

高等学校“十三五”规划教材

C语言程序设计

李 聪 曾志华 江 伟 主编



中国铁道出版社有限公司
CHINA RAILWAY PUBLISHING HOUSE CO., LTD.

高等学校“十三五”规划教材

C语言程序设计

李 聪 曾志华 江 伟 主 编
朱 倩 张晓芳 副主编
聂玉峰 主 审

贵州师范学院内部使用

内 容 简 介

本书根据高等学校学生的特点,本着从基础性、实用性出发的原则编写而成,内容包括C语言及算法概述,数据类型、运算符与表达式,顺序结构程序设计,选择结构程序设计,循环结构程序设计,数组,函数,指针,结构体和枚举类型,文件等。本书深入浅出、通俗易懂,案例题型丰富,一些题目来自实际生活,有利于培养学生利用编程解决实际问题的能力,提高学生对C语言的综合实践能力。

本书适合作为高等学校C语言程序设计课程的教材,也可作为全国计算机等级考试(二级)的培训教材或对C语言程序设计感兴趣的读者的自学用书。

图书在版编目(CIP)数据

C语言程序设计 / 李聪, 曾志华, 江伟主编. —北京:
中国铁道出版社有限公司, 2019. 8
高等学校“十三五”规划教材
ISBN 978-7-113-26052-1

I. ①C… II. ①李… ②曾… ③江… 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(2019)第147150号

书 名: C语言程序设计
作 者: 李 聪 曾志华 江 伟

策 划: 徐海英
责任编辑: 翟玉峰 彭立辉
封面制作: 刘 颖
责任校对: 张玉华
责任印制: 郭向伟

编辑部电话: 010-63589185 转 2067

出版发行: 中国铁道出版社有限公司(100054, 北京市西城区右安门西街8号)

网 址: <http://www.tdpress.com/51eds/>

印 刷: 北京柏力行彩印有限公司

版 次: 2019年8月第1版 2019年8月第1次印刷

开 本: 850 mm×1 168 mm 1/16 印张: 14.75 字数: 346 千

书 号: ISBN 978-7-113-26052-1

定 价: 39.80 元

版权所有 侵权必究

凡购买铁道版图书,如有印制质量问题,请与本社教材图书营销部联系调换。电话:(010) 63550836

打击盗版举报电话:(010) 51873659

前言

C语言是当今软件开发领域广泛使用的计算机语言之一,它既具备高级语言的特性,又具有直接操纵计算机硬件的能力,并以其丰富灵活的控制和数据结构、简洁而高效的语句表达、清晰的程序结构和良好的可移植性而拥有大量的使用者。目前,各高校理工科专业大多开设了C语言程序设计课程。同时,C语言程序设计也是全国计算机等级考试(二级)科目之一。

本书是编者在多年从事C语言程序设计教学工作的基础上,结合教学实践编写而成的,凝聚了一线任课教师多年的教学经验。全书共分10章,内容包括:C语言及算法概述,数据类型、运算符与表达式,顺序结构程序设计,选择结构程序设计,循环结构程序设计,数组,函数,指针,结构体和枚举类型,文件。书中列举了学生容易出现问题的典型例题及实际生活中的题目,便于学生深入掌握重点内容,提高实践操作技能。本书体系结构安排合理、重点突出、难度适中;在语言叙述上注重概念清晰、通俗易懂,适应计算机教学实际需要。

本书各章均附有习题,供读者练习思考,以加深对书中内容的理解。另外,书中重点和难点内容均已录制成视频,读者只需扫描书中对应位置的二维码,便可以进行在线学习。同时,本书还同步推出了配套的实验教材《C语言程序设计实验指导与习题集》(李聪、朱倩、张晓芳主编),把C语言程序设计的方法融入实践环节。

本书由李聪、曾志华、江伟任主编,朱倩、张晓芳任副主编。具体编写分工:第1、2章由张晓芳编写,第3、4章由朱倩编写,第5、9、10章由江伟编写,第6、8章由曾志华编写,第7章由李聪编写。全书由李聪提出框架并负责统稿,聂玉峰主审。

