

Index

A

- a command**, vi, 151
- .a extension**, 73
- a modifier, print and x, gdb**, 543
- ABI (Application Binary Interface)**
 - incompatible, 97–98
 - for system calls, 225
- abort function**, 589–590
- accept function**, 388, 391
- Acrobat Reader**, 131
- Add/remove feature in revision control**, 191
- addr2line utility**, 588–589
- Adopted processes**, 243
- Advanced Programmable Interrupt Controller (APIC)**, 462–463
- Advisory locks**, 362
- Affinity masks**, 507–509
- aic79xx module**, 137–138
- alarm function**, 237
- all target**, 77
- Ampersands (&) in Ex mode**, 159
- Analyzing code**
 - cflow for, 207–210
 - compiler for, 214–216
 - ctags for, 210
- Anonymous mapping**, 73, 302, 308
- Anticipatory I/O scheduler**, 284–285
- API (Application Programming Interface)**, 225
- APIC (Advanced Programmable Interrupt Controller)**, 462–463
- Apostrophes (‘)**
 - Ex mode, 159
 - vi, 149–150
- Application Binary Interface (ABI)**
 - incompatible, 97–98
 - for system calls, 225
- Application performance**, 475
 - gcov, 491–494
 - gprof, 488–491
 - instruction efficiency, 480–484
 - ltrace, 484–485
 - OProfile, 494–501
 - processor architecture, 476–480
 - strace, 485–487
 - time, 475–476
- Application Programming Interface (API)**, 225
- Application timing**, 255–256
- apropos command**, 118, 122
- apt-cache command**, 35
- apt-get command**, 35
- apt-key command**, 35
- apt-setup command**, 35
- Apt tool**, 34–35
- ar files**, 5, 12
- Archive files**, 4–5
 - extracting, 11
 - identifying, 6–7
 - querying, 7–11
- as scheduler**, 285
- ascii man page**, 113
- Assembly language**
 - inline, 292
 - in linker process, 68
 - system calls in, 224–225
- assert function**, 589–590
- astyle artistic style**, 206–207
- attach command**, gdb, 553

Attaching running processes, 553

__attribute__ directive, 524–525

Authentication, 19–21

digital signatures for, 21–25

problems in, 25–26

Autocompletion feature, 145

Kate, 183

vi and Vim, 147, 169

autoconf script, 82

Autoindentation

Emacs, 171

Vim, 167

automake tool, 77

Automatic indenting, 145

Automatic storage, 338

Automatic variables, 57–58

auxv files, 344

awatch watchpoints, 540

awk script, 322

B

b command, vi, 150

b suffix, x, gdb, 545

Background processes, 244

Backticks (`), 49, 60–61

backtrace command, gdb, 542, 555

backtrace_symbols function, 588

Backtraces, 587–589

basename function, make, 60

basic.S listing, 224

bcopy function, 112

Beautifiers and browsers, source code,

203–204

astyle, 206–207

cflow, 207–210

compiler, 214–216

cscope, 211

ctags, 210

Doxygen, 212–213

indent code, 204–206

Big Endian byte order, 424, 426

Binaries, miscellaneous, 324–326

Binary packages, 14–16

Binary semaphores, 403

—**bindir option, 75**

binutils package, 569

Black box for debugging, 584–587

Block devices, 257–258, 274–276

Blocked signals, 373

Blocking

in process scheduler, 228–229

timing with, 248–249

Bookmarks, 175

boot man page, 113

bootparam man page, 113

Braces ({})

matching, 144, 182

missing, 93–94

variables, 52

Branching in revision control, 192

break command, gdb, 531, 558

breakpoint command, gdb, 532

Breakpoints

conditional, 534–535

gdb, 531–532

setting, 535–537

syntax, 532–534

and templates, 548–549

brk system call, 310

Browsing and building code

Emacs, 175–176

Vim, 164–166

Brute-force man page matching, 110

bsd autoindentation style, 171

BSD syntax, 435

Buffers

Emacs, 173

file streams, 516–517

file systems, 517–519

I/O devices, 276–280

overflows, 555–556, 571–572

stack, 338

text editor, 145

trace, 584–587

Builds, 41–42

- alternative approaches, 43–44
- background, 42
- documentation, 135–136
- errors and warnings, 78
 - build stage*, 83–86
 - compiler errors*, 86–88
 - compiler warnings*, 88–98
 - configure stage*, 82–83
 - linker errors*, 98–100
 - Makefile mistakes*, 79–82
- GNU tools, 43, 74–77
- imake, 42–43
- libraries for, 69–73
- linking, 67–69
- make program. *See* make program and Makefiles
- process, 74

Bus contention, 459–461**Bus error (SIGBUS) signal, 554****bvi editor, 180, 421–422****bzip2 tool, 4****C****C command**

- Emacs, 173
- vi, 153

C language

- compiler warnings, 93–94
- file streams, 516–517
- preprocessor, 81

C++ language

- breakpoints, 535–537
- compiler warnings, 95–98
- STL, 549–551
- and templates, 548–549

c modifier, print and x, gdb, 543**cache-miss.c listing, 478–480****Cache misses, 458, 481, 483****cachegrind tool, 480–481****Caches**

- CPU, 291–295
- I/O devices, 276–280
- memory, 295–296
- multiprocessor performance, 504

call command, gdb, 547**call function, make, 63****callgrind tool, 580****cb filter, 204****cfile command, Vim, 166****CFLAGS, 66****cfow command, 207–210****cfq scheduler, 285–286****Change commands, vi, 153****Character devices, 257–258, 273–274****charsets man page, 113****Check in in revision control, 191****Check out in revision control, 191–192****chrt command, 236–237****CISC (Complex Instruction Set Computer) processors, 501–502****Clean blocks, 280****clean target, 77****Client/server example**

- local sockets, 387–392
- network sockets, 392–393

client_un.c listing, 390–392**clock function, 249–251****clock_getres function, 253****clock_gettime function, 251–252****CLOCK_IDS, 252–253****clock_t type, 249–251****Clock ticks, 249, 253–255****CLOCKS_PER_SEC macro, 249****clone system call, 320****Closing**

- device drivers, 257
- POSIX message queues, 398

cmdline file, in /proc, 344, 593**cnx command, Vim, 166****Code. *See* Source code****Coherency, memory, 295–296****Coloring, stack, 339**

- Command mode**, vi, 147–148
 - Communication, interprocess.** *See* Interprocess communication (IPC)
 - Compatibility**
 - ABI, 97–98
 - text editor, 146
 - Compiler**
 - code analysis, 214–216
 - errors, 86–88
 - warnings, 88–89
 - C*, 93–94
 - C++*, 95–98
 - format*, 91–93, 525–526
 - implicit declarations*, 89–91
 - Complete Fair Queuing I/O scheduler (CFQ)**, 285
 - Complex Instruction Set Computer (CISC) processors**, 501–502
 - Concurrent Version System (CVS)**, 190–193, 201
 - Conditional breakpoints**, 532, 534–535
 - Conditional constructs**, 64–66
 - configure.in file**, 77
 - configure script**
 - build process, 74–77
 - errors in, 82–83
 - Conflicts in revision control**, 201
 - CONFORMING TO man section**, 112
 - Cons project**, 44
 - Containers**, 549
 - Contention, bus**, 459–461
 - continue command, gdb**, 532, 558
 - control.tar.gz file**, 32
 - Cooperative multitasking**, 227
 - Copy on write technique**, 319–320
 - Core dumps**, 589–591
 - core-file command, gdb**, 554
 - Core files, debugging**, 553–557
 - coreutils info page**, 119
 - Corruption**
 - memory, 578–579
 - stack, 555–556
 - Counters, semaphore**, 402–403
 - cpio files**, 5, 8, 11
 - cpp info page**, 119
 - CPPFLAGS variable**, 82–83
 - cppsym.c listing**, 536
 - cprev command, Vim**, 166
 - CPU**
 - cache, 291–295
 - multicore, 505–506
 - multiprocessing, 456–458
 - parallelism, 501–504
 - utilization
 - affinity masks for*, 507–509
 - vs. efficiency*, 458–459
 - vmstat for*, 469
 - CPU_ macros**, 508
 - cpu_idle function**, 458
 - cpu_set_t type**, 508–509
 - crazy-malloc.c listing**, 303–304
 - crazy-malloc2.c listing**, 308–310
 - crazy-malloc3.c listing**, 312–315
 - Cross-references**
 - help, 121–122
 - man pages, 111
 - cscope tool**, 211
 - ctags command**, 164–166, 210
 - Ctrl key**
 - vi, 151, 154
 - Vim, 161, 165
 - Cursor movement commands**
 - Emacs, 173
 - vi, 149–151
 - Cut commands**
 - Emacs, 174
 - vi, 155–158
 - CVS (Concurrent Version System)**, 190–193, 201
 - cwd file, in /proc**, 344, 593
- D**
- d command**
 - Ex mode, 159, 161
 - vi, 155
 - d modifier, print and x, gdb**, 543

