

Microsoft[®]
Visual C++[™] 2.0
for Win32[®] 大全(五)
——语言参考手册

[美] Microsoft Corporation 著
张茂芝 王 敏 译
沈纪新 审校

清华大学出版社

(京)新登字 158 号

Microsoft® Visual C++™ 2.0 for Win32® 大全(五)——语言参考手册
Microsoft Visual C++ Language Reference
Microsoft Corporation

Copyright 1995 by Microsoft Corporation.

Original English language Edition Copyright © 1994 by Microsoft Corporation.
Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A.

本书中文版由 Microsoft Press 授权清华大学出版社出版。

中华人民共和国国家版权局著作权合同登记章 图字: 01-95-314 号

**未经出版者书面允许,不得以任何方式复制或抄袭本书的内容。
本书封面贴有 Microsoft Press 激光防伪标签,无标签者不得销售。**

图书在版编目(CIP)数据

Microsoft® Visual C++™ 2.0 for Win32® 大全(五):语言参考手册/张茂芝,王敏译.—北京:清华大学出版社,1995
ISBN 7-302-01977-0

I.M… II.①张… ②王… III.C语言-程序设计 IV.TP312C

中国版本图书馆 CIP 数据核字(95)第 17038 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:化学工业出版社印刷厂

发行者:新华书店总店北京科技发行所

开本:787×1092 1/16 印张:33 字数:821 千字

版次:1996年3月第1版 1996年3月第1次印刷

书号:ISBN 7-302-01977-0/TP·912

印数:0001—3000

定价:64.00 元

目 录

第一部分 C语言参考手册

引言.....	3
第1章 C的语言成份	5
1.1 单词	5
1.2 关键字	7
1.3 标识符	8
1.4 常量.....	11
1.4.1 浮点常量	11
1.4.2 整常量	13
1.4.3 字符常量	16
1.5 字符串.....	19
1.6 标点符号和特殊字符.....	21
第2章 程序的结构	22
2.1 源文件和源程序.....	22
2.2 main 函数和程序的执行	25
2.2.1 使用 wmain	26
2.2.2 变元的说明	26
2.2.3 扩展的变元通配符	27
2.2.4 命令行变元的分析	27
2.2.5 定制命令行处理	29
2.3 生命期,作用域,可见性和链接.....	29
2.3.1 生命期	29
2.3.2 作用域与可见性	30
2.3.3 生命期和可见性小节	31
2.3.4 链接	32
2.4 名字空间.....	32
第3章 声明和类型	34
3.1 声明概览.....	34
3.2 存储类.....	36
3.2.1 外部声明的存储类定义符	37
3.2.2 内部声明的存储类定义符	39
3.2.3 函数声明中的存储类定义符	41
3.3 类型定义符.....	41
3.3.1 数据类型定义符与其等价符	42

3.4	类型属性符	43
3.5	被声明符和变量声明	44
3.5.1	简单变量声明	45
3.5.2	枚举声明	46
3.5.3	结构声明	48
3.5.4	联合声明	53
3.5.5	数组声明	56
3.5.6	指针声明	57
3.5.7	相对指针	59
3.5.8	抽象声明	60
3.6	理解更复杂的被声明符	61
3.7	初始化	63
3.7.1	初始化比例类型	63
3.7.2	初始化聚合类型	64
3.7.3	初始化字符串	67
3.8	基本类型的存储	68
3.9	不完全类型	71
3.10	Typedef 定义	72
3.11	扩展存储类属性	74
3.11.1	DLL 输入和输出	75
3.11.2	裸露函数	75
3.11.3	线程局部存储	75
第 4 章	表达式和赋值	77
4.1	操作符和表达式	77
4.1.1	基本表达式	77
4.1.2	左值和右值表达式	79
4.1.3	常量表达式	79
4.1.4	表达式求值	80
4.2	操作符	82
4.2.1	求值的优先级和顺序	82
4.2.2	通常的算术转换	84
4.2.3	后缀操作符	85
4.2.4	一元操作符	89
4.2.5	强制操作符	92
4.2.6	乘操作符	93
4.2.7	加操作符	94
4.2.8	按位移操作符	96
4.2.9	关系和相等操作符	97
4.2.10	按位操作符	98