在本书的编写过程中,邵平凡、黄远林、李庆、邓娟、周冰、余正红、余红珍、

杨艳霞、于海平、刘永真、李雪燕、黄丽等老师提出了许多宝贵意见，在此表示衷心感谢！

由于时间仓促，编者水平有限，书中难免存在疏漏和不妥之处，恳请广大专家、读者批评指正。

编 者

2019年4月

目 录

第 1 章 C 语言及算法概述	1	2.5.1 运算符的优先级与结合性	22
1.1 C 语言的诞生与发展	1	2.5.2 算术运算符	22
1.2 C 语言的特点	2	2.5.3 自增、自减运算符	23
1.3 C 语言的基本结构	3	2.5.4 赋值运算符	25
1.4 C 语言程序的开发过程	4	2.5.5 其他运算符	26
1.5 算法	8	2.5.6 数据类型转换	27
1.5.1 算法概述	9	习题二	29
1.5.2 算法的特征和评价	9	第 3 章 顺序结构程序设计	30
1.5.3 算法的表示方法	10	3.1 C 语句概述	30
习题一	13	3.2 C 语言中数据的输入与输出	31
第 2 章 数据类型、运算符与表达式	14	3.2.1 数据输入 / 输出的概念及在 C 语言中的实现	31
2.1 数据类型	14	3.2.2 格式输出函数——printf()	32
2.2 常见符号	15	3.2.3 格式输入函数——scanf()	36
2.3 常量	16	3.2.4 字符输出函数——putchar()	38
2.3.1 整型常量	16	3.2.5 字符输入函数——getchar()	39
2.3.2 实型常量	17	3.3 顺序结构程序设计举例	40
2.3.3 字符常量	17	习题三	42
2.3.4 字符串常量	18	第 4 章 选择结构程序设计	45
2.3.5 宏常量	18	4.1 关系运算	45
2.4 变量	19	4.1.1 关系运算符	45
2.4.1 变量的定义与初始化	19	4.1.2 关系表达式	46
2.4.2 使用变量时的注意事项	20	4.2 逻辑运算	46
2.5 运算符及表达式	22		

4.2.1 逻辑运算符	46	6.2.2 二维数组元素的初始化	88
4.2.2 逻辑表达式	47	6.2.3 二维数组的引用	89
4.3 if 语句	49	6.3 字符数组	91
4.3.1 if 语句的一般形式	49	6.3.1 字符串	91
4.3.2 if 语句的嵌套	52	6.3.2 字符数组的定义和赋值	91
4.4 条件运算符与条件表达式	54	6.3.3 字符数组的输入和输出	92
4.5 switch 语句	55	6.3.4 常见字符串处理函数	95
4.6 选择结构程序设计举例	58	6.3.5 二维字符数组	98
习题四	60	习题六	99
第 5 章 循环结构程序设计	64	第 7 章 函 数	102
5.1 概述	64	7.1 函数及其作用	102
5.2 while 语句	64	7.2 函数的分类	103
5.3 do...while 语句	66	7.3 函数的定义	105
5.4 for 语句	68	7.3.1 函数定义的一般形式	105
5.5 break 和 continue 语句	71	7.3.2 函数的参数	106
5.5.1 break 语句	71	7.3.3 函数的返回值与 return 语句	107
5.5.2 continue 语句	72	7.4 函数的调用	109
5.6 循环嵌套	73	7.4.1 函数调用的一般形式	109
5.7 循环结构程序设计举例	74	7.4.2 函数调用的方式	109
习题五	76	7.4.3 函数的声明	110
第 6 章 数 组	81	7.5 函数的嵌套调用	111
6.1 一维数组	81	7.6 函数的递归调用	113
6.1.1 一维数组的定义	81	7.7 数组作为函数参数	116
6.1.2 一维数组元素的引用	82	7.7.1 数组元素作为函数实参	116
6.1.3 一维数组元素的初始化	83	7.7.2 数组名作为函数参数	117
6.1.4 一维数组的常见操作	84	7.8 局部变量和全局变量	120
6.1.5 一维数组的应用	86	7.8.1 局部变量	120
6.2 二维数组	87	7.8.2 全局变量	122
6.2.1 二维数组的定义	87		