- d option, GCC, 214–215**
- data.tar.gz file, 32**
- datadir option, 75**
- dbgprintf function, 586**
- dd command, 246–247, 475–476**
- deadline scheduler, 284–285**
- Debian distribution**
 - contents, 28–33
 - package format for, 14
 - updating, 34–35
- Debugging, 513–514**
 - backtraces for, 587–589
 - black box for, 584–587
 - core dumps, 589–591
 - GNU. *See* gdb debugger
 - IPC. *See* Interprocess communication (IPC)
 - memory issues. *See* Memory
 - preprocessor macros, 214–216
 - printf for. *See* printf function
 - procs for, 416, 591–594
 - shared objects. *See* Shared objects
 - shell commands, 85–86
 - signals for, 590–591
- Declaration warnings, 89–91**
- Decrypted passwords in memory, 343**
- Default editors, 143–144**
- Default shared object locations, 564–565**
- #define directive, 520–521**
- delete command**
 - Ex mode, 160
 - gdb, 532
- Delete commands**
 - Emacs, 174
 - vi, 155–158
- delete operator, C++, 572**
- Dependencies**
 - compiler, 214
 - make program, 45–48, 79
 - package, 13, 27–28
 - in updates, 38
- Deprecated features, 96–97**
- Desktop help tools, 120**
- Destructors, 95**
- detach command, gdb, 553**
- /dev directory, 9, 241, 260–263**
- devfs, 266**
- Device Independent (DVI) system, 124**
- Devices and device drivers, 257, 272–273**
 - block, 257–258, 274–276
 - buffer cache, 276–280
 - bus contention, 459–461
 - character, 257–258, 273–274
 - file-system, 259
 - file-system cache, 276–281
 - interrupts, 461–462
 - kernel modules, 259–260
 - lspci for, 464–467
 - network, 259
 - nodes, 257, 260–262
 - major numbers, 264–265
 - minor numbers, 263–264
 - sources, 266–271
 - PIC vs. APIC, 462–463
 - slots, 463–467
- diff command, 193–197**
- diff3 command, 198–201**
- diffstat command, 123**
- Digital signatures**
 - authentication with, 21–25
 - ELF files, 324
 - renaming functions based on, 99
- dir option, 197**
- Direct Memory Access (DMA), 295**
- Dirty blocks, 280**
- Dirty caches, 296**
- display command, gdb, 551–552**
- Displaying**
 - expressions, 551–552
 - variable values, 542–547
- distclean target, 77**
- Distribution formats, 3**
- DMA (Direct Memory Access), 295**
- do/while loops, 522–523**
- doc-linux package, 121**
- DocBook format, 126–127, 138**
- docbook-utils package, 126**

Document Type Definition (DTD), 126
Documentation. *See also* **Help**
 builds, 135–136
 code, 212–213
 cross referencing and indexing, 121–122
 Emacs commands, 172
 formats, 124–131
Dollar signs (\$), Ex mode, 159
Domains, socket, 383
down command, gdb, 542
Doxyfile file, 212–213
Doxygen tool, 212–213
dpkg command, 32
Drivers. *See* **Devices and device drivers**
DTD (Document Type Definition), 126
Dual-core CPUs, 505–506
Dumping
 core files, 553–557
 file data, 420–426
DVI (Device Independent) format, 124
dvipdf tool, 125
Dynamic libraries, 69–73, 561
Dynamic linker, 69, 300, 562–565
Dynamic priority, 230–231

E

e command, Ex mode, 161
-E option, GCC, 214–215
echo command, 50
ECOFF format, 324
edit command, Ex mode, 160
Editing source files. *See* **Text editors**
EF_ALLOW_MALLOC_0 variable, 582
ef, Electric Fence tool, 581
EF_PROTECT_BELOW variable, 583
Effective scheduler priority, 231
Electric Fence tool, 581–583
Elevator algorithms, 282–284
ELF format, 324
ellemtel autoindentation style, 171
else clause, 65

.emacs file, 177–178
Emacs text editor, 143, 146, 170
 browsing and building code, 175–176
 commands and shortcuts, 172–173
 cursor movement, 173
 deleting, cutting, and pasting, 174
 features, 170
 GUI mode, 179
 modes, 170–172
 search and replace, 174–175
 settings, 177–178
 text mode menus, 176–177
 for vi users, 178–179
environ file, in /proc, 344, 592
ENVIRONMENT man section, 111–112
error function, 63
errorformat setting, 168
Errors and warnings, build process, 78
 build stage, 83–86
 compiler errors, 86–88
 compiler warnings, 88–98
 configure stage, 82–83
 linker, 98–100
 Makefile, 79–82
etags utility, 175
etime format, 349
evince tool, 124, 130
Ex mode, 158–163
Exact matching for man pages, 110
Exception conditions, 370
exe file, in /proc, 345, 593
exec function
 child processes, 318, 320, 330, 332, 334
 chrt, 238
 control transfer, 320–321
Executables
 object files, 324
 profiling
gcov for, 491–494
gprof for, 488–491
OProfile for, 494–501
 scripts, 321–323
Execution parallelism, 502

execve function, 320–326
expandtabs setting, 167
.exrc file, 166–167
Extracting archive files, 11

F

f command, vi, 156
f modifier, print and x, gdb, 543
factorial.c listing, 555–556
FAQs, 133
fcntl function, 335
FD_CLOEXEC flag, 335
fd directory, 336
 description, 345
 for open files, 594
 for pipes, 437
fdatasync function, 281, 518–519
Fedora distribution
 GUI-enabled vim package for, 169
 man pages in, 107
 package format for, 14
fflush function, 517, 519
fg command, 244
Fibonacci function, 205
Field groups with top, 450
FIFO (first in, first out)
 pipes, 437–438
 scheduling, 236–237
file command, 6–7, 418–419, 554
File descriptors, 331–336
 leaked, 415
 limits, 337–338
 lsof command, 336–337
file-ipc-better.c listing, 360–362
file-ipc-naive.c listing, 358–360
File streams, 516–517
File systems
 buffering, 517–519
 caches, 276–280
 drivers, 259
 I/O, 274–276

filedat.c listing, 420–422

Files

dumping, 420–426
locking, 362
permissions, 8–10

Filesystem Hierarchy Standard, 121

for man pages, 105
overview, 113

filter function, 62

filter-out function, 62

Filters, Synaptic, 37

Finding

info pages, 118
man pages
 with apropos, 107–110
 with whatis, 110–111
open source tools, 2–3
processes, 352–353

findstring function, 62

firefox package, 23

First in, first out (FIFO)

pipes, 437–438
scheduling, 236–237

Fixed scheduler priority, 230

flawfinder tool, 571

Floating-point error (SIGFPE) signal, 553

Floating-point types with printf, 526

flock function, 413

Flushing caches, 296

Folding in Kate text editor, 183

Footprint, process, 329–331

fopen function, 362

Forcing core dumps, 589–591

Foreground processes, 244

fork-file.c listing, 332–336

fork function, 238, 242, 318, 320, 332, 334

Forks, timing in, 255

Formats

help documentation, 124–131
man pages, 105
package, 14
printf, 91–92, 525–526
process information, 349–351

