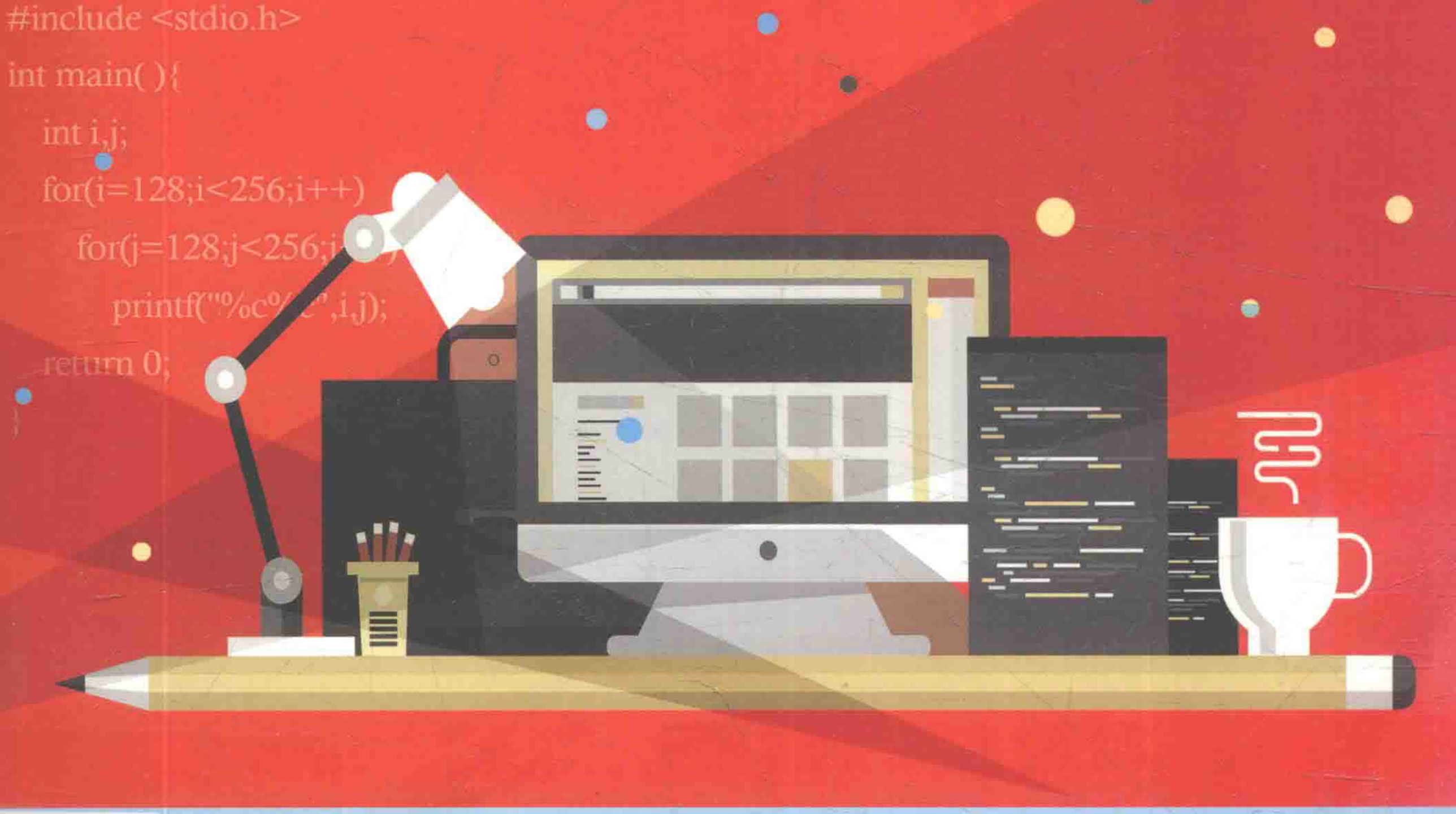
C程序设计

——基于应用导向与任务驱动的学习方法

- 思路分析精准到位，授人以渔
- 原理概念阐述透彻，图表丰富
- 内容讲解深入浅出，激发兴趣

◎ 贺细平 著



中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

C程序设计

——基于应用导向与任务驱动的学习方法

贺细平　著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书采用以解决实际应用问题为导向、以具体编程任务为驱动的模式，将 C 语言的知识无缝融入每个实际应用程序中。作者精心设计了 100 多个应用案例，每个案例均有实现特定功能的、完整的、可运行的程序代码。本书图表丰富，对程序设计的概念、求解问题的思路和方法、程序背后的原理和机制进行了深入剖析。全书共 9 章。第 1 章从简单程序入手，将程序设计相关的基础性概念融入案例，使读者对 C 程序设计有一个整体的、直观的、感性的认识。第 2 章阐述表达复杂逻辑的分支和循环语句的用法，使读者对应用问题中的逻辑具有较好的表达能力。第 3 章阐述了利用数组处理批量数据。第 4 章阐述了如何存储和处理文本型数据。第 5 章全面地阐述了函数这一模块化程序设计利器。第 6 章对具有内存间接访问能力的指针进行了深入阐述。第 7 章讲解如何利用结构体类型创建用户所需新数据类型。第 8 章阐述了如何利用文件实现数据持久化。第 9 章讲解了位运算的规则和用法。

图书在版编目（CIP）数据

C 程序设计：基于应用导向与任务驱动的学习方法/贺细平著. —北京：电子工业出版社，2018.1
ISBN 978-7-121-33232-6

Ⅰ. ①C… Ⅱ. ①贺… Ⅲ. ①C 语言—程序设计 Ⅳ.①TP312.8

中国版本图书馆 CIP 数据核字（2017）第 306156 号

策划编辑：冉 哲
责任编辑：底 波
印 刷：三河市鑫金马印装有限公司
装 订：三河市鑫金马印装有限公司
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：787×1 092 1/16 印张：32.75 字数：880 千字
版 次：2018 年 1 月第 1 版
印 次：2018 年 1 月第 1 次印刷
定 价：89.00 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 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：ran@phei.com.cn。

前　言

当您第一眼见到这本书，一定诧异于它的厚度，但只要您翻开阅读，我想，您一定不会觉得这是一本难啃的“大部头”，而像是一本娓娓道来的程序设计“故事书”。

当我想要编写一本关于程序设计的教材时，难抑内心的激动。我的第一门程序设计语言是BASIC，然后学习 C 语言，后来学过 C++、Java、Python 等程序设计语言。多数学过程序设计的人，对第一门程序设计语言的印象大抵是艰难而晦涩的。一些人秉承对程序设计的执着和热爱，从这种艰难中走过来了，并且从此爱上了程序设计，享受程序设计在解决现实生活的实际应用问题后带来的快乐和成就感。但更多的人对编程望而生畏，面对堪如天书的代码，始终难解心中诸多困惑：这些代码是如何写出来的？为什么要写成这样？必须写成这样吗？为什么我这样写就不对呢？解决此问题还有其他写法吗？为何当我面对实际应用问题时总感到无从下手呢？怎样才能创造出属于自己的程序呢？

