



工业和信息化部职业教育指导委员会“十二五”规划教材
全国高等职业教育计算机系列规划教材

C++程序设计 与案例分析

◎从书编委会

<http://www.phei.com.cn>

Programing & Case is Analytical

C++



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

工业和信息产业职业教育教学指导委员会“十二五”规划教材
全国高等职业教育计算机系列规划教材

C++程序设计 与案例分析

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本教材根据最新颁布的全国计算机等级考试(二级 C++语言程序设计)大纲的要求,结合目前 C++语言程序设计的广泛使用,全书以“工作过程引领”和“项目驱动”形式,围绕“学生成绩管理系统”为主线的实战演练,并融入作者多年从事 C++语言程序设计的实践经验编写而成。

全书共分为 10 章,由浅入深地介绍了 C++语言概述;C++语言的数据类型、表达式和基本运算符;程序设计;数组、指针和引用、函数和预处理;类、对象和友元;派生和继承、多态性和虚函数;运算符重载;模板。全书除了针对各个知识点利用例题讲解以外,还将 2002 年以来国家计算机等级(二级 C++)考试的典型试题以练习题的形式附在各个章节之后,并进行了解释说明,同时还将读者容易犯的错误以列举的形式进行了详细阐述,以方便读者进行更正。每章后都附有实验指导,方便读者上机实际操作。

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++程序设计与案例分析/《全国高等职业教育计算机系列规划教材》编委会编. —北京:电子工业出版社, 2010.10

工业和信息化部职业教育教学指导委员会“十二五”规划教材. 全国高等职业教育计算机系列规划教材

ISBN 978-7-121-11810-4

I. ①C… II. ①全… III. ①C 语言—程序设计—高等学校:技术学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 178253 号

策划编辑:左 雅

责任编辑:左 雅 文字编辑:裴 杰

印 刷:涿州市京南印刷厂

装 订:涿州市桃园装订有限公司

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:23.75 字数:608 千字

印 次:2010 年 10 月第 1 次印刷

印 数:4 000 册 定价:38.00 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。

质量投诉请发邮件至 zltz@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

丛 书 编 委 会

主 任 郝黎明 逢积仁

副 主 任 左 雅 方一新 崔炜 姜广坤 范海波 敖广武 徐云晴 李华勇

委 员 (按拼音排序)

陈国浪	迟俊鸿	崔爱国	丁 倩	杜文洁	范海绍	何福男
贺 宏	槐彩昌	黄金栋	蒋卫祥	李 琦	刘宝莲	刘红军
刘 凯	刘兴顺	刘 颖	卢锡良	孟宪伟	庞英智	钱 哨
乔国荣	曲伟峰	桑世庆	宋玲玲	王宏宇	王 华	王晶晶
温丹丽	吴学会	邢彩霞	徐其江	严春风	姚 嵩	殷广丽
尹 辉	俞海英	张洪明	张 薇	赵建伟	赵俊平	郑 伟
周绯非	周连兵	周瑞华	朱香卫	邹 羚		

本 书 编 委 会

主 编 范海波

副 主 编 蔡桂英 翟海庆 黄国军

参 编 陈秀玲 郭 鑫 赵 薇 刘志凯

丛书编委会院校名单

(按拼音排序)

保定职业技术学院	山东省潍坊商业学校
渤海大学	山东司法警官职业学院
常州信息职业技术学院	山东信息职业技术学院
大连工业大学职业技术学院	沈阳师范大学职业技术学院
大连水产学院职业技术学院	石家庄信息工程职业学院
东营职业学院	石家庄职业技术学院
河北建材职业技术学院	苏州工业职业技术学院
河北科技师范学院数学与信息技术学院	苏州托普信息职业技术学院
河南省信息管理学校	天津轻工职业技术学院
黑龙江工商职业技术学院	天津市河东区职工大学
吉林省经济管理干部学院	天津天狮学院
嘉兴职业技术学院	天津铁道职业技术学院
交通运输部管理干部学院	潍坊职业学院
辽宁科技大学高等职业技术学院	温州职业技术学院
辽宁科技学院	无锡旅游商贸高等职业技术学校
南京铁道职业技术学院苏州校区	浙江工商职业技术学院
山东滨州职业学院	浙江同济科技职业学院
山东经贸职业学院	