- fprintf function**, 91
- Fragmented memory**, 305–306
- frame command**, gdb, 542
- free command**, 449, 452–453, 572
- Frequency, tick**, 253–255
- Frontside bus (FSB)**, 456–458
- fsync function**, 281, 517–519
- ftok function**, 413
- ftpd function**, 320
- ftruncate function**, 366–367
- full device**, 262
- funalloc.c listing**, 579–580
- Functions**
 - calling from gdb, 547
 - declarations, 90
 - hook, 178
 - renaming, 99
 - signal safe, 370
 - user-defined, 63–64
 - variable manipulation, 60
 - wrapper, 523–525
- fuser command**, 417–418

- G**
- G command**, vi, 149
- g suffix, x**, gdb, 545
- GCC**. *See* GNU Compiler Collection (GCC)
- gcc info page**, 119
- gcc project**, 76
- gcore function**, 590
- gcov tool**, 491–494
- gdb debugger**, 529–530
 - attaching to running process with, 553
 - for core files, 553–557
 - with Electric Fence, 582
 - inspecting and manipulating data with gdb, 541–542
 - C++ STL*, 549–551
 - display command*, 551–552
 - function calling*, 547
 - print*, 542–547
 - templates*, 548–549
 - for multithreaded programs, 557–558
 - for optimized code, 558–561
 - running code with, 530–531
 - stopping and restarting execution with, 531–532
 - breakpoints*, 532–537
 - watchpoints*, 537–541
- Gedit text editor**, 184–185
- GENERATE_ tags**, Doxygen, 213
- Gentoo distribution**, 14
- getprotoent function**, 384
- getrlimit function**, 314–315, 340–342
- getrusage function**, 250–251, 255, 328, 342
- Ghostsript tools**, 129–130
- glibc**, 569–570
 - MALLOC_CHECK_ for, 573
 - memusage, 574–576
 - mtrace, 573–574
- glibc-utils package**, 574–576
- gmon.out file**, 498–499
- gnome-help tool**, 107, 117, 120
- gnome-terminal**, 176
- gnometerminal window**, 171
- GNU arch tool**, 190
- gnu autoindentation style**, 171
- GNU Compiler Collection (GCC)**
 - build tools, 43, 74–77
 - C compiler warnings, 93–94
 - C++ compiler warnings, 95–98
 - code analysis, 214
 - debugger. *See* gdb debugger
 - Ghostsript tools, 129–130
 - heap, 305–307
 - info pages, 115–119
- GNU Privacy Guard (GPG)**, 22–25
- gprof tool**, 488–491, 498–499
- Graphviz tool**, 212
- groff language tool**, 131
- gs command**, 130
- gstack script**, 587

GUI text editors, 143, 182–183

- Emacs, 179
- Gedit, 184–185
- Kate and Kwrite, 183–184
- NEdit, 184–186
- SciTE, 186–187
- vi, 147
- Vim, 147, 168–169

gvim program, 168**gvimdiff tool, 198–199****gzip tool, 4, 7****H****h command, vi, 149****h suffix, x, gdb, 545****Hard limits, 340****Hard real-time application, 236****Hashes, 19–20****HAVE_DOT tag, Doxygen, 213****Heap memory**

- analysis, 579–581
- libraries for, 305–307
- overflows, 571–572, 581–583
- and virtual memory, 308–310

helgrind function, 580**hello-sync.c listing, 405****hello-unsync.c listing, 403–404****Help, 103**

- cross referencing and indexing, 121–122
- documentation formats, 124–131
- Internet information sources, 131–134
- kernel information, 134–138
- miscellaneous documentation, 138
- online tools, 103–104
 - desktop, 120*
 - info pages, 115–119*
 - man pages. See man pages*
- package queries, 122–123
- /usr/share/doc directory, 121

help command, Ex mode, 161**hexdump command, 424–425****hier man page, 113****Hints, cache, 294–295****hog.c listing, 448****Hook functions, 178****Hotplug feature, 266–268****HOWTO documents, 133****HTML for documentation, 127–129****htons macro, 392****Hyperthreading feature, 339, 505****Hypertransport connections, 460****HZ macro, 253****I****i command, vi, 151–152****i modifier, print and x, gdb, 543****IA32 architecture, 92****ifdef conditional, 64, 66****ifeq conditional, 64–66****ifndef conditional, 64****ifneq conditional, 64****Illegal instruction (SIGILL) signal, 435, 554****imake program, 42****Implicit declarations, 89–91****Implicit Makefile rules, 52–57****in_addr structure, 393****INADDR_ANY macro, 392****Include files, 87****Incompatible ABI, 97–98****Incremental search mode, 175****indent command, 204–206****Indentation**

- code beautifiers, 204–206
- Emacs, 171

Indexes

- Emacs, 175
- help, 121–122
- Vim, 164–166

info command, gdb, 532, 536, 540, 542, 548, 557

- info pages, 104, 115**
 - recommended, 119
 - searching, 118
 - viewing, 115–118
- init function, 242, 317–318, 320**
- Initializers, reordered, C++, 95–96**
- Inline assembly, 292**
- Inline functions, 523, 559**
- Inline synchronization, 521**
- Inodes, 440–442**
- insecure.c listing, 342–343**
- Insert commands, vi, 151–152**
- Insert mode**
 - vi, 147
 - Vim, 161–162
- Inspecting and manipulating data, gdb debugger, 541–542**
 - C++ standard template library, 549–551
 - display command, 551–552
 - function calling, 547
 - print, 542–547
 - templates, 548–549
- install target, 77–78**
- Installation scripts, 27–28**
- Instantiation, 548**
- Instructions**
 - efficiency, 480–484
 - latency, 503–504
- Interactive processes, 230**
- Internationalization, text editor, 188–189**
- Internet information sources, 131–134**
- Interprocess communication (IPC), 357**
 - debugging, 415
 - dumping file data, 420–426*
 - inodes, 440–442*
 - open files, 415–420*
 - pipes, 437–438*
 - POSIX, 431–434*
 - signals, 434–437*
 - sockets, 438–440*
 - System V, 426–431*
 - message queues, 393
 - POSIX, 397–402*
 - System V, 393–397*
 - pipes, 381–382
 - plain files, 358–363
 - semaphores, 402–407
 - POSIX, 407–410*
 - System V, 410–412*
 - shared memory, 363
 - POSIX, 364–367*
 - System V, 367–370*
 - signals, 370–371
 - handling, 371–373*
 - masks for, 373–376*
 - real-time, 376–378*
 - sending, 371*
 - sigqueue and sigaction, 378–381*
 - sockets, 382
 - client/server, 387–393*
 - creating, 383–385*
 - debugging, 438–440*
 - example, 385–387*
- Interruptible process state, 239**
- Interrupts**
 - device, 461–462
 - sar for, 473
- Interrupts per second information, 469**
- Invalidating cache entries, 295**
- I/O**
 - buffering, 517–519
 - devices, 272–273
 - block devices and file systems, 274–276*
 - buffer cache and file-system cache, 276–280*
 - character, 273–274*
 - information
 - sar for, 472–473*
 - vmstat for, 468–469*
 - scheduler, 282–286
- I/O APIC, 462**
- iostat tool, 473–475**
- IPC. *See* Interprocess communication (IPC)**

