

TURING

图灵程序设计丛书

Linux/UNIX系列

WILEY



Beginning Linux Programming 4th Edition

Linux程序设计

第4版

[英] Neil Matthew 著
Richard Stones 译
陈健 宋健建

- 最负盛名的Linux程序设计图书
- 内容全面深入
- 全球开源社区集体智慧结晶



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Linux/UNIX系列



Beginning Linux Programming 4th Edition

Linux程序设计

第4版

[英] Neil Matthew 著
Richard Stones
陈健 宋健建 译

人民邮电出版社
北京

图书在版编目 (CIP) 数据

Linux程序设计 : 第4版 / (英) 马修 (Matthew, N.),
(英) 斯通斯 (Stones, R.) 著 ; 陈健, 宋健建译. --
北京 : 人民邮电出版社, 2010.6

(图灵程序设计丛书)

书名原文: Beginning Linux Programming, 4th
Edition

ISBN 978-7-115-22821-5

I. ①L… II. ①马… ②斯… ③陈… ④宋… III. ①
Linux操作系统—程序设计 IV. ①TP316.89

中国版本图书馆CIP数据核字(2010)第071258号

内 容 提 要

本书讲述了 Linux 系统及其他 UNIX 风格的操作系统上的程序开发, 主要内容包括标准 Linux C 语言函数库和由不同的 Linux 或 UNIX 标准指定的各种工具的使用方法, 大多数标准 Linux 开发工具的使用方法, 通过 DBM 和 MySQL 数据库系统存储 Linux 中的数据, 为 X 视窗系统建立图形化用户界面等。本书通过先介绍程序设计理论, 再以适当的例子和清晰的解释来阐明它的方式, 帮助读者迅速掌握相关的知识。

本书适合 Linux 的初学者及希望利用 Linux 进行开发的程序人员阅读, 也适合作为高等院校计算机相关专业师生的参考教材。

图灵程序设计丛书

Linux程序设计 (第4版)

-
- ◆ 著 [英] Neil Matthew Richard Stones
 - 译 陈 健 宋健建
 - 责任编辑 傅志红
 - 执行编辑 印星星
 - ◆ 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>
 - 中国铁道出版社印刷厂印刷
 - ◆ 开本: 800×1000 1/16
 - 印张: 41.25
 - 字数: 1133千字 2010年6月第1版
 - 印数: 1-4 000册 2010年6月北京第1次印刷
 - 著作权合同登记号 图字: 01-2008-3298号
 - ISBN 978-7-115-22821-5
-

定价: 99.00元

读者服务热线: (010)51095186 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Original edition, entitled *Beginning Linux Programming, 4th Edition*, by Neil Matthew, Richard Stones, ISBN 9780470147627, published by John Wiley & Sons, Inc.

Copyright ©2008 by John Wiley & Sons, Inc., All rights reserved. This translation published under License.

Translation edition published by POSTS & TELECOM PRESS Copyright ©2010.

Copies of this book sold without a Wiley sticker on the cover are unauthorized and illegal.

本书简体中文版由 John Wiley & Sons, Inc. 授权人民邮电出版社独家出版。

本书封底贴有 Wiley & Sons, Inc. 激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

序

所有的计算机程序员都会随手记下大量笔记，其中的代码示例往往来自前人对使用手册的深入钻研，或者来自Usenet新闻组，来自后者的代码有时连最盲目的探索者也不敢照搬照抄（当然也有另一种观点认为，他们都可以自由地访问Usenet新闻组，并且从来没有停止过对其中代码的使用），但采用这种风格的图书可以说少之又少，这不能不说是一件很奇怪的事情。在因特网中，存在着大量针对程序设计和系统管理特定领域的、短小精悍而又切中问题关键的文档。Linux文档项目发表了一系列的文档，内容涵盖了Linux的各个方面，从在同一台机器上同时安装Linux和Windows到将你的咖啡机连接到Linux系统。你可以通过网址<http://www.tldp.org>来查看Linux文档项目。

从另一方面来看，现在的图书市场充斥着大量这样的图书，它们要么是大部头的巨著，内容详尽而全面，使得你没有时间把它们读完；要么就是完全面向初学者的入门图书，你购买它们只是为了送给朋友（开个玩笑而已）。只有很少的书籍尝试着对大量实际应用领域的基本概念和做法进行介绍。本书就是其中之一，它是对程序员笔记的摘要，经过破译（要认清程序员的笔迹可并非易事）和编辑，并将它们有机地组织起来。

