高职高专计算机系列规划教材

C语言程序设计

路俊维　马雪松　主　编
吴丽丽　副主编
褚建立　主　审

C
PROGRAMMING
LANGUAGE

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高职高专计算机系列规划教材

C 语言程序设计

路俊维　马雪松　主　编
吴丽丽　副主编
褚建立　主　审

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书根据目前高职高专学生的学习模式结合讲义而编写，以任务驱动为导向，以培养学生C语言应用能力为主线，强调理论教学与实训密切结合。本书共分11章，系统地介绍了C语言的运行环境Visual C++ 6.0的编程环境、算法设计、数据类型、结构化程序设计、模块化程序设计、构造类型程序设计和文件系统的基本操作等，并辅以大量的习题，强化重点知识。

本书取材新颖，语言简洁流畅，举例通俗易懂，适用性强，适合作为高等职业院校、成人高校的编程入门教材，也可作为计算机培训机构的培训教材或其他从事计算机程序设计人员的参考书。

图书在版编目（CIP）数据

C语言程序设计/路俊维，马雪松主编. —北京：中国铁道出版社，2009.4
（高职高专计算机系列规划教材）
ISBN 978-7-113-09917-6

Ⅰ.C… Ⅱ.①路…②马… Ⅲ.C语言—程序设计—高等学校：技术学校—教材 Ⅳ.TP312

中国版本图书馆CIP数据核字（2009）第055735号

书　　名：C语言程序设计
作　　者：路俊维　马雪松　主编

策划编辑：秦绪好　王春霞
责任编辑：王占清　　**编辑部电话**：（010）63583215
编辑助理：何红艳
封面设计：付　巍　　**封面制作**：白　雪
版式设计：郑少云　　**责任印制**：李　佳

出版发行：中国铁道出版社（北京市宣武区右安门西街8号　邮政编码：100054）
印　　刷：北京新魏印刷厂
版　　次：2009年6月第1版　　2009年6月第1次印刷
开　　本：787mm×1092mm　1/16　**印张**：14.5　**字数**：334千
印　　数：4 000册
书　　号：ISBN 978-7-113-09917-6/TP·3223
定　　价：23.00元

前　言

C 语言是应用广泛、最具影响力的程序设计语言之一。它数据类型丰富，表达能力强，使用灵活方便，程序执行效率高，可移植性好，是一种理想的结构化程序设计语言。因此，C 语言不仅成为当前开发软件的主流语言，也是全国各高校计算机专业的首选语言。

本书共分 11 章，系统地介绍了 C 语言的运行环境 Visual C++ 6.0 的编程环境、算法设计、数据类型、结构化程序设计、模块化程序设计、构造类型程序设计和文件系统的基本操作等。本书在结构方面遵循深入浅出，由简到繁，台阶式地向前推进的原则，重点让学生了解 C 语言程序设计的基本知识，掌握常用算法和基本程序设计方法，具有应用 C 语言解决实际问题的基本能力。

本书由几位具有丰富教学经验的一线教师根据目前高职高专学生的学习模式结合讲义编写而成。本书的组织注重学生学习的连贯性和渐进性，层次分明，重点突出。书中所有程序都按照结构化程序设计方法编写，以有助于培养学生良好的编程习惯，例题均可在 Visual C++ 6.0 的编程环境中运行。

本书由路俊维和马雪松任主编，吴丽丽任副主编。第 1、2、10 章由路俊维编写，第 9 章及附录由马雪松编写，第 5 章由赵美枝编写，第 3、7 章由霍艳玲编写，第 4、8 章由吴丽丽编写，第 6 章由王彤编写，习题部分由张小志、董会国、张静编写。全书的统稿工作由路俊维、马雪松、吴丽丽负责完成。

在本书的编写过程中，褚建立教授审阅了全书，提出了许多宝贵的意见和建议，北京东方正通科技有限公司的任务经理张勇峰也对本书提出了许多宝贵的意见和建议，在此表示衷心的感谢！同时在本书的编写过程中还参阅了大量的网上资源和其他参考文献，在此对这些资源的提供者和作者一并表示感谢。

