四川省“十二五”普通高等教育本科规划教材

JISUANJI CHENGXU SHEJI JICHU C/C++

计算机程序设计基础

C/C++

主编 景 红

西南交通大学出版社

四川省“十二五”普通高等教育本科规划教材

计算机程序设计基础

C/C++

主编 景 红

西南交通大学出版社
·成 都·

内容摘要

本书以实际应用为主线，由浅入深地介绍计算机程序设计中的基本概念、基础知识和基本技能，使学生（读者）掌握编程解决问题的一般思路与基本方法。其内容主要包括：软件开发、算法描述和C/C++语言基本语法等基础知识，常用数据类型和经典问题通用算法，面向过程、面向对象和STL程序设计的基本方法和基本调试技能。

本书文字通俗易懂，应用案例丰富，叙述较为系统而全面。本书适合作为高等院校学生学习C/C++语言课程的教材，同时也可作为自学C/C++语言的指导和参考书。

图书在版编目（CIP）数据

计算机程序设计基础C/C++／景红主编. —成都：西南交通大学出版社，2017.3（2017.4重印）
四川省“十二五”普通高等教育本科规划教材
ISBN 978-7-5643-5327-8

Ⅰ. ①计… Ⅱ. ①景… 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（2017）第044777号

四川省“十二五”普通高等教育本科规划教材

计算机程序设计基础C/C++

主编　景　红

责任编辑	李芳芳　宋彦博
封面设计	何东琳设计工作室
出版发行	西南交通大学出版社 （四川省成都市二环路北一段111号 西南交通大学创新大厦21楼）
发行部电话	028-87600564　028-87600533
邮政编码	610031
网址	http://www.xnjdcbs.com
印刷	四川煤田地质制图印刷厂
成品尺寸	185 mm × 260 mm
印张	26.75
字数	693千
版次	2017年3月第1版
印次	2017年4月第2次
书号	ISBN 978-7-5643-5327-8
定价	49.50元

课件咨询电话：028-87600533
图书如有印装质量问题　本社负责退换

前　言

在高校本科教育中，“计算机程序设计”是理工科类专业的一门重要公共基础课程。其教学目标是在课程结束时学生能够：养成利用计算机解决问题的思维方式，具有计算思维素养、创新意识和团结合作的工程职业素质；掌握一门高级程序设计语言的基础知识，具有使用计算机编程解决实际问题的基本能力；为未来在本学科领域使用计算机进行应用研究、技术开发等工作奠定基础。

本书是针对高校计算机程序设计课程编写的教材，并根据使用了多年的教材《计算机程序设计基础（C++）》（景红主编，西南交通大学出版社 2009 年版）的教学效果以及课程教改成果，构建起基于丰富实用的案例和以问题为主线的知识体系。全书针对编程求解过程和学生学习特点，从思路分析、数据结构规划、算法设计、程序实现等方面做了比较细致和全面的讨论，并力求语言简明、概念准确、目标明晰。其目标是使学习者通过学习达到：熟悉 Visual C++程序的开发和调试环境；掌握 C/C++语言的基础知识；掌握面向过程、面向对象和 STL 程序设计的基本方法和基本调试技能；掌握常用数据类型（包括基本数据类型、自定义数据类型），以及一些经典问题的通用算法；能够使用 C++语言编程解决一般性问题。

本书的内容组织充分利用了西南交通大学出版社的数字化教学平台，为学习者提供多维度的学习支持，包括：微视频讲解、编程训练、知识点测试及结果分析等。

参加本书撰写工作的人员都是在高校长期从事计算机教学与科研工作的一线教师，并有着丰富的教学经验，他们是：吴燕（3.4 节）、张旭丽（4.4.1～4.4.3 节）、冯晓红（4.4.4～4.4.8 节）、刘霓（5.5 节）、戴克俭（6.7 节）、李茜（7.6 节）、廖革元（项目实战）、景红（其余各章节和数字化资源）。在本书完稿之际，刘金艳、陈小平、张旭丽和凯定吉等老师对该书内容提出了非常宝贵的修改意见，在此我们要表示深深的感谢。同时，我们要感谢西南交通大学从事计算机基础教学工作的全体教师。

