

C++语言大全

● 杨长虹 徐 磊 等编

● 电子工业出版社



中国最大的软件公司
中软总公司



C++语言大全

杨长虹 徐 碚 等编

刘德贵 审

电子工业出版社

(京)新登字 055 号

C++语言大全

杨长虹 徐 砦 等编

刘德贵 审

责任编辑 路 石

特邀编辑 杨宝琪

*

电子工业出版社出版(北京市万寿路)

河北省望都县印刷厂印刷

电子工业出版社发行 各地新华书店经销

*

开本 787×1092 1/32 印张:22 字数:494 千字

1992 年 2 月第 1 版 1992 年 2 月第 1 次印刷

印数:1-10100 册 定价:15.80 元

ISBN 7-5053-2105-6/TP.551

JS/31/05

内容简介

《C++语言大全》是根据国际著名的 C 语言专家 Herbert Schildt 的有关专著编写的。这是一本 C++ 语言的百科全书,包括 C 和 C++ 命令、功能、编程和应用等方面内容,是集专家、编者及 C 语言编程人员的多方面专门经验之作。全书分三大部分,共 23 章和一个附录。第一部分为 C++ 基础,讨论 C 和 C++ 语言的共性,内容包括表达式、C 语句、数组和串、指针、函数、数据类型、输出和输入与处理等;第二部分详细介绍了 C++ 本身的专有特性,讨论了类和对象、虚函数和继承性、构造函数和析构函数、面向目标的输入与输出系统以及函数和运算符重载特性等;第三部分用实例讨论了 C++ 软件的开发技术,包括字符串、弹出窗口和链表等基本应用软件结构技术。

本书内容全面,叙述清晰,为 PC 机及其广大用户和程序开发人员提供了一部很有用的工具书,是计算机应用人员、有关大专院校师生及 PC 机软件开发人员的必备参考书。

参加本书编写的还有曾凡奎和李刚。

序 言

八十年代以来,随着微机在我国应用的普及,C语言逐渐成为众多程序员的主要编程工具之一。计算机语言同其它事物一样也在发展,C++就是C语言的扩充增强版本。C语言是1980年由美国贝尔实验室的Bjarne Stroustrup首次提出的,当时叫“含类的C语言”,1983年改名为C++。1985年和1989年经过两次主要修订,C语言在程序设计艺术和学术上都产生了重大的飞跃。

C语言之所以受到人们的如此青睐,是由于它以高级程序设计语言的结构和编程环境,提供了类似汇编语言那样的系统资源、操纵能力和程序的执行效率。然而,用C语言编程超过10万行以上,程序就会变得相当复杂,难于掌握,因此C++的目的就是要适应大型系统程序的开发,使程序人员可以理解和管理更大、更复杂的程序。C++支持面向对象的程序设计。C++面向对象的特性,使程序结构清晰、便于扩展、易于维护并具有很高的效率。C++不仅有益于面向对象的程序设计,而且编写非面向对象的程序也十分容易。

我们仍然坚持这种观点:对C和C++来讲,都将是“入门容易得道难”。真正掌握C和C++的精髓,充分发挥其潜力,绝非一日之功。

C++的能力和通用性以及C语言的普及,实际上已为C++的发展奠定了坚实的基础。许多程序设计专家预言,九十年代将是C++的年代。

本书是根据国际著名程序设计专家 Herbert Schildt 有关 C 和 C++ 的专著,以及 C 语言的编程实践编写的。它详细介绍了 C++ 的类 C 特性及其专有特性,重点介绍 C++ 的专有特性。

鉴于许多读者已经熟悉或掌握了 C 语言,所以我们将 C++ 的类 C 特性放在 C++ 特性之前单独讨论。本书第一部分介绍 C++ 与 C 语言的共性;第二部分详细讨论 C++ 对 C 语言的扩充和增强;第三部分列举了应用 C++ 和面向对象的程序设计的一些实例。

