



全国高等职业教育规划教材

C语言程序设计 实例教程

李红 王强 主编



电子教案下载网址 www.cmpedu.com



机械工业出版社
CHINA MACHINE PRESS

全国高等职业教育规划教材

C 语言程序设计实例教程

主 编 李 红 王 强

副主编 董 萍 董佳佳 许 宁

参 编 王 飞 王淑芬 张 兵 张春丽



机械工业出版社

本书从实例出发引出各章节知识点, 主要介绍 C 语言编程的基础知识和操作方法, 帮助学生掌握使用 Win-TC 进行程序设计的方法。全书共分 10 章, 内容涵盖了 C 语言基础知识、基本程序设计、数组、函数、指针、结构体、链表、共用体、位运算、文件等方面的知识。本书注重学生上机调试程序能力的培养, 每节后设有课堂精练, 每章后设有实训和习题, 以便学生进一步巩固本章所学知识。

本书以职业能力的培养为出发点, 突出“以学生为中心”的教育理念, 遵循“实例举例—知识点梳理—课堂精练—课后习题”的模式, 重在全面培养学生的多元能力。

本书可作为高职高专“C 语言程序设计”课程的教材, 也可作为 C 语言培训机构的培训教材, 还可作为 C 语言初学者的自学教材。

本书配有授课电子课件, 需要的教师可登录 www.cmpedu.com 免费注册、审核通过后下载, 或联系编辑索取(QQ: 81922385, 电话: 010-88379739)。

图书在版编目(CIP)数据

C 语言程序设计实例教程/李红, 王强主编. —北京: 机械工业出版社, 2010. 9

全国高等职业教育规划教材

ISBN 978 - 7 - 111 - 31648 - 0

I. ①C… II. ①李…②王… III. ①C 语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 164880 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 鹿 征

责任印制: 杨 曦

北京蓝海印刷有限公司印刷

2010 年 9 月第 1 版·第 1 次印刷

184mm×260mm·15.5 印张·382 千字

0001—3000 册

标准书号: ISBN 978 - 7 - 111 - 31648 - 0

定价: 27.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

电话服务

社服务中心: (010)88361066

销售一部: (010)68326294

销售二部: (010)88379649

读者服务部: (010)68993821

网络服务

门户网: <http://www.cmpbook.com>

教材网: <http://www.cmpedu.com>

封面无防伪标均为盗版

全国高等职业教育规划教材计算机专业 编委会成员名单

主 任 周智文

副 主 任 周岳山 林 东 王协瑞 张福强
陶书中 龚小勇 王 泰 李宏达
赵佩华 陈 晴

委 员 （按姓氏笔画排序）

马 伟 马林艺 卫振林 万雅静
王兴宝 王德年 尹敬齐 卢 英
史宝会 宁 蒙 刘本军 刘新强
刘瑞新 余先锋 张洪斌 张 超
杨 莉 陈 宁 汪赵强 赵国玲
赵增敏 贾永江 陶 洪 康桂花
曹 毅 眭碧霞 鲁 辉 裴有柱

秘 书 长 胡毓坚

出版说明

根据《教育部关于以就业为导向深化高等职业教育改革的若干意见》中提出的高等职业院校必须把培养学生动手能力、实践能力和可持续发展能力放在突出的地位,促进学生技能的培养,以及教材内容要紧紧密结合生产实际,并注意及时跟踪先进技术的发展等指导精神,机械工业出版社组织全国近 60 所高等职业院校的骨干教师对在 2001 年出版的“面向 21 世纪高职高专系列教材”进行了全面的修订和增补,并更名为“全国高等职业教育规划教材”。

本系列教材是由高职高专计算机专业、电子技术专业和机电专业教材编委会分别会同各高职高专院校的一线骨干教师,针对相关专业的课程设置,融合教学中的实践经验,同时吸收高等职业教育改革的成果而编写完成的,具有“定位准确、注重能力、内容创新、结构合理和叙述通俗”的编写特色。在几年的教学实践中,本系列教材获得了较高的评价,并有多品种被评为普通高等教育“十一五”国家级规划教材。在修订和增补过程中,除了保持原有特色外,针对课程的不同性质采取了不同的优化措施。其中,核心基础课的教材在保持扎实的理论基础的同时,增加实训和习题;实践性较强的课程强调理论与实训紧密结合;涉及实用技术的课程则在教材中引入了最新的知识、技术、工艺和方法。同时,根据实际教学的需要对部分课程进行了整合。

