HZ BOOKS
华章教育

计算机基础课程系列教材

THE C PROGRAMMING LANGUAGE
THEORY AND PRACTICE

C语言程序设计
理论与实践

孙浩 主编
刘亮 副主编
王宁 张莉萍 参编

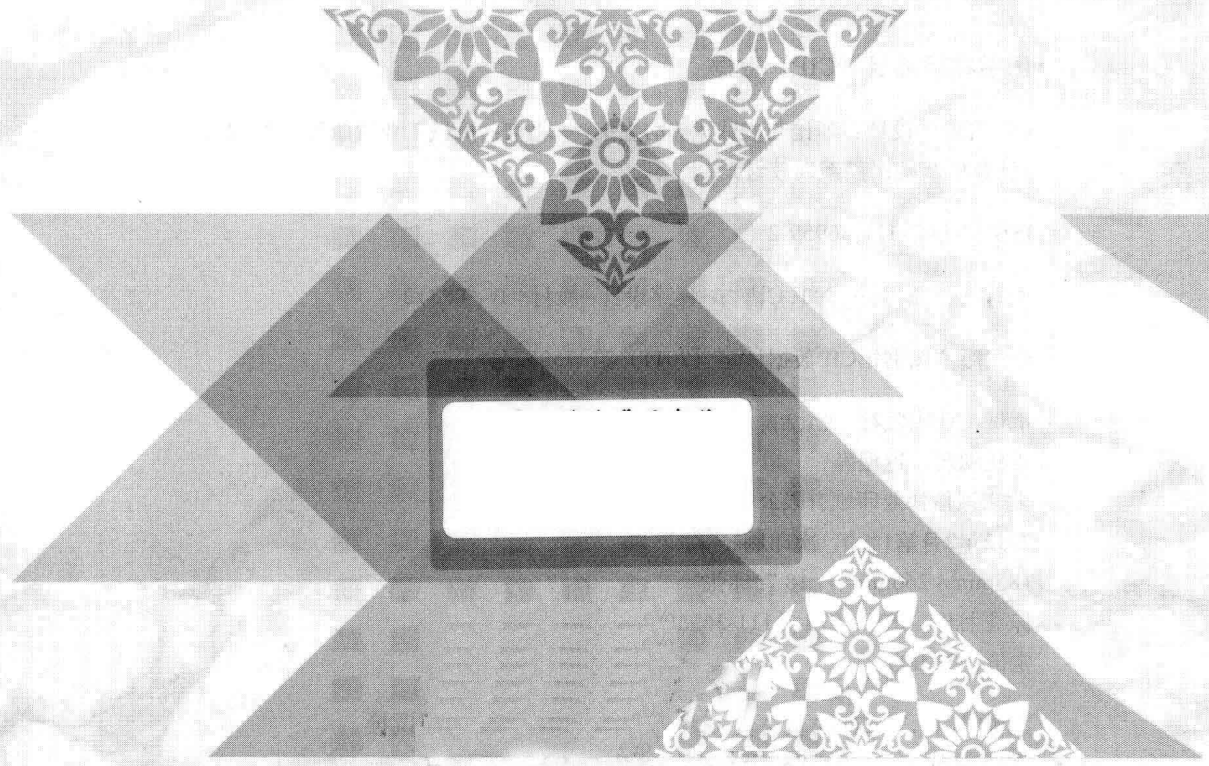
机械工业出版社
China Machine Press

计算机基础课程系列教材

THE C PROGRAMMING LANGUAGE
THEORY AND PRACTICE

C语言程序设计
理论与实践

孙浩 主编
刘亮 副主编
王宁 张莉萍 参编



机械工业出版社
China Machine Press

本书较全面地介绍了C语言程序设计的基础知识和基本编程技能，分为理论教学篇和实践教学篇两部分。理论教学篇的主要内容包括：C数据类型、数据的输入和输出、流程控制、指针、数组、函数、编译预处理、结构体与共用体、文件、位运算、指针的高级应用、程序设计风格与调试等。在内容的编排上，对相关知识点设计了一些随堂练习，注重讲练结合及知识点的融会贯通和实际运用，同时对重要的知识点进行提示和归纳总结，方便读者自学。实践教学篇根据理论教学篇的布局，安排了10个实验和1个课程设计题目。

