



完全手册丛书

Schildt 的经典 C++ 参考大全
(更新与扩展版)

C++: The Complete Reference

Fourth Edition

C++ 参考大全 (第四版)

[美] Herbert Schildt 著 周志荣 朱德芳 于秀山 等译

涵盖 C++ 的国际标准，包括关键字、语法和库

包括一些高级特征，如重载、继承、虚函数、名字空间、模板、STL 和 RTTI

使用所有的 C++ 编译器，包括 Visual C++ 编译器



电子工业出版社
Publishing House of Electronics Industry
<http://www.phei.com.cn>

C++: The Complete Reference Fourth Edition

C++ 参考大全 (第四版)

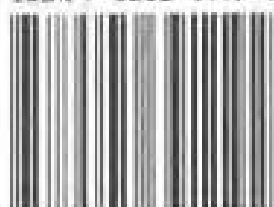
编程大师和计算机畅销书作者Herbert Schildt对他的C++经典指南进行了全面的更新，本书就是这一全面更新的结果。作者利用数百个程序范例为我们演示了C++的方方面面。在本书中，读者会发现关于整个C++语言的细节，包括它的关键字、运算符、预处理器指令和库，甚至有.NET编程所用的扩展关键字的概要介绍。不管是初学者，还是有经验的程序员，本书都对你可能问到的问题做出了解答。

详细内容包括：

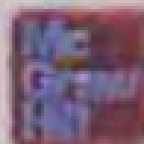
- 数据类型和运算符
- 控制语句
- 函数
- 类和对象
- 构造函数和析构函数
- 函数和运算符重载
- 继承
- 虚函数
- 名字空间
- 模板
- 异常处理
- I/O 库
- 标准模板库 (STL)
- 容器、算法和迭代器
- 面向对象编程的原则 (OOP)
- 运行时类型 ID (RTTI)
- 预处理器

Herbert Schildt 是世界级编程语言专家，他是 C、C++、Java 和 C# 编程语言方面的权威，也是标准化 C++ 的 ANSI/ISO 委员会的一名成员。他撰写的编程方面的书籍全球销量超过 300 万册并被翻译成了多种语言。

ISBN 7-5053-9119-4



9 787505 391192 >



Mc
Graw
Hill



责任编辑：赵红燕
封面设计：毛惠康

本书贴有激光防伪标志，凡没有防伪标志者，属盗版图书

ISBN 7-5053-9119-4/TP · 5270 定价：69.00 元

完全手册丛书

C++ 参考大全

(第四版)

C++: The Complete Reference
Fourth Edition

[美] Herbert Schildt 著

周志荣 朱德芳 于秀山 等译

电子工业出版社
Publishing House of Electronics Industry
北京·Beijing

内 容 简 介

本书是根据著名C语言专家 Herbert Schildt 的著作翻译的。这是一本关于C++语言的百科全书,包括C和C++的命令、功能、编程和应用等方面的内容。全书分为五个部分:C++基础:C子集;C++的专有特征;标准函数库;标准C++类库;C++应用程序范例。详细描述和演示了定义C++语言的关键字、语法、函数、类和特征。其中第一部分全面讨论了C++的C子集;第二部分详细介绍了C++本身的特性,如类和对象、构造函数、析构函数和模板等;第三部分描述了标准函数库;第四部分讨论了标准类库,包括STL(标准模板库);第五部分显示了两个应用C++和面向对象编程的实际例子。

本书内容全面、翔实,是学习C++编程语言的广大学生的一部有用的工具书,也是对C++感兴趣的读者的必备参考书。

Herbert Schildt: C++: The Complete Reference, Fourth Edition. ISBN: 0-07-222680-3.

Copyright © 2003 by The McGraw-Hill Companies, Inc.

Original language published by The McGraw-Hill Companies, Inc. All rights reserved. No part of this publication may be reproduced or distributed in any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

Simplified Chinese translation edition jointly published by McGraw-Hill Education (Asia) Co. and Publishing House of Electronics Industry. Copyright © 2003.

本书中文简体字翻译版由电子工业出版社和美国麦格劳-希尔教育出版(亚洲)公司合作出版。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有 McGraw-Hill 公司激光防伪标签,无标签者不得销售。

版权贸易合同登记号:图字:01-2002-5891

图书在版编目(CIP)数据

C++参考大全:第四版/(美)希尔特(Schildt, H.)著;周志荣等译.-北京:电子工业出版社,2003.9
(完全手册丛书)

书名原文:C++: The Complete Reference, Fourth Edition.

