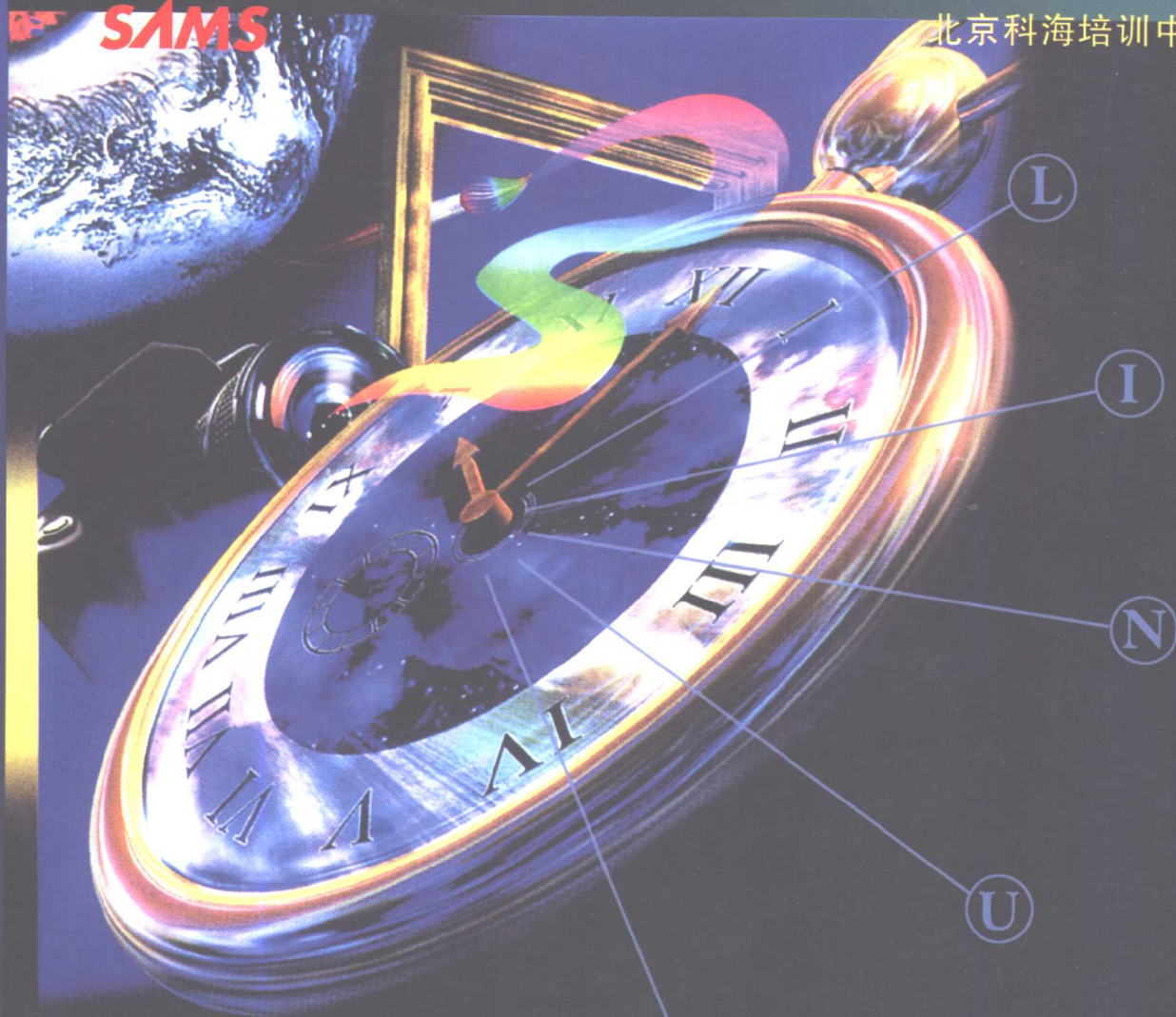


SAMS

北京科海培训中心



(美) K.Wall, M.Watson, M.Whitis et, al 著  
王 勇 王一川 林花军 甘 泉 译

X

# GNU/Linux

## 编程指南

清华大学出版社



SAMS

北京科海培训中心

# GNU/Linux 编程指南

[美] K. Wall, M. Watson, M. Whitis 著  
王勇 王一川 林花军 甘泉 译

清华大学出版社

# (京)新登字 158 号

著作权合同登记号:01-1999-3859

## 内 容 提 要

本书介绍如何在 Linux 的内核、系统调用及应用程序层次上编写应用程序。主要内容包括:建立和定制用于 Linux 应用程序的 GUI 环境,探索 X-Window 系统,介绍 KDE 和 GNOME 库的强大功能,以及系统编程的各个专题,包括进程、信号、内存管理和进程间通信、集成 Internet 功能等。本书的独到之处是具体讲述了 RPM 之类的软件包管理工具以及如何编写有用的文档使之把应用程序转交给用户。这通常是其他编程书中所没有的。

本书内容全面,既讨论编程所需用到的工具,也讨论了 GNU/Linux 的工作环境,并细致地讲述了 Linux/Unix 具体编程的独到之处。该书适用于中高级程序开发人员。

## Linux®Programming Unleashed

Copyright ©June 1999 by Sams

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher.

