

491

"After 20 years, this is still the best exposition of the workings of a 'real' operating system."

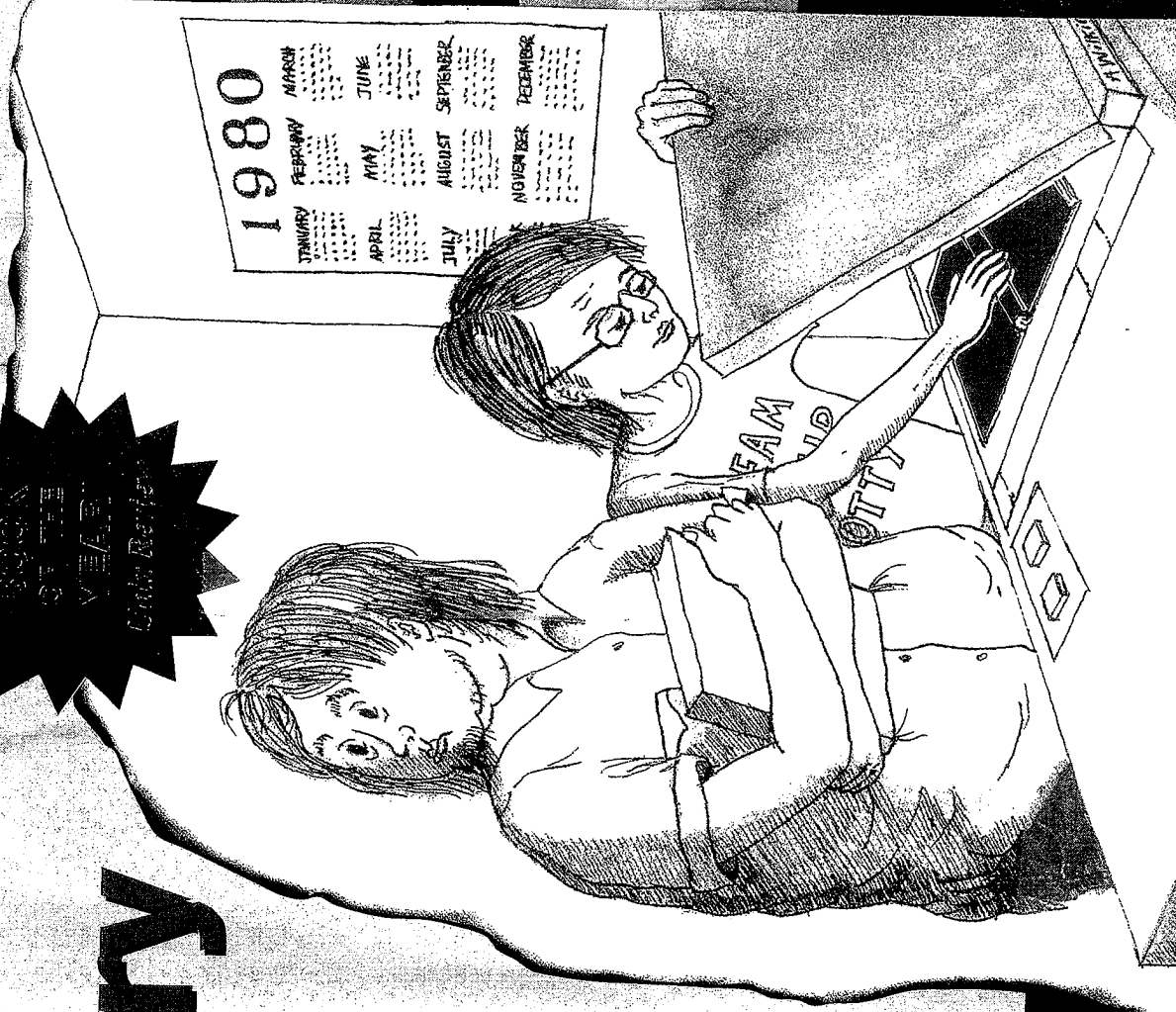
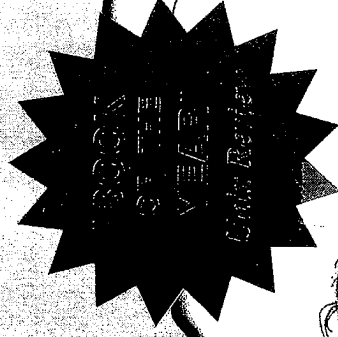
Ken Thompson

