微视频版

高等学校计算机基础教育教材精选

C语言程序设计（第2版）

孙改平 王德志 主 编
吴 静 盛建瓴 王晓菊 乔 良 副主编
郭 红 主 审

清华大学出版社

高等学校计算机基础教育教材精选

C语言程序设计

（第2版）

孙改平 王德志 主 编
吴 静 盛建瓴 王晓菊 乔 良 副主编

清华大学出版社
北京

内 容 简 介

本书是按照普通高等院校大学计算机程序设计课程的培养目标和基本要求，结合全国计算机等级考试(二级)最新考试大纲，由多年从事计算机基础教学、具有教学经验的教师编写。全书共分10章，系统地介绍了程序设计概述、数据类型、顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、函数、指针、结构体和共用体、文件等内容。

本书内容丰富翔实、语言通俗易懂，通过一些有趣的案例激发读者的学习兴趣，通过一些实用案例讲解知识点，把一些繁杂的知识点分散到不同的示例中讲解并应用，通过几个典型案例贯穿整个知识体系。

本书适合作为高等院校C程序设计课程的教材，也可作为计算机各类培训班的教材或计算机及相关工作的科技人员、计算机爱好者及各类自学人员的参考书。

图书在版编目(CIP)数据

C语言程序设计/孙改平，王德志主编. —2版. —北京：清华大学出版社，2019
(高等学校计算机基础教育教材精选)
ISBN 978-7-302-52292-8

Ⅰ. ①C… Ⅱ. ①孙… ②王… 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(2019)第028655号

责任编辑：龙启铭
封面设计：何凤霞
责任校对：时翠兰
责任印制：杨　艳

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦A座　　**邮　　编**：100084
社 总 机：010-62770175　　**邮　　购**：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课件下载：http://www.tup.com.cn,010-62795954
印 刷 者：北京富博印刷有限公司
装 订 者：北京市密云县京文制本装订厂
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：24.5　　**字　　数**：568千字
版　　次：2016年3月第1版　2019年7月第2版　　**印　　次**：2019年7月第1次印刷
定　　价：45.00元

产品编号：077476-01

前言

本书第1版自2016年3月出版以来,先后经过多次印刷,在多所院校得到很好的应用,颇受广大师生的好评。本书在使用过程中,得到了众多读者的意见反馈,在此向他们表示感谢!本书是在第1版基础上,进一步完善教材内容和原有电子资源,新增加了扫码学习的视频资源,方便广大读者。

本书既注重概念,使读者建立起对程序设计和C语言的清晰理解,又引导学生学以致用,使学生在较短的时间内初步学会用C语言编写程序,具有初步的编程知识和能力。本书的讲解是不断地提出问题、解决问题,再进一步提出问题,并逐步解决问题的过程。使学生养成由简到繁、逐步求精的编程习惯。

全书共分10章,第1章介绍计算机与程序设计语言基础知识、C语言的发展和特点、C语言的应用等;第2章详细介绍C语言中常用的数据类型,如整型、实型和字符型等;第3章为顺序结构程序设计,介绍赋值运算符与赋值表达式、算术运算符与算术表达式、宏定义与宏替换等,重点介绍数据的格式化输入与输出;第4章为选择结构程序设计,介绍算法及其描述方法、关系运算符与关系表达式、逻辑运算符与逻辑表达式、条件运算符与条件表达式、单分支与双分支以及多分支选择结构;第5章为循环结构程序设计,详细介绍while循环、do-while循环和for循环三种循环结构语句应用,以及C语言中实现流程的转移控制语句;第6章为数组,介绍一维数组、二维数组、字符数组的定义、引用和初始化等;第7章为函数,介绍函数的概念、函数声明、函数定义、函数调用、数组作为函数参数、变量的作用域和存储类型等;第8章为指针,介绍指针的概念、指针变量的定义、指针与数组、指针与函数、指针的高级应用等;第9章为结构体和共用体,介绍结构体类型和结构体变量、结构体数组、结构体指针、链表、共用体等;第10章为文件,介绍文件的概念、分类、文件的打开与关闭、文件的读写操作等。