本书中文简体字版由美国培生教育出版集团 SAMS 公司授权北京科海培训中心和清华大学出版社出版。未经出版者书面允许不得以任何方式复制或抄袭本书内容。

**本书贴有清华大学出版社标签,无防伪标签者不得销售。**

**版权所有,盗版必究。**

书 名: GNU/Linux 编程指南

作 者: K. Wall, M. Watson, M. Whitis

译 者: 王勇 王一川 林花军 甘泉

出版者: 清华大学出版社(北京清华大学校内, 邮编 100084)

印刷者: 北京朝阳科普印刷厂

发 行: 新华书店总店北京科技发行所

开 本: 16 印张: 37 字数: 900 千字

版 次: 2000 年 6 月第 1 版 2000 年 7 月第 2 次印刷

印 数: 5 001~3000

书 号: ISBN 7-302-03913-5/TP·2287

定 价: 62.00 元

# 引言

GNU/Linux 从一开始就提供了丰富的编程环境,而且这个环境只能越来越丰富。egcs 和 pgcs 这两个崭新的编译器加盟到 GNU 项目的 gcc 中,而后者是 GNU/Linux 最初的编译器。事实上在本书即将付印时,作为 GNU 项目“监护人”的自由软件基金会已宣布 gcc 将由 egcs 的创建者和维护者一并维护。一大批不同风格的编辑器也跟刚强的却因其不可思议的精致而颇遭诽谤的 vi 和 emacs 站在同一战壕中。很大程度上受 Linux 内核的驱动,GNU 的 C 函数库 glibc(也称为 libc6)的新版本已经成为标准 C 函数库。Linux 的黑客们已把 GNU 项目久经使用的开发套件磨砺成功能强大的工具。新的窗口部件集已在人们钟爱的老式 UNIX 特性旁边站稳脚跟。Lesstif 是 Motif 1.2 的一个自由的源代码级兼容的实现;基于出自 TrollTech 公司的 Qt 类库的 K 桌面环境(简称 KDE)回应了由 X 联盟(X Consortium)的公共桌面环境(简称 CDE)引发的桌面挑战。

## 本书的目标

我们打算在本书中向读者介绍如何在 Linux 的内核、系统调用及应用程序层次上编写程序。我们将集中在 C 语言上,因为 C 语言仍然是 Linux 的通用语言。介绍完一些基本的开发工具后,我们就讨论系统编程,接下来是关于进程间通信和网络编程的部分。

在探讨以基于文本的工具或图形工具(X Windows 系统)编写 Linux 用户接口程序的部分后,本书将由专门化主题汇编成的一个部分结束技术讨论,内容包括 shell 编程、安全考虑以及 GNU 项目 gdb 调试器的使用。本书的最后三章探讨其他编程书籍中通常忽略的一个话题:把应用程序转交给用户。这几章告诉你如何使用诸如 RPM 之类的软件包管理工具,如何编写有用的文档,并讨论了许可权问题和选择。如果我们尽到了编写本书的职责,那么它应该能够为读者参与到被称为“GNU/Linux”的社会性和技术性大潮中做好准备。

## 本书的读者群

熟悉其他操作系统但没有接触过 GNU/Linux 的程序员将从本书学到在 GNU/Linux 下编程的牢固入门知识。我们既讨论需用到的工具,也讨论 GNU/Linux 的工作环境。

有经验的 UNIX 程序员将发现 GNU/Linux 的编程思想与 UNIX 极为相似。我们希望向他们强调所能碰到的差异。GNU/Linux 的一个重要话题是最大范围的移植能力,因为它运行在种类不断增长的平台之上,包括 Inter i386、Sun Sparc、Digital Alpha、MIPS 处理器、Power PC 以及基于 Motorola 68000 的 Macintosh 计算机。

中等程度的 C 程序员也能从本书中学到不少东西。一般地说,对 GNU/Linux 的编程与其他类 UNIX 系统的编程相似,因此我们将先把他们引入变成实际的 UNIX 程序员的道路,再介绍 Linux/UNIX 具体编程的独到之处。

## 本书的组织

本书不是 C 语言的入门书,不过你可从中很快温习一下久违的 C 语言。你应该能够读懂 C 语言代码,理解常见的 C 语法。我们选定的工具很少偏离可从 GNU 项目得到的工具箱。其原因很简单:GNU 软件是每个 Linux 发行版本的标准配置。这也是我们合称它们的 GNU/Linux 的原因。

开头七章讨论开发系统的设置和标准 GNU/Linux 开发工具的使用:

- . gcc
- . make
- . autoconf
- . diff
- . patch
- . RCS
- . emacs

第 2 部分讨论系统编程主题。要是你对标准 C 函数库不大熟悉,第 9 章可用来清理这样的紊乱状态。第 10 章讨论 Linux 的文件操纵例程。第 11 章回答“什么是进程?”的问题并给出与进程和作业控制关联的系统调用。第 12 章讨论如何获取系统信息。到第 13 章时我们将走上讲台传布出错检查是件大事的理由。当然,我们还会告诉你如何去做。第 14 章专门探讨 Linux 内存管理的怪异行为。