全书的程序均作了源程序盘片,可以提供用户使用。每张盘(含邮费)40 元,需要者可和北京丰台区郑常庄 307 号软件中心(100039)杨长虹联系。

编 者

1992 年 1 月

目 录

第一部分 C++ 基础:类 C 特性

第一章 C 语言概述	(3)
1.1 C 语言的起源	(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	(7)
1.5 C 语言的程序结构	(9)
1.6 库和连接	(11)
1.7 分离编译	(12)
第二章 表达式	(14)
2.1 五种基本数据类型	(14)
2.2 修饰基本类型	(16)
2.3 标识符命名	(17)
2.4 变量	(18)
2.4.1 变量在哪里说明	(18)
2.4.2 局部变量	(18)
2.4.3 形式参数	(22)
2.4.4 全局变量	(23)
2.5 存取修饰符	(25)
2.5.1 const	(25)
2.5.2 volatile	(27)

2.6	存储分类符	(28)
2.6.1	外部的	(28)
2.6.2	静态变量	(30)
2.6.3	寄存器变量	(33)
2.7	变量初始化	(35)
2.8	常量	(36)
2.8.1	十六进制和八进制常量	(37)
2.8.2	串常量	(37)
2.8.3	反斜线字符常量	(37)
2.9	运算符	(39)
2.9.1	赋值运算符	(39)
2.9.2	赋值中的类型转换	(40)
2.9.3	多重赋值	(41)
2.9.4	算术运算符	(42)
2.9.5	增量和减量运算符	(43)
2.9.6	关系和逻辑运算符	(44)
2.9.7	位操作符	(47)
2.9.8	? 操作符	(52)
2.9.9	指针操作符 & 和 *	(53)
2.9.10	编译时操作符 sizeof	(55)
2.9.11	逗号操作符	(56)
2.9.12	点(.)和箭头(->)操作符	(56)
2.9.13	括号和方括号操作符	(57)
2.9.14	关于优先级的小结	(58)
2.10	表达式	(58)
2.10.1	求值顺序	(59)
2.10.2	表达式中的类型转换	(59)
2.10.3	构成符	(60)
2.10.4	空格与括号	(61)

2.10.5 C 语言中的简写形式	(62)
第三章 C 语言的语句	(63)
3.1 C 语言的真值和假值	(64)
3.2 选择语句	(64)
3.2.1 if 语句	(64)
3.2.2 嵌套 if 语句	(66)
3.2.3 多重嵌套 if: 阶梯型 if-else-if 语句	(68)
3.2.4 操作符? 的替代	(69)
3.2.5 条件表达式	(73)
3.2.6 switch 开关语句	(74)
3.2.7 嵌套的 switch 语句	(78)
3.3 迭代语句	(79)
3.3.1 for 循环	(79)
3.3.2 for 循环的变化型式	(80)
3.3.3 无限循环	(85)
3.3.4 没有循环体的 for 循环	(86)
3.3.5 while 循环	(87)
3.3.6 do-while 循环	(90)
3.4 转移语句	(91)
3.4.1 return 语句	(91)
3.4.2 goto 语句	(92)
3.4.3 break 语句	(93)
3.4.4 exit 语句	(95)
3.4.5 continue 语句	(96)
3.5 表达式语句	(98)
3.6 块语句	(99)
第四章 数组和字符串	(100)
4.1 一维数组	(101)
4.2 产生指向数组的指针	(102)

4.3	向函数传递一维数组	(103)
4.4	字符串	(104)
4.5	二维数组	(106)
4.6	多维数组	(113)
4.7	带下标的指针	(114)
4.8	数组初始化	(116)
4.9	棋盘游戏实例	(119)
第五章	指针	(124)
5.1	指针是什么?	(125)
5.2	指针变量	(125)
5.3	指针运算符	(126)
5.4	指针表达式	(128)
5.4.1	指针赋值	(128)
5.4.2	指针运算	(128)
5.4.3	指针比较	(130)
5.5	指针和数组是紧密相联的	(133)
5.6	多级间址	(135)
5.7	指针的初始化	(137)
5.8	指向函数的指针	(139)
5.9	C 语言的动态分配函数	(142)
5.10	指针应用中的某些问题	(144)
第六章	函数	(149)
6.1	函数的一般形式	(149)
6.2	函数的作用域规则	(150)
6.3	函数变元	(151)
6.3.1	传值调用和引用调用	(152)
6.3.2	引用调用的建立	(153)

