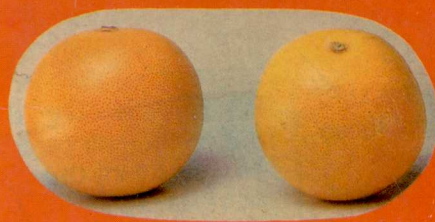


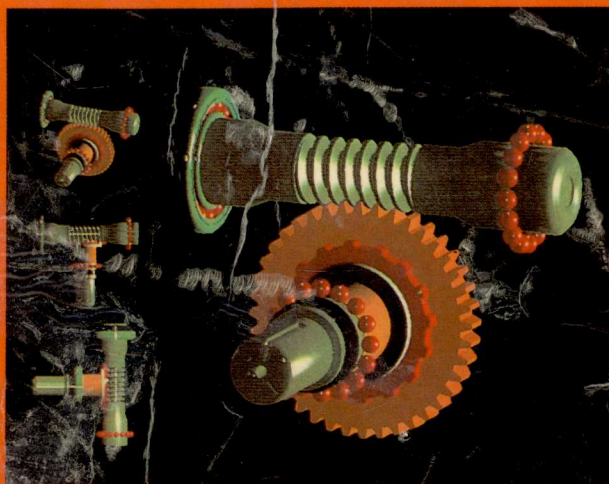
计算机 C/C++ 语言系列丛书

曾国庆 编写
黄志同



VISUAL C++

C 库函数详解



希望

学苑出版社

计算机 C/C++ 语言系列丛书

Visual C++ C 库函数详解

曾国庆 黄志同 编写
万 博 审校

学苑出版社

(京)新登字 151 号

内 容 简 介

本书按字母顺序着重介绍了 Visual C++ 的每个库函数,它们的功能、用法、原型所在的头文件、返回值、相关函数等信息。同样适用于 Microsoft C 5, Microsoft C 6, Microsoft C/C++ 7.0, Visual C++ 1.0, Visual C++ 1.5, Visual C++ 2.0, QuickC 2.5, QuickC for Windows, Turbo C 2.0, Turbo C++, Turbo C++ for Windows, Borland C++ 2.0, Borland C++ 3.0, Borland C++ 3.1, Borland C++ 4.0, Borland C++ 4.5 和 ANSI, 是各种 C 和 C++ 的 C 库函数的大荟萃。本书适合于所有用 C 编写 DOS 和 Windows 应用程序的读者。

欲购本书的用户可直接与北京海淀 8721 信箱书刊部联系,电话 2562329, 邮政编码 100080。

计算机 C/C++ 语言系列丛书

Visual C++ C 库函数详解

编 写: 曾国庆 黄志同
审 校: 万 博
责任编辑: 甄国亮
排 版: 万博图书创作社
出版发行: 学苑出版社 邮政编码:100036
社 址: 北京市海淀区万寿路西街,11 号
印 刷: 北京昌平建华印刷厂印刷
开 本: 787×1092 1/16
印 张: 22.50 **字数:** 544 千字
印 数: 1~5000 册
版 次: 1994 年 8 月北京第 1 版第 1 次
本册定价: 33.00 元
ISBN7-5077-0875-6/TP·24

学苑版图书印、装错误可随时退换

目 录

第一章 文件和目录管理	1
1.1 MS-DOS 文件系统	1
1.1.1 路径名	1
1.1.2 作为文件的设备	2
1.1.3 文件属性	2
1.1.4 文件句柄	3
1.2 基本文件的目录管理任务	3
1.2.1 改变驱动器和目录	5
1.2.2 改变文件允许	7
_access	9
_chdir	9
_chdrive	10
_chmod	11
_chsize	12
_filelength	12
_fstat	13
_fullpath	14
_getcwd	15
_getdcwd	16
_getdrive	16
_isatty	17
_locking	18
_makepath	19
_mkdir	19
_mktemp	20
remove	21
rename	21
_rmdir	22
_searchdev	22
_setmode	23
_splitpath	24
_stat	24
_umask	25

