第 1 部分：出错处理
第 2 部分：文件 IO
第 3 部分：文件和目录
第 4 部分：I/O 库
第 5 部分：系统数据文件和信息
第 6 部分：进程环境
第 7 部分：进程控制
第 8 部分：进程关系
第 9 部分：信号
第 10 部分：线程
第 11 部分：线程控制
第 12 部分：守护进程
第 13 部分：高级 IO
第 14 部分：进程间通信
第 15 部分：套接字
第 16 部分：终端 I/O
第 17 部分：伪终端

出错处理：

1. 函数名： strerror ()
   功 能 ：返回指向错误信息字符串的指针
2. 函数名： perror ()
   功 能 ：将上一个函数发生错误的原因输出到标准设备 (stderr)。它首先输出由 msg 指向的字符串，然后是一个冒号，一个空格，接着是对应于 errno 值的出错信息，最后是一个换行符。

文件 IO：

1. 函数名：open()
   功 能 ：打开文件
2. 函数名：creat()
   功 能 ：创建一个新文件
3. 函数名：close()
   功 能 ：关闭一个打开的文件
4. 函数名：lseek()
   功 能 ：显式地为一个打开的文件设置其偏移
5. 函数名：read()
   功 能 ：从打开的文件中读数据
6. 函数名：write()
   功 能 ：向打开的文件写数据
7. 函数名：pread()
   功 能 ：带偏移量地原子的从文件中读取数据。相当于顺序调用 lseek 和 read，但无法中断其定位的读操作。
8. 函数名：pwrite()
   功 能 ：带偏移量地写数据到文件中
9. 函数名：dup()
   功 能 ：复制一个现存的文件的描述，新文件描述符一定是当前可用文件描述中的最小数值。
10. 函数名：dup2()
    功 能 ：复制一个现存的文件的描述，可以用 fd2 参数指定新的描述符数值。是个原子操作。
11. 函数名：sync()
    功 能 ：将缓冲区数据写回存储设备，它并不等待实际写磁盘操作结束。

12. 函数名：fsync()
    功 能 ：将缓冲区数据写回存储设备，只对单一文件起作用，并等待实际写磁盘操作结束。
13. 函数名：fdatasync()
    功 能 ：将缓冲区数据写回存储设备，只影响文件的数据部分。
14. 函数名：fcntl()
    功 能 ：改变已打开文件的性质
15. 函数名：ioctl()
    功 能 ：I/O 操作的杂物符，不能用本章其他函数表示的 I/O 操作通常都可以使用 I/O 操作表示。

文件和目录：

1. 函数名：stat()
   功 能 ：返回与此命名文件相关的信息结构
2. 函数名：fstat()
   功 能 ：获取已在描述符 fildes 上打开文件的有关信息。
3. 函数名：lstat()
   功 能 ：返回该符号链接的有关信息。
4. 函数名：access()
   功 能 ：确定文件的访问权限。
5. 函数名：umask()
   功 能 ：为进程设置文件模式创建屏蔽字，并返回以前的值。
6. 函数名：chmod()
   功 能 ：在指定文件上更改现有文件的访问权限。
7. 函数名：fchmod()
   功 能 ：对已打开的文件更改文件的访问权限。
8. 函数名：chown()
   功 能 ：更改路径参数 path 指定的文件的用户 ID 和组 ID。
9. 函数名：fchown()
   功 能 ：更改参数 fd 指定的用户 ID 和组 ID。
10. 函数名：lchown()
    功 能 ：更改符号链接本身的用户 ID 和组 ID，而不是符号链接指向的文件。
11. 函数名：truncate()
    功 能 ：将参数 path 指定的文件大小改为参数 length 指定的大小。 如果原来的文件大小比参数 length 大，则超过的部分会被删除。
12. 函数名：ftruncate()
    功 能 ：将参数 fd 指定的文件大小改为参数 length 指定的大小。 如果原来的文件大小比参数 length 大，则超过的部分会被删除。
13. 函数名：link()
    功 能 ：创建一个指向现有文件的链接。
14. 函数名：unlink()
    功 能 ：删除一个现有的目录项。只有当文件的链接计数达到 0 时，文件的内容才可被删除。
15. 函数名：remove()
    功 能 ：解除对一个文件或目录的链接。对于文件，remove 的功能与 unlink 相同。对于目录，remove 的功能与 rmdir 相同。
16. 函数名：rename()
    功 能 ：为文件或目录更名。
17. 函数名：symlink()
    功 能 ：创建一个符号链接。
18. 函数名：readlink()
    功 能 ：打开符号链接本身。

19. 函数名：mkdir()
    功 能 ：创建目录。
20. 函数名：rmdir()
    功 能 ：删除目录。
21. 函数名：opendir()
    功 能 ：打开目录，在失败的时候返回一个空的指针。
22. 函数名：readdir()
    功 能 ：读取目录。
23. 函数名：rewinddir()
    功 能 ：重设读取目录的位置为开头位置
24. 函数名：closedir()
    功 能 ：closedir()关闭参数 dir 所指的目录流。
25. 函数名：telldir()
    功 能 ：记录目录流的位置。
26. 函数名：seekdir()
    功 能 ：设置目录流目前的读取位置
27. 函数名：chdir()
    功 能 ：更改当前的工作目录，和 fchdir 比参数不同。
28. 函数名：fchdir()
    功 能 ：更改当前的工作目录，和 chdir 比参数不同。
29. 函数名：getcwd()
    功 能 ：获取当前的工作目录。