Lions' Commentary on UNIX[®] 6th Edition

with Source Code

John Lions

Foreword by Dennis Ritchie



**Lions' Commentary on
UNIX 6th Edition
with Source Code**



COMPUTER
CLASSICS
REVISITED

Peter H. Salus, *Series Editor*

Other Books in the Series:

Packet Communication

*Before the Internet Volume 1:
Planning the ARPANET*

*Before the Internet Volume 2:
Building the ARPANET*

Lions' Commentary on UNIX 6th Edition with Source Code

Originally circulated as two restricted-release volumes:

**UNIX OPERATING SYSTEM
SOURCE CODE LEVEL SIX**

and

**A COMMENTARY ON THE
UNIX OPERATING SYSTEM**

John Lions

With Forewords by Dennis M. Ritchie and Ken Thompson;
Prefatory Notes by Peter H. Salus and Michael Tilson;
a Historical Note by Peter H. Salus; and Appreciations by Greg Rose,
Mike O'Dell, Berny Goodheart, Peter Collinson, and Peter Reintjes



**PEER-TO-PEER
COMMUNICATIONS**

Lions' Commentary on UNIX 6th Edition with Source Code

Selection and compilation of materials in this volume Copyright © 1996 by Peer-to-Peer Communications, Inc. All rights reserved.

A Commentary on the UNIX Operating System, Copyright © 1977 by John Lions. All rights reserved.

Portions reprinted by permission of The Santa Cruz Operations, Inc. (SCO). To the extent SCO has an intellectual property interest in the material contained herein, SCO has granted a license to publish solely for the purpose of creating an educational work. SCO grants no license for any other use of this material. None of the computer software source code contained in this publication may be copied, distributed, modified, transferred, or used in any way. It has been printed here solely for educational purposes in the study of an historical landmark in computer history.

Computer Classics Revisited Series

Series Editor: Peter H. Salus

Published by Peer-to-Peer Communications, Inc.

P.O. Box 640218

San Jose, California 95164-0218, U.S.A.

Phone: 408-435-2677

Fax: 408-435-0895

E-mail: info@peer-to-peer.com

Worldwide Web address: <http://www.peer-to-peer.com>

Cover Design: Gordon Haber

Cover Illustration: Anne Wilkinson

Production: Bookwrights of Charlottesville, VA

Printing: Gilliland Printing, Inc.

Printed in the United States of America

3 4 5 6 7 8 9 10

ISBN 1-57398-013-7

Peer-to-Peer Communications offers discounts on this book when ordered in bulk quantities. For more information, or to receive a current catalog, please contact the publisher at the address above.

We are grateful to Addison-Wesley Publishing Company for permission to quote from Peter H. Salus' *A Quarter Century of UNIX* (1994).

SCO® and The Santa Cruz Operation® are registered trademarks of the Santa Cruz Operation, Inc. in the USA and other countries. UNIX® is a registered trademark in the USA and other countries, licensed exclusively through X/Open Company Limited.

DEDICATION

To Ken Thompson and Dennis Ritchie who created the UNIX software system and made the writing of this source code and commentary possible, to my former colleagues and students at the University of New South Wales for whom this book was originally prepared, and to my wife Marianne and daughters Katherine and Elizabeth who never tried to understand UNIX but always supported my work.

Lions Book /n./ The two parts of this book contained (1) the entire source listing of the UNIX Version 6 kernel, and (2) a commentary on the source discussing the algorithms. These were circulated internally at the University of New South Wales beginning 1976-77, and were, for years after, the *only* detailed kernel documentation available to anyone outside Bell Labs. Because Western Electric wished to maintain trade secret status on the kernel, the Lions book was never formally published and was only supposed to be distributed to affiliates of source licensees...In spite of this, it soon spread by *samizdat* to a good many of the early UNIX hackers.

