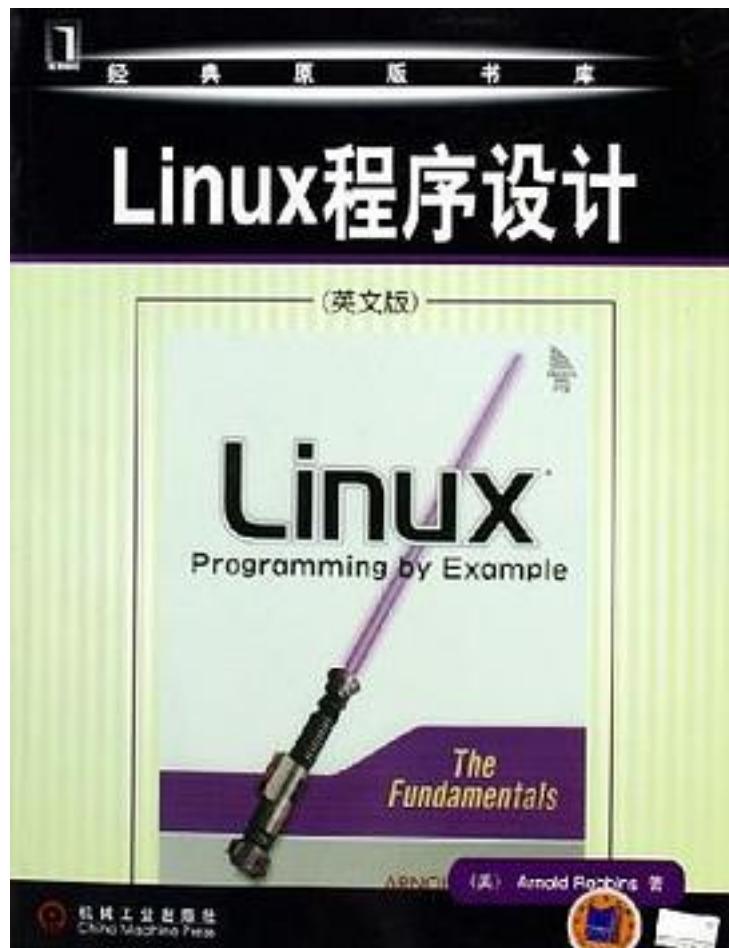


# Linux程序设计



[Linux程序设计 下载链接1](#)

著者:Neil Matthew

出版者:人民邮电出版社

出版时间:201005

装帧:平装

isbn:9787115228215

时至今日，Linux系统已经从一个个人作品发展为可以用于各种关键任务的成熟、高效和稳定的操作系统，因为具备跨平台、开源、支持众多应用软件和网络协议等优点，它得到了各大主流软硬件厂商的支持，也成为广大程序设计人员理想的开发平台。

本书是Linux程序设计领域的经典名著，以简单易懂、内容全面和示例丰富而受到广泛好评。中文版前两版出版后，在国内的Linux爱好者和程序员中也引起了强烈反响，这一热潮一直持续至今。本书是国内读者翘首以待的第4版，此次新版内容组织更加严谨，译者更是细心雕琢，保留了这部权威著作的原汁原味。

对Linux所提供的功能全面而准确的阐述，以及贯穿全书的示例程序体验，使本书不仅成为初学者的最佳Linux程序设计指南，而且是中高级程序员不可或缺的参考书。

## 作者介绍:

Neil Matthew 和Richard

Stones世界知名的Linux/UNIX专家，有数十年Linux/UNIX开发经验和从业经历。他们使用过几乎所有UNIX版本，并精通C/C++、LISP、Fortran、Perl、Tcl和Prolog等各种语言。他们从事过各种软件项目，从实时嵌入式系统到会计系统和零售信息系统。除本书外，他们还合著了PostgreSQL、MySQL方面的图书。