归纳起来,本系列教材具有以下特点:

- 1) 围绕培养学生的职业技能这条主线来设计教材的结构、内容和形式。
- 2) 合理安排基础知识和实践知识的比例。基础知识以“必需、够用”为度,强调专业技术应用能力的训练,适当增加实训环节。
- 3) 符合高职学生的学习特点和认知规律。对基本理论和方法的论述要容易理解、清晰简洁,多用图表来表达信息;增加相关技术在生产中的应用实例,引导学生主动学习。
- 4) 教材内容紧随技术和经济的发展而更新,及时将新知识、新技术、新工艺和新案例等引入教材。同时注重吸收最新的教学理念,并积极支持新专业的教材建设。
- 5) 注重立体化教材建设。通过主教材、电子教案、配套素材光盘、实训指导和习题及解答等教学资源的有机结合,提高教学服务水平,为高素质技能型人才的培养创造良好的条件。

由于我国高等职业教育改革和发展的速度很快,加之我们的水平和经验有限,因此在教材的编写和出版过程中难免出现问题和错误。我们恳请使用这套教材的师生及时向我们反馈质量信息,以利于我们今后不断提高教材的出版质量,为广大师生提供更多、更适用的教材。

机械工业出版社

前 言

由于 C 语言兼备高级语言和低级语言的特点，所以既适合开发系统软件，又适合开发应用软件。其语法灵活、书写格式自由、易学易用，深受广大程序设计人员的青睐。

本书从高职高专的实际情况出发，围绕全国计算机等级考试的知识点来确定章节内容。在实例选取上，力求做到让复杂问题简单化，让简单问题实用化，旨在树立学生的程序设计思想，培养学生编写与调试程序的能力，突出“以学生为中心”的教育理念。本书的编写遵循“实例举例—知识点梳理—课堂精练—课后习题”的模式，深入浅出，充分培养学生的创新能力、实践能力和自学能力。

在时间安排上，建议采用课堂讲授、上机实践、课后练习相结合的方式，讲授时间约占 50%，上机学习与练习时间约占 50%。由于本书的编者都是高职高专院校的一线教师，均长期从事 C 语言课程的教学与科研工作，不仅具有丰厚的 C 语言的专业功底，而且对高职高专学生的特点、认知能力和学习情况等方面都有深入了解。

本书由李红、王强任主编，董萍、董佳佳和许宁任副主编。其中第 1、2 章由北京工业大学许宁编写，第 3 章由山东工业职业技术学院董佳佳编写，第 4、5 章由三门峡职业技术学院董萍编写，第 6、7 章由北京信息职业技术学院李红编写，第 8、9、10 章由吉林电子信息职业技术学院王强编写。目录、附录等内容由王飞、王淑芬、张兵、张春丽整理。

由于编者水平有限，书中难免有疏漏和不妥之处，敬请广大读者批评指正。

编 者

目 录

出版说明

前言

第 1 章 C 语言概述与程序逻辑	1
1.1 C 语言概述	1
实例 1 C 语言简介——简单的 C 语言应用程序	1
实例 2 C 语言环境使用介绍——用“*”输出字母“A”形状	3
1.2 程序和程序逻辑	6
实例 3 程序与算法的概念——梵塔推理	7
实例 4 算法图形描述——求 $n!$	8
1.3 课后习题	10
1.3.1 实训	10
1.3.2 习题	10
第 2 章 C 语言基础知识	12
2.1 常量和变量	12
实例 5 常量和变量——输出常量与变量的值	12
2.2 数据类型	15
实例 6 基本数据类型——输出不同数据类型的值	15
实例 7 不同数据类型间的转换——强制类型转换练习	18
2.3 常用的运算符和表达式	20
实例 8 运算符及表达式——输出各表达式的结果	21
实例 9 关系、逻辑运算符和表达式——关系运算和逻辑运算的结果	23
实例 10 运算符的优先级——复杂表达式的运算结果	25
2.4 课后习题	27
2.4.1 实训	27
2.4.2 习题	27
第 3 章 基本程序设计	31
3.1 顺序结构程序设计	31
实例 11 字符型数据的输入和输出——顺序输出字符	31
实例 12 格式化数据的输入和输出——互换两个变量的值	32
3.2 选择结构程序设计	38
实例 13 if 语句——根据条件确定公司是否已经为司机投保	38
实例 14 switch 语句——计算长方形、圆形和三角形的面积	41
3.3 循环结构程序设计	45
实例 15 while 循环语句——求 $1+2+\dots+100$ 的和	45
实例 16 for 循环语句——统计大写字母和小写字母的个数	46