我们在第 15 章~第 18 章中分别进行使用管道、消息队列、共享内存区和信号灯进行进程间通信的讨论。第 19 章~第 23 章讨论如何基于 TCP/IP 网络协议编写程序。在第 24 章一般性地介绍如何创建和使用编程用函数库(包括从 libc5 到 libc6 的转换)之后,我们在第 25 章讨论编写设备驱动程序和内核模块,因为给最新的高速硬件设备或系统设备提供内核支持需要花相当多的编程精力。

接下去在第 26 章~第 33 章讨论用户接口编程。其中第 26 章和第 27 章讨论字符模式编程;第 26 章讨论使用 termcap 和 termios 直接控制终端,第 27 章讨论使用 ncurses 的更简便方法。在第 28 章简短介绍 X 编程,第 29 章将集中讨论使用 Motif 和 Athena 窗口部件集。使用 GTK 函数库的 X 编程是第 30 章的主题,使用 Qt 函数库(KDE 的基础)的 X 编程是第 31 章的主题,Java 编程则在第 32 章讨论。我们还在第 33 章讨论使用 OpenGL 进行三维图形编程。

第 5 部分涵盖三个特殊目的的主题。其中第 34 章讨论 bash 的 shell 编程。第 35 章讨论安全相关的编程问题。第 36 章讨论使用 gdb 调试程序。

本书最后讨论如何把自己的编程项目转发出去。第 37 章介绍 tar 和 RPM 软件包管理工具。程序文档极为重要,因此我们在第 38 章指导你如何编写手册页面以及如何使用某些基于 SGML 的建档工具。第 39 章最后查看软件许可证使用方面的重大问题。

## 关于作者

### **Kurt Wall**

Kurt Wall 是犹他州盐湖城的一位 Linux 书作者、咨询专家外加发烧友。他从 1993 年开始使用 Linux。业余时间除写书外,他还担任盐湖 Linux 用户组的副总和国际 Informix 用户组 Linux SIG 的老总。不坐在计算机屏幕前的时候,他跟他们 10 岁的 Ashleigh Rae 和 8 岁的 Zane Thomas 两个小孩一块玩,喝精制的咖啡,每天中午小憩一番。

### **Mark Watson**

Mark Watson 是关于人工智能、Java 和 C++ 等书的作者。他是 Intelligenesis 公司的一位高级 AI 软件工程师。Watson 先生跟妻子 Carol 一起住在亚利桑那州 Sedona,享受着溜狗、赏乐、烹调、下棋的恬静生活。

### **Mark Whitis**

Mark Whitis 是自学成材的计算机咨询工程师。他拥有自己的公司——自由电子试验室 (Free Electron Labs),又为设计数字 (Digital By Design) 公司工作。他曾在学术、公司和军事部门的多个研发实验室从事过硬件和软件开发,也在商业部门从事过软件开发。他给兼容 UNIX 的操作系统开发软件的历史已超过 10 年,其中包括 Linux 上的 4 年左右。他从来不用 Windows。他的工作包括安全咨询、系统与网络维护以及科学应用程序、设备驱动程序、客户/服务器应用程序、仿真软件、诊断软件和基于 Web 的商业事务系统的开发。他还是一些公开可得的软件包的作者。他目前住在弗吉尼亚州 Charlottesville。他的网站地址为 <http://www.freelabs.com/unleashed/>。

## 关于译者

本书第 1 部分由清华大学王一川同志翻译;第 2 部分由装备指挥技术学院林花军同志翻译;第 3 部分由北京大学刘勇同志翻译;第 4 部分由北京大学王勇同志翻译;甘泉同志翻译了本书的第 5 部分和附录。

## 有关原书附光盘的特殊说明

由于英文原版书在出版前才决定不附带光盘销售,故原版书中多处提到的引用“所附光盘”的内容,本译文中也相应多处出现,特此说明。读者若需要本书中的实例程序和符号表函数库以及需了解其中的原由可直接访问原版书作者的个人主页:

<http://www.freelabs.com/~whitis/software/symbol/>

作者主页可能有最新的 Linux 版本和相关的信息。

# 目 录

## 第 1 部分 Linux 编程工具包

|                                |      |
|--------------------------------|------|
| <b>第 1 章 概貌</b> .....          | (1)  |
| 1.1 Linux 既往成就 .....           | (1)  |
| 1.2 Linux 预期成就 .....           | (1)  |
| 1.3 Linux 简史 .....             | (2)  |
| 1.4 Linux 与 UNIX .....         | (2)  |
| 1.5 Linux 编程 .....             | (2)  |
| 1.6 为何选择 Linux 编程 .....        | (3)  |
| 1.7 小结 .....                   | (3)  |
| <b>第 2 章 设置开发系统</b> .....      | (4)  |
| 2.1 硬件选择 .....                 | (4)  |
| 2.1.1 硬件选择上的考虑 .....           | (4)  |
| 2.2 处理器/主板 .....               | (5)  |
| 2.2.1 板上 I/O .....             | (6)  |
| 2.2.2 处理器 .....                | (6)  |
| 2.2.3 BIOS .....               | (7)  |
| 2.2.4 内存 .....                 | (7)  |
| 2.2.5 机箱和电源 .....              | (7)  |
| 2.3 用户交互硬件:视频、声音、键盘及鼠标 .....   | (8)  |
| 2.3.1 显卡 .....                 | (8)  |
| 2.3.2 显示器 .....                | (9)  |
| 2.3.3 声卡 .....                 | (10) |
| 2.4 键盘与鼠标 .....                | (10) |
| 2.5 通信设备、端口及总线 .....           | (11) |
| 2.5.1 调制解调器 .....              | (11) |
| 2.5.2 网络接口卡 .....              | (12) |
| 2.5.3 SCSI .....               | (13) |
| 2.5.4 USB 与火线(IEEE 1394) ..... | (13) |
| 2.5.5 串行卡(包括多端口) .....         | (13) |
| 2.5.6 IRDA .....               | (14) |
| 2.5.7 PCMCIA 卡 .....           | (14) |
| 2.5.8 ISA 即插即用设备 .....         | (14) |
| 2.6 存储设备 .....                 | (14) |
| 2.6.1 硬盘 .....                 | (14) |