IPC_PRIVATE macro, 393
 ipcrm command, 426, 430
 ipcs command, 370, 426, 428–430
 IPPROTO_ flags for socket protocols, 385
 Iterators, C++, 550

J

j command, vi, 149
 jadetex package, 126
 java autoindentation style, 171
 JBellz Trojan, 17–18
 Jed text editor, 181–182
 Jiffies, 253
 Jobs, process, 346
 Joe text editor, 181–182

K

k command, vi, 149
 Kate text editor, 183–184
 Kconfig files, 136
 kdvi tool, 124
 Kernel, 221–222

- clock ticks, 253–255
- devices and device drivers. *See* Devices and device drivers
- information about, 134–138
- I/O scheduler, 282–286
- memory management. *See* Memory; Virtual memory
- process scheduler. *See* Process scheduler
- user mode vs. kernel mode, 222–226

 Kernel modules, 259–260
 Kernel space, 222–223, 226
 Kernel threads, 281
 Key mapping, vi, 148
 khelpcenter program, 107, 117–118, 120
 kill function, 354, 371, 378–380, 531, 591
 kmem device, 262

kmsg device, 262
 Knoppix distribution

- GUI-enabled vim package for, 169
- man pages in, 107
- package format for, 14

 .ko extension, 260
 konqueror tool, 130
 K&R autoindentation style, 171
 kswapd thread, 449
 kvim program, 168–169
 Kwrite text editor, 183

L

l command, vi, 149
 L1 cache, 291, 481–482
 L2 cache, 291–292, 482–483
 L3 cache, 292
 Labels in revision control, 192
 LAPIC (Local APIC), 462
 Last in, first out (LIFO) buffers, 338
 Latency

- instruction, 503–504
- scheduling, 235

 LaTeX system, 124
 Lazy TLB flushing, 290
 LBA (logical block addressing), 258
 ld command, 67
 LD_LIBRARY_PATH variable, 564
 ld info page, 119
 LD_PRELOAD variable, 564–565, 581–582
 ldconfig command, 564, 568
 ldd command, 68–69, 562–563, 566
 LDFLAGS variable, 82–83
 ldlinux.so linker, 564
 Leaks

- file descriptors, 415
- memory, 570
 - mtrace for, 573–574
 - valgrind for, 577–578

 Leaky.c listing, 577–578

Leaning toothpick syndrome, 163
Least recently used (LRU) pages, 448
less program, 116
libc info page, 119
 —**libdir option**, 75
Libraries
 for builds, 69–73
 for heap, 305–307
 naming convention, 73
 STL, 549–551
librt library, 433
Lifetime, variable, 52
LIFO (last in, first out) buffers, 338
Line buffering, 515–516
#line directives, 215
Linker errors, 98–100
Linking builds, 67–69
lint program, 88–89
Linus Elevator algorithm, 282–284
linux autoindentation style, 171
Linux Documentation Project, 133
listen function, 388, 391
Listen sockets, 388
Listing shared objects, 566–567
Little Endian byte order, 424, 426
Local APIC (LAPIC), 462
Local sockets
 client/server example, 387–392
 debugging, 439
Locating shared objects, 564–565
Locked memory, 310–315, 339–340
lockf function, 360, 362
Logical block addressing (LBA), 258
LRU (least recently used) pages, 448
ls command, 418
lsuf command, 336–337
 in debugging, 416–417
 inodes, 441
 pipes, 437
 semaphores, 434
 shared memory, 428
 sockets, 438–439
lspci tool, 464–467
ltrace command, 434, 484–485

M

m command, vi, 154
Macros
 debugging, 214–216
 do/while in, 522–523
 POSIX scheduling policy, 238
 printf, 519–521
madvise function, 295–296
magic field, 325
Magic numbers, 324–326
mailing-list archive (MARC), 134
Mailing lists, 134
Major device numbers, 264–265
Major page faults, 289, 447
make program and Makefiles, 42, 44–45.
 See also Builds
 conditional constructs, 64–66
 dependencies, 214
 error detection by, 84–86
 install stage, 77–78
 minimal, 52–57
 mistakes, 79–82, 84
 rules and dependencies, 45–48
 sources, 66–67
 targets, 77–78
 user-defined functions, 63–64
 variables
 automatic, 57–58
 defining, 48–49
 in implicit rules, 53–55
 manipulating, 59–63
 referencing and modifying, 52
 white space and newlines in, 50–51
 Vim, 166
make target, 77
makeinfo program, 125–126
makeprg setting, 168
makewhatis tool, 122
MALLOC_CHECK_ variable, 573
malloc function, 303–304
 overcommitment by, 309–310
 and overflow, 572
 page table mappings with, 446–447
 with pointers, 569

- MALLOC_TRACE variable**, 574
- malloc function**, 307
- Malware**, 17–18
- man pages**, 103–105
 - contents, 111–112
 - organization, 105–107
 - recommended, 112–114
 - searching
 - with apropos*, 107–110
 - with whatis*, 110–111
- Mandatory locks**, 362
- mandb version**, 122
- mandir option**, 76
- Mandriva distribution**, 14
- Mangling, name**, 99
- MAP_ANONYMOUS flag**, 308
- map command**, Ex mode, 160
- Maps, memory**, 592
 - anonymous, 73, 302, 308
 - virtual memory, 329–330
- maps file**, in */proc*, 345
- MARC (mailing-list archive)**, 134
- Masks**
 - affinity, 507–509
 - magic numbers, 325
 - signals, 435
- massif tool**, 579–581
- md5sum command**, 20–21, 33
- md5sums file**, 33
- mebibytes**, 114
- mem driver**, 258, 261–262, 264
- mem file**, in */proc*, 345
- memcheck function**, 577
- memcpy function**, 112
- Memory**
 - debugging, 569–570
 - buffer overflows*, 571–572
 - corruption*, 578–579
 - heap analysis*, 579–581
 - heap overflow*, 581–583
 - leaks*, 570, 573–574, 577–578
 - MALLOC_CHECK_*, 573
 - mtrace*, 573–574
 - pointers*, 569–570
 - statistics*, 574–576
 - Valgrind tools*, 576–581
 - decrypted passwords in, 343
 - kernel modes, 222
 - for libraries, 71
 - locked, 310–315, 339–340
 - mapping, 73, 302, 308, 592
 - for processes, 303–310
 - resident and locked*, 339–340
 - usage*, 353–354
 - system, 310
 - in system performance, 446
 - page faults*, 446–448
 - swapping*, 448–456
 - text editors, 187–188
 - virtual. *See* Virtual memory
 - vmstat information, 468
- Memory coherency**, 295–296
- Memory leaks**, 570
 - mtrace for*, 573–574
 - valgrind for*, 577–578
- Memory Management Unit (MMU)**, 288–289, 581
- Memory-related ps formats**, 350
- memset function**, 533
- memusage function**, 574–576
- memusagestat function**, 576
- mepis distribution**, 14
- Merging revision changes**, 192–197
- Message queues, IPC**, 393
 - POSIX, 397–402, 432–433
 - System V, 393–397, 429–430
- Meta keys**, 171
- Metadata**, 5
- Minibuffer mode**, 173
- Minimal Makefiles**, 52–57
- Minor device numbers**, 263–264
- Minor page faults**, 289, 447
- Minus signs (-)**, vi, 149
- Missing braces**, 93–94
- Missing include files**, 87
- Missing parentheses**, 93

