HZ BOOKS
华章教育

普通高等教育“十一五”国家级规划教材

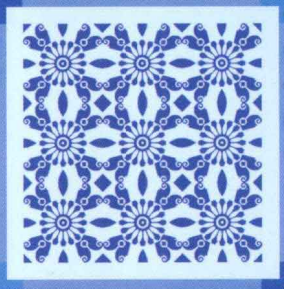
高等院校计算机教材系列

PROGRAMMING IN C

C语言程序设计

第2版

顾治华 陈天煌 孙珊珊 编著



机械工业出版社
China Machine Press

国家级规划教材

高等院校计算机教材系列

PROGRAMMING IN C
C语言程序设计

第2版

顾治华 陈天煌 孙珊珊 编著

机械工业出版社
China Machine Press

本书的写作融入了作者多年的教学经验，充分考虑到初学者的能力、认知水平、知识结构等因素，遵照循序渐进、由浅入深的原则，较系统地介绍C语言程序设计知识，内容涵盖算法及算法设计、数据描述与基本操作、选择结构程序设计、循环结构程序设计、数组、指针、函数与模块化程序设计、结构体和共用体、编译预处理、文件，并对常用程序设计方法及C++语言知识进行了简单介绍。

本书文字叙述通俗易懂，理论阐述简明科学，并辅以大量经典实用的实例和习题来加深读者对理论知识的理解，可作为高等院校理工科专业程序设计课程的教材，也适合程序设计初学者自学使用。

图书在版编目（CIP）数据

C语言程序设计 / 顾治华，陈天煌，孙珊珊编著. —2版. —北京：机械工业出版社，2012.4

（高等院校计算机教材系列）

ISBN 978-7-111-37462-6

Ⅰ. C…　Ⅱ. ① 顾…　② 陈…　③ 孙…　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（2012）第023170号

机械工业出版社（北京市西城区百万庄大街22号　邮政编码　100037）
责任编辑：刘立卿
三河市杨庄长鸣印刷装订厂印刷
2012年5月第2版第1次印刷
185mm×260mm · 20印张
标准书号：ISBN 978-7-111-37462-6
定价：35.00元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88378991，88361066
购书热线：（010）68326294，88379649，68995259
投稿热线：（010）88379604
读者信箱：hzjsj@hzbook.com

前　言

C语言是计算机学科的核心课程，也是其他理工科专业计算机基础训练的必修课。C语言功能丰富，表达能力强，使用灵活方便，应用面广，目标程序效率高，可移植性好，兼备高级程序语言和低级程序语言的诸多优点，所以在计算机工程实践中得到了广泛的应用。

在多年的教学过程中我们发现，许多学生在学完了“C语言程序设计”课程后，虽然能够了解和掌握一些语句的语法知识和语义，但不会应用语言来编写程序，把编程视为十分艰难而又高不可攀的工作。为帮助学生学会程序设计，我们总结多年的教学经验编写了本书。

本书在介绍C语言的同时，注重讲解如何应用C语言来编程，试图帮助读者克服畏难情绪，在轻松、愉快的气氛中探索程序设计的奥妙。学习编程技巧是一个不断实践、反复练习的过程。本书编写的目标就是希望帮助读者缩短这个过程，迅速提高C语言程序设计能力和水平。本书中的算法思维训练和编程思想同样适用于其他高级程序设计语言。

本书具有如下特点：

1）面向初学者，使略有计算机基础的人都能较容易地学会C语言编程。本书所采用的程序实例充满趣味性和实用性，语言叙述通俗易懂，难点分散，概念清晰，层次分明。

2）专门介绍了一些程序设计的常用方法，如穷举法、迭代法、递推法等，帮助读者提高程序设计能力和技巧。

3）注重各部分知识的综合应用训练，以提高程序设计能力为目标，并使读者在学习和掌握一门语言的同时养成良好的程序设计习惯。

4）习题中选用了部分等级考试试题，对读者参加相关考试也具有实用性。

5）为了帮助读者充分利用Internet上十分丰富的学习资源，本书在附录中提供了百余个中英文的关键词，由这些关键词可搜索到更多更新的C语言程序设计的文献资料。

本书在写作过程中与多名讲授该课程的教师进行过深入讨论，汲取了许多宝贵的教学经验。同时，本书的编写工作还得到了院系领导及机械工业出版社的大力支持。在此，一并表示衷心的感谢。

