

LINUX



- ❖ 内容全面：涵盖近400个Linux常用C函数
- ❖ 实例讲解：每个函数均以操作实例进行讲解
- ❖ 查询方便：提供功能索引和字母索引

C函数 实例速查手册

陈先在 张丽萍 编著



Linux



人民邮电出版社
POSTS & TELECOM PRESS

LINUX

C函数 实例速查手册

陈先在 张丽萍 编著

Linux



人民邮电出版社
北京

图书在版编目 (CIP) 数据

Linux C函数实例速查手册 / 陈先在, 张丽萍编著.
北京: 人民邮电出版社, 2009. 1
ISBN 978-7-115-19070-3

I. L… II. ①陈…②张… III. Linux操作系统—手册
IV. TP316.89-62

中国版本图书馆CIP数据核字 (2008) 第167340号

内 容 提 要

Linux 作为一个自由操作系统已经越来越受到业内人士的关注, 该环境下的开发者也日趋增多。开发 Linux 环境下的应用程序时, 需要使用大量的 Linux 函数。这些函数有的属于 Linux 操作系统的 API, 有的属于 C 语言的标准库函数。这些函数数目繁多, 如果强行记忆每一个函数的使用方法显然是不明智的, 因此读者需要一本 Linux 环境下的 C 语言函数速查手册。

本书详细地介绍了 Linux 环境下常用的 C 语言函数, 重点介绍每个函数的参数意义、返回值的意义以及函数使用时的注意事项。本书含有大量实例, 详细讲解每个函数的使用方法, 每个函数均给出了对应的应用实例, 方便读者理解函数并进行模拟实践。本书适合于正在学习 Linux 环境下 C 语言编程的用户阅读, 并可作为开发人员的参考手册。

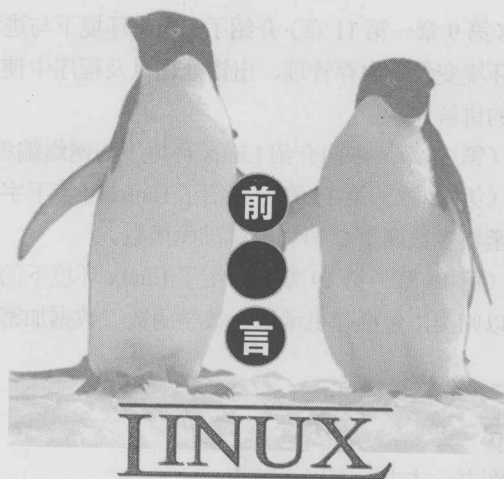
Linux C 函数实例速查手册

-
- ◆ 编 著 陈先在 张丽萍
责任编辑 屈艳莲
执行编辑 黄 焱
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京顺义振华印刷厂印刷
- ◆ 开本: 880×1230 1/32
印张: 23
字数: 686 千字 2009 年 1 月第 1 版
印数: 1-4 000 册 2009 年 1 月北京第 1 次印刷

ISBN 978-7-115-19070-3/TP

定价: 49.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223
反盗版热线: (010)67171154



作为一个 Linux 环境下的 C 程序开发人员，最需要掌握的是什么？毫无疑问，是掌握 Linux 环境下的 C 语言函数。然而令人感到尴尬的是，读者很难找到一本适合开发者使用的此类书，很多的书中缺乏函数使用的实例，读者往往了解了函数的作用却不知如何使用。

为此，笔者搜集了大量 Linux 环境下的 C 函数，并结合自己多年从事 Linux 环境下 C 语言应用程序开发的经验，编写了本书，使开发人员能够真正掌握 Linux 环境下 C 语言函数使用方法。

本书的内容安排

本书共分为 6 个部分，循序渐进地讲解了 Linux 环境下的 C 语言编程技术，从基本概念到具体实践、从系统函数接口的认识到具体操作。

第一部分（第 1 章～第 4 章）介绍了 Linux 环境下与文件操作相关的函数，包括 Linux 操作系统中的文件 I/O、文件系统的结构与组织、Linux 中的特殊文件的使用以及 Linux 环境下基于流的 I/O。