我尽最大努力，使程序设计的每个细节变得简单清晰。为了让您理解程序的来龙去脉，对于每次需要解决的编程任务，不是一次性地抛出最终程序代码，而是必须对解决此问题的思路、方法进行详尽分析。并且遵循“由简单到复杂，由低级到高级”的设计过程，尽可能完整地展示“程序是怎样炼成的”。对于同一编程任务，提供尽可能多的设计思路和不同的算法以及实现代码，帮助您打开程序设计思维的匣子。

本书侧重于培养您作为程序设计者必须具备的计算思维。所谓的计算思维，就是以计算机的方式去思考问题的求解过程。作为机器的“计算机”思考问题的方式与作为万物之灵的“人”的思考问题方式是不同的。“人”通过学习程序设计去理解并掌握“计算机”思考问题的方式，这个过程就是培养计算思维的过程。因此，本书以求解具体应用问题为目标，驱动相关程序设计知识的应用。

本书强调，程序设计语言是求解问题的工具，程序设计语言是为求解问题服务的。本书对语法的讲解以够用为准，不提倡代码中应用古怪、费解的语法。当然，程序设计必须掌握相关语法，有语法错误的程序过不了“编译关”。掌握 C 语言语法不是程序设计学习的重点，更不是学习目的。培养计算思维，能利用程序设计语言解决实际应用问题才是最终目的，学习程序设计必须过实际“运行关”。

C 语言具有语法简洁、概念清晰、底层控制力强等优点，是值得程序设计初学者首选的语言。C 语言虽是面向过程的程序设计语言，但是学好 C 语言将为面向对象的程序设计语言（如 C++、Java、C#、Python 等）的学习打下坚实基础。

学好编程没有捷径，上机练习、独立思考、保持兴趣、学用结合、日积月累、持之以恒是成为编程高手的秘籍。关于如何学习程序设计的建议请参见附录“10.1 关于程序设计的学习方法”（扫描前言中的二维码）。

本书特色：

一、本书贯彻以求解应用问题（实际应用问题的一部分或实际应用问题的简化问题）为导向，以具体“编程任务”为驱动的程序设计学习方法，将每个知识点融入实际编程任务中。因此，展现在您面前的代码是一个完整的、可运行的、有输入/输出的、实现了一定功能的应用

程序，而不是仅仅为了讲解某个知识点的片段的、不完整的代码。

二、作为例题的编程任务生动有趣。每个编程任务力求有现实生活应用背景，让您时刻不忘学习程序设计的目的是能运用计算机程序解决现实生活中或大或小的实际问题，体会计算机是如何按您的意图行动的，体会计算机给我们生活带来的方便，体会计算机的威力与魅力。编程不再是单纯地学习 C 语言语法，也不是纯粹为了实现数学的数值计算。

三、图表丰富。本书秉承“能用图和表表达的，一定画图做表”的思想。因此，书中配有大量图解、表格，大量地运用了类比、对比、小贴士、小问答等形式，尽量用直观的形式帮助您理解程序设计的概念、原理、机制等方面。

四、讲解深入浅出。本书融入了我多年程序设计教学经验、教学成果、应用软件开发经验和对程序设计的体会与理解。对程序中诸多概念的理解，需要程序设计者（以后简称为程序员）对操作系统的有关原理有一定的理解。因此，本书在讲解程序的同时，尽量对发生在程序运行背后的机制——特别是操作系统中与编程相关的机制进行了深入剖析。对操作系统和计算机原理的介绍，能帮助程序员深入地理解程序在底层的运行机制，使程序员在编程时做到“知其然”且“知其所以然”。

五、程序代码“箭指”代码解释，阅读代码一目了然、易读易懂。对于程序代码中每个重要语句，均引出箭头指向相应的代码解释，代码和对应的解释是左邻右舍、如影相随的，方便阅读。此外，在排版上，也尽量将一个完整程序或函数模块的代码排在同一页中，确保代码的形式整体性。

六、本书站在程序员的角度来看待和学习 C 语言，而不是站在 C 语言的角度罗列 C 语言知识本身。站在程序员的角度，面对编程任务时，我们应该思考的是如何运用 C 语言为“我”（即程序员）的设计目标服务。从这个角度出发，您就更容易理解和接受 C 语言的知识了。

本书的例题全部采用编程任务的形式给出。每个编程任务由 8 部分构成：标题、任务描述、输入、输出、输入举例、输出举例、分析，以及参考代码。本书例题采用此形式是基于以下四点考虑的。

其一，这种方式对要解决的任务有清晰、准确的编程描述，因此每个程序代码需要达到的目标和需要实现的功能非常明确。学习程序设计是为了能用自己设计的程序解决实际问题，因此，我们将本书读者的角色定为软件开发者。软件开发是软件开发者按照用户提出的需求进行软件设计的过程。设计得到的软件必须达到指定功能，满足软件用户的需求。描述清晰、准确的软件需求对软件开发至关重要。因为需求的小变化，可能导致软件设计的巨大改变，甚至从头重新设计。

其二，有利于独立思考和寻求解决问题的多种方法，培养计算思维。在达到既定软件开发目标的情况下，鼓励读者学会分析问题，开动脑筋独立思考，尝试用不同思路、不同算法或不同的代码去完成同一个任务，对比不同实现方式之间的优缺点。对于每个编程任务，本书代码仅供参考。本书绝不鼓励读者仅满足于将本书代码照抄照搬，死记硬背。

其三，对 C 语言知识点均采用融入具体编程任务的方式讲解，使我们对每个 C 语言知识要素所适用的实际应用场合有最感性的认识。

其四，方便使用 OJ 作为程序设计在线练习平台。本书的编程任务便于自动裁判（可简单地将“裁判”理解为教师批改学生所交的程序设计作业这一过程）。国内外有许多大学和组织提供了开放式的在线裁判系统（Online Judge，OJ），它能对提交的程序源代码进行自

动裁判。OJ 系统 24 小时在线练习资源丰富，裁判结果公正客观。OJ 系统原本为程序设计竞赛所用，但是好的工具为什么不能用于学习呢？参加过信息学竞赛（IOI，NOI）或大学生程序设计竞赛（CPC）的读者一定对这种编程任务的形式不陌生，因为竞赛题采用此形式。希望本书读者不要对此表示疑惑，学习程序设计当然不是为了参加比赛。在此，只是取其长而用之，更好地服务于学习程序设计这一目标。我早在 2009 年就开始将 OJ 系统作为练习平台引入到信息类本科专业的“C 程序设计”课程教学中，得到了学生和同行的好评与认可。目前，将 OJ 作为程序设计教学练习平台的做法在越来越多的学校的程序设计教学中得到运用。