— Eric S. Raymond, *New Hacker's Dictionary 2nd ed.*

COI

Ackr

Pref:

Fore

A Shr

I

UNIX

A CoI

Appre

A

C

B

P

P

About

CONTENTS

Acknowledgments	viii
Prefatory Notes	
Michael Tilson	ix
Peter H. Salus	ix
Forewords	
Dennis M. Ritchie	x
Ken Thompson	x
A Short Historical Note	
Peter H. Salus	xi

UNIX Operating System Source Code Level Six

A Commentary on the UNIX Operating System

Appreciations

Mike O'Dell
Greg Rose
Berny Goodheart
Peter Collinson
Peter Reintjes

About the Author

ACKNOWLEDGMENTS

My sincere thanks and appreciation go to everyone who helped get this book published even though it took 20 years to get there.

I would like to thank Dan Doernberg and the staff at Peer-to-Peer Communications, Inc. who have supported this publication. To SCO Inc. who followed the philosophy of UNIX and gave permission to publish these texts and arranged for the red tape to be overcome.

I am extremely grateful to Ken Thompson, Peter Collinson, Mike O'Dell, Peter Reintjes, Greg Rose and Michael Tilson who have contributed with their appreciations.

Finally, I am most indebted to Peter Salus and Dennis Ritchie who have both worked so hard producing this book and have battled through the legal barriers, and to Berny Goodheart whose unflinching persistence and invaluable friendship have helped me get this book out.

After producing the initial version of these booklets I was privileged to spend my sabbaticals in 1978, 1983 and 1989 at Bell Laboratories and work as part of the team. The support and friendship given by the many colleagues that I met there will never be forgotten. During that time, strong ties were formed between Bell Laboratories and my Australian colleagues, which resulted in several UNIX pioneers visiting Australia to work and speak at conferences. The Australian UNIX User Group appreciated this and greatly benefited from it.

John Lions

PREFACE

When this much pleasure. piece of technical brilliant technology was itself an study. John at The source code other, and I he A generation key learning to mate UNIX do and business-limited, because source code. available to the still learn from

Michael Tilson,
Mike Tilson is Principal
Officer, The Santa

PREFATORY NOTES

When this book was first published, I was astonished by how much pleasure I got from reading what should have been a dry piece of technical documentation. John Lions had created a truly brilliant technical work. The UNIX operating system kernel code was itself an elegant work, and even today it remains worthy of study. John added a line-by-line analysis that was equally elegant. The source code and the annotations were perfectly suited to each other, and I haven't seen anything to equal this achievement since. A generation of operating system developers used this work as a key learning tool, and I wonder just how much it influenced the ultimate UNIX domination of technical workstations, internet servers, and business-critical computing. At the time, the circulation was limited, because it was an annotation of licensed proprietary source code. I'm very pleased that this work can now be made available to the public. It is part of our technical history, and we can still learn from it today.

Michael Tilson

Mike Tilson is President, UniForum Association and Chief Information Officer, The Santa Cruz Operation, Inc. (SCO).

In early 1993, I began preparing for the 25th birthday of the UNIX Operating System, created in the summer of 1969 by Dennis M. Ritchie, Ken Thompson, Doug McIlroy, and Rudd Canaday. One of the things I hoped to do by summer 1994 was to see legitimate publication of the material in this volume. John Lions was coming to the USENIX Association's celebration in Boston and would receive a Lifetime Achievement Award. Seeing this book would be a suitable tribute.

But that was not to be.

Thanks to the efforts of Dennis Ritchie, AT&T's lawyers stated that they had "no objection" to publication. Negotiations with Novell, purchasers of the UNIX system from AT&T, were sluggish. Then, late in 1995, came the announcement that The Santa Cruz Operation, Inc. ("SCO") had purchased UNIX from Novell. Dennis and I wrote to Mike Tilson and Doug Michels, executives at SCO we knew personally. Mike actually owned a copy of John Lions' work, treasured it, and within a short period of time had arranged with SCO's lawyers for permission.

John Lions' Code and Commentary volumes were the UNIX system's *samizdat* of the '70s and '80s. The code is now out of date. As most of the commentators note, the comments are not. To learn about operating systems one must read and understand code. John's work enabled many of us to do that.

In 1969 AT&T was an assemblage of telecommunications monopolies, including Bell Telephone Laboratories (a.k.a. BTL, Bell, Bell Labs, or The Labs) and the Western Electric Company. The former brought us a variety of research developments, including UNIX. The latter manufactured and marketed products. We have left the citations in the Code and Commentary as they appeared in 1977. The Labs are now part of Lucent; Western Electric is now AT&T Technologies.