由于编者水平和经验有限，书中难免有不足之处，恳请读者提出宝贵意见和建议。

编　者

2009 年 5 月

目　录

第 1 章　C 语言概述

任务

在 Visual C++ 6.0 开发环境中创建、运行 C 源程序。

任务目的

C 语言是计算机编程最重要的语言之一，Visual C++ 6.0 是一个非常好的 C 语言集成开发环境（IDE）。通过使用 Visual C ++ 6.0 对该任务的实现，可使学生熟练掌握 C 语言程序框架结构、Visual C ++6.0 集成开发环境，熟练编辑、调试和编译 C 程序。

任务所需主要相关知识

- C 程序的结构。
- C 程序的执行过程。
- C 程序的上机步骤。
- 编译预处理。

1.1　C 语言的产生和发展

C 语言是目前流行的一种语言，它既可作为系统软件的描述语言，也可用来开发应用软件。

C 语言是在 20 世纪 70 年代初问世的。1978 年，由美国电话电报公司(AT&T)贝尔实验室正式发表了 C 语言，同时由 B.W.Kernighan 和 D.M.Ritchit 合著了著名的 *THE C PROGRAMMING LANGUAGE* 一书，通常简称 *K&R*，也有人称为 *K&R* 标准。但是，在 *K&R* 中并没有定义一个完整的标准 C 语言，后来，美国国家标准学会（American National Standards Institute）在此基础上制定了一个 C 语言标准，于 1983 年发表，通常称为 ANSI C。

1.2　C 语言的特点

C 语言发展如此迅速，而且成为最受欢迎的语言之一，主要原因是它具有强大的功能。许多著名的系统软件，如 DBASEⅢPLUS、DBASE Ⅳ都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序，就更能显示 C 语言的优势，如 PC-DOS、WORDSTAR 等就是用这种方法编写的。归纳起来，C 语言具有下列特点：

1. C 是中级语言

它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。C 语言可以像汇编语言一样

对位、字节和地址进行操作，这是一般高级语言，如 Pascal、FORTRAN、BASIC 等所不具备的。

2. C 是结构式语言

结构式语言的显著特点是代码及数据的分隔化，即程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰，便于使用、维护以及调试。C 语言是以函数形式提供给用户的，这些函数可方便地调用，并具有多种循环、条件语句控制程序流向，从而使程序完全结构化。

3. C 语言功能齐全

C 语言具有各种各样的数据类型，并引入了指针概念，可使程序效率更高。另外，C 语言也具有强大的图形功能，支持多种显示器和驱动器，而且计算功能、逻辑判断功能也比较强大，可以实现决策目的。

4. C 语言适用范围广

C 语言还有一个突出的优点是它适合于多种操作系统，如 DOS、UNIX，也适用于多种机型。但是，C 语言对程序员要求也高，程序员用 C 语言写程序会感到限制少、灵活性大、功能强。

1.3 C 语言程序的结构

用 C 语言编写的源程序，简称 C 程序。C 程序是一种函数结构，一般由一个或若干个函数组成（所谓函数是具有小功能的程序片断，具体内容将在第 7 章介绍），其中必有一个名为 main() 的函数，程序的执行是从 main()函数开始的。

为了说明 C 语言源程序结构的特点，先看下面几个程序，它们由简到难，表现了 C 语言源程序在组成结构上的特点。虽然有关内容还未介绍，但可从这些例子中了解到组成一个 C 源程序的基本部分和书写格式。

【例 1.1】 在屏幕上输出一行文本信息“**欢迎光临**”。

【程序代码】

```
#include "stdio.h"
void main()                              /*主函数*/
  {
    printf("**欢迎光临**\n");           /*在屏幕上输出"**欢迎光临**"*/
  }
```