ISBN 7-5053-9119-4

I.C... II.①希... ②周... III.C语言-程序设计 IV.TP312

中国版本图书馆CIP数据核字(2003)第078457号

责任编辑:赵红燕

印刷者:北京兴华印刷厂

出版发行:电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编:100036

经 销:各地新华书店

开 本:787×1092 1/16 印张:43.25 字数:1107千字

版 次:2003年9月第1版 2003年9月第1次印刷

定 价:69.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换;若书店售缺,请与本社发行部联系。

联系电话:(010)68279077。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。

前 言

如果有一种语言定义了现代化程序设计的概念，它就是C++。C++的语法、风格和理念设立了所有其他语言得以进行评判的标准。此外，C++是一种通用的程序设计语言。当对算法或技巧进行描述时，通常使用C++语法来完成。C++长时间的成功在计算机语言开发的历史上留下了不可磨灭的印记，例如，Java和C#都是从C++中派生出的。坦率地讲，要成为一名专业的程序设计人员，就要精通C++。C++是一种所有程序员都不能忽视的语言。

本书全面描述和演示了定义C++语言的关键字、语法、函数、类和特征。更准确地讲，本书全面描述了标准C++语言。标准C++是由ANSI/ISO C++标准定义的C++版本，也是为所有主要编译器、包括Microsoft的Visual C++和Borland的C++ Builder所支持的C++版本。因此，本书中的内容适用于所有现代编程环境。

自本书前一版出版以来，C++语言并没有变化，然而，计算环境却有了相当大的改变。例如，C语言的一个新标准，即C99得以建立，Java变成了进行Web编程的主要语言，.NET框架发布，C#产生。在过去几年所发生的这些变化中，有一件事保持不变：C++的强大功能。C++一直是并将仍然是未来开发高性能软件的首选语言。

第四版中的新内容

第四版的整个结构和组织与第三版类似。因此，如果你一直在使用第三版，你会非常喜欢第四版。第四版的主要变化涉及整个内容的更新和扩展。有时加入了额外的细节内容，有时对问题的表述方法进行了修改，有时对描述的内容进行了更新以反映当前的编程环境，也添加了几个新的章节。在第一部分适当的地方，提到了我们称为C99的新的C标准，并论述了它与C++的关系。

另外，添加了两个附录。附录A描述了由Microsoft定义的扩展关键字，用于创建.NET框架的可管理代码。附录B描述了机器人。作者一直对机器人感兴趣，也相信许多读者会对作者的实验用机器人感兴趣。当然，生成它的多数软件是用C++编写的。最后，所有的代码范例都用现有的编译器进行了重新测试，包括Microsoft的Visual Studio.NET和Borland的C++ Builder。

本书内容

本书详细讨论了C++语言的所有方面，包括它的基础：C语言。本书分为下面五个部分：

- C++ 基础：C子集
- C++ 语言
- 标准函数库
- 标准C++类库
- C++ 应用程序范例

第一部分全面讨论了C++的C子集。就像大部分读者将要知道的，C语言是C++得以建立的基础。正是C子集定义了C++的基础特征，包括for循环和if语句。它也定义了C++语言块结构、指针和函数的基本性质。因为许多读者对C语言非常熟悉，可称得上是专业水平的，所以在本书第一部分单独讨论C子集就不会使有经验的C程序员重复阅读他们已经知道的东西。相反，他们可以直接进入本书讨论C++专有特征的部分开始学习。

第二部分详细讨论了C语言基础以外的特征，同时也定义了C++语言，包括它的面向对象的特征，如类、构造函数、析构函数、RTTI和模板。因此，第二部分讨论了“使C++成为C++”的那些结构。

第三部分描述了标准函数库，第四部分讨论了标准类库，包括STL（标准模板库），第五部分给出了两个应用C++和面向对象编程的实际例子。

一本面向所有程序员的书

本书是为所有C++程序员设计的，而不管他们的经验水平。然而，本书假定读者能够创建至少一个简单的程序。如果你正在学习C++，本书将是与任何C++教程配套的极好的参考书，也可用做回答特定问题的参考书。有经验的C++支持者将会发现对C++高级特征的详细讨论特别有用。

