普通高等教育“十三五”规划教材

C语言程序设计

C YUYAN CHENGXU SHEJI

肖 捷 侯家利 主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

普通高等教育“十三五”规划教材

C语言程序设计

主　编　肖　捷　侯家利

副主编　王　宁　冯能山　彭富春
陈雪芳　何文斌

主　审　李　勇　徐钦桂

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书是为将 C 语言作为入门语言的程序设计课程而编写的教材，以培养学生程序设计基本能力为目标。

本书包含程序设计和语言知识两条线索，其中程序设计为主线，基于“阶梯递进”模式（案例分析→模仿改写→独立编程 3 个环节），以编程应用为驱动，通过案例和问题引入内容，重点讲解程序设计的思想和方法；同时结合语言知识辅线，穿插讲解相关的语言知识。为了配合本书的学习，作者还编写了与本书配套的《C 语言程序设计实训教程与水平考试指导》，可供读者学习时参考使用。通过本书的学习，学生能较全面地掌握 C 语言的语言知识以及 C 程序设计的基本方法和技巧。

本书适合作为高等院校程序设计课程的教学用书，也可作为从事计算机应用的科技人员的参考书及培训教材。

图书在版编目（CIP）数据

C 语言程序设计 / 肖捷，侯家利主编. —北京：中国铁道出版社，2016.1
普通高等教育“十三五”规划教材
ISBN 978-7-113-21162-2

Ⅰ. ①C… Ⅱ. ①肖… ②侯… Ⅲ. ①C 语言－程序设计－高等学校－教材 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2015）第 321221 号

书　　名：C 语言程序设计
作　　者：肖　捷　侯家利　主编

策　　划：刘丽丽　　读者热线：010-63550836
责任编辑：唐　旭　冯彩茹
封面设计：刘　颖
封面制作：白　雪
责任校对：汤淑梅
责任印制：李　佳

出版发行：中国铁道出版社（100054，北京市西城区右安门西街 8 号）
网　　址：http://www.51eds.com
印　　刷：三河市兴达印务有限公司
版　　次：2016 年 1 月第 1 版　　2016 年 1 月第 1 次印刷
开　　本：787 mm×1 092 mm　1/16　　印张：22.25　　字数：536 千
书　　号：ISBN 978-7-113-21162-2
定　　价：47.00 元

前言

FOREWORD

程序设计是高等院校重要的计算机基础课程，它以编程语言为平台，介绍程序设计的思想和方法。学生通过该课程的学习，不仅要掌握程序设计语言的知识，更重要的是在实践中逐步掌握程序设计的思想和方法，培养求解问题和应用程序语言的能力。因此，这是一门以培养学生程序设计基本方法和技能为目标的程序设计基础课程。目前，C 语言已被许多高等院校列为程序设计课程的首选语言。

C 语言程序设计是一门实践性很强的课程，学生必须通过大量的编程训练，在实践中掌握程序设计语言，培养程序设计的基本能力，并逐步理解和掌握程序设计的思想和方法。因此，培养学生的实际编程能力是课程教学的重点，教材的组织必须满足课程教学的要求。

目前，介绍 C 语言的教材很多，但在多年的教学实践中我们发现，比较适合程序设计入门课程教学的教材并不多。现有的许多教材一般围绕语言本身的体系展开内容，以讲解语言知识特别是语法知识为主线，辅以一些编程技巧的介绍，不利于培养学生的程序设计能力和语言应用能力。当然，C 语言的案例教材也不少，但在案例分析时，问题分析和算法设计描述不够，主要突出程序代码和代码解析两个方面，因此，也不利于培养学生分析问题的逻辑思维能力。

