

## Macmillan Computer Science Series

*Consulting Editor:*

Professor F. H. Sumner, University of Manchester

- A. Abdellatif, J. Le Bihan, M. Limame, *Oracle – A User's Guide*  
S. T. Allworth and R.N. Zobel, *Introduction to Real-time Software Design, second edition*  
Ian O. Angell, *High-resolution Computer Graphics Using C*  
Ian O. Angell and Gareth Griffith, *High-resolution Computer Graphics Using FORTRAN 77*  
Ian O. Angell and Gareth Griffith, *High-resolution Computer Graphics Using Pascal*  
M. Azmoodeh, *Abstract Data Types and Algorithms, second edition*  
C. Bamford and P. Curran, *Data Structures, Files and Databases, second edition*  
Philip Barker, *Author Languages for CAL*  
P. Beynon-Davies, *Information Systems Development, second edition*  
G. M. Birtwistle, *Discrete Event Modelling on Simula*  
B. G. Blundell, C. N. Daskalakis, N. A. E. Heyes and T. P. Hopkins, *An Introductory Guide to Silvar Lisco and HILO Simulators*  
Richard Bornat, *Understanding and Writing Compilers*  
Linda E. M. Brackenbury, *Design of VLSI Systems – A Practical Introduction*  
Alan Bradley, *Peripherals for Computer Systems*  
G. R. Brookes and A. J. Stewart, *Introduction to occam 2 on the Transputer*  
P. C. Capon and P. J. Jinks, *Compiler Engineering Using Pascal*  
J. C. Cluley, *Introduction to Low Level Programming for Microprocessors*  
Eric Davalo and Patrick Naïm, *Neural Networks*  
S. M. Deen, *Principles and Practice of Database Systems*  
C. Delannoy, *Turbo Pascal Programming*  
Tim Denvir, *Introduction to Discrete Mathematics for Software Engineering*  
D. England et al., *A Sun User's Guide, second edition*  
J. S. Florentin, *Microprogrammed Systems Design*  
A. B. Fontaine and F. Barrand, *80286 and 80386 Microprocessors*  
Michel Gauthier, *Ada – A Professional Course*  
J. B. Gosling, *Design of Arithmetic Units for Digital Computers*  
M. G. Hartley, M. Healey and P. G. Depledge, *Mini and Microcomputer Systems*  
J. A. Hewitt and R. J. Frank, *Software Engineering in Modula-2 – An Object-oriented Approach*  
Roger Hutty, *COBOL 85 Programming*  
Roland N. Ibbett and Nigel P. Topham, *Architecture of High Performance Computers, Volume I*  
Roland N. Ibbett and Nigel P. Topham, *Architecture of High Performance Computers, Volume II*  
Patrick Jaulent, *The 68000 - Hardware and Software*  
P. Jaulent, L. Baticle and P. Pillot, *68020-30 Microprocessors and their Coprocessors*  
M. J. King and J. P. Pardoe, *Program Design Using JSP – A Practical Introduction, second edition*  
V. P. Lane, *Security of Computer Based Information Systems*  
Bernard Leguy, *Ada – A Programmer's Introduction*  
M. Léonard, *Database Design Theory*  
David Lightfoot, *Formal Specification Using Z*  
A. M. Lister and R. D. Eager, *Fundamentals of Operating Systems, fifth edition*  
Elizabeth Lynch, *Understanding SQL*  
Tom Manns and Michael Coleman, *Software Quality Assurance, second edition*  
B. A. E. Meekings, T. P. Kudrycki and M. D. Soren, *A Book on C, third edition*  
R. J. Mitchell, *Microcomputer Systems Using the STE Bus*  
R. J. Mitchell, *C++ Object Oriented Programming*  
R. J. Mitchell, *Modula-2 Applied*  
Y. Nishinuma and R. Espesser, *UNIX – First contact*  
Pham Thu Quang and C. Chartier-Kastler, *MERISE in Practice*  
A. J. Pilavakis, *UNIX Workshop*

*continued overleaf*

E. J. Redfern, *Introduction to Pascal for Computational Mathematics*  
Gordon Reece, *Microcomputer Modelling by Finite Differences*  
F. D. Rolland, *Programming with VDM*  
W. P. Salman, O. Tisserand and B. Toulout, *FORTH*  
L. E. Scales, *Introduction to Non-Linear Optimization*  
A. G. Sutcliffe, *Human-Computer Interface Design, second edition*  
C. J. Theaker and G. R. Brookes, *Concepts of Operating Systems*  
M. Thorin, *Real-time Transaction Processing*  
M. R. Tolhurst *et al.*, *Open Systems Interconnection*  
A. J. Tyrrell, *COBOL from Pascal*  
I. R. Wilson and A. M. Addyman, *A Practical Introduction to Pascal with BS6192, second edition*

*Other titles*

I. O. Angell, *Advanced Graphics on VGA and XGA Cards Using Borland C++*  
B. V. Cordingley and D. Chamund, *Advanced BASIC Scientific Subroutines*  
N. Frude, *A Guide to SPSS/PC+, second edition*  
Percy Mett, *Introduction to Computing*  
Tony Royce, *COBOL – An Introduction*  
Tony Royce, *Structured COBOL – An Introduction*  
Barry Thomas, *A PostScript Cookbook*

# Fundamentals of Operating Systems

**A. M. Lister**

*University of Queensland*

**R. D. Eager**

*University of Kent at Canterbury*

**Fifth Edition**



© A. M. Lister 1975, 1979, 1984

© A. M. Lister and R. D. Eager 1988, 1993

All rights reserved. No reproduction, copy or transmission of this publication may be made without written permission.