由于计算机科学技术发展迅速，加之编者水平有限，书中错误在所难免，恳请广大读者和同行批评指正，并多多提出宝贵意见。

作者于武汉理工大学

2012年1月

教学建议

教学章节	教学要求	课时
第1章 C语言程序设计概述	了解程序、程序设计、高级语言的概念，掌握C语言的字符集、词类，了解C语言程序的基本结构	2
第2章 算法及算法设计简介	了解算法的基本概念，掌握算法的设计方法和表式方式，理解并掌握结构化程序设计方法	4
第3章 数据描述与基本操作	了解C语言的数据类型体系和运算体系，掌握各种基本数据类型常量的书写方法和变量的定义、赋值、初始化方法，了解基本运算符的运算规则和优先级别，能正确构成基本类型的表达式，掌握顺序结构的程序设计	4
第4章 选择结构程序设计	了解分支结构的C语言程序设计，熟练掌握关系运算和逻辑运算，能正确选取选择语句来设计选择结构的程序	4
第5章 循环结构程序设计	了解循环结构的程序设计，能熟练掌握C语言的各种循环语句的格式和功能，并能根据循环结构的要求正确选取循环语句来实现循环，掌握简单问题的程序设计	4
第6章 数组	掌握一维数组、多维数组（主要指二维）、字符数组的定义、初始化、数组元素的引用的方法，掌握有关处理字符串的库函数的使用方法	4
第7章 指针	掌握地址、指针、指针变量的概念，能正确定义所需类型的指针变量，能正确将指针变量指向某变量或数组，能正确利用指针变量引用所指向的变量或数组，了解指针数组和多级指针的概念	4
第8章 函数与模块化程序设计	熟练掌握用户函数的结构、设计方法和调用方法，掌握函数调用中数据传递的几种方法，会设计简单的嵌套调用函数，了解递归调用函数、指针型函数的概念，能正确使用书中介绍的各种常用库函数	4
第9章 结构体和共用体	了解结构体、共用体和枚举类型数据的特点，掌握结构体类型的定义方法以及结构体变量、数组、指针的定义、初始化和成员的引用方法，掌握共用体和枚举类型的定义方法和对应变量的定义和引用，掌握用户自定义类型的定义和使用	4
第10章 编译预处理	掌握宏定义和宏替换的一般方法，掌握包含文件的处理方法，了解条件编译的作用和实现方法，了解带参数的主函数的设计和运行方法	2
第11章 文件	掌握缓冲文件系统中有关文件操作的库函数的使用方法，能设计对文件进行简单处理的实用程序	4
总课时	第1~11章建议课时	40
	上机实验建议课时	40

说明：

1）建议课堂教学全部在多媒体教室内完成，实现“讲一演”结合。

2）建议教学分为课堂教学与实验教学两大模块。课堂教学重点讲授C语言的核心知识及程序设计的方法、步骤。实验教学重在培养学生的分析问题和解决问题的能力，提高实际编程能力。

3）第12、13章作为选学内容，帮助有能力的学生进一步提高程序设计的技能。

4）不同学校可以根据各自的教学要求和计划学时数对教学内容进行取舍。

目　录

第1章　C语言程序设计概述

学习计算机程序设计语言是提高计算机知识水平的重要步骤。C语言作为当今最为流行的程序设计语言之一，不但成为计算机专业的必修课程，而且也越来越多地成为非计算机专业的学习课程。本章首先讲述程序及程序设计的基本知识，然后介绍C语言的发展与特点，叙述C语言程序的组成与结构，阐明C语言的上机步骤和方法。

1.1　程序与程序设计

计算机通过执行程序完成其工作，如计算、控制、文字处理、图形处理、网络通信等。所谓程序,就是一组指令和数据的集合。程序就是人与机器进行“对话”的语言，也就是我们常说的程序设计语言。

程序设计语言分为低级语言和高级语言两大类。低级语言直接面向机器，如机器语言和汇编语言；高级语言独立于机器，用高级语言编写的程序在不同的机器上必须使用不同的翻译程序。C语言程序是一种高级语言程序，它必须被翻译成计算机能识别的语言，即机器语言，才能在计算机上运行。