本书较好地解决了传统教材的不足，在组织结构上包含程序设计和语言知识两条线索，以程序设计为主线，基于“阶梯递进”模式（案例分析→模仿改写→独立编程 3 个环节），以编程应用为驱动，通过案例和问题引入内容，重点讲解程序设计的思想和方法，并结合语言知识辅线，穿插讲解相关的语言知识。“案例分析”基于问题求解的基本过程，即问题分析、算法设计、编写程序和调试运行 4 个过程，以前 3 个过程为重点，通过经典案例，着重讲解程序设计的思想、方法和编程风格。“模仿改写”是针对本节中的相关概念和“案例分析”，在每节的模仿改写练习中给出一些难度较低的相关问题，学生可以模仿案例完成，以加深理解，提高兴趣。“独立编程”是“阶梯递进”模式的最后环节，在每章习题中给出一些难度稍大的编程问题，学生可以在前两个环节的基础上独立完成并上机调试通过。因此，本书比较适合作为程序设计入门课程教学的教材，有利于培养学生的程序设计能力和语言应用能力。

在教材的结构设计上，本书注重编程实践，让学生从第 1 周起就练习编程，使程序设计主线贯穿始终。前两章简单介绍一些背景知识和利用计算机求解问题的过程，然后从案例出发，介绍顺序、分支和循环 3 种控制结构的最简单使用形式及函数的简单使用，使学生对 C 语言有一个总体的了解，并学习编写简单的程序，培养学习兴趣。第 3 章介绍 C 语言的基本数据类型和表达式，为后续章节做准备。从第 4 章开始，逐步深入地讲解程序设计的思想和方法，并说明如何应用语言知识解决问题。

本书共有 12 章内容和 1 个附录，分成 4 个部分。第 1 部分：简单程序设计，学习编写简单程序，培养学习兴趣，包括第 1～3 章，第 1 章介绍程序与程序设计语言的知识以及利用计算机求解问题的过程；第 2 章从实例出发，简单介绍顺序、分支和循环 3 种控制结构的

最简单使用形式及函数的简单使用，以及在实例程序中用到的语言知识，使学生对 C 语言有一个总体的了解；第 3 章介绍数据类型和表达式等基本语言知识，为后续章节做准备。第 2 部分：控制结构程序设计，基于简单数据类型，学习编写 3 种控制结构的程序，包括第 4～6 章，通过大量的案例分析，进一步介绍分支结构、循环结构以及函数结构的程序设计思想和方法，侧重基本知识和基本编程能力。第 3 部分：基于构造数据类型的程序设计，学习用模块化方法实现有一定复杂度的编程问题和基本算法等内容，包括第 7～12 章，第 7 章介绍数组类型的基本知识，并通过大量的案例分析，介绍模块化方法在一维数组、二维数组和字符串中的编程应用；第 8 章介绍指针类型的基本知识，并通过大量的案例分析，介绍模块化方法在指针与数组相结合的编程应用；第 9 章介绍结构类型的基本知识，并通过大量的案例分析，介绍模块化方法在结构与数组、结构与指针类型中的编程应用；第 10 章介绍链表的基本知识，并通过案例分析，介绍链表结构的基本操作和模块化方法在链表结构中的编程应用；第 11 章介绍共用体与枚举类型的基本知识，并通过案例分析，简单介绍共用体与枚举类型的编程应用；第 12 章介绍文件的基本知识，并通过案例分析，介绍文件结构的基本操作和模块化方法在文件结构中的编程应用。第 4 部分：附录，以备读者速查。附录 A 为常用字符与 ASCII 代码对照表；附录 B 为 C 库函数，分类列出 ANSI C 的常用标准库函数。附录 C 为常见错误分析，列出常见的编译错误、连接错误和运行错误，分析出错原因并给出相应的解决方法。

为了配合本书的学习，作者还编写了与本书配套的《C 语言程序设计实训教程与水平考试指导》，可供读者学习时参考使用。该书由 5 部分组成，第 1 部分～第 4 部分为实验指导部分，第 5 部分为全国高等学校计算机水平考试的相关试题及参考答案。