如果使用的是 Windows 操作系统

如果你的计算机使用的是Windows操作系统，那么，你选择了正确的语言。C++特别适合于Windows程序设计。然而，本书中的程序都不是Windows程序，相反，它们是基于控制台的程序。理由很简单：从本质上讲，Windows程序很大、很复杂。创建一个小的Windows程序的开销是50到70行代码。要编写演示C++特征的Windows程序，则需要几百行代码。简言之，Windows不是一个讨论编程语言特征的合适的环境。然而，仍然可以使用基于Windows的编译器来编译本书中的程序，因为编译器将自动创建一个控制台会话，在这个会话中执行程序。

Web 上的代码

记住，本书中所有程序的源代码都可以在Web上免费得到，其网址为www.osborne.com。下载这些代码可以使你免于自己键入范例代码。

目 录

第一部分 C++ 基础: C 子集

第 1 章 C 语言概述	3
1.1 C 语言的起源和历史	3
1.2 C 语言是中级语言	4
1.3 C 语言是结构化语言	5
1.4 C 语言是程序员的语言	6
1.5 C 程序的结构	7
1.6 库和链接	8
1.7 分别编译	9
1.8 理解.C 和.CPP 文件扩展	9
第 2 章 表达式	10
2.1 五种基本数据类型	10
2.2 修饰基本类型	10
2.3 标识符名称	12
2.4 变量	12
2.5 const 和 volatile 限定符	16
2.6 存储类限定符	18
2.7 变量初始化	22
2.8 常量	23
2.9 运算符	24
2.10 表达式	36
第 3 章 语句	40
3.1 C 和 C++ 中的真值和假值	40
3.2 选择语句	40
3.3 迭代语句	49
3.4 在选择和迭代语句内声明变量	57
3.5 跳转语句	57
3.6 表达式语句	62
3.7 块语句	62
第 4 章 数组和以 null 结束的字符串	63
4.1 一维数组	63
4.2 生成指向数组的指针	64
4.3 向函数传递一维数组	64
4.4 以 null 结束的字符串	65

4.5	二维数组	67
4.6	多维数组	71
4.7	带下标的指针	72
4.8	数组初始化	73
4.9	棋盘游戏实例	75
第 5 章	指针	79
5.1	什么是指针	79
5.2	指针变量	79
5.3	指针运算符	80
5.4	指针表达式	81
5.5	指针和数组	84
5.6	多级间址	85
5.7	初始化指针	86
5.8	指向函数的指针	87
5.9	C 语言的动态分配函数	89
5.10	指针应用中的问题	91
第 6 章	函数	94
6.1	函数的一般形式	94
6.2	函数作用域的规则	94
6.3	函数变元	95
6.4	传给 main() 的变元 argc 和 argv	99
6.5	return 语句	101
6.6	递归	105
6.7	函数原型	106
6.8	声明变长参数列表	108
6.9	传统的与现代的函数参数声明	108
第 7 章	结构、联合、枚举和用户定义的类型	110
7.1	结构	110
7.2	结构数组	113
7.3	向函数传递结构	113
7.4	结构指针	115
7.5	结构中的数组和结构	118
7.6	位域	118
7.7	联合	120
7.8	枚举	122
7.9	用 sizeof 来保证可移植性	124
7.10	typedef	125
第 8 章	C 风格的控制台 I/O	127
8.1	一个重要的应用说明	127
8.2	读写字符	128

8.3 读写字符串	130
8.4 格式化的控制台 I/O	132
8.5 printf()	132
8.6 scanf()	137
第9章 文件 I/O	143
9.1 C 与 C++ 的文件 I/O	143
9.2 流和文件	143
9.3 流	143
9.4 文件	144
9.5 文件系统基础	144
9.6 fread() 和 fwrite()	153
9.7 fseek() 和随机访问 I/O	155
9.8 fprintf() 和 fscanf()	156
9.9 标准流	157
第10章 预处理器和注释	160
10.1 预处理器	160
10.2 #define	160
10.3 #error	162
10.4 #include	162
10.5 条件编译指令	163
10.6 #undef	166
10.7 使用 defined	166
10.8 #line	166
10.9 #pragma	167
10.10 # 和 ## 预处理器运算符	167
10.11 预定义的宏名	168
10.12 注释	168