计算机只能理解机器语言，不能理解汇编语言和高级语言，必须把汇编语言或者高级语言编写的程序“翻译”成机器语言才能执行。把汇编语言程序翻译成机器语言的过程称为“汇编”，把高级语言程序翻译成机器语言有两种方式：一种是“解释”，一种是“编译”。C语言属于编译型语言。编译的原理就是由编译程序把源程序编译、连接成可执行文件，然后由机器直接执行，具体关系如图1-1所示。

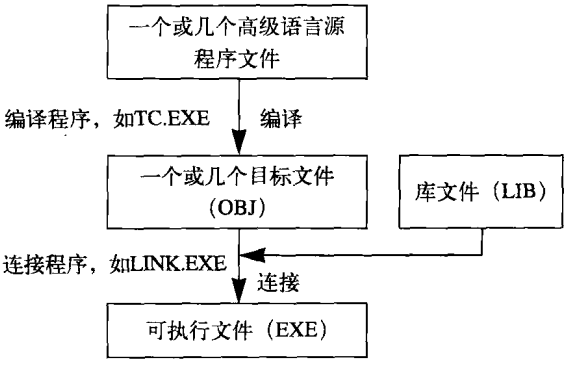


图1-1　C语言源程序的运行过程

C语言程序的运行过程一般为：首先把C语言源程序编辑输入到计算机中，随后调用编译程序对源程序进行编译，产生目标文件，再调用连接程序进行连接，产生可执行文件，最后由机器直接运行可执行文件。

程序员编写的源程序必须遵循编译程序规定的语法，即编写程序的规则。各种类型的语法规定产生了C、PASCAL、BASIC、FORTRAN等语言。C语言语法标准为：

- ANSI C：1983年由美国国家标准协会（ANSI）发布；
- 87 ANSI C：1987年由美国国家标准协会发布。

程序设计所需的开发环境（又称为软件条件）如图1-2所示。

程序开发环境中各程序的功能如下：

源程序编辑程序：编程者输入源程序的编辑工具。

编译程序：把源程序翻译成目标程序。

连接程序：把目标程序及有关的库函数连接成机器可执行的程序文件。

调试程序：帮助编程者定位错误和改正错误的软件工具。

源程序编辑程序（Editor）	集成开发环境IDE (Integrated Developer Environment)
编译程序（Compiler）	
连接程序（Linker）	
调试程序（Debugger）	

图1-2 C语言程序开发环境

程序开发环境的核心是编译程序，它把程序员编写的类自然语言源程序翻译为机器指令；同时，从应用的角度讲，它提供了程序设计的思想。主要的程序设计思想有：结构化程序设计思想和面向对象程序设计思想。典型的程序开发环境及其特点如图1-3所示。

Microsoft	Borland (Inspires)	特点
MSC	Turbo C	结构化程序设计 开发DOS程序
Visual C++	C++ Builder	面向对象程序设计 可视化程序设计环境 专业化Windows 9x/NT程序
Visual Basic	Delphi （类Pascal语法）	面向对象程序设计 可视化程序设计环境 快速设计Windows 9x/NT程序

图1-3 程序开发环境及其特点

程序设计的过程如下：

1）提出问题及要达到的要求，即明确要解决的问题；

2）确定数据结构和所采用的算法，即求出解决问题的方法和思路；

3）编制程序，即用程序设计语言表达出解决问题的步骤；

4）调试程序，即找出程序中的错误并改正，使得程序能达到预期的目的；

5）整理并撰写文档，即记载解决问题的过程及程序输入与输出的结果等。

1.2 C语言简介

1.2.1 C语言的发展历程

C语言是一种流行的高级程序设计语言，它既可以用来编写应用软件，也可以用来编写系统软件。

早期的操作系统都是用汇编语言编写的，但是由于汇编语言依赖于特定的计算机硬件，可移植性差，而且汇编语言编写的程序都比较难以读懂，于是人们就想用一种高级语言来编写系统软件，但在高级语言中却很难实现汇编程序对硬件的操作能力。C语言正是在这种背景下设计出来的。

C 语言是1972年由美国的Dennis Ritchie设计发明的，它由早期的编程语言BCPL（Basic Combined Programming Language）发展演变而来，并首次在UNIX操作系统的 DEC PDP-11 计算机上使用。在1970年，AT&T贝尔实验室的Ken Thompson根据BCPL语言设计出较先进的并取名为B的语言，最后导致了C语言的问世。随着微型计算机的日益普及，出现了许多C语言版本。由于没有统一的标准，使得这些C语言之间出现了一些不一致的地方。为了改变这种状况，美国国家标准协会（ANSI）为C语言制定了一套ANSI标准，它成为C语言标准。

