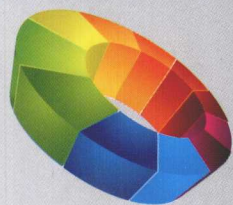


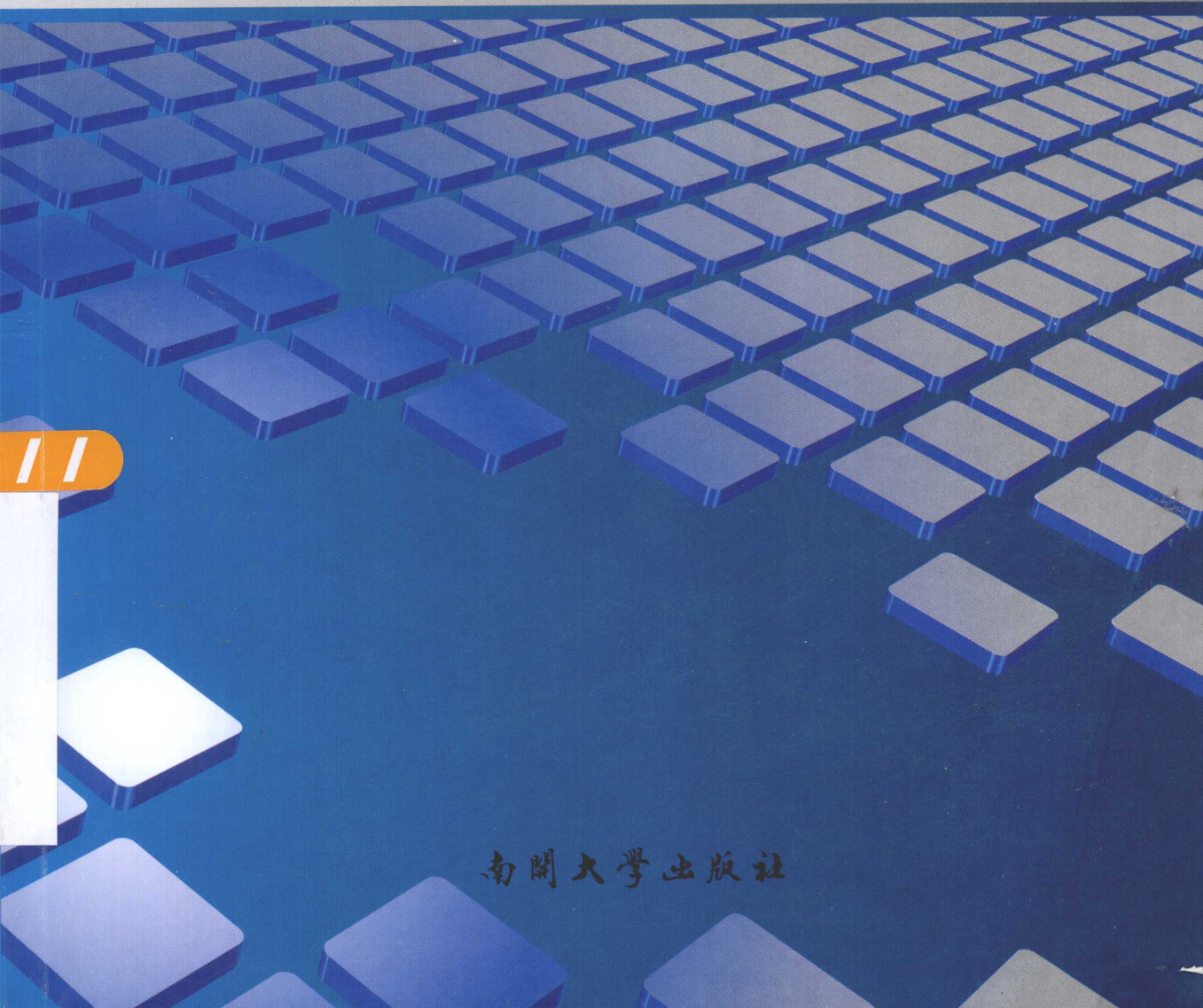
高等院校非计算机专业教材

赵宏 主编

李敏 王恺 王刚 编著



C++程序设计语言



南开大学出版社

· 013049457

TP312C-43
811

高等院校非计算机专业教材

C++程序设计语言

赵宏 主编

李敏 王恺 王刚 编著



北航

C1658419

南开大学出版社
天津

TP312C-43
811

788030810

图书在版编目(CIP)数据

C++程序设计语言/ 赵宏主编. —天津: 南开大学出版社, 2012. 9

高等院校非计算机专业教材

ISBN 978-7-310-04016-2

I. ①C… II. ①赵… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 214150 号

版权所有 侵权必究

南开大学出版社出版发行

出版人: 孙克强

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022)23508339 23500755

营销部传真: (022)23508542 邮购部电话: (022)23502200

*

天津市蓟县宏图印务有限公司印刷

全国各地新华书店经销

*

2012 年 9 月第 1 版 2012 年 9 月第 1 次印刷

260×185 毫米 16 开本 22.625 印张 573 千字

定价: 39.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022)23507125

内容提要

高级程序设计语言 C++ 是在 C 语言基础上拓展而来的一种能够进行面向对象程序设计和传统过程化程序设计的语言，是 C 语言的超集。本书通过大量的程序实例，较详细地介绍了 C++ 语言的基础知识，在初学者容易出现错误和困惑的地方，有针对性地提供了大量的附注，帮助读者更好地理解 C++ 的基本概念和技术。