本书内容新颖，体系合理，内容翔实，通俗易懂，可以作为高等学校相关课程的教材，也可以作为计算机等级考试的辅导用书，还可以作为相关研究人员的参考书。

图书在版编目（CIP）数据

C语言程序设计：理论与实践 / 孙浩主编. —北京：机械工业出版社，2012.2
（计算机基础课程系列教材）

ISBN 978-7-111-36959-2

Ⅰ. C…　Ⅱ. 孙…　Ⅲ. C语言－程序设计－教材　Ⅳ. TP312

中国版本图书馆CIP数据核字（2011）第275952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：迟振春
北京诚信伟业印刷有限公司印刷
2012年2月第1版第1次印刷
185mm × 260mm · 18.75印张
标准书号：ISBN 978-7-111-36959-2
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语言是广泛使用的程序设计语言之一，其功能丰富、表达能力强、使用灵活方便、应用面广、目标程序效率高，既具有高级语言的特性，又具有直接操纵计算机硬件的能力，因此特别适用于编写系统软件。目前，C语言被许多高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，学生在学习的时候只通过理论学习是学不好的，必须通过大量的编程训练，在实践中掌握基础知识，培养程序设计的基本能力，并逐步理解、掌握程序设计的思想和方法。因此，C语言程序设计课程的重点应该是在学生掌握基本理论知识的基础上，培养学生的实践编程能力。

目前，介绍C语言的教材很多，但在多年的教学实践中，我们发现其中大部分教材只注重C语言本身语法知识的阐述，而忽略了培养学生的实践编程能力。这样，尽管学生记住了一大堆语法知识，却写不出像样的程序，不利于培养学生的程序设计能力和语言应用能力。

我们编写本书的目的是让学生在加强基础理论知识学习的基础上，提升实际编写程序的能力。为实现这个目标，我们采用了如下策略：

- 在介绍基础语法知识的同时，阐述基本的编程技巧和注意事项，注重基础语法知识的实际应用。
- 对于用文字语言难以阐述的内容，采用图表来阐述，使得相应的知识点更加清晰、直观，便于学生加强印象。
- 突出显示的板块结构总结了C语言的一些重要特征，并对一些编程技能和注意事项进行了归纳，便于学生参考学习。
- 每部分知识点后都有相应的随堂练习，每章后都设置了相应的习题，方便学生检测自己的学习水平，加深对基础理论知识的理解。

全书以程序设计为主线，以编程应用为驱动，重点介绍程序设计的基本思想和基本方法。本书分为理论教学篇和实践教学篇两部分，其中理论教学篇分为13章，第1~5章侧重于介绍C语言的基础知识和基本编程技能，主要包括C语言的基本特点，基本数据类型和指针，数据处理中的各种表达式以及顺序、分支、循环3种基本的流程控制语句。第6~11章侧重于指针和数组、函数、编译预处理、文件、结构体和共用体、位运算等知识的综合运用，采用结构化程序设计的思想来实现复杂问题的编程和基本的算法。第12章侧重于阐述指针的高级应用，对存储空间的分配和释放问题以及基本的链表应用都做了详细的介绍。第13章介绍良好的C语言程序设计风格以及常见的错误调试等。实践教学篇以训练学生的实际编写程序能力为目的，设置了10个实验和1个课程设计。此外，在附录五里面对常见的算法进行了汇总。