No paragraph of this publication may be reproduced, copied or transmitted save with written permission or in accordance with the provisions of the Copyright, Designs and Patents Act 1988, or under the terms of any licence permitting limited copying issued by the Copyright Licensing Agency, 90 Tottenham Court Road, London W1P 9HE.

Any person who does any unauthorised act in relation to this publication may be liable to criminal prosecution and civil claims for damages.

First edition 1975

Reprinted 1977, 1978

Second edition 1979

Reprinted 1980 (twice), 1981 (four times), 1983 (three times)

Third edition 1984

Reprinted 1984, 1985, 1986

Fourth edition 1988

Reprinted 1989 (twice), 1990, 1991, 1992

Fifth edition 1993

Published by

**THE MACMILLAN PRESS LTD**

Houndmills, Basingstoke, Hampshire RG21 2XS

and London

Companies and representatives

throughout the world

**ISBN 978-0-333-59848-1      ISBN 978-1-349-13283-6 (eBook)**

**DOI 10.1007/978-1-349-13283-6**

A catalogue record for this book is available  
from the British Library.

# Contents

<i>Preface to the First Edition</i>	<i>xi</i>
<i>Preface to the Second Edition</i>	<i>xii</i>
<i>Preface to the Third Edition</i>	<i>xiii</i>
<i>Preface to the Fourth Edition</i>	<i>xiii</i>
<i>Preface to the Fifth Edition</i>	<i>xiv</i>
<b>1 Introduction</b>	<b>1</b>
1.1 Types of operating system	3
1.2 System structure	6
1.3 The 'paper' operating system	8
<b>2 Functions and Characteristics of an Operating System</b>	<b>9</b>
2.1 Operating system functions	9
2.2 Operating system characteristics	13
2.3 Desirable features	15
<b>3 Concurrent processes</b>	<b>17</b>
3.1 Programs, processes and processors	17
3.2 Communication between processes	19
3.3 Semaphores	21
3.4 Monitors	29
3.5 Message passing	30
3.6 Summary	32
<b>4 The System Nucleus</b>	<b>33</b>
4.1 Essential hardware facilities	33
4.2 Outline of the nucleus	35
4.3 Representation of processes	37
4.4 The first-level interrupt handler	38
4.5 The dispatcher	41
4.6 Implementation of <i>wait</i> and <i>signal</i>	45
<b>5 Memory Management</b>	<b>51</b>
5.1 Objectives	51
5.2 Virtual memory	53
5.3 Implementation of virtual memory	54
5.4 Memory allocation policies	66
5.5 The working set model	72

5.6	Implementation in the paper system	74
<b>6</b>	<b>Input and Output</b>	<b>76</b>
6.1	Design objectives and implications	77
6.2	The I/O procedures	80
6.3	The device handlers	82
6.4	Buffering	85
6.5	File devices	87
6.6	Spooling	89
6.7	Conclusion	91
<b>7</b>	<b>The Filing System</b>	<b>92</b>
7.1	Objectives	92
7.2	File directories	93
7.3	Sharing and security	96
7.4	Secondary memory organisation	99
7.5	File system integrity	106
7.6	Opening and closing files	108
7.7	Conclusion	112
<b>8</b>	<b>Resource Allocation and Scheduling</b>	<b>114</b>
8.1	General observations	114
8.2	Allocation mechanisms	115
8.3	Deadlock	117
8.4	The scheduler	124
8.5	Scheduling algorithms	126
8.6	Process hierarchies	131
8.7	Control and accounting	134
8.8	Summary	138
<b>9</b>	<b>Protection</b>	<b>140</b>
9.1	Motivation	140
9.2	Development of protection mechanisms	141
9.3	A hierarchical protection mechanism	145
9.4	General systems	147
9.5	Conclusion	152
<b>10</b>	<b>Reliability</b>	<b>153</b>
10.1	Objectives and terminology	153
10.2	Fault avoidance	155
10.3	Error detection	158
10.4	Fault treatment	160
10.5	Error recovery	161
10.6	Multilevel error handling	163
10.7	Conclusion	165

<b>11</b>	<b>The User Interface</b>	<b>166</b>
	11.1 Some general remarks	166
	11.2 Command languages	167
	11.3 Job control languages	170
	11.4 Modern control languages	173
	11.5 Graphical user interfaces	173
	11.6 The job pool	174
	11.7 System messages	175
	11.8 Passage of a job through the system	175
	<b>Conclusion</b>	<b>177</b>
	<i>Appendix: Monitors</i>	<i>182</i>
	<i>References</i>	<i>185</i>
	<i>Index</i>	<i>191</i>

**to my parents**



## *Preface to the First Edition*

An operating system is probably the most important part of the body of software which goes with any modern computer system. Its importance is reflected in the large amount of manpower usually invested in its construction, and in the mystique by which it is often surrounded. To the non-expert, the design and construction of operating systems has often appeared an activity impenetrable to those who do not practise it. I hope this book will go some way toward dispelling the mystique, and encourage a greater general understanding of the principles on which operating systems are constructed.

