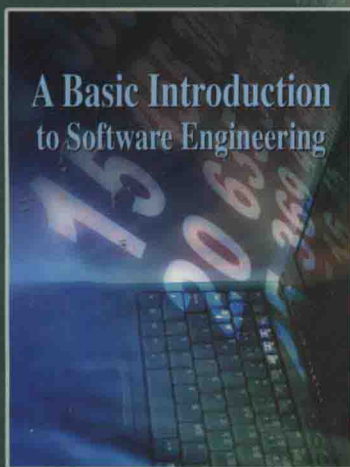


计算机科学教材系列

软件工程基础教程

A Basic Introduction to Software Engineering



英文版

田文洪 主编

王玉林 周俊临 袁敏 副主编



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

软件工程基础教程

A Basic Introduction to Software Engineering

英文版

本书由浅入深、循序渐进地介绍了规范化的软件开发方法所涉及的概念和方法。内容包括过程模型、软件需求、设计工程、生产率和工作量度量、软件测试、软件维护、质量管理、项目管理和风险管理等内容，最后介绍了一些常用的软件开发工具，并给出编码风格上的建议。

本书内容注重理论与实践结合，参考了IEEE和ACM关于软件工程的10个知识体系内容。针对中国学生英文水平，做了针对性的编写，并对重点难点内容进行了中文解释。本书配有在线习题、电子课件、案例分析、实验指导等教学资源，可登录电子工业出版社华信教育资源网（www.hxedu.com.cn），免费注册、下载。

本书将阐释

- 软件工程的发展背景和发展趋势如何？
- IEEE和ACM软件知识体系建议的软件工程的主要内容有哪些？
- 软件工程的共同演进方法核心内容是什么？
- 经典的软件流程包含哪些主要环节？
- 软件工程的主要理论与实践如何结合？
- 如何建立一个全局的需求分析、设计、测试与维护的软件工程观？
- 常用的软件开发工具和编码规范有哪些？