运行结果如图 1-1 所示。

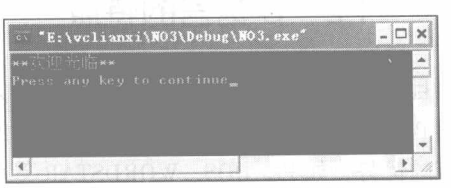


图 1-1　例 1.1 运行结果

注意： 图 1-1 中“E:\vclianxi\NO 3\Debug\NO 3.exe”为本书统一选择的路径，读者可根据自己的习惯选择其他不同的路径，以后类似问题不再重复说明。

【程序说明】

① 程序的功能是在屏幕上输出“**欢迎光临**”。

② include 称为文件包含命令，扩展名为.h 的文件称为头文件。

③ main()为主函数名。每个 C 程序都必须有一个 main()函数，且只能有一个主函数（main()函数），它是 C 程序执行的入口地址。

④ “{}”是函数的界定符，位于“{}”中的内容为函数体，每个函数都必须用一对“{}”将函数体括起来。

⑤ printf()输出语句是系统提供的库函数。

⑥ 每个语句后面有一个分号“;”。

⑦ 位于“/*...*/”之间的内容是注释语句。

程序的功能是从键盘输入“**欢迎光临**”，然后输出结果。在 main()函数之前的两行称为预处理命令（详见后面）。预处理命令还有其他几种，这里的 include 称为文件包含命令，其意义是把尖括号<>或引号""内指定的文件包含到本程序中来，成为本程序的一部分，被包含的文件通常是由系统提供的，其扩展名为.h，因此也称为头文件或首部文件。C 语言的头文件中包括了各个标准库函数的函数原型，因此凡是在程序中调用一个库函数时，都必须包含该函数原型所在的头文件。在本例中，使用了标准输出函数 printf()，该函数在头文件 stdio.h 库中，所以使用该函数时在主函数前用 include 命令包含了 stdio.h 文件。

【例 1.2】计算两个整数的平均值，并输出结果。

【程序代码】

```
#include <stdio.h>
void main()
{
   int x,y;                          /*变量定义*/
   float ave;
   x=46;                             /*为变量 x 赋值*/
   y=78;                             /*为变量 y 赋值*/
   ave=(x+y)/2.0;                    /*计算平均值存放在 ave 变量中*/
   printf("平均值是%0.1f",ave);      /*在屏幕上输出平均值*/
}
```

运行结果如图 1-2 所示。

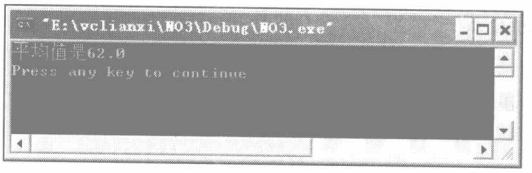


图 1-2　例 1.2 运行结果

【程序说明】

① #include 是编译预处理命令，放在源程序的最前面，编译预处理语句后面不加分号。

② 程序中变量在使用前要定义，定义变量的语句放在可执行语句之前。

③ 函数 printf()是系统提供的函数，在将“平均值是%0.1f”输出时，“%0.1f”由 ave 的值取代。

④ C 程序中除了可用库函数外还可使用程序员自行编写的用户函数。

⑤ C 程序的书写格式自由，一行可以写一条或几条语句，一条语句也可以写在多行上。C 程序没有行号，每条语句和变量定义必须用一个分号结尾。

在例题中的主函数体又分为两部分：一部分为说明部分，另一部为分执行部分。说明部分是指对变量的类型说明和变量名称的定义。例 1.1 中未使用任何变量，因此无说明部分。C 语言规定，源程序中所有用到的变量都必须先说明后使用，否则将会出错。这一点是编译型高级程序设计语言的一个特点，与解释型的 BASIC 语言是不同的，说明部分是 C 源程序结构中很重要的组成部分。本例中使用了两个变量 x、y，这两个变量的类型都是整型，故用类型说明符 int 来说明这两个变量。说明部分后的四行为执行部分或称为执行语句部分，用以完成程序的功能。执行部分的第一、二行是赋值语句，给变量赋初值，第三行是计算平均值并把平均值存放到 ave 变量中，第四行是用 printf()函数输出变量 ave 的值，即输出平均值，程序结束。