|              |                                  |             |
|--------------|----------------------------------|-------------|
| 2.6.2        | 可移动磁盘设备 .....                    | (14)        |
| 2.6.3        | CD-ROM/DVD .....                 | (15)        |
| 2.6.4        | 磁带备份设备 .....                     | (15)        |
| 2.7          | 外围设备 .....                       | (15)        |
| 2.7.1        | 打印机 .....                        | (15)        |
| 2.7.2        | 扫描仪 .....                        | (16)        |
| 2.7.3        | 数字相机 .....                       | (17)        |
| 2.7.4        | 家居自动控制设备 .....                   | (17)        |
| 2.8          | 完备型系统 .....                      | (17)        |
| 2.9          | 便携系统 .....                       | (18)        |
| 2.10         | 安装 .....                         | (18)        |
| 2.11         | 小结 .....                         | (21)        |
| <b>第 3 章</b> | <b>使用 GNU cc .....</b>           | <b>(22)</b> |
| 3.1          | GNU cc 特性 .....                  | (22)        |
| 3.2          | 使用简介 .....                       | (22)        |
| 3.3          | 常用命令行选项 .....                    | (24)        |
| 3.3.1        | 函数库和包含文件 .....                   | (25)        |
| 3.3.2        | 出错检查及警告 .....                    | (26)        |
| 3.4          | 优化选项 .....                       | (27)        |
| 3.5          | 调试选项 .....                       | (28)        |
| 3.6          | GNU C 扩展 .....                   | (29)        |
| 3.7          | 小结 .....                         | (30)        |
| <b>第 4 章</b> | <b>使用 GNU make 管理项目 .....</b>    | <b>(31)</b> |
| 4.1          | 为何使用 make .....                  | (31)        |
| 4.2          | 编写 makefile .....                | (31)        |
| 4.3          | 有关 makefile 编写的详细规则 .....        | (33)        |
| 4.3.1        | 伪目标 .....                        | (33)        |
| 4.3.2        | 变量 .....                         | (33)        |
| 4.3.3        | 环境变量、自动变量及预定义变量 .....            | (35)        |
| 4.3.4        | 隐式规则 .....                       | (36)        |
| 4.3.5        | 模式规则 .....                       | (36)        |
| 4.3.6        | 注释 .....                         | (37)        |
| 4.4          | 额外的 make 命令行选项 .....             | (37)        |
| 4.5          | 调试 make .....                    | (37)        |
| 4.6          | 常见的 make 出错消息 .....              | (38)        |
| 4.7          | 有用的 makefile 目标 .....            | (38)        |
| 4.8          | 小结 .....                         | (38)        |
| <b>第 5 章</b> | <b>使用 autoconf 创建自配置软件 .....</b> | <b>(39)</b> |
| 5.1          | 理解 autoconf .....                | (39)        |

|              |                          |             |
|--------------|--------------------------|-------------|
| 5.1.1        | 创建 configure.in 文件 ..... | (40)        |
| 5.1.2        | 构造文件 .....               | (40)        |
| 5.1.3        | 有用的 autoconf 实用工具 .....  | (41)        |
| 5.2          | 内置宏 .....                | (41)        |
| 5.2.1        | 候选程序测试 .....             | (41)        |
| 5.2.2        | 库函数测试 .....              | (42)        |
| 5.2.3        | 头文件测试 .....              | (43)        |
| 5.2.4        | 结构测试 .....               | (44)        |
| 5.2.5        | 类型定义测试 .....             | (44)        |
| 5.2.6        | 编译器行为测试 .....            | (44)        |
| 5.2.7        | 系统服务测试 .....             | (45)        |
| 5.2.8        | UNIX 变种测试 .....          | (45)        |
| 5.3          | 普通宏 .....                | (46)        |
| 5.4          | 一个带注解的 autoconf 脚本 ..... | (47)        |
| 5.5          | 小结 .....                 | (52)        |
| <b>第 6 章</b> | <b>比较和归并源文件 .....</b>    | <b>(53)</b> |
| 6.1          | 比较文件 .....               | (53)        |
| 6.1.1        | 理解 cmp 命令 .....          | (53)        |
| 6.1.2        | 理解 diff 命令 .....         | (55)        |
| 6.1.3        | 理解 diff3 命令 .....        | (60)        |
| 6.1.4        | 理解 sdiff 命令 .....        | (63)        |
| 6.2          | 准备源代码补丁 .....            | (63)        |
| 6.2.1        | patch 命令行选项 .....        | (64)        |
| 6.2.2        | 创建补丁 .....               | (64)        |
| 6.2.3        | 使用补丁 .....               | (65)        |
| 6.3          | 小结 .....                 | (65)        |
| <b>第 7 章</b> | <b>使用 RCS 控制版本 .....</b> | <b>(66)</b> |
| 7.1          | 术语 .....                 | (66)        |
| 7.2          | RCS 基本用法 .....           | (67)        |
| 7.2.1        | ci 和 co .....            | (67)        |
| 7.2.2        | RCS 关键字 .....            | (69)        |
| 7.2.3        | ident 命令 .....           | (70)        |
| 7.3          | rcsdiff .....            | (71)        |
| 7.4          | 其他 RCS 命令 .....          | (73)        |
| 7.4.1        | rcsclean .....           | (74)        |
| 7.4.2        | rlog .....               | (74)        |
| 7.4.3        | rcs .....                | (74)        |
| 7.4.4        | rcsmerge .....           | (74)        |
| 7.5          | 小结 .....                 | (75)        |

