

高等教育“十三五”规划教材

# C语言程序设计 案例教程（第2版）

主 编 刘会超 杨锋英

高等教育“十三五”规划教材

# C 语言程序设计案例教程 (第 2 版)

主 编 刘会超 杨锋英  
副主编 魏雪峰 崔英杰  
汪 洋 吴海涛

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书主要内容包括认识 C 语言、简单的 C 语言程序、分支结构程序设计、循环结构程序设计、函数、数据类型与数据的输入输出、数组、指针、结构体、文件和综合案例等。每章由学习目标、主要内容、教学案例、相关知识、本章小结、习题和实训项目构成。

本书以能力培养为目标，用案例引入知识，用任务驱动教学。按照读者的认知规律和特点选择案例，把知识融入案例中。本书围绕案例中的任务展开知识点教学，在实际任务的驱动下引导读者学习 C 语言基础知识与编程技能，把 C 语言教学从传统的“讲授+上机”模式向“做中学、学中做”模式转变。

本书可作为高等院校计算机及相关专业的高级语言教材，也可供 C 语言爱好者学习使用。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

### 图书在版编目 (CIP) 数据

C 语言程序设计案例教程 / 刘会超, 杨锋英主编. —2 版. —北京: 电子工业出版社, 2019.7  
ISBN 978-7-121-36762-5

I. ①C… II. ①刘… ②杨… 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 106588 号

责任编辑: 祁玉芹

印 刷: 三河市良远印务有限公司

装 订: 三河市良远印务有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 19.25 字数: 468 千字

版 次: 2015 年 1 月第 1 版

2019 年 7 月第 2 版

印 次: 2019 年 7 月第 1 次印刷

定 价: 52.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888, 88258888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式: [qiyuqin@phei.com.cn](mailto:qiyuqin@phei.com.cn)。

# 前言

P R E F A C E

C 语言是国内外应用广泛、最具影响力的计算机语言之一，是大学理工科专业学生的必修课。为使初学者对 C 语言有一个很好的入门，作者融合多年的教学经验和教学资源编写了本书。这是一本面向广大初学者的 C 语言教程，最大的特色是以任务导学、案例丰富、深入浅出、立体配套。针对初学者的特点，力求做到将复杂的概念用简洁浅显的语言娓娓道来。

本书的创新在于以能力培养为目标，用案例引入知识，用任务驱动教学。按照读者的认知规律和特点选择案例，把知识融入案例。围绕案例中的任务展开知识点教学，在实际任务的驱动下，引导读者学习 C 语言基础知识与编程技能，引导 C 语言教学从传统的“讲授+上机”模式向“做中学、学中做”模式转变。

书中每个案例包括任务描述、任务分析、解决方案和源程序 4 部分，并且提供与教学案例相关知识的习题和实训项目作为读者练习巩固之用。为了保证知识的系统性与完整性，拓宽知识面，在相关案例后增加了相关知识与知识拓展；另外，本书还配有电子教学参考资料包（包括书中所有案例的源代码、电子教案、习题参考答案）。

本书内容共分为 11 章，第 1 章认识 C 语言，第 2 章简单的 C 语言程序，第 3 章分支结构程序设计，第 4 章循环结构程序设计，第 5 章函数，第 6 章数据类型与数据的输入/输出，第 7 章数组，第 8 章指针，第 9 章结构体，第 10 章文件，第 11 章综合案例——学生成绩管理系统，以及附录和参考书目。在内容上，新版教程删除了第 1 版的“常用算法”章节，增加了一个“综合案例——学生成绩管理系统”章节。变更主要是考虑到算法在一些专业的实际教学中用不到，且内容比较浅显，而综合案例可以为相关专业的课程设计教学或综合实训提供支持。

本书由刘会超和杨锋英担任主编，崔英杰、魏雪峰、汪洋、吴海涛担任副主编。第 2、3、4 章由崔英杰编写，第 1、10 章由魏雪峰编写，第 9 章由汪洋编写，第 6 章由吴海涛编写，第 8、11 章由刘会超编写，第 5、7 章及附录由杨锋英编写。全书由耿红琴统稿审核。