在教材的组织和编写风格上，注重以下几个方面：

1）提出问题。通过应用场景的引入和分析，使学生对C语言中各种语言成分的重要作用有一个比较清楚的认识。

2）分析问题。提供关键的算法说明，明确指明数据获取部分、加工处理部分和结果展现部分的实现方法。

3）解决问题。注重对于样本程序的分析和理解，源程序要点部分有注释说明。

4）理解编程语言。对于新引入的语言成分给出详细的说明，并通过一些知识点小结使学

生加深理解。

5）拓展应用。通过程序功能的不断变化（例题以及随堂练习），使学生开拓思路，更好地理解算法和程序语言。

6）阶段小结。归纳本章重点，并提出深入学习的指导建议。

为方便教学，我们为使用该书的教师提供了丰富的教学资源，包括电子课件、源程序以及习题参考答案等，需要的教师可以从华章网站（www.hzbook.com）下载。

本书的作者全部来自重庆邮电大学和重庆邮电大学移通学院从事C语言教学工作的一线老师，有着丰富的教学经验。其中第1、2、3、4、5、9、10、11、13章由孙浩编写，第6、7、8、12章由刘亮编写，实验一到实验七和附录由王宁编写，实验八到实验十和课程设计由张莉萍编写。全书由孙浩进行统稿审阅。感谢闫会峰老师、向碧群老师和武汉商业服务学院的汪婷老师为本书所做的贡献，感谢东北大学的张锡哲博士、哈尔滨工程大学的吕天阳博士和吉林大学的王上博士对我们工作的支持。

由于作者水平有限，书中难免存在谬误之处，敬请读者指正。谢谢！作者的联系方式：sunhao2001@163.com。

孙 浩

2011年12月

教学建议

教学目的和要求

本课程以C语言为基础，教授程序设计语言的基本概念和基本理论，使学生掌握高级程序设计的基本方法和基本技巧，能够独立编写简单的基本程序，能理解和修改已有的程序。此外，结合实践环节，使学生具备一定的分析问题和解决问题的能力，能完成小系统的设计和实现。

通过本课程的学习，学生应达到下列要求：

1）正确理解C程序设计语言的各种语言成分。

2）掌握结构化程序设计方法，形成良好的程序设计风格。

3）具备一定的程序设计能力。

4）具备较强的上机操作和程序调试技能。

先修课：大学计算机基础

课程内容及其安排

章	学习要点及教学目的	讲授学时	实验学时及内容
第1章　C语言概览	了解： • C语言的历史 • C语言的特点 掌握： • C语言的程序结构 • 在Visual C++环境下编辑、编译、连接和运行一个C语言程序的基本步骤	2	2（实验一）
第2章　C数据揭秘	了解： • C语言的基本语法要素 • C语言的各种数据类型 掌握： • 整型常量、浮点型常量、字符常量、字符串常量、符号常量的表示方法 • 整型变量、浮点型变量、字符变量的表示方法 • 变量的初始化 • sizeof运算符	6	2（实验二）
第3章　数据的输入和输出	了解： • 输入和输出的基本概念 掌握： • putchar和getchar函数 • puts和gets函数 • 格式输入和输出函数scanf和printf	4	2（实验三）
第4章　C语言流程控制	了解： • 程序流程图 • 顺序、选择和循环的基本概念 • C语言语句的基本概念	14	4（实验四、实验五）

（续）

章	学习要点及教学目的	讲授学时	实验学时及内容
第4章　C语言流程控制	掌握： • 算术运算符和算术表达式 • 赋值运算符和赋值表达式 • 顺序结构程序设计 • 关系运算符和关系表达式 • 逻辑运算符和逻辑表达式 • if语句和switch语句 • 选择结构程序设计 • 自增、自减运算符 • while、do…while语句、for语句 • break语句和continue语句 • 循环结构程序设计与循环的嵌套 难点： • 自增、自减运算符 • break和continue语句的区别	14	4（实验四、实验五）
第5章　初识指针	了解： • 地址、指针的基本概念 • 指针变量的基本概念 掌握： • 指针变量的定义、初始化和引用 • 指针的简单运算 • 动态存储管理malloc和free函数	2	0
第6章　数组与指针	了解： • 数组的概念 • 数组存储方式 掌握： • 一维数组的定义、初始化和引用 • 一维数组与指针 • 字符数组和字符串、常见的字符串函数 • 指针的运算和比较 • 二维数组的定义、初始化和引用 • 对二维数组的理解 • 指针数组 难点： • 数组名与指针变量 • 对二维数组的理解 • 指针数组	10	4（实验六）
第7章　函数与模块化程序设计	了解： • 函数的概念 • 模块化程序设计的基本原则 • 内部函数与外部函数 掌握： • 函数的定义 • 形参与实参 • 函数的调用、值传递过程 • 函数的嵌套调用与递归调用 • 函数、数组与指针的综合运用 • 变量的作用域与存储类别	10	4（实验七、实验八）