|                                       |             |
|---------------------------------------|-------------|
| <b>第 8 章 在 Emacs 中创建程序 .....</b>      | <b>(76)</b> |
| 8.1 Emacs 简介 .....                    | (76)        |
| 8.1.1 启动与停止 Emacs .....               | (76)        |
| 8.1.2 移动 .....                        | (78)        |
| 8.1.3 插入文本 .....                      | (79)        |
| 8.1.4 删除文本 .....                      | (79)        |
| 8.1.5 查找和替换 .....                     | (80)        |
| 8.1.6 保存和打开文件 .....                   | (81)        |
| 8.1.7 多窗口 .....                       | (82)        |
| 8.2 支持编程的特性 .....                     | (83)        |
| 8.2.1 缩进约定 .....                      | (83)        |
| 8.2.2 语法高亮度显示 .....                   | (84)        |
| 8.2.3 使用注释 .....                      | (84)        |
| 8.2.4 使用 Emacs 进行编译 .....             | (85)        |
| 8.2.5 定制简介 .....                      | (87)        |
| 8.3 使用 Emacs Lisp 自动完成 Emacs 操作 ..... | (89)        |
| 8.4 小结 .....                          | (91)        |

## 第 2 部分 系统编程

|                              |             |
|------------------------------|-------------|
| <b>第 9 章 I/O 例程 .....</b>    | <b>(92)</b> |
| 9.1 文件描述符 .....              | (92)        |
| 9.2 使用文件描述符的系统调用 .....       | (92)        |
| 9.2.1 open()调用 .....         | (93)        |
| 9.2.2 close()调用 .....        | (94)        |
| 9.2.3 read()调用 .....         | (94)        |
| 9.2.4 write()调用 .....        | (94)        |
| 9.2.5 ioctl()调用 .....        | (94)        |
| 9.2.6 fcntl()调用 .....        | (95)        |
| 9.2.7 fsync()调用 .....        | (95)        |
| 9.2.8 ftruncate()调用 .....    | (96)        |
| 9.2.9 lseek()调用 .....        | (96)        |
| 9.2.10 dup()和 dup2()调用 ..... | (96)        |
| 9.2.11 select()调用 .....      | (97)        |
| 9.2.12 fstat()调用 .....       | (101)       |
| 9.2.13 fchown()调用 .....      | (102)       |
| 9.2.14 fchmod()调用 .....      | (102)       |
| 9.2.15 fchdir()调用 .....      | (103)       |
| 9.2.16 flock()调用 .....       | (103)       |
| 9.2.17 pipe()调用 .....        | (103)       |
| 9.3 文件类型 .....               | (104)       |

|                                                     |              |
|-----------------------------------------------------|--------------|
| 9.3.1 普通文件 .....                                    | (104)        |
| 9.3.2 磁带 I/O .....                                  | (106)        |
| 9.3.3 串口 I/O .....                                  | (109)        |
| 9.3.4 打印机接口 .....                                   | (109)        |
| 9.3.5 声卡 .....                                      | (109)        |
| 9.4 小结 .....                                        | (109)        |
| <b>第 10 章 文件操作 .....</b>                            | <b>(110)</b> |
| 10.1 文件操作函数 .....                                   | (110)        |
| 10.1.1 打开/关闭文件 .....                                | (110)        |
| 10.1.2 基本的读和写 .....                                 | (111)        |
| 10.1.3 状态函数 .....                                   | (111)        |
| 10.2 格式化输出 .....                                    | (112)        |
| 10.2.1 格式化输入 .....                                  | (114)        |
| 10.2.2 基于字符和行的输入输出 .....                            | (115)        |
| 10.2.3 文件定位 .....                                   | (116)        |
| 10.2.4 缓冲区控制 .....                                  | (116)        |
| 10.2.5 删除和改名 .....                                  | (117)        |
| 10.2.6 临时文件 .....                                   | (117)        |
| 10.3 小结 .....                                       | (118)        |
| <b>第 11 章 进程控制 .....</b>                            | <b>(119)</b> |
| 11.1 进程的属性 .....                                    | (119)        |
| 11.2 系统调用及库函数 .....                                 | (120)        |
| 11.2.1 fork()系统调用 .....                             | (120)        |
| 11.2.2 exec()函数族 .....                              | (120)        |
| 11.2.3 system()和 popen()函数 .....                    | (121)        |
| 11.2.4 clone()函数调用 .....                            | (121)        |
| 11.2.5 wait()、waitpid()、wait3()以及 wait4()系统调用 ..... | (122)        |
| 11.2.6 select()系统调用 .....                           | (122)        |
| 11.2.7 信号 .....                                     | (122)        |
| 11.2.8 程序的中止 .....                                  | (124)        |
| 11.2.9 闹钟和计时器 .....                                 | (125)        |
| 11.3 调度参数 .....                                     | (126)        |
| 11.4 线程 .....                                       | (127)        |
| 11.4.1 pthread_create()函数 .....                     | (128)        |
| 11.4.2 pthread_exit()函数 .....                       | (128)        |
| 11.4.3 pthread_join()函数 .....                       | (128)        |
| 11.4.4 属性的控制 .....                                  | (129)        |
| 11.4.5 pthread_atfork()函数 .....                     | (130)        |
| 11.4.6 取消线程 .....                                   | (130)        |
| 11.4.7 pthread_cleanup_push()宏 .....                | (131)        |