7.9 变量的存储类别	124	第 9 章 结构体和枚举类型	157
7.9.1 静态存储方式与动态存储 方式	124	9.1 概述	157
7.9.2 自动变量	125	9.2 结构体类型的声明	158
7.9.3 外部变量	125	9.3 定义结构体变量	159
7.9.4 寄存器变量	126	9.3.1 结构体变量的定义	159
7.9.5 静态变量	127	9.3.2 结构体类型的嵌套定义	161
7.9.6 存储类型总结	129	9.3.3 结构体变量的初始化	162
习题七	130	9.3.4 结构体变量的引用	163
第 8 章 指 针	135	9.4 结构体数组	165
8.1 地址与指针的概念	135	9.4.1 结构体数组的定义	165
8.1.1 地址的基本概念	135	9.4.2 结构体数组的初始化	166
8.1.2 内存的访问方式	135	9.5 指向结构体类型变量的指针	168
8.2 指针变量	136	9.5.1 结构体指针变量的定义 和初始化	168
8.2.1 指针变量的定义	136	9.5.2 通过结构体指针变量引用 结构体变量成员	169
8.2.2 指针变量的引用	137	9.5.3 指针变量作为结构体成员	170
8.2.3 指针变量作为函数参数	138	9.5.4 指向结构体数组的指针	170
8.3 指针与一维数组	141	9.5.5 向函数传递结构体变量 的值	172
8.3.1 指向数组元素的指针	141	9.6 枚举类型	174
8.3.2 通过指针引用数组元素	142	9.6.1 枚举类型的定义	175
8.3.3 数组名作函数参数	144	9.6.2 枚举变量的定义和使用	175
8.4 字符串与指针	146	习题九	177
8.5 指针与二维数组	149	第 10 章 文 件	181
8.5.1 指针与二维数组的关系	149	10.1 文件概述	181
8.5.2 指向一维数组的指针变量	150	10.2 文件类型指针	183
8.6 指针数组	151	10.3 文件的打开和关闭	183
8.6.1 指针数组	151		
8.6.2 指向指针的指针	153		
习题八	154		

10.3.1 文件的打开 (fopen() 函数)	184	10.6 文件检测函数	195
10.3.2 文件的关闭 (fclose() 函数)	185	习题十	195
10.4 文件读 / 写操作	186	附 录	198
10.4.1 按字符读 / 写文件	186	附录 A ASCII 码表	198
10.4.2 字符串读 / 写函数	188	附录 B 运算符的优先级和结合性	203
10.4.3 按数据块读 / 写文件	189	附录 C C 语言中的关键字	204
10.4.4 按格式读 / 写文件	191	附录 D 常用的 C 语言库函数	205
10.5 文件的随机读 / 写	192	附录 E 各章习题参考答案	209
		参考文献	228

第 1 章 C 语言及算法概述

在当今世界，了解与运用计算机知识是走向成功的至关重要的前提条件。同时，程序与程序设计是使用计算机进行开发的核心，没有程序的计算机可以说是毫无价值的。本章主要简单地介绍计算机语言、C 语言发展与特点、C 语言程序的开发过程以及算法的知识。

1.1 C 语言的诞生与发展

语言，是人类最重要的交际工具，是人们进行沟通交流的各种表达符号。语言就广义而言，是一套共同采用的沟通符号、表达方式与处理规则，符号会以视觉、声音或者触觉方式来传递。狭义上的语言是指人类沟通所使用的语言——自然语言。一般人都必须通过学习才能获得语言能力，学习语言的目的是交流观念、意见、思想等。当人类发现了某些动物能够以某种方式沟通时，就诞生了动物语言的概念。计算机诞生以后，人类需要给予计算机指令，这种指令就是通常所说的计算机语言。

计算机程序设计语言指用于人与计算机之间进行通信的语言，是人与计算机之间传递信息的媒介。人的指令通过计算机语言传达给计算机，计算机按照其对指令的理解进行相应的操作。一般来说，计算机语言由“字符”和“语法规则”组成，由这些字符和语法规则组合而成的计算机的各种指令（以语句形式体现）就是计算机语言。

计算机语言的种类很多，总体来说可以分为机器语言、汇编语言和高级语言，其中只有机器语言可以被计算机硬件直接识别，而其他两种语言都需要翻译成机器语言才能够使计算机执行相关操作。

(1) 机器语言：直接用二进制代码指令表达的计算机语言，指令是用 0 和 1 组成的一串代码，它们有一定的位数，并分成若干段，各段的编码表示不同的含义。

(2) 汇编语言：面向机器底层的程序设计语言。在汇编语言中，用助记符代替了机器指令的操作码，用地址符号或标号代替了指令或操作数的地址，这样就增强了程序的可读性并简化了代码的编写难度。汇编语言亦称为符号语言。