本书这一版经过了审阅和更新，反映了目前Linux开发的现状。

Alan Cox
Linux内核维护者

前 言

欢迎阅读本书第4版，这是一本针对在Linux系统和其他UNIX风格的操作系统上进行程序开发的易于使用的指南性读物。

在本书中，我们的目标是介绍对于Linux程序员来说非常重要的主题，这些主题的涵盖面非常广泛。书名中的“beginning”更多的是指书中的内容而不是读者的技能。我们对本书的内容组织进行了精心的安排，以帮助读者更多地了解Linux所提供的功能，而不管读者现有的经验有多少。Linux程序设计是一个很大的领域，我们的目标是对广泛领域中的大量主题都进行介绍，从而让读者在每个主题上都具备足够的入门知识。

读者对象

如果你是一位程序员，希望利用Linux（或UNIX）提供给软件开发者的工具来加快程序开发的进度，尽量减少编程时间并让你的程序充分利用Linux系统所提供的功能，那么本书将非常适合你。书中明确清晰的解释和分步骤的实验，将帮助你迅速提高编程能力和掌握所有的关键技术。

我们假设读者具备一些C或C++语言的编程经验，这些经验可能来自Windows系统或其他一些操作系统。但我们会尽量保持书中示例程序的简单，即便你不是一个C语言编程专家，也可以轻松地阅读本书。如果存在需要直接比较Linux程序设计和C/C++程序设计的情况，我们都会在书中指出。

如果你刚开始学习Linux，请注意，这不是一本介绍Linux安装和配置的图书。如果你想多学习一些Linux系统管理方面的知识，请阅读其他的参考书籍，如Christopher Negus的Linux Bible 2007 Edition Wiley, ISBN 978-0470082799）。

本书的目标是作为一本教程，向读者介绍大多数Linux系统上都有的各种工具和函数/函数库集，同时本书也可以作为一本方便使用的参考手册。本书的特点是简单易懂、内容广泛、示例丰富。

主要内容

本书希望你达到以下几个学习目标。

- 掌握标准Linux C语言函数库和由各种Linux或UNIX标准指定的其他工具的使用方法。
- 掌握如何使用大多数标准Linux开发工具。
- 学会通过DBM和MySQL数据库系统存储Linux中的数据。
- 理解如何为X视窗系统建立图形用户界面。我们将同时使用GTK（GNOME环境的基础）和Qt（KDE环境的基础）函数库。
- 拥有开发自己的实际应用程序的信心和能力。

在讨论这些主题时，我们首先介绍编程理论，然后通过适当的例子和清晰的解释来阐明它。通过这种方式，你可以在第一遍的学习中就能够迅速掌握相关知识。如有必要，你还可以回顾这些内容以重温所有的基本要素。

书中示例程序主要是为了演示一组函数的用法或某些新概念的实际使用情况。贯穿全书有一个大型的示例项目：一个简单的用于记录音乐CD详细资料的数据库应用程序。随着知识面的扩展，你可以按照自己的意愿开发、重新实现和扩展这个项目。虽然如此，这个CD应用程序对本书的任何一章来说都不是必需的，所以只要你愿意也可以忽略它，但我们认为它对书中讨论的技术提供了一些有用的和深入的示范，并且它还有助于讲解每个高级主题。我们对这个应用程序的第一次讨论出现在本书第2章的结尾处，它显示了一个非常大的shell脚本是如何组织的，shell如何处理用户输入、如何构造菜单以及如何存储和检索数据。

在简要介绍完编译程序、链接函数库和访问在线手册的基本概念后，将全面介绍shell编程。然后你将投入到C语言程序设计中，我们在这里讨论的内容包括文件操作、从Linux环境中获取信息、处理终端的输入输出和curses函数库（它使得交互式的输入和输出更易于管理）。最后你将用C语言重新实现CD应用程序。应用程序的设计方法没有变化，但新的代码中将用curses函数库提供一个基于屏幕的用户界面。

接下来我们讨论数据管理。为了学习dbm数据库函数库的使用方法，我们将再次重新实现这个应用程序，但这次实现所采用的设计方法将贯穿本书后续的一些章节。在其后的一章中，我们将介绍数据是如何使用MySQL存储在一个关系数据库中的，并且我们还将在该章的稍后部分重新使用这种数据存储技术，以便读者了解两种技术的区别。随着这些应用程序的规模越来越大，我们接下来需要介绍调试、源代码控制、软件发行和makefile文件等具体内容。