I/O 库：

1. 函数名：fwide()
   功 能 ：设置流的定向（流的定向决定了所读、写的字符是单字节还是多字节）。
2. 函数名：setbuf()
   功 能 ：打开或者关闭缓冲区。
3. 函数名：setvbuf()
   功 能 ：使用 setvbuf 可以精确地指定所需的缓冲类型，使用 mode 参数实现。
4. 函数名：fflush()
   功 能 ：冲洗流中的信息，该函数通常用于处理磁盘文件。
5. 函数名：fopen()
   功 能 ：打开一个指定的文件。
6. 函数名：freopen()
   功 能 ：在一个指定的流上打开一个指定的文件，如流已经打开，则先关闭该流。
7. 函数名：fdopen()
   功 能 ：获取一个现有的文件描述符。
8. 函数名：fclose()
   功 能 ：关闭一个打开的流。
9. 函数名：getc()
   功 能 ：一次读一个字符，可以实现为宏。
10. 函数名：fgetc()
    功 能 ：一次读一个字符，不能实现为宏。
11. 函数名：getchar()
    功 能 ： 一次读一个字符。
12. 函数名：ferror()
    功 能 ：检查文件在用各种输入输出函数进行读写时是否出错。 如 ferror 返回值为 0 表示未出错，否则表示有错。
13. 函数名：feof()
    功 能 ：判断文件是否处于文件结束位置，如文件结束，则返回值为 1，否则为 0。
14. 函数名：clearerr()

功 能 ：用于清除出错标志和文件结束标志，使它们为 0 值。
15. 函数名：ungetc()
功 能 ：从流中读取数据后，可以调用 ungetc 将字符再压送回流中。
16. 函数名：putc()
功 能 ：getc 的输出函数，可以实现为宏。
17. 函数名：fputc()
功 能 ：fgetc 的输出函数，不能实现为宏。
18. 函数名：putchar()
功 能 ：getchar 的输出函数。
19. 函数名：fgets()
功 能 ：每次从指定的流中输入一行。
20. 函数名：gets()
功 能 ：每次从标准输出中输入一行。由于 gets()无法知道字符串的大小，必须遇到换行字符或文件尾才会结束输入，因此容易造成缓存溢出的安全性问题。建议使用 fgets()取代。
21. 函数名：fputs()
功 能 ：每次提供一行输出。对应 fgets()。
22. 函数名：puts()
功 能 ：每次提供一行输出。对应 gets()。
23. 函数名：fread()
功 能 ：从一个文件流中读数据。
24. 函数名：fwrite()
功 能 ：向文件写入一个数据块。
25. 函数名：ftell()
功 能 ：得到文件位置指针当前位置相对于文件首的偏移字节数。
26. 函数名：fseek()
功 能 ：设置偏移位置。
27. 函数名：rewind()
功 能 ：将参数中的指针重新指向流的开头。
28. 函数名：ftello()
功 能 ：除函数返回值类型不同外，其与 ftell()相同。
29. 函数名：fseeko()
功 能 ：除参数类型不同外，其与 fseek()相同。
30. 函数名：fgetpos()
功 能 ：依据当前文件的句柄，获取当前访问指针位置信息。
31. 函数名：fsetpos()
功 能 ：将文件指针定位在 pos 指定的位置上。
32. 函数名：printf()
功 能 ：将格式化数据写到标准输出。
33. 函数名：fprintf()
功 能 ：将格式化数据写到标准流。
34. 函数名：sprintf()
功 能 ：将格式化数据写到参数数组 buf 中，在数据末尾自动加一个 null 字节。
35. 函数名：snprintf()
功 能 ：sprintf()可能会造成 buf 的缓冲区溢出，为此，引入 snprintf 函数，超出数组的数据被丢弃。
36. 函数名：vprintf()、vfprintf()、vsprintf()、vsnprintf()
功 能 ：类似于上面 4 种，只是可变参数表（…）变成了 arg。
37. 函数名：scanf()
功 能 ：按用户指定的格式从键盘上把数据输入到指定的变量之中。
38. 函数名：fscanf()
功 能 ：从一个流中执行格式化输入，fscanf 遇到空格和换行时结束，注意空格时也

结束。这与 fgets 有区别，fgets 遇到空格不结束。

39. 函数名：sscanf()
    功 能 ：从一个字符串中读进与指定格式相符的数据。
40. 函数名：vscanf()、vfscanf()、vsscanf()
    功 能 ：类似于上面 3 种，只是可变参数表（…）变成了 arg。
41. 函数名：fileno()
    功 能 ：每个标准 I/O 流都有一个与其相关联的文件描述符，调用 fileno()可以获得其文件描述符。
42. 函数名：tmpnam()
    功 能 ：产生一个与现有文件名不同的一个有效路径名字符串。
43. 函数名：tmpfile()
    功 能 ：创建一个临时二进制文件，在关闭该文件或程序结束时将自动删除这种文件。
44. 函数名：tempnam()
    功 能 ：是 tmpnam()的一个变体，允许调用者为所产生的路径名指定目录和前缀。
45. 函数名：mkstemp()
    功 能 ：类似于 tmpfile()，但该函数返回的不是文件指针，而是临时文件的打开文件描述符。mkstemp()创建的临时文件不会自动被删除。

系统数据文件和信息：

1. 函数名：getpwuid()
   功 能 ：通过用户的 uid 查找用户的 passwd 数据。
2. 函数名：getpwnam()
   功 能 ：通过用户的登录名查找用户的 passwd 数据。
3. 函数名：getpwent()
   功 能 ：从密码文件（/etc/passwd）中读取一项用户数据，该用户的数据以 passwd 结构返回。
4. 函数名：setpwent()
   功 能 ：将 getpwent()的读写地址指回密码文件开头。
5. 函数名：endpwent()
   功 能 ：与 getpwnam 配套使用。使用 getpwnam()查看完口令文件后，一定要调用 endpwent()关闭这些文件。
6. 函数名：getspnam()
   功 能 ：通过用户的登录名查找用户的阴影口令文件数据。（某些系统将加密口令存放在另一个称为阴影口令（shadow password）的文件中）由于阴影文件保存的是密码，所以没有 getspXid()之类的函数。
7. 函数名：getspent()
   功 能 ：从阴影口令文件中读取一项用户数据。
8. 函数名：setspent()
   功 能 ：将 getpwent()的读写地址指回文件开头。
9. 函数名：endspent()
   功 能 ：与 getspnam 配套使用。使用 getspnam()查看完口令文件后，一定要调用 endspent()关闭这些文件。
10. 函数名：getgrgid()
    功 能 ：依参数 gid 指定的组识别码逐一搜索组文件，找到时便将该组的数据以 group 结构返回。
11. 函数名：getgrnam()
    功 能 ：依参数指定的组名称逐一搜索组文件，找到时便将该组的数据以 group 结构返回。
12. 函数名：getgrent()
    功 能 ：从组文件（/etc/group）中读取一项组数据，该数据以 group 结构返回。
13. 函数名：setgrent()
    功 能 ：打开组文件（若它尚未打开）并反绕（反绕即将指针设为开头位置）它。