第二部分（第 5 章～第 8 章）介绍了 Linux 环境下与进程操作相关的函数，包括 Linux 操作系统中进程运行的环境、Linux 操作系统中对进程的控制、Linux 环境下进程之间的通信方法以及线程的相关操作。



第三部分（第 9 章～第 11 章）介绍了 Linux 环境下与进程环境相关的函数，包括对环境变量、内存管理、出错处理以及程序中使用正则表达式的操作的函数的讲解。

第四部分（第 12 章）专门介绍 Linux 环境下与网络编程相关的函数。

第五部分（第 13 章～第 15 章）介绍了 Linux 环境下字符、字符串处理函数。这些函数多数属于 C 语言的标准库函数。

第六部分（第 16 章～第 19 章）介绍了 Linux 环境下的其他函数。这些函数通常难以归类，包括随机函数、数学函数、数据加密函数和随机函数等。

本书的特点

相比同类图书，本书具有以下特点。

1. 内容详实，便于查询

本书以字典的形式呈现给读者，将 Linux 环境下的 C 语言常用函数按照功能进行分类，每一大类作为一章，并对该类的所有函数按函数名首字母进行排列，方便读者查询。

2. 概念准确，易于理解

作为一本 Linux 环境下 C 语言编程参考书，书中涉及的相关概念描述准确，让读者易于理解。本书对每一个概念都使用准确而且精炼的语言进行总结，以更加直观的方式呈现给读者。

3. 实例丰富，强调实践

为了让读者易于掌握 Linux 环境下的 C 语言函数，本书列举了大量实例进行讲解，通过这些实例，读者便可更加深入地理解相关概念，从而达到灵活使用 Linux 系统函数接口编写程序的目的。另外，通过对这些例子的学习，使读者熟悉函数的使用方法。

4. 代码规范，注释丰富

本书所涉及程序源代码层次清楚，语句简洁，体现了代码优美的原则，这样从一开始便给读者树立了良好的榜样，有利于读者养成良好的编写代码习惯。在严谨编写代码的同时，作者不忘对每一个代码中的难点进行讲解，帮助读者更加深入地了解程序的执行流程。

5. 讲解循序渐进，学习梯度设置科学

本书强调循序渐进的讲解方式，从前至后，层层推进，步步深入，学习难度适中，学习梯度设置科学，使读者非常容易掌握。

适合阅读本书的读者

- ☐ Linux 环境 C 语言编程初学者。
- ☐ Linux 系统函数接口研究人员。
- ☐ Linux 服务器程序开发人员。
- ☐ 嵌入式 Linux 程序开发人员。
- ☐ Linux 桌面应用开发的人员。
- ☐ 需要在 Linux 环境下进行毕业设计的师生。
- ☐ 想了解 Linux 环境下 C 语言编程的其他人员。

本书由陈先在、张丽萍组织编写，同时参与编写、资料整理和代码编写的有刘丹、刘冠军、罗思红、孙飞、王朋章、王石磊、王新平、文奇、吴琪、席国庆、谢超文、臧勇、张国强、张家春、郭玉敏、贺道权、胡斯登、江成海、姜海峰、李峥、利建昌、栗菊民、刘波等，在此一并表示感谢。

