全国高等院校计算机基础教育“十三五”规划教材

C语言程序设计

C YUYAN CHENGXU SHEJI

侯向丹　李　智　主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

全国高等院校计算机基础教育“十三五”规划教材

C语言程序设计

主　编　侯向丹　李　智

副主编　彭玉青　于丽梅　袁玉倩　李艳萍

于东敏　肖国玺　刘　明

中国铁道出版社

CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书全面地讲述了C语言程序设计的基础知识、方法和技巧，主要包括绪论，数据类型、运算符和表达式，程序设计基础，程序控制结构，函数，编译预处理指令，数组，指针，结构、联合、枚举和自定义类型，文件及I/O函数。本书对每个知识点都进行了详细的介绍，配有相应的例题、小思考、视频讲解等，指出编写程序时易犯的错误，引导学生举一反三，进行多方位思考。本书配有大量的习题，能让学生很好地掌握C语言的基本知识、编程方法和技巧。

本书实用性强，内容丰富，难易适中，适合作为高等院校学生学习C语言程序设计的教材，也可以作为C语言开发用户的参考用书。

图书在版编目（CIP）数据

C语言程序设计/侯向丹,李智主编. —北京:中国铁道出版社，2019.1

全国高等院校计算机基础教育“十三五”规划教材

ISBN 978-7-113-25319-6

Ⅰ.①C… Ⅱ.①侯… ②李… 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294782号

书　　名：C语言程序设计

作　　者：侯向丹　李　智　主编

策　　划：张　彤　魏　娜　　**读者热线：**（010）63550836

责任编辑：贾　星　徐盼欣

封面设计：刘　颖

责任校对：张玉华

责任印制：郭向伟

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.tdpress.com/51eds/

印　　刷：北京铭成印刷有限公司

版　　次：2019年1月第1版　2019年1月第1次印刷

开　　本：787 mm×1 092 mm　1/16　印张：13　字数：310千

书　　号：ISBN 978-7-113-25319-6

定　　价：36.00元

前　言

随着教学改革的不断深入，C 语言程序设计作为计算机专业学生接触的第一门语言类课程，在课程体系、教学内容、教学手段和教学模式等方面都有了新的变化。为了适应这种形势的需求，我们组织了长期从事计算机程序设计课程教学、具有丰富实践经验的教师编写本书，供广大高校师生选用。

本书力求由简到繁、深入浅出地讲述程序设计的基础知识、方法和技巧等。本书概念清楚、例题丰富、易于理解、实用性强，将讲授的知识点融入典型例题中，并配有相应的习题，其中部分知识点配备了微视频讲解，可以很好地帮助学生理解概念、开拓思路、提高编程能力。书中完整的程序代码均在 VC6.0 环境中调试通过，并在书中插入了运行结果示意图供学生参考。另外，在附录中还配了四套模拟测试卷（附参考答案），以及 ASCII 码表。

无论读者以前是否接触过程序设计语言，通过本书的学习均可以顺利进入计算机编程世界。

本书由河北工业大学侯向丹、李智任主编，河北工业大学彭玉青、于丽梅、袁玉倩、李艳萍、于东敏、肖国玺、刘明任副主编。具体编写分工如下：第 1、2 章由李智、李艳萍编写，第 3、4、5 章由侯向丹、刘明编写，第 6、7、8 章由彭玉青、于丽梅、于东敏编写，第 9、10 章由肖国玺、袁玉倩编写。全书最后由侯向丹统编定稿。

在本书编写过程中，得到了河北工业大学人工智能与数据科学学院和廊坊分院很多老师的支持和帮助，在此一并表示最真诚的谢意。

由于编者水平有限，书中难免有不妥和疏漏之处，恳请各位专家和广大读者批评指正。

编　者

2018 年 10 月

目　录

第 1 章

绪　论

本章首先介绍 C 语言的发展和特点，并通过简单的实例，介绍包含注释、编译预处理和程序主体在内的 C 语言程序基本结构，最后对在 Visual C++ 6.0（以下简称 VC6）集成开发环境下 C 语言程序开发的四个基本阶段进行了详细说明。