由于编写时间仓促，书中难免有疏漏和不妥之处，衷心希望广大读者，尤其是任课教师提出宝贵的意见和建议，以便再版时修正。

编者

2019 年 5 月

前言

# 目录

C O N T E N T S

第 1 章 认识 C 语言 .....	1
案例 1 用计算机求解圆的面积 .....	1
相关知识——计算机求解问题的步骤 .....	2
案例 2 使用 Dev-C++ 环境 .....	5
相关知识——C 语言的程序结构 .....	9
知识拓展——算法 .....	10
本章小结 .....	15
习题 .....	15
实训项目 .....	16
第 2 章 简单的 C 语言程序 .....	18
案例 1 超市收费程序的设计 .....	18
相关知识——算术表达式与赋值表达式 .....	19
案例 2 超市促销活动收费程序的设计 .....	26
相关知识——二分支 if 语句 .....	27
案例 3 超市收银程序的设计 .....	29
相关知识——while 语句 .....	30
本章小结 .....	31
习题 .....	31
实训项目 .....	32
第 3 章 分支结构程序设计 .....	33
案例 1 计算阶梯电费 .....	33
相关知识——关系表达式与逻辑表达式 .....	35
相关知识——if 语句 .....	41
案例 2 简单算术计算器的设计 .....	44
相关知识——switch 语句 .....	46
案例 3 自动售货机商品价格的查询 .....	48

本章小结 .....	50
习题 .....	50
实训项目 .....	57
<b>第 4 章 循环结构程序设计 .....</b>	<b>59</b>
案例 1 日积跬步 .....	59
相关知识——for 语句 .....	60
案例 2 寻找行李箱密码 .....	63
相关知识——do-while 语句 .....	64
案例 3 幸运编号 .....	65
相关知识——break 语句与 continue 语句 .....	66
案例 4 打印九九乘法表 .....	68
相关知识——多重循环 .....	70
案例 5 猜数游戏 .....	71
相关知识——随机函数 .....	73
案例 6 百钱买百鸡 .....	74
相关知识——穷举法 .....	76
案例 7 八戒吃西瓜 .....	76
本章小结 .....	78
习题 .....	78
实训项目 .....	83
<b>第 5 章 函数 .....</b>	<b>85</b>
案例 1 居民日常计费系统 .....	85
相关知识——函数基础 .....	88
案例 2 最小公倍数 .....	91
相关知识——函数的嵌套调用 .....	92
案例 3 “魔幻”长方体 .....	93
相关知识——变量进阶 .....	94
案例 4 猜年龄 .....	99
相关知识——递归思想 .....	100
案例 5 汉诺塔游戏 .....	102
案例 6 幕后英雄 .....	105
相关知识——编译预处理 .....	106
本章小结 .....	111
习题 .....	111
实训项目 .....	115

<b>第 6 章 数据类型与数据的输入/输出</b> .....	<b>118</b>
6.1 数据类型 .....	118
6.2 数据的输入/输出 .....	125
6.3 运算符和表达式 .....	134
6.4 类型转换 .....	138
本章小结 .....	139
习题 .....	140
<b>第 7 章 数组</b> .....	<b>142</b>
案例 1 天外有天 .....	142
相关知识——一维数组的增删改查操作 .....	143
案例 2 网店热销手机排行榜 .....	147
相关知识——冒泡排序与选择排序 .....	149
案例 3 揪心的房价 .....	151
相关知识——折半查找 .....	152
案例 4 生存游戏 .....	153
相关知识——筛法 .....	155
案例 5 矩阵转置 .....	156
相关知识——二维数组 .....	157
案例 6 杨辉三角形 .....	159
案例 7 信息加密处理 .....	160
相关知识——字符数组与字符串处理操作 .....	161
案例 8 统计单词个数 .....	167
案例 9 进制转换 .....	168
本章小结 .....	169
习题 .....	169
实训项目 .....	173
<b>第 8 章 指针</b> .....	<b>175</b>
案例 1 寻找存折密码 .....	175
相关知识——指针与指针变量 .....	176
案例 2 拨云见日 .....	179
相关知识——指针的交换 .....	180
案例 3 猜宝游戏 .....	181
相关知识——指针作为函数参数 .....	183
案例 4 产品使用寿命统计分析 .....	184
相关知识——一维数组与指针 .....	186

