



大学生创意·创新·创业教育与实践系列教材
2014年国家级教学成果一等奖作品

程序设计算法基础

(大学生程序设计竞赛推荐教材 I)

主编 梁冰 冯林
主审 吴文虎 房鸣

高等教育出版社



大学生创

教育与实践系列教材

2014年国家级教子成果一等奖作品

CHENGXU SHEJI SUANFA JICHU

程序设计算法基础

(大学生程序设计竞赛推荐教材 I)

主编 梁冰 冯林 副主编 高品 孙木鑫 张璨
主审 吴文虎 房鸣

高等教育出版社·北京

内容简介

这是一本面向计算机专业或者计算机爱好者的算法教材,旨在将更多对程序算法感兴趣,但又苦于无从入手的同学带进算法的大门。全书共分11章,第1章介绍Linux操作系统与C++编程环境,第2章简单介绍初中级算法,第3章介绍基础数据结构,第4章介绍枚举、递推、递归、贪心、分治、哈希和二分等基础算法设计,第5章介绍简单排序算法,第6章介绍图论的相关知识,第7章介绍并查集和线段树两种高级数据结构,第8章介绍KMP、字典树、Z算法和马拉车算法等处理字符串的数据结构,第9章介绍深度优先搜索、宽度优先搜索、双向宽度优先搜索、A*搜索和一些剪枝常用的策略,第10章介绍初等数论,第11章介绍动态规划,重点讲述背包问题。

本书可作为高等学校计算机专业、软件工程专业教学用书,以及ACM大赛参考用书。

图书在版编目(CIP)数据

程序设计算法基础 / 梁冰, 冯林主编. --北京:
高等教育出版社, 2018.5

ISBN 978-7-04-049192-0

I. ①程… II. ①梁… ②冯… III. ①电子计算机—
算法理论—高等学校—教材②程序设计—高等学校—教材
IV. ①TP301.6②TP311.1

中国版本图书馆CIP数据核字(2017)第326416号

策划编辑 孙琳 责任编辑 孙琳 封面设计 李小璐 版式设计 童丹
插图绘制 杜晓丹 责任校对 李大鹏 责任印制 刘思涵

出版发行	高等教育出版社	网 址	http://www.hep.edu.cn
社 址	北京市西城区德外大街4号		http://www.hep.com.cn
邮政编码	100120	网上订购	http://www.hepmall.com.cn
印 刷	山东临沂新华印刷物流集团有限责任公司		http://www.hepmall.com
开 本	787mm×1092mm 1/16		http://www.hepmall.cn
印 张	18.75	版 次	2018年5月第1版
字 数	370千字	印 次	2018年5月第1次印刷
购书热线	010-58581118	定 价	38.60元
咨询电话	400-810-0598		

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换

版权所有 侵权必究

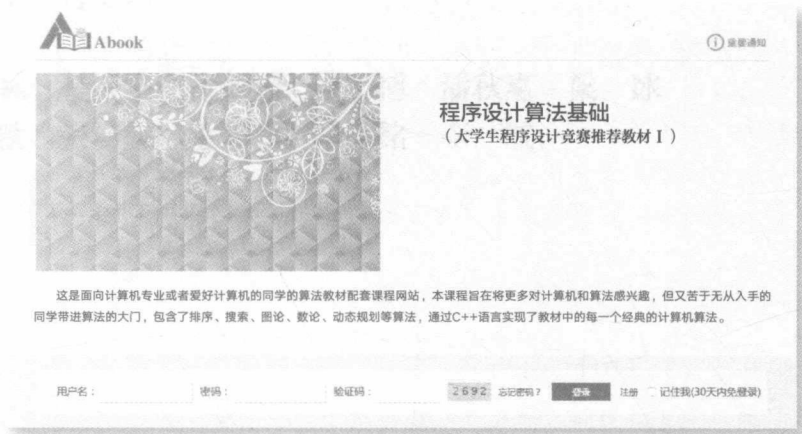
物料号 49192-00

程序设计 算法基础

(大学生程序设计竞赛
推荐教材 I)