本书由肖捷、侯家利任主编，王宁、冯能山、彭富春、陈雪芳、何文斌任副主编。全书由肖捷统稿，编写分工为：第 1、2、10、11 章由肖捷编写，第 3 章由何文斌编写，第 4、5 章由侯家利编写，第 6、7 章由肖捷、王宁共同编写，第 8、9 章由肖捷、彭富春共同编写，第 12 章由冯能山编写，附录由陈雪芳编写。东莞理工学院李勇教授和徐钦桂教授认真、仔细地审阅了全书，并提出了许多宝贵意见，在此表示衷心感谢。另外，在本书编写、修订过程中，许多老师和同学都提出了宝贵的意见和建议，在此一并表示感谢。

为了便于读者学习，本书还提供了大量的教学资源，读者可以登录东莞理工学院“C 语言程序设计”课程教学网站 http://172.27.2.1/cweb，共享课程教学资源。另外，读者也可以通过电话（13549379596）或 E-mail（398948928@qq.com）与作者联系，获取课程教学资源。

由于编者水平有限，加之时间仓促，疏漏和不足之处在所难免，敬请读者批评指正。

编　者

2015 年 11 月

目 录

CONTENTS

第1章

C语言程序设计概述

本章要点

- 程序的概念，程序设计语言发展的几个阶段。
- 算法的概念，算法描述方法——自然语言、流程图和伪代码。
- C语言的特点。
- C语言程序的基本框架。
- C语言的主要语法单位。
- C语言程序上机步骤。
- 实现问题求解的过程。

计算机本身是没有生命的机器，要使计算机能够运行起来，为人类完成各种各样的工作，就必须让它执行相应的程序。这些程序都是依靠程序设计语言编写出来的。在众多的程序设计语言中，C语言作为一种高级程序设计语言，既具有高级语言的方便性、灵活性和通用性等特点，又兼备低级语言的特性，提供程序员直接操作计算机硬件的功能。适合各种类型的软件开发，深受软件工程技术人员的青睐。

本章主要从程序设计的角度，介绍有关程序设计的基本概念、程序设计语言的发展、C语言程序的基本结构、算法、特点以及程序设计求解问题的一般步骤等。

1.1 程序与程序语言

1.1.1 程序的基本概念

程序的概念来源于日常生活，通常，完成一项复杂的任务需要分解成一系列的具体步骤。这些按一定的顺序安排的具体步骤就是程序。例如，开学典礼程序、联欢晚会程序等。

随着计算机的出现和发展，程序成了计算机的专有名词，计算机都是在程序的控制下运行的。所谓计算机程序，就是用计算机语言描述的解决某一问题的一系列加工步骤，是符合一定语法规则的符号序列。程序设计就是借助计算机语言，告诉计算机要处理什么（即处理数据）以及如何处理（即处理步骤）。执行程序就是向计算机发出一系列指令，让计算机按程序规定的步骤和要求解决特定问题。

同一问题有不同的解决方法，不同用户编写的程序也并不完全相同。而且不同的程序有不同的效率，这就涉及程序的优化，涉及程序所采用的数据结构和算法等方面。

1.1.2 程序设计语言

程序设计离不开程序设计语言。了解程序设计语言的发展过程，有助于读者加深对程序设计语言的认识，使其能更好地利用程序设计语言解决实际问题。

程序设计语言的发展很快，新的程序设计语言不断出现，功能也越来越强大。从其发展过程来看，程序设计语言的发展大致经历了以下几个阶段。

1. 机器语言

所谓机器语言，就是指计算机能够识别的指令集合，即指令系统。在机器语言中，每条指令都用二进制0和1组成的序列来表示。例如，某计算机的加法指令为10000000，减法指令为10010000。不同类型的计算机，机器语言也不相同。

用机器语言编写的程序，计算机可以直接执行，且执行效率高，这是机器语言的优点。但机器语言的指令不直观，难认、难记、难理解，容易出错，编程缺乏通用性，编程人员需要查阅机器指令系统，编程效率低。因此，目前很少直接用机器语言编写程序。