全书共分 19 章，分别介绍了程序设计的基本概念、C++ 程序的基本组成，数据类型、常量和变量，运算符、表达式和语句，程序控制结构，函数初步变量的存储类型，数组，指针和引用，字符串，函数，构造数据类型，编译预处理，类与对象，继承，多态性，运算符重载，输入/输出流，文件，模板，MFC 入门等内容。

为提高学习效率，另有与本书配套出版的《C++ 程序设计语言上机实习指导与习题集》，精心为各章选编了配套的上机实习内容，并在思想方法、算法和语法上给出了相应的指导，最后还选编了配套的经典习题。

本套教材是专门为高等院校非计算机专业 C++ 语言程序设计课程编写的教学用书，面向 C++ 初学者，不要求读者已经熟悉相关的编程概念和有 C 语言方面的背景知识。本套教材也适合自学者使用。

前 言

2006年3月,美国科学基金会计算机与信息科学工程部主任周以真(Jeannette M. Wing)教授首先提出并定义了“计算思维”这一概念:“计算思维是运用计算机科学的基础概念进行问题求解、系统设计,以及人类行为理解等涵盖计算机科学之广度的一系列思维活动。”计算思维吸取了问题求解中所采用的一般数学思维方法,现实世界中巨大、复杂系统的设计与评估的一般工程思维方法,以及复杂性、智能、心理、人类行为的理解等的一般科学思维方法。

在我国,计算思维的重要性也已引起了科学家和教育界的高度重视。教育部高等学校计算机基础课程教学指导委员会主任委员陈国良院士积极地倡导把培养学生的“计算思维”能力作为计算机基础教学的核心任务,并由此建设更加完备的计算机基础课程体系和教学内容。

程序设计能力是计算思维能力的重要组成部分。目前,不仅许多计算机专业和多数软件学院的程序设计课程选择了C++作为程序设计的第一门语言,而且越来越多的理工科专业也把C++作为计算机基础课,这一方面是由于C++是应用最广的面向对象语言,另一方面是由于它有利于初学程序设计的学生学习一般的编程技巧。南开大学理工科公共计算机程序设计课程选用的就是C++语言,该课程的目标是使学生掌握一门高级程序设计语言并且具备基本的程序设计能力。