本书提供所有编程任务的描述、测试用例数据和标程，并且不断补充高质量的编程任务作为练习或测试用。读者（包括教师或学生）可在 OJ 上练习、实验、测试和上机考试。如果您所在学校尚未建立 OJ 系统，可自主开发 OJ，也可利用开源系统部署自己的 OJ，或者直接利用互联网上开放的 OJ 系统。如果 OJ 上没有想要练习的编程任务，则需要先在 OJ 上添加它。欢迎使用湖南农业大学程序设计在线练习系统（http://210.43.224.19/oj）。

本书适合作为本科低年级程序设计课程教材，也非常适合程序设计初学者自学使用。对参加奥林匹克信息学竞赛的队员和参加 ACM/ICPC 大学生程序设计竞赛的队员来说，也是一本非常好的入门教材。对于有一定程序设计基础的读者，本书也不失参考价值。书中有许多对程序深入的剖析很有启发意义，值得一读。

众所周知，C 程序设计课程是计算机类专业、信息类专业极其重要的专业基础课。我从事本科程序设计专业基础课一线教学十余年，希望能有一本读起来不那么枯燥，同时又不失专业性和系统性的面向程序设计初学者的 C 语言图书，这是我写本书的意图。如何利用本书，各位见仁见智。

希望通过本书带给读者更多愉悦的程序设计经历，提振编程信心，激发编程的兴趣，为今后的学习、工作、科研培养良好的计算思维和软件设计基础。

本书的写作是我将头脑中纷繁的思绪变成有条理文字的过程，既艰辛又充满快乐。常常为了设计一个恰到好处的编程任务或为了更好地表述某个概念，灵感突现，即使是已卧床或半夜醒来，也立刻记录，唯恐遗漏。本书力求知识更加系统、表述更加准确、语言更加通俗、例子更加贴近生活，这使写作过程充满挑战性，字句斟酌，直到自己满意为止，以致成书过程如此漫长。对本书内容安排、章节设置、设置每个例子代码甚至每段表述，都经过反复琢磨和权衡，力求语言描述精准、思想表达透彻。漫长的成书过程，让我体会到了写书的不易，不过，本书写作过程带给我更多的是快乐。在写作期间，不仅有将存在于脑海的点滴心得随着键盘的敲击变成文字的快感，而且，在此期间我的儿子不经意间长成了帅小伙，陪伴他的时间总是短暂而欢快的。我的妻子虽常常担心因长时伏案而有腰椎疾病的我，但她送来键盘旁的一杯热茶、一碟水果，顿时让我满血复活。特别感谢我的妻子陈海燕女士包容我无数个日夜以计算机为伴而少有陪伴她，家务操持多劳她费心，虽偶有抱怨，但忍韧而坚强。谨以此书献给我的家人。

感谢电子工业出版社高等教育分社谭海平社长和冉哲编辑对我蜗牛般写稿进度的容忍。

感谢我的学生卢晨曦、邵振宇、王舒心、王鹏、陈慧、张洋、唐朝宇、廖颜勤、姚沛丰、熊嘉奇、唐航、周子翔、沈煜恒为本书的校对付出了辛勤劳动。

虽然我对本书写作用心尽力，但由于学识水平有限，错误与不足之处在所难免，恳请批评指正（我的邮箱：390199309@qq.com）。

限于篇幅，本书第 1 章至第 9 章的综合应用实例和知识拓展部分以及附录部分，以扫描二维码下载相应文档的形式提供。

附录：

贺细平

目　录

第1章　邂逅程序设计——初识C语言

生活在信息时代的我们在日常生活中已经离不开计算机和软件（程序）了。

日常生活中最常见的“计算机”是微型计算机，俗称“电脑”，有台式电脑、笔记本电脑、平板电脑、工作站、服务器等。顾名思义，可以理解为“用电的大脑”、“有着闪电般计算速度的大脑”、“会计算的机器”。甚至，日常使用的智能手机也可以看作掌上计算机。

我们几乎每天都在浏览互联网、收发电子邮件、在线聊天、玩游戏、看电影、听音乐、写文档、绘图、打电话、发短信、发微博、发微信等，数不胜数，这些都离不开各种各样的程序和软件。这些软件运行在各式各样的电子设备中，如台式计算机、笔记本电脑、平板电脑、手机游戏机、MP3、DVD、数码相册、导航仪、自动柜员机，甚至在电视机、机顶盒、冰箱、洗衣机、空调等家电中都有“程序”在运行。

想知道这些软件或计算机程序是怎么设计吗？我们能设计这样的软件或程序吗？

千里之行，始于足下。本书将为读者开启神奇的程序设计之门。

一直以来，我们是软件或程序的使用者、程序的消费者，直到今天，我们才有机会成为程序的设计者、生产者或创造者。编程能让我们实现自我价值。从此我们将与程序设计结下不解之缘，我们将在学习中体会程序设计的奇妙、程序设计的美和程序设计的魅力。从今天起，当我们在设计程序时，角色已悄然发生了改变，由“看热闹的外行”转变为“看门道的内行”，程序设计人员通常称为程序员，你想成为编程高手吗？你想成为出色的程序员吗？本书将为您打开充满趣味和奥妙的程序设计大门，现在开始我们愉快的程序设计之旅吧！

程序设计语言有很多种，如C++、Java、C#、Python、PHP、Perl、JavaScript、VisualBasic、Pascal、Fortran、Delphi、Ruby等。C语言仍然是值得学习的语言，因为C语言简洁精练、操控能力强、程序运行效率高、能很好地锻炼计算思维。此外，C语言是系统编程的首选语言，许多操作系统（如Windows、Linux、Android等）主要使用C语言实现。

“计算机”（电脑）解决问题的方式与“人”解决问题方式大太一样。通过本门课程的学习，你将有深刻体会。所谓计算思维，通俗地说，就是以计算机的特有方式去思考问题、解决问题。可以说掌握了计算思维，就掌握了程序设计的本质。

计算机有最核心能力，体现在3个方面：快速计算能力、海量存储能力、高速通信能力。

计算机深入到了生活的方方面面，具有各种神奇功能或“智能”，所有的这些功能都离不开软件（或程序）。人们设计各种各样的程序实现特定的功能来满足生产生活的各种需要。程序的功能是通过程序代码控制计算机的运行，从而达到预定的目标的。这些目标包括得到计算结果、控制外部设备、可视化呈现等。