由于时间仓促，加之水平有限，书中不足之处在所难免，敬请读者批评指正。本书责任编辑的联系方式是 huangyan@ptpress.com.cn，欢迎来信交流。

编者

2008 年 10 月

目 录

第 1 章 初级 I/O 函数	1
1.1 close 函数: 关闭已经打开的文件.....	2
1.2 creat 函数: 创建一个文件.....	3
1.3 dup 函数: 复制文件描述符.....	5
1.4 dup2 函数: 复制文件描述符到指定的位置.....	6
1.5 fcntl 函数: 改变文件的状态.....	8
1.6 fsync 函数: 将缓冲区数据回写到磁盘文件.....	11
1.7 lseek 函数: 移动文件的读写位置.....	13
1.8 open 函数: 打开一个文件.....	15
1.9 read 函数: 读取文件的数据.....	17
1.10 sync 函数: 将所有打开的文件写回磁盘.....	18
1.11 write 函数: 输出内容到文件.....	21
第 2 章 文件管理操作函数	23
2.1 access 函数: 判断进程是否具有访问文件的权限.....	24
2.2 alphasort 函数: 按照字母顺序排序目录结构.....	26
2.3 chdir 函数: 改变当前的工作目录.....	28
2.4 chmod 函数: 改变文件的权限.....	30
2.5 chown 函数: 改变文件的所有者.....	33
2.6 chroot 函数: 改变进程的根目录.....	35
2.7 closedir 函数: 关闭目录文件.....	37
2.8 fchdir 函数: 使用文件描述符改变当前的工作目录.....	38
2.9 fchmod 函数: 改变一个已经打开的文件的权限.....	41
2.10 fstat 函数: 得到一个打开文件的状态.....	43



目录

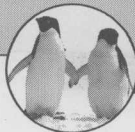
2.11	truncate 函数: 改变一个打开文件的大小	46
2.12	ftw 函数: 遍历目录树	48
2.13	get_current_dir_name 函数: 取得当前的工作目录	50
2.14	getcwd 函数: 取得进程的当前工作目录	52
2.15	lchown 函数: 改变符号链接文件的拥有者	54
2.16	link 函数: 建立一个硬链接	56
2.17	lstat 函数: 得到符号链接的文件状态	58
2.18	nftw 函数: 遍历目录树	62
2.19	opendir 函数: 打开目录文件	64
2.20	readdir 函数: 读取目录文件的内容	66
2.21	readlink 函数: 读取符号链接文件本身的内容	69
2.22	realpath 函数: 将相对目录路径转换成绝对路径	70
2.23	remove 函数: 删除文件或者目录	72
2.24	rename 函数: 更改文件的名称或位置	73
2.25	rewinddir 函数: 回卷目录文件	75
2.26	seekdir 函数: 文件定位函数	76
2.27	stat 函数: 取得文件状态	78
2.28	symlink 函数: 创建一个符号链接	80
2.29	telldir 函数: 取得目录文件的读取位置	83
2.30	truncate 函数: 改变文件大小	85
2.31	umask: 设置创建文件时使用的权限掩码	87
2.32	unlink 函数: 删除文件的一个硬链接	89
第3章 标准 I/O 函数		91
3.1	clearerr 函数: 清除错误标志	92
3.2	fclose 函数: 关闭流文件描述符	93
3.3	fdopen 函数: 转换文件描述符到文件指针	95
3.4	feof 函数: 文件结束判断函数	96
3.5	fflush 函数: 刷新流文件读写缓存	98
3.6	fgetc 函数: 从流文件中读取一个字符	100