## 目录: 第1章 入门

- 1.1 UNIX、Linux和GNU简介
  - 1.1.1 什么是UNIX
  - 1.1.2 什么是Linux
  - 1.1.3 GNU项目和自由软件基金会
  - 1.1.4 Linux发行版
- 1.2 Linux程序设计
  - 1.2.1 Linux程序
  - 1.2.2 文本编辑器
  - 1.2.3 C语言编译器
  - 1.2.4 开发系统导引
- 1.3 获得帮助
- 1.4 小结

## 第2章 shell程序设计

- 2.1 为什么使用shell编程
- 2.2 一点哲学
- 2.3 什么是shell
- 2.4 管道和重定向
  - 2.4.1 重定向输出
  - 2.4.2 重定向输入
  - 2.4.3 管道
- 2.5 作为程序设计语言的shell
  - 2.5.1 交互式程序
  - 2.5.2 创建脚本
  - 2.5.3 把脚本设置为可执行
- 2.6 shell的语法
  - 2.6.1 变量
  - 2.6.2 条件
  - 2.6.3 控制结构
  - 2.6.4 函数
  - 2.6.5 命令
  - 2.6.6 命令的执行
  - 2.6.7 here文档
  - 2.6.8 调试脚本程序
- 2.7 迈向图形化：dialog工具
- 2.8 综合应用