身处教学一线的我们，力图撰写一本更适合培养学生计算机程序设计能力的好教材。但我们也深知，不同的学习者有着不同的个性特征、认知结构、学习动机和学习风格等，因此本教材不一定能满足所有学习者的需要，同时撰写过程中失误在所难免。所以，我们衷心希望广大读者能够不吝赐教。

编　者

2017 年 1 月于西南交通大学

目　录

[基础篇——SP]

[提高篇—OOP]

[提高篇—GP]

[基础篇——SP]

在计算机程序设计中，通常把为得到问题的解而执行的一步一步的操作称为过程，把基于功能分析及每个功能由计算机的一个操作过程实现的程序设计方法称为**面向过程的程序设计**，也称为传统的程序设计。这种程序设计采用结构化程序设计（Structured Programming，SP）模式来完成。

基础篇的预期学习成果是学习者能够很好地理解和掌握“什么时候使用”和“怎样使用”结构化程序设计技术来编程解决实际问题。

而编程学习获得成功的更重要的因素是学习者的学习兴趣和更多的动手实践。

让我们一起从这里开始吧!

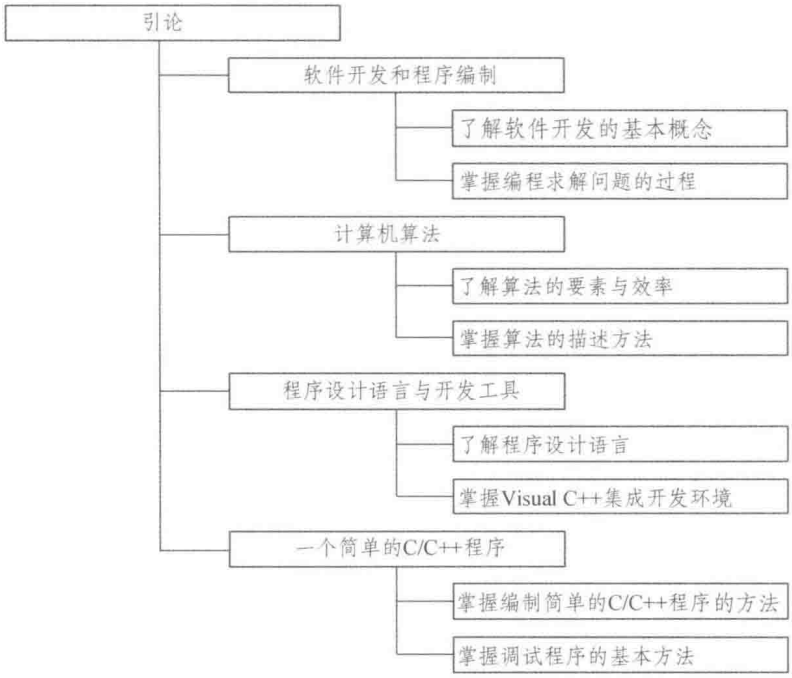
第1章 引 论

微课程

随着计算机技术的迅猛发展，大量应用软件的面世，使得计算机用户在日常应用中遇到的问题多数都可以借助现有的软件来解决，例如 Office、Matlab 等。

但是，在工程领域中，遇到的很多问题往往是繁杂多变的，所以没有什么软件可以包罗万象；而且使用通用软件解决专业特殊问题时，其效率往往会很低，甚至可能无法胜任。在这些情况下，用户按照需求自行开发能够完成特定功能的应用软件，就成为唯一的解决办法。

本章的预期学习成果：



1.1 软件开发和程序编制

微课程

软件是一种无形的逻辑产品，并且只有在解决实际问题的过程中被证实是可行的和被用户所接受时，才能成为产品以及具有存在的价值。而**软件开发方法**是保证软件生产过程和质量的一套规范，它的主要内容体现在：**明确的工作步骤、具体的文档格式和确定的评价标准**等方面。

1.1.1 软件开发过程

通常可以将一个软件从问题的提出到投入使用的过程划分为若干个阶段，每个阶段有相对独立的任务并按照预期进度完成，如图 1.1 所示。这样，也就可以将一个软件开发全过程按照系统化、严格约束和可量化的方式组织起来，既方便对软件开发的组织管理和团队成员的分工协作，又可以极大地提高软件的质量。