为了方便广大师生的教学和学习,本书还提供了配套的电子教案和有关的素材文件。

本书由孙改平、王德志主编,吴静、盛建瓴、王晓菊、乔良为副主编。第1、4章由王晓菊编写,第2、3章由盛建瓴编写,第5章由吴静编写,第6、7章由孙改平编写,第9章由乔良编写,第8、10章和附录由王德志编写,全书的视频资源由王德志老师录制,最后由孙改平、王德志进行统稿。

在本书的编写过程中得到了各级领导的关心和大力支持，C 语言程序设计课程的任课教师郭红、鞠宏军、朱冬梅、陈超、郭晓欣、万雪芬、刘明艳、吴晓丹等为本书提出了宝贵的意见和建议，在此一并表示感谢。

在本书的编写过程中，参考了国内外的相关研究成果和著作，部分已列入本书后面的参考文献，在此感谢涉及的所有专家和研究人员。

尽管作者做出了种种努力，付出了许多劳动，但由于水平有限，时间仓促，书中不妥或疏漏之处在所难免，恳请使用本书的广大同行和读者对本书提出宝贵意见，帮助我们不断地完善本书。

作　者

2019 年 5 月

目录

第1章 程序设计概述

本章主要介绍计算机程序设计语言的发展、分类、C 语言的发展和特点等相关概念，并通过几个简单的 C 程序实例，介绍 C 程序的基本结构、工作原理和操作环境。通过本章的学习，可以掌握 C 语言程序的结构，初步了解 C 程序设计的步骤及上机调试的简单方法。

1.1 计算机与程序设计语言

计算机不能完全自动进行所有的工作，更不是“万能”的。计算机的每一个操作都是根据人们事先制定的指令进行的。例如，要求计算机进行加法运算，必须事先编好指令，输入计算机，才能让计算机进行相应的操作。程序是一组计算机能识别和执行的指令序列。每一条指令让计算机执行特定的操作，一个特定的指令序列完成一定的功能。为了使计算机系统能实现各种功能，需要成千上万个程序，这些程序都是由计算机软件设计人员根据需要设计的，并且都是通过程序设计语言来编写的。

程序设计语言是人和计算机交换信息的工具，到目前为止，程序设计语言的发展历程主要包括机器语言、汇编语言和高级语言三大类，前两类依赖于计算机硬件，有时统称为低级语言，而高级语言与计算机硬件依赖关系较小。

1.1.1 机器语言

机器语言是计算机硬件系统能够直接识别的计算机语言，不需要翻译。机器语言中的每一条语句实际上是一条二进制数形式的指令代码，都由 0 和 1 组成。0 和 1 的数码组合不仅用来表述数据、符号，而且也用来表述计算机所进行的操作（如加、减、乘、除等）的命令。

例如，计算累加器 A＝8＋10 的机器语言程序如表 1.1 所示。

表 1.1 计算 A＝8＋10 的机器语言及注释

机器语言程序	注　　释
10110000 00001000	把 8 存放到累加器 A 中
00101100 00001010	将 10 与累加器中的 8 相加，结果存在 A 中
11110100	程序结束

对于不同的计算机硬件，其机器语言是不同的，因此，针对某一种计算机所编写的机器语言程序，多数不能在另一种计算机上运行。另外，用机器语言编写程序，工作量大、难于记忆、容易出错、调试修改麻烦，而且程序的直观性差，不容易移植，因此在初期只有极少数的计算机专业人员会编写计算机程序。机器语言的优点是编写的程序能直接在机器上运行，因此它的执行效率比较高，能充分发挥计算机的速度性能，同时用机器语言编写的程序占用的存储空间相对较小。

1.1.2 汇编语言

汇编语言是用助记符来代替机器指令的操作码，用地址符号代替操作数。由于这种符号化的做法，所以汇编语言也被称为符号语言。汇编语言要比机器语言直观，容易理解和记忆。例如用 ADD 表示加、SUB 表示减、JMP 表示跳转、MOV 表示数据的传送指令等。

例如，实现 8＋10 的汇编语言程序为：

```
MOV AX, 08H          ;将 8 送到寄存器 AX 中
MOV BX, 0AH          ;将 10 送到寄存器 BX 中
ADD BX, AX           ;将 AX 和 BX 中的数值相加，结果存在 BX 中
```