主编 梁 冰
冯 林
主审 吴文虎
房 鸣

- 1 计算机访问<http://abook.hep.com.cn/1248725>, 或手机扫描二维码、下载并安装 Abook 应用。
- 2 注册并登录, 进入“我的课程”。
- 3 输入封底数字课程账号(20 位密码, 刮开涂层可见), 或通过 Abook 应用扫描封底数字课程账号二维码, 完成课程绑定。
- 4 单击“进入课程”按钮, 开始本数字课程的学习。



课程绑定后一年为数字课程使用有效期。受硬件限制, 部分内容无法在手机端显示, 请按提示通过计算机访问学习。

如有使用问题, 请发邮件至 abook@hep.com.cn。



扫描二维码
下载 Abook 应用

<http://abook.hep.com.cn/1248725>

大学生创意·创新·创业教育与实践系列教材

编写委员会

主任：冯林

副主任：吴振宇 付冬娟

委员：冯林 吴振宇 付冬娟 金博 潘秋惠 梁冰
邱铁 张崴 姚翠莉 李胜铭 李航

前言

这是一本关于算法的教材。算法是一系列解决问题的清晰指令，可以说是程序设计的灵魂。同一问题可用不同算法解决，而一个算法的质量优劣将影响程序的执行效率。算法分析的目的在于选择合适算法和改进算法。评价一个算法的好坏主要是通过算法运行的时间长短和占用空间的大小来考虑。对于计算机相关专业的学生或者爱好计算机的学生来说，无论是学习还是工作，或多或少都会应用到一些算法的知识。而目前国内大型互联网公司招聘时的笔试和面试都以算法为主，可见算法的重要性是不言而喻的。ACM ICPC (ACM International Collegiate Programming Contest, ACM 国际大学生程序设计竞赛) 是一项由美国计算机协会主办的旨在展示大学生创新能力、团队精神，以及在压力下编写程序、分析和解决问题能力的年度竞赛。ACM 程序设计竞赛的题目强调算法的高效性与正确性。参赛选手只有编写出能够在规定时间内运行完成若干组严格的测试数据且结果全部正确的程序，才能得到分数。本教程将以 ACM 程序设计竞赛的题目为基础，介绍一些比较初级的算法。

本书是面向初学者的一本算法教材。即使是从未接触过算法，甚至还没有接触过编程语言，都可以将本书当作算法入门的读物。本书旨在将更多对计算机和算法感兴趣，但又苦于无从入手的同学带进算法的大门。本书依次介绍了一些包括排序算法在内的、基础的数据结构和算法设计。相信当读者掌握了这些内容之后，会对算法和程序设计有一个新层次的认识并产生浓厚的兴趣；之后重点介绍并查集、线段树和一些字符串处理方面的高级数据结构，还将介绍搜索、图论、动态规划和数论等程序设计竞赛中常用到的算法。对于每个算法，本书都有图文并茂的翔实讲解；每章节的后面都有针对该节知识点的例题讲解，每道例题都是国内外著名程序在线判题系统中的原题，而且对每道例题，都会从理解题意开始，详细讲解解题的思路，并附有完整的可以正确通过测试样例的代码，供读者研究学习。除了例题，每一章的最后还有一些练习题供读者巩固本节中学到的知识，如果读者对这些习题仍感觉无从下手，可以参考每道练习题后附带的思路分析来帮助整理解题思路。

本书共分 11 章，第 1 章介绍 Linux 操作系统与 C++ 编程环境；第 2 章简单介绍初级算法；第 3 章介绍基础数据结构；第 4 章介绍枚举、递推、递归、贪心、分治、哈希和二分等基础算法设计；第 5 章介绍简单排序算法；第 6 章介绍图论的相关知识，包括最短路径问题和最小生成树问题的一些经典算法；第 7 章介绍并查集和线段树两种高级数据结构；第 8 章介绍 KMP、字典树、Z 算法和马拉车算法等处理字符串的数据结构；第 9 章介绍搜索的相关算法，包括深度优先搜索、宽度优先搜索、双向宽度优先搜索、A* 搜索和一些剪枝常用的策略；第 10 章介绍初等数论；第 11 章介绍动态规划，重点介绍背包问题。本书的例题代码都是集训队成员测试提交通过的正确代码。