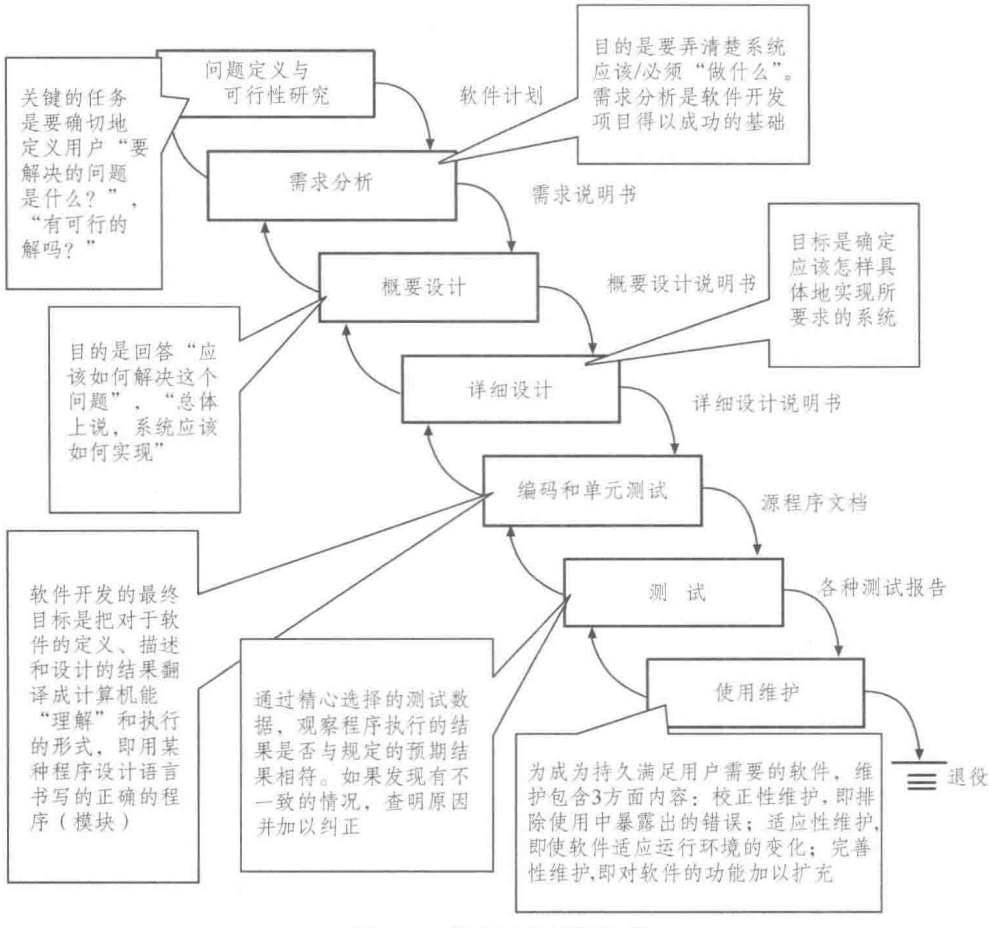


图 1.1 软件开发过程示例

1.1.2 编制程序的基本方法

编制程序是软件开发的核心，也是一个将人类思维转化为计算机思维的过程。传统的程序设

计主要经过以下步骤来完成。

（1）分析问题

所谓分析问题，即通过收集原始资料，取得对要解决的问题的清晰理解，进而确定解决问题的目标以及实现该目标所需要的条件。

（2）规划数据结构与设计算法

简单地说，一个数据结构就是一类数据的表示及其相关操作的集合。规划数据结构就是规划如何表示和存储该问题的数据。而设计如何求解问题的方法与步骤，则是计算机算法要解决的问题。建构合理的数据结构往往可以简化算法，而好的算法设计又可以使程序具有更高的效率。

（3）验证算法

所谓验证算法，即使用多组样本数据，通过手工计算，对方案的正确性进行验证，看看它是否能够按照预想进行工作。当编制的程序不太长时，往往会将这一步省略。

（4）编码实现

所谓编码实现，即选用一种程序设计语言（如 C 语言、C++语言、Java 语言等）将前面的规划和设计转换成计算机能够理解的程序。这时特别需要注意的是：遵循编程规则和规范是程序具备可读性、可靠性和可维护性的基本保证。