(3) 高级语言：所谓高级语言是相对于汇编语言而言的。由于汇编语言依赖于硬件体系，且助记符量大而难记，于是人们又发明了更加易用的高级语言。其语法和结构更类似普通英文，且由于远离对硬件的直接操作，使得一般人经过学习之后都可以掌握。本书所要介绍的 C 语言

就是一种高级语言。

这 3 种语言在时间上有着继承和发展的关系，从机器语言到高级语言，计算机语言变得越来越接近于人类语言（自然语言），从而使得编程变得越来越简单。计算机语言的发展对计算机的普及具有十分重要的贡献。

未来的计算机语言的发展将不再是单纯的语言标准，其趋势是完全面向对象、更易表达现实世界和更易为人编写。计算机语言将不仅被专业编程人员使用，人们完全可以用编程的手段定制真实生活中的一项工作流程。

在 C 语言诞生以前，系统软件主要是用汇编语言编写的。由于汇编语言程序依赖于计算机硬件，其可读性和可移植性都很差；但一般的高级语言又难以实现对计算机硬件的直接操作（这正是汇编语言的优势），于是人们盼望有一种兼有汇编语言和高级语言特性的新语言。

C 语言是贝尔实验室于 20 世纪 70 年代初研制出来的，后来又被多次改进，并出现了多种版本。80 年代初，美国国家标准化协会（ANSI）根据 C 语言问世以来各种版本对 C 语言的发展和扩充，制定了 ANSIC 标准（1989 年再次做了修订）。

目前，在微机上广泛使用的 C 语言编译系统有 Microsoft C、Turbo C、Borland C 等。虽然它们的基本部分都是相同的，但还是有一些差异，所以要注意所使用的 C 编译系统的特点和规定。

1.2 C 语言的特点

C 语言作为一种计算机语言，有其个性、独特之处，也有其不足的地方。下面介绍 C 语言的特点。

1. 简洁紧凑、灵活方便

C 语言一共有 32 个关键字、9 种控制语句，程序书写自由，主要用小写字母表示。C 语言以接近英语国家的自然语言和数学语言为语言的表达形式，容易理解。C 语言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，可以像汇编语言一样对位、字节和地址进行操作，而这三者是计算机最基本的工作单元。

2. 运算符丰富

C 语言的运算符包含的范围很广泛，共有 34 个运算符。C 语言把括号、赋值、强制类型转换等都作为运算符处理，从而使其运算类型极其丰富，表达式类型多样化。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。

3. 数据类型丰富

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数据类型的运算。C 语言也引入了指针概念，使程序效率更高。

4. 结构化程序设计语言

C 语言具有结构化程序设计语言所要求的三大基本结构，层次清晰，逻辑性强，便于维护、调试。

5. 程序设计自由度高

一般的高级语言语法检查比较严，几乎要求检查出所有的语法错误。而 C 语言允许程序编写者有较大的自由度，客观上降低了对程序员的要求，但这种不严格事实上也给程序留下了出

现一些潜在错误的可能性，降低了程序的健壮性。

6. 允许直接访问物理地址

在计算机世界中，位（bit）是最小的单位，1 bit 就是 1 个二进制位，只有两种取值：0 和 1。C 语言能进行位运算，能实现汇编语言的大部分功能，能对硬件直接操作。很多嵌入式系统中的单片机都提供 C 语言编译器，如 51 系列单片机、MSP430、ARM 等。

7. 程序生成代码质量高

机器语言是能被计算机直接执行的语言，效率最高。汇编语言次之，基本接近于机器语言的效率。而 C 语言一般只比用汇编程序生成的目标代码效率低 10% ~ 20%。

8. 可移植性好

C 语言的一个突出优点就是适合于多种操作系统（如 DOS、UNIX），也适用于多种机型。用 C 语言编写的程序不需要做很多改动就可以从一种机型移到另一种机型上运行。

总之，C 语言既具有高级语言的特点，又具有低级语言的特点；既是一种成功的系统设计语言，又是一种实用的程序设计语言；既能用来编写不依赖计算机硬件的应用程序，又能用来编写各种系统程序。尽管 C 语言也有不足，如对数组下标越界不进行检查、对变量类型约束不严格等，但它仍是一种很受欢迎、应用广泛的程序设计语言。

1.3 C 语言的基本结构

前两节介绍了 C 语言的诞生、发展及特点，那么 C 语言具有什么样的结构？本节将通过一个简单的例子来说明 C 语言的基本结构。

【例 1.1】 在屏幕上显示“Hello World！”。

```
01 #include <stdio.h>
02 int main()      /* 主函数中实现在屏幕输出一行字符串 "Hello World!" */
03 {
04     printf("Hello World!");    // 输出字符串 Hello World!
05     return 0;
06 }
```