1.1　C 语言的发展和特点

1.1.1　C 语言的发展

C 语言的原型是 ALGOL 60 语言。1963 年，英国的剑桥大学将 ALGOL 60 语言发展为 CPL 语言；1967 年，英国剑桥大学的 Martin Richards 对 CPL 语言进行简化，推出没有类型的 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，开发了简单并且很接近于硬件的 B 语言，并用 B 语言实现了第一个 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，并取了 BCPL 的第二个字母作为这种语言的名字，这就是 C 语言。1973 年，Ken Thompson 和 D. M. Ritchie 合作把 UNIX 操作系统用 C 语言重新写了一遍。随着 UNIX 的发展，C 语言自身也在不断地完善。1977 年，D. M. Ritchie 发表了不依赖于具体机器系统的 C 语言编译文本《可移植的 C 语言编译程序》。

1982 年，很多有识之士和美国国家标准学会（American National Standards Institute，ANSI）为了使 C 语言健康发展，决定成立 C 标准委员会，建立 C 语言的标准。1989 年，ANSI 发布了第一个完整的 C 语言标准——ANSI X3.159—1989，简称 C89，人们习惯称其为 ANSI C。1990 年，C89 被国际标准组织（International Standard Organization，ISO）采纳，称为 ISO/IEC9899：1990，通常简称 C90。1999 年，ISO 发布了新的 C 语言标准，称为 ISO/IEC 9899：1999，简称 C99。2011 年 12 月 8 日，ISO 正式发布了新的标准，称为 ISO/IEC9899：2011，简称 C11。

目前，C 语言已经广泛应用于微、小、大、巨型机上，成为从系统程序设计到工程应用程序等都广泛使用的一种高级程序设计语言。

1.1.2 C语言的特点

C语言的特点可从多方面来阐述，大体可归纳为以下几点。

（1）表达能力强且灵活。C语言是面向结构的程序设计语言，通用性好，不局限于某种机型，它足以取代汇编语言来编写各种系统软件和应用软件。C语言是处于汇编语言和高级语言之间的一种记述性程序设计语言，因此被称为中级语言。但这并不意味着C语言的功能差、难以使用，而是说它把高级语言的基本结构与低级语言的实用性结合了起来，既有高级语言面向用户、容易记忆、便于阅读和书写的特点，又有像汇编语言那样面向硬件和系统、可以直接访问硬件的功能。

（2）程序结构清晰且紧凑。C语言的主要结构成分是函数——C语言中独立的子程序。在C语言中，函数是基本的结构模块，各函数将整体程序分割成若干相对独立的功能模块，完成各自独立的任务。可以建立独立的函数这一点在大型软件工程中是十分关键的，只有这样才能避免不同编程人员的程序由于偶然因素而互相干扰。并且，它为程序模块间的相互调用以及数据传递提供了方便，使得模块化结构的程序不但清晰而且紧凑。

（3）C语言是一种结构化程序设计语言。它提供了一整套循环、条件判断等语句，实现了对程序逻辑流程的有效控制，有利于结构化程序设计。

（4）数据类型和运算符丰富。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类型运算。其中，指针是与机器内存地址相关的说明项，因此指针能让程序员以与机器码相同的形式存取内存的数据。正确地使用指针可有效提高程序的效率。C语言的运算符也很丰富，共有34种，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。

（5）目标程序的质量高，可移植性好。C语言中绝大多数运算符与一般机器指令一致，可直接翻译成机器代码，因此用它编写程序生成的代码质量高。实践证明，其他高级语言相对汇编语言的代码效率要低得多，而C语言的代码效率只比汇编语言低10%~20%，但C语言在描述问题时编程速度快、可读性好、表达能力强等优点是汇编语言无法相比的。C语言虽然具有直接访问硬件的功能，但C语言程序本身并不依存于硬件系统，从而便于在硬件结构不同的机种间实现程序的移植。