接下来，你将看到不同的Linux进程是如何使用各种技术进行通信的，以及Linux程序是如何使用套接字来支持不同机器之间的TCP/IP网络通信的，包括与使用不同处理器架构的机器之间通信的问题。

在掌握了Linux程序设计的基础之后，我们开始讨论图形化程序的创建方法。我们将通过两章的篇幅来介绍相关内容。首先介绍GTK+工具包，它是GNOME开发环境的基础；然后介绍Qt工具包，它是KDE开发环境的基础。

在本书的最后一章，我们简要介绍了Linux的标准，正是这些标准使得不同厂商的Linux发行版保持了足够的相似性，从而使我们编写的程序可以在不同的Linux发行版上运行。

正如你所期望的，本书还包括许多其他内容，但我们希望这里给出的简单介绍能够让你对将讨论的内容有一个清晰的概念。

准备工作

在本书中，我们将给予读者一种Linux程序设计的体验。为了更好地理解各章的内容，你应该在阅读本书时，实际运行书中的程序示例。这将提供一个很好的编程实践体验，并将启发你创建自己的程序。我们希望读者一边阅读一边在Linux系统上实际操作。

Linux可以用在许多不同的系统上。其适应性使得只要设备中有一个处理器芯片，Linux就可以以这样或那样的方式在其上运行。可以运行Linux系统的设备包括基于Alpha、ARM、IBM Cell、Itanium、PA-RISC、PowerPC、SPARC、SuperH、68k以及各种x86系列处理器（32位和64位）的计算机。

我们使用两台不同配置的Linux系统来编写本书并开发书中的程序示例，所以我们可以确信，只要你的机器可以运行Linux，你就可以很好的利用本书。此外，在本书的技术审核阶段，我们还在其

他版本的Linux系统中测试了书中的全部代码。

我们在编写本书时主要使用的是基于x86的系统，但我们所讨论的内容很少是只适用于x86的。虽然在一台有8 MB内存的486机器上运行Linux也是可能的，但要想成功地运行一个现代Linux发行版并运行本书中的程序示例，我们建议你使用Fedora、openSUSE或Ubuntu等比较流行的Linux发行版的最新版本，并采用它们所推荐的硬件配置。

在软件需求方面，我们建议使用你偏爱的Linux发行版的最新版本，并应用当前更新（大多数厂商会通过自动更新的方式在线提供这些更新）以保证你的系统打上了所有的补丁。Linux和GNU工具集都是以GNU通用公共许可证（GPL）的形式发布的。一个典型的Linux发行版的大多数其他组件也都使用GPL许可证或其他开放源代码许可证之一，这意味着它们都具有某些特性，其中之一就是自由。它们的源代码总是可以被自由获取，没有人可以剥夺这种自由。关于GPL的详细资料请见<http://www.gnu.org/licenses/>。关于开放源代码定义和它所使用的各种许可证的详细资料请见<http://www.opensource.org/>。你总是可以获取到对GNU/Linux的支持——你可以自己研究源代码、雇用他人或购买厂商的付费支持。

源代码

当试验本书中的程序示例时，你可以手工输入所有的代码，也可以使用和本书配套的源代码文件。本书使用的所有程序源代码都可以从<http://www.wrox.com>上下载。在该网站中，你只需找到本书所在页面（通过搜索框或使用书名列表），然后在本书内容介绍页面点击Download Code（下载代码）链接，就可以获得本书的所有源代码了。

因为很多图书都有类似的书名，所以通过ISBN搜索图书是最佳的方式。本书的ISBN为978-0-470-14762-7。

在下载了源代码之后，你就可以用压缩工具对其解压。此外，你也可以访问Wrox代码下载主页（<http://www.wrox.com/dynamic/books/download.aspx>），获取本书和其他Wrox图书的源代码。

代码下载说明

我们尽力向读者提供能够清晰阐明书中所讨论概念的示例程序和代码片段。需要指出的是，为了尽可能地解释清楚书中介绍的新功能，我们将采用一种或两种代码风格。

特别要指出的是，我们并没有对调用的每个函数的返回值进行检查，以判断它是否与我们预期的一样。在真正的应用程序代码中，我们肯定要做这样的检查工作，而读者也应该对错误处理采取严格的措施。（本书的第3章将讨论一些捕获和处理错误的方法。）

GUN 通用公共许可证

书中的所有源代码都遵循GNU通用公共许可证第二版（<http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>）的条款。下面的许可说明适用于本书所有的源代码：

```
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
```