（5）测试和调试程序

测试是在计算机上用样本数据运行程序，检查功能和代码的正确性。

调试就是查找和排除程序错误，直到能够得到正确的运行结果为止。通常将程序中的错误称为 Bug，它可能是语法错误，也可能是逻辑错误。借助集成环境的提示信息，大多数语法错误容易被找到和改正；但是要找出逻辑错误则困难得多，因为导致逻辑错误的因素很多。

1.2 计算机算法

微课程

计算机算法规定了利用计算机解决某个（或某类）特定问题的一系列运算（或称操作序列），它是对计算机解题方案的准确与完整的描述。本教材中提到的“算法”，如果没有给出特别说明，均指计算机算法；提到的“程序”，如果没有给出特别说明，均指计算机程序。

1.2.1 算法的表示方法

表示算法的方法有很多，常见的有自然语言、传统流程图、结构化流程图（三种基本结构的流程图、N-S 流程图）、伪代码等。这里简要介绍几种常用方法。

1. 自然语言表示

【案例 1.1】 用自然语言表示求解 n!的算法。

步骤 1：输入 n 值；

步骤 2：将 1 赋值给 t；

步骤 3：将 1 赋值给 i；

步骤 4：计算 t×i，结果赋值给 t；

步骤 5：计算 i+1，结果赋值给 i；

步骤 6：如果 i≤n 成立，则返回到步骤 4，并从步骤 4 继续向下执行；

步骤 7：输出 t（即输出最终结果），计算结束。

很显然，在以上描述中实际隐含着人的阅读习惯，即自然语言描述的是一个从上到下、从左到右的处理流程。

自然语言是人们日常使用的语言，如汉语、英语。使用自然语言描述算法，通俗易懂，但是往往叙述文字冗长，含义不尽严格，容易出现歧义。

2．传统流程图表示

美国国家标准学会 ANSI（American National Standard Institute）规定，用表示特定操作的图符构成的算法图示，称为流程图（也称传统流程图）。流程图已被世界各国程序设计者普遍采用。流程图中的有关符号如图 1.2 所示。