实例 17	do-while 循环语句——求 $1+2+\dots+100$ 和 $1^2+2^2+\dots+30^2$ 的值	48
实例 18	循环的嵌套结构——百钱买百鸡问题	49
3.4	课后习题	52
3.4.1	实训	52
3.4.2	习题	52
第 4 章	数组	55
4.1	一维数组	55
实例 19	一维数组的定义与引用——平均成绩的统计	55
实例 20	一维数组的初始化与引用——查询数据中的最大值	58
4.2	二维数组	60
实例 21	二维数组的定义与引用——统计总成绩及平均成绩	60
实例 22	二维数组的初始化与引用——求矩阵的乘积	63
4.3	字符数组与字符串	66
实例 23	字符数组的定义与引用——字母替换	66
4.4	课后习题	70
4.4.1	实训	70
4.4.2	习题	70
第 5 章	函数	73
5.1	函数的定义与返回值	73
实例 24	函数的定义与引用——判断当天是该年的第几天	73
5.2	函数的调用	76
实例 25	函数的调用——求两个整数之和	76
实例 26	函数的参数形式——求 $1!+2!+3!+\dots+n!$ 的值	77
5.3	函数的嵌套与递归调用	80
实例 27	函数的嵌套调用——编程计算 $(1!)^2+(2!)^2+(3!)^2+(4!)^2+(5!)^2$ 的值	80
实例 28	函数的递归调用——求 $n!$	81
5.4	函数中标识符的作用域与存储类	84
实例 29	函数中标识符的作用域——初识局部变量与全局变量	85
实例 30	函数中标识符的存储类——打印 1~5 的阶乘值	85
5.5	宏替换与文件包含	89
实例 31	不带参数的宏定义——已知半径求周长和面积	89
实例 32	带参数的宏定义——宏名替换表达式	90
实例 33	文件包含——计算两个整数绝对值阶乘的差值	92
5.6	课后习题	94
5.6.1	实训	94
5.6.2	习题	94
第 6 章	指针	98
6.1	指针变量的定义与引用	98
实例 34	指针变量的定义与引用——寻找变量在内存中的“家”	98

6.2 一维数组与指针	103
实例 35 数组名和数组元素的地址——按序输出内存中各“家”的地址	103
实例 36 通过指针引用一维数组元素——本周和下周做值日的同学安排	104
6.3 二维数组和指针	107
实例 37 二维数组名和数组元素的地址——二维数组的成员介绍	108
实例 38 指针数组和行指针——二维数组与指针	109
6.4 函数间参数的传递	114
实例 39 指针变量作为函数的参数——交换两个变量的值后找出最大值	114
实例 40 函数之间传递地址——打印杨辉三角	115
实例 41 指向函数的指针变量——两个数的和值除以差值	117
6.5 指针与字符串	120
实例 42 字符串的存储形式——统计各类字符的个数	121
实例 43 二维数组存储多个字符串——图书查询系统	123
实例 44 字符串常用函数的使用——输出你的姓名	124
6.6 课后习题	128
6.6.1 实训	128
6.6.2 习题	128
第 7 章 结构体和共用体	133
7.1 结构体	133
实例 45 结构体类型变量、指针变量的定义与引用——我的个人信息	133
实例 46 结构体类型的数组的定义与引用——成绩统计	136
实例 47 函数之间结构体类型变量的数据传递——输出排序后的姓名和学号	138
7.2 链表	141
实例 48 单向链表的建立——输出 5 名同学的信息	141
实例 49 动态链表的建立及常用操作——输出学生的信息	143
实例 50 链表的操作——学生信息管理系统	145
7.3 共用体与枚举类型	152
实例 51 共用体类型的定义与变量引用——灵活应用存储空间	152
实例 52 枚举类型——输出给定月份的天数	154
7.4 课后习题	157
7.4.1 实训	157
7.4.2 习题	158
第 8 章 位运算	162
8.1 位运算	162
实例 53 位运算符与运算功能——两个数的几种位运算	162
8.2 位段	167
实例 54 位段——两个数的位运算	167
8.3 课后习题	172
8.3.1 实训	172