_unlink	26
第二章 I/O 流	27
2.1 流	27
2.1.1 缓冲型 I/O	27
2.1.2 文件类型	28
2.1.3 格式化和非格式化 I/O	28
2.1.4 在一个文件中的当前位置	29
2.1.5 预定义流	29
2.1.6 I/O 字符串	29
2.2 基本的 I/O 任务流	30
2.2.1 在一个文件中加入行数	32
2.3 一个简单的通讯录	34
2.4 显示错误信息的方便方法	41
clearerr	44
fclose	44
_fcloseall	45
_fdopen	
feof	4
ferror	48
fflush	48
fgetc	
_fgetchar	
fgetpos	51
fgets	51
_fileno	52
_flushall	53
fopen	53
fprintf	55
fputc	55
_fputc	56
fputs	57
fread	57
freopen	58
fscanf	59
fseek	60
fsetpos	60
_fsopen	61
ftell	62

.....	fwrite	63
.....	getc	64
.....	getchar	64
.....	gets	65
.....	_getw	66
.....	perror	66
.....	printf	67
.....	putc	70
.....	putchar	70
.....	puts	71
.....	_putw	71
.....	rewind	72
.....	_rmtmp	73
.....	scanf	73
.....	setbuf	76
.....	setvbuf	76
.....	_snprintf	77
.....	sprintf	78
.....	sscanf	79
.....	_tempnam	80
.....	tmpfile	80
.....	tmpnam	81
.....	ungetc	82
.....	vfprintf	82
.....	vprintf	84
.....	_vsnprintf	84
.....	vsprintf	85

第三章 低级 I/O 87

.....	3.1 低级 I/O 的基础	87
.....	3.1.1 文本和二进制模式	87
.....	3.1.2 操作系统使用的缓存	88
.....	3.1.3 文件句柄	88
.....	3.1.4 文件句柄的最大数目	88
.....	3.2 基本低级 I/O 任务	89
.....	3.2.1 复制文件	89
.....	3.2.2 使用 _dup, _dup2 重定向 stdout	92
.....	3.3 函数介绍	94
.....	_close	94

_commit	95
_creat	95
_dup	96
_dup2	97
_eof	98
_lseek	98
_open	99
_read	101
_sopen	101
_tell	102
_write	103
第四章 控制台和输入/输出(I/O口)	104
4.1 控制台和输入/输出(I/O口)基础	104
4.1.1 控制台和I/O任务	104
4.1.2 使用inp和outp函数产生声音	105
_cgets	107
_cprintf	108
_cputs	109
_cscanf	109
_getch	110
_getche	110
_inp	111
_inpw	112
_kbhit	112
_outp	113
_outpw	114
_putch	114
_ungetch	115
第五章 字符和数据转换	116
5.1 字符和数据转换的基础	116
5.1.1 区分字符	116
5.1.2 数据转换	117
5.2 字符和数字转换任务	117
5.2.1 把文本字符串转换成小写	117
5.2.2 简单的计算器	120
atof	121
atoi, atol	122

_atold	123
_ecvt, _fcvt	123
_gcvt	124
isalnum	125
isalpha	125
__isascii	126
iscntrl, isdigit, isgraph, islower, isprint, ispunct, isspace, isupper, isxdigit	126
__iscsym, __iscsymf	128
_itoa, _ltoa	128
strtod	129
strtol	130
_trtold	131
strtoul	132
__toascii	132
_tolower, tolower	133
_toupper, toupper	134
_ultoa	134
第六章 串和缓冲区处理	136
6.1 C语言中的串和缓冲区	136
6.1.1 声明串和缓冲区	136
6.1.2 词典式顺序	137
6.1.3 多字节和宽字符串	137
6.1.4 远缓冲区和串	137
6.2 基本的串和缓冲区处理任务	137
6.2.1 拷贝视频存储器	140
6.2.2 语法分析一行文字	142
mblen, _fmbllen	144
mbstowcs, _fmbstowcs	145
mbtowc, _fmbtowc	146
_memccpy, _fmemccpy	147
memchr, _fmemchr	147
memcmp, _fmemcmp	148
memcpy, _fmemcpy	149
_mmmicmp, _fmemicmp	
memmove, _fmemmove	
memset, _fmemset	
_movedata	

strcat, _fstrcat	153
strchr, _fstrchr	154
strcmp, _fstrcmp	155
_strcmpi	155
strcoll	156
strcpy, _fstrcpy	157
strcspn, _fstrcspn	158
_strdup, _fstrdup, _nfstrdup	158
_strerror	159
strerror	160
_stricmp, _fstricmp	161
strlen, _fstrlen	161
_strlwr, _fstrlwr	162
strncat, _fstrncat	163
strncpy, _fstrncpy	163
_strnicmp, _fstrnicmp	164
_strnset, _fstrnset	165
strpbrk, _fstrpbrk	166
strrchr, _fstrrchr	166
_strrev, _fstrrev	167
_strset, _fstrset	168
strspn, _fstrspn	169
strstr, _fstrstr	169
strtok, _fstrtok	170
_strupr, _fstrupr	171
strxfrm	172
_swab	173
wcstombs, _fwcstombs	173
wctomb, _fwctomb	174