策划编辑：史鹏举
责任编辑：史鹏举
责任美编：孙焱津



欢迎登录 **免费** 获取本书教学资源
<http://www.hxedu.com.cn>



ISBN 978-7-121-15408-9



9 787121 154089 >

定价：39.00元

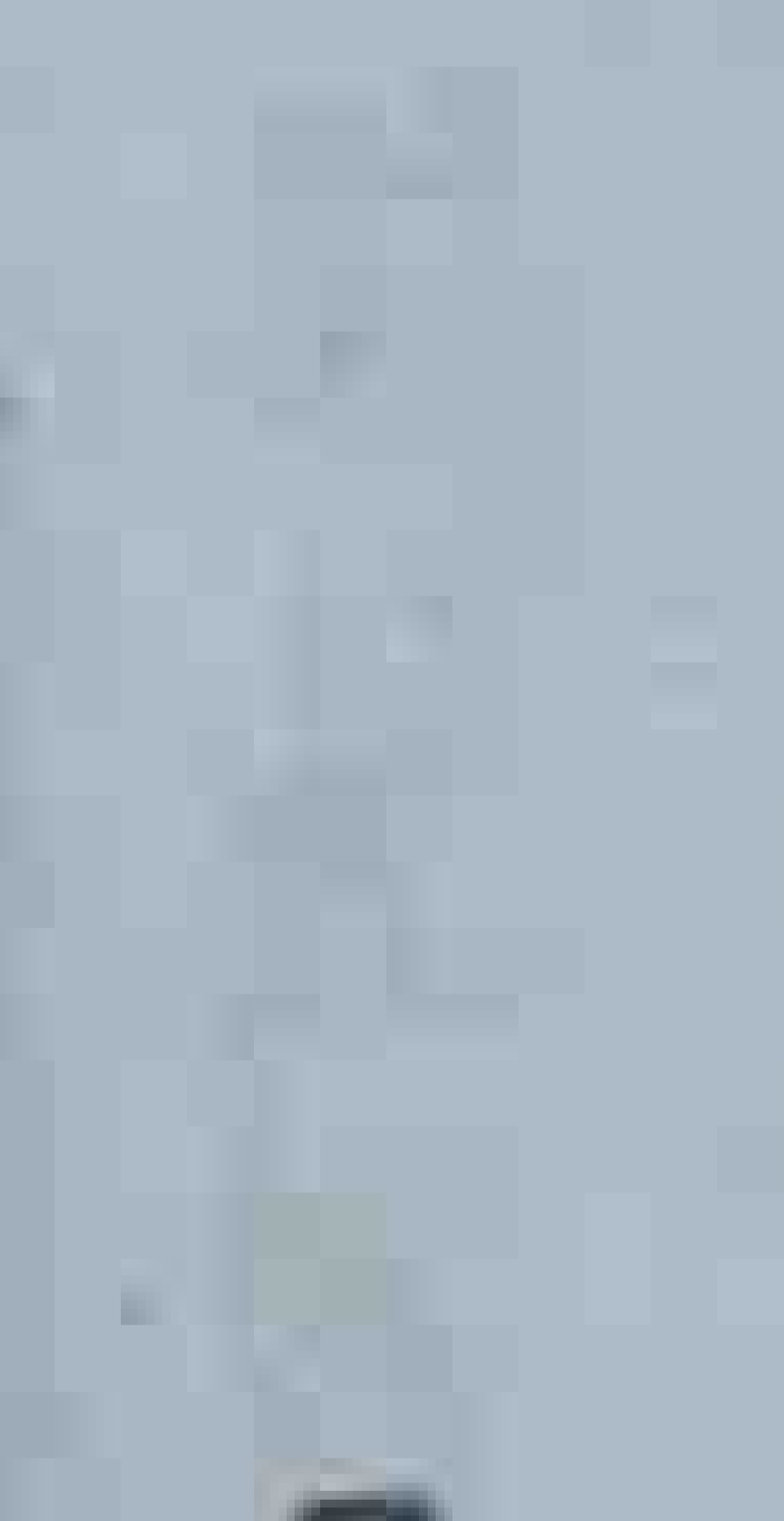
本书贴有激光防伪标志，凡没有防伪标志者，属盗版图书。

计算机科学教程系列

软件工程基础教程

A Basic Introduction to Software Engineering

英文版



电子工业出版社



软件工程基础教程(英文版)

A Basic Introduction to Software Engineering

田文洪 主编

王玉林 周俊临 袁 敏 副主编

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

为了得到好的软件产品,软件开发必须遵循一定的规范和流程。

本书由浅入深、循序渐进地介绍了规范化的软件开发方法所涉及的概念和方法。内容包括过程模型、软件需求、设计工程、生产率和工作量度量、软件测试、软件维护、质量管理、项目管理和风险管理等内容,最后介绍了一些常用的软件开发工具,并给出编码风格上的建议。

本书内容注重理论与实践结合,参考了 IEEE 和 ACM 关于软件工程的 10 个知识体系内容。针对中国学生英文水平,做了针对性的编写,并对重点难点内容进行了中文解释。本书配有在线习题、电子课件、案例分析、实验指导等教学资源,可登录电子工业出版社华信教育资源网(www.hxedu.com.cn),免费注册、下载。

通过本书的学习,可使学生了解软件工程的基本知识和方法,熟悉软件工程的主要环节,掌握软件工程传统经典和现代高级的理论、方法,并能应用到实际的软件项目开发中。

本书贯彻理论与实践相结合的原则,深入浅出,配以大量实例分析,既可作为本科生及低年级研究生“软件工程”课程的教材,也可供软件工程从业人员学习、参考。

Brief Introduction

To develop software with high quality, one has to follow the principles and procedures in software engineering.

This book provides an introduction to software engineering. The book includes process models, software requirements, design engineering, project productivity and effort measurement, software testing, software maintenance, quality management, project management and risk managements etc. In the end of the book, some commonly used software development tools and some good coding styles are also introduced.

This book focuses on theory and practice and refers to contents about software engineering suggested both by IEEE ACM and SWEBOK (Software Engineering Body of Knowledge). The English writing and contents are specially prepared for general Chinese college students to understand easily. This book has online exercises, electronic courseware, and laboratory guidance.

This book will enable students to understand the basic software engineering knowledge and methods, know the main part of software engineering, master classic and modern advanced theories and methods of software engineering, and apply them to real software project development.

This book implements the principle of combining theory and practice, containing a large number of cases, is suitable for undergraduate and graduate students in lower grades.