C语言有许多优点，也存在一些不足，如运算符优先级太多，不便于记忆；类型检验太弱，增加了不安全因素等。

尽管C语言存在不足，但作为一种实用的通用程序设计语言迅速得到广泛普及和应用，特别是在微处理机和微型计算机的软件开发及各种软件工具的开发中，使用C语言的趋势日益增强。

1.2 C语言程序的基本结构及书写格式

1.2.1 C语言程序的基本结构

C语言程序由一个或几个函数组成。下面通过一个简单的C语言程序示例分析C程序的结构特性。

【例1.1】在屏幕上输出文字“Hello World!”。

源程序清单：

```
/*Program 1.1 输出字符串“Hello World!”*/
#include <stdio.h>
void main()
{
   printf("Hello World!\n");    //输出语句
}
```

运行结果示意图如图 1.1 所示。

这是一个最简单的 C 语言源程序，作用是将字符串“Hello World!”显示在屏幕上。下面以本程序为例，分析一下 C 语言的程序结构。C 语言的程序结构主要分为注释部分、编译预处理部分和程序主体部分等三部分。

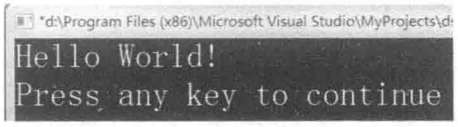

图 1.1 例 1.1 的程序运行结果示意图

1. 注释部分

C 语言的注释格式有两种。第一种注释格式如例 1.1 中的第一行所示，即

```
/*Program 1.1 输出字符串“Hello World!”*/
```

这行代码是注释部分，是用“/*”和“*/”括起来的内容，可以跨越一行或者多行，“/”和“*”之间、“*”和“/”之间不能有空格，否则会出错。

第二种注释格式用“//”表示注释，例如语句

```
printf("Hello World!\n");   //输出语句
```

其中的“//输出语句”即为注释部分。只能注释一行。

注释的作用是给程序设计人员一种提示或记号，主要目的是提高程序的可读性。一旦编译器在源文件中发现了注释部分，就会忽略注释，所以注释并不参与程序的运行。

注释一般出现在两个地方：一是程序的开头，说明程序或者模块的名称、用途等信息，称为序言型注释；二是注解性注释，用于程序的语句中，对难懂的地方加以说明。

好的程序员应该养成用注释给程序做注解的习惯，因为在编写较复杂的程序时，可能会忘记某些代码是做什么的或者它们是如何运行的。加入足够的注释，可以确保自己（或其他程序员）在很长时间之后也可以理解该程序的目的以及其编写思路。

2. 编译预处理部分

例如，例 1.1 的程序第二行“#include <stdio.h>”，其中符号“#”说明这是一个预处理命令，编译器将在编译过程开始前的初始化预处理阶段处理这些命令。预处理命令通常出现在源程序的开头。

在例 1.1 中，“#include <stdio.h>”的作用是把头文件 stdio.h 中的内容加入到源程序中，stdio.h 中定义了一些关于标准 C 语言库提供的有关输入/输出的函数信息。

C 语言本身没有输入/输出语句，在本程序中，由于后面使用到了 printf()函数，而 printf()函数的原型声明在头文件 stdio.h 中，所以要将它包含进来。

3. 程序主体部分

```
void main()
{
    printf("Hello World!\n");    //输出语句
}
```

这条语句是例 1.1 程序的主体部分，定义了 main()函数，main()函数又称主函数。C 语言源程序是一种函数式的结构，所谓函数可以理解为一个语句组执行一组特定的功能。C 语言源程序由一个或者多个函数构成，每个 C 语言程序有且仅有一个主函数。程序从主函数的开头开始执行，最终在主函数结束。下面对主函数的结构进行分析。主函数是由函数头部和函数体构成的。