|                            |                             |              |
|----------------------------|-----------------------------|--------------|
| 11.4.8                     | pthread_cond_init()函数 ..... | (131)        |
| 11.4.9                     | pthread_equal()函数 .....     | (132)        |
| 11.4.10                    | 互斥 .....                    | (132)        |
| 11.5                       | 编程示例 .....                  | (132)        |
| 11.5.1                     | 子库 .....                    | (133)        |
| 11.5.2                     | 例子 child_demo1.c 程序 .....   | (143)        |
| 11.5.3                     | 例子程序 child_demo2.c .....    | (145)        |
| 11.5.4                     | 例子程序 child_demo3.c .....    | (150)        |
| 11.6                       | 小结 .....                    | (151)        |
| <b>第 12 章 访问系统信息 .....</b> |                             | <b>(152)</b> |
| 12.1                       | 进程的信息 .....                 | (152)        |
| 12.1.1                     | cmdline 文件 .....            | (153)        |
| 12.1.2                     | environ 文件 .....            | (153)        |
| 12.1.3                     | fd 目录 .....                 | (153)        |
| 12.1.4                     | mem 文件 .....                | (153)        |
| 12.1.5                     | stat 文件 .....               | (153)        |
| 12.1.6                     | status 文件 .....             | (155)        |
| 12.1.7                     | cwd 符号链接 .....              | (155)        |
| 12.1.8                     | exe 符号链接 .....              | (155)        |
| 12.1.9                     | maps 文件 .....               | (155)        |
| 12.1.10                    | root 符号链接 .....             | (155)        |
| 12.1.11                    | statm 文件 .....              | (155)        |
| 12.2                       | 普通的系统信息 .....               | (155)        |
| 12.2.1                     | /proc/cmd/line 文件 .....     | (155)        |
| 12.2.2                     | /proc/cpuinfo 文件 .....      | (155)        |
| 12.2.3                     | /proc/devices 文件 .....      | (156)        |
| 12.2.4                     | /proc/dma 文件 .....          | (156)        |
| 12.2.5                     | /proc/filesystems 文件 .....  | (156)        |
| 12.2.6                     | /proc/interrupts 文件 .....   | (156)        |
| 12.2.7                     | /proc/ioports 文件 .....      | (156)        |
| 12.2.8                     | /proc/kcore 文件 .....        | (156)        |
| 12.2.9                     | /proc/kmsg 文件 .....         | (156)        |
| 12.2.10                    | /proc/ksyms 文件 .....        | (156)        |
| 12.2.11                    | /proc/loadavg 文件 .....      | (157)        |
| 12.2.12                    | /proc/locks 文件 .....        | (157)        |
| 12.2.13                    | /proc/mdstat 文件 .....       | (157)        |
| 12.2.14                    | /proc/meminfo 文件 .....      | (157)        |
| 12.2.15                    | /proc/misc 文件 .....         | (157)        |
| 12.2.16                    | /proc/modules 文件 .....      | (157)        |
| 12.2.17                    | /proc/mounts 文件 .....       | (157)        |
| 12.2.18                    | /proc/pci 文件 .....          | (157)        |
| 12.2.19                    | /proc/rtc 文件 .....          | (158)        |