大连理工大学是全国高校较早倡导并开展创新创业教育的学校，自 1984 年以来，学校大力开展以突出创新创业实践为特色的创新创业教育。1995 年在全国率先成立以学生创新创业教育为主体的教学改革示范区——创新教育实践中心，开展创新创业教育课程体系、教学内容、教学方法、教学模式等方面的改革，探索与之配套的管理运行机制，将创造性思维与创新方法融入教学实践中，在课堂教学中树立 CDIO 工程教育新理念，倡导“做中学”，在实践环节构建了“个性化、双渠道、三结合、四层次、多模式”的创新教育实践教学新体系，取得了一系列成果，在全国高校产生了很大的影响。“创造性思维与创新方法”和“创新教育基础与实践（系列）”课程分别被评为国家精品资源开放课程，“大学生程序设计竞赛初级教材”是“创新教育基础与实践”系列课程的核心课程，面向大连理工大学 ACM 创新实践班学生开设。

本书是由梁冰、冯林组织大连理工大学 ACM 集训队员在多年的教学案例和训练题库的基础上编写完成的。参加编写的 ACM 历届集训队员有高品、张璨、孙木鑫、刘卓、胡骏、戴宇心、杨文冕、赵汉光、陈梁坚、刘博等同学，鉴于他们在本书的编写和代码的调试中付出的辛勤劳动，在此对他们的工作表示衷心的感谢。本书内容丰富，除了教材提供的内容外，还提供了部分扩展资源和应用实例，为了节省纸张，这些扩展资源和应用实例可在高等教育出版社相关网站下载。

此外，在本书的撰写过程中，除了参考文献和正文中标出的引用来源外，还参考了国内外的相关研究成果和网站资源，没有一一列出，在此感谢涉及的所有单位、专家和研究人員。

担任本书主审的是清华大学吴文虎教授、北京邮电大学计算机学院房鸣教授，感谢两位教授提出的宝贵意见和建议。因编者水平有限，书中的错误和不足之处在所难免，欢迎广大读者来信 Liangbing@dlut.edu.cn 批评指正，帮助我们不断地完善本书。

编者

2017 年 8 月

目 录

第 1 章 Linux 操作系统与编程环境	1	3.3.1 vector	55
1.1 Linux 基础	1	3.3.2 set	58
1.2 编译器	1	3.3.3 map	60
1.2.1 Code::Blocks 安装	2	3.3.4 priority_queue	62
1.2.2 Code::Blocks 编程环境配置	2	3.3.5 例题讲解	64
1.2.3 Code::Blocks 编写程序	3	3.4 练习题	67
1.3 编译 C++ 文件	7	第 4 章 基本算法设计	69
1.4 ACM 国际大学生程序设计竞赛	8	4.1 枚举	69
1.5 自动评测系统	11	4.1.1 枚举算法的定义	69
1.5.1 评测系统反馈	11	4.1.2 枚举算法的解题过程	69
1.5.2 国内知名评测系统	12	4.1.3 枚举算法的特点	70
第 2 章 算法入门	13	4.1.4 例题讲解	70
2.1 快速幂取模算法	13	4.2 递推	73
2.1.1 模运算	13	4.2.1 递推的概念	73
2.1.2 幂取模的计算	16	4.2.2 递推与数列	73
2.1.3 例题讲解	19	4.2.3 斐波那契数列	73
2.2 算法	21	4.2.4 递推的两种顺序	74
2.2.1 算法的定义	21	4.2.5 例题讲解	74
2.2.2 学习算法的意义	22	4.3 递归	76
2.2.3 算法复杂度分析	22	4.3.1 递归的定义	76
第 3 章 基本数据结构	28	4.3.2 递归的要求	76
3.1 基本线性数据结构	28	4.3.3 递归与递推	77
3.1.1 线性表	28	4.3.4 例题讲解	78
3.1.2 栈	35	4.4 贪心算法	80
3.1.3 队列	37	4.4.1 贪心算法的概念	80
3.1.4 例题讲解	40	4.4.2 贪心算法的原理	80
3.2 二叉搜索树	45	4.4.3 例题讲解	81
3.2.1 二叉搜索树的定义	45	4.5 分治算法	83
3.2.2 二叉搜索树的实现	47	4.5.1 分治的基本思想	83
3.3 C++ 标准模板库	55	4.5.2 分治的一般解题步骤	83