（续）

章	学习要点及教学目的	讲授学时	实验学时及内容
第8章　编译预处理	了解： • 条件编译 掌握： • 无参和带参宏定义 • 文件包含	2	0
第9章　结构体、共用体与枚举	了解： • 链表的操作 掌握： • 结构体类型 • 结构体变量的定义、初始化和引用 • 结构体数组 • 结构体指针 • typedef进行类型定义 • 共用体变量的定义、初始化和引用 • 枚举类型	4	2
第10章　文件操作	了解： • 文件的概念、分类 • 文件类型指针 • 文件出错检测 掌握： • 文件的打开和关闭 • 文件的顺序读写 • 文件的随机读写	4	2（实验九）
第11章　位运算	了解： • 位运算的基本概念 • 位段 掌握： • 按位与、按位或、按位异或、取反运算符 • 左移位和右移位运算符	2	0
第12章　指针的高级应用（选讲）	了解： • 悬空指针问题 • 链表结点的插入、删除操作 掌握： • 存储空间的分配与释放 • 空指针问题	2	2（实验十）
第13章　C程序设计风格与调试（选讲）	了解： • C语言程序设计风格 • 常见的错误分析 掌握： • 程序错误分析方法 • 程序错误调试方法	2	0
合计		64	24

考试与课程设计

课程采用理论与实践相结合的方式，理论学时64，实验学时24。学期总成绩采用笔试加上机考试的方式，笔试成绩占60%，上机考试成绩占40%。建议信息类专业安排一周的课程设计，指导学生独立完成C语言小系统的设计与开发。

目　录

第二部分 实践教学篇

附 录

第一部分 理论教学篇

第1章 C语言概览

在本章中将学习到如下内容：

☆ C语言的发展历程和特点

☆ 如何编写和运行第一个C语言程序

欢迎来到C语言的世界，C语言是国际上广泛流行的专业化编程语言，深受软件编程人员的欢迎，现在，我们将一起学习这一强大而实用的计算机编程语言。

1.1 C语言的生命力

1.1.1 C语言的发展历程和趋势

C语言是在B语言的基础上发展起来的，它的原型是ALGOL 60语言（也称A语言）。ALGOL 60是一种面向问题的高级语言，它远离底层硬件，不适合编写系统程序。1963年，剑桥大学将ALGOL 60语言发展成为CPL（Combined Programming Language）。CPL语言比ALGOL 60语言更加接近硬件，但规模比较大，实现困难。1967年，剑桥大学的Matin Richards对CPL语言进行简化，形成了BCPL（Basic Combined Programming Language）。1970年，美国贝尔实验室的Ken Thompson在BCPL语言的基础上做了进一步简化，设计出简单且接近硬件的B语言（取BCPL的第一个字母），之后采用该语言编写了第一个UNIX操作系统，并在PDP-7上实现。1973年，贝尔实验室的Dennis M. Ritchie在B语言的基础上设计出一种新的语言——C语言（取BCPL的第二个字母）。C语言既保持了BCPL和B语言的精练、接近硬件的优点，又克服了它们过于简单、数据无类型的缺点。1977年，D. M. Ritchie发表了不依赖具体机器的C语言编译文本《可移植的C语言编译程序》，简化了C移植到其他机器上的工作，推动了UNIX操作系统在各种机器上的实现。1987年，Brian W. Kernighan和Dennis M. Ritchie（合称K&R）合著了影响深远的名著《The C Programming Language》[⊖]，从而使C语言成为目前世界上广泛使用的高级程序设计语言之一。

C语言自面世以来进行了多次改进，出现了各种不同的版本。1983年，美国国家标准化协会（ANSI）根据这些版本制定了新的标准，成为ANSI C。1987年，ANSI又公布了新的标准——87 ANSI C。1990年，国际标准化组织（International Standard Organization，ISO）接受87 ANSI C为ISO C的标准（ISO 9899—1990）。1994年，ISO又修订了C语言标准。目前流行的C语言编译系统大多是以ANSI C为基础进行开发的。

随着计算机技术的不断发展，20世纪90年代出现了面向对象的编程语言和开发平台，许多软件开发商开始转向使用C++和Java语言来进行大规模的项目开发。不管这些较新的语言如何

⊖ 本书中文版和影印版已由机械工业出版社引进出版，书名为《C程序设计语言》，中文版书号：7-111-12806-0，影印版书号：7-111-19626-0。——编辑注

