

CONTENTS

Preface	xiii
Notes on the book's structure and layout.....	xvii
Acknowledgments	xxi

SECTION 1—FUNDAMENTALS

1. An example test series	1
The first cycle of testing	1
The second cycle of testing	11
What will happen in later cycles of testing	16
2. The objectives and limits of testing	17
You can't test a program completely	17
The tester's objective: Program verification?	23
So, why test?	25
3. Test types and their place in the software development process	27
Overview of the software development stages	30
Planning stages	32
Testing during the planning stages	33
Design stages	35
Testing during the design stages	39
Glass box code testing is part of the coding stage	41
Regression testing	49
Black box testing	50
Maintenance	57
4. Software errors	59
Quality	59
What is a software error?	60
Categories of software errors	60
5. Reporting and analyzing bugs	65
Write Problem Reports immediately	66
Content of the Problem Report	66
Characteristics of the Problem Report	74
Analysis of a reproducible bug	76
Tactics for analyzing a reproducible bug	79
Making a bug reproducible	82

SECTION 2—SPECIFIC TESTING SKILLS

6. The problem tracking system	87
The prime objective of a problem tracking system	90
The tasks of the system	90
Problem tracking overview	90
The users of the tracking system	97
Mechanics of the database	106
Further thoughts on problem reporting	115
Glossary	121
7. Test case design	123
Characteristics of a good test	124
Equivalence classes and boundary values	125
Visible state transitions	132
Race conditions and other time dependencies	133
Load testing	134
Error guessing	135
Function equivalence testing: automation, sensitivity analysis & random input	135
Regression testing: checking whether a bug fix worked	139
Regression testing: the standard battery of tests	140
Executing the tests	141
8. Testing printers (and other devices)	143
Some general issues in configuration testing	144
Printer testing	146
9. Localization testing	169
Was the base code changed?	170
Work with someone fluent in the language.	170
Is the text independent from the code?	171
Translated text expands	171
Character sets	171
Keyboards	172
Text filters	172
Loading, saving, importing, and exporting high and low ASCII	173
Operating system language	173
Hot keys	173
Garbled in translation	173
Error message identifiers	174
Hyphenation rules	174
Spelling rules	174
Sorting rules	174
Uppercase and lowercase conversion	174
Underscore rules	174

Printers	175
Sizes of paper	175
CPU's and video	175
Rodents	175
Data formats and setup options	175
Rulers and measurements	176
Culture-bound graphics	176
Culture-bound output	176
European product compatibility	176
Memory availability	176
Do GUIs solve the problem?	177
Automated testing	177
10. Testing user manuals	179
Effective documentation	179
The documentation tester's objectives	180
How testing documentation contributes to software reliability	181
Become the technical editor	182
Working with the manual through its development stages	183
Online help	188
11. Testing tools	189
Fundamental tools	189
Automated acceptance and regression tests	191
Standards	197
Translucent-box testing	200
12. Test planning and test documentation	203
The overall objective of the test plan: product or tool?	204
Detailed objectives of test planning and documentation	205
What types of tests to cover in test planning documents	210
A strategy for developing components of test planning documents	213
Components of test planning documents	217
Documenting test materials	242
A closing thought	253

SECTION 3—MANAGING TESTING PROJECTS AND GROUPS

13. Tying it together	255
Software development tradeoffs	257
Software development models	258
Quality-related costs	264
The development time line	266
Product design	267
Fragments coded: first functionality	274
Almost alpha	275

Alpha	277
Pre-beta	286
Beta	286
User interface (UI) freeze	293
Pre-final	295
Final integrity testing	299
Release	301
Project post-mortems	301
14. Legal consequences of defective software	303
Breach of contract	305
Torts: lawsuits involving fault	317
Whistle blowing	340
15. Managing a testing group	343
Managing a testing group	344
The role of the testing group	345
A test group is not an unmixed blessing	349
An alternative? Independent test agencies	350
Scheduling tips	352
Your staff	359
Appendix: common software errors	363
User interface errors	375
Error handling	396
Boundary-related errors	399
Calculation errors	401
Initial and later states	403
Control flow errors	406
Errors in handling or interpreting data	416
Race conditions	421
Load conditions	423
Hardware	427
Source, version, and ID control	430
Testing errors	432
References	437
Index	451
About the Authors	480