4.2.11	逻辑操作符	99
4.2.12	条件表达式操作符	100
4.2.13	赋值操作符	101
4.2.14	顺序求值操作符	103
4.3	类型转换	103
4.3.1	赋值转换	104
4.3.2	类型强制转换	108
4.3.3	函数调用转换	109
第5章	语句	110
5.1	概述	110
5.2	break 语句	111
5.3	复合语句	111
5.4	continue 语句	112
5.5	do-while 语句	113
5.6	表达式语句	113
5.7	for 语句	114
5.8	goto 和带标号的语句	115
5.9	if 语句	116
5.10	空语句	117
5.11	return 语句	117
5.12	switch 语句	118
5.13	try-except 语句	121
5.14	try-finally 语句	123
5.15	while 语句	124
第6章	函数	125
6.1	概述	125
6.2	函数定义	126
6.2.1	函数属性	128
6.2.2	DLL 输入和输出函数	130
6.2.3	裸体函数	133
6.2.4	存储类	136
6.2.5	返回类型	137
6.2.6	形参	138
6.2.7	函数体	139
6.3	函数原型	140
6.4	函数调用	141
6.4.1	实参	143
6.4.2	用可变个数的参数调用	144
6.4.3	递归函数	145

附录A C语言语法总结	146
A.1 定义和约定	146
A.2 词法	146
A.3 短语结构语法	150
附录B 由实现所定义的行为	157
B.1 翻译: 诊断	157
B.2 环境	157
B.3 标识符	158
B.4 字符	158
B.5 整数	160
B.6 浮点数学	161
B.7 数组和指针	162
B.8 寄存器: 寄存器的可用性	162
B.9 结构, 联合, 枚举, 和位域	162
B.10 限制符: 访问 Volatile 对象	164
B.11 声明符: 最大数目	164
B.12 语句: switch 语句的限制	164
B.13 预处理命令	164
B.14 库函数	165

第二部分 C++ 语言参考手册

引言	175
第1章 词法约定	177
1.1 概述	177
1.2 单词	178
1.3 注释	179
1.4 标识符	180
1.5 C++ 关键字	181
1.6 标点符号	182
1.7 操作符	182
1.8 文字	185
1.8.1 整数常数	185
1.8.2 字符常数	187
1.8.3 浮点常数	189
1.8.4 串文字	190
第2章 基本概念	193
2.1 术语	193
2.2 声明与定义	194
2.2.1 声明	194

2.2.2	定义	195
2.3	作用域	195
2.3.1	声明点	196
2.3.2	隐藏名子	197
2.3.3	函数形参的作用域	198
2.4	程序和链接	199
2.4.1	链接的种类	199
2.4.2	具有文件作用域的名子的链接	199
2.4.3	具有类作用域的名子的链接	200
2.4.4	具有块作用域的名子的链接	200
2.4.5	没有链接的名子	200
2.4.6	对非 C++ 函数的链接	201
2.5	启动和结束	202
2.5.1	程序启动——main 函数	203
2.5.2	程序结束	206
2.5.3	对启动附加的考虑	207
2.5.4	对结束附加的考虑	207
2.6	存储类	209
2.6.1	自动	209
2.6.2	静态	209
2.6.3	寄存器	209
2.6.4	外部	210
2.6.5	对象的初始化	210
2.7	类型	212
2.7.1	基本类型	212
2.7.2	带尺寸的整数类型	213
2.7.3	派生类型	214
2.7.4	类型名	220
2.8	左值与右值	220
2.9	名空间	221
2.10	数字界限	221
2.10.1	整数界限	221
2.10.2	浮点界限	222
第 3 章	标准转换	224
3.1	整数升级	224
3.2	整数转换	225
3.2.1	将带符号的转为无符号的	225
3.2.2	将无符号的转为带符号的	226
3.2.3	标准转换	226