1.2.2 C语言的特色

1. C语言的优点

人们之所以要采用C语言编制各种各样的程序，是因为C语言有许多独特的优点：

1）C语言简洁、灵活。C语言不像FORTRAN那样有严格的格式，它的程序格式书写自由；与Pascal相比，C语言的关键字简练、源程序短，输入的工作量比较小。采用C语言编程，可以使程序员专注于算法设计，不必过多地考虑格式的限制。

2）C语言有丰富的运算符，这会使源程序精练，生成的代码质量高，运行速度快。

3）C语言数据类型丰富，能实现各种复杂的运算，尤其指针类型数据，使程序更加灵活、多样。

4）C语言语法限制不是很严格。例如，C语言对数组下标越界不做检查，由程序员保证程序的正确性；同时，对变量类型的使用比较灵活，例如，整型与字符型及逻辑型数据可以互相通用。

5）C语言可以直接访问物理地址和计算机硬件，能进行位操作，可以实现汇编语言的很多功能。因此，C语言具有高级语言和低级语言的双重功能，可以用来编写系统软件。

6）C语言编写的程序可移植性好，一般不做修改或者做少量的修改就能运行于不同的计算机和不同的操作系统。

读者在编写C语言程序，特别是在与其他语言对比时，会对以上的优点有更加深刻的体会。

2. C语言的特点

1）C语言是结构化程序设计语言。

2）C语言是模块化的程序设计语言，其程序由许多函数组成 。C语言编制的程序必须有一个称为main()的主函数，而且只能有一个主函数；“{”和“}”分别表示函数的起点和终点，相当于PASCAL的BEGIN...END；函数之间可以相互调用、递归调用。但一般函数不能调用主函数。

3）C语言程序可以调用其他文件的函数，这样，一个C语言程序可以由许多文件组成，便于合作开发。

4）C语言的一个语句可以放在一行，也可以放在多行；C语言程序的一行可以放多个语句。C语言的语句都要用“；”作为结束标志。但是，为了便于阅读，编写C语言程序应遵循一定的规则，如嵌套循环时应该有缩行。

5）为便于C语言程序的维护和帮助人们理解程序，C语言的关键语句应该有注释，注释部分必须用“/*”和“*/”括起来，并且“/”和“*”之间不能有空格，编译程序在编译时会忽略掉“/*”和“*/”之间的内容。

6）C语言的程序一般要有头文件，头文件在程序的开始用“#include”做出说明，头文件中可以是对程序中所用变量的说明，也可以是引用的库函数。

7）C语言区分大小写，因此在使用C语言时应特别注意。

8）C语言的程序总是从主函数开始执行，并且终止于主函数。

总之，C语言灵活性大、功能强，可以编写出各种类型的程序。程序员使用C语言限制少，可以自由地编程。但是，从学习语言的角度来说，学习C语言比学习其他高级程序设计语言要难一些。学习C语言的困难主要来自于C语言的灵活性，只要掌握C语言的基本语法规则，多上机练习，C语言还是可以学好的。

1.3 简单的C语言程序

我们还是以“Hello World!”作为本书的开始程序。

【例1-1】最简单的C语言程序。

```
main()
{
   printf("Hello World!\n");
}
```

程序执行后在屏幕上输出：

```
Hello World!
```

其中main表示“主函数”，每个C程序都必须有一个main函数。函数体由大括弧{ }括起来。本例中主函数内只有一个输出语句，printf是C语言中的输出函数（详见第3章）。双引号内的字符串按原样输出，"\n"是换行符，即在输出“Hello World!”后回车换行，语句最后以分号结束。

【例1-2】两个数求和。

```
#include <stdio.h>
main()
{
   int a,b,sum;                    /*定义变量 */
   a=3;                            /*给变量a赋值 */
   b=6;                            /*给变量b赋值 */
   sum=a+b;                        /*求a、b的和并赋给变量sum */
   printf("sum=%d\n",sum);         /*输出运算的结果 */
}
```

程序执行后的输出结果为：

```
sum=9
```

程序先定义三个变量（变量就是存放数据的存储单元），然后分别给变量a和变量b赋值，变量sum是变量a和变量b的和，程序输出的结果正是a和b的和。"%d"表示输出的是十进制整数。