第二部分 C++ 的专有特征

第11章 C++ 语言概述	172
11.1 C++ 的起源	172
11.2 什么是面向对象的程序设计	173
11.3 C++ 基础	174
11.4 老的 C++ 与现代 C++	179
11.5 C++ 的类	182
11.6 函数重载	185
11.7 运算符重载	187
11.8 继承	187
11.9 构造函数和析构函数	191
11.10 C++ 的关键字	194

11.11 C++ 程序的一般形式	195
第 12 章 类和对象	196
12.1 类	196
12.2 结构和类是相互关联的	198
12.3 联合和类是相互关联的	200
12.4 友元函数	201
12.5 友元类	205
12.6 内联函数	206
12.7 在类中定义内联函数	207
12.8 带参数的构造函数	208
12.9 带一个参数的构造函数：特例	210
12.10 静态类成员	211
12.11 何时执行构造函数和析构函数	216
12.12 作用域分辨符	217
12.13 嵌套类	217
12.14 局部类	217
12.15 向函数传递对象	218
12.16 返回对象	220
12.17 对象赋值	221
第 13 章 数组、指针、引用和动态分配运算符	222
13.1 对象数组	222
13.2 指向对象的指针	224
13.3 C++ 指针的类型检查	226
13.4 this 指针	226
13.5 指向派生类型的指针	228
13.6 指向类成员的指针	229
13.7 引用	231
13.8 格式问题	236
13.9 C++ 的动态分配运算符	237
第 14 章 函数重载、拷贝构造函数和默认变元	245
14.1 函数重载	245
14.2 重载构造函数	246
14.3 拷贝构造函数	249
14.4 查找重载函数的地址	252
14.5 重载的过去与现在	253
14.6 默认的函数变元	253
14.7 函数重载和二义性	257
第 15 章 运算符重载	261
15.1 创建成员运算符函数	261
15.2 使用友元函数的运算符重载	266

15.3	重载 new 和 delete	270
15.4	重载某些特殊运算符	277
15.5	重载逗号运算符	282
第 16 章	继承	284
16.1	基类访问控制	284
16.2	继承和保护成员	285
16.3	继承多个基类	289
16.4	构造函数、析构函数和继承	290
16.5	准许访问	296
16.6	虚基类	297
第 17 章	虚函数与多态性	302
17.1	虚函数	302
17.2	继承虚属性	305
17.3	虚函数是分层的	306
17.4	纯虚函数	308
17.5	使用虚函数	310
17.6	早期绑定与后期绑定	312
第 18 章	模板	313
18.1	通用函数	313
18.2	应用通用函数	319
18.3	通用类	322
18.4	关键字 typename 和 export	330
18.5	模板的功用	331
第 19 章	异常处理	332
19.1	异常处理基础	332
19.2	处理派生类异常	338
19.3	异常处理选项	339
19.4	理解 terminate() 和 unexpected()	343
19.5	uncaught_exception() 函数	345
19.6	exception 和 bad_exception 类	345
19.7	异常处理的应用	345
第 20 章	C++ 输入/输出系统基础	347
20.1	老的 C++ I/O 与现代的 C++ I/O	347
20.2	C++ 的流	347
20.3	C++ 的流类	348
20.4	格式化的 I/O	349
20.5	重载 << 和 >>	358
第 21 章	C++ 文件的输入/输出	368
21.1	<fstream> 和文件类	368

21.2	打开和关闭文件	368
21.3	读写文本文件	370
21.4	无格式和二进制 I/O	372
21.5	其他 get() 函数	376
21.6	getline() 函数	376
21.7	检测 EOF	377
21.8	ignore() 函数	379
21.9	peek() 和 putback() 函数	379
21.10	flush() 函数	380
21.11	随机访问	380
21.12	I/O 状态	383
21.13	定制的 I/O 和文件	384
第 22 章	运行时类型标识与强制转换运算符	387
22.1	运行时类型标识	387
22.2	强制转换运算符	394
22.3	dynamic_cast	394
第 23 章	名字空间、转换函数和其他高级主题	403
23.1	名字空间	403
23.2	std 名字空间	409
23.3	创建转换函数	411
23.4	const 成员函数与 mutable	414
23.5	volatile 成员函数	415
23.6	explicit 构造函数	415
23.7	成员初始化语法	416
23.8	利用关键字 asm	420
23.9	连接说明	421
23.10	基于数组的 I/O	421
23.11	C 与 C++ 的区别	426
第 24 章	标准模板库	428
24.1	STL 概述	428
24.2	容器类	430
24.3	一般的操作原理	431
24.4	vector 容器	432
24.5	list 容器	439
24.6	map 容器	447
24.7	算法	452
24.8	使用函数对象	459
24.9	string 类	464
24.10	关于 STL 的最后一点说明	473