3.3	浮点转换	226
3.4	浮点和整数的转换	227
3.4.1	浮点到整数	227
3.4.2	整数到浮点	227
3.5	算术转换	227
3.6	指针转换	228
3.6.1	空指针	229
3.6.2	指向 void 类型的指针	229
3.6.3	指向对象的指针	229
3.6.4	指向函数的指针	229
3.6.5	指向类的指针	229
3.6.6	表达式	230
3.6.7	被 const 或 volatile 限定的指针	231
3.7	引用转换	231
3.8	指向成员的指针的转换	231
3.8.1	整常数表达式	231
3.8.2	指向基类成员的指针	232
第 4 章	表达式	233
4.1	表达式的种类	233
4.1.1	初等表达式	234
4.1.2	后缀表达式	235
4.1.3	使用单目操作符的表达式	243
4.1.4	使用双目操作符的表达式	253
4.1.5	使用条件操作符的表达式	263
4.1.6	常数表达式	264
4.1.7	使用显式类型转换的表达式	265
4.1.8	使用指向成员的指针操作符的表达式	268
4.2	表达式的语义	270
4.2.1	求值的顺序	270
4.2.2	表达式的注释	272
第 5 章	语句	274
5.1	概述	274
5.2	标号语句	275
5.2.1	配合 goto 语句使用标号	275
5.2.2	在 case 语句中使用标号	275
5.3	表达式语句	276
5.3.1	空语句	276
5.4	复合语句(块)	277
5.5	选择语句	277

5.5.1	if 语句	278
5.5.2	switch 语句	278
5.6	迭代语句	281
5.6.1	while 语句	281
5.6.2	do 语句	282
5.6.3	for 语句	282
5.7	Jump(跳转)语句	284
5.7.1	break 语句	284
5.7.2	continue 语句	285
5.7.3	return 语句	285
5.7.4	goto 语句	286
5.8	声明语句	286
5.8.1	自动对象的声明	286
5.8.2	静态对象的声明	288
5.9	异常处理	290
5.9.1	try, catch 及 throw 语句	290
5.9.2	构造的异常处理	295
第 6 章	声明	296
6.1	说明符	296
6.1.1	存储类说明符	297
6.1.2	函数说明符	299
6.1.3	typedef 说明符	302
6.1.4	friend 说明符	305
6.1.5	类型说明符	305
6.2	枚举声明	309
6.2.1	枚举符名字	312
6.2.2	枚举符常量的定义	312
6.2.3	转换和枚举类型	312
6.3	链接规范	313
6.4	模板规范	315
6.4.1	引用模板	316
6.4.2	函数模板	316
6.4.3	成员函数模板	317
6.4.4	显式实例化	317
6.4.5	与其它实现方法的区别	318
第 7 章	声明符	319
7.1	概述	319
7.2	类型名	321
7.3	抽象声明符	321

7.3.1	消除二义性	322
7.3.2	指针	323
7.3.3	引用	324
7.3.4	成员指针	329
7.3.5	数组	333
7.3.6	函数	336
7.3.7	缺省参数	342
7.4	函数定义	344
7.4.1	带有可变参数表的函数	345
7.5	初始化	346
7.5.1	初始化 const 对象的指针	347
7.5.2	非初始化对象	347
7.5.3	初始化静态成员	347
7.5.4	初始化聚集	348
7.5.5	初始化字符数组	350
7.5.6	初始化引用	351
第 8 章	类	353
8.1	概述	353
8.1.1	定义类类型	354
8.1.2	类类型对象	356
8.2	类名	357
8.2.1	声明和访问类名	358
8.2.2	typedef 语句和类	359
8.3	类成员	359
8.3.1	类成员声明语法	360
8.3.2	在成员中声明未定义长度的数组	362
8.3.3	类成员数据的存储	362
8.3.4	成员命名限制	363
8.4	成员函数	363
8.4.1	成员函数综述	364
8.4.2	this 指针	365
8.5	静态数据成员	368
8.6	联合	369
8.6.1	联合中的成员函数	369
8.6.2	作为类类型的联合	370
8.6.3	联合的成员数据	370
8.6.4	无名联合	370
8.7	位域	371
8.7.1	使用位域的限制	372

