## Contents

<table>
<thead>
<tr>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreword</td>
<td>xv</td>
</tr>
<tr>
<td>Preface</td>
<td>xix</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>xxv</td>
</tr>
</tbody>
</table>

### SECTION 1  INTRODUCTION TO SOFTWARE COST ESTIMATION

#### Chapter 1. Introduction

- How Software Cost-Estimating Tools Work                                  4
- Cautions About Accidental Omissions from Estimates                       15
- Software Cost Estimating and Other Development Activities                17
- References                                                              20

#### Chapter 2. The Origins of Software Cost Estimation                   23

- The Early History of Software Cost Estimation                           24
- The Expansion and Use of Functional Metrics for Software Cost Estimating 28
- References                                                              32

#### Chapter 3. Six Forms of Software Cost Estimation                     33

- Overview of Manual Software-Estimating Methods                          34
- Overview of Automated Software-Estimating Methods                       36
- Comparison of Manual and Automated Estimates for Large Software Projects 48
- References                                                              49

#### Chapter 4. Software Cost-Estimating Tools and Project Success and Failure Rates  53

- Probabilities of Software Project Success or Failure                    55
- References                                                              59
Chapter 5. Sources of Error in Software Cost Estimation

Judging the Accuracy of Software Cost Estimates

Classes of Software Estimation Errors

References

SECTION 2 PRELIMINARY ESTIMATION METHODS


Rules of Thumb Based on Lines-of-Code Metrics

Rules of Thumb Based on Ratios and Percentages

Rules of Thumb Based on Function Point Metrics

Function Point Sizing Rules of Thumb

Rules of Thumb for Schedules, Resources, and Costs

Rules of Thumb Using Activity-Based Cost Analysis

Summary and Conclusions

References

Chapter 7. Manual Estimating Methods Derived from Agile Projects and New Environments

Metrics Used for Rules of Thumb

Rules of Thumb for Manual Software Cost Estimates

Component-Based Development

Dynamic System Development Method (DSDM)

Enterprise Resource Planning (ERP) Deployment

Extreme Programming (XP)

International Outsourcing

Object-Oriented (OO) Development

Capability Maturity Model (CMM)

Software Methods with Only Partial Rules of Thumb

Cleanroom Development

Crystal Development Approach

Feature-Driven Development (FDD)

ISO 9000-9004 Quality Standards

Iterative or Incremental Development

Pattern-Based Software Development

Quality Function Deployment (QFD)

Rapid Application Development (RAD)

Scrum

six-sigma for Software

Spiral Software Development

Unified Modeling Language (UML)

Use Cases for Software Requirements

Web-Based Applications

Summary and Conclusions

References
Chapter 8. Automated Estimates from Minimal Data 189
   Stage 1: Recording Administrative and Project Information 190
   Stage 2: Preliminary Sizing of Key Deliverables 203
   Stage 3: Producing a Preliminary Cost Estimate 219
   Summary and Conclusions 224
   References 225

SECTION 3 SIZING SOFTWARE DELIVERABLES

Chapter 9. Sizing Software Deliverables 229
   General Sizing Logic for Key Deliverables 229
   Sizing Methods Circa 2007 230
   Pattern Matching from Historical Data 232
   Using Historical Data to Predict Growth in Requirements 233
   Mathematical or Statistical Attempts to Extrapolate Size from Partial Requirements 234
   Arbitrary Rules of Thumb for Adding Contingency Factors 235
   Freezing Requirements at Fixed Points in Time 236
   Producing Formal Cost Estimates Only for Subsets of the Total Application 237
   Volume of Function Point Data Available 245
   Software Complexity Analysis 247
   Software Sizing with Reusable Components 258
   Overview of the Basic Forms of Software Sizing Metrics 260
   Source Code Sizing 269
   Sizing Object-Oriented Software Projects 275
   Sizing Text-Based Paper Documents 277
   Sizing Graphics and Illustrations 283
   Sizing Bugs or Defects 286
   Sizing Test Cases 293
   The Event Horizon for Sizing Software Artifacts 295
   What Is Known as a Result of Sizing Software Projects 297
   Strengths and Weaknesses of Software Size Metrics 299
   Summary and Conclusions 301
   References 302

SECTION 4 COST-ESTIMATING ADJUSTMENT FACTORS

Chapter 10. Compensation and Work-Pattern Adjustments 307
   Manual and Automated Methods of Adjustment 308
   Exclusions from Normal Software Cost Estimates 310
   Setting Up the Initial Conditions for a Cost Estimate 313
   Variations in Burden Rates or Overhead Costs 316
   Variations in Work Habits and Unpaid Overtime 318
   References 324
Chapter 16. Estimating Design Inspections 421
- Inspection Literature 421
- Inspection Process 423
- Value of Inspections 426
- References 432