2.8.1 需求

2.8.2 设计

2.8.3 应用程序的说明

2.9 小结

第3章 文件操作

3.1 Linux文件结构

3.1.1 目录

3.1.2 文件和设备

3.2 系统调用和设备驱动程序

3.3 库函数

3.4 底层文件访问

3.4.1 write系统调用

3.4.2 read系统调用

3.4.3 open系统调用

3.4.4 访问权限的初始值

3.4.5 其他与文件管理有关的系统调用

3.5 标准I/O库

3.5.1 fopen函数

3.5.2 fread函数

3.5.3 fwrite函数

3.5.4 fclose函数

3.5.5 fflush函数

3.5.6 fseek函数

3.5.7 fgetc、getc和getchar函数

3.5.8 fputc、putc和putchar函数

3.5.9 fgets和gets函数

3.6 格式化输入和输出

3.6.1 printf、fprintf和sprintf函数

3.6.2 scanf、fscanf和sscanf函数

3.6.3 其他流函数

3.6.4 文件流错误

3.6.5 文件流和文件描述符

3.7 文件和目录的维护

3.7.1 chmod系统调用

3.7.2 chown系统调用

3.7.3 unlink、link和symlink系统调用

3.7.4 mkdir和rmdir系统调用

3.7.5 chdir系统调用和getcwd函数

3.8 扫描目录

3.8.1 opendir函数

3.8.2 readdir函数

3.8.3 telldir函数

3.8.4 seekdir函数

3.8.5 closedir函数

3.9 错误处理

3.9.1 strerror函数

3.9.2 perror函数

3.10 /proc文件系统

3.11 高级主题：fcntl和mmap

3.11.1 fcntl系统调用

3.11.2 mmap函数

3.12 小结

第4章 Linux环境

4.1 程序参数

- 4.1.1 getopt
  - 4.1.2 getopt\_long
  - 4.2 环境变量
    - 4.2.1 环境变量的用途
    - 4.2.2 environ变量
  - 4.3 时间和日期
  - 4.4 临时文件
  - 4.5 用户信息
  - 4.6 主机信息
  - 4.7 日志
  - 4.8 资源和限制
  - 4.9 小结
- 第5章 终端
- 5.1 对终端进行读写
  - 5.2 与终端进行对话
  - 5.3 终端驱动程序和通用终端接口
    - 5.3.1 概述
    - 5.3.2 硬件模型
    - 5.4 termios结构
      - 5.4.1 输入模式
      - 5.4.2 输出模式
      - 5.4.3 控制模式
      - 5.4.4 本地模式
      - 5.4.5 特殊控制字符
      - 5.4.6 终端速度
      - 5.4.7 其他函数
    - 5.5 终端的输出
      - 5.5.1 终端的类型
      - 5.5.2 识别终端类型
      - 5.5.3 使用terminfo功能标志
    - 5.6 检测击键动作
    - 5.7 虚拟控制台
    - 5.8 伪终端
    - 5.9 小结
- 第6章 使用curses函数库管理基于文本的屏幕
- 6.1 用curses函数库进行编译
  - 6.2 curses术语和概念
  - 6.3 屏幕
    - 6.3.1 输出到屏幕
    - 6.3.2 从屏幕读取
    - 6.3.3 清除屏幕
    - 6.3.4 移动光标
    - 6.3.5 字符属性
  - 6.4 键盘
    - 6.4.1 键盘模式
    - 6.4.2 键盘输入
  - 6.5 窗口
    - 6.5.1 WINDOW结构
    - 6.5.2 通用函数
    - 6.5.3 移动和更新窗口
    - 6.5.4 优化屏幕刷新
  - 6.6 子窗口
  - 6.7 keypad模式
  - 6.8 彩色显示

- 6.9 pad
- 6.10 CD唱片应用程序
  - 6.10.1 新CD唱片应用程序的开始部分
  - 6.10.2 main函数
  - 6.10.3 建立菜单
  - 6.10.4 操作数据库文件
  - 6.10.5 查询CD数据库
- 6.11 小结
- 第7章 数据管理
  - 7.1 内存管理
    - 7.1.1 简单的内存分配
    - 7.1.2 分配大量的内存
    - 7.1.3 滥用内存
    - 7.1.4 空指针
    - 7.1.5 释放内存
    - 7.1.6 其他内存分配函数
  - 7.2 文件锁定
    - 7.2.1 创建锁文件
    - 7.2.2 区域锁定
    - 7.2.3 锁定状态下的读写操作
    - 7.2.4 文件锁的竞争
    - 7.2.5 其他锁命令
    - 7.2.6 死锁
  - 7.3 数据库
    - 7.3.1 dbm数据库
    - 7.3.2 dbm例程
    - 7.3.3 dbm访问函数
    - 7.3.4 其他dbm函数
  - 7.4 CD唱片应用程序
    - 7.4.1 更新设计
    - 7.4.2 使用dbm数据库的CD唱片应用程序
- 7.5 小结

## 8.5 小结

### 第9章 开发工具

9.1 多个源文件带来的问题

9.2 make命令和makefile文件

9.2.1 makefile的语法

9.2.2 make命令的选项和参数

9.2.3 makefile文件中的注释

9.2.4 makefile文件中的宏

9.2.5 多个目标

9.2.6 内置规则

9.2.7 后缀和模式规则

9.2.8 用make管理函数库

9.2.9 高级主题：makefile文件和子目录

9.2.10 GNU make和gcc

### 9.3 源代码控制

9.3.1 RCS

9.3.2 SCCS

9.3.3 RCS和SCCS的比较

9.3.4 CVS

9.3.5 CVS的前端程序

9.3.6 Subversion

### 9.4 编写手册页

### 9.5 发行软件

9.5.1 patch程序

9.5.2 其他软件发行工具

9.6 RPM软件包

9.6.1 使用RPM软件包文件

9.6.2 安装RPM软件包

9.6.3 创建RPM软件包

9.7 其他软件包格式

9.8 开发环境

9.8.1 KDevelop

9.8.2 其他开发环境

## 9.9 小结

### 第10章 调试

10.1 错误类型

10.2 常用调试技巧

10.2.1 有漏洞的程序

10.2.2 代码检查

10.2.3 取样法

10.2.4 程序的受控执行

10.3 使用gdb进行调试

10.3.1 启动gdb

10.3.2 运行一个程序

10.3.3 栈跟踪

10.3.4 检查变量

10.3.5 列出程序源代码

10.3.6 设置断点

10.3.7 用调试器打补丁

10.3.8 深入学习gdb

10.4 其他调试工具

10.4.1 lint：清理程序中的“垃圾”