I am proud to have been instrumental in finally bringing this to print. I can retire my many-generation photocopy now.

Peter H. Salus

Peter H. Salus, former Executive Director of USENIX and the Sun User Group, is Series Editor of "Computer Classics Revisited."

A HISTORICAL NOTE

In 1977, John Lions completed his *Commentary on the UNIX Operating System*, companion volume to his reproduction of the V6 source code. These two slim books may be the most important computer items never to be published. I asked him about the *Source Code* and the *Commentary*. He said:

I was teaching Operating Systems; I was competing with a colleague who was teaching Compilers by getting students to write real compilers; so a code **reading** exercise seemed a good idea. Also **our** Unix license wasn't explicit enough to forbid the activity. Why Unix? There wasn't much choice. It was highly competent as you know, and it was much better than the competition (we had also acquired Brinch Hansen's SOLO system).

The March 1977 *UNIX NEWS* (vol. 2, no. 3) announced the availability of the book, to licensees, together with a note by Ferentz: "Ken Thompson has seen the first version of the book and

reports that it is a good job." The price, including airmail, was \$A17.70 (under \$20 US, at that time). The UKUUG newsletter announced the availability of the code and commentary, but the next issue stated that future orders should be placed with Bell Laboratories and by 1978 the volumes were no longer available. They must be the most frequently photocopied books in the entire area of computer science. They carry the appropriate copyright notices and the restriction to licensees, but once again, there was no way that Western Electric could stem the circulation of something of such value. I confess to possessing both a many-generation photocopy and a copy, in the bright orange and red covers, inscribed to me by John Lions. ...

From Peter H. Salus, *A Quarter Century of UNIX* (Reading, MA: Addison-Wesley Publishing, 1994), 127-130; reprinted with the permission of the publisher.

COPY NO. 050B NAME
PROPERTY OF BELL LABORATORIES, INC.
COPY TO BE RETURNED TO: COMPUTING INFORMATION
SERVICE MH 2F-128
UNIX OPERATING SYSTEM SOURCE CODE VERSION 6

UNIX OPERATING SYSTEME SOURCE CODE LEVEL SIX

This booklet has been produced for students at the University of New South Wales taking courses 6.602B and 6.657G.

It contains a specially edited selection of the UNIX Operating System source code, such as might be used on a typical PDP11/40 computer installation.

The UNIX Software System was written by K. Thompson and D. Ritchie of Bell Telephone Laboratories, Murray Hill, N.J. It has been made available to the University of New South Wales under a licence from the Western Electric Company.

THIS INFORMATION IS PROPRIETARY AND IS THE PROPERTY OF BELL TELEPHONE LABORATORIES, INC. ITS REPRODUCTION OR DISCLOSURE TO OTHERS, EITHER ORALLY OR IN WRITING, IS PROHIBITED WITHOUT WRITTEN PERMISSION OF BELL LABORATORIES.