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

排版约定

为了帮助读者更好地理解本书内容，随时把握学习重点，全书将使用以下一些排版约定：

书中像这样的文字框中记录的是一些重要的、不应该被忘记的、非常关键的信息。它们与周边的内容直接相关。

对当前讨论内容的技巧、提示、窍门和旁白都会像这样缩进放置并将字体设置为楷体。

当我们进行介绍时，我们将把一些重要的单词用楷体印刷，需要读者输入的字符用**粗体**印刷。组合键的格式为：**Ctrl+A**。

我们使用3种不同的方式来印刷代码和终端会话：

```
$ who
root      tty1      Sep 10 16:12
rick      tty2      Sep 10 16:10
```

对于命令行，它的样式如上面代码的顶部所示，而它的输出结果则以常规风格印刷。字符\$是提示符（如果命令需要超级用户来执行，则提示符会用字符#来替代），粗体字的文本是需要读者输入的命令，然后按下回车键执行该命令。其后采用相同字体但不是黑体的所有文本都是该黑体字命令的输出。在上面的例子中，你输入命令who，然后将在命令下面看到输出的结果。

Linux定义的函数和结构的原型使用黑体字来印刷，如下所示：

```
#include <stdio.h>

int printf (const char *format, ...);
```

在我们的代码示例中，带有底纹的部分是新的、重要的内容，如下所示：

```
/* 这样印刷的是新的、重要的代码。*/
```

而如果代码采用的是如下所示不带底纹的风格，就表示它的内容没有那么重要：

```
/* 这样印刷的是以前出现过的代码。*/
```

当程序代码的内容在一章中有增加时，后来添加的代码首次出现时以加底纹的风格给出，其后就不再加底纹了。例如，一个新的程序如下所示：

```
/* 代码示例 */
/* 到此结束 */
```

如果我们在该章的稍后部分增加了这个程序的内容，新增代码将带有底纹：

```
/* 代码示例 */
/* 这一行和下一行 */
/* 是新增的代码 */
/* 到此结束 */
```

我们要提到的最后一个约定是，我们在每个程序示例开始之前都会加上一个“实验”标题，其目的是为了将代码分隔开，突出显示其组成部分，同时可以显示应用程序的进度。当我们觉得必要时，还会在代码之后加上“实验解析”部分，来解释代码中与前面理论有关的关键之处。我们发现这两个约定有助于把非常难于理解的代码清单分解为相对简单的部分。

勘误表

我们已经尽力保证本书的文字和程序代码没有任何错误。但是人无完人，错误总是难免的。如果你找到了本书中的错误，比如拼写错误或代码错误，我们将非常感谢可以得到你的反馈。指正错误不仅可以为其他读者节省时间，同时也可以帮助我们提高图书的品质。

要找到本书的勘误表，请访问<http://www.wrox.com>，然后使用搜索框或书名列表来找到本书。在本书的页面，点击Book Errata（图书勘误表）链接。在该链接指向的页面中，你可以看到由Wrox编辑发布的所有针对本书提交的勘误。你也可以通过网址<http://www.wrox.com/misc-pages/booklist.shtml>找到一个完整的图书列表，其中包括指向每本书勘误表的链接。

如果在本书的勘误表中没有找到你发现的错误，可以访问网址<http://www.wrox.com/contact/techsupport.shtml>，填写该页面上的表格以将你发现的错误发送给我们。我们将检查你发送过来的信息，如果它是正确的，我们将在本书的勘误表中发布该信息，并在本书的下一版中修正该问题。

p2p.wrox.com

为参与作者和同行的讨论，你可以加入P2P论坛，它的网址是p2p.wrox.com。这个论坛是一个基于Web的系统，你可以在其上发布与Wrox图书和相关技术有关的消息，并与其他读者和技术用户交流。这个论坛还提供了订阅功能，当有你感兴趣的主题发布时，论坛会通过电子邮件把这些消息发送给你。Wrox的作者、编辑、其他行业专家和与你一样的读者都会到这个论坛探讨一些问题。

在<http://p2p.wrox.com>中，你将找到很多不同的论坛，这些论坛不仅有助于你阅读本书，而且也有助于你开发自己的应用程序。要加入这些论坛，你只需按如下步骤操作即可。

- (1) 访问p2p.wrox.com并点击Register链接。
- (2) 阅读使用条款并点击Agree按钮。
- (3) 填写加入论坛所必需的信息和你想要提供的其他可选信息，然后点击Submit按钮。
- (4) 你将收到一封电子邮件，该邮件告诉你如何验证你的账号并完成加入程序。

