



普通高等院校“十三五”精品规划教材

PUTONG GAODENG YUANXIAO “SHISANWU” JINGPIN GUIHUA JIAOCAI

C语言 程序设计

C YUYAN
CHENGXU SHEJI

主 编 ● 李 涛 王观玉 任廷艳
副主编 ● 胡 洋
主 审 ● 石云辉

普通高等院校“十三五”精品规划教材

C 语言程序设计

主 编 李 涛 王观玉 任廷艳

副主编 胡 洋

主 审 石云辉

西南交通大学出版社

· 成 都 ·

图书在版编目 (C I P) 数据
C 语言程序设计 / 李涛, 王观玉, 任廷艳主编. —
成都: 西南交通大学出版社, 2018.8
ISBN 978-7-5643-6363-5

I. ①C... II. ①李... ②王... ③任... III. ①C 语言 -
程序设计 - 高等学校 - 教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2018) 第 189946 号

C 语言程序设计

主编 李 涛 王观玉 任廷艳

责任编辑 黄淑文

封面设计 何东琳设计工作室

出版发行 西南交通大学出版社
(四川省成都市二环路北一段 111 号
西南交通大学创新大厦 21 楼)

邮政编码 610031

发行部电话 028-87600564 028-87600533

官网 <http://www.xnjdcbs.com>

印刷 成都蓉军广告印务有限责任公司

成品尺寸 185 mm× 260 mm

印张 14.75

字数 370 千

版次 2018 年 8 月第 1 版

印次 2018 年 8 月第 1 次

定价 39.00 元

书号 ISBN 978-7-5643-6363-5

课件咨询电话: 028-87600533

图书如有印装质量问题 本社负责退换

版权所有 盗版必究 举报电话: 028-87600562

前 言

C 语言是一种短小精悍的计算机高级程序设计语言，它是结构化程序设计语言。C 语言应用广泛，结构简单、数据类型丰富、表达能力强、使用灵活方便。C 语言既有高级语言的优点，又具有低级语言的许多特点。用 C 语言编写的程序，具有速度快、效率高、代码紧凑、可移植性好等优点。利用 C 语言，可编制各种系统软件和应用软件。

本书是作者根据多年 C 语言教学经验积累，面向计算机及其他理工类专业的本科学生编写的，在编写过程中努力做到结构紧凑、概念准确、叙述流畅、重点突出、例题丰富、实用性强。本书各章节引入了大量的实例来说明相关的内容，力求让读者尽快上手。本书在编写上有以下的特点：

(1) 精选理论，强化实践，突出技能。

(2) 结构清晰、新颖。

(3) 在内容的组织上考虑了 C 语言的特点和读者群的特点，力求用条理、简洁、通俗的语言，准确清楚地表达概念。

(4) 本着循序渐进的原则，对 C 语言的基本概念和语法规则做了系统讲解。全书的内容前导后续衔接、过渡自然，知识点的组织思路清晰，安排恰当，尽量避免后续知识提前出现给读者造成不必要的障碍。这是作者多年教学经验的总结，也是本书的一大特色。

(5) 本书提供了大量典型的例题，通过对典型例题的分析和学习，使读者加深对知识的理解和掌握。

(6) 本书例题程序和实验参考程序均在 VC 6.0++环境下调试通过。由于篇幅有限，书中的程序只给出了一种参考程序，读者在学习过程中可以举一反三。

(7) 本书的例题程序书写规范，读者通过学习和效仿，可以逐步养成良好的编程习惯。

要学好 C 语言，除了要勤于思考外，还要加强实践才能收到更好的效果。我们在多年的教学实践过程中发现：不少人在学习 C 语言时，感到入门难，掌握起来更难，对很多问题的理解支离破碎。所以，建议初学者在学习 C 语言时一定要做到“三多”，多看程序、多写程序、多上机调试程序，只有这样才能尽快地掌握和运用 C 语言去解决实际问题。