流行，C在软件开发产业中仍然是一种重要的编程语言，已被广泛用于嵌入式系统编程，在汽车、照相机、DVD播放器等现代化设备的微处理编程中起着举足轻重的作用。此外，C语言还适用于操作系统的开发，伴随UNIX和Linux的成长，它将扮演更重要的角色。因此，在未来很长的一段时间内，C语言仍将保持强劲的势头。

1.1.2 C语言的特点

C语言具有很强的生命力，与其他编程语言相比，有着自己独特的特点。其主要特点如下：

（1）语言简洁且结构清晰

C语言共有32个关键字、9种控制语句，程序书写形式自由灵活。C语言程序通常由多个函数组成，便于进行模块化和结构化编程，使得程序结构清晰明了、可读性强。

（2）表达能力强

C语言不仅提供了丰富的运算符和数据类型，还提供了强大的功能库，使得程序员可以快速、灵活地编写程序，精确地控制计算机按照自己的意愿工作。

（3）高效率的编译性语言

C语言生成的目标代码质量高，运行速度快。对于较大的程序，源代码可以分别存放，单独编译后再链接在一起，形成可执行文件。

（4）可移植性好

采用C语言编写的程序基本上可以不做修改，直接运行于各种型号的计算机和各种操作系统。

1.2 开发第一个C程序

1.2.1 编写第一个C程序

为了让读者对C语言有一个感性的认识，首先来看一个用C语言编写的程序。

程序1.1　1_1.c程序

```
#include <stdio.h>                          /*文件包含预处理*/
void main()                                 /*main()函数，C程序的入口*/
{
     int add(int x, int y);                 /*对被调用函数add的声明*/
     int a, b, sum;                         /*定义变量a、b和sum */
     printf("Please input two number(like:3,5): \n");  /*使用printf()函数打印输入提示*/
     scanf("%d,%d", &a, &b);                /*使用scanf()函数对变量a、b赋值*/
     sum = add(a, b);                       /*调用函数add，并将函数返回值赋给sum*/
     printf("%d + %d = %d\n", a, b, sum);     /*使用printf()函数输出结果*/
}
int add(int x, int y)                       /*定义函数add，x、y为形式参数*/
{
     return (x + y);                        /*将x + y的和返回，通过add带回到调用函数的位置*/
}
```

程序1.1实现的功能是求从键盘上输入的两个整数之和，图1-1显示了程序在VC++ 6.0环境下的运行结果。如何使用VC++ 6.0来运行这个程序，稍后将作详细介绍。