近几年的教学经验表明,由于公共计算机基础课课时有限,把C++语言作为高级语言程序设计的教学语言对于教师和学生都是有难度的,主要的问题是C++的规模和C++程序的复杂程度,往往使刚刚步入大学的学生感到困难重重。教材的编写一直是一个艰巨而具有探索性的工作。面对非计算机专业的理工科学生,如何使初学者在有限的课时内打下良好的程序设计基础,目前仍有许多值得探索的地方。

国内外同类教材主要是面向计算机专业的学生,即使是为非计算机专业的理工科学生编写的教材,也仅仅是对专业教材的简单取舍,较少考虑非计算机专业的理工科学生学习程序设计的需求和特点,在内容和程序实例的选取上,文字叙述存在不足,在教材的编写风格上也过于传统,较难激发学生的学习兴趣。

本书的编著者力求做到:

(1) 适应21世纪课程体系和教学内容改革方向要求,以培养学生的“计算思维”能力为核心目标,抓住授课对象是非计算机专业的本科一年级学生的特点,注重内容的选取和章节的安排,力求做到:学习内容循序渐进;文字叙述简单、易于理解;全书难易得当、重点突出,以适合非计算机专业学生和C++语言自学者学习程序设计语言时使用。

(2) 注重语法规则的清晰讲解和配备完整的程序实例。加强程序实例的选择和比例配置,训练和培养学生分析、解决问题的思想及能力。

(3) 根据初学者容易出错的地方,给出了大量的提示、提问和学习指导,适合课堂教学

和自学。

(4) 注重理论与实际相结合, 同步为本书编写了配套的实习指导和习题集, 根据各章节的内容给出相应的上机实习内容, 强化并丰富相应的习题, 以逐步提高学生的程序设计能力, 使他们能够使用 C++ 高级语言解决实际的问题。

本书由担任南开大学信息学院公共计算机基础教学部高级语言程序设计课程的任课老师, 结合多年的教学经验, 根据我国高校的非计算机专业理工科学生学习程序设计课时少、时间短的特点编写。赵宏负责第 1 章至第 5 章的编写并统编全书, 第 6 章至第 9 章和第 19 章由王恺编写, 第 10 章、第 11 章和第 15 章至第 18 章由李敏编写, 第 12 章至第 14 章由王刚编写。本书的编写还得到了南开大学出版社张燕老师的大力支持。

本书的编著者参考了国内外许多 C++ 程序设计语言方面的书籍, 力求有所突破和创新。但由于能力和水平的限制, 书中难免有不妥乃至错误之处, 请阅读本书的老师和同学指正。

编 者

2012 年 4 月于南开园

目 录

第 1 章 初识 C++	1
1.1 程序设计语言	1
1.2 第一个 C++ 程序	2
1.3 程序设计的基本概念	11
1.4 小结	14
1.5 学习指导	14
第 2 章 数据类型、常量和变量	17
2.1 数据在计算机中存储的基本概念	17
2.2 基本数据类型	23
2.3 常量	24
2.4 变量	27
2.5 小结	29
2.6 学习指导	29
第 3 章 运算符、表达式和语句	31
3.1 运算符和表达式	31
3.2 类型转换	39
3.3 语句	40
3.4 小结	43
3.5 学习指导	43
第 4 章 程序控制结构	44
4.1 顺序结构	44
4.2 选择结构	44
4.3 循环结构	50
4.4 转向语句	55
4.5 应用实例	57
4.6 小结	60
4.7 学习指导	61