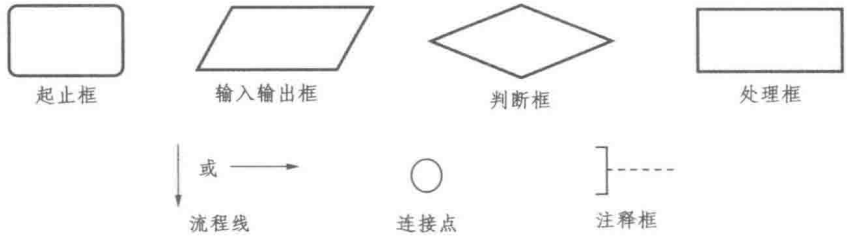


图 1.2　流程图符号

【案例 1.2】　用传统流程图表示求解 n!的算法，如图 1.3 所示。

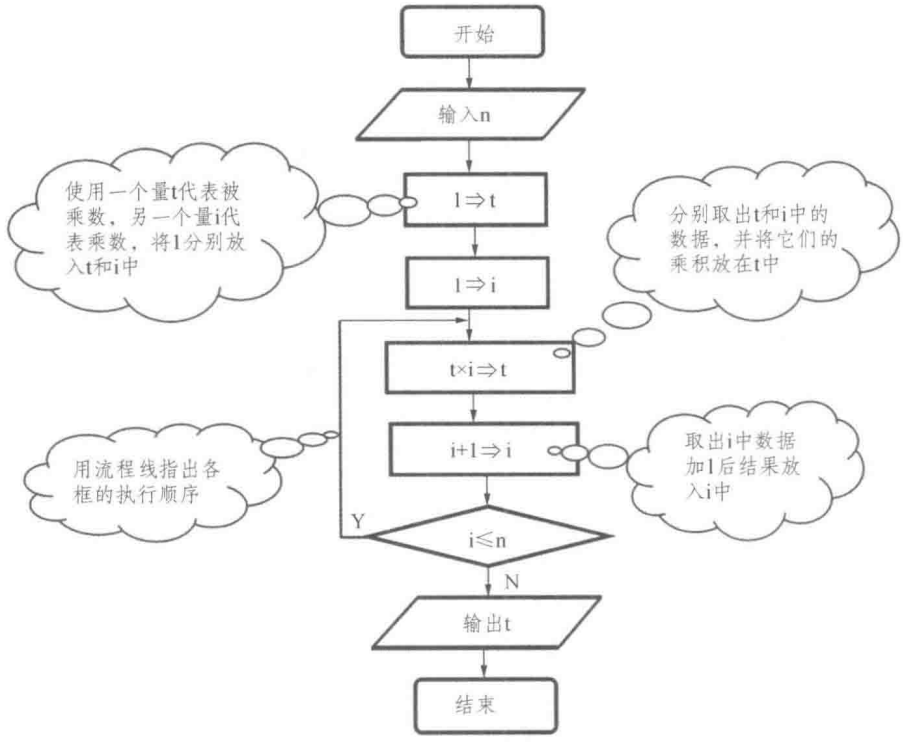


图 1.3　案例 1.2 的算法流程图

3．结构化流程图表示

（1）三种基本结构的流程图

1966 年，学者 Bohra 和 Jacopini 提出了以下述三种基本结构作为表示算法的基本单元。其中的模块 A 或模块 B 代表的是一个或一组操作（或运算）。

① **顺序结构**。顺序结构如图 1.4 所示。在顺序结构中，按照流程线确定的顺序依次执行，即从 a 点进入结构，首先执行 A，然后执行 B，最后从 b 点脱离该结构。

② **选择结构**。选择结构又称选取结构，如图 1.5 所示。从 a 点进入结构，按照流程，首先对给出的条件 P 进行判断，然后根据判断结果的成立与否来确定执行流程：如果 P 成立，则执行 A；如果 P 不成立，则执行 B[参见图 1.5（a）]或不执行任何操作[参见图 1.5（b）]。最后，从 b 点脱离该结构。

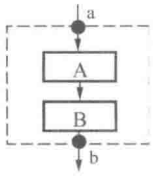


图 1.4　顺序结构

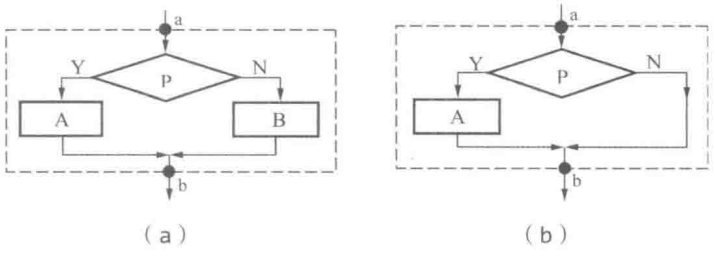


图 1.5　选择结构

分析选择结构不难发现：结构中无论条件 P 是否成立，只能执行 A 或 B 中之一，一定不可能既执行 A 又执行 B。

③ **循环结构**。循环结构又称重复结构，如图 1.6 所示。循环即重复地做。循环结构通常可以分为当型和直到型两类。

如图 1.6（a）所示，在当型循环结构中，从 a 点进入，重复执行“对条件 P1 进行判断，如果 P1 成立，则执行 A”，而当条件 P1 不成立时，循环结束，从 b 点脱离该结构。

如图 1.6（b）所示，在直到型循环结构中，从 a 点进入，首先执行 A，然后重复执行“对条件 P2 进行判断，如果 P2 不成立，则执行 A”，直到条件 P2 成立时，循环结束，从 b 点脱离该结构。

分析循环结构不难发现：① 结构内一定不存在“死循环（即没有止境的循环）”；② 两种类型的循环在使用时是可以互换的，也就是说，凡是可以使用当型循环处理的问题，也可以使用直到型循环解决，反之亦然。