14. 函数名：endgrent()
    功 能 ：关闭组文件。和 getgrent()配套使用。
15. 函数名：getgroups()
    功 能 ：用来取得目前用户所属的附加组。
16. 函数名：setgroups()
    功 能 ：设置附加组
17. 函数名：initgroups()
    功 能 ：initgroups()用来从组文件（/etc/group）中读取一项组数据，若该组数据的成员中有参数 user 时，便将参数 group 组识别码加入到此数据中。
18. 函数名：uname()
    功 能 ：返回与当前主机和操作系统有关的信息。
19. 函数名：gethostname()
    功 能 ：返回主机名，该名字就是 TCP/IP 网络上主机的名字。
20. 函数名：time()
    功 能 ：返回当前时间和日期。
21. 函数名：gettimeofday()
    功 能 ：获得当前精确时间（1970 年 1 月 1 日到现在的时间）。与 time()相比，gettimeofday()提供了更高的分辨率（最高为微秒级）。
22. 函数名：gmtime()
    功 能 ：把日期和时间转换为格林威治(GMT)时间的函数。
23. 函数名：localtime()
    功 能 ：把从 1970-1-1 零点零分到当前时间系统所偏移的秒数时间转换为本地时间，而 gmtime 函数转换后的时间没有经过时区变换，是 UTC 时间 。
24. 函数名：mktime()
    功 能 ：将时间转换为自 1970 年 1 月 1 日以来持续时间的秒数，发生错误时返回-1。
25. 函数名：asctime()
    功 能 ：把 timeptr 指向的 tm 结构体中储存的时间转换为字符串字符串格式返格式为：Www Mmm dd hh:mm:ss yyyy。
26. 函数名：ctime()
    功 能 ：把日期和时间转换为字符串。
27. 函数名：strftime()
    功 能 ：根据区域设置格式化本地时间/日期，函数的功能将时间格式化，或者说格式化一个时间字符串。

进程环境：

1. 函数名：exit()
   功 能 ：正常终止一个进程。终止前先执行一些清理处理（包括调用执行各终止处理程序等）。由 ISO C 说明。
2. 函数名：_Exit()
   功 能 ：正常终止一个进程。立即进入内核，不做清理处理。由 ISO C 说明。
3. 函数名：_exit()
   功 能 ：正常终止一个进程。立即进入内核，不做清理处理。由 POSIX.1 说明。
4. 函数名：atexit()
   功 能 ：注册清理处理函数。
5. 函数名：malloc()
   功 能 ：分配指定字节数的存储区。
6. 函数名：calloc()
   功 能 ：为指定数量具指定长度的对象分配存储空间。
7. 函数名：realloc()
   功 能 ：更改以前分配区的长度（增加或减小）。
8. 函数名：free()
   功 能 ：释放存储空间。

9. 函数名：getenv()
   功 能 ：获取环境变量值。
10. 函数名：putenv()
   功 能 ：改变或增加环境变量的内容，把字符串加到当前环境中。
11. 函数名：setenv()
   功 能 ：改变或增加环境变量。通过此函数并不能添加或修改 shell 进程的环境变量，或者说通过 setenv 函数设置的环境变量只在本进程，而且是本次执行中有效。如果在某一次运行程序时执行了 setenv 函数，进程终止后再次运行该程序，上次的设置是无效的，上次设置的环境变量是不能读到的。
12. 函数名：unsetenv()
   功 能 ：删除环境变量。
13. 函数名：setjmp()
   功 能 ：非局部跳转语句，在希望返回到的位置调用 setjmp，setjmp 保存当前的寄存器里面的内容，longjmp 是恢复这些内容。longjmp 返回 setjmp 程序当前的状态。
14. 函数名：longjmp()
   功 能 ：非局部跳转语句，在希望返回到的位置调用 setjmp，setjmp 保存当前的寄存器里面的内容，longjmp 是恢复这些内容。longjmp 返回 setjmp 程序当前的状态。
15. 函数名：getrlimit()
   功 能 ：获取资源限制。（每个进程都有一个资源限制）
16. 函数名：setrlimit()
   功 能 ：设置资源限制。

进程控制：

1. 函数名：getpid()
   功 能 ：获取进程 ID。
2. 函数名：getppid()
   功 能 ：获取进程的父进程 ID。
3. 函数名：getuid()
   功 能 ：获取实际用户 ID。
4. 函数名：geteuid()
   功 能 ：获取有效用户 ID。
5. 函数名：getgid()
   功 能 ：获取组 ID。
6. 函数名：getegid()
   功 能 ：获取有效组 ID。
7. 函数名：fork()
   功 能 ：创建一个新进程。子进程返回 0，父进程返回子进程 ID。
8. 函数名：vfork()
   功 能 ：创建一个新进程，和 fork 不同的是 vfork()并不将父进程的地址空间完全复制到子进程，而且 vfork()保证子进程先运行。
9. 函数名：wait()
   功 能 ：阻塞进程。
10. 函数名：waitpid()
   功 能 ：阻塞进程， waitpid()有一个选项可使调用者不阻塞。
11. 函数名：exec 函数族
   功 能 ：用 fork()可以创建新进程，用 exec 可以执行新程序。
12. 函数名：setuid()
   功 能 ：设置实际用户 ID 和有效用户 ID。
13. 函数名：setgid()
   功 能 ：设置实际组 ID 和有效组 ID。