8.3.2 习题	172
第 9 章 文件	174
9.1 文件	174
实例 55 文件的概念——读/写文件	174
实例 56 文件的打开与关闭——奇偶数的不同去向	176
实例 57 文件的读/写——将数据写入文件	180
实例 58 文件的定位与检测函数——字母定位与逆序输出	184
9.2 课后习题	188
9.2.1 实训	188
9.2.2 习题	188
第 10 章 综合实训	191
10.1 实例 1——学生成绩管理系统	191
10.1.1 项目实训目的	191
10.1.2 系统功能描述	191
10.1.3 系统总体设计	192
10.1.4 程序实现	195
10.2 实例 2——电子时钟	208
10.2.1 项目实训目的	208
10.2.2 系统功能描述	208
10.2.3 系统总体设计	208
10.2.4 程序实现	211
10.3 实例 3——俄罗斯方块游戏	216
10.3.1 项目实训目的	216
10.3.2 系统功能描述	216
10.3.3 系统总体设计	217
10.3.4 程序实现	220
10.4 实训	229
附录	230
附录 A 标准 ASCII 码字符集	230
附录 B C 语言关键字	232
附录 C 常用的 C 库函数	233
参考文献	238

第 1 章 C 语言概述与程序逻辑

1.1 C 语言概述

学习目标

- 了解 C 语言的工作机制
- 掌握 C 语言程序的基本构成
- 了解 C 语言程序的编译过程
- 熟悉 Dev-C++ 的运行环境

实例 1 C 语言简介——简单的 C 语言应用程序

实例任务

输出一个表达式“1+1”的结果，然后再输出一个字符串“Hello World!”，输出一串汉字“北京欢迎您!”。程序的运行结果如图 1-1 所示。



图 1-1 程序运行结果

程序代码

```
#include "stdio.h"
/*当引用一些输入输出函数时，要在程序开始使用此语句*/
main() /*C 语言程序的主函数，程序从这里开始执行*/
{
    int i; /*变量声明*/
    i=1+1; /*执行语句*/
    printf("1+1=%d\n",i);
    /*输出 1+1 的结果 2，“\n”表示输出时将插入点光标移到下一行的起始位置*/
    printf("Hello World!\n");
    /*输出字符串：Hello World! */
    printf("北京欢迎您!\n");
    /*输出一串汉字：北京欢迎您! */
    getch();
    /*Win-TC 环境下，输出时使用此语句显示输出框*/
}
```

相关知识

1. C 语言的发展历程

在程序中，`main`、`print`、`include` 等都是大家熟知的英文单词，而计算机是不识别这些单词的。实际上，在 C 语言产生之前，人们编写系统软件主要是使用汇编语言。由于用汇编语言编写的程序依赖于计算机硬件，所以其可读性和可移植性都比较差；而一般高级语言又不具备低级语言能直观地对硬件实现控制和操作、使程序快速执行的特点。在这种情况下，人们迫切需要一种既有一般高级语言特性，又有低级语言特性的语言。于是 C 语言就应运而生了。

C 语言的产生和发展与 UNIX 操作系统有很大的关系，其发展历程简述如下。

1972~1973 年间，美国贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计出了 C 语言。当时的 C 语言只是为描述和实现 UNIX 操作系统的一种工作语言，且只在贝尔实验室内部使用。

1973 年，K.Thompson 和 D.M.Ritchie 两人合作，将 UNIX 操作系统 90%以上的代码用 C 语言改写，即 UNIX 第 5 版。

1975 年，UNIX 操作系统第 6 版公布后，C 语言突出的优点引起了人们的普遍关注。

1977 年，出现了可移植的 C 语言。

1978 年，UNIX 操作系统第 7 版公布，K.Thompson 和 D.M.Ritchie 以该版 C 编译程序为基础，合著《The C Programming Language》一书。该书所介绍的 C 语言成为后来人们广泛使用的 C 语言版本的基础，被称为标准 C 语言。