案例 5 实现简单电子表格 .....	191
相关知识——二维数组与指针 .....	193
案例 6 信息解密 .....	194
相关知识——字符指针 .....	195
案例 7 常任理事国国名的字典次序 .....	197
相关知识——指针数组与多级指针 .....	198
案例 8 我的程序我作主 .....	200
相关知识——带参的 main .....	203
案例 9 由我差遣 .....	204
相关知识——函数指针与指针函数 .....	206
案例 10 我心飞翔 .....	207
相关知识——动态内存空间管理函数 .....	209
本章小结 .....	210
习题 .....	210
实训项目 .....	218
<b>第 9 章 结构体 .....</b>	<b>219</b>
案例 1 学生信息表 .....	219
相关知识——结构体的定义与结构体变量 .....	220
案例 2 民主选举得票统计 .....	226
相关知识——结构体数组 .....	227
案例 3 输出班长的基本信息 .....	229
相关知识——结构体指针 .....	231
案例 4 摸球游戏 .....	233
相关知识——枚举类型 .....	235
案例 5 设计教师与学生通用的表格 .....	236
相关知识——共用体类型 .....	238
本章小结 .....	239
习题 .....	240
实训项目 .....	244
<b>第 10 章 文件 .....</b>	<b>245</b>
案例 1 导出学生信息到文件 .....	245
相关知识——文件的基本操作 .....	248
案例 2 学生信息的导入 .....	252
案例 3 学生信息的备份 .....	255
相关知识——文件定位函数 .....	256
本章小结 .....	257

习题	257
实训项目	261
<b>第 11 章 综合案例——学生成绩管理系统</b>	<b>262</b>
综合案例 学生成绩管理系统	262
相关知识——创建链表	271
相关知识——遍历链表	274
相关知识——在链表中插入节点	284
相关知识——删除链表中的节点	289
本章小结	296
习题	296
<b>附录 A ASCII 码表</b>	<b>298</b>
<b>附录 B C 语言运算符的优先级与结合性</b>	<b>299</b>
<b>参考书目</b>	<b>300</b>

# 第1章

# 认识C语言

## 学习目标

通过本章内容的学习,使读者熟悉 Dev-C++环境,掌握在其中创建、编辑、编译、运行 C 语言项目的过程,并且掌握 C 语言程序的基本格式和用计算机解决问题的思路、方法和步骤。

## 主要内容

- ◆ 程序设计的基本概念。
- ◆ 用计算机解决问题的思路、方法和步骤。
- ◆ C 语言程序的基本格式。
- ◆ Dev-C++环境的使用。

## 案例 1 用计算机求解圆的面积

### 【任务描述】

输入圆的半径,计算其面积。

### 【任务分析】

该任务中的量有 3 个,即圆的半径、圆周率及圆的面积,其中圆周率是已知量;圆的半径和圆的面积是未知量;圆的半径是在计算机运行程序时输入的,而圆的面积是利用公式圆周率 $\times$ 半径的平方求得的。

### 【解决方案】

- (1) 定义符号 PI 表示圆周率 3.14159。
- (2) 定义实型变量  $r$  和  $area$ ,  $r$  表示半径,  $area$  表示圆的面积。

- (3) 调用函数 `scanf` 为半径  $r$  输入数据。
- (4) 将  $\text{PI}$  和  $r$  代入面积公式求圆的面积 `area`。
- (5) 输出圆的面积 `area`。

### 【源程序】

```
/*程序名称: 1_1.c */
#include <stdio.h> /*预编译*/
#define PI 3.14159 /*定义符号PI, 代表圆周率3.14159*/
int main() {
    double r, area; /*定义变量r, area*/
    printf("请输入圆的半径: "); /*提示输入半径的值*/
    scanf("%lf", &r); /*输入半径的值*/
    area = PI * r * r; /*计算圆的面积*/
    printf("半径是%.2lf的圆的面积是 %lf\n", r, area ); /*输出圆的面积*/
    return 0;
}
```

### 【说明】

从这个案例中可以了解一个 C 语言程序的基本结构和书写格式。

C 语言程序由函数组成, 每一个 C 语言项目的文件中有且只能有一个主函数 (`main` 函数), 它指示计算机执行程序时的入口。

`double r, area;`是变量定义语句, 用来定义实数类型变量  $r$  和 `area`, 分别保存圆的半径和面积。

`printf()`和 `scanf()`是两个函数调用, 其中 `printf` 函数的功能是把输出内容输出到显示器; `scanf` 函数的功能是从键盘输入指定类型的数据。二者都是由系统定义的标准函数, 可在程序中直接调用, 但使用时需在程序开头添加预编译命令 “`#include <stdio.h>`”。