6.3.3 用数组调用函数	(154)
6.4 传给 main() 的变元 argc 与 argv	(158)
6.5 返回语句	(162)
6.5.1 从函数返回	(162)
6.5.2 返回值	(164)
6.6 返回非整型值的函数	(166)
6.7 函数原型	(168)
6.8 返回指针	(171)
6.9 void 类型的函数	(172)
6.10 main() 返回什么?	(173)
6.11 递归	(174)
6.12 说明变长参数表	(176)
6.13 传统的与现代的函数说明	(176)
6.14 实现问题	(178)
6.14.1 参数和通用函数	(178)
6.14.2 效率	(178)
第七章 结构、联合、枚举及用户定义的类型	(181)
7.1 结构	(182)
7.1.1 引用结构元素	(184)
7.1.2 结构赋值	(185)
7.2 结构数组	(185)
7.3 向函数传递结构	(186)
7.3.1 向函数传递结构元素	(186)
7.3.2 向函数传递完整结构	(187)
7.4 结构指针	(188)
7.5 结构中的数组和结构	(192)
7.6 位域	(193)
7.7 联合	(196)

7.8	枚举	(199)
7.9	用 sizeof 增强移植性	(202)
7.10	类型定义 typedef	(203)
第八章	控制台 I/O	(205)
8.1	一个重要的应用说明	(206)
8.2	读写字符	(206)
8.2.1	getchar() 的有关问题	(208)
8.2.2	getchar() 的替代	(208)
8.3	读写字符串	(209)
8.4	格式化的控制台 I/O	(213)
8.5	printf()	(213)
8.5.1	打印字符	(214)
8.5.2	打印数字	(215)
8.5.3	显示一个地址	(216)
8.5.4	%n 描述符	(217)
8.5.5	格式化代码修饰符	(217)
8.5.6	最小域宽符	(217)
8.5.7	精度描述符	(219)
8.5.8	调整输出	(220)
8.5.9	处理其它数据类型	(221)
8.5.10	* 和 # 修饰符	(221)
8.6	scanf()	(222)
8.6.1	格式说明符	(223)
8.6.2	输入数字	(223)
8.6.3	输入无符号整数	(224)
8.6.4	用 scanf() 读单个字符	(225)
8.6.5	读字符串	(225)
8.6.6	读入地址	(226)
8.6.7	%n 描述符	(226)

8.6.8	使用一个扫描集	(226)
8.6.9	丢弃期望的空白符	(228)
8.6.10	控制串中的非空白符	(228)
8.6.11	必须向 scanf() 传递地址	(228)
8.6.12	格式化修饰符	(229)
8.6.13	压缩输入	(229)
第九章	ANSI C 的标准文件 I/O	(231)
9.1	历史回顾	(231)
9.2	流和文件	(232)
9.2.1	流	(232)
9.2.2	文件	(233)
9.3	文件系统基础	(234)
9.3.1	文件指针	(235)
9.3.2	打开一个文件	(236)
9.3.3	关闭一个文件	(238)
9.3.4	写一个字符	(238)
9.3.5	读一个字符	(239)
9.3.6	使用 fopen(), getc(), putc() 和 fclose()	(240)
9.3.7	使用 feof()	(242)
9.3.8	用 fputs() 和 fgets() 操作字符串	(244)
9.3.9	rewind()	(245)
9.3.10	ferror()	(246)
9.3.11	删除文件	(249)
9.3.12	清除一个流	(250)
9.4	fread() 和 fwrite()	(250)
9.5	fseek() 和随机存取 I/O	(253)
9.6	fprintf() 和 fscanf()	(255)
9.7	标准流	(256)
9.7.1	控制台 I/O 连接	(257)

9.7.2 使用 freopen()重定向标准流	(258)
第十章 C 语言的预处理程序和注释	(260)
10.1 C 语言的预处理程序	(260)
10.2 #define	(261)
10.3 #error	(264)
10.4 #include	(264)
10.5 条件编译指令	(265)
10.5.1 #if, #else, #elif, 和 #endif	(265)
10.5.2 #ifdef 和 #ifndef	(269)
10.6 #undef	(270)
10.7 #line	(271)
10.8 #pragm	(272)
10.9 #和##预处理操作符	(272)
10.10 预定义的宏名	(273)
10.11 注释	(274)