C 源程序结构的一般形式如下：

```
    预处理命令序列
    void main()
{
    变量定义序列
    执行语句序列
}
```

其中：

① 预处理命令序列：书写程序相关的预处理文件。

② 变量定义序列：是声名部分，用来定义程序中所用到的变量。

③ 执行语句序列：是程序的执行部分，由若干语句组成，完成对数据的运算及各种处理。

以上这三个序列称为 C 程序结构上的三大区域，这三大区域在程序中不可调换位置，程序也将按这个顺序执行。

1.4 算　　法

1.4.1 程序与算法

程序通常指为了让计算机完成特定任务而设计的有序指令的集合。一个程序应包括：对数据的描述，在程序中要指定数据的类型和数据的组织形式，即数据结构（data structure）；对操作的描述，即操作步骤，也就是算法（algorithm）。

Nikiklaus Wirth 提出的公式：

算法+数据结构=程序

解决一个问题的方法和步骤称为算法。

1.4.2 算法的特性

- 有穷性：一个算法的操作步骤应是有限的，而不能是无限的。
- 确定性：算法中每一个步骤应当是确定的，而不能应当是含糊的、模棱两可的。
- 有零个或多个输入。
- 有一个或多个输出。
- 有效性：算法中每一个步骤应当能有效地执行，并得到确定的结果。

程序设计人员必须会设计算法，并根据算法写出程序。

【例 1.3】对一个大于 2 的正整数，判断它是否是一个素数。

实例分析：所谓素数是指除 1 和该数本身外，不能被其他任何整数整除的数，判断一个数 n 是否为素数的方法：将 n 作为被除数，用 2～（n-1）各个整数轮流作除数，如果都不能整除 n，则 n 为素数。

设这个数是 n，按如下步骤判断 n 是否为素数：

① 输入 n 的值；

② 设 i=2；

③ n 除以 i，求得余数 r；

④ 判断：如果 r=0，表示 n 能被 i 整除，则 n 不是素数，算法结束；否则，执行步骤⑤；

⑤ i+1 赋值给 i；

⑥ 如果 i<n，返回步骤③；否则打印 n 是素数，算法结束。

实际上，n 不必除以 2～（n-1）之间的全部整数，只须除以 2～sqrt(n-1)之间的整数即可。例如，判断 17 是否是素数，只须将 17 除以 2、3、4 即可，如果除不尽，n 必为素数。因此步骤⑥可改为：如果 i≤sqrt(n)，返回步骤③；否则，打印 n 是素数，算法结束。

从以上两个实例看出，对待问题，首先要分析题目，然后寻找一种实现问题的方法，这种问题的具体化就是算法。

1.4.3　算法的表示

算法的表示方法很多，常见的有自然语言、传统流程图、N-S 图、伪代码等。

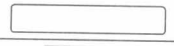
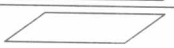
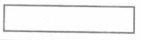
1．用自然语言表示算法

自然语言就是人们日常使用的语言。自然语言表示的算法通俗易懂，但篇幅较长，表达上容易引起理解方面的“歧义性”，所以除了很简单的问题，一般不用自然语言表示算法。

2．传统流程图表示算法

传统流程图是用一组规定的图形符号、流程线和文字说明来表示各种操作、方法，它直观形象，易于理解。ANSI 规定了一些常用流程图符号，如表 1-1 所示。

表 1-1　传统流程图符号

符　号	符号名称	含　义
	起止框	表示算法的开始和结束
	输入/输出框	输入/输出操作
	判断框	对框内的条件进行判断
	处理框	对框内的内容进行处理
	流程线	表示流程的方向
	连接点	通常用于换页处，表示两个具有同一标记的“连接点”应连接在一起