第七章 搜索和排序 175

7.1 搜索和排序任务基础	175
7.1.1 查寻环境变量	175
bsearch	177
_lfind, _lsearch	178

8.1.1	Local Time, GMT 和 UTC	181
8.1.2	时间转换	182
8.2	基本的日期和时间管理任务	182
8.2.1	获取和打印当前日期和时间	183
8.2.2	打印一份月历	184
	asctime	189
	clock	190
	ctime	191
	difftime	191
	_ftime	192
	gmtime, localtime	193
	mktime	193
	strdate	194
	strftimeI	195
	_strtime	196
	time	197
	_tzset	197
	_utime	198
第九章	数学函数	200
9.1	浮点数	200
9.1.1	浮点数的存储格式	200
9.1.2	浮点变量的类型	200
9.1.3	浮点型程序包	201
9.1.4	数学函数中的错误	201
9.2	数学函数的类型	202
9.2.1	计算一项贷款的每月偿还钱数	205
9.2.2	计算 sines 和 cosines	206
	abs	207
	acos, acosl	207
	asin, asinl	208
	atan, atanl	209
	atan2, atan2l	209
	Bessel 函数	210
	_cabs, _cabsl	211
	ceil ceill	212
	_clear87, _control87	212
	cos, cosl	214
	cosh, coshl	214

_dieetombsbin, _dmsbintoieee	215
div	216
exp, expl	216
fabs, fabsl	217
_fieetombsbin, _fmsbintoieee	217
floor, floorl	218
fmod, fmodl	219
_fpreset	219
frexp, frexpl	220
_hypot, _hypotl	221
labs	221
ldexp, ldexpl	222
ldiv	222
log, log10, logl, log10l	223
_lrotl, _lrotr	224
_matherr, _matherrl	224
__max, __min	225
modf, modfl	226
pow, powl	226
rand	227
_rotl, _rotr	228
sin, sinl	228
sinh, sinhl	229
aqrt, aqrtl	230
srand	230
_status87	231
tan, tanl	231
tanh, tanhl	232

第十章 进程控制 233

10.1 管理进程	233
10.1.1 环境	233
10.1.2 信号 signals	234
10.1.3 可变长度参数列表	234
10.1.4 场所	234
10.2 基本进程控制任务	234
10.2.1 一个微型外壳	237
10.3 函数介绍	242
abort	242

assert	242
atexit, _fatexit	243
_cexit, _c_exit	244
_execl, _execle, _execlp, _execlpe;	
_execv, _execve, _execvp, _execvpe	244
exit	246
_exit	246
getenv	247
_getpid	248
localeconv	248
longjmp	249
_onexit, _fonexit	250
_putenv	251
raise	252
setjmp	252
setlocale	253
signal	254
_spawnl, _spawnle, _spawnlp,	
_spawnlpe, _spawnv, _spawnve	
_spawnvp, _spawnvpe	255
system	257
va_arg, va_end, va_start(ANSI 版本)	257
va_arg, va_end, va_start(UNIX 版本)	259
第十一章 内存管理	261
11.1 内存管理基础	261
11.1.1 分段内存寻址	261
11.1.2 内存模式	261
11.1.3 近指针, 远指针, 巨指针	262
11.1.4 __near, __far, __huge 关键字	262
11.1.5 堆	262
11.1.6 基 堆	263
11.1.7 虚拟存储器	264
11.2 基本的内存管理任务	264
11.2.1 分配内存和检测堆	266
_malloc	270
_bfreeseg	271
_bheapseg	271
calloc, _bcalloc, _fcalloc, _healloc, _ncalloc	272

