

The Pragmatic Programmer



from journeyman
to master

Soranaree University of Technology



31051000657425

Andrew Hunt
David Thomas

Foreword by Ward Cunningham

Contents

FOREWORD	xiii
PREFACE	xvii
1 A PRAGMATIC PHILOSOPHY	1
1. The Cat Ate My Source Code	2
2. Software Entropy	4
3. Stone Soup and Boiled Frogs	7
4. Good-Enough Software	9
5. Your Knowledge Portfolio	12
6. Communicate!	18
2 A PRAGMATIC APPROACH	25
7. The Evils of Duplication	26
8. Orthogonality	34
9. Reversibility	44
10. Tracer Bullets	48
11. Prototypes and Post-it Notes	53
12. Domain Languages	57
13. Estimating	64
3 THE BASIC TOOLS	71
14. The Power of Plain Text	73
15. Shell Games	77
16. Power Editing	82
17. Source Code Control	86
18. Debugging	90
19. Text Manipulation	99
20. Code Generators	102

4 PRAGMATIC PARANOIA	107
21. Design by Contract	109
22. Dead Programs Tell No Lies	120
23. Assertive Programming	122
24. When to Use Exceptions	125
25. How to Balance Resources	129
5 BEND, OR BREAK	137
26. Decoupling and the Law of Demeter	138
27. Metaprogramming	144
28. Temporal Coupling	150
29. It's Just a View	157
30. Blackboards	165
6 WHILE YOU ARE CODING	171
31. Programming by Coincidence	172
32. Algorithm Speed	177
33. Refactoring	184
34. Code That's Easy to Test	189
35. Evil Wizards	198
7 BEFORE THE PROJECT	201
36. The Requirements Pit	202
37. Solving Impossible Puzzles	212
38. Not Until You're Ready	215
39. The Specification Trap	217
40. Circles and Arrows	220
8 PRAGMATIC PROJECTS	223
41. Pragmatic Teams	224
42. Ubiquitous Automation	230
43. Ruthless Testing	237
44. It's All Writing	248
45. Great Expectations	255
46. Pride and Prejudice	258

Appendices

A RESOURCES	261
Professional Societies	262
Building a Library	262
Internet Resources	266
Bibliography	275
 B ANSWERS TO EXERCISES	 279
 INDEX	 309