如例 1.3，对一个大于 2 的正整数，判断它是否是一个素数的算法流程如图 1-3 所示。

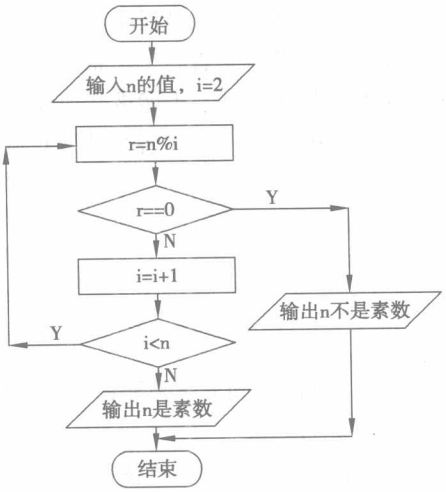


图 1-3　例 1.3 的算法流程图

3. 用 N-S 流程图表示算法

针对传统流程图存在的问题，1973 年，美国学者提出了一种新型流程图：N-S 流程图，其主要特点是完全取消了流程线，不允许随意出现控制流，全部算法写在矩形框内，该矩形框以三种基本结构（顺序、选择、循环）描述符号为基础复合而成（见图 1-4、图 1-5、图 1-6）。

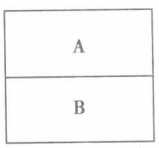


图 1-4　顺序结构

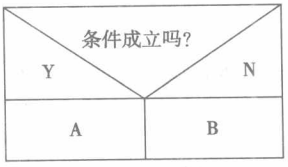


图 1-5　选择结构

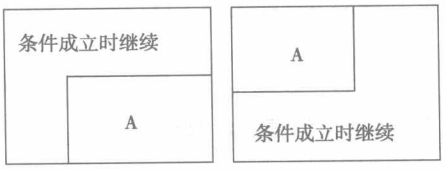


图 1-6　循环机构

4. 伪代码描述算法

伪代码是一种介于自然语言和程序设计语言之间的文字和符号，用来描述算法。伪代码的表现形式比较灵活，没有严格的语法格式。

【例 1.4】用伪代码描述找三个数中最大数的算法。

```
INPUT num1,num2,num3
Num1->Max
If num2>Max     THEN num2->max
If num3>Max     THEN num3->max
PRINT Max
```

1.5　程序设计的步骤

程序设计就是针对给定问题进行设计、编写代码、调试代码的过程。因此，程序设计者必须先充分了解问题才能编写出合适的应用程序。程序设计的一般步骤如下：

1. 分析问题

首先，根据问题的具体要求进行需求分析，对现有的信息加以整理，然后在分析的基础上，将实际问题抽象化，建立相应的数学模型并确定解决方案。

2．设计算法

根据建立的数学模型和确定的方案，详细规划解决问题的步骤。若是简单的问题，绘制流程图即可；若是比较复杂的问题，则采用伪代码或其他算法描述。

3．编写程序

根据确定的算法，选用合适的程序设计语言，将算法按所选语言的规则描述出来，即形成源程序。编写程序时尽量以模块与对象方式来编写；同时，切记在程序中加上必要的注释。因为一个较复杂的程序若没有注释，不但别人看不懂，时间长了可能自己也忘了当初为什么这么写。

4．调试运行与维护程序

编写好的程序须进行验证、测试（testing）、调试（debugging）与维护（maintainance）。各种条件下的数据都要输入，不论条件成立与否都要执行，以便证实程序正确无误。若发现问题即对程序进行修改，然后再运行和检验，直到得出正确结果。

1.6　C 语言源程序的上机步骤

前面已经看到了一些用 C 语言编写的程序，但它们是不能直接运行的。因为计算机只能识别和执行由 0 和 1 组成的二进制指令，而不能识别和执行用高级语言写的程序。为了使计算机能执行高级语言所写的程序，必须先用一种称为“编译程序”的软件，把程序翻译成二进制形式的“目标程序”，然后将该目标程序与系统的函数库和其他目标程序连接起来，形成可执行的目标程序才能被机器所执行。