10.4.2 函数调用工具

10.4.3 用prof/gprof产生执行存档

10.5 断言

10.6 内存调试

10.6.1 ElectricFence函数库

10.6.2 valgrind

10.7 小结

第11章 进程和信号

11.1 什么是进程

11.2 进程的结构

11.2.1 进程表

11.2.2 查看进程

11.2.3 系统进程

11.2.4 进程调度

11.3 启动新进程

11.3.1 等待一个进程

11.3.2 僵尸进程

11.3.3 输入和输出重定向

11.3.4 线程

11.4 信号

11.4.1 发送信号

11.4.2 信号集

11.5 小结

第12章 POSIX线程

12.1 什么是线程

12.2 线程的优点和缺点

12.3 第一个线程程序

12.4 同时执行

12.5 同步

12.5.1 用信号量进行同步

12.5.2 用互斥量进行同步

12.6 线程的属性

12.7 取消一个线程

12.8 多线程

12.9 小结

第13章 进程间通信：管道

13.1 什么是管道

13.2 进程管道

13.3 将输出送往popen

13.3.1 传递更多的数据

13.3.2 如何实现popen

13.4 pipe调用

13.5 父进程和子进程

13.5.1 管道关闭后的读操作

13.5.2 把管道用作标准输入和标准输出

13.6 命名管道：FIFO

13.6.1 访问FIFO文件

13.6.2 高级主题：使用FIFO的客户/服务器应用程序

13.7 CD数据库应用程序

13.7.1 目标

13.7.2 实现

13.7.3 客户接口函数

13.7.4 服务器接口server.c

13.7.5 管道

13.7.6 对CD数据库应用程序的总结

13.8 小结

## 第14章 信号量、共享内存和消息队列

### 14.1 信号量

#### 14.1.1 信号量的定义

#### 14.1.2 一个理论性的例子

#### 14.1.3 Linux的信号量机制

#### 14.1.4 使用信号量

### 14.2 共享内存

#### 14.2.1 shmget函数

#### 14.2.2 shmat函数

#### 14.2.3 shmdt

#### 14.2.4 shmctl

### 14.3 消息队列

#### 14.3.1 msgget函数

#### 14.3.2 msgsnd函数

#### 14.3.3 msgrcv函数

#### 14.3.4 msgctl函数

### 14.4 CD数据库应用程序

#### 14.4.1 修改服务器函数

#### 14.4.2 修改客户函数

### 14.5 IPC状态命令

#### 14.5.1 显示信号量状态

#### 14.5.2 显示共享内存状态

#### 14.5.3 显示消息队列状态

### 14.6 小结

## 第15章 套接字

### 15.1 什么是套接字

### 15.2 套接字连接

#### 15.2.1 套接字属性

#### 15.2.2 创建套接字

#### 15.2.3 套接字地址

#### 15.2.4 命名套接字

#### 15.2.5 创建套接字队列

#### 15.2.6 接受连接

#### 15.2.7 请求连接

#### 15.2.8 关闭套接字

#### 15.2.9 套接字通信

#### 15.2.10 主机字节序和网络字节序

### 15.3 网络信息

#### 15.3.1 因特网守护进程(xinetd/inetd)

#### 15.3.2 套接字选项

### 15.4 多客户

#### 15.4.1 select系统调用

#### 15.4.2 多客户

### 15.5 数据报

### 15.6 小结

## 第16章 用GTK+进行GNOME编程

### 16.1 X视窗系统简介

#### 16.1.1 X服务器

#### 16.1.2 X客户端

#### 16.1.3 X协议

#### 16.1.4 Xlib库

#### 16.1.5 X工具包

#### 16.1.6 窗口管理器

#### 16.1.7 创建GUI的其他方法——平台无关的窗口API

16.2 GTK+简介  
