.2.1 C++ 语言的起源 .....	(229)
12.2.2 C++ 语言与 C 语言的联系 .....	(230)
12.2.3 C++ 语言与 C 语言的主要区别 .....	(230)
12.2.4 C++ 语言与 C 语言的细小区别 .....	(230)
12.3 C++ 语言上机过程 .....	(233)
习 题 .....	(234)

<b>13 类与对象</b> .....	(235)
13.1 类的定义和类的实例 .....	(235)
13.1.1 类的声明 .....	(235)
13.1.2 类的成员 .....	(236)
13.1.3 对象 .....	(239)
13.2 构造函数与析构函数 .....	(240)
13.2.1 初始化列表 .....	(240)
13.2.2 构造函数 .....	(241)
13.2.3 拷贝构造函数 .....	(243)
13.2.4 析构函数 .....	(245)

13.3 类的聚合 .....	(246)
13.3.1 类的聚合 .....	(246)
13.3.2 类的对象成员初始化 .....	(247)
习 题 .....	(249)
<b>14 继承与派生 .....</b>	<b>(250)</b>
14.1 C++ 的继承和派生机制 .....	(250)
14.1.1 派生类的定义 .....	(250)
14.1.2 派生类生成过程 .....	(251)
14.1.3 派生方式 .....	(252)
14.2 派生类中构造函数和析构函数 .....	(257)
14.2.1 派生类的构造函数 .....	(257)
习 题 .....	(259)
附 录 .....	(260)
参 考 文 献 .....	(271)



程序设计语言是程序设计人员与计算机进行对话的语言,它遵循一定的规则和形式,构成各种应用程序。

## 1.1 C 语言的由来与发展

C 语言诞生于 1972~1973 年间美国贝尔实验室,由 Denis. M. Ritchie 创建。

C 语言的产生基于两个方面的需要。一是为满足 UNIX 操作系统开发的需要,UNIX 操作系统是一个通用的、开放式的计算机管理系统;二是为拉近高级语言与硬件之间距离的需要。

C 语言集高级语言与汇编语言的优点于一身。它面对实际应用的需要而产生和发展,直至今日仍不改初衷。

对 C 语言追本求源,可追溯到 ALGOL 语言。1960 年出现的 ALGOL 语言是一种面向问题的高级语言,因其远离硬件,不适于开发系统软件。1963 年,英国剑桥大学推出 CPL 语言,CPL 语言比 ALGOL 语言接近硬件一些,但规模较大,难以实现。1969 年,剑桥大学的 M. Richards 对 CPL 语言进行简化,推出 BCPL 语言(Basic Combined Programming Language)。1970 年,贝尔实验室的 K. Thompson 为 DEC 公司 PDP-7 计算机上运行的早期 UNIX 操作系统设计了一种类 BCPL 语言,称为 B 语言。B 语言规模小,接近硬件,1971 年在 PDP-11 计算机上实现,并写了 UNIX 操作系统。BCPL 语言和 B 语言都是无类型语言,过于简单,功能有限。1960~1971 年是 C 语言的萌芽阶段。

1972~1973 年间,贝尔实验室的 D. M. Ritchie 在保留 B 语言优点的基础上,创建了 C 语言。1973 年,UNIX 操作系统被用 C 语言改写,称为 UNIX 第 5 版。最初的 C 语言只是一种 UNIX 操作系统的工作语言,依附于 UNIX 系统,主要在贝尔实验室内部使用。UNIX 以后的第 6 版、第 7 版、SYSTEM III 和 SYSTEM V 都是在第 5 版的基础上发展起来的,C 语言也作了多次改进。到 1975 年,UNIX 第 6 版的公布,使 C 语言受到人们的普遍关注。

UNIX 操作系统的广泛使用,促进了 C 语言的迅速发展及普及,同时,C 语言的发展与普及也促进了 UNIX 操作系统的推广。1978 年,出现了独立于 UNIX 和 PDP 计算机的 C 语言,从而使 C 语言被迅速移植到大中小型计算机与微型机上。当年,B. W. Kernighan 和 D. M. Ritchie 以 UNIX 第 7 版的 C 编译程序为基础,出版了影响深远的名著《C 程序设计语言》。

C++ 语言是 C 语言发展的新阶段,是一种更好的 C 语言,是应用广泛的面向对象的程序设计语言。

目前,C 语言已经发展成为迄今为止最流行的计算机程序设计语言。

K&R 的名著《C 程序设计语言》出版不久,各种 C 语言编译系统纷纷出现。这本书的存在,使得绝大多数 C 语言保持着高度的兼容性,K&R 给出了 C 语言的经典定义,使其成为 C 语言的世界标准。

然而,C 语言的广泛应用在各开发团体之间出现了问题。各机构推出了自己的 C 语言



与版本,某些执行过程的微小差别不时引起 C 程序之间的不兼容。美国国家标准协会(ANSI)从 1983 年开始,经过长达 5 年的努力,制定了 C 语言的新标准——ANSIC,现在提及 C 语言的标准就是指该新标准。ANSIC 比原标准 C 有很大的发展,解决了经典定义中的二义性,给出了 C 语言的新特点。K&R 也以新标准改写了他们的经典著作。任何 C 程序都必须遵循 ANSIC 标准,本教材的主体也以 ANSIC 作为基础。

目前,微机上广泛使用的 C 语言为 Borland 公司的产品 TURBO C 和 Microsoft 公司的产品 MSC。TURBO C 2.0 完全支持新标准 C,提供集成开发操作环境,本教材以此作为实践环境。C++ 程序设计语言主要版本有 Borland C++ 4.0、Borland C++ 5.0、Visual C++ 5.0、Visual C++ 6.0。

## 1.2 C 语言的特点

C 语言是一种通用、灵活、结构化、标准化、使用普遍的编程语言,能完成用户想得到的任何任务,特别适合进行系统程序设计和对硬件进行操作的场合。C 语言本身不对程序员施加过多限制,是一种专业程序员优先选择的语言。

C 语言的主要特点如下:

(1) C 语言简洁、紧凑,压缩了一切不必要的成份。ANSIC 有 32 个关键字,TURBO C 有 58 个关键字、9 种控制语句,书写形式自由。

(2) C 语言运算丰富,将括号、赋值、强制类型转换、取变量地址等都以运算实现。ANSIC 提供 34 种运算符,TURBO C 提供了 44 种运算符,灵活使用这些运算符可以实现其他高级语言难以实现的操作。C 语言的表达式简练、多样、灵活、实用,加上分号可以构成语句。

(3) C 语言数据类型丰富,具有现代语言的各种数据类型;用户能扩充数据类型,能够实现各种复杂的数据结构,完成各种问题的数据描述。尤其是 C 语言的指针类型,非常有特色,可指向各种数据,完成各种数据的高效处理。C 语言对数据不但作类型上的描述,还提供存储属性选择。

(4) C 语言是一种结构化程序设计语言,具有结构化语言所要求的三种基本结构,即顺序、分支、循环结构。C 语言用函数作为结构化程序设计的实现工具,实现程序的模块化。C 程序由若干程序文件组成,一个程序文件由若干函数构成。

(5) C 语言允许直接访问物理地址,能进行位运算,能实现汇编语言的大部分功能,能直接对硬件进行操作。C 语言的双重性使它被称为“高级语言中的低级语言”,或称为“中级语言”,表示将低级语言和高级语言的特点集于一身。

人们通常将语言分类如下:

