非计算机专业计算机公共课系列教材

高级语言程序设计——C语言

主编 汪同庆 张 华

WUHAN UNIVERSITY PRESS
武汉大学出版社

非计算机专业计算机公共课系列教材

高级语言程序设计——C语言

主编　汪同庆　张　华
参编　关焕梅　侯梦雅　刘　珺　王　鹃　谭明新

WUHAN UNIVERSITY PRESS
武汉大学出版社

图书在版编目(CIP)数据

高级语言程序设计:C 语言/汪同庆,张华主编. —武汉:武汉大学出版社,2010.2

非计算机专业计算机公共课系列教材

ISBN 978-7-307-07584-9

Ⅰ.高… Ⅱ.①汪… ②张… 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(2010)第 015797 号

责任编辑:林 莉 责任校对:刘 欣 版式设计:支 笛

出版发行:**武汉大学出版社** (430072 武昌 珞珈山)

(电子邮件:cbs22@whu.edu.cn 网址:www.wdp.com.cn)

印刷:通山金地印务有限公司

开本:787×1092 1/16 印张:20.25 字数:515 千字 插页:1

版次:2010 年 2 月第 1 版 2010 年 2 月第 1 次印刷

ISBN 978-7-307-07584-9/TP·352 定价:33.00 元

非计算机专业计算机公共课
系列教材编委会

前 言

C 语言是一种功能强大、编程灵活、特色鲜明，深受国内外广大科技人员和编程者喜爱的计算机语言。自 20 世纪 90 年代以来，我国大多数高校不仅为计算机专业，而且为非计算机专业都开设了 C 语言课程。全国计算机等级考试、全国计算机应用技术证书考试、全国计算机软件专业技术资格及水平考试等都将 C 语言纳入了考试科目。可以这样说，现在的很多编程高手都是从学习 C 语言入门的。因此，掌握好这门课程对每一位立志成为优秀程序员的初学者是大有裨益的。

本书针对非计算机专业的应用特点和全国计算机等级考试大纲的要求，重点对 C 语言程序的开发环境，基本语句，基本数据类型，构造类型，指针类型，控制结构和文件操作进行了全面介绍。考虑到许多学校把 C 语言课程安排在“大一”学年，而高等数学内容还未学完，因此书中在举例时摒弃了一些复杂的应用，便于自学。全书内容精练，结构合理，概念清晰，通俗易懂，实用性强，各章都附有大量的习题和上机操作题供学生实训练习，以期让读者能尽快和轻松地迈进程序设计的大门。全书共 13 章，主要内容包括：

- 计算机语言与程序设计基本知识
- 数据类型、运算符和表达式
- 顺序结构程序设计
- 选择结构程序设计
- 循环结构程序设计
- 函数
- 指针
- 数组
- 字符串
- 结构体、共用体和枚举
- 编译预处理
- 位运算
- 文件

本书基于 Visual C++ 6.0 编程环境，书中全部例程均在 Visual C++ 6.0 编程环境下编译、链接和执行。本书的第 1、10、11 章由关焕梅编写，第 2 章由侯梦雅编写，第 3 章由王鹃编写，第 4、5、6 章由汪同庆编写，第 7、8、9 章由张华编写，第 12 章由谭明新编写，第 13 章由刘珺编写。全书由汪同庆、张华统编定稿。

在本书的编写过程中，得到了武汉大学珞珈学院领导和武汉大学出版社的大力支持，许多老师在编写过程中给予了大量的帮助并提出了宝贵意见，在此表示衷心感谢。

受作者水平所限，书中难免存在错漏之处，恳请同行专家和热心读者批评指正。

作　者

2010年1月

目 录

第1章 计算机语言与程序设计基本知识

作为一种程序设计语言，C 语言既具有高级语言的特性，又具有低级语言的特性。它可以作系统设计语言，编写系统程序；也可以作为应用程序设计语言，编写不依赖硬件的应用程序。C 语言以其强大的功能、灵活的应用，深受广大用户青睐。

本章主要介绍计算机语言分类、计算机语言处理程序、C 语言的发展和特点、计算机程序、算法及其表示、结构化程序设计、C 程序的基本构成，以及 C 程序的开发环境等。

1.1 计算机语言

1.1.1 计算机语言分类

人和计算机交流信息使用的语言称为计算机语言或程序设计语言。计算机语言通常分为机器语言、汇编语言和高级语言三类。

1. 机器语言

机器语言是用二进制代码表示的机器指令的集合。机器语言是计算机硬件系统能够直接识别和执行的唯一语言，因此，它的效率最高、执行速度最快。但不同型号的计算机，其机器语言是不相通的，因此程序不容易移植。

2. 汇编语言

汇编语言是一种把机器语言“符号化”的语言，汇编语言的指令和机器语言的指令基本上一一对应，机器语言直接用二进制代码，而汇编语言使用了助记符，如用 ADD 表示加法指令，MOV 表示减法指令等。汇编语言仍然依赖于机器。

汇编语言比机器语言容易理解和记忆，但汇编语言源程序不能在计算机中直接执行。

3. 高级语言

高级语言不依赖于机器，更接近于自然语言或数学语言。高级语言的种类很多，如 C、C++、Java、Visual Basic、Delphi 和 JavaScript 等。

高级语言具有面向用户、可读性强、容易编程和维护等特点。

同汇编语言一样，高级语言源程序也不能在计算机中直接执行。

1.1.2 计算机语言处理程序

计算机语言处理程序一般是由汇编程序、编译程序、解释程序和相应的操作程序等组成。它是为用户设计的编程服务软件，其作用是将汇编语言源程序或高级语言源程序翻译成计算机能识别的机器语言程序。

汇编语言源程序需要通过“汇编程序”翻译成机器语言程序。

高级语言源程序有两种翻译方式：编译和解释。

1. 编译方式

编译方式首先利用“编译程序”将源程序翻译成目标程序，然后通过“链接程序”将目标程序链接成可执行程序。如图 1.1 所示。可执行程序可以脱离源程序和编译程序单独运行，所以编译方式效率高，执行速度快。编译型高级语言有 C、C++和 Delphi 等。

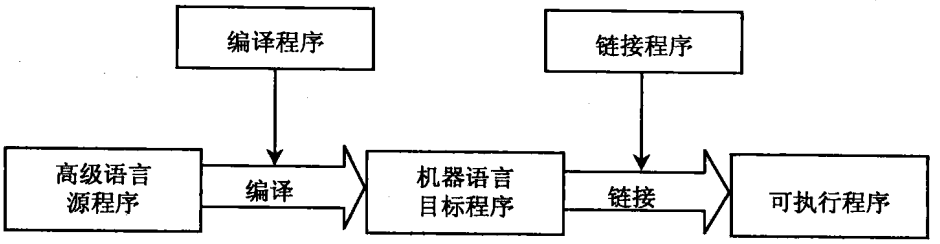

图 1.1　高级语言源程序的编译方式

2. 解释方式

解释方式是利用“解释程序”对源程序逐句翻译，逐句执行。解释过程不产生目标程序，基本上是翻译一行执行一行，边翻译边执行。如果在解释过程中发现错误就会给出错误信息，并停止解释和执行。如果没有错误就解释执行到最后的语句。与编译方式相比，解释方式效率低、执行速度慢。但解释方式具有良好的动态特性，调试程序方便，跨平台特性好。很多脚本语言如 JavaScript、VBScript 和 PHP 等都是解释型高级语言。

也有一些高级语言将编译方式和解释方式结合起来，如 Java。Java 语言源程序先由 JIT（Just-In-Time）编译器翻译为字节码，再用解释方式执行字节码。

1.1.3　C 语言简介

C 语言是美国贝尔实验室的 Dennis Richie 在 1972 年开发的，开发的主要目的是用于设计 UNIX 操作系统。之所以称为 C 语言，是因为它的前身是 B 语言。而 B 语言是由 Ken Thompson 于 1970 年为第一个 UNIX 系统开发的语言。

随着 UNIX 操作系统的广泛使用，C 语言也得到了迅速的发展，世界各地的程序员都使用它来编写各种程序。然而，不久后，不同的组织开始使用自己的 C 语言版本，不同版本实现之间微妙的差别令程序员头痛。为解决这种问题，美国国家标准化组织（ANSI）于 1983 年成立了一个委员会（X3J11），以确定 C 语言的标准。该标准（ANSI C）于 1989 年正式采用。国际标准化组织（ISO）于 1990 年采用了一个 C 标准（ISO C）。ISO C 和 ANSI C 实质上是同一个标准，通常被称为 C89 或 C90。现代的 C 语言编译器绝大多数都遵守该标准。

最新的标准是 C99 标准。制定该标准的意图不是为语言添加新特性，而是为了满足新的目标（例如支持国际化编程），所以该标准依然保持了 C 语言的本质特性：简短、清楚和高效。目前，大多数 C 语言编译器没有完全实现 C99 的所有修改。本书将遵循 C89 标准，并不涉及 C99 的修改。

C 语言的主要特点如下：

（1）C 语言是一种结构化程序设计语言。

C 语言提供了结构化程序所必需的基本控制语句，如选择语句和循环语句等，实现了对

逻辑流的有效控制。C 语言采用模块化的方法来解决实际问题。

（2）C 语言具有丰富的数据类型。

C 语言除提供整型、实型和字符型等基本数据类型外，还提供了由基本数据类型构造出的复杂数据结构，如数组和结构等。C 语言还提供了与地址密切相关的指针类型。

（3）C 语言具有丰富的运算符。

C 语言提供了 44 种运算符，运算能力十分丰富。多种数据类型与丰富的运算符相结合，能使表达式更具灵活性，可以实现其他高级语言难以实现的功能，同时也提高了执行效率。

（4）C 语言结构紧凑，使用方便、灵活。

C 语言只有 32 个关键字，9 种控制语句，大量的标准库函数可供直接调用。C 程序书写形式自由。

（5）C 语言具有自我扩充能力。

一个 C 程序基本上是各种函数的集合，这些函数由 C 语言的函数库支持，并可以重复被用在其他程序中。用户可以不断地将自己开发的函数添加到 C 语言函数库中。由于有了大量的函数，C 语言编程也就变得简单了。

（6）C 语言可移植性好。

C 语言是可移植的，这意味着为一种计算机系统（如一般的 PC 机）编写的 C 程序，可以在不同的系统（如 HP 的小型机）中运行，而只需作少量的修改或不加修改。这种可移植性也体现在不同的操作系统之间，如 Windows 和 Linux。

（7）C 语言具有低级语言的特性。

C 语言既具有高级语言的特性，又具有低级语言的特性，因此可用于编写各种软件。

1.2　程序设计

1.2.1　计算机程序

利用计算机处理问题，必须编写使计算机能够按照人的意愿工作的程序。所谓程序，就是计算机解决问题所需要的一系列代码化指令、符号化指令或符号化语句。著名的计算机科学家沃思（Wirth）提出过一个著名的公式来表达程序的实质，即

程序 = 数据结构 + 算法

也就是说，“程序是在数据的某些特定的表示方式和结构的基础上，对抽象算法的具体描述”。但是，在实际编写计算机程序时，还要遵循程序设计方法，在运行程序时还要有软件环境的支持。因此，有学者将上述公式扩充为：

程序 = 数据结构 + 算法 + 程序设计方法 + 语言工具

即一个应用程序应该体现四个方面的成分：采用的描述和存储数据的数据结构，采用的解决问题的算法，采用的程序设计的方法和采用的语言工具和编程环境。

在学习利用计算机语言编写程序时要掌握三个基本概念。一是语言的语法规则，包括常量、变量、运算符、表达式、函数和语句的使用规则；二是语义规则，包括单词和符号的含义及其使用规则；三是语用规则，即善于利用语法规则和语义规则正确组织程序的技能，使程序结构精练、执行效率高。

此外，还要弄清“语言”和“程序”的关系。语言是构成程序的指令集合及其规则，程

序是用语言为实现某一算法组成的指令序列。学习计算机语言是为了掌握编程工具，它本身不是目的。当然，脱离具体语言去学习编程是十分困难的，因此两者有密切的联系。

1.2.2 算法及其表示

1. 算法的概念

所谓算法是指对解题方案的准确而完整的描述。对于一个问题，如果可以通过一个计算机程序，在有限的存储空间内运行有限长的时间而得到正确的结果，则称这个问题的算法是可解的。但算法不等于程序，也不等于计算方法。

算法有以下四个基本特征：

（1）可行性。

算法总是在某个特定的计算工具上执行，因此算法在执行过程中往往要受到计算工具的限制，使执行结果产生偏差。算法和计算公式是由差别的，在设计一个算法时，必须考虑它的可行性，否则将得不到满意的结果。

（2）确定性。

算法中的每一个步骤必须有明确的定义，不能产生歧义。

（3）有穷性。

算法必须能在有限的时间（合理时间）内做完，即能在执行有限个步骤后终止。

（4）拥有足够的情报。

一个算法是否有效，还取决于为算法所提供的情报是否足够。当算法拥有的情报不够时，算法可能无效。

2. 算法的表示

有多种方法可以表示算法，如自然语言、伪代码、流程图、N-S 图和问题分析图（PAD）等。下面介绍如何用自然语言、流程图和 N-S 图来表示算法。

（1）用自然语言表示算法。

用自然语言表示算法，就是用人们日常使用的语言来描述或表示算法的方法。

【例 1.1】 从键盘输入三个整数，将最大数输出。

用自然语言表示如下：

步骤 1：声明整型变量 x、y、z 和 max。其中：x、y、z 分别代表三个整数，max 代表较大数；

步骤 2：从键盘输入三个整数，分别存入 x、y、z 中；

步骤 3：比较 x 和 y。如果 x > y，则将 x 的值赋给 max，即 max = x；否则将 y 的值赋给 max，即 max = y；

步骤 4：比较 z 和 max。如果 z > max，则将 z 的值赋给 max，即 max = z；否则 max 的值不变；

步骤 5：输出 max 的值。

用自然语言表示算法方便、通俗，但文字冗长，容易出现“歧义性”，不方便表示选择结构和循环结构的算法。因此，除特别简单的问题外，一般不用自然语言表示算法。

（2）用流程图表示算法。

用流程图表示算法，就是用一些图框和方向线来表示算法的图形表示法。美国国家标准化组织（ANSI）规定了一些常用的流程图符号，用这些不同的图框符代表不同的操作，如图

1.2 所示。

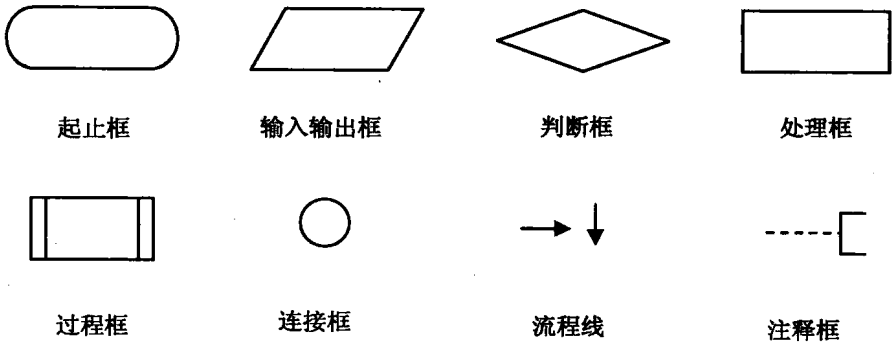

图 1.2　流程图图框

若将【例 1.1】算法用流程图表示，如图 1.3 所示。

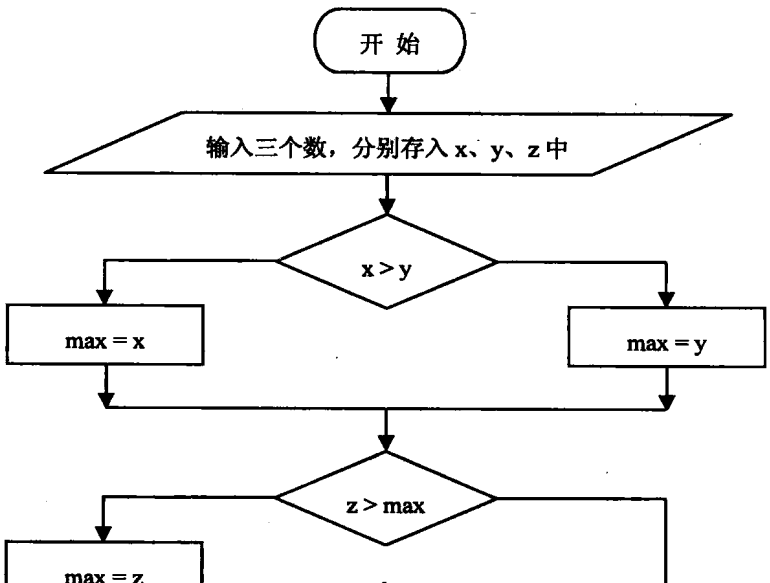

图 1.3　用流程图表示算法