_expand, _bexpand, _fexpand, _nexpand	273
free, _bfree, _ffree, _hfree, _nfree	274
_freect	275
_heapadd, _bheapadd	276
_heapchk, _bheapchk, _fheapchk, _nheapchk	276
_heapmin, _bheapmin, _fheapmin, _nheapmin	277
_heapset, _bheapset, _fheapset, _nheapset	278
_heapwalk, _bheapwalk, _fheapwalk, _nheapwalk	279
malloc, _bmalloc, _fmalloc, _nmalloc	280
_memavl	281
_memmax	282
_msize, _bmsize, _fmsize, _nmsize	282
realloc, _brealloc, _frealloc, _nrealloc	283
_stackavail	284
_vfree	285
_vheapinit	285
_vheapterm	286
_vload	287
_vlock	288
_vlockcnt	289
_vmalloc	289
_vmsize	290
_vrealloc	291
_vunlock	292

第十二章 DOS 和 BIOS 功能调用

12.1 访问 BIOS 和 DOS 服务例程	293
12.1.1 BIOS 服务例程	294
12.1.2 MS-DOS 功能	294
12.2 基本 DOS 和 BIOS 任务	296
12.2.1 目录列表	296
12.2.2 察看磁盘的物理扇区	299
_bdos	303
_bios_disk	304
_bios_equiplist	307
_bios_keybrd	308
_bios_memsize	309
_bios_printer	310
_bios_serialcom	311

_bios_timeofday	313
_chain_intr	314
_disable	314
_dos_allocmem	315
_dos_close	315
_dos_commit	316
_dos_create, _dos_creatnew	317
_dos_findfirst, _dos_findnext	318
_dos_freemem	319
_dos_getdate	320
_dos_getdiskfree	320
_dos_getdrive	322
_dos_getfileattr	322
_dos_getftime	323
_dos_gettime	324
_dos_getvect	325
_dos_keep	326
_dos_open	326
_dos_read	328
_dos_setblock	329
_dos_setdate	330
_dos_setdrive	330
_dos_setfileattr	331
_dos_setftime	332
_dos_settime	333
_dos_setvect	334
_dos_write	335
_dosexterr	336
_enable	337
_FP_OFF, _FP_SEG	337
_hardrerr, _hardresume, _hardretn	338
_int86, _int86x	339
_intdos, _intdosx	341
_segread	343

第一章 文件和目录管理

几乎所有的应用程序都要将数据和结果存储在文件中,以便能够检索且下次使用它们。为了有效、便利地使用文件,用户应了解在目录层次中如何组织文件,以及在C中如何观察文件。本章简要地描述了MS-DOS操作系统关于文件和目录方面的组织、文件和目录如何命名以及文件内容如何解释。通过阅读本章,读者将为更好地使用第二章“I/O流”及第三章“低级I/O”所介绍的流和低级输入输出函数(I/O)作好了准备,因为文件和目录管理任务一般优先于I/O。另外,本章还描述了具有其它处理功能的函数,如搜索一个特定文件或检查某一文件是否存在。

Visual C++包含了一套完整的关于文件和目录管理任务的补码函数,诸如创建和删除目录,查找文件,以及检测文件属性。本章简要介绍如何使用这些函数。本章最后一部分作为参考指南,列出了所有文件和目录管理函数。

1.1 MS-DOS 文件系统

与大多数操作系统,诸如UNIX和Digital Equipment Corporation的VMS相类似,MS-DOS将文件和目录组织为一个树型层结构。所有在同一硬盘或软盘中的文件和目录都在一个唯一的根目录下(由反斜杠\表示)。用户不必创建根目录,MS-DOS在用户格式化一软盘时或硬盘分区时(对硬盘的物理划分)已经创建了根目录。每一目录依次包含许多子目录和文件——这就是树型层次结构的组织方法。

1.1.1 路径名

由于文件系统可存于软盘或硬盘上,MS-DOS需用户指定一驱动器以便彻底标识一个文件。实际上,一个文件完整的路径名指定驱动器,而对于根目录上的所有目录也要指定驱动器。一个文件完整的路径名以驱动器名开始,后跟冒号":";然后接着是一个反斜杠表明是根目录。子目录名一个接一个出现在反斜杠后面。文件名是路径名中最后一部分。MS-DOS中文件名最多由八个字符组成,后跟点"."。然后是三个字符的扩展名。Visual C++库中包含诸如_fullpath, _makepath和splitpath的函数,来对路径名进行处理。

1.1.1.1 当前驱动器

完整的路径名对于标识一个文件在系统中的位置是必要的,但并不是任何时候都要提供完整的路径名。MS-DOS 的确有当前驱动器和当前工作目录的概念。如果在一路径名中没有指定的驱动器名,则引用的文件将假设是在当前驱动器中。同样,如果忽略目录名,表明用户引用的文件在当前工作目录中。