第三部分 标准函数库

第 25 章 基于 C 的输入 / 输出函数	477
25.1 clearerr 函数	477
25.2 fclose 函数	477
25.3 feof 函数	478
25.4 ferror 函数	478
25.5 fflush 函数	478
25.6 fgetc 函数	478
25.7 fgetpos 函数	479
25.8 fgets 函数	479
25.9 fopen 函数	479
25.10 fprintf 函数	480
25.11 fputc 函数	481
25.12 fputs 函数	481
25.13 fread 函数	481
25.14 freopen 函数	481
25.15 fscanf 函数	482
25.16 fseek 函数	482
25.17 fsetpos 函数	483
25.18 ftell 函数	483
25.19 fwrite 函数	483
25.20 getc 函数	483
25.21 getchar 函数	484
25.22 gets 函数	484
25.23 perror 函数	484
25.24 printf 函数	484
25.25 puts 函数	486
25.26 putchar 函数	486
25.27 puts 函数	487
25.28 remove 函数	487
25.29 rename 函数	487
25.30 rewind 函数	487
25.31 scanf 函数	487
25.32 setbuf 函数	490
25.33 setvbuf 函数	490
25.34 sprintf 函数	490
25.35 sscanf 函数	490
25.36 tmpfile 函数	491
25.37 tmpnam 函数	491
25.38 ungetc 函数	491

25.39	vprintf, vfprintf 和 vsprintf 函数	492
第 26 章	字符串与字符函数	493
26.1	isalnum 函数	493
26.2	isalpha 函数	493
26.3	iscentrl 函数	493
26.4	isdigit 函数	494
26.5	isgraph 函数	494
26.6	islower 函数	494
26.7	isprint 函数	494
26.8	ispunct 函数	494
26.9	isspace 函数	495
26.10	isupper 函数	495
26.11	isxdigit 函数	495
26.12	memchr 函数	495
26.13	mememp 函数	495
26.14	memcpy 函数	496
26.15	memmove 函数	496
26.16	memset 函数	496
26.17	strcat 函数	496
26.18	strchr 函数	497
26.19	strcmp 函数	497
26.20	strcoll 函数	497
26.21	strcpy 函数	497
26.22	strcspn 函数	498
26.23	strerror 函数	498
26.24	strlen 函数	498
26.25	strncat 函数	498
26.26	strncmp 函数	498
26.27	strncpy 函数	499
26.28	strpbrk 函数	499
26.29	strrchr 函数	499
26.30	strspn 函数	499
26.31	strstr 函数	500
26.32	strtok 函数	500
26.33	strxfrm 函数	500
26.34	tolower 函数	500
26.35	toupper 函数	501
第 27 章	数学函数	502
27.1	acos 函数	502
27.2	asin 函数	502

27.3	atan 函数	503
27.4	atan2 函数	503
27.5	ceil 函数	503
27.6	cos 函数	503
27.7	cosh 函数	503
27.8	exp 函数	504
27.9	fabs 函数	504
27.10	floor 函数	504
27.11	fmod 函数	504
27.12	frexp 函数	504
27.13	ldexp 函数	505
27.14	log 函数	505
27.15	log10 函数	505
27.16	modf 函数	505
27.17	pow 函数	506
27.18	sin 函数	506
27.19	sinh 函数	506
27.20	sqrt 函数	506
27.21	tan 函数	506
27.22	tanh 函数	507
第 28 章	时间、日期和定位函数	508
28.1	asctime 函数	508
28.2	clock 函数	509
28.3	ctime 函数	509
28.4	difftime 函数	509
28.5	gmtime 函数	509
28.6	localeconv 函数	510
28.7	localtime 函数	511
28.8	mktime 函数	511
28.9	setlocale 函数	511
28.10	strftime 函数	512
28.11	time 函数	512
第 29 章	动态分配函数	514
29.1	calloc 函数	514
29.2	free 函数	514
29.3	malloc 函数	515
29.4	realloc 函数	515
第 30 章	实用函数	516
30.1	abort 函数	516