前 言

自从教育部公布关于全国中、高等职业教育首批示范专业（点）和加强示范专业建设的通知以来，示范专业成为中、高等职业教育教学领域改革、提高教育教学质量和办学效益的试验和示范基地。各国家级、省市级示范院校努力推进职业教育观念、专业建设机制的创新，增强职业教育适应经济结构调整、技术进步和劳动力市场变化的能力，全面实施素质教育，坚持为社会生产第一线培养高素质劳动者和实用型人才。

在国家政策的引导和市场需求的双重作用下，中、高等职业教育招生规模逐年扩大，生源特点不断变化，专业设置和培养目标不断调整，这对职业学校的专业、课程、教材建设与改革提出了更高的要求，更大的挑战。

计算机专业是职业教育中招生规模最大、开设学校最多的专业之一，因而亟需一批走在教学改革前列的示范专业学校，将最新的教学改革和专业、课程、教材建设的最新成果加以普及，以引领和带动其他学校的进步，达到教育部建设示范专业学校的目的。

本书特点如下：

1. 编写模式采用“工作过程引领”和“项目驱动”等方式，通过例题、练习、实战演练和习题的训练提高编程能力。
2. 教材内容符合就业岗位及实际要求，以增加学生学习兴趣和实用性为第一要务。
3. 该教材编入了 2002 年以来国家计算机等级 C++ 二级考试的典型试题，作为每章每节的练习题以方便读者学习和借鉴。

本书共分 10 章，各章的内容布局如下：第 1 章介绍 C++ 语言概述；第 2 章介绍 C++ 语言的数据类型、表达式和基本运算符；第 3 章介绍程序设计；第 4 章介绍数组、指针和引用；第 5 章介绍函数和预处理；第 6 章介绍类、对象和友元；第 7 章介绍派生和继承；第 8 章介绍多态性和虚函数；第 9 章介绍运算符重载；第 10 章介绍模板。

本书强调在实战演练和做习题的过程中强化知识点的掌握，所有练习均提供了分析和答案，便于读者对难点及重点知识的快速掌握，以具备达到独立编写程序的能力。

本书可以作为各类院校“C++ 语言”或相关课程的教材，也可作为各类培训班有关课程的教材。对于参加全国计算机二级 C++ 考试的读者，也不失为一本相当实用的参考书。衷心希望每一位读者能够从本书中获益，同时欢迎对书中的不足提出宝贵意见。

本书由范海波负责组织编写并统稿。第 1 章由范海波编写；第 2 章由刘志凯编写；第 3 章由陈秀玲和黄国军编写；第 4、6 章由郭鑫编写；第 5、7 章由翟海庆编写；第 8、9 章由蔡桂英编写；第 10 章由赵薇编写。

作 者

目 录

第 1 章 C++语言概述	(1)
1.1 C++语言的发展过程	(1)
1.2 C++语言的特点	(2)
1.3 C++语言的基本语法	(3)
1.3.1 C++程序构成	(3)
1.3.2 书写程序时应遵循的规则	(4)
1.3.3 C++语言的字符集	(5)
1.3.4 C++语言词汇	(5)
1.4 面向对象的程序设计技术	(6)
1.5 Visual C++ 6.0 开发环境及 C++上机过程	(7)
1.5.1 Visual C++ 6.0 简介	(7)
1.5.2 Visual C++ 6.0 主界面	(8)
1.5.3 在 Visual C++ 6.0 环境中开发 C++程序	(8)
1.6 本章小结及常见错误列举	(13)
1.7 实战演练	(14)
习题	(14)
第 2 章 C++语言的数据类型、表达式和基本运算符	(16)
2.1 数据类型	(16)
2.1.1 基本数据类型	(16)
2.1.2 非基本数据类型	(17)
2.1.3 应用举例	(20)
练习 1	(21)
2.2 常量与变量	(22)
练习 2	(24)
2.3 整型数据	(25)
2.4 实型数据	(26)
2.5 字符型数据	(26)
2.6 变量赋初值和类型转换	(28)
练习 3	(30)
2.7 运算符简介	(31)
2.8 算术运算符和算术表达式	(31)
2.9 关系运算符和关系表达式	(32)
2.10 逻辑运算符和逻辑表达式	(33)