在 MS-DOS 下,一个 drive 指一设备——软盘或一硬盘驱动器。驱动器名的一般意义如下所述:

- 驱动器 A: 和 B: 分别指第一和第二个软盘驱动器。
- 驱动器 C: 至 Z: 表示硬盘驱动器。C 表示第一个硬盘驱动器。

要在用户程序中得到当前驱动器并且改变它,可以分别使用 `_getdrive` 和 `_chdrive` 函数。

1.1.1.2 当前工作目录

就如 MS-DOS 对当前驱动器进行跟踪那样,它对当前目录也如此。在用户 C 程序中,可以通过调用 `_getcwd` 函数来得到当前目录的完整路径名。

可以使用 DOS 命令 CD(改变目录)来改变当前目录,使用 MKDIR 或 MD(创建目录)命令,可以创建一新目录,反之,使用 RD(删除目录)命令可删除一空目录。Visual C++ 包含诸如 `_chdir`, `_mkdir` 和 `_mdir` 函数,可以在用户程序中使用它们执行上述操作。

1.1.2 作为文件的设备

虽然许多文件指的是存储在磁盘上的文件但是有些文件名指的是设备,诸如一打印机或一串行通讯口。此处介绍一些指定是设备的文件名:

- AUX 指第一个串行通讯口。
- COM1 和 COM2 分别指第一个和第二个串行通讯口。
- CON 指控制台——即键盘和显示屏幕的组合。
- LPT1, LPT2 和 LPT3 分别是第一、第二和第三并行打印口。
- NVL 是虚设备——将输出送至 NVL,不会产生任何效果。
- PRN 指第一并行打印口。

对于面向磁盘操作的程序或数据文件,用户不能使用上述文件名——不管是大写还是小写。

1.1.3 文件属性

除了文件名处,一个 DOS 文件还有下列重要特性,影响着 I/O 操作。

- 权限设置确定文件类型及它的存取特性——文件是只读的或者可读也可写。
- 翻译模式决定了在输入/输出操作过程中文件内容是如何被翻译的。

1.1.3.1 允许设置

一文件的允许设置指出对该文件允许的操作。MS-DOS 中,用户可对文件标识为只读或可读写(MS-DOS 中不支持提供只写文件)。用户可以使用 `_cdcmd` 函数来对一文件的允许设置进行设置。`_caccess` 函数用来对允许的文件确定存取类型。

注意

当使用函数诸如 `fopen` 打开一文件时,必须指定存取模式——即用户需打开文件的模式。例如,可以用诸如只读,只写或可读写的读写允许来打开文件。

1.1.3.2 翻译模式

当用户使用 `fopen` 函数打开文件时,除了指定读写存取外,还要指定文件内容如何被解释的方法,包含下列两种:

- 文本模式
- 二进制模式

在二进制模式中,文件中的每一字节都毫无更改地提供给用户程序。在文本模式中当用户程序从文件中读或写入至文件时,发生下列翻译和解释:

- 当使用一文件时,一回车换行符被翻译为一单一的换行符。另外,在读文件时 Ctrl+Z 被解释成文件尾(EOF)。
- 当写文件时,一新行被扩充成一回车换行符对,并且这些字符被写入到文件中去。

当打开文件时,可指定翻译模式。也可通过使用 `_setmode` 函数来更改一已打开文件的翻译模式。

1.1.4 文件句柄

虽然文件名在 DOS 命令中指定一个文件是最常用的普通方法,但在 C I/O 库中可通过其它方法来指定文件。正如将在第二章所述,当使用 `fopen` 函数打开文件时,将返回一指向 FILE 结构的指针——该指针在所有的后继 I/O 操作中指向该文件。同样道理,如果使用 `_open`, `_sopen`, 或 `_creat` 函数(参考第三章具体章节)来创建或打开文件时,这些函数将返回一整型文件标识符,称文件句柄。本章介绍了许多需要句柄作为参数的函数,诸如 `_chsize`, `_filelength`, `_fstat`, `_isatty` 和 `_locking`。

1.2 基本文件的目录管理任务

至此用户对 MS-DOS 文件系统已有了大致了解,可查阅表 1.1。此表按字母顺序列出了文件和目录管理函数。表 1.1 使用户对每个函数有一快速的大概了解,表 1.2 通过各函数执行的任务,列出所有函数。