将这段程序代码在 Dev-C++ 开发环境中编辑后, 通过 C 语言的编译系统编译并运行。程序首先提示用户输入圆的半径, 然后计算圆的面积, 最后在屏幕上显示所求圆的面积。

## 相关知识——计算机求解问题的步骤

### (一) 程序和程序设计语言

#### 1. 程序

计算机诞生之前就有了“程序”的概念, 《现代汉语词典》解释“程序”就是事情进行的先后次序, 如日常说的“工作程序”“会议程序”等, 本书所讲的程序是计算机程序。

计算机程序是让计算机完成某项特定任务而编写的一组指示计算机工作的指令。计算机就像一个士兵, 无条件地服从长官(程序员)的命令。为了完成一项任务, 长官下达的一系列命令就是“程序”。

简单的程序只有几条指令, 而复杂的程序多达数千万条指令。任务的规模越大, 内容越复杂, 所需要的程序指令就越多, 程序的结构也就越复杂。仅一个 Windows 操作系统就

有几千万条的指令，所以给计算机下命令需要团队的集体智慧。

为了有效地指挥计算机工作，需要开展程序设计。这是设计、编写、调试程序的方法和过程，是目标明确的智力活动。它要求程序员首先对需要完成的任务有一个比较清晰的认识，然后按照计算机可以识别的方式来组织相应的指令以形成程序。最后将程序提交给计算机执行，从而完成预定任务。由于任务的复杂性和多样性，因此程序设计工作也不可能一蹴而就。需要在设计过程中不断地修改和完善，最终满足任务的需求，这就是程序的调试和测试过程。

## 2. 程序设计语言

程序设计语言为程序员编写一个好的程序提供了所需要的抽象机制、组织原则及控制结构，其一基础是一组符号和一组规则，符号按照规则构成的符号串的总体就是程序设计语言。程序设计语言包含3个要素，即语法、语义和语用。语法表示程序的结构或形式，包括构成程序设计语言的各个符号之间的组合规律，但不涉及这些符号的特定含义及使用者；语义表示程序的含义，包括程序设计语言中各个符号及按语法形成的符号串的特定含义，也不涉及使用者；语用就是程序设计语言的实际应用，是人和计算机之间沟通的工具。

机器语言是计算机所能直接识别的语言，是由0和1（即二进制）组成的指令序列。由于人们书写和理解二进制数据存在一定的困难，于是诞生了汇编语言。

汇编语言用英文字母或符号串来替代机器语言，把不易理解和记忆的机器语言按照对应关系转换成汇编指令，使得汇编语言比机器语言易于阅读和理解。但是汇编语言依赖于硬件，程序的可移植性差。而且程序员在使用新的计算机时必须学习该机器对应的汇编指令，大大增加了工作量，为此产生了高级语言。

高级语言比汇编语言更接近人类的自然语言，因此易于理解、记忆和使用。常见的高级语言有C、C++、Java、Python等。高级语言不能被计算机直接识别，需要将其“翻译”成机器语言，这个过程称为“编译”（也称为“解释”）。编译过程由相应计算机语言的编译程序自动完成，不需要手工处理。

## （二）计算机求解问题的步骤

计算机求解问题是人们解决某一问题的方法和步骤的计算机化，或者说通过计算机来表达人们对某一问题的解决方法。一般来说，用计算机解决一个具体问题时，首先从具体问题抽象出一个适当的数学模型；其次设计一个解此数学模型的算法；最后编制程序进行调试与测试，直至得到最终解。具体来说使用计算机求解问题的步骤包括问题分析与算法设计、编写、编译，以及运行与调试程序，下面以案例1为例说明。

### 1. 问题分析与算法设计

为了计算圆的面积，根据问题描述需要输入圆的半径。然后根据圆的面积公式（ $\pi r^2$ ）求出圆的面积，算法设计如下：

- （1）为计算机提供圆的半径值（输入）。
- （2）根据计算公式求圆的面积（处理）。
- （3）将计算的结果显示在屏幕上（输出）。

## 2. 编辑程序

为了让计算机能代替人工完成以上流程,需要将以上流程转化成计算机可以识别的指令序列(即程序),下面的代码是运用C语言对这个任务的计算机描述:

```
#include <stdio.h>
#define PI 3.14159
int main(){
    double r, area;
    printf( "请输入圆的半径: " );
    scanf("%lf", &r);
    area = PI*r*r;
    printf("半径是%.2lf 的圆的面积是 %lf\n", r, area);
    return 0;
}
```