4.5.3	分治的特点	84	6.2.1	图的深度优先遍历	135
4.5.4	归并排序	84	6.2.2	图的宽度优先遍历	137
4.5.5	例题讲解	85	6.2.3	图的拓扑排序	137
4.6	模拟	85	6.2.4	例题讲解	140
4.6.1	高精度计算	86	6.3	最小生成树	140
4.6.2	矩阵运算	91	6.3.1	Kruskal 算法	140
4.6.3	例题讲解	94	6.3.2	Prim 算法	142
4.7	哈希	97	6.4	单源最短路径	143
4.7.1	直接寻址表	97	6.4.1	Dijkstra 算法	144
4.7.2	哈希表	98	6.4.2	Bellman-Ford 算法	146
4.7.3	例题讲解	104	6.4.3	SPFA 算法	147
4.8	二分法	107	6.4.4	差分约束系统	149
4.8.1	二分查找	107	6.4.5	例题讲解	150
4.8.2	二分逼近	108	6.5	每对顶点的最短路径	154
4.8.3	求解性问题的二分策略	110	6.5.1	最短路径和矩阵乘法	154
4.8.4	例题讲解	112	6.5.2	Floyd 算法	157
4.9	练习题	115	6.5.3	例题讲解	158
第 5 章	排序算法	118	6.6	练习题	161
5.1	基于比较的排序算法	118	第 7 章	并查集和线段树	164
5.1.1	简单排序	118	7.1	并查集	164
5.1.2	快速排序	122	7.1.1	并查集的基本概念	164
5.1.3	限制和优势	123	7.1.2	并查集的操作	165
5.2	基于统计的排序算法	124	7.1.3	例题讲解	171
5.2.1	计数排序	124	7.2	线段树	176
5.2.2	基数排序	125	7.2.1	线段树的概念与性质	176
5.3	例题讲解	127	7.2.2	线段树的基本操作	178
5.4	练习题	127	7.2.3	例题讲解	186
第 6 章	图的基本算法	128	7.3	练习题	190
6.1	图的定义及存储方法	128	第 8 章	字符串问题	193
6.1.1	图的定义	128	8.1	Trie 树	193
6.1.2	有向图和无向图	129	8.1.1	Trie 树的基本概念	193
6.1.3	路径与连通	129	8.1.2	Trie 树的操作	194
6.1.4	图的存储结构	130	8.1.3	例题讲解	197
6.2	图的遍历及拓扑排序	135	8.2	KMP 算法	200