用汇编语言编写的程序称为汇编语言“源程序”，计算机能够直接识别的语言只有机器语言，因此汇编语言的源程序是不能在计算机上直接运行的，需要用“汇编程序”把它翻译成机器语言程序后方可运行。

汇编语言编写的程序比机器语言编写的程序易读、易检查、易修改，同时保持了机器语言执行速度快、占用存储空间少的优点。但是，汇编语言也是“面向机器”的语言，通用性和可移植性差。

1.1.3 高级语言

高级语言接近于自然语言和数学语言。高级语言允许用英文单词编写解题程序，所用的运算符、运算式与数学公式差不多。由于高级语言采用自然语汇，并使用与自然语言相近的语法体系，所以它的程序设计方法比较接近于人们的习惯，编写出的程序更容易阅读和理解。

高级语言是一类面向过程或面向对象的语言，它不依赖于具体的机器，独立于计算机硬件，通用性和可移植性都比较好。高级语言的特点是易学、易用、易维护，人们可以更有效、更方便地利用它编写各种用途的计算机程序。

例如，计算 8＋10，并把结果赋值给变量 c，在 C 语言中可将它表示成：

```
c= 8+10;
```

用某种高级语言编写的程序称为高级语言源程序，也简称为源程序。与汇编语言一样，要让计算机理解高级语言源程序的意图，必须先将源程序“翻译”成机器指令形式的程

序，然后再让计算机执行。不同的高级语言采用的翻译方式不同，但归纳起来有两种方法，即编译方式和解释方式。

（1）编译方式：把用高级语言编写的源程序翻译成目标程序的过程称为编译。完成编译工作的软件称为编译程序。源程序经过编译后，若无错误就会生成一个等价的目标程序，对目标程序再进行链接、装配后，便得到执行程序，最后运行执行程序即可得到程序的运行结果，其中的执行程序全部由机器指令组成，运行时不依附于源程序，运行速度快。但这种方式不够灵活，每次修改源程序后，必须重新编译、链接。目前使用的 FORTRAN、Pascal、C、C++ 、Visual C++ 、C# 、Ada 等高级语言都采用这种方式。

（2）解释方式：解释方式也是将高级语言转换为机器能够识别的语言，但与编译方式不同的是，解释方式是边扫描源程序、边进行翻译，然后执行，即解释一句，执行一句，不生成目标程序。完成解释工作的软件称为解释程序。这种方式运行速度慢，但执行中可以进行人机对话，随时改正源程序中的错误。BASIC、Java 等高级语言采用这种方式处理。编译程序与解释程序的区别如图 1.1 所示。

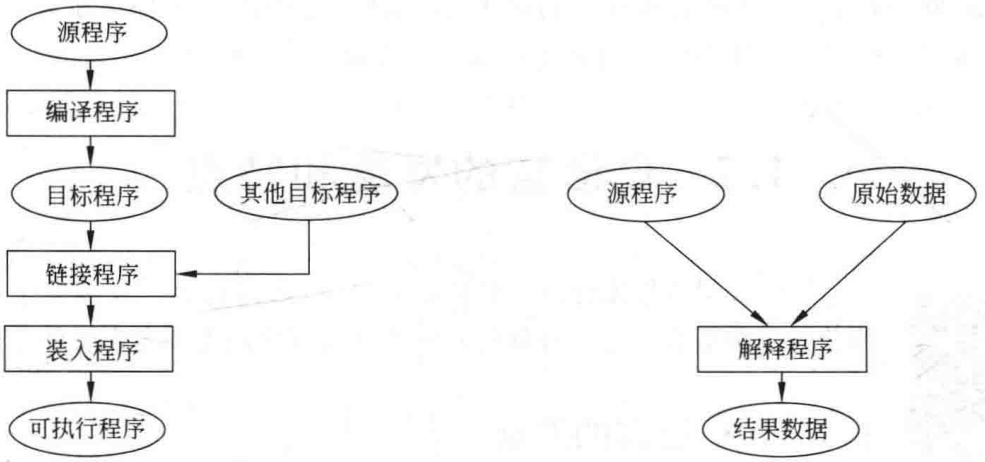

图 1.1　编译程序与解释程序的区别

这两种方式各有特点，其中编译方式可以节省计算机时间，而解释方式则可节省计算机存储空间。

高级语言经历了如下几个不同的发展阶段。

1. 非结构化的语言

初期的语言属于非结构化的语言，编程风格比较随意，只要符合语法规则即可，没有严格的规范要求，程序中的流程可以随意跳转。人们往往追求程序执行的效率而采用了许多“小技巧”，使程序变得难以阅读和维护。早期的 BASIC、FORTRAN 和 ALGOL 等都属于非结构化的语言。