第 5 章 函数初步与变量的存储类型	62
5.1 函数的基本概念	62
5.2 函数的定义	62
5.3 函数的传值调用	64
5.4 函数声明	66
5.5 变量的存储类型、作用域和生存期	67
5.6 小结	70
5.7 学习指导：学习使用程序调试工具——Debug	71
第 6 章 数 组	72
6.1 数组的概念	72
6.2 一维数组	73
6.3 二维数组	78
6.4 小结	81
6.5 学习指导	81
第 7 章 指针和引用	82
7.1 指针的概念	82
7.2 指针变量的定义、初始化和访问	83
7.3 指针的运算	85
7.4 指针与数组	89
7.5 指向指针的指针	92
7.6 const 指针	93
7.7 堆内存分配	94
7.8 引用	97
7.9 小结	99
7.10 学习指导	99
第 8 章 字 符 串	101
8.1 字符串的表示方法	101
8.2 字符串的输入输出	102
8.3 指针与字符串	105
8.4 常用的字符串函数	108
8.5 小结	113
8.6 学习指导	113

第 9 章 函 数	115
9.1 函数的调用机制	115
9.2 函数的递归调用	116
9.3 带默认形参值的函数	119
9.4 内联函数	121
9.5 函数重载	122
9.6 函数指针	125
9.7 函数与指针	126
9.8 函数与引用	132
9.9 小结	135
9.10 学习指导	136
第 10 章 构造数据类型	137
10.1 结构体	137
10.2 枚举	144
10.3 类型重定义 typedef	146
10.4 小结	147
10.5 学习指导	147
第 11 章 编译预处理	149
11.1 宏定义	149
11.2 条件编译	152
11.3 文件包含和多文件结构	154
11.4 小结	159
11.5 学习指导	159
第 12 章 类与对象	161
12.1 类和对象的基本概念	161
12.2 类的定义	162
12.3 对象	164
12.4 构造函数和析构函数	169
12.5 拷贝构造函数	174
12.6 类的静态成员	179
12.7 友元	183
12.8 this 指针	186
12.9 string 类	188
12.10 应用实例	193
12.11 小结	210
12.12 学习指导	211

第 13 章 继 承	212
13.1 什么是继承	212
13.2 派生类的定义	213
13.3 函数重定义	215
13.4 保护 (protected) 成员	216
13.5 派生类的继承方式	217
13.6 派生类的构造函数与析构函数	220
13.7 类型兼容	224
13.8 多重继承	226
13.9 应用实例	231
13.10 小结	232
13.11 学习指导	233
第 14 章 多 态 性	234
14.1 多态性的概念	234
14.2 虚函数	235
14.3 抽象类	242
14.4 应用实例	244
14.5 小结	259
14.6 学习指导	260
第 15 章 运算符重载	261
15.1 运算符重载的概念	261
15.2 运算符重载的方法	263
15.3 运算符重载的规则	272
15.4 特殊运算符的重载	274
15.5 类类型转换	280
15.6 小结	283
15.7 学习指导	283
第 16 章 输入/输出流	284
16.1 输入/输出流简介	284
16.2 输入/输出流的成员函数	286
16.3 输入/输出的格式控制	292
16.4 自定义数据类型的输入/输出	298
16.5 小结	302
16.6 学习指导	302

第 17 章 文 件	303
17.1 文件与流	303
17.2 文件的打开与关闭	305
17.3 文本文件的顺序读写	306
17.4 二进制文件的顺序读写	309
17.5 文件的随机读写	313
17.6 小结	319
17.7 学习指导	320
第 18 章 模 板	321
18.1 函数模板	321
18.2 类模板	328
18.3 小结	333
18.4 学习指导	333
第 19 章 MFC 入门	334
19.1 认识 MFC	334
19.2 一个简单的 MFC 应用程序	335
19.3 常用控件	340
19.4 MFC 应用程序开发实例	341
19.5 小结	348
19.6 学习指导	349
参考文献	350

第1章 初识 C++



导 读

要使用计算机完成解决问题的的工作，就需要学会一种程序设计语言，并掌握程序设计的基本方法。本章将简单介绍高级程序设计语言 C++，并通过对第一个 C++ 程序的学习，使读者对 C++ 源程序的组成和组成元素，集成开发环境 Visual C++ 2005 的编辑、编译、连接和运行步骤等有一个初步的了解，并能够编写具有简单输入输出功能的 C++ 程序。最后还介绍了程序设计的基本概念、步骤和方法，