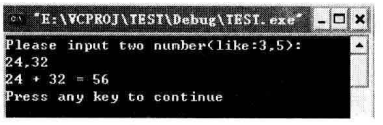


图1-1　程序1.1的运行结果

程序1.1不要求读者完全理解，但希望读者对C程序有一个初步的印象。

图1-2总结了一个典型的C程序的各个组成部分，从中可以看到C程序是由函数构成的，一

个C源程序有且仅有一个main()函数，可以包含多个其他函数，当然也可以没有其他函数，仅有一个main()函数。

程序1.1是由一个main()函数和一个add()函数组成的C程序，下面我们将从每一行代码出发，探讨隐藏在代码背后的细节。

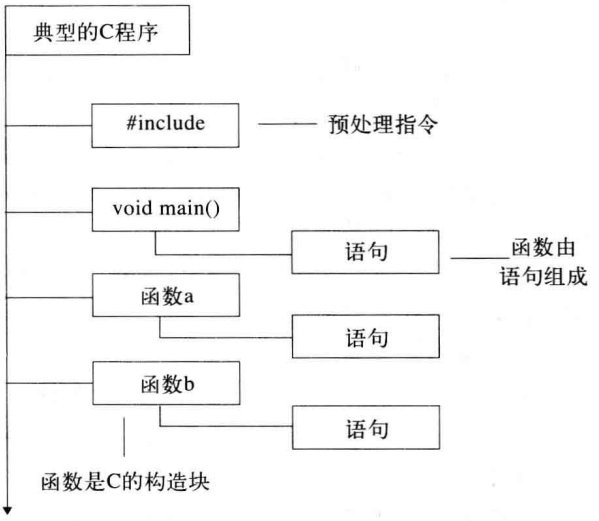


图1-2 C程序剖析图

1. 文件包含预处理指令

#include <stdio.h>，这是一个C预处理指令，该行告诉编译器包含文件stdio.h中的全部信息，其作用相当于在程序中该行所在的位置键入了stdio.h的完整内容。stdio.h文件是标准输入输出头文件（standard input/output header），由C编译系统提供，它包含了有关输入和输出的函数（如printf()、scanf()等）的信息以供编译器使用。在C语言中，将出现在文件顶部的信息集合称为头（header），这些文件通常以.h作为扩展名。当然文件包含预处理指令也可以包含用户定义的其他文件，它的基本格式为：

```
#include <文件名> 或者 #include "文件名"
```

这两者之间的主要区别是：使用<>时，编译系统到存放C库函数头文件的目录中去寻找要包含的文件，而使用""时系统先在用户当前目录中寻找要包含的文件，若找不到再按照<>的方式进行查找。

小提示：文件包含预处理指令的作用

文件包含预处理指令非常有用，它可以节省程序设计人员的重复劳动。例如，某个单位在进行某个项目开发的时候如果需要使用一组固定的数据（如重力加速度为9.8、圆周率为3.1415926……），就可以把这些经常用到的固定的数据定义成一个头文件，项目组的成员需要使用这些数据时只需使用#include命令将头文件包含在自己的文件里就可以了，从而避免每个成员重复定义这些数据，也便于修改相应的值。

2. main()函数

void main()，这行代码声明了main()函数。C程序中必须有一个main()函数，它表示C程序中的主函数，**C程序的执行总是从该函数开始**。如果在main()函数中调用了其他函数，调用结束后流程将返回到main()函数，**在main()函数中结束整个程序的运行**。

void指明了main()函数的返回类型，由于void表示“空”，意味着main()函数的返回类型是“空”，即不返回任何类型的值。有时，也可用int main()声明main()函数，它表明main()函数的返回类型是整数（integer）。

3. 注释

在程序1.1中，有很多诸如“/*文件包含预处理*/”这样的内容，通过阅读这些内容可以很好地帮助我们理解这个程序。包含在/* */之间的部分就是程序注释，用于对程序代码的操作及函数功能进行说明，让我们更容易理解程序代码实现的功能，便于程序开发人员对代码进行更好的维护。在/*和*/之间所有的内容都将被编译器忽略。在写程序注释的时候，可以单放一行

或者是多行。下面是一个多行注释的例子。

```
/*注释的第一行，
这一行仍然是注释。*/
```

单行程序代码的注释也可采用“//”加注释内容的方式。

4. 大括号与函数体

在程序1.1中，不管是main函数还是add函数都有一对“{...}”，这对大括号划定了main函数和add函数的界限。在C语言中，所有的函数都使用花括号来表示函数的开始和结束，在大括号之间的部分就是函数体。函数体由若干语句构成，这些语句描述了函数的功能。

在C语言中，大括号还有一个功能，即用来把某些语句聚集成一个单元或代码块，这些单元或代码块称作复合语句。

5. 变量的定义

int a,b,sum，这一行代码定义了三个变量a、b和sum。前面的int说明这三个变量的类型是整型，即这三个变量的值只能是整数，不能有小数点和小数部分。编译器使用这个信息为a、b和sum分配相应的存储空间。在C语言中，变量代表内存中具有特定属性的一个存储单元，它用来存放数据，即变量的值，在程序运行期间，这些值是可以变化的。一个变量应该有一个名字，以便在程序中引用。

符号int是C语言中的一个关键字，它代表着C语言中一个基本的数据类型。关键字是用来表达语言的单词，不能将它们用于其他目的。例如，我们不能将int用做函数或者变量的名字。在C语言中总共有32个保留关键字，见附录二。

在程序1.1中，a、b和sum都是定义的变量。下面定义的变量都是不合法的：

int money. dollar, 123_num, a<b, #45, %12#

因为在C语言中规定，**变量的名字只能由字母、数字和下划线三种符号组成，并且第一个字符必须是字母或者下划线**。除了变量之外，C语言对所有的标识符（变量、符号常量、函数、数组、类型等）的命名都必须遵守这个规则。下面列出的都是合法的标识符：

sum, _sum, Class, m_total, a_1, b1

需要注意的是，**C编译系统严格区分大小写字母**，Class和class是两个不同的标识符，同样，BASIC和basic也是不相同的。一般情况下，变量名用小写字母表示，与人们的日常习惯保持一致。