## 3. 编译

编好程序(C语言源程序文件,扩展名为.c)后,下一步运用C语言的编译器对其进行编译形成可执行的程序(扩展名为.exe)。

## 4. 运行与调试

程序通过语法检查,编译生成可执行文件后在Dev-C++集成开发环境或操作系统环境中运行(Run)。

如果程序有语义错误,则需要对程序进行调试,即查找并修改错误。

在Dev-C++环境中调试过程常用的方法是设置断点并观察变量。

(1) 设置断点:在Dev-C++环境的当前语句行按F4键,设置断点标记(默认语句行的底色变为红色,并且行首有断点标记红点,其中有对号),程序调试到此行停下。如果当前语句行已经设置断点标记,则在该语句行按F4键取消断点标记。

(2) 观察变量:对于设置断点的程序,按F5键开始单步调试,即将执行的调试行底色变为蓝色且行首变为蓝色箭头。

Dev-C++环境的调试器如图1-1所示。



图 1-1 Dev-C++环境的调试器

当程序运行到设有断点行时暂停执行,程序员此时可以通过单击【添加查看】按钮设置需要观察的变量,在调试输出窗口中给出该程序行之前此变量的当前值。

Dev-C++源程序及调试输出窗口如图1-2所示。

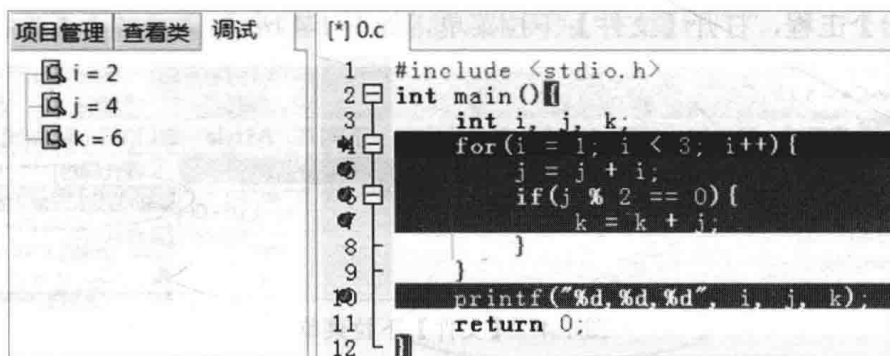


图 1-2 Dev-C++源程序及调试输出窗口

判断变量的当前值是不是所预期的,如果不是,说明该断点之前有错误发生;如果是,说明该断点之前满足要求,单击【下一步】按钮或者按 F7 键继续。通过设置断点与观察变量可以有效地排查程序的出错范围。

结束调试按 F9 键或者单击【停止执行】按钮。

### 【思考题】

- (1) 思考 C 语言程序的基本结构。
- (2) 模仿案例 1, 输入长方形的长和宽, 计算并输出长方形的面积。

## 案例 2 使用 Dev-C++环境


### 【任务描述】

利用 Dev-C++环境完成基本对话程序的运行。

### 【任务分析】

完成本任务首先需要成功安装并启动 Dev-C++, 然后创建项目和程序文件, 使用该环境的编辑和编译功能将源程序转换为可执行程序后运行该程序得到结果。

### 【解决方案】

在安装并配置 Dev-C++之后(本书采用的是简体中文版), 双击桌面上的  快捷方式, 或者单击【开始】→【所有程序】→【Bloodshed Dev-C++】下的 Dev-C++程序, 都可以打开 Dev-C++的运行界面, 如图 1-3 所示。