8.2.1	BF 算法简介	200	10.2.3	例题讲解	240
8.2.2	KMP 算法原理和实现	201	10.3	线性方程与同余方程	240
8.2.3	例题讲解	205	10.3.1	线性方程	240
8.3	Z 算法与 Manacher 算法	207	10.3.2	例题讲解	241
8.3.1	Z 算法	207	10.4	乘法逆元	241
8.3.2	Manacher 算法	210	10.4.1	整数集合下逆元的求解方法	241
8.3.3	例题讲解	214	10.4.2	例题讲解	242
8.4	练习题	214	10.5	中国剩余定理	242
第 9 章	搜索	217	10.5.1	中国剩余定理	242
9.1	状态空间和状态空间搜索	217	10.5.2	中国剩余定理的扩展	244
9.2	宽度优先搜索	218	10.6	质数筛法与质因数分解	246
9.2.1	基本概念	218	10.6.1	埃拉托斯特尼 (Eratosthenes) 筛法	246
9.2.2	算法分析与实现	218	10.6.2	欧拉 (Euler) 筛法	246
9.2.3	例题讲解	219	10.6.3	质因数分解	247
9.3	深度优先搜索	223	10.7	欧拉函数	249
9.3.1	基本概念	223	10.7.1	欧拉函数与欧拉定理	249
9.3.2	算法分析与实现	223	10.7.2	例题讲解	251
9.3.3	例题讲解	224	10.8	原根与剩余系	251
9.4	双向宽度优先搜索	226	10.9	指数方程与高次同余方程	253
9.5	A^* 搜索	227	10.9.1	指数方程	253
9.5.1	基本概念	228	10.9.2	高次同余方程	257
9.5.2	算法分析与实现	228	10.9.3	例题讲解	260
9.5.3	例题讲解	229	10.10	高斯消元	264
9.6	剪枝	229	10.11	练习题	268
9.6.1	基本概念	229	第 11 章	动态规划入门	272
9.6.2	可行性剪枝	230	11.1	动态规划概述	272
9.6.3	最优性剪枝	230	11.1.1	数字三角形	272
9.6.4	例题讲解	230	11.1.2	组合数	274
9.7	练习题	235	11.1.3	动态规划方法求解的问题类型	275
第 10 章	初等数论	237	11.1.4	例题讲解	275
10.1	初等数论简介	237	11.2	背包问题	275
10.2	最大公约数和扩展欧几里得算法	237	11.2.1	0-1 背包	275
10.2.1	欧几里得算法	238	11.2.2	完全背包	277
10.2.2	扩展欧几里得算法	239			

11.2.3 多重背包	278	11.3.3 矩阵链相乘问题	283
11.3 经典动态规划问题	281	11.4 练习题	284
11.3.1 最长上升子序列	281	参考文献	285
11.3.2 最长公共子序列	282		

第 1 章 Linux 操作系统与编程环境

Linux 是在 1991 年发展起来的与 UNIX 兼容的操作系统，可以免费使用。它的源代码可以自由传播且可任意修改、充实、发展，因开发者的初衷是要共同创建一个完美、理想并可以免费使用的操作系统。在介绍完 Linux 操作系统之后，将对 Code::Block 的编译环境进行详细阐述，包括安装、配置环境、编写程序等。

1.1 Linux 基础

Linux 是一个完整的 32 位的多用户多任务操作系统，因此不需要先安装 DOS 或其他的操作系统（MS Windows, OS2, MINIX）就可以进行直接的安装。Linux 最早起源是在 1991 年 10 月 5 日由一位芬兰的大学生 Linux Torvalds 写的 Linux 核心程序的 0.02 版，但其后的发展却几乎都是由互联网上的 Linux 社团（Linux Community）互通交流而完成的。Linux 不属于任何一家公司或个人，任何人都可以免费取得甚至修改它的源代码（source code）。Linux 上的大部分软件都是由 GNU（GNU is Not Unix 的递归缩写，又称革奴计划）倡导发展起来的，所以软件通常都会在附着 GNU Public License（GPL）的情况下被自由传播。GPL 是一种可以免费获得自由软件的许可证，因此 Linux 使用者的使用活动基本不受限制（只要用户不将它用于商业目的），而不必像使用微软产品那样，需要为购买许可证付出高价，还要受到系统安装数量的限制。

Linux 的核心具有 Windows 无法比拟的稳定性和高效性，在不使用 Windows 的情况下，它占用系统资源较少，可以使一台 Intel 486 摇身一变成为高效工作站。对于想要学习 UNIX 的用户来说，Linux 是他们熟悉 UNIX 操作环境，通往“骨灰级”高手境界的一大捷径。最重要的是，Linux 上有公认的世界最好的 C 语言编译器 gcc。

1.2 编译器

用高级计算机语言，如 C、C++，编写的程序需要经过编译器编译，才能转化成机器能够执行的二进制代码。C/C++的编译环境非常多，对于学习 C/C++语

言的初学者而言，编译器并不重要，重要的是学习 C/C++语言本身，不过用好的编译环境进行程序的编写和调试确实很方便。本节将介绍 C/C++语言的编译环境——Codeblocks。

1.2.1 Code::Blocks 安装

Linux 系统下安装 Codeblocks，在控制台下直接输入：`sudo apt-get install codeblocks`，就完成了 Codeblocks 的安装；或者直接在 Ubuntu 软件中心里，搜索关键字 Codeblocks 就能找到并安装。在 Windows 系统下需要下载 Codeblocks 的安装文件进行安装。