注意，不加入P2P论坛也可以阅读论坛中的消息，但是如果你想要发布自己的消息，你必须加入论坛。

加入论坛后，你就可以发布新消息并回复其他用户发布的消息了。你可以在任何时间阅读Web站点上的消息。如果希望某个论坛能将最新的消息通过电子邮件发送给你，你可以点击论坛列表中该论坛名称旁边的Subscribe to this Forum图标。

要获得如何使用Wrox P2P的更多信息，你可以阅读P2P FAQ列表中的问题及其答复，这些问题与论坛软件的工作原理及很多与P2P和Wrox图书相关的常见问题有关。要阅读FAQ，你可以点击任何P2P页面上的FAQ链接。

致谢

感谢许多帮助本书出版的人。

Neil要感谢他的妻子Christine, 谢谢她的理解, 感谢他的孩子Alex和Adrian, 没有抱怨他们的父亲只顾在书房中写作。

Rick要感谢他的妻子Ann以及孩子Jennifer和Andrew, 他们非常理解和支持自己的父亲在晚上和周末要全神贯注地写书。

我们要感谢Wiley出版社的工作人员, 正是他们的努力使得本书的第4版得以发行。感谢Carol Long启动了该项目并整理了合同, 感谢Sara Shlaer杰出的编辑工作和Timothy Boronczyk出色的技术审查。我们还要感谢Jenny Watsonfor找出了书中所有的冗余内容, 并让本书顺利通过管理层的审核, 感谢Bill Barton确保本书完美地组织和呈现, 还要感谢文字编辑Kim Cofer。我们还非常感谢Eric Foster-Johnson对本书第16章和第17章所作的出色工作。可以说, 如果没有大家的共同努力, 本书不可能做到像现在这么好。

我们还要感谢我们的老板, Scientific Generics、Mobicom和Celesio在本书4个版本的出版过程中给予的支持。

最后, 我们要向两位帮助促成本书出版的重要人士致以崇高的敬意。首先是Richard Stallman, 他开发了优秀的GNU工具, 提出了自由软件环境的思想(现在它已通过GNU/Linux成为现实)。第二位是Linus Torvalds, 他开展并持续鼓舞着协同开发工作, 向我们提供了一个不断改善的Linux内核。

目 录

第1章 入门.....	1	2.6.5 命令.....	42
1.1 UNIX、Linux 和 GNU 简介.....	1	2.6.6 命令的执行.....	58
1.1.1 什么是 UNIX.....	1	2.6.7 here 文档.....	61
1.1.2 什么是 Linux.....	2	2.6.8 调试脚本程序.....	63
1.1.3 GNU 项目和自由软件基金会.....	3	2.7 迈向图形化: dialog 工具.....	63
1.1.4 Linux 发行版.....	3	2.8 综合应用.....	68
1.2 Linux 程序设计.....	4	2.8.1 需求.....	68
1.2.1 Linux 程序.....	4	2.8.2 设计.....	68
1.2.2 文本编辑器.....	5	2.8.3 应用程序的说明.....	76
1.2.3 C 语言编译器.....	5	2.9 小结.....	77
1.2.4 开发系统导引.....	7	第3章 文件操作.....	78
1.3 获得帮助.....	12	3.1 Linux 文件结构.....	78
1.4 小结.....	14	3.1.1 目录.....	79
第2章 shell 程序设计.....	15	3.1.2 文件和设备.....	79
2.1 为什么使用 shell 编程.....	15	3.2 系统调用和设备驱动程序.....	80
2.2 一点哲学.....	16	3.3 库函数.....	81
2.3 什么是 shell.....	16	3.4 底层文件访问.....	82
2.4 管道和重定向.....	18	3.4.1 write 系统调用.....	82
2.4.1 重定向输出.....	18	3.4.2 read 系统调用.....	83
2.4.2 重定向输入.....	19	3.4.3 open 系统调用.....	84
2.4.3 管道.....	19	3.4.4 访问权限的初始值.....	85
2.5 作为程序设计语言的 shell.....	20	3.4.5 其他与文件管理有关的系统 调用.....	88
2.5.1 交互式程序.....	20	3.5 标准 I/O 库.....	91
2.5.2 创建脚本.....	21	3.5.1 fopen 函数.....	91
2.5.3 把脚本设置为可执行.....	22	3.5.2 fread 函数.....	92
2.6 shell 的语法.....	23	3.5.3 fwrite 函数.....	92
2.6.1 变量.....	23	3.5.4 fclose 函数.....	92
2.6.2 条件.....	27	3.5.5 fflush 函数.....	93
2.6.3 控制结构.....	29	3.5.6 fseek 函数.....	93
2.6.4 函数.....	39		