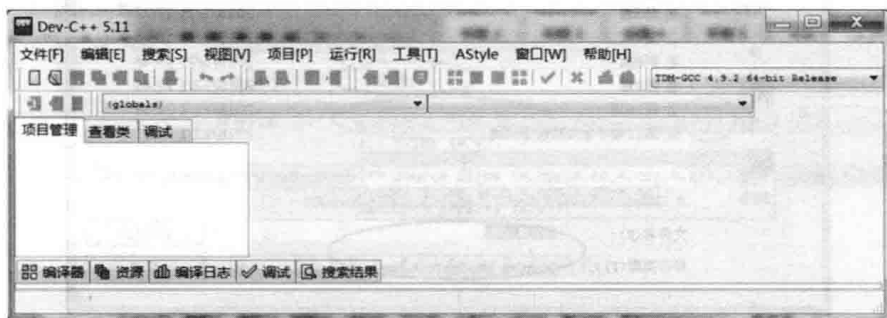


图 1-3 Dev-C++的运行界面

为创建一个工程，打开【文件】下拉菜单，如图 1-4 所示。

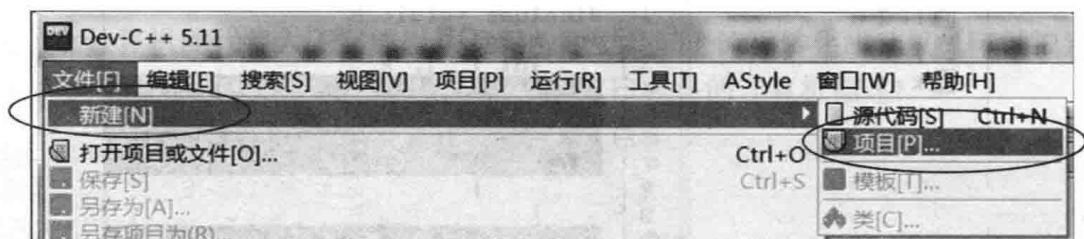


图 1-4 【文件】下拉菜单

选择【新建】→【项目】命令，弹出【新项目】对话框。打开【Basic】选项卡，选择“Empty Project”选项和“C 项目”单选按钮，如图 1-5 所示。



图 1-5 选择“Empty Project”选项和“C 项目”单选按钮

在【名称】文本框中输入项目名“第 1 章”，单击【确定】按钮，弹出【另存为】对话框，如图 1-6 所示。



图 1-6 【另存为】对话框

在【保存在】下拉列表框中选择保存路径，本例为“E:\源程序”。单击【保存】按钮，打开创建成功的界面，如图 1-7 所示。

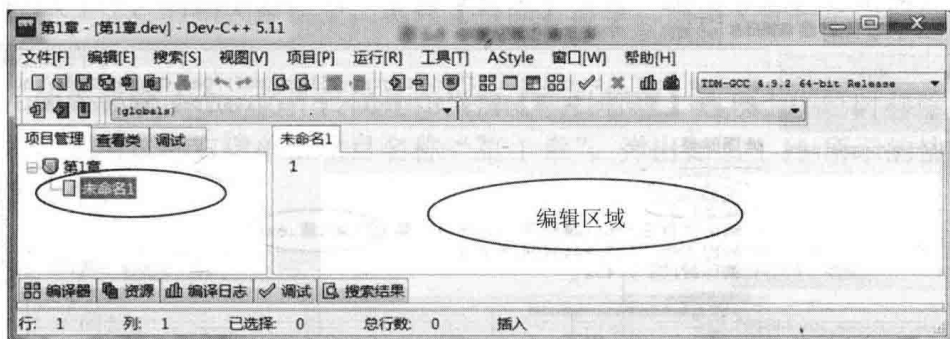



图 1-7 创建成功的界面

在编辑区域中输入程序 1\_2.c 的代码：

```
/*程序名称: 1_2.c */
#include <stdio.h>
int main(){
    printf("hello,my darling!");
    return 0;
}
```