2. 结构化语言

结构化语言提出了“结构化程序设计方法”，规定程序必须由具有特定的基础结构（顺序结构、选择结构、循环结构）构成，程序中的流程不允许随意跳转，程序总是由上到下顺序执行各个基本结构。这种程序结构清晰，易于编写、阅读和维护。QBASIC、C 语言等都属于结构化语言。这些语言的特点是支持结构化程序设计方法。

以上两种语言都是基于过程的语言，在编写程序时需要具体指定每一个过程的细节。在编写规模较小的程序时，还能得心应手，但在处理规模较大的程序时，就显得捉襟见肘了。在实践的发展中，人们又提出了面向对象的程序设计方法。程序面对的不是过程的细节，而是一个个对象，这些对象是由数据以及对数据进行的操作组成的。

3. 面向对象语言

面向对象语言是一类以对象作为基本程序结构单位的程序设计语言，用于描述的设计是以对象为核心，而对象是程序运行的基本成分。这些语言中提供了类、继承等。C++ 、C# 、Visual Basic 和 Java 等都是支持面向对象程序设计的语言。

1.2 C 语言的发展和特点

C 语言是当代流行的计算机语言之一，因为它是一种独立于机器、结构化的高级语言。它允许软件开发者开发程序而无须考虑硬件平台。

1.2.1 C 语言的发展

C 语言的根源是 ALGOL 60 语言，它是在 20 世纪 60 年代初提出的。ALGOL 语言是第一个使用块结构的高级语言。计算机科学界提出了初步的结构化程序设计概念，并由计算机科学家 Corrado Bohm、Guiseppe Jacopini 和 Edsger Dijkstra 进行了推广。

1963 年英国剑桥大学推出了 CPL(Combined Programming Language，组合程序设计语言)，它是在 ALGOL 60 的基础上开发的，但规模较大。

1967 年，英国剑桥大学的 Martin Richards 在 CPL 语言基础上，经过简化，开发出了 BCPL(Basic Combined Programming Language，基本组合程序设计语言)，主要用于编写系统软件。

20 世纪 60 年代，美国贝尔实验室的研究员 Ken Thompson(以下简称为 ken)看到阿波罗 11 号载人飞船登月成功，觉得很酷，自己设计了一个叫“Space Travel”的游戏。在没有商业游戏的年代，这种 DIY(Do It Yourself)的游戏被别人喜欢也是一件很酷的事情。这个游戏先是在 Multics 系统上编写的，后来又在 CECOS 系统上重写。能运行这两个系

统的机器都是笨重的大型机，虽然运行能力很强，但显示效果很差，同时机时费非常高。因此 ken 与他的同事 Dennis M. Ritchie(以下简称为 dmr)一起寻找免费的“游戏机”，找到了一台空闲的机器——PDP-7。PDP-7 是一台小型机，由 DEC 公司制造，拥有当时最先进的图形处理能力。那时计算机的主要用途是数据处理，图形处理能力并不太重要，因此 PDP-7 被闲置了。但这台机器没有操作系统，而游戏必须使用操作系统的一些功能，于是他着手为 PDP-7 开发操作系统。后来，这个操作系统被命名为 UNIX。直到今天，UNIX 仍然是最受信任的操作系统，它既支撑着军队、政府、电力、电信和银行等大型机构的关键业务，又是苹果 Mac 系列计算机的动力之源，甚至 iPhone、iPod Touch 的部分魅力也拜其所赐。

UNIX 起初是用汇编语言编写的，由于汇编语言缺少可移植性，针对一种计算机编写的汇编程序不能在另一种计算机上直接使用，必须重写，因此 ken 和 dmr 决定改用高级语言编写 UNIX，这样它就可以在更多类型的计算机上运行。