3.7	fgetpos 函数: 返回流文件当前读写位置	101
3.8	fgets 函数: 从流文件中读出一行数据	103
3.9	fileno 函数: 获得流文件的文件描述符	104
3.10	fopen 函数: 打开文件	106
3.11	fputc 函数: 将字符写入流文件	108
3.12	fputs 函数: 将字符串写入流文件	110
3.13	fread 函数: 从流文件中读出数据	111
3.14	freopen 函数: 打开文件	113
3.15	fseek 函数: 移动流文件的读写位置	115
3.16	fsetpos 函数: 设置流文件当前读写位置	117
3.17	ftell 函数: 得到流文件的读取位置	119
3.18	fwrite 函数: 向流文件中写入数据	120
3.19	getc 函数: 从文件中读出一个字符	122
3.20	getchar 函数: 从标准输入中读入一个字符	124
3.21	gets 函数: 从标准输入中读入一行数据	125
3.22	mktemp 函数: 产生临时文件的文件名	126
3.23	putc 函数: 将字符写入文件	127
3.24	putchar 函数: 将字符输出到标准输出中	129
3.25	puts 函数: 将字符串输出在标准输出中	131
3.26	rewind 函数: 将流文件的读写位置移动到文件的起始位置	132
3.27	setbuf 函数: 设置文件缓冲区	134
3.28	setbuffer 函数: 设置文件缓冲区	136
3.29	setlinebuf 函数: 设置行缓冲模式	137
3.30	setvbuf 函数: 设置自定义文件缓冲区	139
3.31	tmpfile 函数: 建立临时的二进制文件	141
3.32	ungetc 函数: 将字符写回文件	142
第4章	格式化输入输出函数	145
4.1	fprintf 函数: 格式化输出到文件	146
4.2	fscanf 函数: 格式化从文件中读出数据	148



4.3	printf 函数: 格式化输出	149
4.4	scanf 函数: 格式化输入	150
4.5	snprintf 函数: 格式化字符串输入	152
4.6	sprintf 函数: 格式化字符串打印	154
4.7	sscanf 函数: 格式化字符串取值	156
4.8	vfprintf 函数: 可变参数输出函数	158
4.9	vscanf 函数: 格式化输入	160
4.10	vprintf 函数: 格式化输出函数	162
4.11	vscanf 函数: 格式化输入函数	164
4.12	vsnprintf 函数: 格式化字符串复制	166
4.13	vsprintf 函数: 格式化字符串复制	167
4.14	vsscanf 函数: 格式化字符串输入	169
第5章 进程控制		173
5.1	abort 函数: 进程异常终止函数	174
5.2	atexit 函数: 进程终止处理函数	175
5.3	execl 函数: 执行文件函数	177
5.4	execle 函数: 执行文件的函数	179
5.5	execlp 函数: 从 PATH 环境变量中查找文件并执行	181
5.6	execv 函数: 执行文件	182
5.7	execve 函数: 执行文件	184
5.8	execvp 函数: 执行文件	186
5.9	exit 函数: 结束进程运行的函数	188
5.10	_exit 函数: 结束进程运行的函数	190
5.11	fork 函数: 建立一个新的进程	192
5.12	getgid 函数: 取得实际进程组 ID	193
5.13	getegid 函数: 取得有效进程组 ID	195
5.14	getpid 函数: 取得有效组 ID	196
5.15	getppid 函数: 取得父进程的进程识别码	198
5.16	getpriority 函数: 取得进程执行优先级	199

5.17	longjmp 函数: 跳到原先 setjmp 存储的堆栈环境	201
5.18	on_exit 函数: 设置程序正常结束前调用的函数	202
5.19	setjmp 函数: 存储当前堆栈环境	204
5.20	setgid 函数: 设置进程的进程组 ID	205
5.21	setuid 函数: 设置进程组 ID	207
5.22	setpriority 函数: 设置程序进程优先级	208
5.23	siglongjmp 函数: 跳到原先 sigsetjmp 存储的堆栈环境	211
5.24	sigsetjmp 函数: 存储当前堆栈环境和屏蔽的信号集	213
5.25	system 函数: 执行 Shell 命令	216
5.26	vfork 函数: 建立一个新的进程	218
5.27	wait 函数: 等待子进程中断或结束	220
5.28	waitpid 函数: 等待子进程中断或结束	222
5.29	wait3 函数: 等待进程结束并且输出子进程统计信息	224
第 6 章	进程间通信函数	229
6.1	ftok 函数: 获得项目相关的 IPC 键值	230
6.2	msgctl 函数: 消息队列操作函数	231
6.3	msgget 函数: 创建或打开消息队列	234
6.4	msgrcv 函数: 读取消息队列	236
6.5	msgsnd 函数: 向消息队列中写入消息	238
6.6	semctl 函数: 信号量操作函数	240
6.7	semget 函数: 创建或打开信号量	243
6.8	semop 函数: 释放或获取信号量	244
6.9	shmat 函数: 导入共享内存	247
6.10	shmctl 函数: 共享内存操作函数	249
6.11	shmdt 函数: 将共享内存从程序中脱离	251
6.12	shmget 函数: 创建共享内存	253
6.13	mkfifo 函数: 创建 fifo 管道函数	255
6.14	pclose 函数: 关闭管道文件	257
6.15	pipe 函数: 创建匿名管道	258