Chapter 17. Estimating Programming or Coding 435
- The Impact of Reusability on Programming 442
- The Impact of Experience on Programming 444
- The Impact of Bugs or Errors on Programming 444
- The Impact of Unpaid Overtime on Programming 446
- The Impact of Creeping Requirements on Programming 448
- The Impact of Code Structure and Complexity on Programming 449
- The Impact of Unplanned Interruptions on Programming 450
- The Impact of Application Size on Programming 451
- The Impact of Office Space and Ergonomics on Programming 452
- The Impact of Tools on Programming 454
- The Impact of Programming Languages on Programming 455
- The Impact of Schedule Pressure on Programming 459
- References 459

Chapter 18. Estimating Code Inspections 461
- Code Inspection Literature 461
- Effectiveness of Code Inspections 462
- Considerations for Estimating Code Inspections 466
- References 470

- The Literature on Change Management 473
- Measuring Software Change 475
- Changes in User Requirements 479
- Changes in Specifications and Design 480
- Changes Due to Bugs or Defect Reports 481
- Summary and Conclusions 482
- References 483

- General Forms of Software Testing 491
- Specialized Forms of Software Testing 495
- Forms of Testing Involving Users or Clients 498
- Number of Testing Stages 499
- Testing Pattern Variations by Industry and Type of Software 501
- Testing Pattern Variations by Size of Application 503
Testing Stages Noted in Lawsuits Alleging Poor Quality 504
Using Function Points to Estimate Test-Case Volumes 505
Using Function Points to Estimate the Numbers of Test Personnel 507
Testing and Defect-Removal Efficiency Levels 508
Using Function Points to Estimate Testing Effort and Costs 510
Testing by Developers or by Professional Test Personnel 512
Test Case Coverage 514
The Factors That Affect Testing Performance 514
References 515

Chapter 21. Estimating User and Project Documentation 519
Estimating Tools and Software Documentation 521
Quantifying the Number and Sizes of Software Document Types 523
Software Documentation Tools on Lagging and Leading Projects 527
References 529

Chapter 22. Estimating Software Project Management 531
The Roles of Software Project Management 535
Project Managers Who Are Also Technical Contributors 537
Project Management for Hybrid Projects Involving Hardware and Software 538
Project Management and External Schedule Pressures 538
Project Management Tools 539
Project Management on Large Systems with Many Managers 542
Time-Splitting, or Managing Several Projects Simultaneously 544
The Span of Control, or Number of Staff Members per Manager 545
Managing Multiple Occupation Groups 546
The Presence or Absence of Project Offices for Large Systems 548
Experience Levels of Software Project Managers 549
Quality-Control Methods Selected by Project Managers 550
Project Managers and Metrics 551
Summary of Project Management Findings 551
References 551

SECTION 6 MAINTENANCE AND ENHANCEMENT COST ESTIMATING

Chapter 23. Maintenance and Enhancement Estimating 557
Nominal Default Values for Maintenance and Enhancement Activities 562
Metrics and Measurement Problems with Small Maintenance Projects 566
Best and Worst Practices in Software Maintenance 567
Software Entropy and Total Cost of Ownership 570
Installing New Releases and Patches from Software Vendors 572
Major Enhancements 573
Minor Enhancements 574
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrics Conversion</td>
<td>604</td>
</tr>
<tr>
<td>Automatic Sizing from User Requirements</td>
<td>606</td>
</tr>
<tr>
<td>Activity-Based Costs for Agile, Object-Oriented, and Web Projects</td>
<td>608</td>
</tr>
<tr>
<td>Complexity Analysis of Software Applications</td>
<td>610</td>
</tr>
<tr>
<td>Value Analysis of Software Applications</td>
<td>611</td>
</tr>
<tr>
<td>Risk Analysis and Software Cost Estimates</td>
<td>613</td>
</tr>
<tr>
<td>Including Specialists in Software Cost Estimates</td>
<td>615</td>
</tr>
<tr>
<td>Reuse Analysis and Software Cost Estimates</td>
<td>617</td>
</tr>
<tr>
<td>Process Improvement Estimation</td>
<td>622</td>
</tr>
<tr>
<td>Methodology Analysis and Software Cost Estimating</td>
<td>625</td>
</tr>
<tr>
<td>Summary and Conclusions About Software Cost-Estimating Research</td>
<td>629</td>
</tr>
</tbody>
</table>

Index 631