（1）函数头部。例 1.1 中主函数第 1 行定义如下：void main()。这里定义了主函数的头部，包括函数类型和函数名。

注意：这行代码的结尾没有分号，main 后面的小括号不能省略。

关键字 void 是系统定义的关键字，表示空类型，说明了该函数的返回值类型，即 main( )函数不返回任何值。关于 void 的介绍会在后面的第 5 章中出现。

（2）函数体。主函数的第 2 行和第 4 行中的{}中间的内容是函数体部分。包含了函数的所有具体语句。函数体可以包含一条或多条语句，也可以不包含任何语句。当不包含任何语句时，这种函数什么都不做。

① 输出语句。主函数的第 3 行是输出语句“printf("Hello World!\n");”，printf()函数是 C 语言提供的标准输出函数，其功能是在屏幕上将双引号之间的内容原样输出，“\n”是转义字符，它的作用是将光标移到下一行的开始处。

② C 语言语句组成。语句是组成程序的基本单元，主要包含表达式语句、控制语句、空语句等。每条语句最后要用分号“;”结尾，分号是语句的组成部分。C 程序中的语句含有各种运算符、关键字、表达式、函数调用等。例如，“printf("Hello World!\n");”就是函数调用语句。

1.2.2 C 语言程序的书写格式

（1）用 C 语言书写程序时较为自由，既可以一行写多条语句，也可以一条语句分几行来写，每条语句以“;”结尾。

（2）C 语言要求关键字使用小写字母。C 编译程序区分大小写。如 A 和 a 为两个不同的标识符。

（3）尽管 C 的书写格式比较自由，但为避免程序书写的层次混乱不清，增强程序的可读性，一般采用有一定格式的习惯写法。为了结构层次分明，书写程序时不同结构层次的语句从不同的起始位置开始，同一结构层次中的语句缩进同样个数的字符位置。同一结构层次中的花括号对{ }也缩进同样个数的字符位置。例如，例 1.1 中 printf()调用语句向内缩进一定的字符位置。

（4）一般一条语句占一行位置。为增加可读性，可适当加一些注释行或空行。

（5）程序中所有的符号均为英文半角符号。

请看如下 C 语言程序书写格式示例。

【例 1.2】在屏幕上显示 100 次 1~9 的值。

源程序清单：

```
#include <stdio.h>
void main()      //输出 100 次 1~9 的值
{
    int count,t;
    for(t=0;t<100;++t)
    {
        count=1;
        for(;;)
        {
            printf("%d",count);
            count++;
            if(count==10) break;
        }
    }
}
```

1.3 C 语言程序的开发过程

创建 C 语言程序分为编辑源程序、编译源文件、连接目标文件及库文件和运行程序等四个基本阶段，下面分别讲述每个阶段的作用。

1.3.1 编辑源程序

编辑源程序是创建和编辑 C 语言的源代码，源代码就是编写程序指令的总称。绝大部分 C 编译系统带有独立的编辑程序，可用于输入或修改源程序。也可以使用操作系统提供的编辑程序编辑源程序，如 UNIX 操作系统下的 ED、DOS 下的 EDLIN、Windows 下的 Write 等。

源程序经编辑程序由键盘输入后，生成源程序文件，以文本文件的形式存储到外部存储器（如硬盘）中，这个文本文件一般称为源文件。源文件的名字由用户选定，扩展名为 .c 或 .cpp。例如，filename.c、example.c、test.cpp 等均为 C 语言源程序的合法文件名。

1.3.2 编译源文件

在程序运行之前，需要用 C 语言的编译程序对源程序文件进行编译。由编译程序进行语法检查，若没有错误，则编译后生成目标文件。目标文件是由“目标代码”组成的，目标代码是指可在机器上直接运行的机器码。C 语言目标文件是扩展名为 .obj 的文件。

如果编译程序发现错误，编译结束后，会将所有的错误信息列出。双击该错误信息，光标会定位到可能出错的代码行，便于用户修改。一般编译系统给出的出错信息分为两种：一种是错误（error）；一种是警告（warning）。凡是检查出有 error 信息的程序不生成目标程序，必须改正后再次编译，直到无误后才能生成目标文件；如果编译仅有 warning 信息，则不影响生成目标程序。