本章难度指数★，教师授课 2 课时，学生上机练习 2 课时。

1.1 程序设计语言

计算机指令就是由计算机设计者设计的指挥计算机工作的指示和命令，计算机指令系统是计算机所有指令的集合。计算机唯一可以读懂的“语言”就是计算机的指令，叫做**机器语言**，也被称为**低级语言**。程序就是按一定顺序排列的指令集合，执行计算机指令的过程就是计算机的工作过程。程序员把要计算机完成的任务以指令序列的形式写出来，就是机器语言程序设计。然而，直接使用机器语言来编写程序很困难。

为解决使用机器语言编写程序困难的问题，高级（程序设计）语言逐渐发展起来了，例如本书将要介绍的 C++。**高级语言**的语法符合人类自然语言的习惯，便于普通用户编程。但是计算机无法直接运行用高级语言编写的程序，因此需要为高级语言编写相应的翻译程序，这些翻译程序被称为**编译程序、编译系统或编译器**，它们的功能就是把用高级语言写出的程序翻译为计算机能够识别和运行的机器语言程序。

有了高级程序设计语言及其编译系统的帮助，人们就可以集中精力编制出规模更大、结构更复杂的程序。

1.1.1 选择 C++ 程序设计语言的原因

计算机程序设计课程需要选择教学语言。选择 C++ 作为程序设计课程语言的理由：

- 面向对象程序设计技术已逐渐成为主流设计技术，无论是在开发领域还是在教学领域，以结构化程序设计为主要特征的 C 和 Pascal 已经不适应形势发展的需要。
- 面向对象程序设计技术并不取代结构化程序设计和一般程序设计的技巧，C++ 对于结构化程序设计和一般程序设计有较大的兼容性。
- 由于世界各大公司的竞相开发，C++ 语言在各种不同计算机机型上都有优秀的编译系统和相应的开发环境。

● C++是主流的软件开发语言，为广大程序员和软件开发商所接受，并且新的软件开发语言 Java、C#与 C++语言基本一致。

● 国内外出现了一批以 C++语言为主讲授程序设计技术的教科书，还有一些以 C++为主要描述语言的计算机专业的相关教科书，如数据结构、算法设计与分析等。

1.1.2 C、C++和 Visual C++

● C 语言

C 语言是一种高级语言，诞生于 20 世纪 70 年代早期。Bell 实验室为开发 UNIX 操作系统，在旧语言的基础上，开发了能将低级语言的效率、硬件访问能力和高级语言的通用性、可移植性融合在一起的 C 语言。C 语言直至今天仍然被广泛应用，C++、Java 和 C#语言沿用了 C 的大多数语法。

● C++语言

C++语言也是一种高级语言，在 20 世纪 80 年代同样诞生于 Bell 实验室。C++语言是在 C 语言的基础上，引入面向对象的特征，所开发出的一种过程性与对象性相结合的程序设计语言，最初称为“带类的 C”，1983 年取名为 C++。1998 年国际标准化组织和美国国家标准局制定了 C++标准，称为 ISO/ANSI C++，也就是平时所称的标准 C++。标准 C++及其标准库进一步体现了 C++语言设计的初衷。

C++语言开发的宗旨是使面向对象技术和数据抽象成为软件开发者的一种真正实用技术。经过许多次的改进、完善，目前的 C++具有两方面的特点：第一，C++是 C 语言的超集，与 C 语言兼容，这使得许多 C 代码不经修改就可以经 C++编译器进行编译；第二，C++支持面向对象程序设计，被称为真正意义上的面向对象程序设计语言。

● C++开发工具——Visual C++

Visual C++是一款由微软公司开发的软件，是专门负责开发 C++软件的工具，称为集成开发环境（Integrated Development Environment, IDE）。集成开发环境包括编写和修改源代码的文本编辑器、C++编译器、连接器、程序调试和运行以及其他程序开发的辅助功能和工具。通过这个工具，可以大大地提高程序员开发程序的效率。本书采用 Visual C++ 2005 作为开发工具。