8.8	嵌套类声明	372
8.8.1	访问特权和嵌套类	373
8.8.2	嵌套类中的成员函数	373
8.8.3	友元函数和嵌套类	374
8.9	类作用域中的类型名	375
第9章	派生类	376
9.1	概述	376
9.1.1	单继承	376
9.1.2	多重继承	380
9.1.3	虚基类层次结构	380
9.1.4	类协议实现	381
9.1.5	基类	381
9.2	多重基类	382
9.2.1	虚基类	382
9.2.2	名字的二义性	384
9.3	虚函数	387
9.4	抽象类	390
9.4.1	使用抽象类的限制	391
9.5	作用域规则总结	392
9.5.1	二义性	392
9.5.2	全局名	392
9.5.3	名字和受限名	392
9.5.4	函数参数名	393
9.5.5	构造函数初始化	393
第10章	成员访问控制	394
10.1	控制对类成员的访问	394
10.2	访问说明符	394
10.3	基类的访问说明符	395
10.3.1	访问控制和静态成员	397
10.4	友元	398
10.4.1	友元函数	398
10.4.2	作为友元的类成员函数和类	399
10.4.3	友元声明	400
10.4.4	在类声明中定义友元函数	401
10.5	保护成员访问	401
10.6	对虚函数的访问	401
10.7	多重访问	402
第11章	特殊成员函数	403
11.1	构造函数	404

11.1.1	构造函数做什么	404
11.1.2	声明构造函数的规则	404
11.1.3	构造函数和数组	407
11.1.4	构造的次序	407
11.2	析构函数	407
11.2.1	声明析构函数	408
11.2.2	使用析构函数	408
11.2.3	析构的顺序	409
11.2.4	显式析构函数调用	410
11.3	临时对象	411
11.4	转换	411
11.4.1	转换构造函数	412
11.4.2	转换函数	414
11.5	new 和 delete 操作符	415
11.5.1	operator new 函数	415
11.5.2	处理内存不够的情况	417
11.5.3	operator delete 函数	420
11.6	使用特殊成员函数进行初始化	421
11.6.1	显式初始化	422
11.6.2	初始化数组	423
11.6.3	初始化静态对象	424
11.6.4	初始化基和成员	424
11.7	拷贝类对象	427
11.7.1	编译器生成的拷贝	428
11.7.2	按成员赋值和初始化	428
第 12 章	重载	430
12.1	概述	430
12.1.1	参数类型的区别	430
12.1.2	重载函数的限制	431
12.2	声明匹配	432
12.3	参数匹配	433
12.3.1	参数匹配和 this 指针	435
12.3.2	参数匹配和转换	435
12.4	重载函数的地址	438
12.5	重载操作符	439
12.5.1	操作符重载的通用规则	440
12.5.2	一元操作符	441
12.5.3	二元操作符	443
12.5.4	赋值	444

12.5.5	函数调用	445
12.5.6	下标	445
12.5.7	类成员访问	446
附录A	文法小结	448
A.1	Keywords	448
A.2	Expressions	448
A.3	Declarations	453
A.4	Declarators	456
A.5	Classes	457
A.6	Statements	459
A.7	Microsoft Extensions	460
附录B	Microsoft 特定的修饰符	461
B.1	基址寻址	461
B.1.1	--based	461
B.2	调用和命名约定修饰符	462
B.2.1	--cdecl	463
B.2.2	--fastcall	463
B.2.3	--stdcall	464
B.3	扩展存储类属性	464
B.3.1	扩展属性语法	464
B.3.2	thread 属性	465
B.3.3	naked 属性	467
B.3.4	dllexport 和 dllimport 属性	469
B.3.5	在 C++ 中使用 dllimport 和 dllexport	472
B.4	嵌入汇编器	473
附录C	表格	475