|               |                        |              |
|---------------|------------------------|--------------|
| 12.2.20       | /proc/stat 文件 .....    | (158)        |
| 12.2.21       | /proc/uptime 文件 .....  | (158)        |
| 12.2.22       | /proc/version 文件 ..... | (158)        |
| 12.2.23       | /proc/net 子目录 .....    | (158)        |
| 12.2.24       | /proc/scsi 子目录 .....   | (159)        |
| 12.2.25       | /proc/sys 目录 .....     | (159)        |
| 12.3          | 函数库和实用工具 .....         | (160)        |
| 12.4          | 小结 .....               | (160)        |
| <b>第 13 章</b> | <b>出错处理 .....</b>      | <b>(161)</b> |
| 13.1          | 出错处理简述 .....           | (161)        |
| 13.2          | C 语言机制 .....           | (161)        |
| 13.2.1        | assert()宏 .....        | (161)        |
| 13.2.2        | 使用预编译 .....            | (163)        |
| 13.2.3        | 标准库工具 .....            | (165)        |
| 13.2.4        | 对错误代码 errno 的理解 .....  | (165)        |
| 13.2.5        | 使用函数 abort() .....     | (166)        |
| 13.2.6        | 使用函数 exit() .....      | (166)        |
| 13.2.7        | 使用函数 atexit() .....    | (167)        |
| 13.2.8        | 使用函数 strerror() .....  | (168)        |
| 13.2.9        | 使用函数 perror() .....    | (168)        |
| 13.3          | 系统日志工具 .....           | (169)        |
| 13.3.1        | 用户程序 .....             | (173)        |
| 13.4          | 小结 .....               | (173)        |
| <b>第 14 章</b> | <b>内存管理 .....</b>      | <b>(174)</b> |
| 14.1          | C 内存管理的回顾 .....        | (174)        |
| 14.1.1        | malloc()函数的使用 .....    | (174)        |
| 14.1.2        | calloc()函数的使用 .....    | (174)        |
| 14.1.3        | realloc()函数的使用 .....   | (175)        |
| 14.1.4        | free()函数的使用 .....      | (175)        |
| 14.1.5        | alloca()函数的使用 .....    | (175)        |
| 14.2          | 内存映像文件 .....           | (177)        |
| 14.2.1        | mmap()函数的使用 .....      | (177)        |
| 14.2.2        | munmap()函数的使用 .....    | (178)        |
| 14.2.3        | msync()函数的使用 .....     | (178)        |
| 14.2.4        | mprotect()函数的使用 .....  | (179)        |
| 14.2.5        | 锁住内存 .....             | (179)        |
| 14.2.6        | mremap()函数的使用 .....    | (179)        |
| 14.2.7        | 用内存映像实现 cat(1) .....   | (180)        |
| 14.3          | 发现并修改内存管理错误 .....      | (181)        |
| 14.3.1        | 一个有问题的程序 .....         | (182)        |

|        |                            |       |
|--------|----------------------------|-------|
| 14.3.2 | 用 mpr 和 check 定位内存错误 ..... | (183) |
| 14.3.3 | Electric Fence .....       | (185) |
| 14.3.4 | 使用 Lint Brush .....        | (187) |
| 14.4   | 小结 .....                   | (188) |

### 第 3 部分 进程间通信和网络编程

#### 第 15 章 IPC 入门:使用管道 .....

(189)

|        |              |       |
|--------|--------------|-------|
| 15.1   | 管道使用简介 ..... | (190) |
| 15.1.1 | 无名管道 .....   | (190) |
| 15.1.2 | 有名管道 .....   | (191) |
| 15.2   | 小结 .....     | (193) |

#### 第 16 章 消息队列 .....

(194)

|      |                      |       |
|------|----------------------|-------|
| 16.1 | 创建一个使用消息队列示例程序 ..... | (194) |
| 16.2 | 运行这个消息队列示例程序 .....   | (196) |
| 16.3 | 小结 .....             | (196) |

#### 第 17 章 共享内存 .....

(197)

|      |                         |       |
|------|-------------------------|-------|
| 17.1 | 把 Linux 配置成使用共享内存 ..... | (197) |
| 17.2 | 创建一个使用共享内存的例子程序 .....   | (197) |
| 17.3 | 运行共享内存示例程序 .....        | (199) |
| 17.4 | 小结 .....                | (200) |

#### 第 18 章 信号灯 .....

(201)

|      |                  |       |
|------|------------------|-------|
| 18.1 | 使用信号灯的示例程序 ..... | (201) |
| 18.2 | 运行信号灯示例程序 .....  | (205) |
| 18.3 | 小结 .....         | (206) |

#### 第 19 章 TCP/IP 和套接口编程 .....

(207)

|        |                        |       |
|--------|------------------------|-------|
| 19.1   | 支持套接口编程的系统调用 .....     | (207) |
| 19.1.1 | socket .....           | (207) |
| 19.1.2 | bind .....             | (208) |
| 19.1.3 | listen .....           | (209) |
| 19.1.4 | connect .....          | (209) |
| 19.1.5 | recv .....             | (210) |
| 19.1.6 | send .....             | (210) |
| 19.2   | 使用套接口的客户/服务器例子程序 ..... | (211) |
| 19.2.1 | 服务器例子程序 .....          | (211) |
| 19.2.2 | 客户例子程序 .....           | (212) |
| 19.2.3 | 运行客户和服务例子程序 .....      | (213) |