1983 年，美国国家标准化协会（ANSI）根据 C 语言问世以来的各种版本，对 C 语言进行发展和扩充，并制定了新的标准，称为 ANSI C。

1990 年，国际标准化组织（ISO）制定了 ISO C 标准。

1972 年以来，C 语言几经修改和发展，出现了多个版本。C 语言是国际上广泛流行的计算机高级语言，既可用于写系统软件，也可用于写应用软件。目前在微机机上使用的版本有多个，各有特点，但它们一般都是以 ANSI C 为基础编写的，其中比较常用的版本有 Microsoft C、Quick C、Turbo C、WinTC、C-Free、Dev-C++等。

2. C 语言程序的构成

C 语言应用程序是由函数构成的，以 `main()` 函数作为入口开始执行应用程序。`main()` 是 C 程序的入口函数，每个 C 程序必须有 `main()` 函数，且每个 C 程序只能有一个 `main()` 函数。

{ } 括起来的部分，称为函数体，是函数的执行部分。函数体中，每条以分号“;”结尾的元素，称为语句。C 语言的语句必须以分号结尾。语句可以一行写一条，也可以一行写多条，一个单独的“;”可以自成为一条语句。

程序中的第一行语句：`#include "stdio.h"`或`#include <stdio.h>`，用于告诉编译器在本程序中包含标准输入/输出库的信息。函数体中 `printf` 函数是一个用于打印输出的库函数，后面小括号中的内容为这个函数的参数。

函数体中，语句“`int i;`”是在进行变量 `i` 的定义。变量是内存中用于存放数据的元素，必须先定义，后引用。

程序中“`/*.....*/`”符号中所描述的内容，属于某语句或某段程序的注释，为非执行语句，起到帮助读者理解程序的作用。

3. 程序的执行方式

计算机只能识别机器语言（即二进制代码），例如：11011010。但用二进制代码编程难以记忆、检错，故只能用高级语言编程。因此，计算机须完成一个翻译过程，即将高级语言翻译成机器语言，如图 1-2 所示。

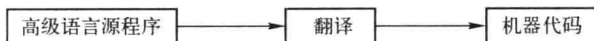


图 1-2 程序翻译过程示意图

这种翻译过程分 3 种情况。一种是汇编程序，其语言源程序代码与机器指令一一对应；另一种是编译程序，按某种约定将源代码翻译为目标代码，通过连接程序将目标程序与所调用的标准函数库连接为一体，然后执行程序，如常用的 Pascal、C 语言等；还有一种是解释程序，与编译程序思想相似，但是一条一条翻译，译出一句执行一句，如早期的 BASIC 语言。

对于编译程序，编译执行是在编写完代码后，通过特别的工具软件将源代码编译成目标代码进而转换成机器代码（即可执行程序），然后直接交给操作系统执行。也就是说，程序是作为一个整体来运行的。这类程序的优点是执行速度比较快，编译连接之后可以独立在操作系统上运行，不需要其他应用程序的支持；缺点是不利于调试，每次修改之后都要执行编译连接等步骤，才能看到其执行结果。此外，有些集成开发环境与操作系统之间存在一定的依赖性，不同操作系统需要的编译器可能不同，因此在一个系统上编译的程序到另外一个系统上并不一定能够运行。

C 语言的编译执行过程如图 1-3 所示。

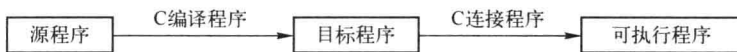


图 1-3 编译执行示意图

4. C 语言的特点

1) 语言简洁，使用方便灵活。C 语言关键字少，ANSI C 标准总共只有 32 个关键字和 9 种控制语句，压缩了一切不必要的成分。C 语言的书写形式比较自由，表示方法简洁。

2) 可移植性好。相对于硬件依赖性很强的汇编语言而言，C 语言通过编译来得到可执行代码。统计资料表明：C 语言编译程序 80%的代码是公共的，稍加修改即可用于其他的计算机。

3) 表达能力强，表达方式灵活，可进行结构化程序设计。另外一个重要的特点是，它能直接操作计算机硬件。当今应用广泛的嵌入式技术和单片机技术，控制硬件的代码，很多都是由 C 语言开发的。