Missing public keys, 24–25
Missing tabs, 79–80
mkfifo function, 382
mkfs command, 274
mknod command, 260–261, 382
mkswap command, 298
mlock function, 310–311
mlockall function, 311
mmap function
 devices, 274–275
 memory coherency, 295–296
 memory fragmentation, 306–308
 pointers, 570
 shared memory, 364
MMU (Memory Management Unit), 288–289, 581
Modeless text editors, 170
Modes
 Emacs, 170–172
 vi, 147–148
Modifying
 process affinity, 508
 variables, 52
modinfo command, 137–138
Modules
 documentation, 137–138
 kernel, 259–260
monotone tool, 190
mostlyclean target, 77
Motherboards, multiprocessor, 504–505
mount command, 276
mounts file, in `/proc`, 345
mpg123 program, 17–18
mpstat tool, 473–475
mq_close function, 398
mq_getattr function, 398
mq_notify function, 433
mq_open function, 397–398
mq_receive function, 399, 402
mq_send function, 399
mq_setattr function, 398–399
mq_unlink function, 398–399
mqueue file system, 433

mqueuepseudo file system, 432
MS_INVALIDATE flag, 296
MS_SYNC flag, 296
msg_receive function, 402
msgctl function, 394
msgget function, 393–394
msgrcv function, 396–397
msgsnd function, 394
msync function, 295–296, 364
Multicore CPUs, 505–506
Multiprocessor performance, 501
 Frontside Bus, 456–458
 SMP hardware, 501
 CPU parallelism, 501–504
 motherboards, 504–505
 multicore CPUs, 505–506
 SMT, 505
 SMP programming, 506
 affinity masks, 507–509
 scheduler, 506–507
Multitasking
 cooperative, 227
 preemptive, 228
 process. *See* Process scheduler
Multithreaded programs, 557–558
munlock function, 310
munmap function, 364
muntrace function, 574

N

n command, vi, 150
-N flag, diff, 195
Names
 finding processes by, 352–353
 in libraries, 73
 in man pages, 106
 mangling, 99
 package, 28
 sending process signals by, 354–355
Nano text editor, 181–182

nasty.c listing, 533
Native POSIX Threads Library (NPTL), 509
ncurses library, 450
NEdit text editor, 184–186
netstat command, 438–440
Network devices, 259
Network sockets
 client/server example, 392–393
 debugging, 441
Network Time Protocol (NTP) package, 127
new-corrupt.cpp listing, 578–579, 581
new operator, 572
Newlines
 printf, 515–516
 in variables, 50–51
Newsgroups, 134
next command, gdb, 531, 558
nice command, 234–235, 238
nm command
 for linking, 72
 for symbols, 99, 567–568
Nodes, device, 257, 260–262
 major numbers, 264–265
 minor numbers, 263–264
 sources, 266–271
Nonvirtual destructors, 95
noop scheduler, 282–284
NPTL (Native POSIX Threads Library), 509
NTP (Network Time Protocol) package, 127
null device, 262
Null pointer arguments, 93
Number of buffers per session feature, 147
NVRAM driver, 264
nvrाम module, 267–268

O

o command, vi, 151
o modifier, print and x, gdb, 543
objdump command, 559–560, 567–568
Object files, executable, 324

Octal dump command, 425–426
od command, 425–426
off_t type, 525
Official package names, 28
offset field for magic numbers, 325
Offsets, memory pages, 288
Online help tools, 103–104
 desktop, 120
 info pages, 115–119
 man pages. *See* man pages
OOM (out-of-memory) killer, 297, 309
oom_adj file, in /proc, 345
oom_score file, in /proc, 345
opannotate tool, 496, 499
opcontrol tool, 495–497
Open files
 information for, 593
 IPC debugging, 415–420
open function
 device drivers, 257
 permissions for, 332
Open source tools, 1
 archive files, 4–5
 extracting, 11
 identifying, 6–7
 querying, 7–11
 defined, 2
 distribution formats, 3
 finding, 2–3
 packages. *See* Packages
 significance, 2–3
Opening POSIX message queues, 397–398
OpenSSH source code, 18
operator man page, 113
opgprof tool, 496, 498–499
opreport tool, 496, 498
.oprofile file, 496
OProfile profiler, 494–495
 for applications, 497–500
 opcontrol tool, 495–497
 remarks, 500–501
oprofiled tool, 496–498
Opteron CPUs, 504–505

Optimized code, debugging, 558–561
 Out-of-memory (OOM) killer, 297, 309
 Out-of-order steps in optimized code, 559
 Output modifiers, print and x, gdb, 543–544
 Overcommit, memory, 309–310
 Overflow
 buffer, 555–556, 571–572
 heap, 581–583
 stack, 338, 555–556
 Overlays, shared objects for, 562
`overrun.c` listing, 540–541

P

p command, vi, 155

-p option, patch, 197

Packages

contents, 27–33
 package managers, 12–14
 queries, 29–30, 122–123
 security for, 17–26
 source and binary, 14–16
 updating, 33–39
 working with, 16–17

PAE (Physical Address Extension), 302

Page faults, 289, 446–448

Page tables, 289

Pages in virtual memory. *See* Virtual memory

Parallel bus, 463, 465

Parallelism, CPU, 501–504

Parentheses ()

macros, 520
 missing, 93

Partitions, swap, 297–298

Passwords in memory, 343

Pasting

Emacs, 174
 vi, 155–158

patch command, 193–197

Patching, libraries for, 70

patsubst function, 60, 62

Pattern rules, 55

pause function, 242

pax utility, 5

PCI bus, 459, 463–465

PCI Express (PCIe) connections, 460–461,
 463

PCI-X bus, 463

PDF (Portable Document Format), 130–131

PDF_HYPERLINKS tag, Doxygen, 213

pdflatex tool, 125

pdflush daemon, 281

Percent signs (%)

Ex mode, 159
 vi, 149

Performance, 445

application. *See* Application performance
 multiprocessor. *See* Multiprocessor performance
 printf, 514–515
 system. *See* System performance

Periods (.)

Ex mode, 159
 vi, 154

Permissions, 8–10, 332

`permute.cpp` listing, 551–552

PF_ flags for socket domains, 383

pgrep command, 352–353

Physical Address Extension (PAE), 302

PIC (Programmable Interrupt Controller),
 462–463

Pico text editor, 181–182

PID directory, 344–345

PIDs

with gdb, 553
 System V memory, 428
 System V message queues, 430

pinfo command, 117

pipe function, 381–382

Pipelines, 502–505

Pipes, IPC, 381–382, 437–438

pkg-config tool, 567

pkgtool package manager, 13–14

pskill command, 354–355

Plain files for IPC, 358–363

Plus signs (+), vi, 149

pmap function, 299–301, 305, 353–354, 592

Pointers

- freeing, 569–570
- null arguments, 93

Polling in IPC, 376**pop command, vi, 165****Popping, stack, 338****port device, 262****Portable Document Format (PDF), 130–131****portage package manager, 13–14****Position-independent code, 563****POSIX API**

- IPC debugging, 431–434
- library, 69
- message queues, 397–402, 432–433
- semaphores, 407–410, 433–434
- shared memory, 364–367, 431–432

posix-msgq-ex.c listing, 399–402**posix_sem.c listing, 407–410, 433–434****posix-shm.c listing, 365–367****Postmortem debugging, 554****PostScript language, 129–130****Pound signs (#)**

- Ex mode, 163
- macros, 520–521

Preemptable kernels, 229**Preempt in process scheduler, 228–229****Preemptive multitasking, 228****Prefetching, cache, 294****—prefix option, 75****Preprocessor macros**