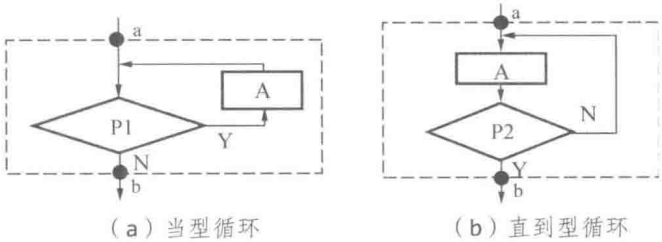


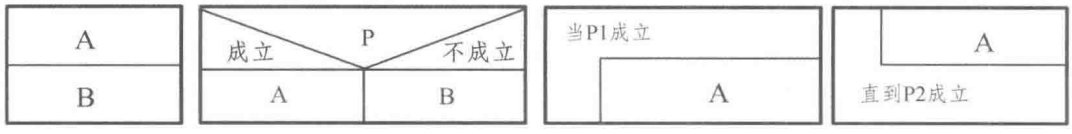
图 1.6　循环结构

综上所述，三种基本结构的共同点是：只有一个入口（如 a 点），只有一个出口（如 b 点），

结构内的每一部分都有机会被执行。一个算法无论多么复杂，终究可以分解为由顺序、选择和循环三种基本结构组合而成。或者说，可以组合应用这三种基本结构来解决任何复杂问题。而由这三种基本结构所构成的算法称为结构化算法。

（2）N-S 流程图表示

1971 年，国外学者 I. Nassi 和 B. Shneiderman 提出了一种新的流程图形式，称为 N-S 流程图，它在三种基本结构流程图的基础上完全去掉了流程线，如图 1.7 所示。



（a）顺序结构　（b）选择结构　（c）当型循环　（d）直到型循环

图 1.7　N-S 流程图

【案例 1.3】　用 N-S 流程图表示求解 n!的算法，如图 1.8 所示。

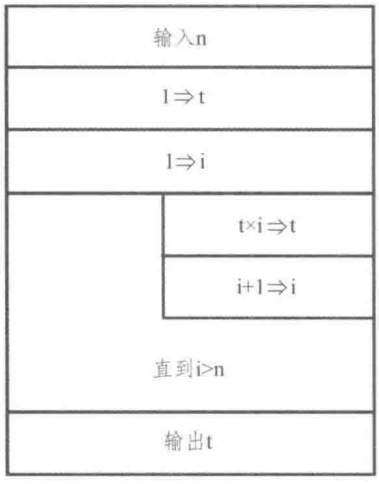


图 1.8　案例 1.3 的算法流程图

1.2.2　算法要素与效率

1. 算法的特征

对同一个（或同一类）问题可以设计不同的解决方案。但是，就算法本身而言，它应具有以下主要特征：

① 有效性：算法中的每一步骤都能够被计算机所理解和执行，而不是抽象和模糊的概念。

② 有穷性：无论算法有多么复杂，它都必须在有限步骤之后结束运行，而不能是无限的。

③ 确定性：算法的每一步骤都要有确定的执行顺序，而不能有任何歧义。

④ 有零个或多个输入：部分数据在操作之前需要通过外界赋值，称为算法的输入，它是算法加工的对象。一个算法可以没有输入，也可以有多个输入。

⑤ 有一个或多个输出：算法执行过程中对外界产生的任何影响（算法的运算结果）称为算法的输出。一个算法至少有一个输出，也可以有多个输出。

2．算法的性能

要设计一个“好”的算法，不仅需要保证算法正确，还需要考虑算法的性能。通常需要考虑以下主要因素：

① 正确性：对于任意的一组输入，包括合理的输入和不合理的输入，总能得到预期的输出。

② 可读性：一个清晰易读的算法有助于对算法的理解，易于调试和修改。

③ 健壮性：对于非法的输入，能够做出适当的反应或相应的处理，而不会产生一些“莫名其妙”的结果，更不会引起灾难性的后果。

④ 高效性：算法占用的计算机资源（主要是运行时间和存储空间）越少，效率越高。一般而言，算法的效率与算法占用的计算机资源量成反比。

3．算法的评价

面对同一个（或同一类型）问题的不同解决方案，通常又应该如何抉择呢？

【案例 1.4】 在一组升序排列的 n 个数据中，查找是否存在某指定数据。

算法 1：设计这个问题的解决方案，很自然而然采用的方法是顺序查找，即从第一个数据（也称为元素）开始逐一比较。此时，最好的情况是只需要比较一次，即第一个数据就是要查找的数据；最坏的情况是需要比较 n 次，即最后一个数据是要查找的数据，或者最后一个数据也不是要查找的数据，即指定数据不存在。假设每个元素与要查找数据相同是等概率的，则平均需要比较 n/2 次。