1.3.3 连接目标文件及库文件

源程序经编译程序编译后产生的目标文件还不能在机器上直接运行，因为它只是一个浮动的程序模块，其中的机器码指令的内存地址没有绝对确定，因此程序需进行重定位以确定绝对内存地址，这项工作由连接程序来完成。

所有语言编译程序都是与标准函数库文件一起提供的，保存在函数库文件中的函数也都是可重定位的。当C语言程序调用了一个标准库函数或其他函数时，编译程序只记下它的名字，随后由连接程序将用户编写的程序产生的目标代码同它所调用函数的代码结合起来。经过连接后产生的文件为可执行文件，其扩展名为.exe，可以直接在机器上运行。

1.3.4 运行程序

经过编译和连接后产生的可执行文件，只需在操作系统提示符下输入可执行文件名即可执行程序。如果发现执行结果不对，依然要回到编辑阶段重新检查代码。

C语言程序开发过程如图1.2所示。

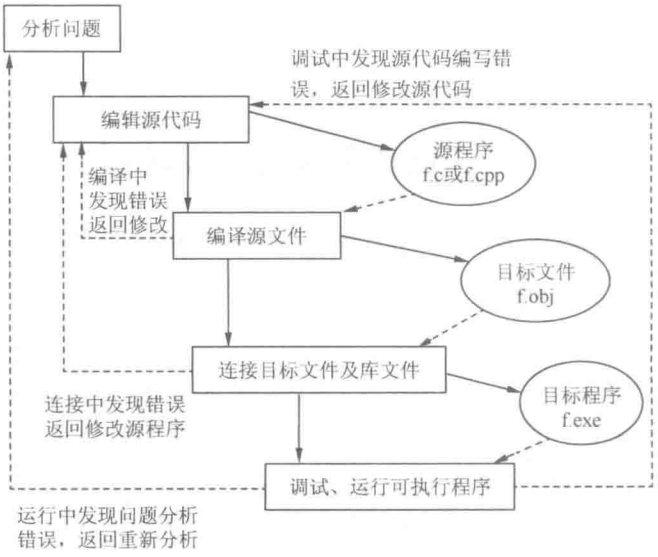

图1.2 C语言程序开发过程

1.4 C语言程序的开发环境

本书选用VC6作为开发环境。VC6是C/C++的集成开发环境。

1.4 VC6开发环境

1. 编辑源程序

编辑源程序的过程如下：

（1）启动VC6。成功地安装VC6以后，可以在Windows窗口下依次选择“开始”→“程序”→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0命令，启动VC6进入VC6的集成开发环境，如图1.3所示。

（2）建立一个新工程项目，如图1.4所示。选择Win32 Console Application选项，创建一个基于DOS平台的项目文件；在“位置”文本框中选择该工程项目所存放的位置，如“G:\学习\C”在“工程名称”文本框中输入该项目名，如myproject。然后单击“确定”按钮。