虽然计算机的功能、形态各异，但目前绝大多数计算机的组织结构仍是冯·诺依曼计算机体系结构。按照冯·诺依曼计算机体系结构的思想，计算机系统有如下特点。

（1）采用二进制。数据和指令都采用二进制，而非十进制。因为二进制只有两种状态，用0和1表示，如开关电路的“通”与“断”，因此易于用电子电路实现。

（2）存储程序。数据和指令都以相同的方式存放在存储器，依次取出指令，分析指令的含义，然后执行指令。执行指令的过程如下：读取操作数，根据所要执行的计算操作计算出结果，

存放结果。程序和指令无区别地统一存储使计算机的硬件易于设计与实现。

（3）顺序执行。指令是线性地、串行地、逐条指令地被执行。串行和线性意味着程序的执行有严格的先后顺序，这使得程序易于设计与调试，易于理解和把握。

（4）五大部分。计算机硬件由运算器、控制器、存储器、输入设备和输出设备组成。

冯·诺依曼计算机体系结构给出了计算机的组成和运行的宏观蓝图，这能帮助我们更好地理解计算机、程序及程序设计。

计算机程序执行模式是串行模式，即程序代码或计算机指令是顺序地、串行地被执行。因此程序代码可以单步调试。并行程序本质上是多个串行程序的并发运行，因此串行程序设计是并行程序设计的基础。

小问答：

问：软件和程序有什么区别？

答：软件一般是指具有较完整功能的、相对复杂的程序。

1.1 第一个程序——我会算加法

编写第一个程序。

想看看程序长得像什么样子了吗？马上开始编写属于自己的第一个程序。

按如下任务的要求，编写一个 C 语言程序实现指定的功能。

编程任务 1.1：我会算加法

任务描述：给定任意两个整数 A、B，输出两者之和。