4) 生成的目标代码质量高。C 语言仅比汇编语言生成的代码效率低 10%~20%，是其他高级语言无法匹敌的。

实例 2 C 语言环境使用介绍——用“*”输出字母“A”形状

实例任务

用“*”组成笔画，输出字母“A”的形状。程序运行结果如图 1-4 所示。



图 1-4 程序运行结果

程序代码

```
#include "stdio.h"
main()
{   printf("    *\n");
    printf("   **\n");
    printf("  ***\n");
    printf(" ****\n");
    printf(" *   *\n");
    printf("**   *\n");
    getch(); }
```

相关知识

1. Dev-C++简介

Dev-C++是一个 C/C++的开发工具，它是一款自由的软件，遵守 GPL 协议。它集合了 GCC、MinGW32 等众多自由软件，是不断升级的免费软件。它的开发环境包括多页面窗口、工程编辑器及调试器等。在工程编辑器中集合了编辑器、编译器、连接程序和执行程序，为减少编辑错误，提供了高亮度语法显示。它有完善的调试功能，能满足不同层次的用户需求，但它难以胜任规模较大的软件项目。基于完善的调试功能，本教材选用 Dev-C++为开发工具。

2. 显示运行结果

在使用 Turbo C 时，可通过一步操作查看程序的运行结果，但在 Dev-C++编写程序后，找不到运行结果的窗口。如果要查看结果，就需要在程序中进行代码操作，分以下 3 种情况：

在主函数最后添加语句“getch();”，它只适用于 C 程序。

在主函数最后添加语句“getchar();”，它适用于 C/C++程序。

在主函数最后添加语句“system("pause");”，它适用于 C/C++程序。

3. 创建并运行 C 程序

打开 Dev-C++，选择菜单“文件”→“新建”→“源代码”命令或按〈Ctrl+N〉组合键，此时光标位于窗口代码编辑区，如图 1-5 所示。

从图 1-5 所示窗口的光标处开始添加代码，如图 1-6 所示。

选择菜单“运行”→“编译”命令或按〈Ctrl+F9〉组合键进行编译，然后再选择菜单“运行”→“运行”命令或按〈Ctrl+F10〉组合键查看结果。也可一步完成，使用“运行”→

“编译运行”命令或按〈F9〉键来完成编译运行。编译时如果文件尚未保存，系统就会弹出“保存文件”对话框，此时输入文件名后，在“保存类型”下拉列表框中请选择“C source files(*.c)”，如图 1-7 所示。