算法 2：采用折半查找的方法（也称二分查找），即先和居于中间位置的元素比较。如果比较结果为相等，则查找成功；如果比较结果为中间位置的元素大于要查找的数，则继续在前半部分查找，否则就继续在后半部分查找。该算法最多只需要比较 $\log_2 n$ 次。

这个案例表明：通常求解一个问题可能会有多种算法可供选择，选择的主要标准是算法所需要的存储空间少和执行更快。只有在对这些可行的算法进行分析以后，才能知道哪一个算法效率更高。

通常把对算法效率的度量，称为算法的复杂性分析。复杂性是算法运行所需要的计算机资源的量，是依赖于算法要解决的问题的规模、算法的输入和算法本身的函数（即一段程序代码）。其中，需要的时间资源的量称作时间复杂性，需要的存储空间资源的量称作空间复杂性。而对于一个给定的算法，怎样计量它的复杂性，可参见其他相关书籍，这里不再赘述。

随着计算机运算速度和存储容量的直线增长，有人认为低效的算法可以由高速的计算机来弥补，这种观点实际上是不正确的。因为，随着经济的发展、社会的进步、科学研究的深入，要求计算机解决的问题越来越复杂、规模越来越大，其超线性增长导致的时耗的增长和空间需求的增长，决非计算机速度和容量的线性增长带来的时耗的减少和存储空间的扩大所能抵消的。因此，利用计算机解决实际问题时，应着眼于寻求更高效的算法。

1.3 程序设计语言和开发工具

程序设计语言，俗称编程语言或高级语言。人们利用程序设计语言能够准确地向计算机发出指令。而且为了方便用户的使用，每种编程语言都提供了服务于程序开发的集成工具。例如，本教材使用的开发工具 Visual Studio 2010 是由 Microsoft 公司推出的集成开发环境。

1.3.1 概 述

在程序设计中，每种高级语言都有自己的特点和适用范围。例如，BASIC 是一种很适合初学者学习的编程语言；Fortran 是一种广泛应用于工程领域科学计算的编程语言；C 是一种结构化程序设计语言，更多地用于系统的底层程序设计；C++是在 C 的基础上发展起来的面向对象程序设计语言，更适合于高并发和实时处理等领域；Java 是一种网络编程语言，更多地用于企业应用级软件开发；C#作为微软.NET 架构的首选语言，可以用来开发桌面应用程序和智能手机应用程序；Python 是一种解释型的面向对象编程语言，更适用于快速应用程序开发等。

目前，由 Microsoft 公司推出的流行开发产品有 Visual C++ 2010 及更高版本。Visual C++ 2010 是 Visual Studio 2010 的一部分，并且更适用于开发面向 Windows 7 及以上的应用程序。而 Visual Studio 是一套功能完备的集成开发环境，它集编辑、编译、调试、运行等于一体，支持 Visual C、Visual Basic、Visual C#等多种语言。利用它可以为 Microsoft 平台创建多类应用程序，如控制台应用程序（也称为 Console 应用程序，在命令行方式下运行）、Windows 应用程序（实现 Windows 窗体形式的操作界面）、ASP.NET Web 应用程序（实现 B/S 软件架构）等。

1.3.2 Visual C++集成开发环境及其使用

Visual C++ 2010 的集成开发环境（IDE）为项目管理与配置、源代码编辑与调试等提供了强大的支持，是 C/C++程序开发过程中不可缺少的工具。那么，如何在 Visual C++ 2010 环境中编辑和运行 C++程序呢？

1.创建项目

首先需要启动 Visual Studio 2010，进入“新建项目”窗口。具体方法有以下两种：一种是在“起始页”面上单击“新建项目...”;另一种是在“文件”菜单上连续选择“新建”→“项目...”。接下来，在“新建项目”窗口中（参见图 1.9），单击左部“项目类型”区中的“Visual C++”，然