通过以上程序，总结如下：

1. 函数是 C 程序的基本单位

函数是 C 程序的基本单位，即 C 程序是由函数构成的。

(1) 一个 C 程序必须有且仅有一个用 main 作为名字的函数，这个函数通常称为主函数。C 程序总是从 main() 函数开始执行，最后由 main() 函数结束，而与它在程序中的位置无关。

(2) 根据需要，一个 C 程序可以包含零到多个用户自定义函数。关于自定义函数的相关知识将在第 7 章进行详细介绍。

(3) 函数中可以调用系统提供的库函数，在调用之前需要将相应的头文件包含到本文件中。例如，使用系统输入/输出函数时，需要用编译预处理命令：

```
#include <stdio.h>
```

将头文件 stdio.h 包含到本文件中。stdio.h 是 C 编译程序提供的许多头文件之一，其中含有标准

输入/输出函数。C 语言本身并没有输入/输出语句，输入/输出功能需要通过函数来实现，而用到哪种类型的函数，就需要包含相应类型的头文件。程序中调用标准输入与输出函数之前，必须先写上 `#include<stdio.h>` 预处理命令，并独占一行，一般写在源程序文件的开始部分。

2. 函数由函数首部和函数体两部分组成

(1) 函数首部包括对函数返回值类型、函数名、形参类型、形参名的说明，具体内容详见第 7 章。当然，有时函数也可以没有形参，这时函数名后的一对圆括号不能省略，如例 1.1 中对主函数 `main()` 的定义。

(2) 函数体由函数首部下面最外层的一对花括号中的内容组成，包括变量声明语句和执行语句。变量声明语句是对函数中“对象”的描述，执行语句是对函数所要实现的“动作”的描述，由一系列可执行语句组成。

3. C 程序的书写格式与规范

(1) 除复合语句外，C 语句都是以分号作为结束标志的。注意，本书程序中语句前面的行号并非程序语句的一部分，只是为了便于解释程序语句的功能而额外添加的。

(2) C 程序的书写格式比较自由，既允许在一行内写多条语句，也允许将一条语句分写在多行，而不必加任何标识。为了提高程序的可读性和可测试性，建议读者模仿本书例题的书写格式书写程序，一行内只写一条语句，养成良好的、规范的程序设计风格。

(3) 在 C 程序中还有一些用“/*”和“*/”包含起来的内容，被称为注释，是对程序功能的必要说明和解释。

C 编译程序并不对注释的内容进行语法检查，可用英文也可以用汉字来书写注释内容。分为单行注释和多行注释（也称为块注释）。

C99 允许使用双斜线“//”进行单行注释，只要字符“//”在一对引号之外的任何地方出现，一行中双斜线后面的部分都会被处理为注释。

多行注释由“/*”开头，由“*/”结尾，两者中间的内容都被当作注释来处理。写注释时应注意左斜线“/”和星号“*”之间不能留有空格。

虽然有无注释并不影响程序的功能和正确性，但由于注释能起到“提示”代码的作用，有助于读者更快、更好地理解程序，提高程序的可读性。因此，规范化程序设计提倡给程序添加必要的注释。

1.4 C 语言程序的开发过程

如果没有一个完整的、交互性良好的集成开发环境（Integrated Development Environment, IDE），C 语言很难解决实际问题。Windows 环境下，适合 C 语言开发的 IDE 很多，如 CodeBlocks、VC++6.0、Turbo C、Dev-C++ 和 Visual Studio 2015 等，这里介绍在 Visual Studio 2015 集成开发环境下的 C 语言开发过程。

任何语言都可以理解为一种用于交流的工具。例如，汉语，用于会说汉语的人之间进行交流；英语，用于会说英语的人之间进行交流。而 C 语言可以理解为一种用于人与计算机交流的一种工具，用于解决实际的问题。正确安装 Visual Studio 2015 后，便可以在其上进行 C 语言程序的开发。