2.11 赋值运算符和赋值表达式	(34)
2.12 强制类型转换运算符	(35)
2.13 自增、自减运算符	(36)
2.14 位运算符	(36)
练习 4	(37)
2.15 本章小结及常见错误列举	(38)
2.16 实战演练	(39)
习题	(41)
第 3 章 程序设计	(43)
3.1 基本语句	(43)
3.2 顺序控制语句及实例	(44)
3.2.1 输入 cin	(44)
3.2.2 输出 cout	(46)
练习 1	(47)
3.2.3 putchar()	(47)
3.2.4 getchar()	(48)
3.2.5 输入/输出格式操作符	(49)
3.2.6 顺序结构程序设计举例	(51)
3.3 选择控制语句及实例	(53)
3.3.1 简单 if 语句	(53)
3.3.2 if-else 语句	(54)
3.3.3 if-else if 语句	(55)
练习 2	(56)
3.3.4 嵌套的 if 语句	(57)
3.3.5 switch 语句	(59)
3.4 循环控制语句	(62)
3.4.1 while 语句	(62)
练习 3	(63)
3.4.2 do-while 语句	(64)
练习 4	(65)
3.4.3 for 语句	(66)
练习 5	(67)
3.4.4 goto、continue、break 和 return 语句	(67)
3.4.5 循环语句的比较	(69)
3.4.6 循环的嵌套	(69)
3.4.7 应用举例	(70)
练习 6	(73)
3.5 综合程序设计	(73)
3.6 本章小结及常见错误列举	(75)

3.7 实战演练	(77)
习题	(78)
第4章 数组、指针和引用	(83)
4.1 一维数组	(83)
4.1.1 一维数组的定义	(83)
4.1.2 一维数组的初始化	(84)
4.1.3 一维数组应用举例	(85)
4.2 二维数组	(87)
4.2.1 二维数组的一般定义形式	(87)
4.2.2 二维数组初始化	(88)
4.2.3 二维数组应用举例	(89)
4.3 数组与字符串	(91)
4.3.1 字符数组	(91)
4.3.2 字符数组的使用	(92)
4.3.3 字符串处理函数	(92)
4.4 指针	(94)
4.4.1 指针的概念	(94)
4.4.2 指针的使用	(96)
4.4.3 指针的运算	(98)
4.5 指针与数组	(99)
4.5.1 指针与数组的关系	(99)
4.5.2 数组指针与指针数组	(100)
4.6 动态内存分配	(101)
4.6.1 new 操作符	(102)
4.6.2 delete 操作符	(102)
4.7 指针与函数	(103)
4.7.1 指针作为函数参数	(103)
4.7.2 函数指针	(104)
4.8 引用	(105)
4.8.1 引用的定义	(105)
4.8.2 函数参数的引用传递方式	(107)
4.9 本章小结及常见错误列举	(108)
4.10 实战演练	(109)
习题	(112)
第5章 函数和预处理	(115)
5.1 函数的定义和调用	(115)
5.1.1 函数的概述	(116)
5.1.2 函数的定义	(116)
5.1.3 函数的调用	(117)

练习 1	(119)
5.2 函数的参数传递、返回值及函数声明	(119)
5.2.1 函数的参数传递及传值调用	(120)
5.2.2 函数返回值	(123)
5.2.3 函数声明	(130)
练习 2	(131)
5.3 全局变量和局部变量	(135)
5.4 函数的调用机制	(136)
5.5 作用域与标识符的可见性	(139)
5.5.1 存储类型与标识符的生命期	(141)
5.5.2 存储类型	(141)
5.5.3 生命期	(144)
练习 3	(144)
5.6 函数的递归调用	(146)
5.7 函数的重载、内联及默认参数	(150)
5.7.1 函数重载	(150)
5.7.2 默认参数	(153)
5.7.3 内联函数	(155)
练习 4	(156)
5.8 头文件与多文件结构	(158)
5.8.1 头文件	(159)
5.8.2 多文件结构	(159)
5.9 编译预处理	(162)
5.9.1 宏定义命令	(162)
5.9.2 文件包含命令	(163)
5.9.3 条件编译命令	(164)
练习 5	(166)
5.10 本章小结及常见错误列举	(167)
5.11 实战演练	(169)
习题	(174)
第 6 章 类、对象和友元	(178)
6.1 类	(178)
6.1.1 类的一般结构	(178)
6.1.2 成员函数的定义	(180)
6.1.3 类作用域	(181)
6.2 对象	(182)
6.3 构造函数与析构函数	(183)
6.3.1 构造函数	(184)
6.3.2 复制构造函数	(186)