30.2	abs 函数	516
30.3	assert 函数	516
30.4	atexit 函数	517
30.5	atof 函数	517
30.6	atoi 函数	517
30.7	atol 函数	518
30.8	bsearch 函数	518
30.9	div 函数	518
30.10	exit 函数	519
30.11	getenv 函数	519
30.12	labs 函数	519
30.13	ldiv 函数	519
30.14	longjmp 函数	520
30.15	mblen 函数	520
30.16	mbstowcs 函数	520
30.17	mbtowc 函数	520
30.18	qsort 函数	521
30.19	raise 函数	521
30.20	rand 函数	522
30.21	setjmp 函数	522
30.22	signal 函数	522
30.23	srand 函数	522
30.24	strtod 函数	523
30.25	strtol 函数	523
30.26	strtoul 函数	523
30.27	system 函数	524
30.28	va_arg, va_start 和 va_end 函数	524
30.29	wctombs 函数	524
30.30	wctomb 函数	525
第 31 章	宽字符函数	526
31.1	宽字符分类函数	526
31.2	宽字符 I/O 函数	528
31.3	宽字符串函数	529
31.4	宽字符串转换函数	529
31.5	宽字符数组函数	530
31.6	多字节 / 宽字符转换函数	530

第四部分 标准 C++ 类库

第 32 章	标准 C++ I/O 类	534
32.1	I/O 类	534
32.2	I/O 头文件	535

32.3	格式化标记和 I/O 操作算子	536
32.4	几个数据类型	537
32.5	重载<<和>>运算符	538
32.6	通用的 I/O 函数	538
第 33 章	STL 容器类	549
33.1	容器类	549
第 34 章	STL 算法	564
34.1	adjacent_find	564
34.2	binary_search	564
34.3	copy	565
34.4	copy_backward	565
34.5	count	565
34.6	count_if	565
34.7	equal	565
34.8	equal_range	565
34.9	fill 和 fill_n	566
34.10	find	566
34.11	find_end	566
34.12	find_first_of	566
34.13	find_if	567
34.14	for_each	567
34.15	generate 和 generate_n	567
34.16	includes	567
34.17	inplace_merge	567
34.18	iter_swap	568
34.19	lexicographical_compare	568
34.20	lower_bound	568
34.21	make_heap	568
34.22	max	568
34.23	max_element	569
34.24	merge	569
34.25	min	569
34.26	min_element	569
34.27	mismatch	570
34.28	next_permutation	570
34.29	nth_element	570
34.30	partial_sort	570
34.31	partial_sort_copy	571
34.32	partition	571
34.33	pop_heap	571

34.34	prev_permutation	571
34.35	push_heap	572
34.36	random_shuffle	572
34.37	remove, remove_if, remove_copy 和 remove_copy_if	572
34.38	replace, replace_copy, replace_if 和 replace_copy_if	573
34.39	reverse 和 reverse_copy	573
34.40	rotate 和 rotate_copy	573
34.41	search	574
34.42	search_n	574
34.43	set_difference	574
34.44	set_intersection	574
34.45	set_symmetric_difference	575
34.46	set_union	575
34.47	sort	575
34.48	sort_heap	576
34.49	stable_partition	576
34.50	stable_sort	576
34.51	swap	576
34.52	swap_ranges	576
34.53	transform	576
34.54	unique 和 unique_copy	577
34.55	upper_bound	577
第 35 章	STL 迭代器、分配器和函数对象	578
35.1	迭代器	578
35.2	函数对象	585
35.3	分配器	590
第 36 章	字符串类	592
36.1	basic_string 类	592
36.2	char_traits 类	598
第 37 章	数字类	600
37.1	complex 类	600
37.2	valarray 类	603
37.3	数字算法	614
第 38 章	异常处理和杂项类	618
38.1	异常	618
38.2	auto_ptr	619
38.3	pair 类	621
38.4	本地化	621
38.5	其他有趣的类	621