1.2 第一个 C++程序

本节通过讲解和分析一个简单的 C++程序，让读者对 C++源程序的组成有一个初步的认识，了解在 Visual C++ 2005 集成开发环境中进行程序编辑、编译、连接和运行的基本步骤，并能够立即上手编写一些非常简单的 C++程序。

【例 1-1】第一个 C++程序。程序的功能是：若用户在键盘上输入问候语“大家好！”，程序会将用户输入的话输出到屏幕上。

【解】完整的程序代码如下：

```
/*  
这是我的第一个 C++程序——p1.cpp  
设计日期：2012 年 4 月  
*/  
  
#include<iostream>                                // 预处理命令
```

```
using namespace std;           // 命名空间
int main()                     // 主函数
{
    char str[10];              // 变量
    cout<<"请输入你的问候: "; // 输出
    cin>>str;                  // 输入
    cout<<"你的问候是: ";      // 输出
    cout<<str<<endl;          // 输出
    return 0;                  // 函数返回
}
```

程序运行时,首先在屏幕上提示一句话“请输入你的问候:”;然后用户在键盘上输入“大家好!”后,按回车键;最后程序将“大家好!”再输出到屏幕上。

1.2.1 C++源程序的组成

一个 C++程序一般由预处理命令、命名空间、函数、语句、变量、输入/输出和注释等几部分组成。下面通过分析例 1-1 程序的语句含义来了解 C++程序的组成。

● 预处理命令

“#include<iostream>”是预处理命令,它使程序具有输入/输出功能。C++的预处理命令包括:宏定义命令、文件包含命令和条件编译命令,都是以“#”开始,第 11 章将详细介绍。

● 函数和函数返回

“int main()”是程序的主函数。一个 C++程序一般由多个函数组成。这些函数可以是用户根据需要编写的自定义函数,也可以是直接调用的由系统提供的标准库函数。函数体用一对花括号“{”和“}”括起来。**任何一个程序必须有且仅有一个主函数 main(),程序从主函数开始执行。**本程序只有一个函数——主函数 main()。

程序中的“return 0;”是函数返回语句,表示结束函数的调用。对于有函数值的函数,必须使用返回语句,以便将值返回给调用函数。

第 5 章和第 9 章将详细介绍 C++函数。

● 命名空间

“using namespace std;”是 using 编译指令,表示使用命名空间 std。命名空间是为了解决 C++中的变量、函数的命名冲突的问题而设置的。解决的办法就是将变量定义在一个不同名字的命名空间中。就好像张三有一台电脑,李四也有一台同样的电脑,为了能够清楚地区分这两台电脑,就要让它们分属于不同的人。上面例子中的 main()函数需要能够访问位于命名空间 std 的“cin”、“cout”和“endl”定义。

“using namespace std;”也可以放在 main()函数里面。在多函数程序中,为了使多个函数都能够访问命名空间 std,只需将“using namespace std;”放在所有函数定义之前。本书采用的就是这种方法。

如果不使用“using namespace std;”,在需要使用命名空间 std 中的元素时,可以使用前缀“std::”,例如:

```
std::cout<<str<<std::endl;
```

● 语句

语句是函数的基本组成单元。C++语句分为简单语句和复合语句，可以组成或应用于顺序结构、选择结构和循环结构等三种结构。任何一条简单语句都必须以分号“;”结束。复合语句是用花括号“{”和“}”括起来的多条简单语句。本程序中包含的都是简单语句。第3章和第4章将分别介绍C++的语句和三种程序控制结构。