书写程序时应遵循的规则

从书写清晰，便于阅读、理解和维护的角度出发，在书写程序时应遵循以下规则：

1）一个说明或一个语句占一行。

2）用“{}”括起来的部分，通常表示程序的某一层次结构。“{}”一般与该结构语句的第一个字母对齐，并单独占一行。

3）低一层次的语句或说明可比高一层次的语句或说明缩进若干格后书写，以便看起来更加清晰，增加程序的可读性。

在编程时应力求遵循这些规则，以养成良好的编程风格。

1.4 C语言程序的上机步骤

C语言是编译型语言，源程序必须经过编译才能在计算机上执行。C语言程序的上机步骤可大致分为下列几步：

1）上机输入与编辑源程序；

2）对源程序进行编译；

3）与库函数连接；

4）运行可执行的目标程序。

1.4.1 在Turbo C环境下运行C程序的步骤

在Turbo C所在的目录下直接键入TC，就可以打开C语言程序的集成开发环境，如图1-4所示。

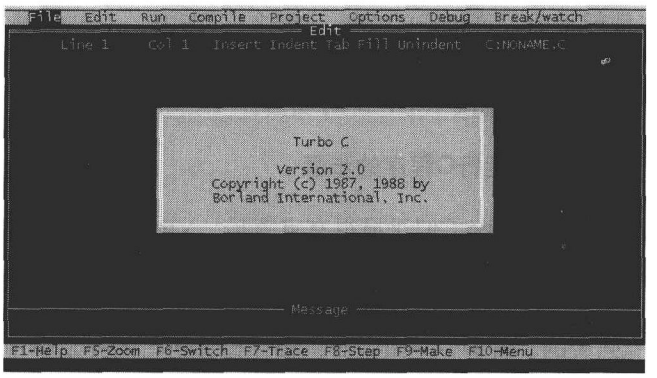


图1-4 C语言程序的集成开发环境

在Turbo C集成开发环境中，有一行主菜单，其中包括File、Edit、Run、Compile、Project、Options、Debug、Break/watch 8个菜单项。

编程者可以通过以上菜单项来选择使用Turbo C集成开发环境所提供的各项主要功能。以上8个菜单项分别代表文件操作、编辑、运行、编译、项目、选项、调试、中断/观察等功能。按F10键就可选中某主菜单项，随后可用“←”键和“→”键移动光标来选择你所需要的菜单项，选定了主菜单项，再按回车键就可打开下级菜单。

File菜单用来对文件进行操作，包括装载文件、建立新文件、存储文件等。按Alt+F组合键可以打开File下拉菜单，如图1-5所示。

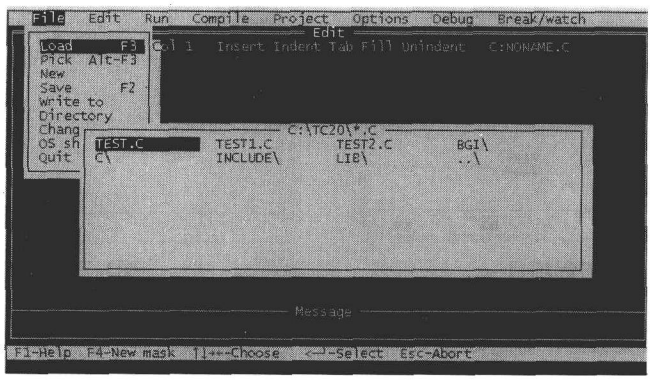


图1-5 File菜单

“New”命令用于建立一个新文件，选择“New”命令后集成开发环境就会打开编辑窗口，读者可以在此输入C语言源程序。

读者编辑完源程序后，要对源程序进行编译，生成可执行文件方可执行。按下Alt+C组合键，选择“Compile to OBJ”生成目标文件，然后选择“Link EXE file”连接目标文件；或者直接选择“Make EXE file”生成可执行文件。如果源程序没有语法和语义错误，可以生成后缀为“.exe”的可执行文件。如果源程序存在错误，集成开发环境会指出错误所在的行，读者可以打开编辑窗口修改源程序。