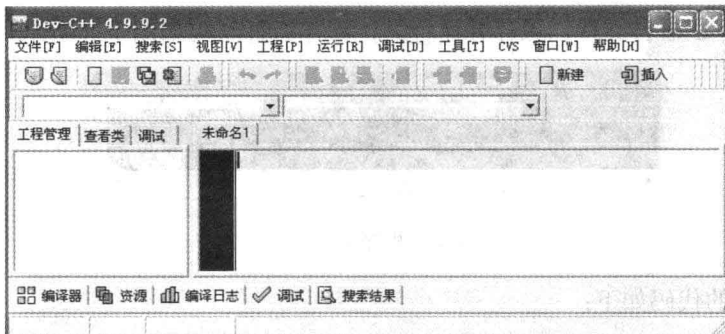


图 1-5 新建程序窗口

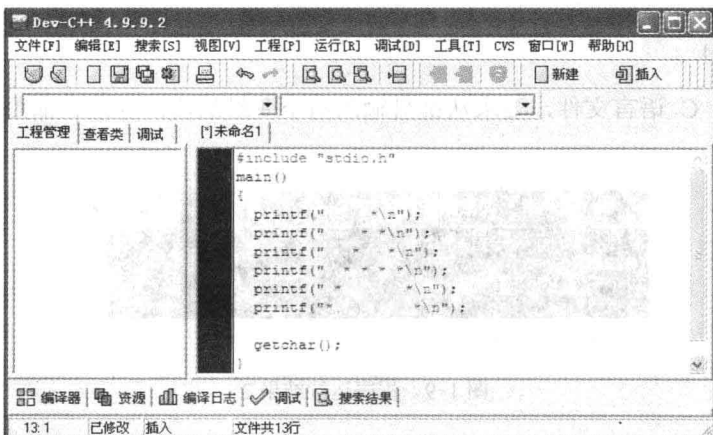


图 1-6 添加代码的窗口



图 1-7 “保存文件”对话框

如果选择“运行”命令或按〈F9〉键后，就可以看到如图 1-4 所示的运行结果。如果程序运行过程中陷入死循环，可使用〈Ctrl+C〉组合键结束程序的运行。

课堂精练

1. 新建一个 C 语言文件，保存并运行之，程序运行结果如图 1-8 所示。



图 1-8 程序运行结果 1

此程序添加的代码如下：

```
#include "stdio.h"
main()
{ printf("学习无难事，只怕有心人。");
  getch();}
```

2. 新建一个 C 语言文件，要求从键盘输入两个变量的值，并求和输出。程序的运行结果如图 1-9 所示。

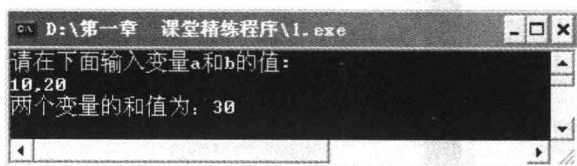


图 1-9 程序运行结果 2

程序代码如下：

```
#include "stdio.h"
main()
{ int a,b,sum; /*定义 3 个变量*/
  printf("请在下面输入变量 a 和 b 的值:\n");
  /*输出一行提示信息，\n 表示将光标输出到下一行*/
  scanf("%d,%d",&a,&b); /*从键盘输入两个变量的值*/
  getch();
  sum=a+b; /*将 a 与 b 的和值存放到变量 sum 中*/
  printf("两个变量的和值为: %d",sum); /*输出 sum 的和值*/
  getch(); }
```

1.2 程序和程序逻辑

学习目标

- 掌握算法与程序的概念

● 了解算法的基本描述方法

实例3 程序与算法的概念——梵塔推理

实例任务

将1号柱大小不等的3个物体，移动到3号柱上，顺序必须与1号柱顺序一致。要求每次只移动一个物体，而且每根柱上小块物体必须置于大块物体之上。其移动过程如图1-10所示。

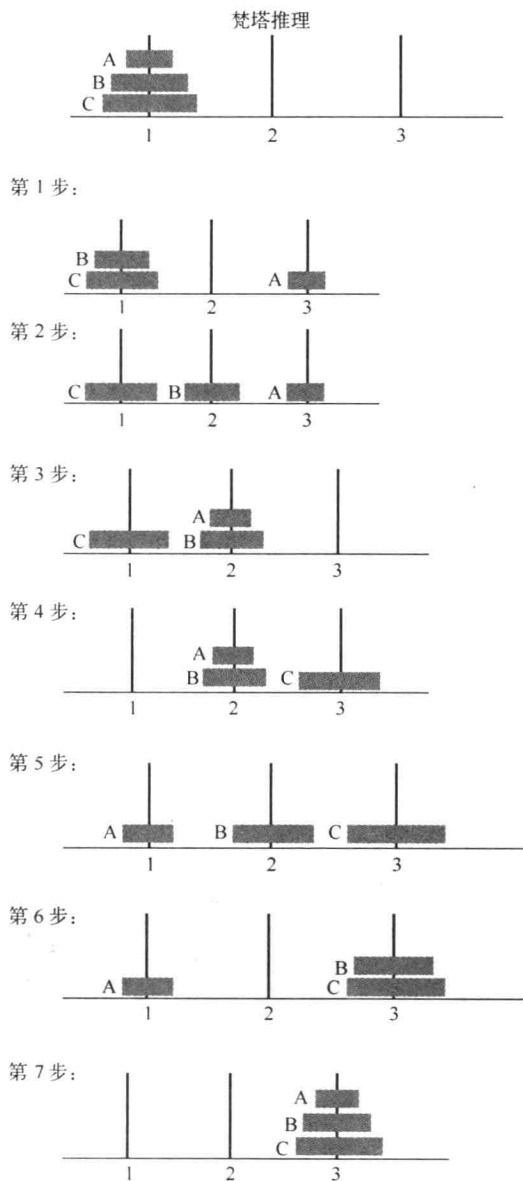


图1-10 梵塔推理移动过程图