6.3.3	析构函数	(188)
6.4	静态成员	(189)
6.4.1	静态数据成员	(189)
6.4.2	静态成员函数	(191)
6.5	类对象作为成员	(193)
6.6	对象数组	(194)
6.7	this 指针	(195)
6.8	友元	(196)
6.8.1	友元函数	(196)
6.8.2	友元的使用	(198)
6.9	本章小结及常见错误列举	(200)
6.10	实战演练	(203)
	习题	(208)
第 7 章	派生和继承	(211)
7.1	派生类	(211)
7.1.1	派生类的定义格式	(212)
7.1.2	派生类生成过程	(213)
7.2	访问控制	(217)
7.2.1	公有继承	(218)
7.2.2	私有继承	(219)
7.2.3	保护继承	(220)
	练习 1	(222)
7.3	派生类的构造函数和析构函数	(229)
7.3.1	构造函数	(229)
7.3.2	析构函数	(236)
	练习 2	(237)
7.4	虚基类	(244)
7.4.1	作用域分辨符	(245)
7.4.2	虚基类说明	(247)
7.4.3	虚基类的初始化	(250)
	练习 3	(252)
7.4.4	派生和继承实例	(254)
7.5	本章小结及常见错误列举	(268)
7.6	实战演练	(269)
	习题	(275)
第 8 章	多态性和虚函数	(279)
8.1	静态联编和动态联编	(279)
	练习 1	(281)
8.2	虚函数	(281)

8.2.1	虚函数的说明	(281)
8.2.2	多继承中的虚函数	(282)
8.2.3	虚函数的限制	(284)
	练习 2	(286)
8.3	纯虚函数和抽象类	(290)
8.3.1	纯虚函数	(290)
8.3.2	抽象类	(292)
	练习 3	(294)
8.4	抽象类的实例	(294)
	练习 4	(297)
8.5	本章小结及常见错误列举	(300)
	练习 5	(302)
8.6	实战演练	(304)
	习题	(315)
第 9 章	运算符重载	(325)
9.1	运算符重载概述	(325)
	练习 1	(328)
9.2	单目运算符重载	(328)
	练习 2	(330)
9.3	双目运算符重载	(331)
	练习 3	(332)
9.4	关系运算符重载	(334)
9.5	赋值运算符重载	(335)
9.5.1	复合赋值运算符重载	(335)
9.5.2	直接赋值运算符重载	(337)
9.6	下标运算符重载	(339)
9.7	运算符 new 与 delete 重载	(340)
9.8	逗号运算符重载	(341)
9.9	类型转换运算符重载	(342)
9.10	运算符重载应用实例	(343)
	练习 4	(344)
9.11	本章小结及常见错误列举	(346)
	练习 5	(347)
9.12	实战演练	(348)
	习题	(352)
第 10 章	模板	(357)
10.1	模板的概念	(357)
10.2	函数模板	(358)
10.2.1	函数模板定义	(358)

10.2.2 函数模板的使用	(358)
练习 1	(360)
10.3 类模板	(360)
10.3.1 类模板定义	(360)
10.3.2 类模板的使用	(361)
练习 2	(362)
10.4 本章小结及常见错误列举	(362)
10.5 实战演练	(363)
习题	(364)
参考文献	(365)

第1章 C++语言概述

本章考点

了解C++语言的基本符号、词汇，掌握C++程序和基本构成；能够使用Visual C++ 6.0集成开发环境编辑、编译、运行与调试程序。

学习目标

- 了解C++语言的基本符号。
- 了解C++语言的词汇（保留字、标识符、常量、运算符、标点符号等）。
- 能够使用Visual C++ 6.0集成开发环境编辑、编译、运行与调试程序。
- 掌握C++程序的基本框架（结构程序设计框架、面向对象程序设计框架等）。

1.1 C++语言的发展过程

C++是一种通用的程序设计语言，它以其独特的语言机制和全面支持面向对象编程的优势受到了很多程序员的钟爱，并在计算机科学领域得到了广泛的应用，例如，读者所熟悉的 Windows 操作系统就是用 C++语言和 C 语言编写的。

