

Bill Fran

055704



Software Engineering in the UNIX[®]/C Environment

William B. Frakes
Software Productivity Consortium

Christopher J. Fox
AT&T Bell Laboratories

Brian A. Nejmeh
Instep Incorporated



Prentice-Hall International, Inc.

ISBN 0-13-826496-1

This edition may be sold only in those countries to which it is consigned by Prentice-Hall International. It is not to be re-exported and it is not for sale in the U.S.A., Mexico, or Canada.

©1991 by AT&T



Published by Prentice-Hall, Inc.
A Division of Simon & Schuster
Englewood Cliffs, New Jersey 07632

All rights reserved. No part of this book may be reproduced, in any form or by any means, without permission in writing from the publisher.

UNIX® is a registered trademark of UNIX System Laboratories, Inc.

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

ISBN 0-13-826496-1

Prentice-Hall International (UK) Limited, London
Prentice-Hall of Australia Pty. Limited, Sydney
Prentice-Hall Canada Inc., Toronto
Prentice-Hall Hispanoamericana, S.A., Mexico
Prentice-Hall of India Private Limited, New Delhi
Prentice-Hall of Japan, Inc., Tokyo
Simon & Schuster Asia Pty. Ltd., Singapore
Editora Prentice-Hall do Brasil, Ltda., Rio de Janeiro
Prentice-Hall, Inc., Englewood Cliffs, New Jersey

For Gloria—Bill Frakes

For Susan—Chris Fox

For my wife and friend Laurie and our children
Mary Elizabeth and Emily Elise—Brian Nejmeh

Contents

PREFACE viii

CHAPTER 1 INTRODUCTION 1

- 1.1 Software Engineering 1
- 1.2 UNIX System and C Language 2
- 1.3 State of the Art in Software Engineering 2
- 1.4 Software Products, Projects, and Methodologies 3
- 1.5 Software Project Size and Type 4
- 1.6 Software Life Cycle and Life-Cycle Models 5
- 1.7 Attributes of Software Quality 6
- 1.8 Goals and Organization of Book 8

CHAPTER 2 CONCEPT EXPLORATION AND REQUIREMENTS SPECIFICATION 10

- 2.1 Concept Exploration Phase 10
- 2.2 Requirements Specification Phase 13
- 2.3 Prototyping Aid in Early Phases of Life Cycle 17
- 2.4 Conclusion 24

CHAPTER 3	SOFTWARE DESIGN PROCESS	26
3.1	Software Design Concepts and Principles	27
3.2	Software Design Methods	31
3.3	Graphical Design Tools	39
3.4	Program Design Languages	41
3.5	Contents of a Design Document	42
CHAPTER 4	PROGRAM READABILITY	46
4.1	Internal Documentation	47
4.2	External Documentation	63
4.3	Conclusion	64
CHAPTER 5	LOW-LEVEL PROGRAMMING	66
5.1	Parameterization	66
5.2	C Preprocessor	70
5.3	C Function Libraries	73
5.4	Expressions	74
5.5	Types in C	76
5.6	Improvement of Efficiency	78
5.7	Conclusion	80
CHAPTER 6	HIGH-LEVEL PROGRAMMING	81
6.1	Implementation of Modules in C	81
6.2	C Modularization Example	86
6.3	Module Size and Complexity Standards	90
6.4	Conclusion	91
CHAPTER 7	UNIX TOOLS FOR CODING PHASE OF LIFE CYCLE	93
7.1	Program Generation Tools	94
7.2	Static Analysis Tools	105
7.3	Dynamic Analysis Tools	108
7.4	Conclusion	114
CHAPTER 8	SOFTWARE TESTING AND QUALITY ASSURANCE	117
8.1	Software Quality Assurance	117
8.2	Reviews	118
8.3	Code Inspections	121

8.4	Process Audits	124
8.5	Testing Process	125
8.6	Who Should Test Software	126
8.7	Testing Resource Allocation	127
8.8	Regression Testing	137
8.9	Debugging	139
8.10	Testing of account	142
CHAPTER 9	SOFTWARE MAINTENANCE AND CONFIGURATION MANAGEMENT	148
9.1	Software Maintenance	148
9.2	Software Configuration Management	152
CHAPTER 10	FUTURE TRENDS	161
10.1	Evolution of UNIX Programming Environment	161
10.2	User Interfaces	162
10.3	Data Repositories	163
10.4	Project Management Support	165
10.5	Document Production Support	166
10.6	Front-End Support	166
10.7	Conclusion	166
APPENDIX A	account PROJECT DOCUMENTS	169
APPENDIX B	account SOURCE CODE	201
APPENDIX C	ENGINEERING DOCUMENT TEMPLATES	234
APPENDIX D	RULES OF C PROGRAMMING PRACTICE	240
APPENDIX E	CODE INSPECTION CHECKLIST	244
APPENDIX F	REFERENCES	249
INDEX		256