输入：两个整数 A、B，两个数之间用空格分隔。

输出：A 与 B 的和。

输入举例：12 34

输出举例：46

分析：本问题虽简单，但具有典型“三部曲”结构：输入—计算—输出。程序必须能接受输入的两个整数，然后做求和运算，最后将结果输出。输入采用 C 语言的库函数 scanf( )完成，求和利用 C 语言的算术运算之一，即加法运算符“+”实现，输出结果利用 C 语言的库函数 printf( )函数实现。

“工欲善其事，必先利其器”，让我们先来熟悉编程工具和编程环境。可用于 C 程序设计的集成开发环境（Integrated Development Enviornmont，IDE）有许多。本书采用 Code::Blocks 作为 C 程序设计的集成开发环境。

操作步骤如下，共有 4 大步。

第 1 大步：在 Code::Blocks 中创建一个名为 aAddb 的新工程。

第 1.1 步：运行 Code::Blocks。第一次运行前，需要在计算机安装此软件。可从 www.codeblock.org 下载最新版 Code::Blocks 安装包，它是开源软件。如果在 Windows 下使用，建议下载带有 MinGW 编译环境的安装包，它带有 gcc 编译器和 gdb 调试器，如 codeblocks-12.11mingw-setup.exe 安装包。Code::Blocks 运行后界面如图 1.1 所示。

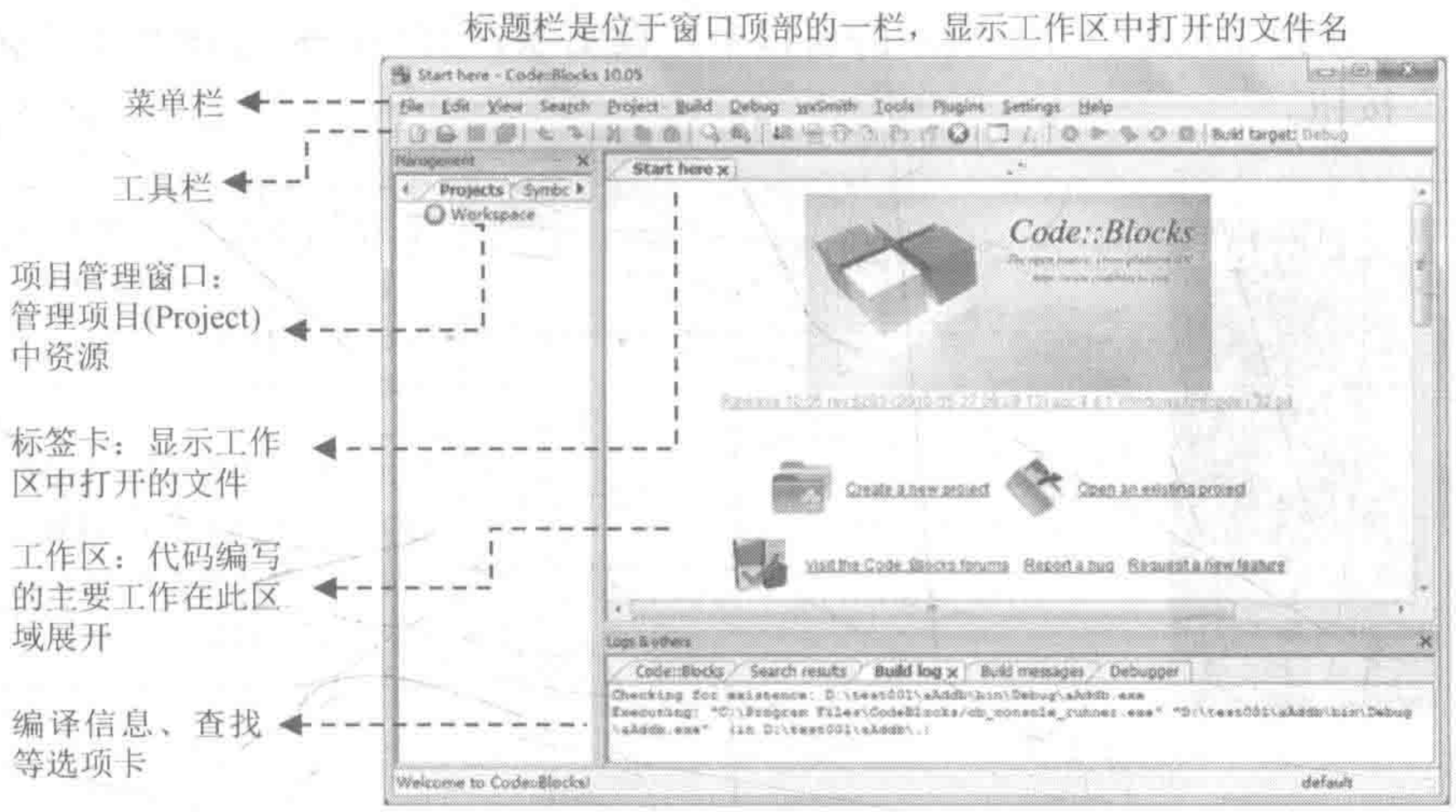
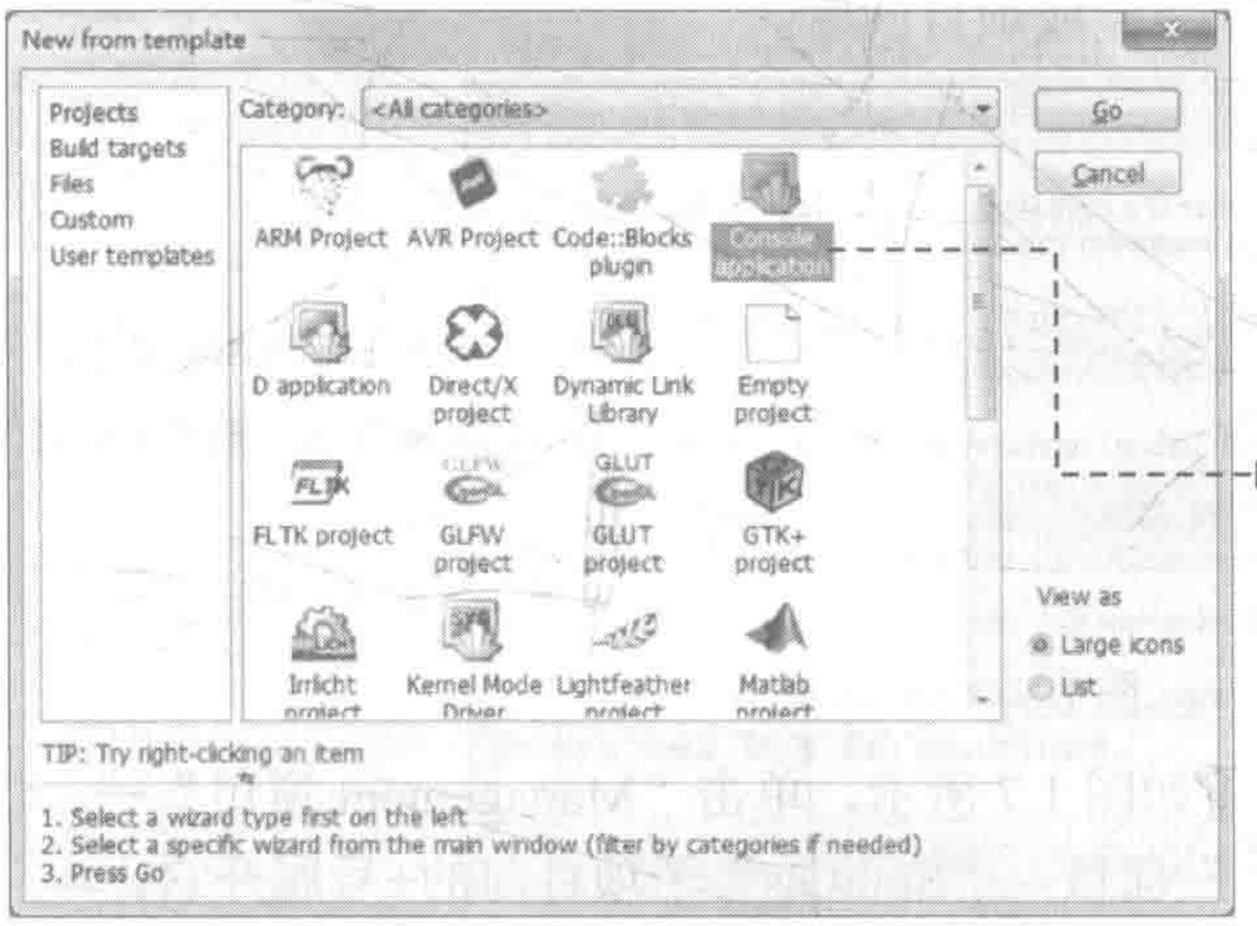