C++语言是直接从 C 语言发展过来的，C 语言的出现是与 UNIX 操作系统密切联系在一起的，C 语言的本身也有一个产生和发展的过程。具体的发展情况如下：

1960 年出现的 ALGOL 60 是一种面向问题的高级语言，由于它离硬件比较远，所以不宜用来编写系统程序。

1963 年英国的剑桥大学推出了 CPL 语言。CPL 语言在 ALGOL 60 的基础上比较接近硬件，但规模比较大，难以实现。

1967 年英国剑桥大学 Martin Richards 对 CPL 语言做了简化，推出了 BCPL (Basic Combined Programming Language) 语言。

1970 年美国贝尔实验室的 Ken Thomson 以 BCPL 语言为语言基础，又做了进一步简化，设计出了既简单又很接近硬件的 B 语言，并用 B 语言编写了第一个 UNIX 操作系统，在 PDP-7 上实现。但 B 语言缺乏丰富的数据类型，又以字长编址，有一定的缺陷。



自 1971 年开始, D.Ritchie 用了一年左右的时间, 在 B 语言的基础上加入了丰富的数据类型和强有力的数据结构, 从而形成了 C 语言。Dennis Ritchie 开发 C 语言的主要目的是为了能够更好地描述 UNIX 操作系统。

1973 年 Ken Thomson 和 D.M. Ritchie 两人合作把 90% 以上的 UNIX 用 C 语言改写。

1978 年 Brian W. Kernighan、Ken Thomson 和 Dennis Ritchie 三人合作, 写了一本著名的书《The C Programming Language》, 该书介绍的 C 语言被称为标准 C 语言。

1983 年, 美国国家标准化协会 (ANSI) 建立了一个委员会, 着手制定 ANSI 的标准 C。

1988 年, ANSI 公布了标准 ANSIC。这个标准的大部分特性已经由现代的编译系统所支持, 使得 C 的可移植性很强。

1989 年, 正式制订了 C 语言的标准: ANSIC。

最初导致 C++ 诞生的原因是在 Bjarne 博士等人试图去分析 UNIX 内核的时候, 这项工作开始于 1979 年 4 月, 当时由于没有合适的工具能够有效地分析由于内核分布而造成的网络流量, 以及怎样将内核模块化。同年 10 月, Bjarne 博士完成了一个可以运行的预处理程序, 称为 Cpre, 它为 C 加上了类似 Simula 的类机制。在这个过程中, Bjarne 博士开始思考是不是要开发一种新的语言, 当时贝尔实验室对这个想法很感兴趣, 就让 Bjarne 博士等人组成一个开发小组, 专门进行研究。

当时不是叫做 C++, 而是 C With Class, 这是把它当做一种 C 语言的有效扩充。由于当时 C 语言在编程界居于“老大”的地位, 要想发展一种新的语言, 最强大的竞争对手就是 C 语言, 所以当时有两个问题最受关注: C++ 要在运行时间、代码紧凑性和数据紧凑性方面能够与 C 语言相媲美, 但是还要尽量避免在语言应用领域的限制。在这种情况下, 一个很自然的想法就是让 C++ 语言从 C 语言继承过来, 但是 Bjarne 博士更具有先见之明, 他为了避免受到 C 语言的局限性, 参考了很多的语言, 例如, 从 Simula 继承了类的概念, 从 ALGO168 继承了运算符重载、引用及在任何地方声明变量的能力。

1983 年 12 月, Rick Mascitti 建议命名为 CPlusPlus, 即 C++。在经历了 3 次对 C++ 的修订后, 于 1994 年 ANSI / ISO 委员会制订了 ANSI C++ 标准的草案。以后又经过不断完善, 1998 年 11 月, 正式批准了 ANSI C++ 标准。C++ 包含 C 的全部特征、属性和优点, 同时又增加了对面向对象编程的支持, 而这一关键特性引发了软件业关于程序设计开发的一场革命, 使人们从过去的过程式开发技术和结构化开发技术过渡为面向对象开发技术。

1.2 C++语言的特点

C++ 语言的主要特点有两个方面: 全面支持 C 语言与面向对象。

1. 全面兼容 C 语言

C++ 语言是从 C 语言发展而来的, 它保持了 C 语言的简洁、高效和接近汇编语言等特点。同时, 对 C 语言的类型系统进行了改革和扩充, 堵塞了 C 语言中的许多漏洞, C++