（3）创建源文件。刚刚创建的空白项目中没有任何文件，此时可选择“文件”菜单中的“新建”命令，会弹出“新建”对话框，如图1.5所示。

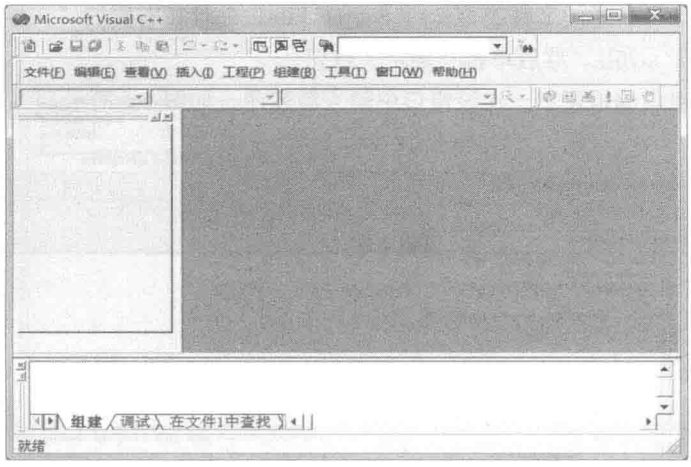

图 1.3 VC6 集成开发环境

图 1.4 创建新工程

图 1.5 创建新文件

在该对话框中选择新建的文件类型 C++ Source File，并且在右侧的“文件名”文本框中输入新建的源文件名，如 myfile。然后单击“确定”按钮。

此时会弹出源程序编辑窗口，在该窗口中输入源程序，如图 1.6 所示。

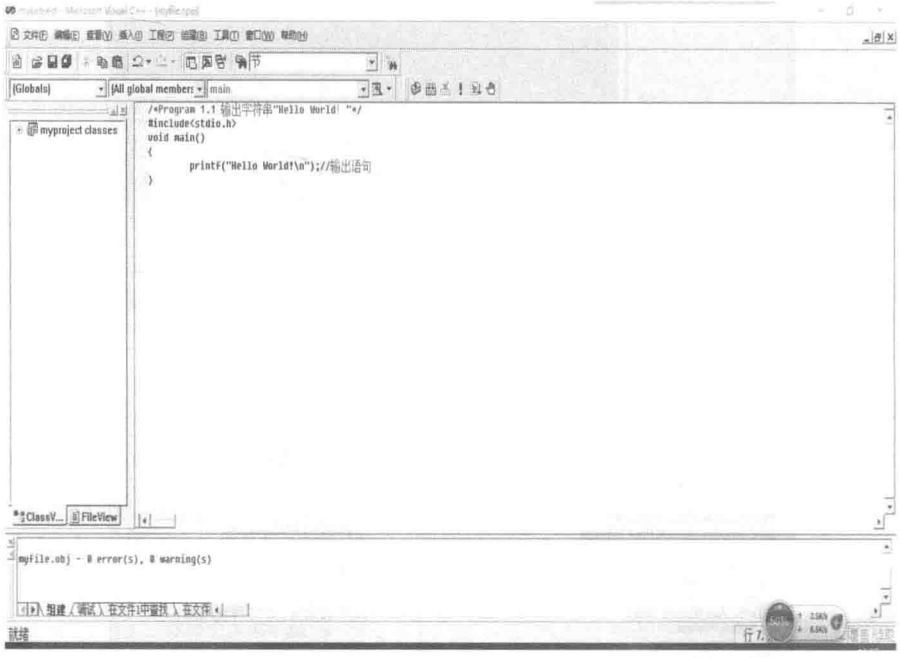

图 1.6　源程序编辑窗口

注意：在“文件名”文本框中输入程序名时，若要新建.c 文件则一定要加上.c 的扩展名，否则默认的是 C++的.cpp 文件。

（4）保存源文件。编辑完源文件，可以选择“文件”菜单中的“保存”命令， 保存源文件。

2. 编译源文件

选择“编译”菜单中的“编译”命令，或按快捷键 Ctrl+F7，编译该源文件，生成目标文件.obj 文件。

3. 连接目标文件并生成可执行文件

选择“编译”菜单中的“生成可执行文件”命令，或按快捷键 F7，可连接目标文件并生成可执行文件。

4. 调试方法

在编译或连接时，会出现 Output 窗口，该窗口显示系统在编译或连接程序时的信息，如图 1.7 所示。图 1.7 所示信息表示该源程序没有错误。若编译或连接时出现错误，则在该窗口中标识出错误文件名，发生错误的行号及错误的原因等信息，如图 1.8 所示。

在图 1.8 中，语句“printf("Hello World!\n")”漏写分号，出现错误。Output 窗口提示执行出错信息，并指出出错的位置。这时，不能生成可执行文件，必须在源程序中找出错误原因并修改源程序后，再次进行编译、连接，直至生成可执行文件。