图 1.1　Code::Blocks 用户界面

第 1.2 步：打开菜单“File—New—Project”选项新建软件项目，如图 1.2 所示。在此窗口中选择 Console application 选项，表示新建项目类型为控制台应用程序。



选择“Console application”选项，建立控制台应用程序。

控制台应用程序有如下特征：程序的标准输入是从键盘输入的，标准输出是向屏幕输出的；一般不支持鼠标操作；程序的运行界面一般是字符界面，不是通常所见的Windows图形化用户界面。控制台应用程序的环境简单、适合程序设计初学者。本书所有程序都为控制台应用程序。

图 1.2　选择新建项目类型

然后单击“Go”按钮，弹出如图 1.3 所示的窗口，勾选 Skip this page next time 复选项，以后新建项目时将不再出现此提示窗口。在此窗口单击“Next”按钮。

第 1.3 步：在弹出如图 1.4 所示的窗口中选择“C”语言作为程序设计语言。

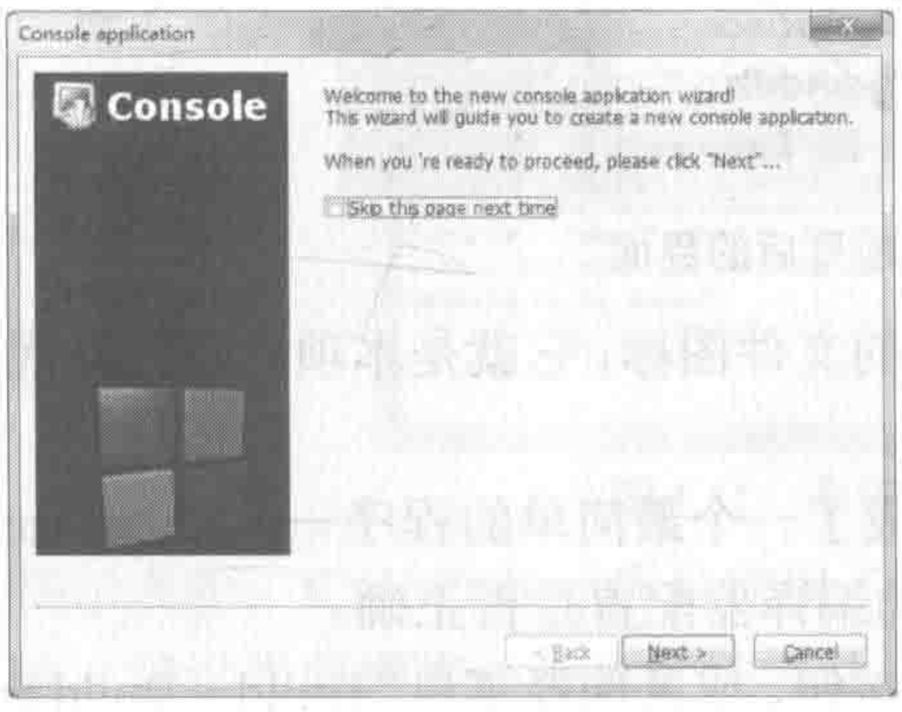


图 1.3　新建项目时的提示

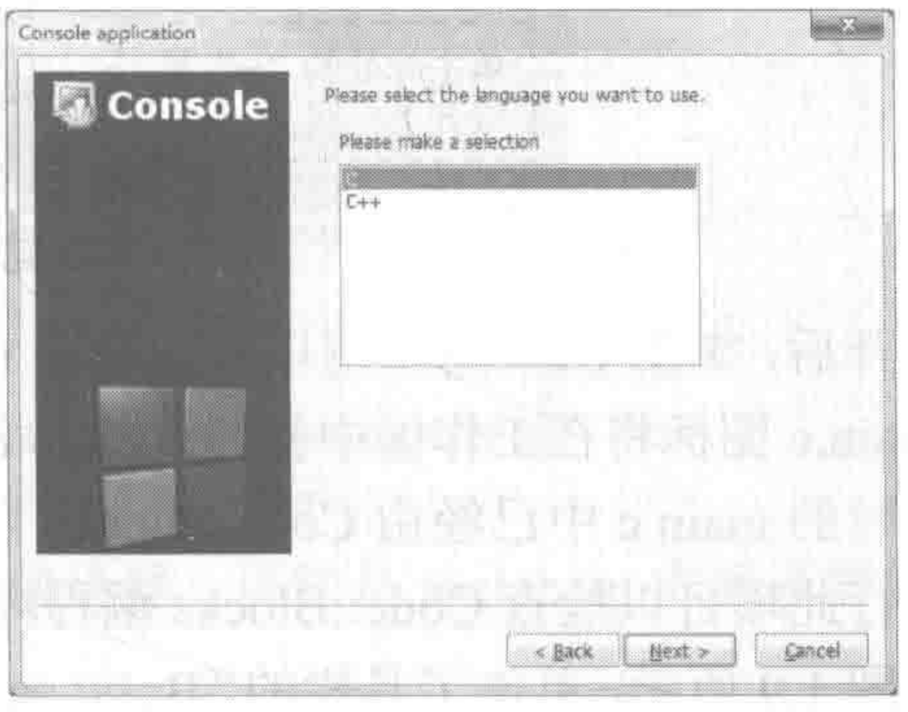


图 1.4　选择程序设计语言

第 1.4 步：在弹出的如图 1.5 所示的窗口中，设定项目的名称并选择项目存放的文件夹。单击“Next”按钮。

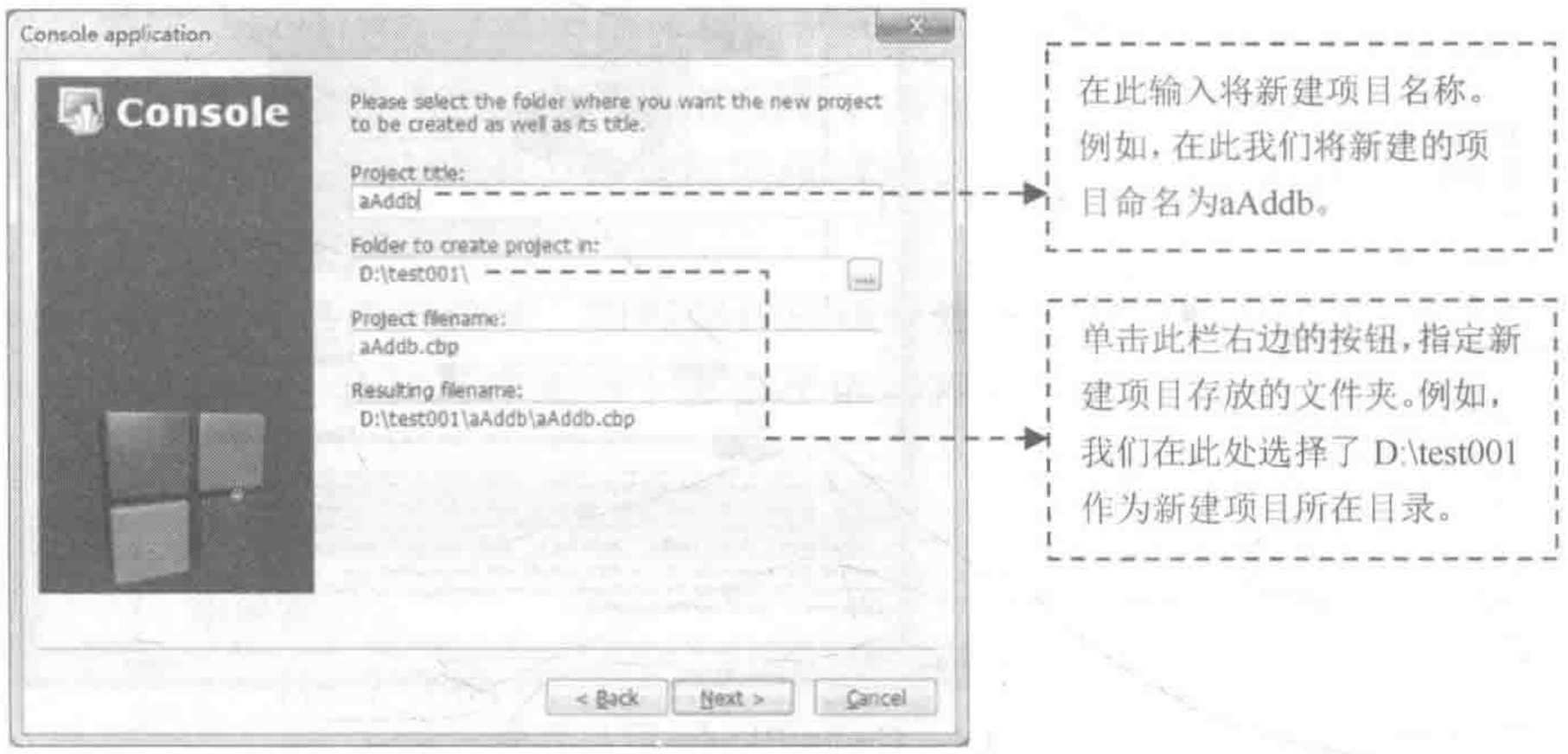


图 1.5　设定项目名称和项目所在文件夹

第 1.5 步：选择编译器，如图 1.6 所示。此处选择默认的 GNU GCC Compiler，即 GCC 编译器即可，直接单击“Finish”按钮，完成项目创建。