第三部分 预编译器参考手册

引言	483
第1章 预处理器	484
1.1 编译的阶段	484
1.2 预处理指令	485
1.2.1 #define 指令	486
1.2.2 #error 指令	487
1.2.3 #if, #elif, #else 和 #endif 指令	488
1.2.4 #ifdef 和 #ifndef 指令	491
1.2.5 #include 指令	491
1.2.6 #line 指令	493
1.2.7 空(Null)指令	494

1.2.8	#undef 指令	494
1.3	预处理操作符	494
1.3.1	字符串化符(#)	495
1.3.2	字符化符(# @)	496
1.3.3	符号粘贴符(# #)	496
1.4	宏	497
1.4.1	宏和 C++	497
1.4.2	预定义宏	498
第 2 章	Pragma 指令	499
2.1	C++ 编译器专有的编译指令	499
2.1.1	init_seg	499
2.1.2	pointer_to_member	500
2.1.3	vtordisp	501
2.2	C 和 C++ 编译器的编译指令	501
2.2.1	alloc_text	501
2.2.2	auto_inline	502
2.2.3	check_stack	502
2.2.4	code_seg	502
2.2.5	comment	503
2.2.6	data_seg	504
2.2.7	function	504
2.2.8	hdrstop	504
2.2.9	include_alias	505
2.2.10	inline_depth	506
2.2.11	inline_recursion	506
2.2.12	intrinsic	506
2.2.13	message	507
2.2.14	optimize	507
2.2.15	pack	508
2.2.16	setlocale	509
2.2.17	warning	509
附录A	语法总结	511
A.1	定义	511
A.2	约定	511
A.3	预处理器语法	511

第一部分

C语言参考手册



引 言

C语言是一种通用编程语言,并以高效率、低开销和良好的可移植性而闻名。而这些特性使它几乎优于任何一种编程语言,用C语言编程编写速度快,程序紧凑且易于与系统接口,因此C语言已被证明特别适用于系统编程。编写得好的C语言程序其运行速度通常和用汇编语言写的程序的运行速度一样快,且非常易于程序员阅读和维护。

C是一种十分灵活的编程语言,它将编程中的许多决定交给程序员去做。由于具有这一特性,C语言中的各种限制较少,例如在数据类型转换方面。虽然这一特性可以使你的编程工作更容易,但必须十分了解程序将怎样运行。本书提供C语言成份的信息和Microsoft实现的特性。本C语言语法源于ANSI X3.159—1989, *American National Standard for Information Systems-Programming Language-C*(以下称为ANSI C标准)。尽管这不是ANSI C标准的一部分。附录A中“C语言语法总结”将提供该语法和如何阅读与使用该语法的定义。

第一部分不讨论C++语言的编程。有关C++的内容请看本书第二部分《C++语言参考手册》。

注意 有关Microsoft产品的支持,看Introducing Visual C++。

ANSI 约 定

Microsoft C符合ANSI C标准所述的C语言标准。和联机参考一样,Microsoft C所做的扩充在本书的文字和语法中将予以标明。由于该扩充不是ANSI C标准的一部分,所以扩充的部分可能会限制程序在不同系统中的移植。在缺省的情况下允许Microsoft所做的扩充。如在编译中使用/Za选项则禁止该扩充,所有非ANSI C标准的代码都将被报错或警告。

本书的约定^①

本书使用如下印刷约定。

示 例	描 述
STDIO.H	表示文件名,段名,寄存器和操作系统命令中的项。
char, - alloca	黑体字表示关键字,操作符,语言特殊字符,库函数。在讨论语法时黑体字必须严格按所示的输入。 许多函数和常量以单下划线或双下划线开头。这是其名字必不可少的一部分,例如,声明常量 <code>--cplusplus</code> 被编译器所识别,必须输入开头的双下划线。

^① “本部分的约定”也适用于《C++语言参考手册》和《预编译器参考手册》——译者注。