● 变量

程序中的“char str[10];”是变量定义语句。char表示变量的数据类型是字符型，str是数组名，str[10]表示str是有10个字符型变量的数组。

C++程序需要将数据放在内存单元中，变量名就是内存单元中数据的标识符，通过变量名来存储和访问相应的数据。第2章将详细介绍数据类型和变量的内容。

● 输入/输出

C++程序中的输入语句用来接受用户的输入，输出语句用来向用户显示程序的运行结果。

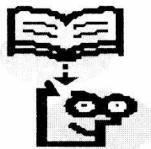
程序中的“cout<<“请输入你的问候:”;”是输出语句，用cout（读C-Out）将用两端加上双引号表示的一个字符串“请输入你的问候:”输出到屏幕上。语句的中“<<”叫**插入运算符**，不能省略。输出语句“cout<<str<<endl;”的功能是将数组str中的字符串输出到屏幕上，然后再在屏幕上输出一个换行，即endl。

程序中的“cin>>str;”是输入语句，用cin（读C-In）将用户通过键盘输入的字符串放在数组str中。语句的中“>>”叫**提取运算符**，也不能省略。

关于cin和cout及其使用，将在第16章详细介绍C++的输入/输出时一并讲解。

● 注释

程序中用到了注释。注释的作用就是帮助程序员阅读源程序。编译器在编译程序时将忽略注释。注释有两种方式：“//”是单行注释符号，其后面直到一行结束的内容都为注释内容；使用“/*”和“*/”，则其中的内容都为注释内容。例1-1中采用了两种注释方式。



提示:

- 源程序中表示函数体的一对花括号“{”、“}”不能缺省，且须成对出现。
- 不能去掉“#include<iostream>”和“using namespace std;”，否则程序无法使用cout和cin等输入/输出操作。
- 字符串是一个整体，两端的双引号不能缺省。
- 每一条语句后面的分号不能缺省。
- 程序中的尖括号“<”、“>”，双引号和分号必须是半角符号。

1.2.2 C++源程序的组成元素

1. C++的字符集

字符集是组成语言词法的一套符号。程序员编写的源程序不能使用字符集以外的字符，否则编译系统无法识别。C++语言的字符集包括：

- 大写、小写英文字母，共52个；
- 阿拉伯数字0~9，共10个；
- 运算符、标点符号及其他字符，共30个；

+ - * / % = ! & | ~ ^ < >
 ; : ? , . ' " \
 () [] { } # _ 空格

2. 标识符

标识符是指由程序员定义的词法符号,用来给变量、函数、数组、类、对象等实体命名。定义标识符应该遵守以下规则:

- 标识符只能使用大小写字母、数字和下划线“_”。
- 标识符的首字符必须是字母或下划线。
- 标识符中不能含有空格、标点符号和其他字符。例如,下列标识符是合法的:

abc、_data、x1、ch2_1、CC、student1

下列标识符是不合法的:

5student、1000%、student3-1、page 5、r=d

- C++标识符区分大写字母和小写字母。例如, SUM、sum、Sum 等是不同的标识符。
- C++对标识符的长度没有限制,但 ANSI C99 标准中只保证名称中的前 63 个字符有意义,不同的编译系统对标识符的长度要求也不同。
- 标识符不能与关键字同名。

3. 关键字

关键字又称**保留字**,是组成编程语言词汇表的标识符,用户不能再用它们标识其他实体。表 1-1 是 C++的关键字(按字母顺序排列)。

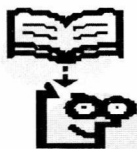
表 1-1 C++的关键字

asm	auto	bool	break	case	catch
char	class	const	const_case	continue	default
delete	do	double	dynamic_case	else	enum
explicit	export	extern	false	float	for
friend	goto	if	inline	int	long
mutable	namespace	new	operator	private	protected
public	register	reinterpret_cast	return	short	signed
sizeof	static	static_case	struct	switch	template
this	throw	true	try	typedef	typeid
typename	union	unsigned	using	virtual	void
volatile	wchar_t	while			

C++还有一些用单词或字母组合来代替操作符的替代标记,用户也不能用替代标记来表示其他实体。表 1-2 是 C++的替代标记。关于运算符的含义将在第 3 章详细介绍。

表 1-2 C++的替代标记

替代标记	所替代的运算符	替代标记	所替代的运算符
and	&&	not_e	!=
and_eq	&=	or	
bitand	&	or_eq	!=
bitor		xor	^
compl	~	xor_eq	^=
not	!		