2. 汇编语言

由于机器语言的编程效率低，为了减轻程序开发人员的编程负担，开始采用一些助记符号来表示机器语言中的机器指令，这样便出现了汇编语言。助记符号一般采用代表某种操作的英文单词缩写，与机器语言相比，便于识别和记忆。例如，上述两条加法指令和减法指令可以用助记符号ADD和SUB来表示。

用汇编语言编写的程序称为源程序，计算机不能直接执行，必须经过汇编程序翻译成机器语言程序才能执行。

对比机器语言，汇编语言指令和机器语言指令具有一一对应关系，不同类型的计算机，其汇编语言也不尽相同，编程时仍需要熟悉机器的内部结构，比较烦琐。但相对机器语言而言，其编程效率有较大提高。在实际应用中，如果对程序运行时间比较严格，与硬件操作比较紧密，程序设计人员还是常用汇编语言编程来解决实际问题的。

3. 高级语言

机器语言和汇编语言都是面向机器的编程语言，同属低级语言的范畴。主要缺点是编程效率低、需要熟悉机器硬件。为了克服低级语言的这一缺点，提高程序设计人员的编程效率，出现了面向算法过程的程序设计语言，称为高级语言。例如，Fortran语言、Pascal语言、C语言等。

高级语言比较接近自然语言的形式，功能强大，一条语句相当于多条汇编语言指令或机器语言指令。编程时，不需要熟悉机器的内部结构，编程人员可以把精力集中在研究问题的求解方法上，大大降低了编程的难度，提高了编程的效率和质量，而且设计的程序也更容易阅读和理解。因此，在实际应用中被广泛用来解决实际问题。

当然，计算机也不能直接执行高级语言源程序，必须经过编译和连接过程，将其翻译成机器语言程序才能由计算机执行。

4. 面向任务的程序设计语言

高级语言是面向过程的编程语言，用高级语言编程求解一个复杂问题，必须首先分析解题过程，描述问题的求解算法，然后才能用高级语言编程实现。面向任务的程序设计语言是非过程化语言，无须知道问题如何求解，只需描述求解什么问题，便可编程实现。

数据库语言便是一种面向任务的程序设计语言。例如，SQL Server是一种关系数据库系

统，它提供了数据库查询语言（SQL），采用SQL提供的select语句，便可方便、快速地查询出数据库中的数据信息。例如，查询语句select * from student where sex='男'，其功能就是查询学生信息表（student）中所有男性学生的全部信息。

由于面向任务的程序设计语言不仅大大降低了编程的复杂度，而且提高了应用程序的开发速度和质量，使得编程工作不再是计算机专业人员的专利，许多非计算机专业人员也能很方便地使用面向任务的程序设计语言开发自己的应用程序。这类语言被广泛应用在管理信息系统应用软件的开发方面。

5. 面向对象的程序设计语言

面向对象的程序设计语言于20世纪90年代开始流行，目前已成为程序设计的主流语言。C++就是一种非常优秀的面向对象的程序语言，它是由C语言发展而来的。

面向对象方法学是一种分析方法、设计方法和思维方法的综合。它的出发点和所追求的基本目标就是使人们分析、设计和实现一个系统的方法尽可能接近人们认识一个系统的方法。

面向对象的编程，程序被看成相互协作的对象集合，每个对象都是某个类的实例，所有的类构成一个通过继承关系相联系的层次结构。面向对象程序设计就是针对客观事物（对象）设计程序，与面向过程的编程方法相比，编程工作更加直观、清晰，编程效率更高，更适合开发大型复杂的软件。

面向对象的程序设计语言通常具有类的定义功能、对象的生成功能、消息传递机制和类的继承机制。目前，Java就是一种流行的、被广泛应用的面向对象的程序设计语言。

综上所述，可以知道，每一种语言都有其优点和不足，对于不同的问题，需要根据实际情况来选择程序设计语言，以便更加高效、更加优质地解决相关的问题。

1.2 算法及其描述

在程序设计中，也经常涉及算法。瑞士著名计算机科学家Niklaus Wirth提出了程序定义的著名公式：**程序=数据结构+算法**。这个公式说明了程序与算法的关系，也足以说明算法在程序设计中的重要性。下面从算法的概念和描述方法两个方面讨论算法问题。