生成可执行文件后，按下Alt+R组合键，选择“Run”，或者直接按下Ctrl+F9组合键，可以

执行此文件。也可以退出集成开发环境，在可执行文件所在的目录下，直接输入可执行文件名，也可以执行该文件。

如果在编译和连接时出现“出错信息”，则需要重新编辑（修改）源程序。修改后仍然需要进行编译和连接，最后再运行可执行的文件（程序）。所以，上机步骤往往是一个循环往复的过程。

以上介绍的上机步骤只需上机试一下，即可明白。

1.4.2 在Visual C++ 6.0环境下运行C程序的步骤

随着计算机技术的发展，Turbo C似乎显得有点落后，Visual C++ 6.0（后面简称VC6.0）占据了越来越大的市场份额，大多数学校的上机环境都选用VC系统，计算机等级考试也开始使用VC系统。为了适应发展，我们在此介绍一下VC6.0。

VC6.0是一款功能强大的、针对C++面向对象程序设计语言的开发环境，由于C++语言是在C语言的基础上扩展而成的，所以C语言程序也能在该环境下运行。

下面就简单介绍在VC6.0环境下运行C程序的步骤。

1） 创建工程项目。首先打开VC。选择“新建”建立一个新工程，选择“Win32 Console Application”，在“工程名称”中输入工程的名字，如“hello world”，在“位置”中设置好路径。单击“确定”按钮，如图1-6所示。

2）新建C源文件。选择“新建”，在弹出的对话框中选择“C++ Source File”，在“文件名”中输入文件名称，如“hello”，单击“确定”按钮，如图1-7所示。

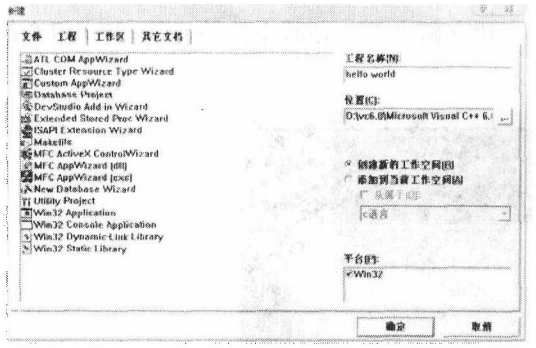


图1-6 创建工程项目

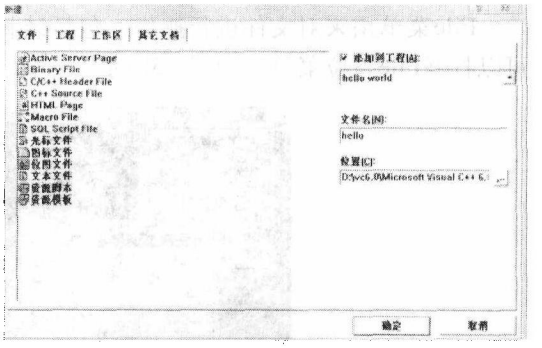


图1-7 新建C源文件

注意：填入C源文件名一定要加上扩展名“.c”，否则系统会为文件添加默认的C++源文件扩展名“.CPP”。

3）编辑C源程序。在编辑框中输入源程序，如图1-8所示。

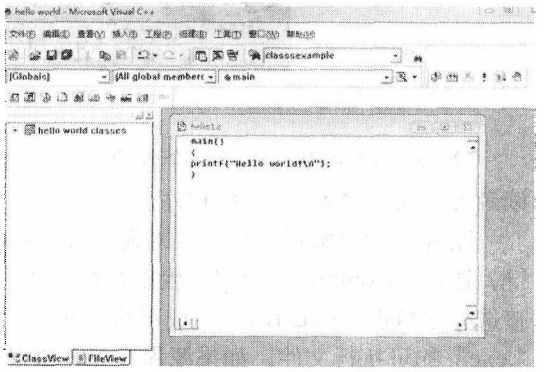


图1-8 编辑源文件

4）编译，连接，运行。选择主菜单“组建(Build)”中的“编译（Compile)”命令，或单击工具条上的图标或者按Ctrl+F7组合键，系统开始对源文件进行编译。注意，这里系统只编译当前文件而不调用连接器或其他工具。输出（Output）窗口将显示编译过程中检查出的错误或警告信息，在错误信息处单击鼠标右键或