在C语言中，所有的变量都必须在使用之前先定义，只有定义后的变量在程序中才能使用。

6. 输入输出函数

在程序1.1中，我们输入了两个整数，然后计算这两个整数的和并输出到屏幕上。这里的输入和输出如何实现呢？实际上，C语言本身不提供输入输出语句，输入输出操作是由C函数库中的函数来实现的。C标准库函数提供了一些输入输出函数，如程序1.1中的scanf()函数和printf()函数。在使用的时候需要注意，不要误以为它们是C语言提供的“输入输出语句”。scanf和printf不是C语言的关键字，而只是函数的名字。

printf函数的作用是输出若干个任意类型的数据，例如，程序1.1中的输出语句printf("Please input two number(like:3,5): \n")表明将“Please input two number(like:3,5):”原样输出到显示屏上并换行（"\n"）；而printf("%d + %d = %d\n", a, b, sum)则不仅将“+”和“=”原样输出，还将“%d”转换成了相应变量的值输出。这里，第一个%d转换成变量a的值24，第二个%d转换成变量b的值32，第三个%d则转换成变量sum的值56。“%d”表明输出变量的数据类型是整数。

scanf函数的作用是输入数据，程序1.1中通过scanf("%d,%d", &a, &b)将我们在键盘上输入

的两个数据24和32分别传给变量a和b。“%d”表明输出变量的数据类型是整数。“%d,%d”表明输入的两个整数之间必须用“,”隔开。需要注意的是，变量a和b必须分别表示为&a和&b，其中“&”是C语言的取地址运算符，初学者容易在这里出现错误，应加以注意。

7. 函数和函数调用

一个函数通常由两部分组成：函数的首部和函数体。函数首部包括函数名、函数类型、函数参数（形参）、参数类型。

例如，程序1.1中add函数的首部如下：

```
int      add      (int      x ,      int      y)
 ↓        ↓        ↓        ↓        ↓        ↓
函数类型  函数名  函数参数类型  函数参数名  函数参数类型  函数参数名
```

一个函数名后面必须跟一对圆括号，函数参数可以有多个，也可以没有。

函数体是紧跟函数首部下面的大括号内的部分，包括了实现函数特定功能的变量定义及若干执行语句。有时，函数体也可以没有变量定义和执行语句，只有一对大括号。

函数在被其他函数调用时，需要先声明。例如，程序1.1中，main()函数中的代码：

```
int add(int x, int y);
```

就是对被调函数add的一个声明。对函数的声明需与函数定义的函数名以及形式参数的个数、类型和顺序保持一致。当然，在函数声明的语句中，也可省略具体的形式参数名字，只给出形式参数的类型。例如：

```
int add(int, int);
```

如果被调函数的定义出现在主调函数之前，则可以不加声明。这是因为编译系统已经知道了已定义的被调函数，并根据函数定义的相关信息对函数的调用作出正确性检查。

8. 赋值

sum = add(a,b)，这行程序除了调用函数外，还有一个赋值的功能，表示将add函数的返回值赋值给变量sum。该语句之前有一个定义变量的语句“int a,b,sum”，它表明在计算机内存中为变量a、b和sum分配存储空间，而这个赋值语句则将数值存放在变量sum的存储空间中。我们也可以根据需要修改sum的值，这正是将sum称为变量的原因。需要注意的是，赋值运算符“=”的结合顺序自右向左，即将“=”右边表达式的结果赋值给左边的变量。赋值语句后面的分号也是必不可少的，C程序的每个语句后面都需要有一个分号。

随堂练习

仿照程序1.1，请编写一个程序求出从键盘上输入的三个整数之和，并输出结果。

1.2.2 运行C程序的方法

计算机硬件只能理解和执行二进制的计算机指令，用C语言编写的程序不能直接被计算机理解和执行，需要一个软件将相应的程序转换成计算机能直接理解的二进制指令序列。这种软件就是编译器（compiler）。编译器首先对源程序进行词法分析，然后进行语法和语义分析，最后生成可执行的代码。如果程序中有语法错误，编译器会直接指出程序中的语法错误，但编译器发现不了程序中的逻辑错误，必须通过程序调试才能发现。