- debugging, 214–216
- printf, 519–521

print command, gdb, 541–547**print_trace function, 588–589****printf function, 514**

- buffering
 - and file streams, 516–517*
 - and file systems, 517–519*
- format warnings, 91–92, 525–526
- gdb, 541
- performance effects, 514–515
- preprocessor help, 519–521
- remarks, 528–529
- synchronization issues, 515–516

tips, 527–528

wrapper functions, 523–525

Priorities

process scheduler

- and fairness, 229–233*
- and nice, 234–235*
- real-time, 235–237*

queued messages, 397

/proc directory, 268–269, 344–346**/proc/cpuinfo file, 477–478****/proc/pid/fd directory, 437****Process scheduler, 226**

blocking, preemption, and yielding in,
228–229

overview, 226–227

priorities

- and fairness, 229–233*
- and nice, 234–235*
- real-time, 235–237*

process states, 239–245

real-time processes, 238–239

timing, 246–249

applications, 255–256

kernel clock tick, 253–255

system time units, 249–253

Processes, 317–318

clone, 320

communication between. *See* Interprocess
communication (IPC)

copy on write, 319–320

environment, 592–593

exec, 320–321

executable object files, 324

executable scripts, 321–323

file descriptors, 331–338

finding, 352–353

footprints, 329–331

fork and vfork, 318

information for

displaying, 346–348

formats for, 349–351

sar for, 471–472

vmstat for, 468

limits, 340–343

Processes, *continued*

- memory for, 303–310
 - resident and locked*, 339–340
 - usage*, 353–354
 - virtual*, 298–302
- miscellaneous binaries, 324–326
- and procs, 343–346
- real-time, 238–239
- scheduling. *See* Process scheduler
- signals for, 354–355
- stack for, 338–339
- states, 239
 - sleeping vs. running*, 240–241
 - stopped*, 243–245
 - zombies and Wait*, 241–243
- synchronizing, 327–329

Processor architecture

- instruction efficiency, 480–484
- x86info for, 476–480

procs file system

- for debugging, 416, 442, 591–594
- process information provided by, 343–346
- vs. sysfs, 268–269

procps package, 346, 349–351**Profiling executables**

- gcov for, 491–494
- gprof for, 488–491
- OProfile for, 494–501

profme.c listing, 488–491**—program-prefix option, 76****—program-suffix option, 76****Programmable Interrupt Controller (PIC), 462–463****Projects in revision control, 191****Protected output, 403****Protocols, socket, 384–385****ps function, 235, 243**

- for finding processes, 352–353
- formats, 349–351
- options, 233
- overview, 346–348
- for semaphores, 431
- for states, 239–240

Pseudotargets, 47–48

- pthread_create function, 320, 509
- pthread_self function, 509
- pthread_setaffinity_np function, 509
- pthread_setschedparam function, 238
- Public keys, missing, 24–25
- Pushing, stack, 338
- Put commands, vi, 155–158
- python autoindentation style, 171

Q**q command, Ex mode, 161****Quad-core CPUs, 506****Quantum, 228, 237****Queries**

- archive file, 7–11
- package, 29–30, 122–123

queryformat option, 30**Question marks (?)**

- Ex mode, 159
- vi, 150

Queues, message, 393

- POSIX, 397–402, 432–433
- System V, 393–397, 429–430

quit command, vi, 160**R****-R flag, patch, 195****Race conditions, 334, 357****raise function, 380, 591****ramdisk device**

- memory for, 310
- vs. tmpfs, 280

random device, 262**rbreak command, gdb, 549****RCS (Revision Control System), 190–191****Read operations**

- blocking, 248–249
- character devices, 273
- POSIX message queues, 399

Readability

- automatic variables for, 57–58
- indentation, 204–206

readelf command, 567–569**README files, 121, 123, 135****Real-time applications, 235–236****Real-time priorities, 235–236**

- FIFO scheduling, 236–237
- round-robin scheduling, 237

Real-time processes, 238–239**Real-time signals, 376–378****Reaping processes, 327****Red Hat distribution**

- package format, 14
- updating, 37–39

Reduced Instruction Set Computer (RISC)

- processors, 501–502

Referencing variables, 52**regex man page, 114****register pseudofile, 325****Registers**

- SMT, 505
- vi, 157

Regular expressions

- man pages, 110
- text editors, 145

Removing POSIX message queues, 398–399**Renaming functions, 99****renice command, 234****Reordered initializers, 95–96****Repeated vi commands, 148****Resident memory, 339–340****Restarting execution with gdb, 531–532**

- breakpoints, 532–537
- watchpoints, 537–541

Reviewing revision changes, 193–197**Revision control, 189**

- basics, 189–190
- diff and patch, 193–197
- RCS, 190–191
- reviewing and merging changes, 193–197
- terms, 191–193

Revision Control System (RCS), 190–191**RISC (Reduced Instruction Set Computer)**

- processors, 501–502

RLIMIT_ flags, 341–342**rlimit structure, 314, 340–341****rmmod command, 260, 267****root file, in /proc, 345****Round-robin scheduling, 237****RPM (RPM Package Manager) package, 13–14**

- contents, 28–31
- updating, 35–36

rpm utility, 23**rpm2cpio command, 31****rt-sig.c listing, 377–378****Rules, Makefiles, 45–48, 52–57****run command, gdb, 531****runex script, 232****Running out of process memory, 303–310****Running processes, 239**

- attaching to, 553
- vs. sleeping, 240–241

rwatc command, gdb, 540**S****S command, vi, 153****s modifier, print and x, gdb, 543****SA_SIGINFO flag, 379****sar tool, 469–470**

- interrupt activity, 473
- I/O information, 472–473
- process information, 471–472
- remarks, 473–474
- virtual memory information, 470–471

—sbindir, configure, 75**scanf function, 92****SCHED_FIFO macro, 238–239****sched_getaffinity function, 508****SCHED_OTHER macro, 238****SCHED_RR macro, 238****sched_setaffinity function, 508****sched_setscheduler function, 238–239****sched_yield function, 228**

Scheduler

- I/O, 282–286
- process. *See* Process scheduler
- SMP programming, 506–507

Scheduler-lock feature, 558**Scheduler-related process formats, 351****schedutils package, 507****SciTE text editor, 186–187****Scons project, 44****Scope, make variable, 52****Scripts**

- executable, 321–323
- installation, 27–28
- typos in, 323

SCSI devices, 263, 269–271**Search and replace feature**

- Emacs, 174–175
- Ex mode, 163

Searching

- info pages, 118
- man pages
 - with apropos*, 107–110
 - with whatis*, 110–111

Security

- kernel modes for, 222
- packages, 17–26
- shared objects, 565

SEE ALSO man section, 111**Segmentation violation signal. *See* SIGSEGV
(segmentation violation) signal****select function, 388****self directory, in /proc, 336****sem_destroy function, 410****sem_init function, 410****sem_open function, 408–410****sem_post function, 407****sem_t type, 433****sem_unlink function, 409****sem_wait function, 406****Semaphores, IPC, 402–407**

- POSIX, 407–410, 433–434
- System V, 410–412, 430–431

semop function, 412**semtimedop function, 431****Sending signals to processes, 371****server_un.c listing, 388–390****set command**

- Ex mode, 160
- gdb, 547

set args command, gdb, 530–531**set scheduler-lock command, gdb, 558****setaffinity functions, 509****setgid bit, 8–10****setrlimit function, 312, 314–315, 339–342****Settings**

- Emacs, 177–178
- vi, 166–168

setuid bit, 8–10**setvbuf function, 516–517, 519****SGML Document Type Definition, 126****Shared memory, IPC, 363**