▲延伸阅读：实际用户 ID 就是当前计算机用户 ID，有效用户 ID 就是文件拥有者 ID。保存的设置用户 ID 是有效 ID 的副本，当要临时改变有效用户时，该 ID 就是用来保存

更改前的有效用户 ID，以便恢复数据。

14. 函数名：setreuid()
    功 能 ：设置实际用户 ID 及有效的用户 ID。
15. 函数名：setregid()
    功 能 ：设置实际组 ID 及有效组 ID。
16. 函数名：seteuid()
    功 能 ：设置有效用户 ID。
17. 函数名：setegid()
    功 能 ：设置有效组 ID。
18. 函数名：system()
    功 能 ：执行参数中的 shell 命令。
19. 函数名：times()
    功 能 ：获得进程及已终止子进程的墙上时钟时间、用户 CPU 时间和系统 CPU 时间。

进程关系：

1. 函数名：getpgrp()
   功 能 ：获取调用进程的进程组 ID。
2. 函数名：getpgid()
   功 能 ：类似于 getpgrp()，获取调用进程的进程组 ID，不同的是 getpgid()带个参数。
3. 函数名：setpgid()
   功 能 ：设置进程组 ID。
4. 函数名：setsid()
   功 能 ：建立一个新会话。
   ▲延伸阅读：会话：一个或多个进程组的集合
   开始于用户登录
   终止与用户退出
   此期间所有进程都属于这个会话期
5. 函数名：getsid()
   功 能 ：获取调用进程会话进程的进程组 ID。
6. 函数名：tcgetpgrp()
   功 能 ：返回前台进程组的进程组 ID。
7. 函数名：tcsetpgrp()
   功 能 ：设置前台进程组 ID。
8. 函数名：tcgetsid()
   功 能 ：获取会话首进程的会话 ID。

信号：

1. 函数名：signal()
   功 能 ：用于通知运行时系统，当某种特定的“软件中断”发生时调用特定的程序。
2. 函数名：kill()
   功 能 ：将信号发送给进程或进程组。
3. 函数名：raise()
   功 能 ：将信号发送给进程或进程组，允许进程向自身发送信号。
4. 函数名：alarm()
   功 能 ：设置一个计时器，在将来某个指定的时间该计时器会超时。
5. 函数名：pause()
   功 能 ：使调用进程挂起直到捕捉到一个信号。
6. 函数名：sigemptyset()
   功 能 ：清空信号集。
7. 函数名：sigfillset()
   功 能 ：填满信号集，使其包括所有信号。
8. 函数名：sigaddset()

功 能 ：添加信号到信号集
9. 函数名：sigdelset()
功 能 ：删除信号集
10. 函数名：sigismember()
功 能 ：判断信号是否在信号集里。
11. 函数名：sigprocmask()
功 能 ：检测或更改信号屏蔽字，或都在一个步骤中同时执行这两个操作。
12. 函数名：sigpending()
功 能 ：返回在送往进程的时候被阻塞挂起的信号集合。
13. 函数名：sigaction()
功 能 ：检查或修改与指定信号相关联的处理动作，查询或设置信号处理方式。
14. 函数名：sigsetjmp()
功 能 ：类似 setjmp()，sigsetjmp()增加了一个参数。
15. 函数名：siglongjmp()
功 能 ：类似 longjmp()。
16. 函数名：sigsuspend()
功 能 ：接收到某个信号之前，临时用 mask 替换进程的信号掩码，并暂停进程执行，直到收到信号为止。
17. 函数名：abort()
功 能 ：使异常程序终止。
18. 函数名：sleep()
功 能 ：执行挂起一段时间。
19. 函数名：psignal()
功 能 ：第二个参数 msg（通常是程序名）输出到标准出错文件，后接一个冒号和一个空格，再接着对该信号的说明，最后是一个换行符。该函数类似于 perror。
20. 函数名：strsignal()
功 能 ：类似于 strerror()。给出一个信号编号，strsignal 将返回说明该信号的字符串。应用程序可用该字符串打印关于接收到信号的出错消息。
21. 函数名：sig2str()
功 能 ：将信号编号映射为信号名。
22. 函数名：str2sig()
功 能 ：将信号名映射为信号编号。

线程：

1. 函数名：pthread_equal()
功 能 ：使用这个函数来对两个线程 ID 进行比较。
2. 函数名：pthread_self()
功 能 ：获得自身的线程 ID。
3. 函数名：pthread_creat()
功 能 ：创建线程。
4. 函数名：pthread_exit()
功 能 ：退出线程。
5. 函数名：pthread_join()
功 能 ：等待一个线程的结束，线程间同步的操作。
6. 函数名：pthread_cancel()
功 能 ：请求取消同一进程中的其他进程。
7. 函数名：pthread_cleanup_push()
功 能 ：线程可以建立多个清理处理程序（类似使用 atexit()注册的那些函数称为清理处理程序）。处理程序记录在栈中，使用 pthread_cleanup_push()注册线程清理处理程序。
8. 函数名：pthread_cleanup_pop()
功 能 ：删除线程清理处理程序。

▲延伸阅读：POSIX 线程 API 中提供了一个 pthread_cleanup_push()/pthread_cleanup_pop()函数对用于自动释放资源 --从 pthread_cleanup_push()的调用点到 pthread_cleanup_pop()之间的程序段中的终止动作（包括调用 pthread_exit()和取消点终止）都将执行 pthread_cleanup_push()所指定的清理函数。