J. Lions
Department of Computer Science
The University of New South Wales.
November 1977

Second Printing

5372 ARDY	0100	0489 EMFILE	24	5696 IREAD	0400	0160 PUSER	100	7975 TBDELAY	006000
7993 ASLEEP	0100	0496 EMLINK	31	5627 ISGID	02000	0158 PWAIT	40	2615 TBIT	020
7992 BUSY	040	0488 EMFILE	23	5694 ISGID	02000	2605 R0	(0)	7984 TIMEOUT	01
8617 BUSY	0400	0484 ENDEV	19	7987 ISOFEN	04	2606 R1	(-2)	7961 TTHIWAT	50
4584 B-ASYNC	0400	0468 ENDEXT	2	5626 ISUID	04000	2607 R2	(-9)	7951 TTIPRI	10
4576 B-BUSY	010	0474 ENDEXEC	8	5693 ISUID	04000	2608 R3	(-8)	7962 TLOWAT	30
4586 B-DELWRI	01000	0478 ENMEM	12	5628 ISVIX	01000	2609 R4	(-7)	7952 TTOPRI	20
4574 B-DONE	02	0493 ENDSFC	28	5695 ISVTX	01000	2610 R5	(-6)	7963 TTYHOG	256
4575 B-ERROR	04	0480 ENDTBLK	15	5684 ITEXT	040	2611 R6	(-3)	0311 UBMAR	0177660
4579 B-MAP	040	0485 ENTDIR	20	5680 IUPD	02	2612 R7	(1)	0308 UDSA	0177660
4577 B-PHYS	020	0490 ENDTTY	25	5683 IWANT	020	7971 RAW	040	0306 UISA	0177640
4573 B-READ	01	0472 ENXIO	6	5630 IWRITE	0200	5094 RCOM	04	0304 UISD	0177600
4583 B-RELOC	0200	8612 EDF	3	5697 IWRITE	0200	8014 RDRENB	01	3706 UMODE	0170000
4581 B-WANTED	0100	0467 EPERM	1	0165 KL	0177560	8614 RDRENB	01	2659 UMODE	0170000
4572 B-WRITE	0	0497 EPIPE	32	8008 KLADDR	04	5121 RHRCOM	0	2662 USER	020
0140 CANBSIZ	256	0495 EROFS	30	8009 KLBASE	0177560	5363 RKADDR	0177400	0103 USIZE	16
8840 CAP	01	8618 ERROR	0100000	7968 LCASE	04	5120 RHRCOM	070	7977 VDELAY	040000
7990 CARR_ON	020	0494 ESPIPE	29	8812 LPADDR	100	5120 RHRCOM	060	8610 WAITING	1
7955 CEDT	004	0469 ESRCH	3	8819 LPHWAT	100	5363 RKADDR	0177400	5093 WCOM	02
7954 CERASE	'#'	0491 ETXTBSY	26	8818 LPLWAT	50	0315 R0	02	5373 WLD	020000
7958 CINTR	0177	7973 EUNVP	0200	8817 LPPRI	10	0106 ROOTINO	1	0316 W0	04
7956 CKILL	'@'	0483 EXDEV	18	8821 MAXCOL	80	0317 RW	06	7985 WOPEN	02
1509 CLOCK1	0177546	3018 EXPRI	-1	0135 MAXMEM	(64*32)	3707 SCHMAG	10	7967 XTABS	02
1510 CLOCK2	0172540	8847 FORM	014	0130 NBUF	15	2660 SETD	4		
8609 CLOSED	0	5519 FPIPE	04	0143 NCALL	20	0385 SIDL	10		
0141 CMAPSIZ	100	5517 FREAD	01	0146 NCLIST	100	0120 SIGEMT	7		
7957 CQUIT	034	5518 FWRITE	02	8012 NDL11	0	0121 SIGFPT	8		
7976 CRDELAY	030000	5095 G0	01	0134 NEXEC	3	0120 SIGEMT	7		
7970 CRMOD	020	5368 G0	01	0132 NEXE	100	0114 SIGHUP	1		
5374 CTRLRDY	0200	7966 HUPCL	01	0131 NINODE	100	0115 SIGINT	2		
0107 DIRSIZ	14	-0147 HZ	60	8011 NKL11	1	0117 SIGINS	4		
8010 DLBASE	0175610	5681 IACC	04	7974 NLDELAY	001400	0115 SIGINT	2		
7980 DONE	0200	5620 IALLOC	0100000	0133 NMOUNT	5	0119 SIGIOT	6		
8616 DONE	0200	5687 IALLOC	0100000	0105 NODVE	(-1)	0122 SIGKIL	9		
8815 DONE	0200	5092 IENABLE	0100	0139 NOFILE	15	0126 SIGPIPE	13		
5369 DRESET	014	5370 IENABLE	0100	0144 NPROC	50	0116 SIGQUIT	3		
5371 DRY	0200	7981 IENABLE	0100	5364 NRK	4	0124 SIGSEG	11		
8013 DSRDY	02	8615 IENABLE	0100	5365 NRKBLK	4872	0125 SIGSYS	12		
0473 E2BIG	7	8814 IENABLE	0100	0113 NSIG	20	0118 SIGTRC	5		
0479 EACCESS	13	5631 IEXEC	0100	0145 NTEXT	40	0391 SLOAD	01		
0477 EAGAIN	11	5698 IEXEC	0100	0104 NULL	0	0393 SLOCK	04		
0475 EADDF	9	5624 IFLBK	060000	7972 ODDP	0100	0384 SRUN	100		
2658 EBIT	1	5691 IFLBK	060000	8843 OPEN	04	0142 SMAPSIZ	3		
0481 EBUSY	16	5623 IFCHR	020000	8607 PCADDR	0177550	0384 SRUN	3		
0476 ECHILD	10	5690 IFCHR	020000	8624 FCIHWAT	250	0137 SSIZE	20		
0318 ED	010	5622 IFDIR	040000	8620 FCIPRI	30	0382 SSLEEP	1		
0482 EEXIST	17	5621 IFMT	060000	8623 FCOHWAT	100	7988 SSTART	010		
0466 EFAULT	106	5688 IFMT	060000	8622 FCOLWAT	50	0387 SSTOP	6		
0470 EINTR	4	5625 ILARG	010000	8621 FCDPRI	40	0394 SSWAF	010		
0487 EINVAL	22	5692 ILARG	010000	0155 FINDD	-90	0392 S8YS	02		
0471 EIO	5	5679 ILOCK	01	7715 PIPSIZ	4096	0395 STRC	020		
0486 EISDIR	21	5682 IMOUNT	010	0157 PPIPE	1	0166 SW	0177570		
8842 EJECT	02	3914 IPCPRI	(-1)	0156 PRIBIO	-50	0383 SWAIT	2		
8820 EJLINE	60	5629 IREAD	0400	0164 PS	0177776	0396 SWTED	040		
				0159 PSLEP	90	2661 SYS	0104400		
				0154 PSWP	-100	0386 SZOMB	5		

```

2500 #
2501 /*
2502 */
2503
2504 /*
2505 * Structure of the coremap and swapmap
2506 * arrays. Consists of non-zero count
2507 * and base address of that many
2508 * contiguous units.
2509 * (The coremap unit is 64 bytes,
2510 * the swapmap unit is 512 bytes)
2511 * The addresses are increasing and
2512 * the list is terminated with the
2513 * first zero count.
2514 */
2515 struct map
2516 {
2517     char *m_size;
2518     char *m_addr;
2519 };
2520 /* ----- */
2521
2522 /*
2523 * Allocate size units from the given
2524 * map. Return the base of the allocated
2525 * space.
2526 * Algorithm is first fit.
2527 */
2528 malloc(mp, size)
2529 struct map *mp;
2530 {
2531     register int a;
2532     register struct map *b;
2533
2534     for (b = mp; b->m_size; b++) {
2535         if (b->m_size >= size) {
2536             a = b->m_addr;
2537             b->m_addr += size;
2538             if ((b->m_size -= size) == 0)
2539                 do {
2540                     b++;
2541                     (b-1)->m_addr = b->m_addr;
2542                     } while((b-1)->m_size == b->m_size);
2543             return(a);
2544         }
2545     }
2546     return(0);
2547 }
2548 /* ----- */
2549