|                                                         |              |
|---------------------------------------------------------|--------------|
| 19.2.4 使用 Web 浏览器作为客户运行服务器例子程序 .....                    | (213)        |
| 19.3 一个简单的 Web 服务器程序和 Web 客户程序 .....                    | (214)        |
| 19.3.1 实现一个简单的 Web 服务器程序 .....                          | (214)        |
| 19.3.2 实现一个简单的 Web 客户程序 .....                           | (216)        |
| 19.3.3 测试这个 Web 服务器程序和 Web 客户程序 .....                   | (216)        |
| 19.3.4 使用 Netscape Navigator 作为客户运行这个简单 Web 服务器程序 ..... | (217)        |
| 19.4 小结 .....                                           | (218)        |
| <b>第 20 章 UDP: 用户数据协议 .....</b>                         | <b>(219)</b> |
| 20.1 一个发送 UDP 数据的例子程序 .....                             | (219)        |
| 20.2 一个接收 UDP 数据的例子程序 .....                             | (220)        |
| 20.3 运行这两个 UDP 例子程序 .....                               | (221)        |
| 20.4 小结 .....                                           | (222)        |
| <b>第 21 章 使用多播套接口 .....</b>                             | <b>(223)</b> |
| 21.1 把 Linux 配置成支持多播 IP .....                           | (223)        |
| 21.2 几个使用多播 IP 的例子程序 .....                              | (224)        |
| 21.2.1 使用多播 IP 广播数据的例子程序 .....                          | (225)        |
| 21.2.2 监听多播 IP 广播的例子程序 .....                            | (226)        |
| 21.2.3 运行这两个多播 IP 例子程序 .....                            | (227)        |
| 21.3 小结 .....                                           | (228)        |
| <b>第 22 章 非阻塞套接口 I/O .....</b>                          | <b>(229)</b> |
| 22.1 使用非阻塞套接口 I/O 的例子程序 .....                           | (229)        |
| 22.2 运行这个非阻塞套接口 I/O 的例子程序 .....                         | (232)        |
| 22.3 小结 .....                                           | (232)        |
| <b>第 23 章 用于 TCP 套接口的一个 C++ 类库 .....</b>                | <b>(233)</b> |
| 23.1 C++ 客户/服务器类的设计 .....                               | (233)        |
| 23.1.1 理解客户程序的设计 .....                                  | (233)        |
| 23.1.2 理解服务器设计 .....                                    | (235)        |
| 23.2 C++ 客户/服务器类的实现 .....                               | (236)        |
| 23.2.1 实现客户程序 .....                                     | (236)        |
| 23.2.2 实现服务器 .....                                      | (237)        |
| 23.3 测试这个客户/服务器类 .....                                  | (238)        |
| 23.4 小结 .....                                           | (240)        |
| <b>第 24 章 使用库 .....</b>                                 | <b>(241)</b> |
| 24.1 比较 libc5 和 libc6 .....                             | (241)        |
| 24.2 库操作工具 .....                                        | (242)        |
| 24.2.1 理解 nm 命令 .....                                   | (242)        |
| 24.2.2 理解 ar 命令 .....                                   | (242)        |

|               |                       |              |
|---------------|-----------------------|--------------|
| 24.2.3        | 理解 ldd 命令 .....       | (243)        |
| 24.2.4        | 理解 ldconfig .....     | (243)        |
| 24.2.5        | 环境变量和配置文件 .....       | (244)        |
| 24.3          | 编写并使用静态库 .....        | (244)        |
| 24.4          | 编写并使用共享库 .....        | (249)        |
| 24.5          | 使用动态加载的共享对象 .....     | (251)        |
| 24.5.1        | 理解 dl 接口 .....        | (251)        |
| 24.5.2        | 加载共享对象 .....          | (251)        |
| 24.5.3        | 使用共享对象 .....          | (252)        |
| 24.5.4        | 检查错误 .....            | (252)        |
| 24.5.5        | 卸载共享对象 .....          | (252)        |
| 24.5.6        | 使用 dl 接口 .....        | (252)        |
| 24.6          | 小结 .....              | (254)        |
| <b>第 25 章</b> | <b>设备驱动程序 .....</b>   | <b>(255)</b> |
| 25.1          | 驱动程序类型 .....          | (255)        |
| 25.1.1        | 静态链接的内核设备驱动程序 .....   | (255)        |
| 25.1.2        | 可加载的内核模块 .....        | (255)        |
| 25.1.3        | 共享库 .....             | (256)        |
| 25.1.4        | 无特权用户模式程序 .....       | (256)        |
| 25.1.5        | 特权用户模式程序 .....        | (256)        |
| 25.1.6        | 守护进程 .....            | (257)        |
| 25.1.7        | 字符设备与块设备 .....        | (257)        |
| 25.2          | 演示用的硬件 .....          | (257)        |
| 25.2.1        | 步进电机工作原理 .....        | (258)        |
| 25.2.2        | 标准或双向并口 .....         | (260)        |
| 25.3          | 开发配置 .....            | (262)        |
| 25.4          | 低层端口 I/O .....        | (262)        |
| 25.5          | 引发使用设备驱动程序的中断 .....   | (264)        |
| 25.6          | 使用 DMA 访问内存 .....     | (265)        |
| 25.7          | 一个简单的用户模式测试驱动程序 ..... | (265)        |
| 25.8          | 调试内核级驱动程序 .....       | (266)        |
| 25.9          | 下半部与上半部 .....         | (267)        |
| 25.10         | 创建一个内核驱动程序 .....      | (267)        |
| 25.10.1       | 查看源代码 .....           | (267)        |
| 25.10.2       | 编译驱动程序 .....          | (291)        |
| 25.10.3       | 使用内核驱动程序 .....        | (291)        |
| 25.10.4       | 未来发展方向 .....          | (292)        |
| 25.11         | 其他信息来源 .....          | (293)        |
| 25.12         | 小结 .....              | (293)        |

## 第 4 部分 用户界面编程

|               |                     |              |
|---------------|---------------------|--------------|
| <b>第 26 章</b> | <b>底层终端控制 .....</b> | <b>(294)</b> |
|---------------|---------------------|--------------|