- POSIX, 364–367, 431–432
- System V, 367–370, 426–429

Shared objects, 561

- creating, 563
- listing, 566–567
- locating, 564–565
- security with, 565
- symbols in, 567–569
- unused, 567
- using, 562

Shell commands

- debugging. *See* Debugging
- Makefiles, 79–80
- syntax errors, 85–86

shell function, make, 60**shiftwidth setting, 167****shm_open function, 364, 366–367, 369****shm_unlink function, 364****shmat function, 368****shmctl function, 368****shmdt function, 368****shmget function, 368–369****Shortcuts, Emacs, 172–173**

- SI_ flags used with struct siginfo, 381**
- Side effects, printf, 528–529**
- side-effects.c listing, 528–529**
- SIGABRT signal, 554, 589**
- sigaction function, 372–373, 378–381**
- sigaddset function, 373**
- SIGBUS signal, 554**
- SIGCHLD signal, 243, 327, 375–376, 378**
- SIGCONT signal, 244–245, 591**
- sigdelset function, 373**
- SIGFPE signal, 553**
- sighandler_t type, 372**
- SIGILL signal, 435, 554**
- siginfo structure, 380**
- SIGINT signal, 436**
- sigismember function, 373**
- SIGKILL signal, 244, 372**
- signal system call, 379, 532**
- Signal safe functions, 370**
- Signals**
 - for debugging information, 590–591
 - IPC, 370–371
 - debugging, 434–437*
 - handling, 371–373*
 - masks for, 373–376*
 - real-time, 376–378*
 - sending, 371*
 - sigqueue and sigaction, 378–381*
 - for processes, 354–355
- Signatures**
 - authentication with, 21–25
 - ELF files, 324
 - renaming functions based on, 99
- sigpending function, 376**
- sigprocmask.c listing, 374–376**
- sigprocmask function, 373**
- sigqueue function, 378–381**
- SIGQUIT signal, 240**
- SIGRTMAX macro, 376**
- SIGRTMIN macro, 376, 378**
- SIGSEGV (segmentation violation) signal**
 - core files, 553
 - with Electric Fence, 583
 - heap overflow, 572
 - invalid addresses, 289
 - pointers, 569
 - stack overflow, 338, 342, 571
- sigset_t type, 373**
- SIGSTOP signal, 372, 591**
- sigsuspend function, 376**
- SIGTERM signal, 240, 244**
- SIGTSTP signal, 244**
- SIGTTIN signal, 244–245**
- SIGTTOU signal, 245**
- Single-stepping, 558**
- 64-bit types with printf, 525–526**
- Size**
 - container, 550
 - executables, 72
 - memory pages, 288
- size method, 550**
- skill command, 354–355**
- Slackware distribution, 14**
- Slashes (/)**
 - Ex mode, 159
 - vi, 150
- sleep command, 246**
- Sleeping process state, 240–241**
- Slots, device, 463–467**
- smaps file, in /proc, 345**
- SMP. See Symmetric Multiprocessing System (SMP) computers**
- SMT (Symmetric Multithreading), 505**
- snprintf function, 91**
- .so extension, 73**
- SOCK_ socket types, 384**
- sockaddr structure, 393**
- sockaddr_un structure, 391**
- socket system call, 383, 387**
- socketpair.c listing, 385–387**
- socketpair system call, 385–387**

Sockets

- domains, 383
- IPC, 382
 - client/server*, 387–393
 - creating*, 383–385
 - debugging*, 438–440
 - example*, 385–387
- protocols, 384–385
- types, 383

Soft limits, 340**Soft real-time applications, 236****son-of-hog.c listing, 451–456****sort function, make, 62****Source code, 141**

- analysis
 - cflow*, 207–210
 - compiler for*, 214–216
 - ctags*, 210
- beautifiers and browsers, 203–204
 - astyle*, 206–207
 - cflow*, 207–210
 - compiler*, 214–216
 - cscope*, 211
 - ctags*, 210
 - Doxygen*, 212–213
 - indent code*, 204–206
- code coverage, 491–494
- editing. *See* Text editors
- packages, 14–16
- revision control, 189
 - basics*, 189–190
 - diff and patch*, 193–197
 - RCS*, 190–191
 - reviewing and merging changes*, 193–197
 - terms*, 191–193
- VPATH for, 66–67

SourceForge.net site, 132–133**Spaces in scripts, 323****Split windows in SciTE, 187****sprintf function, 91****Square brackets ([]), vi, 149****ssh-keygen tool, 115****sshd daemon, 320, 331–332****Stack coloring, 339****Stacks**

- corruption, 555–556
- overflow, 338, 342, 571
- for processes, 338–339
- tag, 164–166

Stale caches, 295**Stalls, pipeline, 503–504****Standard Template Library (STL), 549–551****start command, gdb, 531****Starved requests, 284****stat command, 419–420, 442****stat file, in /proc, 345****States, process, 239**

- sleeping vs. running, 240–241
- stopped, 243–245
- zombies and Wait, 241–243

Static libraries, 69**Static linking, 71–73****Static priority, 231****Static storage, 338****statm file, in /proc, 345****status file, in /proc, 345****stdarg.h file, 524****stdout stream, 516****step command, gdb, 531, 558****sticky bit, 8, 11****STL (Standard Template Library), 549–551****Stopped process state, 239, 243–245****Stopping execution with gdb, 531–532**

- breakpoints, 532–537
- watchpoints, 537–541

strace command

- for program performance, 485–487
- purpose, 225
- for signals, 436–437
- timing with, 246–248, 360

strace package, 132**strftime function, 92****String types**

- for file data dumps, 422–423
- with printf, 526

- strings command, 422–423
 - strip function, make, 62
 - Stroustrup autoindentation style, 171
 - strptime function, 92
 - stty command, 245
 - subst function, make, 62
 - substitute command, vi, 163
 - Substitution references, 59
 - Subversion tool, 190, 201
 - Suffix rules for object code, 54–55
 - suffixes man page, 114
 - summer-proj.c listing, 492
 - summer-proj.c.gcov listing, 493–494
 - Superscalar units, 502
 - swapoff command, 297, 452
 - swapon command, 297, 452
 - Swapped memory. *See* Virtual memory
 - swapvals function, C++ template example, 548
 - Switched fabric architectures, 460
 - Symbols in shared objects, 567–569
 - Symmetric Multiprocessing System (SMP)
 - computers, 295, 457
 - hardware, 501
 - CPU parallelism*, 501–504
 - motherboards*, 504–505
 - multicore CPUs*, 505–506
 - programming, 506
 - affinity masks*, 507–509
 - scheduler*, 506–507
 - Symmetric Multithreading (SMT), 505
 - Synaptic tool, 36–37
 - sync system call, 106, 281
 - Synchronization
 - caches, 296
 - inline, 521
 - printf issues, 515–516
 - processes, 327–329
 - Syntax highlighting, 144, 147
 - syntax setting, 168
 - /sys directory, in /proc, 268–269
 - syscall function, 224–225
 - sysconf function, 249, 254, 337
 - sysfs feature, 268–272
 - sysstat package
 - iostat and mpstat, 474–475
 - sar, 469–474
 - sysstat service, 472–474
 - System calls, 223–226
 - System memory, 310
 - System performance, 445
 - CPU utilization, 456–459
 - devices and interrupts, 459–467
 - memory issues, 446
 - page faults*, 446–448
 - swapping*, 448–456
 - sysstat package
 - iostat and mpstat*, 474–475
 - sar*, 469–474
 - virtual-memory status, 467–469
 - System software installation, 18
 - System time units, 249–253
 - System V API
 - IPC debugging, 426–431
 - message queues, 393–397, 402, 429–430
 - semaphores, 410–412, 430–431
 - shared memory, 367–370, 426–429
 - sysv-msgq-example.c listing, 394–396
 - sysv_sem.c listing, 410–412
 - sysv-shm.c listing, 368–370
- ## T
- t command, vi, 156
 - t modifier, print and x, gdb, 543
 - Tab completion, gdb, 537
 - Tab expansion, vi, 147
 - Tabs, Makefiles, 79–80
 - tabstop setting, 167
 - tag command, vi, 165
 - Tags, Vim, 164–166
 - tags command, vi, 165
 - Tape archive (tar) files, 4–6, 8, 12
 - Targets, make, 77
 - taskset command, 507
 - tbreak command, gdb, 531