16.2.1 GLib类型系统  
16.2.2 GTK+对象系统  
16.2.3 GNOME简介  
16.2.4 安装GNOME/GTK+开发库  
16.3 事件、信号和回调函数  
16.4 组装盒构件  
16.5 GTK+构件  
16.5.1 GtkWindow  
16.5.2 GtkEntry  
16.5.3 GtkSpinButton  
16.5.4 GtkButton  
16.5.5 GtkTreeView  
16.6 GNOME构件  
16.7 GNOME菜单  
16.8 对话框  
16.8.1 GtkDialog  
16.8.2 模式对话框  
16.8.3 非模式对话框  
16.8.4 GtkMessageDialog  
16.9 CD数据库应用程序  
16.10 小结

第17章 用Qt进行KDE编程  
17.1 KDE和Qt简介  
17.2 安装Qt  
17.3 信号和槽  
17.4 Qt构件  
17.4.1 QLineEdit  
17.4.2 Qt按钮  
17.4.3 QComboBox  
17.4.4 QListView  
17.5 对话框  
17.5.1 QDialog  
17.5.2 QMessageBox  
17.5.3 QInputDialog  
17.5.4 使用qmake简化makefile文件的编写

17.6 KDE的菜单和工具栏  
17.7 使用KDE/Qt编写CD数据库应用程序

17.7.1 主窗口  
17.7.2 AddCdDialog  
17.7.3 LogonDialog  
17.7.4 main.cpp

17.8 小结

第18章 Linux标准

18.1 C编程语言  
18.1.1 发展历史简介  
18.1.2 GNU编译器集  
18.1.3 gcc选项

18.2 接口和LSB  
18.2.1 LSB标准函数库  
18.2.2 LSB用户和组  
18.2.3 LSB系统初始化  
18.3 文件系统层次结构标准  
18.4 更多标准

18.5 小结  
• • • • • (收起)

[Linux程序设计](#) [下载链接1](#)

## 标签

Linux

程序设计

编程

计算机

操作系统

C

Programming

linux

## 评论

比UNIX环境高级编程(第3版)更轻松

---

linux入门，把需要的部分，跟着书本敲一遍，就能有大概的理解。

---

[先前所读]基本通读，不算很难，但学到的东西也不错。

linux编程入门，get hands dirty!

缺点在于有些内容陈旧了，比如代码里有一段c的函数声明

翻看同事的，并不是每个章节都有看过，用到的时候再好好看看。

短小精悍，没有steven高级编程的事无巨细的讲解，却非常容易抓住关键点，配合steven的高级环境编程就完美了。

经典的入门书。内容丰富，值得一看。

不错的书,我觉得和深入理解计算机系统一起学会更好  
还有我认为这也可以当linux的入门书,不看鸟哥,逆向学习linux  
不过代码里还有些错误,有时间把错误登到我的网站上

入门还行，看之前已有杂乱的底子，对我而言，看这本书是系统的梳理。

很粗糙很快的读完了，了解了个大概。

以前在HW时读过。嗯，还是APUE合口味。

终于翻完一本正经书了

讲的很宽泛，都不深入，只适合入门，要想真正研究Linux的程序设计，还得APUE啊

简单的过了一遍，起码对linux程序代码不再陌生了，具体读代码时还需要查手册，要熟练掌握还是要把书里面的代码自己跑一遍才行。

## 案头工具书

基础的系统调用介绍，涉及范围挺广，学着很带劲

初学，入门用。个人感觉和鸟哥的基础篇差不多。

鸟哥的私房菜可以不用看了。从Shell开始吧

简单的入门书籍，APUE的铺垫

啰里巴嗦，慢慢悠悠，完全不珍惜读者时间，，给外行读也行，可又彪特别底层的函数，作者是想写给谁读的？前四章总一篇就能说完，完全没有深度，有没有归纳写得散漫当清单查也不行

[Linux程序设计 下载链接1](#)