1.2.2 Code::Blocks 编程环境配置

首次启动 Code::Blocks，会出现如图 1.1 所示对话框来通知自动检测到 GNU GCC Compiler 编译器，用鼠标单击对话框右侧的 Set as default 按钮，然后再单击 OK 按钮。

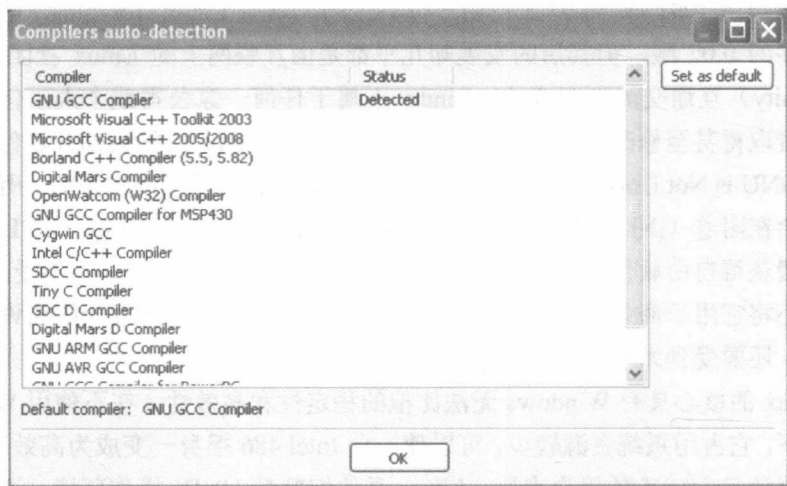


图 1.1 自动检测 GNU GCC Compiler

接下来进入 Code::Blocks 的主界面，弹出一个标签为 Tips of the Day 的对话框，如图 1.2 所示。把“Show tips at startup”前面的勾去掉，然后选择 Close，这样下次启动就不会再出现这个小对话框。

进入 Code::Blocks 主界面，选择主菜单 Settings，弹出一个窗口，如图 1.3 所示。接下来分别对环境 (Environment)、编辑器 (Editor)、编译器和调试器 (Compiler and debugger) 三个子菜单进行配置。

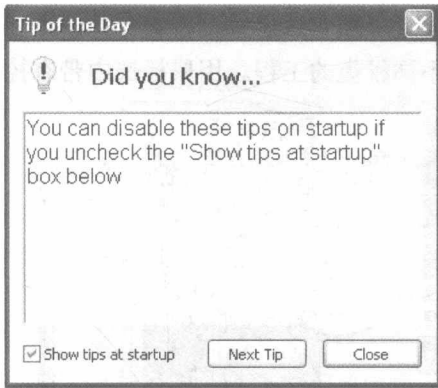


图 1.2 Tips of the Day 对话框

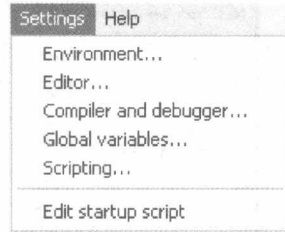


图 1.3 Settings 窗口

1.2.3 Code::Blocks 编写程序

Code::Blocks 创建一个工作空间 (workspace) 跟踪当前的工程 (project)。还可以在当前的工作空间创建多个工程。一个工程就是一个或者多个源文件 (包括头文件) 的集合。源文件 (source file) 就是程序中包含源代码的文件。创建一个工程可以方便地把相关文件组织在一起。一个工程刚建立时, 一般仅仅包含一个源文件。

1. 创建工程

在主菜单 File 的下拉菜单中选择二级菜单 New, 然后从子菜单中选择 Project, 还可以从 Code::Blocks 主界面中选择 Create a new project 进行创建, 如图 1.4 所示。