9. 函数名：pthread_detach()
   功 能 ：使线程进入分离状态。
   ▲延伸阅读：在默认情况下，线程的终止状态会保存到对该线程调用 pthread_join，如果线程已经处于分离状态，线程的底层存储资源可以在线程终止时立即被收回。当线程被分离时，并不能用 pthread_join 函数等待它的终止状态。线程的分离状态决定一个线程以什么样的方式来终止自己。在默认情况下线程是非分离状态的，这种情况下，原有的线程等待创建的线程结束。只有当 pthread_join（）函数返回时，创建的线程才算终止，才能释放自己占用的系统资源。而分离线程不是这样子的，它没有被其他的线程所等待，自己运行结束了，线程也就终止了，马上释放系统资源。程序员应该根据自己的需要，选择适当的分离状态。
10. 函数名：pthread_mutex_init()
    功 能 ：初始化多线程互斥锁。
    ▲延伸阅读：互斥量是一个可以处于两态之一的变量：解锁和加锁。
11. 函数名：pthread_mutex_destroy()
    功 能 ：销毁互斥锁。
12. 函数名：pthread_mutex_lock()
    功 能 ：对互斥量进行加锁。
13. 函数名：pthread_mutex_trylock()
    功 能 ：尝试对互斥量进行加锁。
14. 函数名：pthread_mutex_unlock()
    功 能 ：解锁互斥量。
15. 函数名：pthread_rwlock_init()
    功 能 ：初始化读写锁。
16. 函数名：pthread_rwlock_destroy()
    功 能 ：销毁读写锁。
    ▲延伸阅读：读写锁有三种状态：读模式下加锁状态，写模式下加锁状态，不加锁状态。一次只能有一个线程可以占有写模式的读写锁，但可以有多个线程同时占有读模式的读写锁。
17. 函数名：pthread_rwlock_rdlock()
    功 能 ：在读模式下锁定读写锁。
18. 函数名：pthread_rwlock_wrlock()
    功 能 ：在写模式下锁定读写锁。
19. 函数名：pthread_rwlock_unlock()
    功 能 ：解锁。
20. 函数名：pthread_rwlock_tryrdlock()
    功 能 ：尝试在读模式下锁定读写锁。
21. 函数名：pthread_rwlock_trywrlock()
    功 能 ：尝试在写模式下锁定读写锁。
22. 函数名：pthread_cond_init()
    功 能 ：对条件变量进行初始化。
23. 函数名：pthread_cond_destroy()
    功 能 ：销毁条件变量。
    ▲延伸阅读：条件变量就是等待某一条件的发生，和信号一样。条件变量上的基本操作有：触发条件(当条件变为 true 时)；等待条件，挂起线程直到其他线程触发条件。
24. 函数名：pthread_cond_wait()
    功 能 ：等待条件变为真，如果在给定时间内条件不满足，则生成一个代码出错码的返回变量。

25. 函数名：pthread_cond_timedwait()
功 能 ： 和 pthread_cond_wait()相似，只是多了个 timeout 参数，指定等待时间。

线程控制：线程栈属性、进程共享属性、读写锁属性、互斥量类型属性、条件变量属性

1. 函数名：pthread_attr_init()
功 能 ： 初始化线程属性（即 pthread_attr_t 结构体）。
2. 函数名：pthread_attr_destroy()
功 能 ： 销毁线程属性。
3. 函数名：pthread_attr_getdetachstate()
功 能 ： 获取当前的 detachstate 线程属性。
4. 函数名：pthread_attr_setdetachstate()
功 能 ： 把线程属性 detachstate 设置为下面的两个合法值之一：设置为 PTHREAD_CREATE_DETACHED，以分离状态启动线程；或设置为 PTHREAD_CREATE_JOINABLE，正常启动线程，应用程序可以获取线程的终止状态。
5. 函数名：pthread_attr_getstack()
功 能 ： 获取线程栈属性。
6. 函数名：pthread_attr_setstack()
功 能 ： 设置线程栈属性。
7. 函数名：pthread_attr_getstacksize()
功 能 ： 获取线程栈的默认大小。
8. 函数名：pthread_attr_setstacksize()
功 能 ： 设置线程栈的默认大小。
9. 函数名：pthread_attr_getguardsize()
功 能 ： 线程属性 guardsize 控制着线程栈末尾之后用以避免栈溢出的扩展内存大小。该函数获取 guardsize 的大小。
10. 函数名：pthread_attr_setguardsize()
功 能 ： 设置 guardsize 的大小。
11. 函数名：pthread_getconcurrency()
功 能 ： 获取线程属性中的并发度。
12. 函数名：pthread_setconcurrency()
功 能 ： 设置线程属性中的并发度。
▲延伸阅读：并发度控制着用户线程可以映射的内核线程或进程的数目。
13. 函数名：pthread_mutexattr_init()
功 能 ： 初始化 pthread_mutexattr_t。
▲延伸阅读：同步是指多个线程访问同一变量时保持变量一致性的方法。pthread_mutexattr_t 是线程同步对象的属性结构体。
14. 函数名：pthread_mutexattr_destroy()
功 能 ： 销毁 pthread_mutexattr_t。
15. 函数名：pthread_mutexattr_getpshared()
功 能 ： 查询 pthread_mutexattr_t 结构，得到它的进程共享属性。
16. 函数名：pthread_mutexattr_setpshared()
功 能 ： 修改进程共享属性。
17. 函数名：pthread_mutexattr_gettype()
功 能 ： 得到互斥量类型属性。
18. 函数名：pthread_mutexattr_settype()
功 能 ： 设置互斥量类型属性。
19. 函数名：pthread_rwlockattr_init()
功 能 ： 初始化读写锁属性（pthread_rwlockattr_t）。
20. 函数名：pthread_rwlockattr_destroy()
功 能 ： 销毁读写锁属性（pthread_rwlockattr_t）。
21. 函数名：pthread_rwlockattr_getpshared()