编写一个程序需要很多工作，包括编辑程序、编译和调试等过程。目前，许多程序都有相应的编程环境，程序员可以直接在该环境中完成上述工作。

C程序（*.c或者*.cpp）编写好后，必须先对其进行编译得到目标程序文件（*.obj），再将目标程序与系统提供的库函数等进行连接，才能得到可执行的目标程序（*.exe）。为了编译、

连接和运行C程序，必须要有相应的C编译系统。目前使用的C编译系统大多都是集成环境（IDE）的，把程序的编辑、编译、连接和运行等操作全部集中在一个界面上进行。本书将以VC++ 6.0集成开发环境作为工具来进行C程序开发。

1. 进入VC++ 6.0集成环境

在Windows环境下，先找到VC++ 6.0集成环境所在的安装目录，从中找到可执行文件MSDEV.exe，直接双击（或者直接双击电脑“所有程序”中的VC++ 6.0快捷方式）就可以打开VC++ 6.0集成环境，如图1-3所示。在VC++ 6.0集成环境的上部，有一行主菜单，我们可以通过这些菜单来使用VC++ 6.0集成环境所提供的各种功能。

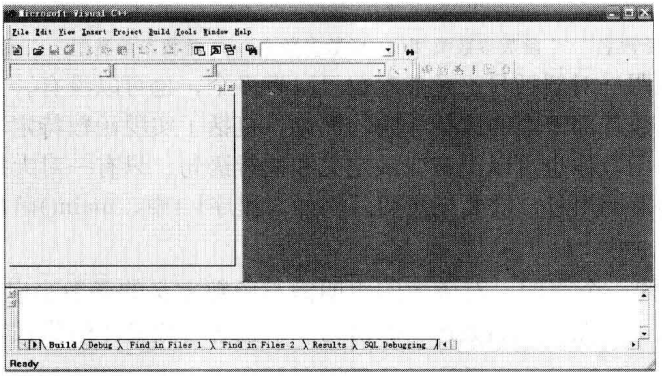


图1-3　VC++ 6.0集成环境

在图1-3所示的File菜单中选择New，出现如图1-4所示的对话框。在该对话框中，选中Win32 Console Application，在右边的Project name文本框中输入工程的名称，在Location里面选择工程保存的路径后，单击OK按钮，出现如图1-5所示的对话框。

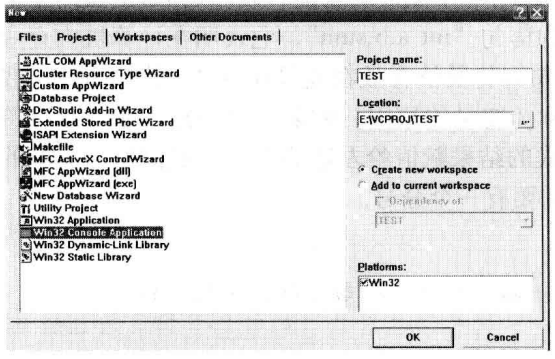


图1-4　新建工程对话框

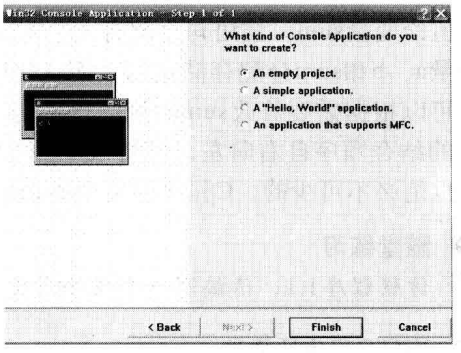


图1-5　工程创建完成对话框

在图1-5中，读者可以创建一个An empty project，在这个工程里面，所有的代码都由读者自己编写，集成开发环境不生成任何代码。然后单击Finish按钮。

现在，我们就进入了VC工程的编辑界面，如图1-6所示。TEST工程下有三个虚拟文件目录，分别用于管理工程中的各类文件。其中，Source Files存放源文件，Header Files存放源文件所需的、用户自定义的头文件，Resource Files则存放其他相关的资源文件。需要注意的是，这三个文件目录在工程存放的实际目录中是不存在的。