The material in the book is based on a course of lectures I have given for the past few years to undergraduate students of computer science. The book is therefore a suitable introduction to operating systems for students who have a basic grounding in computer science, or for people who have worked with computers for some time. Ideally the reader should have a knowledge of programming and be familiar with general machine architecture, common data structures such as lists and trees, and the functions of system software such as compilers, loaders and editors. It will also be helpful if he or she has had some experience of using a large operating system, seeing it, as it were, from the outside.

The first two chapters of the book define the functions of an operating system and describe some common operating system characteristics. Chapter 3 establishes the process as a basic concept in the discussion of the concurrent activities which go on inside an operating system, and describes how processes communicate with each other. The rest of the book then describes the construction of an operating system from the bottom up, starting at the interface with the machine hardware and ending at the interface with the user. By the end of the book the system which has been constructed is seen to possess the features demanded at the beginning.

Throughout the book I have tried to show how each stage in the construction of an operating system naturally follows on the previous ones, and have emphasised the logical structure of the system as a whole. I have done this for two reasons. The first is pedagogical: my experience indicates that students gain a better understanding of complex material if it is presented in a coherent manner. The second is frankly polemic: this is the way I believe operating systems should be built. Attention to structure and logical dependence are the best means we have of building operating systems which are easy to understand, easy to maintain, and relatively error free. Finally, I would like to thank

the many friends and colleagues who have helped in the writing of this book. In particular I would like to thank David Lyons, who has been a fertile source of ideas and comment; and David Howarth, who made valuable comments on an earlier draft. Thanks are also due to Colin Strutt, Morris Sloman, John Forecast, Ron Bushell and Bill Hart, who have made constructive suggestions for improving the text.

ANDREW LISTER

## *Preface to the Second Edition*

Any book on computer science suffers from the fact that its subject matter is undergoing rapid change, both technological and conceptual. A book on operating systems is no exception: the hardware on which systems run is being revolutionised by the large scale integration of components, and ideas about what operating systems should do are being modified by changing user requirements and expectations. Although this book is about ‘fundamentals’, which can be expected to be less volatile than other topics, it would be foolish to ignore the developments which have occurred in the four years since publication.

Consequently I have amended the text in three main ways for this edition. First, there is an additional chapter on reliability, a topic whose importance has increased dramatically as reliance on computer systems has grown in many areas of life. Secondly, the references to real-life operating systems have been updated to include systems which have come into use since the first edition, and references to the literature have been similarly updated. Thirdly, I have altered the presentation of certain topics so as better to reflect current ideas, and have expanded the conclusion to indicate what developments can be expected in the future.

Other changes in the text have been motivated by feedback from readers of the first edition. Here I owe a great debt to my students, who have had the perception to point out errors, and the good sense to expose inadequate exposition. Any improvements in this respect are due largely to them, though any remaining inadequacies are, of course, my own.

A.L.

## ***Preface to the Third Edition***

Almost a decade after publication it is reassuring to know that the ‘fundamentals’ of the title remain essentially the same. However, as the systems of the early 1970s fade into distant memory I have found it necessary to update the text by reference to their successors. I have also taken the opportunity to introduce one or two new topics, and to improve the presentation of several others. In this latter respect I am indebted to Morris Sloman and Dirk Vermeir for their valuable suggestions, and, as ever, to my students.

A.L.

## ***Preface to the Fourth Edition***

I feel honoured to have been asked to update this book for its fourth edition. I have used all of the earlier editions as part of my own teaching of operating systems, and am glad to have had the opportunity to revise this excellent text to reflect recent developments. The main changes are in the chapter on filing systems, but there are many other areas where new systems and techniques have merited mention. Finally, I would like to thank the person who first taught me about operating systems many years ago when I was a student – Andrew Lister.

BOB EAGER

## ***Preface to the Fifth Edition***

It is now nearly twenty years since the first edition of this book was published. Much has changed, yet much has remained the same.

Because true fundamentals do not alter, there have been no radical revisions to the text. However, we have taken the opportunity to update a number of areas, with the inclusion of more examples drawn from modern systems. We have deleted mention of some systems which have long disappeared, but references to important early systems still remain. We have also updated and expanded the list of references, and hope that this will direct the reader towards further areas of interest.

BOB EAGER AND ANDREW LISTER