编译提供了更好的类型检查和编译时的分析，能检查出更多的类型错误。改善了 C 语言的安全性，比 C 语言更安全。

2. 支持面向过程的程序设计

C++语言最重要的特点是支持面向对象的程序设计，使用 C++编程，编程效率高；由于面向对象的方法更接近人类认识世界的方法，C++对于问题更容易描述，程序更容易理解与维护；C++模板对库代码的重用提供了支持；C++是一种支持多种程序设计方法的语言，提供对过程化和基于对象的程序设计方法的支持。

1.3 C++语言的基本语法

1.3.1 C++程序构成

1. 函数

一个 C++程序是由若干个函数构成的。函数分为库函数（标准函数）和自定义函数。库函数一般是由系统提供的。一个完整的 C++语言程序只有一个主函数。

2. 预处理命令

预处理命令以位于行首的符号“#”开始，C++提供的预处理命令有宏定义命令、文件包含命令和条件编译命令 3 种。

3. 输入和输出

C++程序中总是少不了输入和输出的语句，以实现与程序内部的信息交流。特别是屏幕输出的功能，几乎每个程序都要使用，这些语句的功能是把计算的结果显示在屏幕上。

4. 程序语句

语句是组成程序的基本单元。函数是由若干条语句组成的。但是，空函数是没有语句的。语句是由单词组成的，单词间用空格符分隔，C++程序中的语句又是以分号结束。语句除了有表达式语句和空语句之外，还有复合语句、分支语句、循环语句和转向语句等若干类。

5. 变量

多数程序都需要说明和使用变量。广义讲，对象包含了变量，即将变量也称为一种对象，狭义讲，将对象看做是类的实例，对象是指某个类的对象。

6. 其他

除了以上讲述的 5 个部分以外，还有其他组成部分。例如，符号常量和注释信息也是程序的一部分。C++中都尽量把常量定义为符号常量，在 C++的程序中出现的是符号常量，该符号常量代表着某个确定的常量值。定义符号常量的方法有两种：一种是使用



宏定义的命令来定义符号常量，例如：

```
#define PI 3.1415
```

其中，`define` 是宏定义命令的关键字，它是预处理命令，该命令定义一个符号常量 `PI`，它所代表的常量值是 3.1415。另一种方法使用常量关键字 `const`，例如：

```
const double PI=3.1415;
```

这也是将符号常量 `PI` 定义为 3.1415。

使用符号常量会带来许多好处：便于修改，便于移植，增加可读性等。

注释信息也是 C++ 程序中的一部分，较为复杂的或大型的程序都少不了注释信息。注释信息是对所编写程序做解释的。因此，加上注释信息自然可以提高对程序的可读性。注释信息的写法有两种方式。对于一行注释信息的情况下，使用“//”符号加在注释信息前面。对于连续多行注释信息的情况下，使用“/*”和“*/”比较方便。

下面是一个简单的 C++ 程序的例子（计算圆的面积，要求用户输入半径 `R`，在屏幕上输出面积 `S`）：

```
#include <iostream>           //预处理命令
#define PI 3.1415926          //预处理命令
using namespace std;          //全局声明部分
int main ()                    //主函数首部
{
    int R,S;                   //函数内声明部分，声明 R 和 S 这两个变量
    cout << "半径 R=";         //输出执行语句
    cin >> R;                   //输入执行语句，输入 R 的值
    S=PI*R*R;                  //S 和 R 的关系表达式，S=PI X R X R,面积公式
    cout << "面积 S=" << S;    //输出执行语句，输出 S 计算后的值
    return 0;                  //执行语句，返回
}
```

1.3.2 书写程序时应遵循的规则

C++ 语言程序的书写格式自由度高，灵活性强，随意性大，如一行内可写一条语句，也可写几条语句；一个语句也可分写在多行内。具体要求如下：

(1) 一行一般写一条语句。短语句可以一行写多个。长语句可以一条写多行。分行原则是不能将一个单词分开。

(2) 不同结构层次的语句从不同的起始位置开始，即在同一结构层次中的语句缩进同样的字数。

(3) 表示结构层次的大括弧写在该结构化语句第一个字母的下方，与结构化语句对齐，并占用一行。

(4) 适当加些空格和空行。