1.6.1　C 语言源程序的执行过程

C 语言源程序编译、连接生成可执行程序（*.exe 文件），一般须经过四个步骤：编辑、编译和调试、连接、运行，具体步骤如图 1-7 所示。

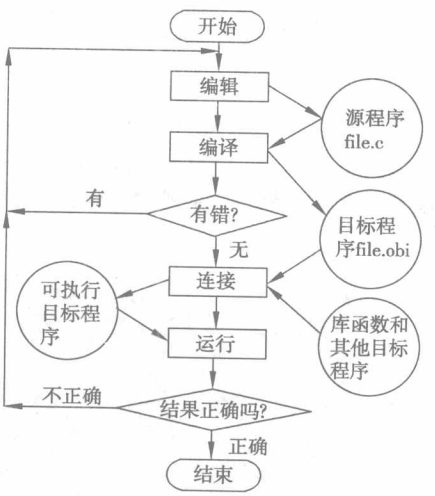


图 1-7　C 语言源程序的执行流程图

1．编辑

完成代码的基本输入和编辑工作，该过程的使用方法和普通文件编辑软件的用法相同。

2. 编译与调试

完成程序的语法和逻辑错误的检查与修改。该过程是一个反复的过程，即先通过语法检查，如果发现错误，则返回到编辑过程进行修改，修改完成后，再进行语法检查，直到没有语法错误为止。逻辑错误只有到连接过程和运行过程才能发现。

3. 连接

将目标程序与系统文件进行连接，生成可执行程序。如果在这个过程中出现错误，返回编辑状态进行修改，然后重复上述步骤，直到没有错误为止。

4. 运行

让计算机运行编写的程序，如果运行的结果和期望的不一样，就须返回到开始进行检查、修改，重复上面的过程，直到运行出正确的结果为止。

1.6.2 Visual C++ 6.0 开发环境

Visual C++ 6.0 是 Microsoft 公司推出的基于 Windows 的 C/C++开发工具，它是 Microsoft Visual Studio 套装软件的一个有机组成部分。

Visual C++ 6.0 需要一台运行 Windows 98/2000/XP 的计算机为工作平台，要求有足够的内存和其他资源支持各种工具，要求硬件的最低配置应该是 Pentium 166MHz、64MB 内存，至少 1GB 的硬盘空间。从 Microsoft Visual Studio 的光盘中运行 Visual C++ 6.0 安装程序（Setup.exe），完成之后，就可以通过“开始”菜单运行 Visual C++ 6.0。

Visual C++ 6.0 软件包包含了许多独立的组件，如编辑器、编译器、连接器、生成实用程序、调试器以及各种各样的开发工具。Microsoft Visual Studio 把 Visual C++ 6.0 的所有工具结合在一起，集成为一个整体，通过一个由窗口、对话框、菜单、工具栏、快捷键和宏组成的开发环境。

1.6.3 使用 Visual C++ 6.0

本节主要介绍使用 Visual C++ 6.0 集成开发环境来编写 C 语言程序。

1. 启动 Visual C++ 6.0

在 Windows 环境下，选择“开始”|“程序”|Microsoft Visual Studio 6.0|Microsoft Visual C++ 6.0 命令，即可启动 Visual C++ 6.0，如图 1-8 所示。用户界面常称做 Visual C++ 6.0 集成开发环境（IDE）。

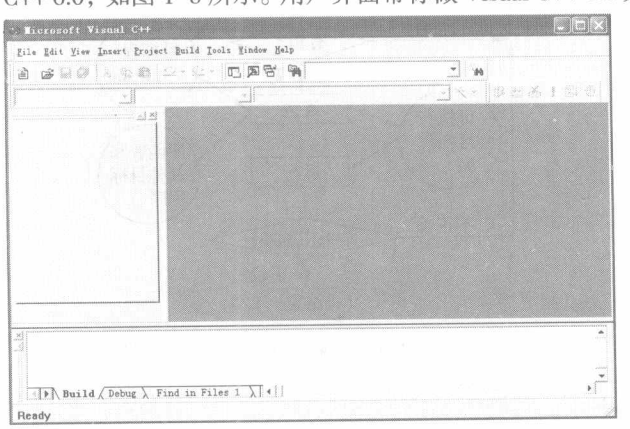


图 1-8 Visual C++6.0 用户界面