功 能 ：获取读写锁的进程共享属性。

22. 函数名：pthread_rwlockattr_setpshared()
    功 能 ：设置读写锁的进程共享属性。
23. 函数名：pthread_condattr_init()
    功 能 ：初始化条件变量属性。
24. 函数名：pthread_condattr_destroy()
    功 能 ：销毁条件变量属性。
25. 函数名：pthread_condattr_getshared()
    功 能 ：获取条件变量的共享进程属性。
26. 函数名：pthread_condattr_setshared()
    功 能 ：设置条件变量的共享进程属性。
27. 函数名：ftrylockfile()
    功 能 ：尝试获取与给定 FILE 对象关联的锁。
    ▲延伸阅读：从信号处理程序返回后，重新恢复到断点处执行的过程中，函数所依赖的环境没有发生变化，就说这个函数是可重入的。如果一个函数在同一时刻可以被多个线程安全地调用，就称该函数是线程安全的。
28. 函数名：flockfile()
    功 能 ：获取与给定 FILE 对象关联的锁。
29. 函数名：funlockfile()
    功 能 ：释放锁。
30. 函数名：getchar_unlocked()
    功 能 ：不加锁版本的基于字符 I/O。除非被 flockfile（或 ftrylockfile）和 funlockfile 的调用包围，否则尽量不要调用这四个函数。
31. 函数名：getc_unlocked()
    功 能 ：不加锁版本的基于字符 I/O。除非被 flockfile（或 ftrylockfile）和 funlockfile 的调用包围，否则尽量不要调用这四个函数。
32. 函数名：putchar_unlocked()
    功 能 ：不加锁版本的基于字符 I/O。除非被 flockfile（或 ftrylockfile）和 funlockfile 的调用包围，否则尽量不要调用这四个函数。
33. 函数名：putc_unlocked()
    功 能 ：不加锁版本的基于字符 I/O。除非被 flockfile（或 ftrylockfile）和 funlockfile 的调用包围，否则尽量不要调用这四个函数。
34. 函数名：pthread_key_creat()
    功 能 ：创建线程与私有数据相关联的键。这个键将用于获取对线程私有数据的访问权。
    ▲延伸阅读：线程私有数据指的是各个线程不同的线程独立的数据。
35. 函数名：pthread_key_delete()
    功 能 ：取消键与线程私有数据值之间的关联关系。
36. 函数名：pthread_once()
    功 能 ：使用初值为 PTHREAD_ONCE_INIT 的 once_control 变量保证 initfn()函数（即第二个参数）在本进程执行序列中仅执行一次。
37. 函数名：pthread_getspecific()
    功 能 ：获得线程私有数据的地址。
38. 函数名：pthread_setspecific()
    功 能 ：把键和线程私有数据关联起来。
39. 函数名：pthread_setcancelstate()
    功 能 ：修改线程的可取消状态。
    ▲延伸阅读：可取消状态发生有两个，分别是 PTHREAD_CANCEL_ENABLE，也可以是 PTHREAD_CANCEL_DISABLE。
40. 函数名：pthread_testcancel()
    功 能 ：在程序中添加取消点。
    ▲延伸阅读：取消点是线程检查是否被取消并按照请求进行动作的一个位置（取消的

地点）。

41. 函数名：pthread_setcanceltype()
    功 能 ： 修改取消类型。
42. 函数名：pthread_sigmask()
    功 能 ： 和 sigprocmask 函数基本相同，检测或更改信号屏蔽字，不同的是 pthread_sigmask 工作在线程中，并且失败时返回错误码。
43. 函数名：sigwait()
    功 能 ： 等待一个或多个信号发生。
44. 函数名：pthread_kill()
    功 能 ： 把信号发送到线程。
45. 函数名：pthread_atfork()
    功 能 ： pthread_atfork 建立 fork 处理程序，清除进程从父进程中得到的锁状态。

守护进程：

1. 函数名：openlog()
   功 能 ： 打开系统记录的文件（即守护进程 syslog 记录的信息）。
2. 函数名：syslog()
   功 能 ： 记录至系统记录。
3. 函数名：closelog()
   功 能 ： 关闭已打开的 syslogd 的连接。
4. 函数名：setlogmask()
   功 能 ： 设置进程的记录优先级屏蔽字。

高级 IO：

1. 函数名：fcntl()
   功 能 ： 这是一个记录锁，可以改变已打开的文件性质。
   ▲延伸阅读：文件锁用来锁定整个文件，记录锁用于锁定文件的部分区域。
2. 函数名：isastream()
   功 能 ： 判断描述符是否引用一个流。
3. 函数名：select()
   功 能 ： 异步 IO，确定一个或多个套接口的状态，判断是否可读、可写或异常，并等待一段时间。
   ▲延伸阅读：进程告诉内核，当一个描述符已经准备好可以进行 IO 时，用一个信号通知它，这种技术称为异步 IO。
4. 函数名：pselect()
   功 能 ： 作用类似 select()。
5. 函数名：poll()
   功 能 ： 类似于 select()，不同的是，poll 并不为每个状态（可读性、可写性和异常状态）构造一个描述符集，而是构造一个 pollfd 数组，每个数组指定一个描述符编号以及对其所关心的状态。
6. 函数名：FD_ISSET()
   功 能 ： select()有三个类型为 fd_set 的参数，分别是需要判断是读、写、异常的描述符集。这 4 个 FD_开头的函数就是操作这些描述符集的函数。FD_ISSET()用于判断描述符是否在给定的描述符集中。
7. 函数名：FD_CLR()
   功 能 ： 清除描述符集中一指定位。
8. 函数名：FD_SET()
   功 能 ： 设置描述符集中一指定位。
9. 函数名：FD_ZERO()
   功 能 ： 清除描述符集中所有位。
10. 函数名：readv()
    功 能 ： 在一次函数调用中读多个非连续缓冲区。readv 总是先填满一个缓冲区，然