按 Ctrl+S 组合键或者单击主工具栏中的  保存按钮或者选择【文件】→【保存】命令，弹出【保存为】对话框，如图 1-8 所示。

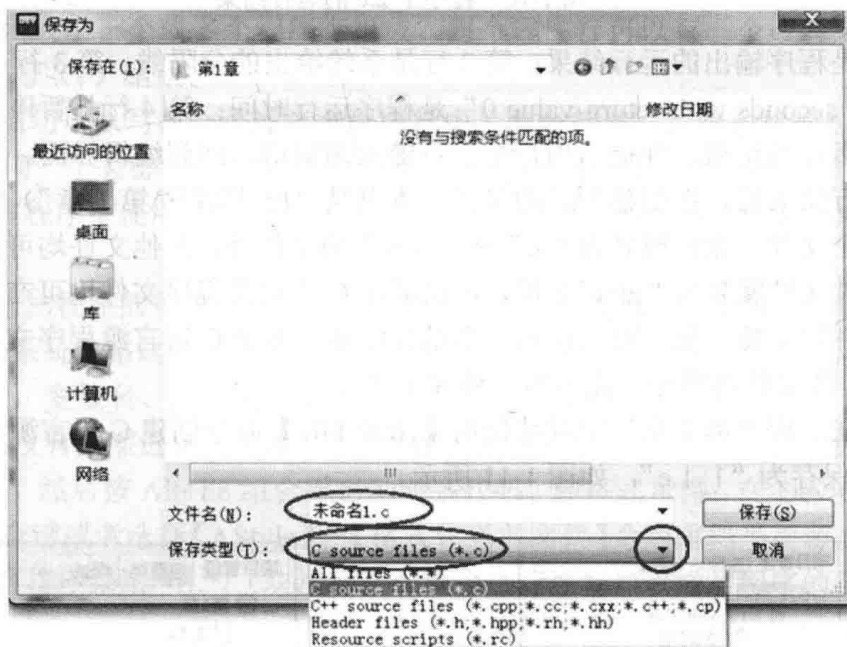



图 1-8 【保存为】对话框

在【文件名】文本框中输入文件名“1\_2”，保存类型选择为“C source files(\*.c)”。单击【保存】按钮，将该程序保存在“E:\源程序\第1章”中。按 F9 键或者单击编译运行工

具栏中的  按钮或者选择【运行】→【编译】命令，C语言编译器编译该程序，编译结果如图 1-9 所示。

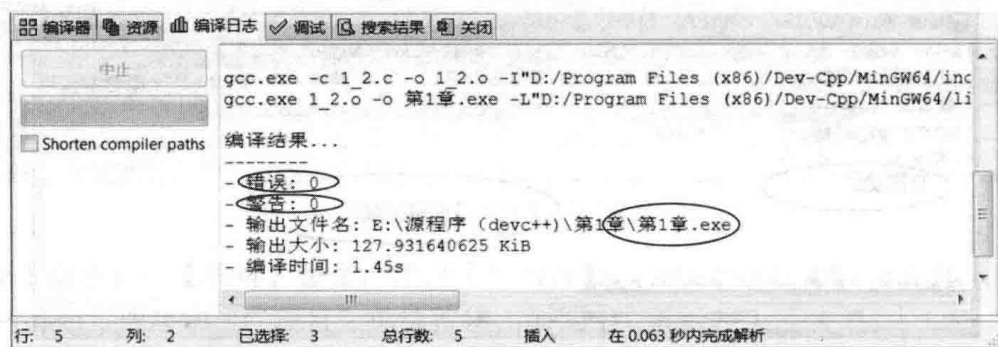



图 1-9 编译结果

从程序 1\_2.c 的编译信息看出在编译时没有错误和警告，得到可执行文件“第 1 章.exe”。按 F10 键或者单击编译运行工具栏中的  按钮或者选择【运行】→【运行】命令，程序的运行结果如图 1-10 所示。

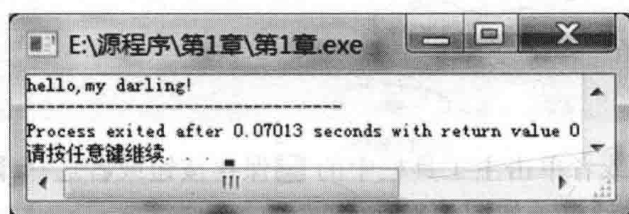


图 1-10 程序 1\_2.c 的运行结果

第 1 行是程序输出的运行结果；第 2 行是系统给出的分隔线；第 3 行“Process exited after 0.07013 seconds with return value 0”是程序运行时间；第 4 行是暂停信息，有助于用户查看程序运行的结果，并提示用户按任意键关闭窗口，回到编辑界面。

程序运行结束后，在创建项目的目录（本书是“E:\源程序\第 1 章”）中可以看到其中已经生成多个文件。除扩展名为“.c”和“.dev”的文件外，其他文件均可删除。因为每次运行项目文件（扩展名为“.dev”）后，再次编译 C 语言源程序文件均可产生这些文件。为解决其他文件过多的问题，可以在同一个项目中建立多个 C 语言源程序文件。这样只有扩展名为“.o”的文件会增加，而其他文件并不增加。

例如，在工程“第 1 章”中继续使用【New File】命令创建 C 语言源程序文件“未命名 2”，将其保存为“1\_1.c”，如图 1-11 所示。



图 1-11 保存为“1\_1.c”

在 C 语言源程序文件 1\_1.c 的编辑界面中输入案例 1 的代码，在这两个 C 语言源程序