本书的目标是详细介绍 C 语言结构化程序设计的基本理论、基本概念、方法和规范，以培养学生的计算机应用编程能力。作者在撰写本书的过程中，既参考了《The C Programming Language (Second Edition)》(Brain W 编著)和《最新 C 语言程序设计》(刘正林等编著)及《C 语言程序设计》(谭浩强编著)等国内外多种教材，也融入了作者多年从事教学和科研的实践经验和体会，同时还吸收了同行专家学者们的意见和建议。

本书共 11 章，其中第 1 章、第 2 章、第 3 章由王观玉教授编写；第 4 章、第 5 章、第 8 章、第 10 章由任廷艳老师编写；第 6 章、第 7 章、第 9 章和第 11 章由李涛老师编写。

本书的出版得到了黔南民族师范学院计算机与信息学院各位领导及同事的关心和支持，得到了贵州省本科高校一流课程建设项目“基于应用型的程序设计课程群”的支持。作者在此表示衷心感谢！

由于作者水平有限，书中疏漏或错误之处在所难免，恳请广大读者批评指正。

作 者

2018 年 6 月

目 录

第 1 章 C 语言概述	1
1.1 程序设计语言简述	1
1.2 C 语言简介	3
1.3 C 语言程序的基本结构	4
1.4 C 语言程序的运行	5
本章小结	6
习 题	6
第 2 章 数据类型	8
2.1 标识符	8
2.2 关键字	8
2.3 数据类型	9
2.4 常 量	10
2.5 变 量	14
本章小结	21
习 题	21
第 3 章 运算符与表达式	23
3.1 表达式与运算符	23
3.2 算术运算符与算术表达式	24
3.3 赋值运算符与赋值表达式	27
3.4 关系运算符与关系表达式	28
3.5 逻辑运算符与逻辑表达式	29
3.6 逗号运算符与逗号表达式	31
3.7 运算符的优先级和结合性	32
本章小结	34
习 题	34
第 4 章 常用的数据输入/输出函数	36
4.1 语 句	36

4.2	数据的输入/输出	37
4.3	字符数据输入/输出	38
4.4	格式输入/输出函数	39
4.5	顺序结构程序应用	45
	本章小结	46
	习 题	46
第 5 章	选择结构程序设计	48
5.1	选择结构	48
5.2	if 语句	48
5.3	switch 语句	54
5.4	选择结构程序应用	57
	本章小结	58
	习 题	59
第 6 章	循环控制	61
6.1	while 语句	61
6.2	do... while 语句	64
6.3	for 语句	66
6.4	3 种循环语句的比较	69
6.5	循环嵌套	70
6.6	转移语句	72
6.7	循环结构程序应用举例	75
	本章小结	80
	习 题	80
第 7 章	数 组	84
7.1	一维数组	84
7.2	二维数组	91
7.3	字符数组	95
7.4	多维数组	104
7.5	数组的排序算法	105
	本章小结	109
	习 题	109
第 8 章	函 数	112
8.1	函数概述	112

8.2	函数的定义	113
8.3	函数的调用	114
8.4	函数的嵌套调用和递归调用	120
8.5	函数与数组	123
8.6	变量的作用域和生存期	126
8.7	内部函数和外部函数	135
	本章小结	137
	习 题	137
第 9 章	指 针	140
9.1	指针概述	140
9.2	数组与指针	147
9.3	字符串与指针	155
9.4	返回指针值的函数	160
9.5	函数的指针和指向函数的指针变量	161
9.6	指针数组和指向指针的指针	163
	本章小结	166
	习 题	167
第 10 章	结构体和共用体	171
10.1	结构体	171
10.2	结构体数组	177
10.3	结构体指针	179
10.4	结构体与函数	183
10.5	链 表	186
10.6	共用体	190
10.7	枚举类型	191
10.8	类型定义	193
	本章小结	195
	习 题	196
第 11 章	文 件	198
11.1	文件概述	198
11.2	文件的打开与关闭	200
11.3	文件的读和写	202
11.4	文件的定位	213
11.5	文件检测函数	217