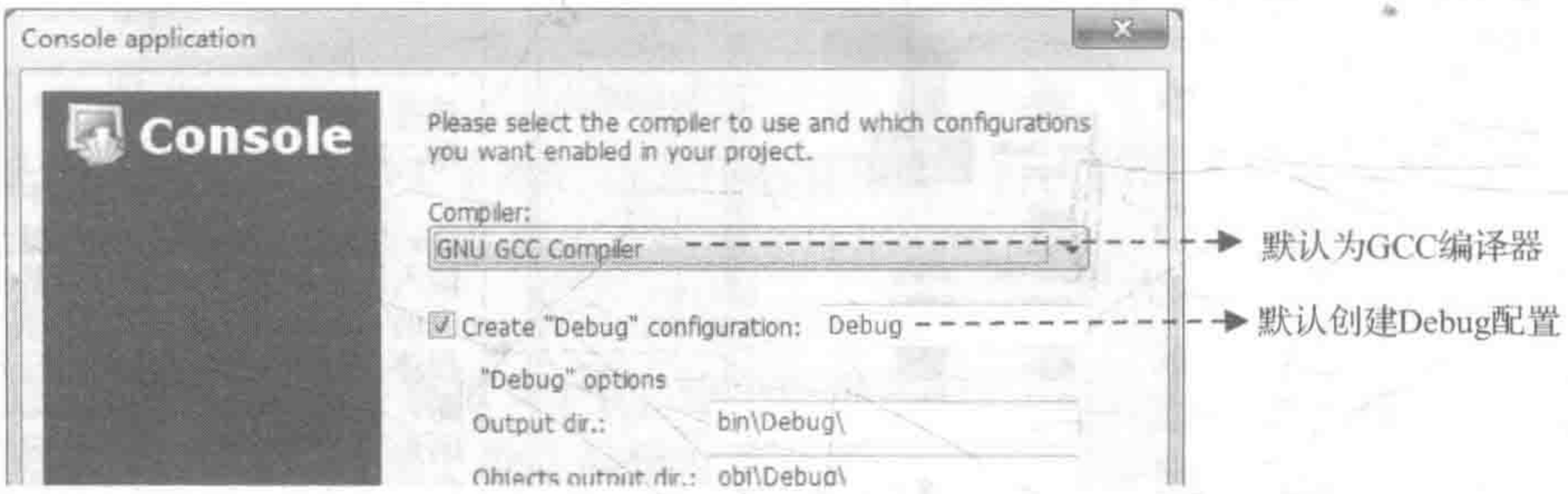


图 1.6　选择编译器

此时，Code::Blocks 编程环境将如图 1.7 所示。单击“Management 窗口”—“Projects”选项卡—“Workspace”—“aAddb”项目—“Sources”子项目前的+号展开它。

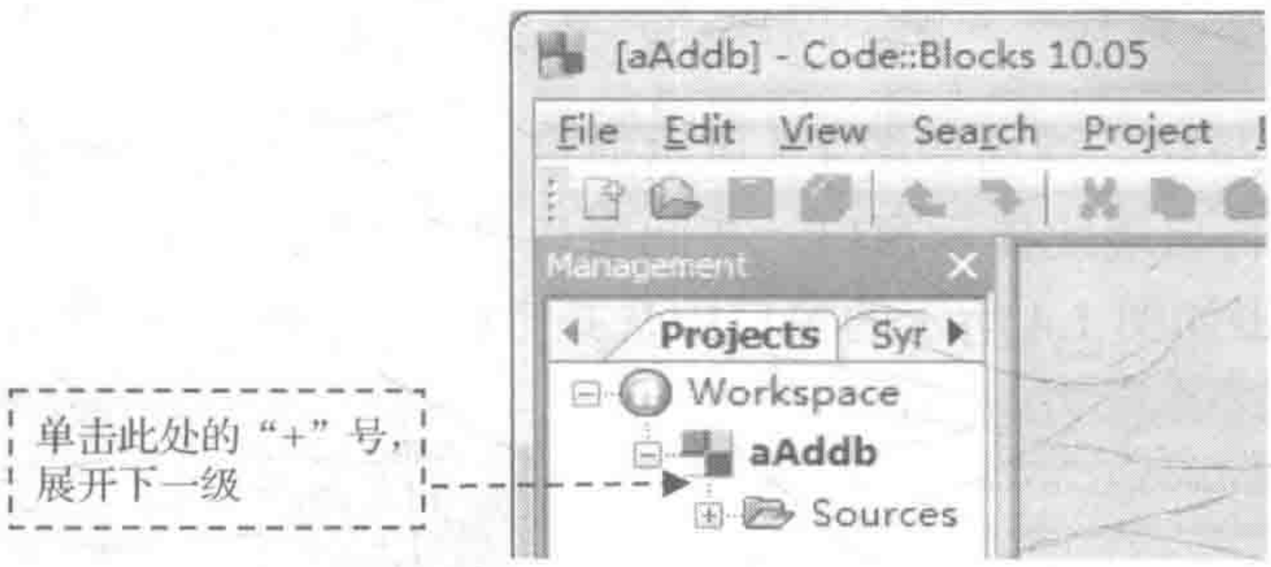


图 1.7　新建 aAddb 项目后的界面

展开后，如图 1.8 所示。可以看到名为 main.c 的文件图标，它就是本项目下的源代码文件。双击 main.c 图标将在工作区中打开 main.c 源文件。

此时的 main.c 中已经由 Code::Blocks 自动生成了一个最简单的程序——Hello world。

运行此项目以检查 Code::Blocks 编程环境下的编译器配置是否正确。

如图 1.9 所示，单击工具栏的“Bulid and run”按钮，或者选择菜单“Build—Build and run”，或者直接按快捷键 F9，就能编译并运行当前程序。