图 1.9 在已安装的模板中选择

后单击中部“模板类型”区中的“win32 控制台应用程序”，并在下部“名称”栏输入项目名称（默认情况下解决方案名称与项目名称相同），以及在“位置”栏指定存储位置（或采用默认位置，或者借助“浏览...”按钮指定位置），然后单击“确定”按钮。最后，在新弹出的“win32 应用程序向导”窗口中，单击“下一步”按钮之后，“附加选项”选择“空项目”，并单击“完成”。

2．向当前项目中添加程序文件

参见图 1.10，如果未显示“解决方案资源管理器”，可以在“视图”菜单中选择它。

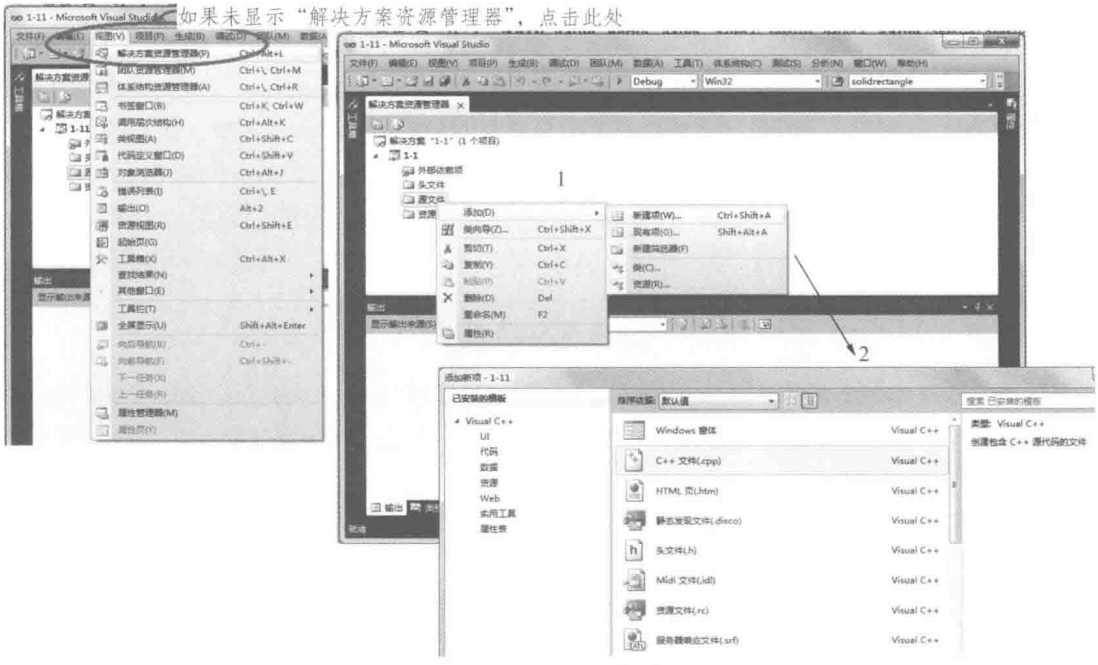


图 1.10　添加源文件和头文件

（1）添加源文件

在“解决方案资源管理器”中，右击“源文件”文件夹，连续选择“添加”→“新建项...”，在弹出的“添加新项”窗口中，单击“C++文件（.cpp）”并输入文件名，然后单击“添加”按钮。然后，在该文件的编辑窗口中输入程序代码即可。

（2）添加头文件

在“解决方案资源管理器”中，右击“头文件”文件夹，连续选择“添加”→“新建项...”，在弹出的“添加新项”窗口中，单击“头文件（.h）”并输入文件名，然后单击“添加”按钮。然后，在该文件的编辑窗口中输入程序代码即可。

（3）编辑、保存和运行程序

参见图 1-11(a)和(b)，单击“保存”按钮将文件保存到指定目录下。执行程序的方法如下：

方法 1：参见图 1-11(a)，单击菜单栏“调试”→“开始执行”，执行程序。

方法 2：参见图 1-11(b)，单击▶（“启动调试”按钮），调试和执行程序。

说明：

◆　在 Visual C++ 2010 环境中，使用方法 2（单击▶）调试和运行程序时，运行结果常常是一闪而过，然后就返回平台的编辑窗口，而无法看清程序输出结果。对此，解决的方法有两种：