双击鼠标左键，可以使输入焦点跳转到引起错误的源代码处（大致位置）进行修改。

选择主菜单“组建（Build）”中的“组建（Build）”命令，或单击工具条上的图标，对最后修改过的源文件进行编译和连接。选择主菜单“组建（Build）”中的“执行(Build Execute)”命令，或单击工具条上的图标！，执行程序，将会出现一个新的用户窗口，按照程序输入要求正确输入数据后，程序即正确执行，用户窗口显示运行的结果。如图1-9所示。

图1-9 运行结果

当然，作为一个成熟的开发环境，VC++6.0还有很多我们需要学习的地方，由于篇幅限制，在这里不再赘述，请参阅附录D或相关资料。

1.5 C语言的基本词法

在C语言中使用的基本词法可分为六类：标识符、关键字、运算符、分隔符、常量、注释符。

1. 标识符

标识符是指常量、变量、语句标号以及用户自定义函数的名称。除库函数的函数名由系统定义外，其余的都由用户自己定义。C语言规定，标识符只能是字母（A～Z，a～z）、数字（0～9）、下划线（_）组成的字符串，并且其第一个字符必须是字母或下划线。例如，以下标识符是合法的：a，x，x3，BOOK_1，sum5，_x7；以下标识符是非法的：3s（以数字开头），s*T（出现非法字符*），-3x（以减号开头），bowy-1（出现非法字符-）。

在使用标识符时还必须注意以下几点：

1）标准C不限制标识符的长度，但标识符的长度受各种版本的C语言编译系统的限制，同时也受到具体机器的限制。例如在某版本C中规定标识符前8位有效，当两个标识符前8位相同时，则被认为是同一个标识符。Turbo C规定标识符的长度为32。在编写程序时，应对系统所规定的标识符的长度有所了解，以免造成不必要的错误。这种错误不会被编译系统发现，所以应特别小心。

2）在标识符中，大小写是有区别的。例如SUM和sum是两个不同的标识符。变量名应尽量使用小写字母，以增加程序的可读性。

3）标识符虽然可由程序员随意定义，但它用于标识某个量的符号，因此，命名应尽量有相应的意义，以便阅读理时能“顾名思义”，一般不用简单、无意义的符号作为变量名，如a、b等。

在C语言中，所有的变量都是先定义后应用，使用没有定义的变量名被认为是“非法”的。

2. 关键字

关键字是由C语言规定的具有特定意义的字符串，通常也称为保留字。用户定义的标识符不应该与关键字相同。C语言的关键字分为以下几类：

1）类型说明符。用于定义和说明变量、函数或其他数据结构的类型，如int、double、float、long、short、auto、signed、static、struct、unsigned、char、enum、extern、register、union等。

2）语句定义符。用于表示一个语句的功能，如条件语句的语句定义符if else，循环语句的语句定义符do、while、for、goto等。

3）预处理命令字。用于表示一个预处理命令，使用时前面要加“#”，如include、define、

ifdef、ifndef、undef、endif等。

关键字后必须有空格、圆括号、尖括号、双引号等分隔符，否则与其他字符一起将会组成新的标识符。

3. 运算符

C语言中含有丰富的运算符。运算符与变量、函数一起组成表达式，表示各种运算功能。运算符由一个或多个字符组成。

4. 分隔符

在C语言中最常用的分隔符有逗号和空格两种。逗号主要用在类型说明和函数参数表中分隔各个变量。空格多用于语句各单词之间，作为间隔符。在关键字和标识符之间必须要有一个以上的空格符作为间隔，否则将会出现语法错误。例如把“int a,”写成“inta;”后，C编译器会把inta当成一个标识符处理，其结果必然出错。

5. 常量

C语言中使用的常量分为数字常量、字符常量、字符串常量、符号常量、转义字符等多种，这些在后续章节中将专门给予介绍。

6. 注释符

C语言的注释符是以“/*”开头并以“*/”结尾的字符串。在“/*”和“*/”之间的即为注释。程序编译时，不对注释做任何处理。注释可出现在程序中的任何位置，用来向用户提示或解释程序的意义。在调试程序中对暂不使用的语句也可用注释符括起来，使编译跳过这些语句不做处理，待调试结束后再去掉注释符。