## 书评

《Linux程序设计》是我的Linux编程入门书籍，也是做为教材使用了一整个学期，在阅读和学习这本书的时候产生了很多的疑问，书里也没有对应的解答，直到……直到我看了APUE，带着这些问题去学习APUE，产生了巨大的能量。总之，推荐这本书，但是这本书也只是入门书籍，站在《Linux…

今天终于读完了这本书,这本书围绕begining展开,对Linux程序设计里的广泛领域的大量主题都进行了介绍,作为一本linux开发领域的入门级和指南性的读物,强烈推荐那些想要进行Linux开发的菜鸟作为自己的入门拐杖,这本书所探讨的主题都具有很高的使用价值,抓大放小,没有把初学者带入...

这本书已经读了一多半了,是一本很入门的书籍,但是在翻译上确实比较绕口。可能是译者功力不够,很多东西中文看不明白,反而是英语能看明白。在内容上,每一章都是一个非常简要的介绍,任何一章拿出来都有事一本书。在这里,真的想批评一下国内的一些译者,不但毁了大师的...

虽然不如APUE那一类书出名,作为Linux初学者,这本书绝对是最佳选择;从shell一直讲到x-window编程,语言平易近人,确实是一本难得的好书,就是价格高了些,许多Linux的技巧性知识在书中也有提及。

讲解细致,语言浅显易懂,但需要C语言基础,哎,为什么大学的老师不用这些课本当教材...如果本科的时候可以看见这本书,不知道可以少走多少弯路...  
对于想学习linux下系统编程基本知识的人,强烈推荐!

代码的例子倒是非常的好,不知道是不是由于翻译的问题,感觉读起来很乱,需要自己理清思路然后读起来就很顺畅了。书里内容很基础,我是一个没有什么linux编程的底子的人,以前最多也就是用个shell,看了这本书后感觉linux方面功力大大提升,不再是单纯的使用了,由于主要是做高...

由于是两个作者共同完成的,如果你打算把它当做入门书籍,从第一章开始读到最后一篇,那会很吃力,而且本书的翻译比较拗口,更是增加了难度。  
另外从编排上,个人觉得没有多少连贯性,只要有一点linux基础的人,都可以从任何章节开始读。这本书内容非常多,但是没有与厚度成正...

粗略比较了一下第三版和第四版,似乎没有太大变化,第四版还少了讲驱动的一章。不明白作者为什么要这样改,前言里也没有提到。

可以将这本书作为学习linux编程的入门级书籍，讲得不深，涉及到的知识面还行，但要深入的话，建议参照《UNIX环境高级编程》来读。  
我读的是英文版的，语言不难，有些地方参考了第三版中文版来读，第三版的翻译不错，不知道第四版咋样，但应该不会差，推荐初学者看。

确实是很好的一本Linux初级读物。我从06年开始接触Ubuntu Linux，记得当时版本还只是5.10。

接着，07-08年开始在UNIX下写各种程序。再后来（09-10）转入计算机图形学的研究，醉心于Windows操作系统下的DirectX渲染技术。

很幸运，在2011年春开始返回到我所热爱的Linux编程领域...

大致翻完了这本书。

内容非常全面，对linux环境下编程相关的各个方面都做了清晰的论述。基本如shell，文件，开发工具，调试，高级如线程，套接字，数据库，图形编程都有涉及。深入浅出，像推开了一扇门，外面的青山流水小桥人家都尽入眼底，就只等你跨步出去。是入门学习的极...

实验室的师兄极力推荐的一本书，看了下目录觉得条理清晰，不过拿到的是英文电子版的，很难静下心来去认真读。

我想学编程，可是每次买一本书，刚开始翻的时候信心十足，翻着翻着，信心就一点点往下减。真是看得要累晕掉了。这本书又这样了，听说是属于编程经典一类的，刚买来挺高兴的。就怕自己坚持不了几天。昨天我哥在网上找到一个猎豹网校，让我看一下。你别说，正好有我要学的这门课...

[Linux程序设计 下载链接1](#)