第二部分 C++ 的专有特征

第十一章 C++概述	(277)
11.1 C++的起源	(278)
11.2 面象对象的程序设计是什么	(279)
11.2.1 对象	(280)
11.2.2 多态性	(280)
11.2.3 继承性	(281)
11.3 C++的程序设计风格	(281)
11.4 C++的类	(287)
11.5 函数重载	(292)
11.6 运算符重载	(296)

11.7	继承性.....	(297)
11.8	构造函数和析构函数.....	(303)
11.9	C++的关键字	(308)
11.10	C++程序的一般结构	(309)
第十二章	类和对象	(310)
12.1	类.....	(310)
12.2	结构和类.....	(315)
12.3	联合和类.....	(317)
12.4	友元函数.....	(320)
12.5	内联函数	(326)
12.6	在类中定义内联函数.....	(329)
12.7	参数化的构造函数.....	(331)
12.8	静态类成员.....	(333)
12.8.1	静态数据成员	(333)
12.8.2	静态成员函数	(337)
12.9	何时执行构造函数和析构函数.....	(340)
12.10	嵌套类	(342)
12.11	作用域分辨符	(342)
12.12	局部类	(343)
12.13	向函数传递对象	(344)
12.14	返回对象	(347)
12.15	对象赋值	(348)
第十三章	数组、指针和引用	(349)
13.1	对象数组.....	(349)
13.2	指向对象的指针.....	(352)
13.3	this 指针	(354)
13.4	指向派生类型的指针.....	(356)

13.5	指向类成员的指针.....	(359)
13.6	引用.....	(362)
13.6.1	引用参数	(363)
13.6.2	向对象传递引用	(366)
13.6.3	返回引用	(368)
13.6.4	独立引用	(369)
13.6.5	对引用的限制	(371)
13.7	格式问题.....	(371)
13.8	C++的动态分配符	(372)
第十四章	函数和运算符重载.....	(382)
14.1	函数重载.....	(382)
14.2	重载的过去和现在.....	(388)
14.3	重载构造函数.....	(388)
14.4	求重载函数的地址.....	(391)
14.5	运算符重载.....	(392)
14.6	使用 friend 重载运算符	(400)
14.6.1	使用 friend 重载++和--	(403)
14.6.2	friend operator 函数增加了灵活性	(405)
14.7	重载的 new 和 delete	(408)
14.8	重载某些特殊运算符.....	(414)
14.8.1	重载 []	(414)
14.8.2	重载 ()	(418)
14.8.3	重载 ->	(421)
14.9	重载逗号运算符.....	(421)
第十五章	继承性.....	(425)
15.1	基类存取控制.....	(425)
15.2	继承多个基类.....	(433)
15.3	构造函数、析构函数和继承	(434)

15.3.1	何时执行构造函数和析构函数	(434)
15.3.2	向基类构造函数传递参数	(438)
15.4	授权存取	(443)
15.5	虚基类	(446)
第十六章	虚函数和多态性	(452)
16.1	虚函数	(452)
16.1.1	继承虚属性	(456)
16.1.2	虚函数的层次性	(458)
16.2	纯虚函数	(461)
16.3	使用虚函数	(464)
16.4	早期和后期联编	(467)
第十七章	C++的 I/O 系统基础	(469)
17.1	C++的流	(470)
17.2	基本的流类	(470)
17.3	格式化的 I/O	(471)
17.3.1	用 ios 成员格式化	(472)
17.3.2	设置格式标志	(473)
17.3.3	清除格式标志	(475)
17.3.4	setf()的重载形式	(475)
17.3.5	检查格式标志	(478)
17.3.6	设置所有标志	(481)
17.3.7	使用 width()、precision()和 fill()	(482)
17.3.8	用操纵符格式化 I/O	(484)
17.4	重载 << 和 >>	(486)
17.4.1	创建自己的插入符	(487)
17.4.2	创建自己的提取符	(494)
17.5	创建自己的操纵符函数	(497)
17.5.1	创建无参数的操纵符	(498)