3.5.7	fgetc、getc 和 getchar 函数	93
3.5.8	fputc、putc 和 putchar 函数	94
3.5.9	fgets 和 gets 函数	94
3.6	格式化输入和输出	94
3.6.1	printf、fprintf 和 sprintf 函数	94
3.6.2	scanf、fscanf 和 sscanf 函数	96
3.6.3	其他流函数	98
3.6.4	文件流错误	99
3.6.5	文件流和文件描述符	99
3.7	文件和目录的维护	100
3.7.1	chmod 系统调用	100
3.7.2	chown 系统调用	100
3.7.3	unlink、link 和 symlink 系统调用	100
3.7.4	mkdir 和 rmdir 系统调用	101
3.7.5	chdir 系统调用和 getcwd 函数	101
3.8	扫描目录	102
3.8.1	opendir 函数	102
3.8.2	readdir 函数	102
3.8.3	telldir 函数	103
3.8.4	seekdir 函数	103
3.8.5	closedir 函数	103
3.9	错误处理	106
3.9.1	strerror 函数	106
3.9.2	perror 函数	106
3.10	/proc 文件系统	107
3.11	高级主题:fcntl 和 mmap	110
3.11.1	fcntl 系统调用	110
3.11.2	mmap 函数	111
3.12	小结	113
第 4 章	Linux 环境	114
4.1	程序参数	114
4.1.1	getopt	116
4.1.2	getopt_long	118

4.2	环境变量	120
4.2.1	环境变量的用途	122
4.2.2	environ 变量	122
4.3	时间和日期	123
4.4	临时文件	129
4.5	用户信息	131
4.6	主机信息	134
4.7	日志	135
4.8	资源和限制	139
4.9	小结	143

第 5 章 终端 144

5.1	对终端进行读写	144
5.2	与终端进行对话	149
5.3	终端驱动程序和通用终端接口	151
5.3.1	概述	151
5.3.2	硬件模型	151
5.4	termios 结构	152
5.4.1	输入模式	153
5.4.2	输出模式	154
5.4.3	控制模式	155
5.4.4	本地模式	155
5.4.5	特殊控制字符	155
5.4.6	终端速度	158
5.4.7	其他函数	159
5.5	终端的输出	162
5.5.1	终端的类型	163
5.5.2	识别终端类型	163
5.5.3	使用 terminfo 功能标志	165
5.6	检测击键动作	170
5.7	虚拟控制台	172
5.8	伪终端	173
5.9	小结	174

第 6 章 使用 curses 函数库管理基于文本的屏幕 175

6.1	用 curses 函数库进行编译	175
6.2	curses 术语和概念	176
6.3	屏幕	178
6.3.1	输出到屏幕	179
6.3.2	从屏幕读取	180

6.3.3	清除屏幕	180
6.3.4	移动光标	180
6.3.5	字符属性	181
6.4	键盘	183
6.4.1	键盘模式	183
6.4.2	键盘输入	184
6.5	窗口	185
6.5.1	WINDOW 结构	185
6.5.2	通用函数	186
6.5.3	移动和更新窗口	187
6.5.4	优化屏幕刷新	190
6.6	子窗口	191
6.7	keypad 模式	193
6.8	彩色显示	195
6.9	pad	198
6.10	CD 唱片应用程序	200
6.10.1	新 CD 唱片应用程序的开始部分	200
6.10.2	main 函数	202
6.10.3	建立菜单	203
6.10.4	操作数据库文件	204
6.10.5	查询 CD 数据库	209
6.11	小结	213
第 7 章	数据管理	214
7.1	内存管理	214
7.1.1	简单的内存分配	214
7.1.2	分配大量的内存	215
7.1.3	滥用内存	218
7.1.4	空指针	219
7.1.5	释放内存	220
7.1.6	其他内存分配函数	221
7.2	文件锁定	222
7.2.1	创建锁文件	222
7.2.2	区域锁定	225
7.2.3	锁定状态下的读写操作	227
7.2.4	文件锁的竞争	232
7.2.5	其他锁命令	236
7.2.6	死锁	236
7.3	数据库	237