7. C语言的字符集

字符是组成语言的最基本的元素。C语言字符集由字母、数字、空格、标点和特殊字符组成。在字符常量、字符串常量和注释中还可以使用汉字或其他可表示的图形符号。

- 字母：小写字母a～z共26个，大写字母A～Z共26个。
- 数字：0～9共10个。
- 空白符：空格符、制表符、换行符等统称为空白符。空白符只在字符常量和字符串常量中起作用；在其他地方出现时，只起间隔作用， 编译程序对它们忽略。因此在程序中使用空白符与否，对程序的编译不会发生影响，但在程序中适当的地方使用空白符将增加程序的清晰性和可读性。

1.6 C语言程序的基本结构

C语言程序设计的基本结构可分为三种：顺序结构、分支结构、循环结构。按照结构化程序设计的观点，任何功能的程序都可以通过这三种基本结构的组合来实现。

1. 顺序结构

如图1-10所示，模块A和B是顺序执行的，即在执行完A的操作之后，才能执行B的操作。顺序结构是最简单的一种基本结构。

我们可以把模块A和模块B合并成一个模块，但无论怎样合并，生成的新模块仍然是一个整体，只能从模块的顶部进入，执行完模块的所有语句之后，再从底部退出模块，这也是模块的基本性质。

2. 分支结构

当根据逻辑条件成立与否，分别选择执行不同的程序模块时，可以使用分支结构，分支结构也称为选择结构。

图1-11即是一个分支结构的流程图，当满足判断条件时，执行模块A；当不满足判断条件时，执行模块B。在程序的一次执行中，模块A和模块B只可能执行其中的一个。在实际应用中，也可能其中一个模块为空，这是允许的，如图1-12所示。

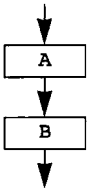


图1-10 顺序结构

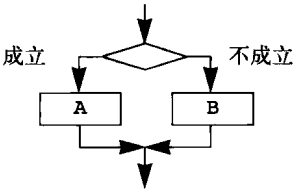


图1-11 分支结构（一）

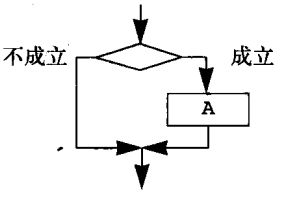


图1-12 分支结构（二）

3. 循环结构

循环结构和顺序结构、选择结构一样，都是构成各种复杂程序的基本构造部件，是一种重要的基本结构。循环结构的特点是在给定条件成立时，反复执行某个程序段。通常我们称给定条件为循环条件，称反复执行的程序段为循环体。循环结构一般分成两种情形：一种是当型循环，一种是直到型循环。在程序执行时，当型循环是先判断条件是否成立，当条件成立时，执行模块A中的程序，然后再判断条件，条件成立时，再执行模块A的程序，这样循环往复，直到当一次条件不成立时，才退出循环模块，执行后继的程序，如图1-13所示。

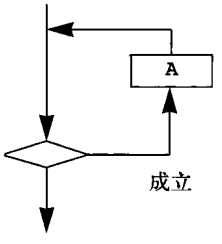


图1-13 当型循环

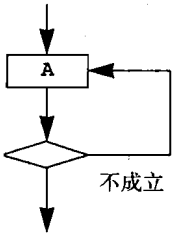


图1-14 直到型循环

直到型循环是先执行一次模块A中的程序，然后判断条件是否成立，如果条件不成立，则继续执行模块A中的程序，然后再判断，如此往复，直到所给的条件成立时，才退出循环程序，如图1-14所示。

在有些问题中，循环模块都是要执行的，至少应该被执行一次，这时当型循环和直到型循环是没有什么区别的，它们都能很好地解决问题。但是，在其他一些问题中，循环模块有可能一次也不会被执行，这时就应该用当型循环。一般说来，不用刻意去设计某个问题是当型循环还是直到型循环，具体问题应具体分析。

在循环的使用中，应避免使循环陷入死循环，即无限地执行循环模块，永远不会退出。这样的程序在实际应用中是没有任何意义的，不会得到什么有价值的结果。在设计循环条件时应特别注意，应保证循环在有限的时间内退出。

其实，基本结构不是只限于上面所说的三种。但是已经证明，由以上三种基本结构组成的程序可以解决任何复杂的问题。三种基本结构所构成的程序属于结构化的程序，它不存在无规律的向前、向后的转向，在本结构内可以用转向语句进行跳转。

本章小结

在计算机的发展历史上，依次经历了机器语言、汇编语言和高级语言的阶段。C语言属于