本章小结	217
习 题	218
附录一 ASCII 表	220
附录二 C 语言中的关键字	221
附录三 运算符的优先级和结合方向	222
附录四 C 库函数	224
参考文献	228

第 1 章 C 语言概述

本章主要内容：

- 程序设计语言的分类；
- C 语言的特点；
- C 语言源程序的执行过程。

1.1 程序设计语言简述

1.1.1 程序与程序设计

计算机能够完成的所有工作都是由人们编写的程序来控制的。程序（program）是指一组指示计算机或其他具有信息处理能力的装置去执行动作或做出判断的指令，通常用某种程序设计语言编写，运行于某种目标体系结构上。程序设计（program design）是给出解决特定问题程序的过程，是软件构造活动中的重要组成部分。程序设计往往以某种程序设计语言为工具，给出这种语言下的程序。程序设计过程应当包括分析、设计、编码、测试、排错等不同阶段。

计算机每执行一个操作都是按照事先指定的内容和步骤进行的，每个步骤的具体内容由计算机能够理解的指令或语句来描述，指令或语句告诉计算机“做什么”和“怎么做”。例如，打开“Word”编写文档，可以分为以下步骤：

- （1）双击“word”图标；
- （2）输入文字；
- （3）保存文件；
- （4）退出“word”。

程序设计的一般过程如图 1.1 所示。

- （1）分析问题：对于要处理问题的条件和目标进行认真分析，选择解决问题的方法。
- （2）设计算法：在分析问题的基础上，设计出解决问题的方法和具体步骤。
- （3）编写程序：选择程序设计语言对所设计的算法进行实现，并对源程序进行编译、链接，生成可执行程序。
- （4）运行测试：运行生成的可执行程序，并对结果进行测试分析，排查纠正程序中的错误。

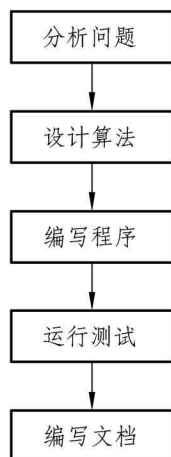


图 1.1 程序设计的一般过程

(5) 编写文档：提供给用户使用的程序，需要向用户提供程序说明书，指出程序功能、运行环境、安装启动以及注意事项等。

1.1.2 程序设计语言

程序设计语言是人和计算机进行交流时采用的语言，随着计算机的发展，程序设计语言已经由低级语言发展到高级语言。

1. 第一代语言——机器语言 (Machine Language)

机器语言是用二进制代码表示的、计算机能直接识别和执行的一种机器指令的集合。它是计算机的设计者通过计算机的硬件结构赋予计算机的操作功能。机器语言具有直接执行速度快的特点。

用机器语言编写程序，编程人员要首先熟记所用计算机的全部指令代码和代码的含义，在编程时，程序员需要自行处理每条指令和每一数据的存储分配和输入输出，记住编程过程中每步所使用的工作单元所处的状态，编写程序花费的时间较长。为了克服机器语言可读性差的缺点，出现了汇编语言。

2. 第二代语言——汇编语言 (Assembly Language)

汇编语言是面向机器的程序设计语言，在汇编语言中，用助记符代替操作码，用地址符号或标号代替地址码。使用汇编语言编写的程序，机器不能直接识别，要由一种程序将汇编语言翻译成机器语言，这种起翻译作用的程序叫汇编程序，汇编程序是系统软件中语言处理系统软件。汇编程序把汇编语言翻译成机器语言的过程称为汇编。

汇编语言便于识别和记忆，执行效率也比较高，适合编写一些对速度和代码长度要求高的程序和直接控制硬件的程序。但是，汇编语言和机器语言一样，仍然是面向机器的语言，使用起来还是比较烦琐，通用性较差。于是，出现了高级语言。