7.3.1	dbm 数据库	237
7.3.2	dbm 例程	238
7.3.3	dbm 访问函数	239
7.3.4	其他 dbm 函数	242
7.4	CD 唱片应用程序	244
7.4.1	更新设计	244
7.4.2	使用 dbm 数据库的 CD 唱片应用程序	245
7.5	小结	262
第 8 章	MySQL	263
8.1	安装	263
8.1.1	MySQL 软件包	264
8.1.2	安装后的配置	266
8.1.3	安装后的故障修复	270
8.2	MySQL 管理	270
8.2.1	命令	270
8.2.2	创建用户并赋予权限	274
8.2.3	密码	276
8.2.4	创建数据库	276
8.2.5	数据类型	277
8.2.6	创建表	278
8.2.7	图形化工具	280
8.3	使用 C 语言访问 MySQL 数据	283
8.3.1	连接例程	283
8.3.2	错误处理	287
8.3.3	执行 SQL 语句	288
8.3.4	更多的函数	301
8.4	CD 数据库应用程序	301
8.4.1	创建表	302
8.4.2	添加数据	304
8.4.3	使用 C 语言访问数据	306
8.5	小结	316
第 9 章	开发工具	317
9.1	多个源文件带来的问题	317
9.2	make 命令和 makefile 文件	318
9.2.1	makefile 的语法	318
9.2.2	make 命令的选项和参数	318
9.2.3	makefile 文件中的注释	321
9.2.4	makefile 文件中的宏	321

9.2.5 多个目标	323	10.3.5 列出程序源代码	370
9.2.6 内置规则	325	10.3.6 设置断点	371
9.2.7 后缀和模式规则	326	10.3.7 用调试器打补丁	374
9.2.8 用 make 管理函数库	327	10.3.8 深入学习 gdb	375
9.2.9 高级主题: makefile 文件和子目录	329	10.4 其他调试工具	375
9.2.10 GNU make 和 gcc	329	10.4.1 lint: 清理程序中的“垃圾”	376
9.3 源代码控制	330	10.4.2 函数调用工具	378
9.3.1 RCS	331	10.4.3 用 prof/gprof 产生执行存档	380
9.3.2 SCCS	336	10.5 断言	381
9.3.3 RCS 和 SCCS 的比较	336	10.6 内存调试	383
9.3.4 CVS	337	10.6.1 ElectricFence 函数库	383
9.3.5 CVS 的前端程序	340	10.6.2 valgrind	384
9.3.6 Subversion	341	10.7 小结	387
9.4 编写手册页	342	第 11 章 进程和信号	388
9.5 发行软件	345	11.1 什么是进程	388
9.5.1 patch 程序	345	11.2 进程的结构	388
9.5.2 其他软件发行工具	347	11.2.1 进程表	390
9.6 RPM 软件包	349	11.2.2 查看进程	390
9.6.1 使用 RPM 软件包文件	349	11.2.3 系统进程	391
9.6.2 安装 RPM 软件包	350	11.2.4 进程调度	393
9.6.3 创建 RPM 软件包	350	11.3 启动新进程	394
9.7 其他软件包格式	358	11.3.1 等待一个进程	399
9.8 开发环境	358	11.3.2 僵尸进程	401
9.8.1 KDevelop	358	11.3.3 输入和输出重定向	403
9.8.2 其他开发环境	360	11.3.4 线程	404
9.9 小结	360	11.4 信号	404
第 10 章 调试	361	11.4.1 发送信号	408
10.1 错误类型	361	11.4.2 信号集	412
10.2 常用调试技巧	362	11.5 小结	415
10.2.1 有漏洞的程序	362	第 12 章 POSIX 线程	416
10.2.2 代码检查	364	12.1 什么是线程	416
10.2.3 取样法	365	12.2 线程的优点和缺点	417
10.2.4 程序的受控执行	367	12.3 第一个线程程序	418
10.3 使用 gdb 进行调试	368	12.4 同时执行	421
10.3.1 启动 gdb	368	12.5 同步	423
10.3.2 运行一个程序	369	12.5.1 用信号量进行同步	423
10.3.3 栈跟踪	369	12.5.2 用互斥量进行同步	427
10.3.4 检查变量	370		