6.16 popen 函数: 创建管道文件	260
第7章 信号函数	263
7.1 alarm 函数: 内核定时器函数	264
7.2 kill 函数: 信号发送函数	265
7.3 pause 函数: 进程暂停执行函数	267
7.4 raise 函数: 向进程自身发送信号函数	269
7.5 sigaction 函数: 高级信号处理方式设置函数	270
7.6 sigaddset 函数: 添加信号到信号集函数	274
7.7 sigdelset 函数: 删除信号集中对应信号函数	275
7.8 sigemptyset 函数: 清空信号集函数	277
7.9 sigfillset 函数: 填充信号集函数	279
7.10 sigismember 函数: 测试某个信号是否在信号集中	280
7.11 signal 函数: 信号处理方式设置函数	282
7.12 sigpause 函数: 暂停进程直到信号到来	285
7.13 sigpending 函数: 查询未决信号	287
7.14 sigprocmask 函数: 信号屏蔽函数	289
7.15 sigsuspend 函数: 暂停直到信号到来	292
7.16 sleep 函数: 让进程休眠一段时间	295
第8章 用户、组函数及环境变量函数	297
8.1 clearenv 函数: 删除所有环境变量的值	298
8.2 cuserid 函数: 获得用户名函数	299
8.3 endgrent 函数: 关闭系统组文件函数	300
8.4 endpwent 函数: 关闭系统口令函数	303
8.5 endutent 函数: 关闭 utmp 文件	305
8.6 fgetgrent 函数: 从文件中读取组数据	306
8.7 fgetpwent 函数: 从文件中读取密码格式数据	309
8.8 getegid 函数: 获得有效(effective gid)组识别符	312
8.9 getenv 函数: 取得环境变量的值	313

8.10	geteuid 函数: 获得有效(effective uid)用户识别符	314
8.11	getgid 函数: 获得组识别符	315
8.12	getgrent 函数: 从文件/etc/group 中读取组信息	316
8.13	getgrgid 函数: 从文件/etc/group 中查找组信息	319
8.14	getgmam 函数: 从文件/etc/group 中查找组信息	320
8.15	getgroups 函数: 返回当前用户所属的组列表	321
8.16	getlogin 函数: 获取当前用户的账号名称	323
8.17	getpw 函数: 获取指定 uid 的用户信息	324
8.18	getpwent 函数: 读取/etc/passwd 文件中的数据	326
8.19	getpwnam 函数: 以用户名获取用户信息	328
8.20	getpwuid 函数: 以用户 ID 获取用户信息	330
8.21	getuid 函数: 获取当前进程所属的用户 ID	331
8.22	getutent 函数: 获取登录信息	332
8.23	getutid 函数: 获取指定用户 ID 的登录信息	334
8.24	getutline 函数: 查找指定的登录信息	336
8.25	initgroups 函数: 初始化组信息	337
8.26	logwtmp 函数: 添加登录记录信息	339
8.27	putenv 函数: 改变环境变量的值	340
8.28	pututline 函数: 添加登录信息	342
8.29	setegid 函数: 设置有效组 ID	344
8.30	setenv 函数: 改变环境变量的值	345
8.31	seteuid 函数: 设置有效用户 ID	347
8.32	setfsuid 函数: 设置进程的文件系统的组 ID	348
8.33	setfsgid 函数: 设置进程的文件系统的用户 ID	349
8.34	setgid 函数: 设置进程的组 ID	351
8.35	setgrent 函数: 将文件 etc/group 的读写位置移动到起始位置	352
8.36	setgroups 函数: 设置当前进程的组识别符	354
8.37	setpwent 函数: 将文件 etc/passwd 的读写位置移动到起始位置	355
8.38	setregid 函数: 设置当前进程的真实以及有效组 ID	358
8.39	setreuid 函数: 设置当前进程的真实以及有效用户 ID	359