第五部分 C++ 应用程序范例

第 39 章 集成新的类：自定义字符串类	624
39.1 StrType 类	624
39.2 构造函数和析构函数	625
39.3 字符串 I/O	626
39.4 赋值函数	627
39.5 连接	628
39.6 子字符串减法	630
39.7 关系运算符	632
39.8 各种字符串函数	632
39.9 完整的 StrType 类	633
39.10 使用 StrType 类	640
39.11 创建和集成新类型	642
39.12 挑战	642
第 40 章 分析表达式	643
40.1 表达式	643
40.2 分析表达式：问题	644
40.3 分析一个表达式	644
40.4 parser 类	645
40.5 剖析一个表达式	646
40.6 一个简单的表达式分析器	648
40.7 向分析器中添加变量	653
40.8 递归下降分析器中的语法检查	660
40.9 构建一个通用的分析器	661
40.10 需要试验的一些东西	667
附录 A C++ 的 .NET 可管理扩展	669
附录 B C++ 和机器人时代	672

参考大全

C++

第一部分

C++ 基础：C 子集

本书分两部分讲述 C++ 语言。第一部分讨论 C++ 的类 C 特征，通常称为 C++ 的 C 子集。第二部分描述 C++ 特有的特征。这两部分合在一起描述了整个 C++ 语言。

我们知道，C++ 是建立在 C 的基础之上的。事实上，C++ 包括了整个 C 语言，并且所有的 C 程序也是 C++ 程序（只有极少的例外）。当发明 C++ 时，是以 C 语言为出发点的。在 C 语言的基础上增加了几个新特性并对其进行扩展以支持面向对象的编程（OOP），然而，C++ 的类 C 特性永远不会被放弃，1989 年发布的 ANSI/ISO C 标准是 C++ 国际标准的基本文档。因此，对 C++ 的理解意味着对 C 的理解。

在本书中，把 C++ 语言划分为两部分——C 语言基础和 C++ 特有的特性，这样做有三个主要的优点：

- 清楚地划出了 C 和 C++ 之间的分界线。
- 熟悉 C 语言的读者可以很容易地发现 C++ 特有的信息。
- 它提供了一个便于讨论与 C 子集密切相关的 C++ 特征的地方。

理解 C 和 C++ 之间的分界线是非常重要的，因为 C 和 C++ 的用途很广，很可能需要同时编写和维护 C 和 C++ 代码。当使用 C 代码时，需要知道 C 在哪里结束，C++ 在哪里开始。许多 C++ 程序员经常会被要求编写限于“C 子集”的代码。对于嵌入式系统编程和现有应用的维护，这种情况是极为常见的。要想成为顶级 C++ 程序员，了解 C 和 C++ 之间的区别是其中的一部分工作。

当把 C 代码转换为 C++ 代码时，理解 C 也是非常有价值的。要以专业的方式做到这一点，就要对 C 有深刻的了解。例如，如果没有全面了解 C 的 I/O 系统，就不能把 I/O 密集的 C 程序以有效的方式转换成 C++ 程序。

许多读者可能都懂 C 语言。在本书中单独讨论 C++ 的类 C 特征会使有经验的 C 程序员很快、很容易地找到关于 C++ 的信息，而不必阅读大量他已经知道的信息。当然，贯穿本书的第一部分，也将提到 C 和 C++ 之间的任何小的区别。还有，通过把 C 的基础和高级特征相分离，C++ 的面向对象的特征使我们可以把重点放到那些高级特征上，因为所有的基础知识都已经讨论过了。

尽管 C++ 包含完整的 C 语言，当编写 C++ 风格的程序时，并不是 C 所提供的所有特征都很常用。例如，对 C++ 程序员来说，C 的 I/O 系统仍然可用，即使 C++ 定义了它自己的面向对象的版本。预处理器是另一个例子。对 C 来讲，预处理器是非常重要的，但是对 C++ 来讲，并非如此。在本书第一部分讨论几个“仅适用于 C”的特征，防止了它们与本书的其他部分相混淆。

记住：在第一部分描述的 C 子集构成了 C++ 的核心和基础，正是在这个基础上，C++ 的面向对象的特征才得以创建。这里描述的所有特征都是 C++ 的一部分，都可以使用。