机器语言和汇编语言属于低级语言。

3. 第三代语言——高级语言（High-level Language）

高级语言是在低级语言的基础上，采用接近于人类自然语言的单词和符号来表示程序，使编程变得更加简单且写出的程序可读性强。高级语言与计算机的硬件结构及指令系统无关，它提供了丰富的数据类型和运算符，可以方便地表示数据的运算和程序的控制结构，能更好地描述各种算法，而且容易学习掌握。

但高级语言编译生成的程序代码一般比用汇编程序语言设计的程序代码要长，执行的速度也慢。

流行的高级语言有 C，C++，Java，C#，Pascal，Python，Lisp，Prolog，FoxPro 等，这些语言的语法、命令格式都不相同。

1.1.3 程序设计方法

目前常用的程序设计方法有面向过程的方法和面向对象的方法。

（1）面向过程的方法：在软件设计和实现过程中，分析出解决问题所需要的步骤，然后用函数把这些步骤一步一步实现，使用的时候一个一个依次调用就可以了。面向过程最重要的是模块化的思想方法。

（2）面向对象的方法：主要是把事物对象化，包含了面向过程的特征，是一种更高级的解决问题的思路，从现实世界中的客观对象出发，尽量运用人类的自然思维方式（如抽象和分类）来构造软件系统，摒弃了数据和功能独立的特点，定义了一组交互的对象。

1.2 C 语言简介

C 语言是国际上广泛流行的面向过程的高级语言，它适合作为系统描述语言，可以用来编写系统软件和应用软件。

C 语言之所以命名为 C，是因为 C 语言源自 Ken Thompson 发明的 B 语言，而 B 语言则源自 BCPL 语言。

1967 年，剑桥大学的 Martin Richards 对 CPL 语言进行了简化，于是产生了 BCPL（Basic Combined Programming Language）语言。

1970 年，美国贝尔实验室的 Ken Thompson 以 BCPL 语言为基础，设计出很简单且很接近硬件的 B 语言（取 BCPL 的首字母），并且他用 B 语言写了第一个 UNIX 操作系统。

1972 年，美国贝尔实验室的 D.M.Ritchie 在 B 语言的基础上最终设计出了一种新的语言，他取了 BCPL 的第二字母作为这种语言的名字，这就是 C 语言。

1973 年，C 语言的主体完成。Thompson 和 Ritchie 用它完全重写了 UNIX。随着 UNIX 的

发展，C 语言自身也在不断地完善。直到今天，各种版本的 UNIX 内核和周边工具仍然使用 C 语言作为最主要的开发语言。

1989 年，ANSI 发布了第一个完整的 C 语言标准——ANSI X3.159—1989，简称“C89”。

1999 年，在做了一些必要的修正和完善后，ISO 发布了新的 C 语言标准，命名为 ISO/IEC 9899：1999，简称“C99”。

2011 年 12 月 8 日，ISO 又正式发布了新的标准，称为 ISO/IEC 9899：2011，简称为“C11”。

C 语言是应用最广泛的语言之一，它具有以下特点。

(1) 简洁紧凑、灵活方便。

C 语言一共只有 32 个关键字，9 种控制语句，程序书写形式自由，区分大小写。C 语言把高级语言的基本结构和语句与低级语言的实用性结合起来。

(2) 运算符丰富。

C 语言的运算符包含的范围很广泛，共有 34 种运算符。C 语言把括号、赋值、强制类型转换等都作为运算符处理，从而使 C 语言的运算类型极其丰富，表达式类型多样化。灵活使用各种运算符可以实现在其他高级语言中难以实现的运算。

(3) 数据类型丰富。

C 语言的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等，能用来实现各种复杂的数据结构的运算。并且 C 语言引入了指针概念，使程序效率更高。

(4) 表达方式灵活实用。

C 语言提供了多种运算符和表达式值的方法，对问题的表达可通过多种途径获得，其程序设计更主动、灵活。它语法限制不太严格，程序设计自由度大，如对整型量与字符型数据及逻辑型数据可以通用等。