```

```

2550 /*
2551 * Free the previously allocated space a
2552 * of size units into the specified map.
2553 * Sort a into map and combine on
2554 * one or both ends if possible.
2555 */
2556 mfree(mp, size, a)
2557 struct map *mp;
2558 {
2559     register struct map *b;
2560     register int t;
2561     register int a;
2562
2563     a = a;
2564     for (b = mp; b->m_addr <= a && b->m_size != 0; b++) {
2565         if (b->mp && (b-1)->m_addr + (b-1)->m_size == a) {
2566             (b-1)->m_size += size;
2567             if (a+size == b->m_addr) {
2568                 (b-1)->m_size += b->m_size;
2569                 while (b->m_size) {
2570                     b++;
2571                     (b-1)->m_addr = b->m_addr;
2572                     (b-1)->m_size = b->m_size;
2573                 }
2574             }
2575         } else {
2576             if (a+size == b->m_addr && b->m_size) {
2577                 b->m_size -= size;
2578                 b->m_size += size;
2579                 } else if (size) do {
2580                     t = b->m_addr;
2581                     b->m_addr = a;
2582                     a = t;
2583                     t = b->m_size;
2584                     b->m_size = size;
2585                     b++;
2586                 } while (size = t);
2587             }
2588         }
2589     }
2590     /* ----- */
2591

```