1.2.1 算法的概念

在日常生活中，做任何事情都是按照一定规则一步一步地进行的，例如，新生开学典礼就是按预设步骤一步一步进行，直到结束。这些预设步骤就是新生开学典礼的算法。

通常认为，算法就是对特定问题求解步骤的一种描述。算法应具备以下5个特点：

① 有穷性。算法必须保证执行有穷步之后结束，不能无止境地执行下去。

② 确定性。算法必须保证每一步必须具有确切的含义，不能有二义性。

③ 有效性。算法必须保证每一步操作都是可执行的。

④ 要有数据输入。算法中操作的对象是数据，因此，算法应该提供数据输入。

⑤ 要有结果输出。算法是用来解决一个给定的问题，因此，算法应该提供结果输出。

1.2.2 算法的描述方法

算法就是对特定问题求解步骤的一种描述。为了描述一个算法，可以用不同的方法。通常的方法包括自然语言、传统流程图、结构化流程图、伪代码等。

1. 自然语言

自然语言就是人们日常使用的语言，可以是汉语、英语，或其他语言。

自然语言描述的算法通俗易懂，便于用户间交流。但文字冗长，含义往往不太严格，需要根据上下文才能判断其正确含义，容易出现歧义性。一般用来描述较简单的算法。

例如，求 sum=1+2+…+n。算法用自然语言描述如下：

算法设计

第一步：置初值，累加和 sum 置 0，累加项 i 置 1。

第二步：输入待求和数的个数 n。

第三步：求累加和，重复执行下面操作，直到 i>n。

- 累加项：sum+i => sum
- 求下一项：i+1 => i

第四步：输出 sum 的值。

2. 流程图

流程图就是借助一组专用的图框和线条来描述算法，用图框表示各种操作，用线条表示操作的执行顺序。它的特点是直观形象，易于理解。美国国家标准学会（ANSI）规定了一组常用的流程图符号（见图 1-1），已被世界各国所采用。

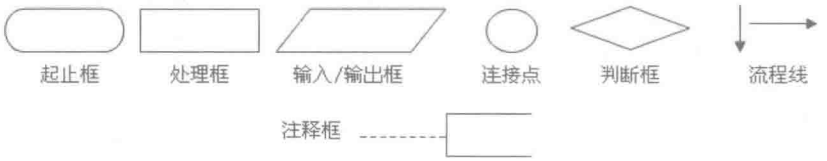

图 1-1 流程图标准化符号

① 起止框：扁圆形，表示流程图的开始或结束。

② 处理框：矩形，表示各种处理功能，其内注明处理名称或简要功能。

③ 输入/输出框：平行四边形，表示数据的输入或输出。

④ 连接点：表示两部分流程图的连接处。

⑤ 判断框：菱形，表示判断，注明判断条件，只有一个入口，可以有多个选择出口。

⑥ 流程线：箭头，表示操作执行顺序。

⑦ 注释框：对流程图中的某些框作补充说明，帮助阅读流程图。

例如，求 sum=1+2+…+n。算法用流程图描述如图 1-2 所示。

3. 伪代码

流程图描述算法直观形象，易于理解。但画起来比较费事，在设计一个算法时，可能需要反复修改，对流程图的修改比较麻烦。流程图适合描述一个算法，但在设计算法过程中使用不太理想（因为需要反复修改）。为了设计算法时方便，常使用一种称为伪代码（Pesudo Code）的工具。

伪代码是用介于自然语言和计算机语言之间的文字和符号来描述算法。它如同一篇文章，自上而下地写下来，每一行（或几行）表示一个基本操作，不需使用图形符号。因此，书写方便，格式紧凑，容易理解，也便于向程序过渡。