Templates, C++, 548–549
Temporary variables, 560
Terminal-based text editors, 143
tetex-bin package, 125
tex command, 126
TeX system, 124
texi2dvi tool, 126
Texinfo language, 115, 125–126
Text editors, 141

- categories, 142–143
- clones, 179–182
- default, 143–144
- Emacs. *See* Emacs text editor
- features, 144–146
- GUI, 182–187
- memory usage, 187–188
- summary, 188–189
- vi. *See* vi text editor; Vim text editor

Text functions, 62–63
Text mode menus, 176–177
Text substitution patterns, 59
32-bit types with printf, 525–526
Thrashing, 297, 310–311, 455
Thread Affinity API, 509
Threads

- affinity masks, 509
- gdb for, 557–558
- hyperthreading technology, 339
- kernel, 281
- timing, 255

3G/1G kernel split, 222, 329
Throughput, pipeline, 503
Ticks, 249, 253–255
time command, 246, 256, 447, 475–476
Time formats, 349–350
times function, 250–251
timespec structure, 226, 251–252
timeval structure, 250–251
Timing in process scheduler, 246–249

- applications, 255–256
- kernel clock tick, 253–255
- system time units, 249–253

TLB (Translation Lookaside Buffer), 289–291
TLDLP FAQs, 133

tmpfs file system

- mounting, 367
- vs. ramdisk, 280
- semaphores, 433
- shared objects, 431

tms structure, 250
tnext command, vi, 165–166
top command, 450–456
tostop setting, 245
tprevious command, vi, 165
trace-buffer.c listing, 584–587
Translation Lookaside Buffer (TLB), 289–291
troff language, 104, 124, 131
Trojans, 17–18
tselect command, vi, 165
Typos in scripts, 323

U

u command, vi, 152, 154
u modifier, print and x, gdb, 543
Ubuntu distribution

- GUI-enabled vim package for, 169
- man pages in, 107
- package format for, 14

udev package, 128, 266–268
ulimit function, 315, 339, 589
UML_LOOK tag, 213
Underruns, 583
Undo feature, 147
undocumented man page, 123
Unified L2 cache, 482
Uninitialized variables, 94
Uninstall features, 13
uninstall target, 77–78
Uninterruptible process state, 239–241
units man page, 114
Unnamed semaphores, 409–410
Unused shared objects, 567
Unused variables, 559
up command, gdb, 542
up2date tool, 34, 37–39

Updating packages, 33–39
Upgrading packages, 13
urandom device, 262
uri man page, 114
url man page, 114
urn man page, 114
USE_PDFLATEX tag, Doxygen, 213
Usenet, 134
User-defined functions, 63–64
USER_HZ macro, 254
User mode vs. kernel mode, 222–226
User space
 kernel modes, 222–223, 226
 memory management. *See* Memory; Virtual memory

V

Valgrind tools, 576–577
 heap analysis, 579–581
 instruction efficiency, 480–484
 memory corruption, 578–579
 memory leaks, 577–578
 remarks, 580

Variables

 displaying values of, 542–547
 make program
 automatic, 57–58
 defining, 48–49
 in implicit rules, 53–55
 manipulating, 59–63
 referencing and modifying, 52
 white space and newlines in, 50–51
 in optimized code, 559–560
 uninitialized, 94

Vectors, 549

vfork system call, 318, 320
vfprintf function, 524
vi text editor, 143, 146. *See also* Vim text editor
 change commands, 153
 command mode, 147–148
 cursor movement commands, 149–151
 cut, paste, and delete commands, 155–158

 Ex mode, 158–161
 features, 147
 for info pages, 116–117
 insert commands, vi, 151–152
 miscellaneous commands, vi, 154
 modes, vi, 147–148
 settings, 166–168

vi users, Emacs for, 178–179

vim-gnome package, 168–169

vim-gtk package, 169

Vim text editor, 143, 146, 181–182

 browsing and building code, 164–166
 features, 147
 GUI mode, 168–169
 insert mode, 161–162

vim-X11 package, 168–169

.vimrc file, 166–167

Viper mode, Emacs, 178

Virtual memory, 286–287

 caches, 291–296
 coherency, 295–296
 and heap, 308–310
 map, 329–330
 MMU for, 288–289
 for processes, 298–302
 sar for, 470–471
 status, 467–469
 swap space, 296–298
 swapping, 448–456
 TLB, 289–291

Viruses, 17–18

vmstat command, 278–279, 467–469

volatile keyword, 561

VPATH variable, 66–67, 80–82

vprintf function, 524

vsprintf function, 524

W

w command

 Ex mode, 161
 vi, 150

- w suffix, x, gdb, 545
 - Wait for zero operation, 412
 - wait system call, 241–243, 327–329
 - Wait states, socket, 438
 - wait3 system call, 327–328
 - wait4 system call, 327–328
 - waitpid system call, 327–328, 334
 - Wall option, 89–92
 - Warnings, compiler, 88–89
 - C, 93–94
 - C++, 95–98
 - format, 91–93, 525–526
 - implicit declarations, 89–91
 - watch command, gdb, 532, 538
 - watch watchpoints, gdb, 540
 - watchpoint command, gdb, 537
 - Watchpoints, gdb, 537–541
 - wchan file, in /proc, 345
 - Wformat-nonliteral option, gcc, 92
 - Wformat-y2k option, gcc, 92
 - whatis command, 110–111, 541, 550
 - White space, 50–51
 - whitesmith autoindentation style, 171
 - Windows text editor compatibility, 146
 - WNOHANG flag, 328–329
 - word function, make, 63
 - wordlist function, make, 63
 - words function, make, 63
 - WordStar text editor, 180
 - Wrapper functions, 523–525
 - wrapscan setting, 167
 - Write back, cache, 293–294
 - Write combining, cache, 294
 - write command, vi, 160
 - write function, 366
 - Write operations
 - blocking, 248–249
 - character devices, 273
 - CPU, 293–294
 - Ex, 160
 - POSIX message queues, 399
 - shared memory, 366
 - Write through, cache, 294
 - WUNTRACED flag, 328
 - www.gnu.org site, 131–132
- ## X
- x command
 - Ex mode, 161
 - gdb, 541–547
 - x modifier, print and x, gdb, 543
 - x86info tool, 476–480
 - Xandros distribution, 14
 - xdvi tool, 124
 - xinetd daemon, 320, 441
 - xit command, Ex mode, 160
 - xman tool, 107
 - xpdf tool, 130
 - xxd command, 423–424
 - xxdiff tool, 198, 201–202
- ## Y
- y command, vi, 155
 - Yank commands, 155–156
 - Yellowdog Updater Modified (yum), 34–36
 - yelp tool, 120
 - Yielding in process scheduler, 228–229
- ## Z
- zb command, vi, 154
 - zero device, 262
 - Zile text editor, 181–182
 - .zip files, 6
 - Zombie process state, 239, 241–243
 - zt command, vi, 154
 - zz command, vi, 154