后再填写下一个。readv 返回读到的总字节数。如果遇到文件结尾，已无数据可读，则返回 0。又称为散布读。

11. 函数名：writev()
    功 能 ：在一次函数调用中写多个非连续缓冲区。将多个数据存储在一起，将驻留在两个或更多的不连接的缓冲区中的数据一次写出去。又称为聚集写。
12. 函数名：readn()
    功 能 ：读指定的 N 字节数据，并处理返回值小于要求值的情况。
13. 函数名：writen()
    功 能 ：写指定的 N 字节数据，并处理返回值小于要求值的情况。
14. 函数名：mmap()
    功 能 ：将一个给定的文件映射到一个存储区域中。
15. 函数名：mprotect()
    功 能 ：更改一个现存映射存储区的权限。
16. 函数名：msync()
    功 能 ：类似于 sync()，只不过作用于存储映射区。
17. 函数名：munmap()
    功 能 ：解除映射区。

进程间通信：

1. 函数名：pipe()
   功 能 ：创建管道。
2. 函数名：popen()
   功 能 ：popen() 函数通过创建一个管道，调用 fork 产生一个子进程，执行一个 shell 以运行命令来开启一个进程。这个进程必须由 pclose() 函数关闭，而不是 fclose() 函数。
3. 函数名：pclose()
   功 能 ：popen() 函数通过创建一个管道，调用 fork 产生一个子进程，执行一个 shell 以运行命令来开启一个进程。这个进程必须由 pclose() 函数关闭，而不是 fclose() 函数。
4. 函数名：mkfifo()
   功 能 ：创建 FIFO（即命名管道）。一般的文件 I/O 函数如 read 等都可用于 FIFO。

▲延伸阅读：进程间通信又称为 IPC，有三种 IPC 称作 XSI IPC，即消息队列、信号量和共享存储器，使用 ipc_perm 结构体规定权限和所有者。每个 IPC 都有一个内部名，是一个非负整数的标识符。键（key）是 IPC 的外部名，类型为 key_t，外部引用时使用键。

5. 函数名：ftok()
   功 能 ：由路径名和项目 ID 产生一个键。
6. 函数名：msgget()
   功 能 ：打开一个现存队列或创建一个新队列。
7. 函数名：msgctl()
   功 能 ：对消息队列执行多种操作。
8. 函数名：msgsnd()
   功 能 ：将数据放到消息队列中。
9. 函数名：msgrcv()
   功 能 ：从消息队列中取用消息。
10. 函数名：semget()
    功 能 ：获得一个信号量集 ID。

▲延伸阅读：理解信号量的关键是信号量的初值表示有多少个共享资源单位可供使用。使用了一个共享资源单位后，信号量减 1。以一个停车场的运作为例。简单起见，假设停车场只有三个车位（信号量初值为 3），一开始三个车位都是空的。这时如果同时来了五辆车，看门人允许其中三辆直接进入，然后放下车拦，剩下的车则必须在入口等待，此后来的车也都不得不在入口处等待。这时，有一辆车离开停车场（信号量加 1），看门人得知后，打开车拦，放入外面的一辆进去，如果又离开两辆（信号量加 2），则又可以放入两辆，如此往

复。

11. 函数名：semctl()
    功 能 ：包含了多种信号量操作。
12. 函数名：semop()
    功 能 ：自动执行信号量集合上的操作数组。
13. 函数名：shmget()
    功 能 ：获得存储标识符。
14. 函数名：shmctl()
    功 能 ：对共享存储执行多种操作。
15. 函数名：shmat()
    功 能 ：一旦创建了共享存储段，进程可以用 shmat()将其连接到它的地址空间中。
16. 函数名：shmdt()
    功 能 ：当对共享存储段操作已经结束时，调用 shmdt()脱离该段。

套接字：

1. 函数名：socket()
   功 能 ：创建一个套接字。
2. 函数名：shutdown()
   功 能 ：套接字通信是双向的，调用这个函数来禁止套接字的输入或输出。
3. 函数名：htonl()
   功 能 ：将无符号长整型数从主机字节序转换成网络字节序。
4. 函数名：htons()
   功 能 ：将短整型数从主机字节顺序转变成网络字节顺序。
5. 函数名：ntohl()
   功 能 ：将无符号长整形数从网络字节序转换为主机字节序。
6. 函数名：ntohs()
   功 能 ：将短整形数由网络字节顺序转换为主机字节顺序。
7. 函数名：inet_ntop()
   功 能 ：将网络字节序的二进制地址转换成文本字符串格式，即“二进制整数”地址转换成“点分十进制”格式。
8. 函数名：inet_pton()
   功 能 ：将文本字符串格式转换成网络字节序的二进制地址，即将“点分十进制”地址转换成“二进制整数”。
9. 函数名：gethostent()
   功 能 ：获取给定计算机的主机信息，如果主机数据文件没有打开，则该函数打开它。如果数据文件打开则返回文件的下一个条目。
10. 函数名：sethostent()
    功 能 ：打开文件，如果文件已打开，将其回绕。
11. 函数名：endhostent()
    功 能 ：关闭文件。
12. 函数名：getnetbyaddr()
    功 能 ：通过 addr 参数获取网络名、网络号。
13. 函数名：getnetbyname()
    功 能 ：通过 name 参数获取网络名、网络号。
14. 函数名：getnetent()
    功 能 ：获取网络名、网络号。
15. 函数名：setnetent()
    功 能 ：打开文件，如果文件已打开，将其回绕。
16. 函数名：endnetent()
    功 能 ：关闭文件。
17. 函数名：getprotobyname()
    功 能 ：返回对应于给定协议名的相关协议信息。