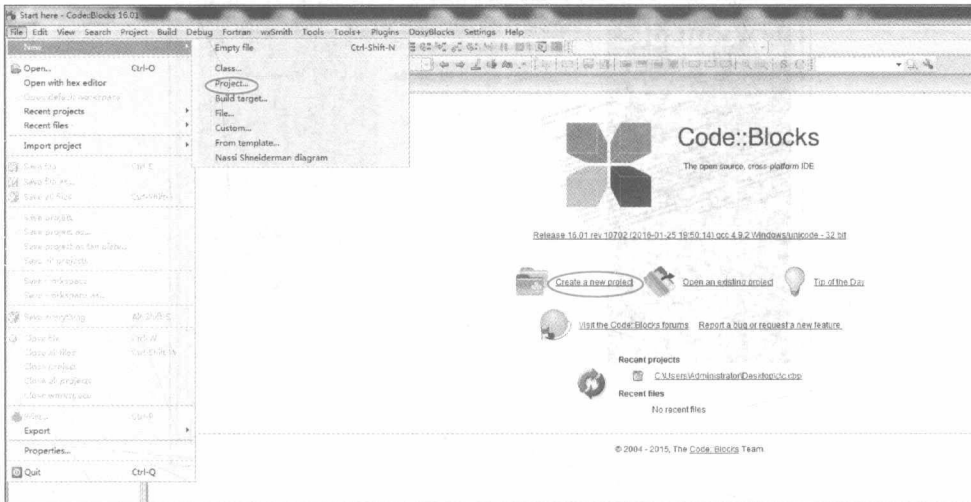


图 1.4 创建工程

创建一个新的工程之后，会打开选择工程模式对话框，如图 1.5 所示。这个窗口中含有很多带有标签的图标，代表不同种类的工程。用鼠标选中带有控制台应用（Console application）标签的图标。