决定使用高级语言后，在语言的选择上，他们又遇到了麻烦，当时已有的高级语言都是面向应用程序编写而设计的，层次太高，不适合用来开发操作系统。DIY 精神再次发挥作用，他俩决定自己设计一个适合编写 UNIX 的高级语言。

1970 年，ken 对 BCPL 语言进行了进一步的提炼，创建了 B 语言(取自 BCPL 的首字母)，并用 B 语言创建了 UNIX 操作系统的早期版本。BCPL 和 B 两者都是“无类型”(typeless)系统程序设计语言。

1972 年至 1973 年间，dmr 和 ken 继续合作，dmr 负责设计新语言，他在 B 语言的基础上推出了 C 语言。该语言保持了 BCPL 和 B 语言的精炼、靠近硬件等优点，并增加了数据类型的概念和其他强有力的特性。ken 和 dmr 合作将 90% 以上的 UNIX 操作系统用 C 语言改写(即 UNIX 第 5 版)。此时的 C 语言主要还是在贝尔实验室内部使用，这一时期的 C 语言被称为“传统 C”。

直到 1975 年 UNIX 第 6 版发布以后，随着许多商用 C 语言编译程序的发布和 UNIX 的日益流行，C 语言终于获得了计算机专业人士的广泛支持。C 语言先后移植到大、中、小和微型计算机上。

1978 年，由 Brian W. Kernighan 和 dmr 所著的 *The C Programming Language*(《C 程序设计语言》)出版后，该语言更加流行了，也成为了实际上的第一个 C 语言标准。

1983 年，美国国家标准协会(ANSI)成立了一个委员会，根据 C 语言问世以来各种版本对 C 语言的发展和扩充，制定了第一个 C 语言标准草案，称为 83 ANSI C。1989 年，ANSI 公布了一个完整的 C 语言标准——ANSI X3.159-1989(简称为 ANSI C 或 C89)。之后，在 1990 年，该标准被国际标准化组织(ISO)批准为国际 C 标准，即 ISO C 标准。1999 年，ISO 又对 C 语言标准进行修订，称为新的国际标准，简称为 C99。2011 年，ISO 又发布了新的标准，称为 ISO/IEC9899:2011，简称为 C11。本书中的所有例题、习题的参考答案所涉及的 C 程序都是在 Microsoft Visual C++ 2010 Express 的开发环境下编译和运行通过的。

1.2.2 C语言的特点

C语言之所以能够得到广泛的应用，主要是它具有突出的优点。C语言的主要特点如下。

1. 语言简洁、紧凑，使用方便、灵活

C语言共有32个关键字、9种控制语句，程序书写的形式自由，主要用小写字母表示，压缩了一切不必要的成分，用C语言开发软件效率会更高。

2. 运算符丰富

C语言包含的运算符共有34种，C语言把括号、赋值、强制类型转换等都按照运算符来处理，使其表达式的种类多样化，运算类型也十分多样。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语言提供了编程所需要的各种数据类型，主要有字符型、整型、浮点型、枚举类型、数组类型、结构体类型、共用体类型、指针类型、空类型等。丰富的数据类型可以表示各种复杂数据结构的运算。

4. 支持结构化、模块化程序设计方法

C语言具有良好的结构化控制语句，如顺序结构的语句、选择结构的语句(如if语句、if-else语句、switch语句等)和循环结构的语句(如while语句、do-while语句、for语句等)。C语言的函数作为程序的模块单位，便于实现程序的模块化。

5. 程序设计自由度大

一般的高级语言的语法检查比较严格，几乎能检查出所有的语法错误，而C语言则放宽了语法检查，允许程序员有较大的设计自由度。C语言对数组下标不做越界检查，对变量和常量的数据类型的使用比较灵活，如允许字符型、整型、逻辑型等数据通用等。语法限制不严格的优点是增强了程序设计的灵活性、提供了容错能力、加强了处理能力；缺点是编写程序容易出错。

6. C语言允许直接对硬件进行操作

C语言既具有高级语言的功能，又具有低级语言的许多功能，可用来编写系统软件，直接对硬件进行操作。

7. 可移植性好

用C语言编写的程序可移植性好，几乎所有的计算机系统都可以使用C语言。