8.40	setuid 函数: 设置进程的用户 ID	360
8.41	setutent 函数: 将文件 utmp 的读写位置移动到起始位置	361
8.42	unsetenv 函数: 清除环境变量的值	363
8.43	updwtmp 函数: 将登录数据记录写入 utmp 文件	365
8.44	utmpname 函数: 设置文件 utmp 的绝对位置	366
第 9 章 错误处理与内存分配函数		369
9.1	brk 函数: 改变数据段大小	370
9.2	calloc 函数: 内存分配函数	371
9.3	ferror 函数: 文件错误判断	373
9.4	free 函数: 释放动态分配内存函数	374
9.5	getpagesize 函数: 获得操作系统内存页大小函数	376
9.6	malloc 函数: 动态内存分配函数	377
9.7	mmap 函数: 文件映射函数	379
9.8	munmap 函数: 释放映射的内存地址	381
9.9	perror 函数: 错误原因输出函数	382
9.10	realloc 函数: 内存截取函数	383
9.11	sbrk 函数: 增加数据空间	385
9.12	strerror 函数: 错误代码查询	387
第 10 章 日志函数、动态函数与正则表达式函数		389
10.1	closelog 函数: 关闭信息记录	390
10.2	dlclose 函数: 关闭动态库文件	391
10.3	dlderror 函数: 动态函数出错处理	393
10.4	dlopen 函数: 打开动态库文件	394
10.5	dlsym 函数: 在动态库文件中查找函数	396
10.6	openlog 函数: 打开记录信息	398
10.7	regcomp 函数: 编译正则表达式	400
10.8	regerror 函数: 编译错误判断函数	401
10.9	regex 函数: 执行正则匹配	403

10.10	regfree 函数: 释放正则模式串资源	405
10.11	syslog 函数: 输出记录信息	406
第 11 章 时间函数		409
11.1	asctime 函数: 字符串时间操作函数	410
11.2	clock 函数: 计算运行时间函数	411
11.3	ctime 函数: 字符串时间函数	412
11.4	difftime 函数: 计算时间差函数	414
11.5	ftime 函数: 取得系统时间	415
11.6	gettimeofday 函数: 取得系统当前时间	417
11.7	gmtime 函数: 日历时间格式函数	418
11.8	localtime 函数: 获得当前系统时间	420
11.9	mktime 函数: 系统时间转换函数	422
11.10	settimeofday 函数: 系统时间设置函数	423
11.11	strftime 函数: 时间格式化输出函数	425
11.12	time 函数: 获得系统当前时间	427
11.13	tzset 函数: 设置系统时区变量 tzname	428
第 12 章 socket 相关函数		431
12.1	accept 函数: 处理 socket 请求函数	432
12.2	bind 函数: 将 socket 描述符与一个套接口绑定	434
12.3	connect 函数: 与远程主机连接	436
12.4	endprotoent 函数: 结束网络协议的读取函数	439
12.5	endservent 函数: 关闭文件/etc/ services	440
12.6	gethostbyaddr 函数: 由 IP 获得主机信息	441
12.7	gethostbyname 函数: 由主机名获得主机信息	443
12.8	getprotobyname 函数: 由协议名获取协议数据	445
12.9	getprotobynumber 函数: 由协议编号获取协议数据	446
12.10	getprotoent 函数: 读取文件/etc/ rotoent 中的网络协议数据	447
12.11	getservbyname 函数: 获得网络服务协议数据	449