(5) 允许直接访问物理地址，对硬件进行操作。

C 语言允许直接访问物理地址，可以直接对硬件进行操作，因此它既具有高级语言的功能，又具有低级语言的许多功能，能够像汇编语言一样对位、字节和地址进行操作，可用来编写系统软件。

(6) 生成目标代码质量高，程序执行效率高。

C 语言描述问题比汇编语言迅速，工作量小、可读性好，易于调试、修改和移植，而代码质量与汇编语言相当。C 语言一般只比汇编语言生成的目标代码效率低 10%~20%。

(7) 可移植性好。

C 语言在不同机器上的 C 编译程序，86%的代码是公共的，所以 C 语言的编译程序便于移植。在一个环境上用 C 语言编写的程序，不改动或稍加改动，就可以移植到另一个完全不同的环境中运行。

1.3 C 语言程序的基本结构

先看一个简单的 C 语言程序例子。

【例 1.1】在计算机屏幕上输出信息“ hello C!”。

```
#include<stdio.h>           //预处理
int main ( )                 //主函数
{
    printf ( "hello C!\n" );
    return 0 ;
}
```

运行结果：



```
hello C!
Press any key to continue
```

程序讲解：

以上 C 程序的基本结构分为两个部分：预处理命令和主函数。

(1) 以“ #” 开头的是预处理命令，#include 为包含命令（行尾没有分号），表示把标准输入输出头文件（standard input & output header file，stdio.h）包含到本程序中。C 语言要求，使用系统给定的函数，如程序中的库函数 printf 函数时，要把含有该函数信息的头文件包含进来。

(2) 程序中的 main 函数称为主函数。一个 C 程序必须包含一个 main 函数，也只能有一个 main 函数，程序从 main 开始执行，在 main 函数结束。main 前的 int 代表需要返回一个整数值（C99 标准要求），有的书中用 void，表示函数没有返回值。

注意：C 语言程序的基本结构具有以下特点。

- 一个 C 语言源程序可以由一个或多个源文件（扩展名为.c）组成。
- 每个源文件可以由一个或多个函数组成。
- 一个源程序不论由多少个文件组成，都有一个且只能有一个 main 函数，即主函数。
- 源程序中可以有预处理命令（include 命令仅为其中的一种），预处理命令通常应放在源文件或源程序的最前面。
- 每个函数的定义包含两部分，函数头和函数体。函数的头部包括函数名、返回类型和用圆括号“（）”括起来的形参说明。函数体是用花括号“{ }”括起来的函数执行部分。
- 函数体中包含各种语句，语句是程序的基本执行单位，每一条语句都以分号“；”结束。

1.4 C 语言程序的运行

在实际编程中，必须根据问题的要求，编写出相应的程序。用高级语言编写的程序称为源程序（source program）。但是计算机只能识别机器语言，在执行高级语言源程序时，必须通过编译程序，把源程序翻译成二进制形式的目标程序（object program），然后再将目标程序与系统的函数库以及其他目标程序连接起来，形成可执行的目标程序。以上过程如图 1.2 所示。