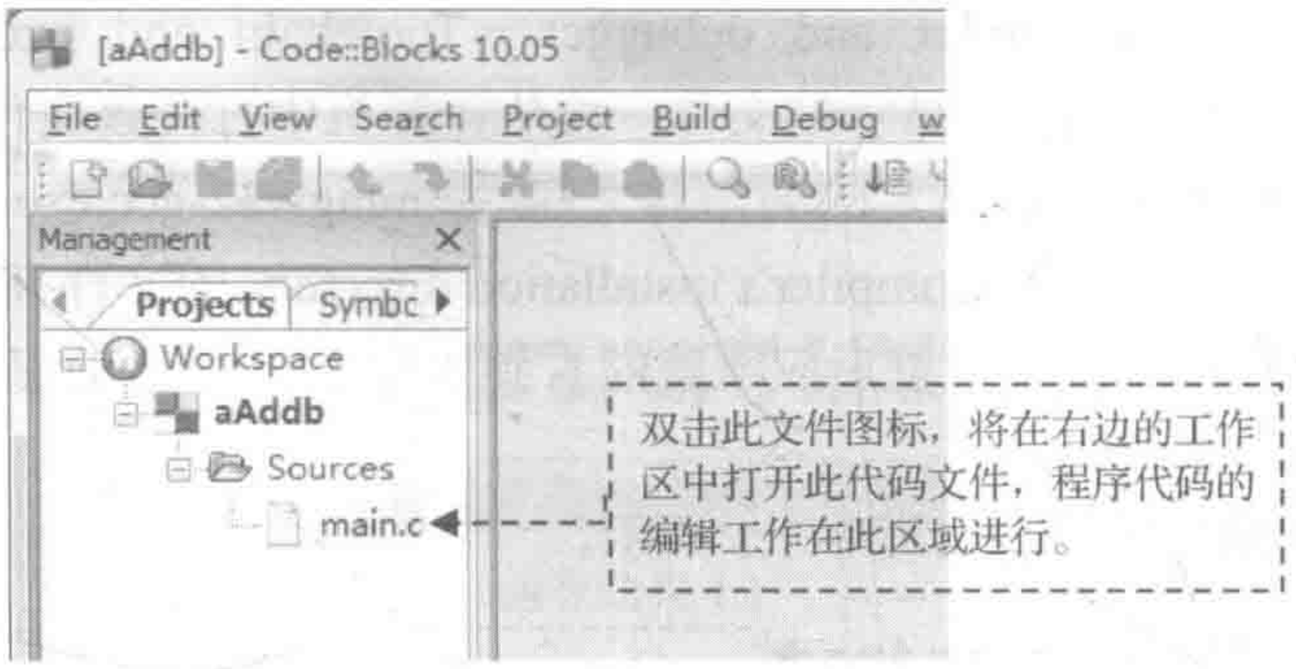


图 1.8　展开项目下的 main.c 文件

图 1.9　运行项目

如果能看到如图 1.10 所示运行结果窗口，意味着 Code::Blocks 编程环境设置正确。接下来就可以在 main.c 文件中编写程序代码了。

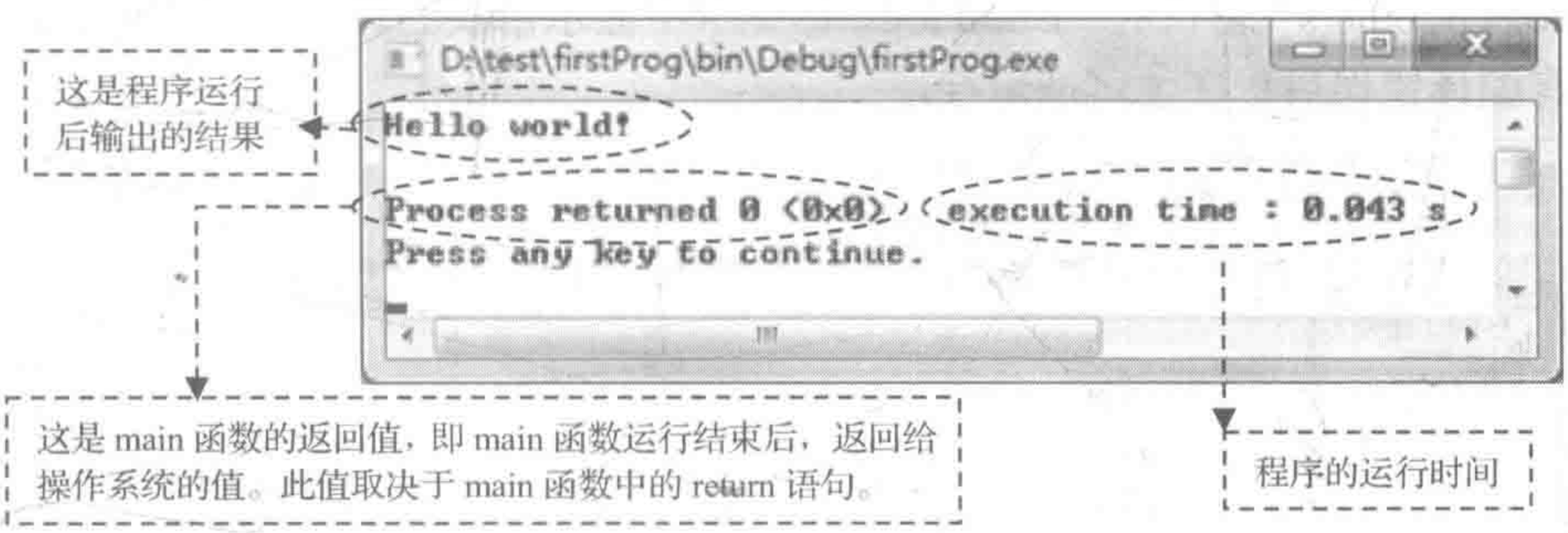


图 1.10　Hello world 程序的运行结果

如果运行后不能看到图 1.10 所示的窗口，并且在工作区下方的 Logs & others 窗口中显示了如下信息，如图 1.11 所示。

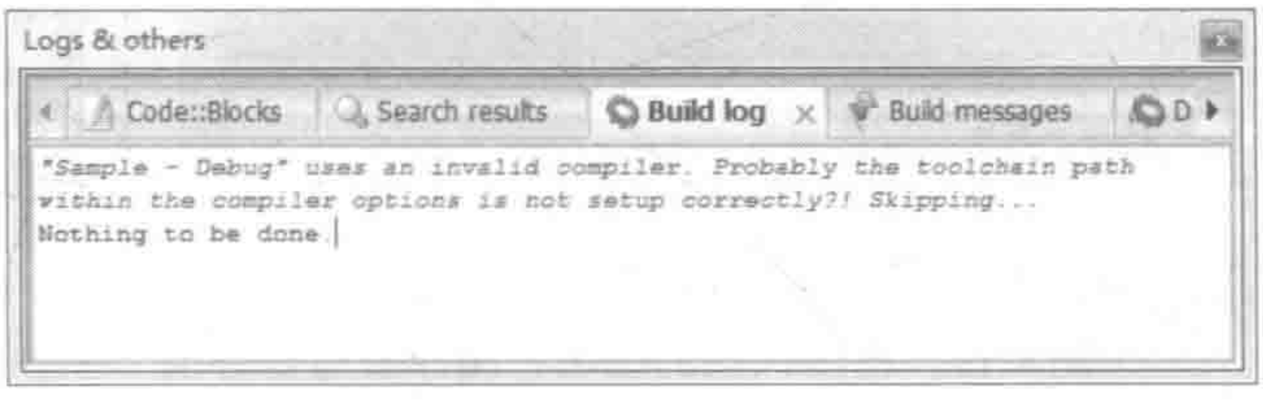


图 1.11　未正确设置编译器时的提示

该提示表明：XXX-Debug 项目使用了无效的编译器。可能是 toolchain 中编译器的路径设置不正确，跳过当前操作，未执行任何动作。

以上错误是因为 Code::Blocks 中编译器设置错误引起的，那么如何正确设置编译器呢？