提示:

● 程序员自己定义的标识符不能与表 1-1 和表 1-2 中的关键字相同, 否则, 程序在运行时会出现错误。

1.2.3 使用 Visual C++ 2005 创建程序

本书使用 Visual C++ 2005 集成开发环境来创建程序。图 1-1 是 Visual C++ 2005 集成开发的主要工作界面。关于 Visual C++ 2005 更详细用法请参考与本书配套的《C++程序设计语言上机实习指导与习题集》一书的附录 A。

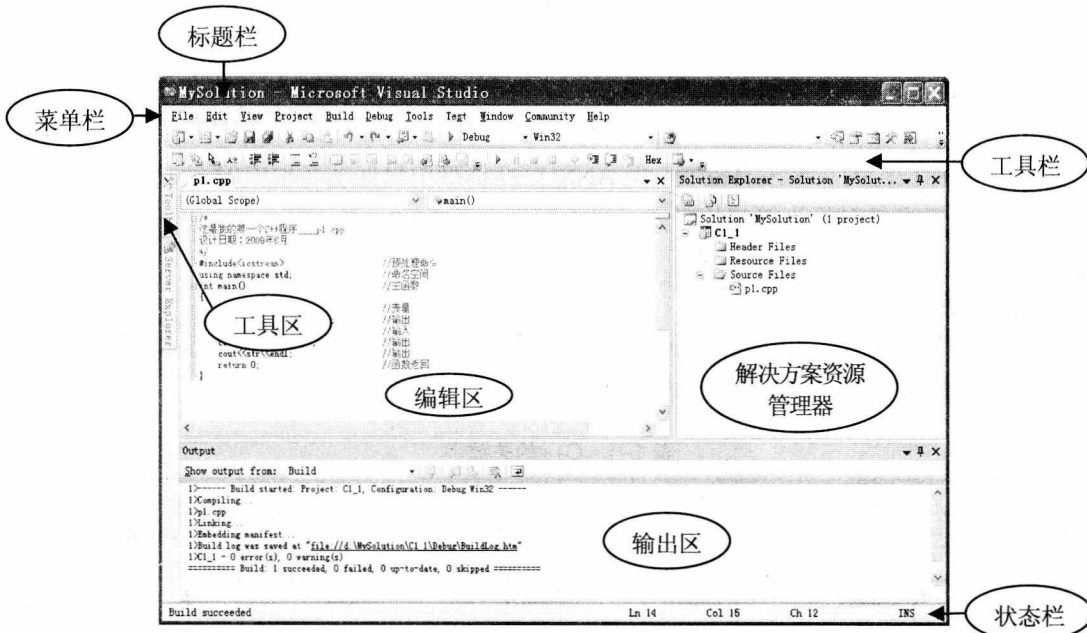


图 1-1 Visual C++ 2005 集成开发环境

下面是在 Visual C++ 2005 (以下简称 VC++) 中编辑、编译、连接和运行第一个 C++ 程序“大家好!”的步骤。

1. 创建项目

开发一个软件, 相当于开发一个项目。项目的作用是协调、组织好一个软件中的所有程序代码、头文件及其他额外资源; 解决方案的作用是组织和管理多个相关的项目, 一般情况下, 一个解决方案都会包含多个项目。使用 VC++ 开发 C++ 程序的第一步是创建一个项目。操作步骤如下:

① 单击图 1-1 左上角的“File”菜单, 然后选择“New”选项, 再在出现的子菜单中选择“Project”, 出现如图 1-2 所示的对话框。