12.6	线程的属性	431	14.3	消息队列	502
12.7	取消一个线程	435	14.3.1	msgget 函数	502
12.8	多线程	438	14.3.2	msgsnd 函数	503
12.9	小结	442	14.3.3	msgrcv 函数	503
第 13 章	进程间通信: 管道	443	14.3.4	msgctl 函数	504
13.1	什么是管道	443	14.4	CD 数据库应用程序	507
13.2	进程管道	444	14.4.1	修改服务器函数	507
13.3	将输出送往 popen	445	14.4.2	修改客户函数	509
13.3.1	传递更多的数据	446	14.5	IPC 状态命令	511
13.3.2	如何实现 popen	447	14.5.1	显示信号量状态	511
13.4	pipe 调用	449	14.5.2	显示共享内存状态	511
13.5	父进程和子进程	451	14.5.3	显示消息队列状态	511
13.5.1	管道关闭后的读操作	453	14.6	小结	512
13.5.2	把管道用作标准输入和标准输出	454	第 15 章	套接字	513
13.6	命名管道: FIFO	456	15.1	什么是套接字	513
13.6.1	访问 FIFO 文件	458	15.2	套接字连接	513
13.6.2	高级主题: 使用 FIFO 的客户/服务器应用程序	464	15.2.1	套接字属性	517
13.7	CD 数据库应用程序	468	15.2.2	创建套接字	519
13.7.1	目标	469	15.2.3	套接字地址	520
13.7.2	实现	469	15.2.4	命名套接字	520
13.7.3	客户接口函数	472	15.2.5	创建套接字队列	521
13.7.4	服务器接口 server.c	478	15.2.6	接受连接	521
13.7.5	管道	481	15.2.7	请求连接	522
13.7.6	对 CD 数据库应用程序的总结	487	15.2.8	关闭套接字	523
13.8	小结	487	15.2.9	套接字通信	523
第 14 章	信号量、共享内存和消息队列	488	15.2.10	主机字节序和网络字节序	525
14.1	信号量	488	15.3	网络信息	527
14.1.1	信号量的定义	489	15.3.1	因特网守护进程 (xinetd/inetd)	531
14.1.2	一个理论性的例子	489	15.3.2	套接字选项	533
14.1.3	Linux 的信号量机制	490	15.4	多客户	534
14.1.4	使用信号量	492	15.4.1	select 系统调用	537
14.2	共享内存	496	15.4.2	多客户	540
14.2.1	shmget 函数	497	15.5	数据报	543
14.2.2	shmat 函数	497	15.6	小结	545
14.2.3	shmdt	498	第 16 章	用 GTK+ 进行 GNOME 编程	546
14.2.4	shmctl	498	16.1	X 视窗系统简介	546
			16.1.1	X 服务器	547
			16.1.2	X 客户端	547

16.1.3	X 协议	547	17.4	Qt 构件	602
16.1.4	Xlib 库	547	17.4.1	QLineEdit	602
16.1.5	X 工具包	547	17.4.2	Qt 按钮	605
16.1.6	窗口管理器	548	17.4.3	QComboBox	609
16.1.7	创建 GUI 的其他方法—— 平台无关的窗口 API	548	17.4.4	QListView	613
16.2	GTK+简介	549	17.5	对话框	615
16.2.1	GLib 类型系统	549	17.5.1	QDialog	615
16.2.2	GTK+对象系统	550	17.5.2	QMessageBox	617
16.2.3	GNOME 简介	550	17.5.3	QInputDialog	618
16.2.4	安装 GNOME/GTK+开发库	551	17.5.4	使用 qmake 简化 makefile 文件的编写	620
16.3	事件、信号和回调函数	554	17.6	KDE 的菜单和工具栏	620
16.4	组装盒构件	556	17.7	使用 KDE/Qt 编写 CD 数据库应用 程序	624
16.5	GTK+构件	559	17.7.1	主窗口	624
16.5.1	GtkWindow	559	17.7.2	AddCdDialog	627
16.5.2	GtkEntry	560	17.7.3	LogonDialog	629
16.5.3	GtkSpinButton	563	17.7.4	main.cpp	630
16.5.4	GtkButton	565	17.8	小结	632
16.5.5	GtkTreeView	568	第 18 章	Linux 标准	633
16.6	GNOME 构件	571	18.1	C 编程语言	634
16.7	GNOME 菜单	572	18.1.1	发展历史简介	634
16.8	对话框	576	18.1.2	GNU 编译器集	634
16.8.1	GtkDialog	577	18.1.3	gcc 选项	635
16.8.2	模式对话框	578	18.2	接口和 LSB	636
16.8.3	非模式对话框	579	18.2.1	LSB 标准函数库	637
16.8.4	GtkMessageDialog	580	18.2.2	LSB 用户和组	638
16.9	CD 数据库应用程序	581	18.2.3	LSB 系统初始化	638
16.10	小结	592	18.3	文件系统层次结构标准	639
第 17 章	用 Qt 进行 KDE 编程	593	18.4	更多标准	642
17.1	KDE 和 Qt 简介	593	18.5	小结	642
17.2	安装 Qt	594			
17.3	信号和槽	596			