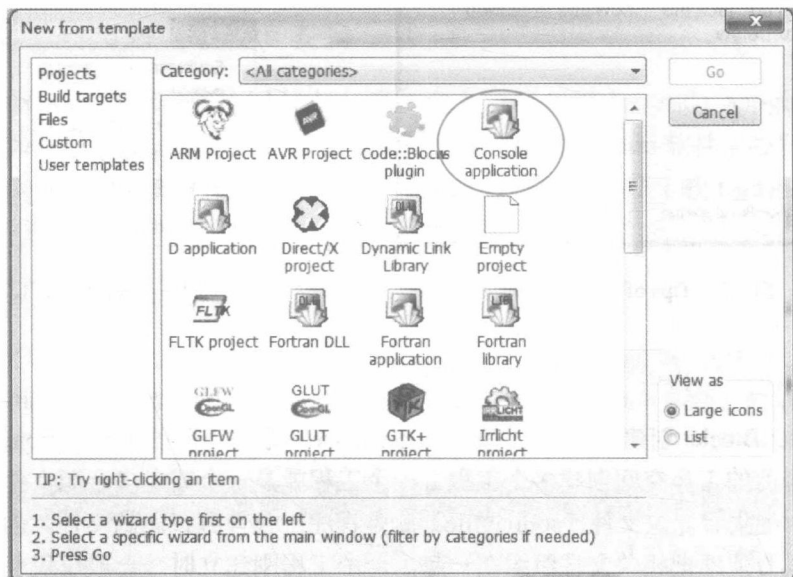


图 1.5 选择工程模式对话框

选择右侧的 Go 按钮，弹出一个控制台对话框，如图 1.6 所示。



图 1.6 控制台对话框

选择 Next 按钮，弹出一个选择语言对话框，如图 1.7 所示。

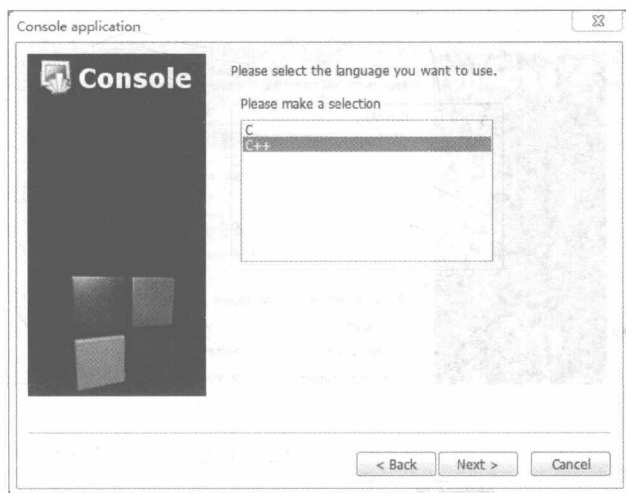


图 1.7 选择语言对话框

在弹出的对话框中有 C 和 C++ 两个选项，选择 C++ 表示编写 C++ 控制台应用程序，选择 C 表示编写 C 控制台应用程序。本书以编写 C++ 程序为例，因此这里选择 C++。接下来单击下方的 Next 按钮，弹出一个工程名称和保存路径对话框，如图 1.8 所示。

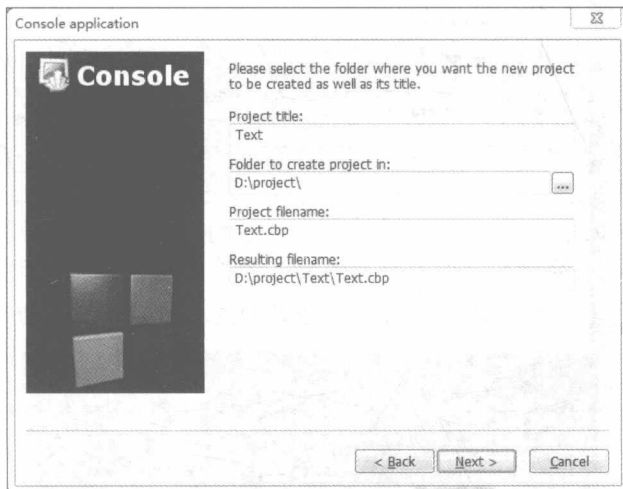


图 1.8 工程名称和保存路径对话框

在弹出的对话框中有 4 个需要填写文字的地方，填上前两个（工程名和工程文件夹路径），后两个位置需要填写的内容可以自动生成。单击 Next 按钮，弹出

一个选择编译程序对话框，如图 1.9 所示。



图 1.9 选择编译程序对话框

编译器选项仍旧选择默认的编译器，剩下的全部打勾。单击 Finish 按钮，则创建了一个名为 Text 的工程。依次展开左侧的 Text、Sources、main.cpp，最后显示文件 main.cpp 的源代码，如图 1.10 所示。

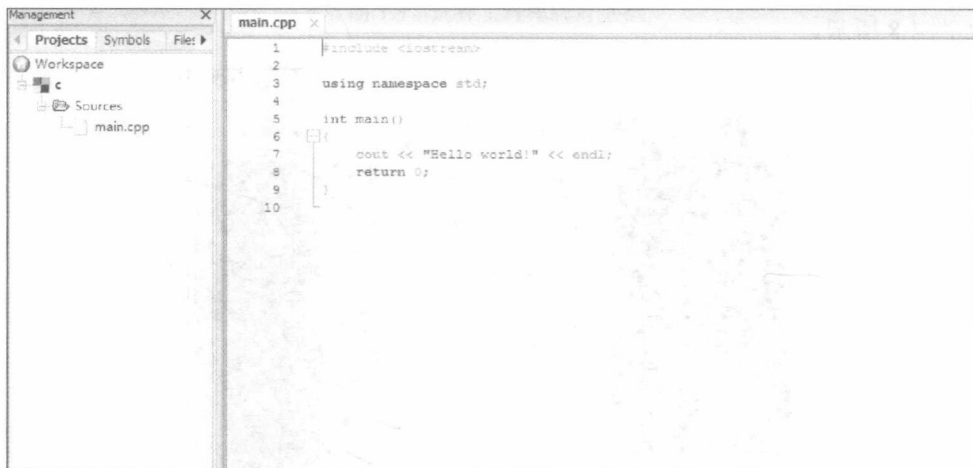


图 1.10 工程主文件

2. 运行程序

创建工程，在工程的主文件中编写程序，需要运行的程序可以直接用快捷键 F9 运行，也可以用鼠标单击 Build and Run 运行程序，如图 1.11 所示。