12.12	getservbyport 函数: 获得网络服务协议数据	450
12.13	getservent 函数: 读取主机网络服务数据	452
12.14	getsockopt 函数: 获得指定 socket 描述符的状态	453
12.15	herror 函数: 错误原因输出函数	455
12.16	hstrerror 函数: 错误代码查询	456
12.17	htonl 函数: 地址字节顺序转换	457
12.18	htons 函数: 地址字节顺序转换	459
12.19	inet_addr 函数: 网络地址类型转换	460
12.20	inet_aton 函数: 网络地址类型转换	461
12.21	inet_ntoa 函数: 网络地址类型转换	463
12.22	listen 函数: 网络 socket 监听	464
12.23	ntohl 函数: 地址字节顺序转换	466
12.24	ntohs 函数: 地址字节顺序转换	468
12.25	recv 函数: 接收消息	469
12.26	recvfrom 函数: 接收消息	471
12.27	recvmsg 函数: 多缓冲读取数据	474
12.28	send 函数: 向远程主机发送数据	477
12.29	sendto 函数: 发送数据到远程主机	479
12.30	setprotoent 函数: 打开网络协议文件	481
12.31	setservent 函数: 打开网络服务文件	483
12.32	setsockopt 函数: 设置 socket 描述符的状态	484
12.33	shutdown 函数: 结束 socket 套接字	486
12.34	socket 函数: 创建网络套接字	488
第 13 章 字符测试函数		491
13.1	isalnum 函数: 判断字符是否为英文字母或数字	492
13.2	isalpha 函数: 判断字符是否为英文字母	493
13.3	isascii 函数: 判断字符是否为 ASCII 字符	495
13.4	isblank 函数: 判断字符是否为空白字符	497
13.5	isctrl 函数: 判断字符是否为 ASCII 的控制字符	498

13.6	isdigit 函数: 判断字符是否为阿拉伯数字	500
13.7	isgraph 函数: 判断字符是否为可打印字符	501
13.8	islower 函数: 判断字符是否为小写英文字母	503
13.9	isprint 函数: 判断字符是否为可打印字符	504
13.10	isspace 函数: 判断字符是否为空格字符	506
13.11	ispunct 函数: 判断字符是否为标点符号	507
13.12	isupper 函数: 判断字符是否为大写英文字母	509
13.13	isxdigit 函数: 判断字符是否为十六进制数字	510
第 14 章	数据转换函数	513
14.1	atof 函数: 将字符串转换成浮点数	514
14.2	atoi 函数: 将字符串转换成整数	515
14.3	atol 函数: 将字符串转换成长整数	517
14.4	ecvt 函数: 将浮点数转换成字符串	519
14.5	fcvt 函数: 将浮点数转换成字符串	521
14.6	gcvt 函数: 将浮点数转换成字符串	523
14.7	strtod 函数: 将字符串转换成浮点数	524
14.8	strtol 函数: 将字符串转换成长整数	526
14.9	strtoul 函数: 将字符串转换成无符号长整数	529
14.10	toascii 函数: 将整数转换成合法的 ASCII 字符	531
14.11	tolower 函数: 将大写字母转换成小写字母	532
14.12	toupper 函数: 将小写字母转换成大写字母	533
第 15 章	字符串处理函数	535
15.1	bcmp 函数: 比较两块内存中的内容	536
15.2	bcopy 函数: 复制内存中的内容	538
15.3	bzero 函数: 将一块内存内容全清零	539
15.4	bfs 函数: 在一个整数中查找第一个值为真的位	541
15.5	index 函数: 查找字符串中第一个出现的指定字符	542
15.6	memccpy 函数: 复制内存中的内容	544