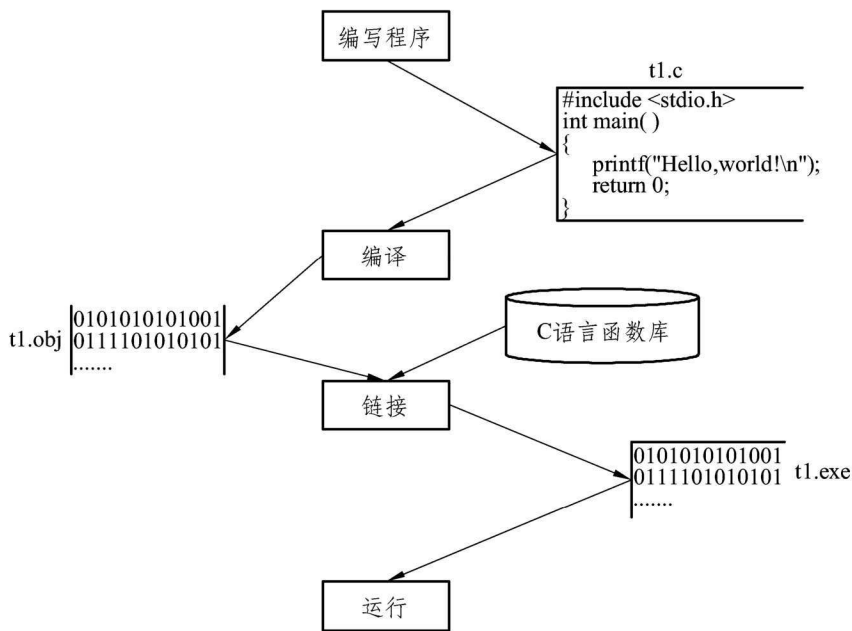


图 1.2 C 语言程序的运行过程

本章小结

(1) 计算机是由程序控制的，要使计算机按照特定步骤去工作，必须用计算机语言编写程序。

(2) 计算机语言分为低级语言和高级语言。机器语言和汇编语言属于低级语言。低级语言依赖于硬件，通用性差。高级语言接近人类自然语言，通用性强。

(3) C 语言程序的基本组成单位是函数，一个 C 语言程序中必须有一个 main 函数。main 函数位置任意。程序从 main 函数开始执行。在函数中包含若干语句，语句以分号结束。

(4) 运行 C 程序的四个步骤：① 编写源程序；② 对源程序进行编译，得到目标程序；③ 将目标程序与库函数链接，得到可执行的目标程序；④ 运行可执行的目标程序。

习 题

一、单项选择题

1. 一个 C 语言程序总是从_____开始执行。
 - A. 书写顺序的第一个函数
 - B. 第一条执行语句
 - C. 主函数 main
 - D. 不确定
2. C 语言的基本组成单位是_____。

- A. 标识符 B. 函数 C. 表达式 D. 语句
3. 以下叙述正确的是_____。
- A. 在对一个 C 程序进行编译的过程中，可发现注释中的拼写错误
- B. 程序中只能有一个 main 函数
- C. C 语言的每行中只能写一条语句
- D. C 语言本身有输入输出的语句

二、思考题

1. 简述程序设计语言的分类，列举几种当前流行的程序设计语言。
2. 简述 C 程序的基本组成结构。

第 2 章 数据类型

本章主要内容：

- 掌握数据类型的概念，了解 C 语言基本数据类型；
- 掌握常量的概念，掌握各种类型常量的特点及表示形式；
- 掌握变量的概念，掌握不同类型变量的定义、赋值和使用方法。

2.1 标识符

在 C 语言中，各种名称都用标识符 (identifier) 来表示，如变量名、符号常量名、函数名、数组名、文件名、指针名等。C 语言的标识符有三类：关键字、预定义标识符和用户标识符。其中前两种由系统命名，已经预先定义好，只有用户标识符需要用户来命名。

注意：标识符的使用有如下规定：

- C 语言标识符由字母、数字和下划线组成，只能以字母、下划线开头。
- 在 C 语言中大写字母与小写字母表示不同的符号。一般约定，变量名用小写字母，符号常量名用大写字母。
- 用户定义的标识符不可与系统的关键字、预定义标识符同名。
- 为了提高程序的可读性，标识符在命名时尽量“见名知义”。即从标识符可知道该变量或函数的含意。例如，year，month，day，age，sex 等，不难从英文单词了解变量的含意。

下面是一些正确和错误标识符命名的实例。

正确形式：count、test23、high_balance

错误形式：2count、hi! there、high..balance

2.2 关键字

关键字 (keywords) 是一种具有特定含意的标识符。关键字又称保留字，在系统中已定义过，需要加以保留。不能用关键字作为用户标识符。C 语言中关键字分为三类。。

- (1) 类型说明符：用于定义、说明变量、函数或其他数据结构的类型，如 int、double 等。
- (2) 语句定义符：用于表示一个语句的功能，如 if、else 就是条件语句的语句定义符。