18. 函数名：getprotobynumber()
    功 能 ：返回对应于给定协议号的相关协议信息。
19. 函数名：getprotoent()
    功 能 ：获取协议名、协议号
20. 函数名：setprotoent()
    功 能 ：打开网络协议的数据文件，如果文件已打开，将其回绕。
21. 函数名：endprotoent()
    功 能 ：关闭文件。
22. 函数名：getservbyname()
    功 能 ：返回与给定服务名对应的包含名字和服务号信息的 servent 结构指针。
23. 函数名：getservbyport()
    功 能 ：返回对应于给定端口号和协议名的相关服务信息。
24. 函数名：getservent()
    功 能 ：获取服务名、服务端口号、服务所用协议。
25. 函数名：setservent()
    功 能 ：打开主机网络服务的数据文件相关函数，如果文件已打开，将其回绕。
26. 函数名：endservent()
    功 能 ：关闭文件。
27. 函数名：getaddrinfo()
    功 能 ：把主机名字和服务名字映射到一个地址。获取指定地址的信息，返回值是一个结构 addrinfo 的链表。
28. 函数名：freeaddrinfo()
    功 能 ：释放 addrinfo 结构。
29. 函数名：gai_strerror()
    功 能 ：如果 getaddrinfo 失败，不能使用 perror()或 strerror()来生成错误消息，替代地，调用 gai_strerror()将返回的错误码转换成错误消息。
30. 函数名：getnameinfo()
    功 能 ：将地址转换成主机名或服务器名
31. 函数名：bind()
    功 能 ：将地址绑定到一个套接字。
32. 函数名：getsockname()
    功 能 ：获取绑定到一个套接字的地址。
33. 函数名：getpeername()
    功 能 ：如果套接字已经和对方连接，调用 getpeername()来找到对方的地址。
34. 函数名：connect()
    功 能 ：建立一个连接。
35. 函数名：listen()
    功 能 ：监听网络，此时，服务器已经准备接收客户端连接请求了。
36. 函数名：accept()
    功 能 ：一旦服务器调用了 listen()，套接字就能接收连接请求。调用 accept()获得连接请求并建立连接。
37. 函数名：send()
    功 能 ：发送数据。
38. 函数名：sendto()
    功 能 ：发送数据，和 send()的区别在于 sendto 允许在无连接的套接字上指定一个目标地址，可用于无连接套接字。
39. 函数名：sendmsg()
    功 能 ：发送数据，可以指定多重缓冲区，和 writev()类似。可用于无连接套接字。
40. 函数名：recv()
    功 能 ：接收数据。
41. 函数名：recvfrom()
    功 能 ：接收数据，可用于无连接套接字。

42. 函数名：recvmsg()
    功 能 ：接收数据送入多个缓冲区，和 readv()类似。可用于无连接套接字。
43. 函数名：setsockopt()
    功 能 ：设置套接字选项。
44. 函数名：getsockopt()
    功 能 ：获取套接字选项。
45. 函数名：sockatmark()
    功 能 ：TCP 支持紧急标记的概念：在普通数据流中紧急数据所在的位置。为帮助数据是否接收到紧急标记，可以使用 sockafmark()。

高级进程间通信：

▲延伸阅读：本章介绍两种 IPC：基于 STREAMS 的管道 UNIX 域套接字。基于 STREAM 的管道是一个双向管道，单个 STREAMS 管道就能向父、子进程提供双向的数据流。UNIX 域套接字用于在同一台机器上运行的进程之间的通信。虽然因特网套接字可用于同一个目的，但 UNIX 域套接字的效率更高。

终端 I/O：

▲延伸阅读：终端(terminal，缩写 tty）是一种设备，不是一个程序，一般说的就是能提供命令行用户界面的设备，典型的是屏幕和键盘，或其他的一些物理终端。屏幕和键盘只是一个终端，可能不够用，又不想增加设备投入，就产生了虚拟终端。虚拟终端是一个程序，职责是模拟终端设备，和虚拟终端的区别表面上在于它以 GUI 形式的窗口出现，内部则是程序结构和系统控制结构有所不同，但本质上差不多。

1. 函数名：tcgetattr()
   功 能 ：获得终端属性（即 termios 结构）。
2. 函数名：tcsetattr()
   功 能 ：设置终端属性。
3. 函数名：cfgetispeed()
   功 能 ：获取输入波特率。
   ▲延伸阅读：波特率，即“位/秒”。
4. 函数名：cfgetospeed
   功 能 ：获取输出波特率。
5. 函数名：cfsetispeed()
   功 能 ：设置输入波特率。
6. 函数名：cfsetospeed()
   功 能 ：设置输出波特率。
7. 函数名：tcdrain()
   功 能 ：等待所有输出都被发送。
8. 函数名：tcflow()
   功 能 ：对输入和输出流控制进行控制。
9. 函数名：tcflush()
   功 能 ：刷清输入缓冲区或输出缓冲区。
10. 函数名：tcsendbreak()
    功 能 ：在一个指定的时间区间内发送连续的 0 位流。
11. 函数名：ctermid()
    功 能 ：确定控制终端的名字。
12. 函数名：isatty()
    功 能 ：判断文件描述符是否为终端设备。
13. 函数名：ttyname()
    功 能 ：获取在指定文件描述符上打开的终端设备的路径名。

伪终端：

1. 函数名：posix_openpt()

功 能 ：打开下一个可用的伪终端主设备。

2. 函数名：grantpt()
   功 能 ：在终端从设备可被使用前，必须设置它的权限，使得应用程序可以访问它，grantpt()提供这样的功能。
3. 函数名：unlockpt()
   功 能 ：用于准予对伪终端从设备的访问，从而允许应用程序打开该设备。
4. 函数名：ptsname()
   功 能 ：用于在给定主伪终端设备的文件描述符时，找到从伪终端设备的路径名。
5. 函数名：ptym_open()
   功 能 ：打开下一个可用的 PTY 主设备。
6. 函数名：ptys_open()
   功 能 ：打开下一个可用的 PTY 从设备。