微课



视频 1-1
C 程序基本结构

本节以屏幕上输出“Hello World!”为例，讲解 C 语言程序在 Visual Studio 2015 环境下的开发步骤。

(1) 新建项目。打开 Visual Studio 2015 环境，选择“文件”→“新建”→“项目”命令，如图 1.1 所示。



图 1.1 新建项目

(2) 选择模板。在弹出的对话框中选择 Visual C++ 选项下方的 Win32 子选项，在右侧窗口中选中 Win32 控制台应用程序选项，并在工程名称位置填写项目的名称，如 Test01。然后，在位置处选择该项目存放的位置，这里存放在 E 盘下的 CProgram 文件夹内，如图 1.2 所示。



图 1.2 “新建项目”对话框

(3) 单击“确定”按钮,弹出“Win32 应用程序向导”对话框,单击“下一步”按钮,如图 1.3 所示。

(4) 在 Win32 应用程序向导中选择“应用程序类型”为“控制台应用程序”,在“附加选项”中选中“空项目”复选框,然后单击“完成”按钮即可,如图 1.4 所示。



图 1.3 “Win32 应用程序向导”对话框



图 1.4 设置应用程序类型及附加选项

(5) 在 Visual Studio 2015 主窗口的解决方案资源管理器中右击“源文件”选项,在弹出的快捷菜单中选择“添加”→“新建项”命令,如图 1.5 所示。



图 1.5 添加新建项

(6) 添加源文件。在弹出的“添加选项”对话框中选择“Visual C++”,然后在右侧选择“C++ 文件(.cpp)”,并在对话框下方填写项目名称(如 Test01.c),最后单击“添加”按钮,如图 1.6 所示。

(7) 编写代码。在程序主视图下编写 C 语言代码,如图 1.7 所示。



图 1.6 添加源文件

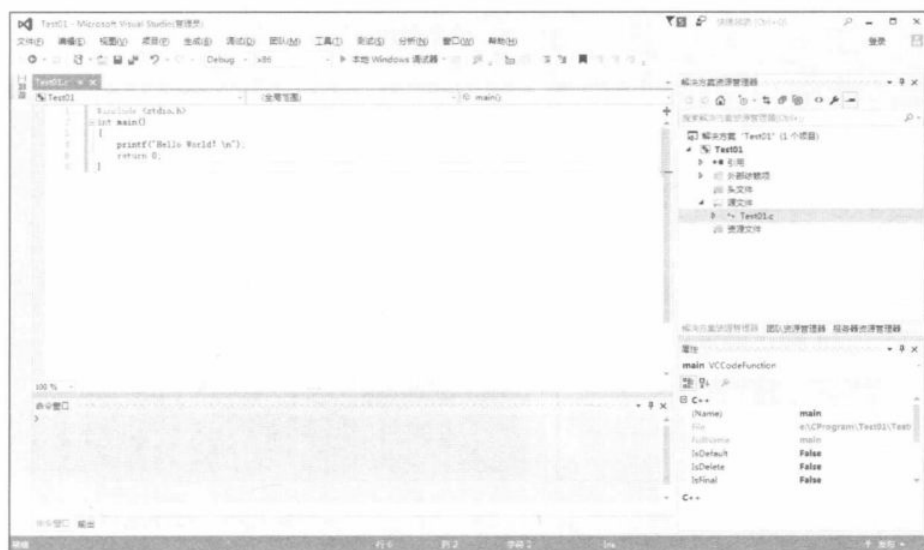


图 1.7 编写代码

程序代码如下：

```
01 #include <stdio.h>
02 int main()
03 {
04     printf("Hello World!\n");
05     return 0;
06 }
```

(8) 编译程序。选择“生成”→“编译”命令，对源代码进行编译。编译成功时，主窗口的下方“输出”窗口中会显示“成功 1 个，失败 0 个，最新 0 个，跳过 0 个”，如图 1.8 所示。

(9) 执行程序。选择“调试”→“开始执行(不调试)”命令，执行程序，如图 1.9 所示。

程序运行结果如图 1.10 所示。

通过以上详细讲解,可以总结出 C 语言程序开发过程:编辑(.c 文件)—编译(.obj 文件)—连接(.exe 文件)—运行。



图 1.8 编译程序



图 1.9 执行程序



图 1.10 程序运行结果

注意:

编写程序一定要先保存,再进行编译,否则如果遇到异常,之前编写的程序或修改的程序不会保存。

1.5 算 法

一般情况下,程序由数据结构和算法两部分组成,由此可见,算法在程序设计中具有举足轻重的地位。一个好的程序必定包含着一个或多个算法。本节主要介绍算法的基本概念、算法的特征和算法性能的评价标准。