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容
版权所有·侵权必究

图书在版编目(CIP)数据

软件工程基础教程:英文/田文洪主编. —北京:电子工业出版社,2012.1
ISBN 978-7-121-15408-9

I. ①软… II. ①田… III. ①软件工程—高等学校—教材—英文 IV. ①TP311.5

中国版本图书馆 CIP 数据核字(2011)第 252568 号

策划编辑:史鹏举

责任编辑:史鹏举

印 刷:北京东光印刷厂

装 订:三河市皇庄路通装订厂

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:16 字数:533 千字

印 次:2012 年 1 月第 1 次印刷

定 价:39.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

前 言

“软件工程”是高等院校计算机教学计划中的一门核心课程，主要内容包括软件工程背景介绍，软件过程模型，软件需求分析，软件设计，软件编码规范，软件测试，软件项目管理，软件项目度量，软件风险管理，软件质量管理等内容。

本书用理论与实践紧密结合的共同演进方法进行介绍，既介绍了传统软件生命周期过程中的主要内容，又介绍了现代软件工程中比较新颖的技术方法，包括新的过程模型、软件作为一种服务等近年来新兴的方法和技术。

本教材涵盖 IEEE 和 ACM 联合研究项目——SWEBOK（工程知识体系）包含的 10 个知识领域的主要内容，主要介绍软件工程的背景知识、软件过程模型、需求分析、需求建模、设计工程、体系结构设计、编码和新的技术趋势需求、测试技术与策略、项目管理概要、风险分析与管理、软件质量管理等内容。通过理论与实践（实际项目）的紧密结合介绍，使得学生在学习的过程中了解和掌握软件工程的基本概念、原理和方法，并能应用到实际项目中去。

本书将阐释：

- 软件工程的发展背景和发展趋势如何？
- IEEE 和 ACM 软件知识体系建议的软件工程的主要内容有哪些？
- 软件工程的核心内容是什么？
- 经典的软件流程包含哪些主要环节？
- 软件工程的主要理论与实践如何结合？
- 如何建立一个全局的需求分析、设计、测试与维护的软件工程观？
- 常用的软件开发工具和编码规范有哪些？

本教材基于作者多年来教学实践与改革的经验，特别是收集了学生的反馈，结合目前软件教材的优点并考虑到学生的学习特点而编著。主要特色包括以下几个方面：

(1) 增强理论与实践相结合，案例贯穿全教材：教学案例源于已实际开发的项目，从教学一开始直至结束，所有概念、方法都会应用于这些教学案例中。随着案例项目的进展分析，完成全部教学内容，这种方法国外相当流行，的确可以在培养学生分析解决问题的能力上发挥更大作用，因而值得研究和发扬。

(2) 突出组织逻辑，增加趣味性：目前的国内教材和选用的国外经典教材，用于本科教学后，学生的普遍反映还是概念原理介绍过多，内容组织的逻辑思路不是很明显，以及介绍得比较技术性，不是很生动等问题。我们希望针对学生的反馈在新编教材中进行改进。

(3) 增加软件产业热门和急需的技术知识：适当介绍软件产业热门和急需的软件/平台等知识，开阔学生的视野并为他们就业做好准备。

本书主要内容包括软件工程简介（第 1 章），过程模型（第 2 章），软件需求（第 3 章），软件设计工程（第 4 章），软件生产率和工作量度量（第 5 章），软件质量管理（第 6 章），测试方法（第 7 章），测试策略（第 8 章），软件维护（第 9 章），软件项目管理（第 10 章），软件风险分析和管理（第 11 章），软件工程开发工具（第 12 章），最后为喜欢实践的读者在附录 A 中准备了“软件项目的一般风格和编码标准”。

本书既可作为本科生及低年级研究生“软件工程”课程的教材，也可供软件工程从业人员学习、参考。

本书由田文洪主编，王玉林、周俊临、袁敏为副主编。孙夏爽、胡金安、董旭、钟元棕、景晨、王浩严为本书的编写做出了贡献，在此表示诚挚的感谢。本书是编写组成员对以上内容大量理论知识与实践经验的积累结果，因时间仓促，可能存在不妥之处，欢迎指正，请发邮件至：SE_Coevolutionary@gmail.com

编 者

Foreword

“Software Engineering” is a core course for students in computer-related disciplines of higher level education. The main topics include software engineering background, software process models, software requirements analysis, software design, software coding standards, software testing, software project management, software project metrics, software risk management, software quality management and so on.

This book is characterized by close integration of theory and practice. It not only describes main content of the traditional software life cycle process, but also introduces some new technologies in the modern software engineering methods, including new process models, software as a service, new parallel distributed programming models etc.

This book covers main content of 10 areas suggested by IEEE and ACM Joint research projects - SWEBOK (The Software Engineering Body of Knowledge). It introduces the background of software engineering, software process models, requirements analysis, requirements modeling, design engineering, architecture, design, coding, and requirements of new technology trends, testing techniques and strategies, project management summary, risk analysis and management, software quality management and so on. Through the close combination of theory and practice, students can understand and master the basic concepts, principles and methods of software engineering and apply them to real projects.

This book will explain:

- What are software engineering backgrounds and development trends?
- What are the major contents of software engineering suggested by IEEE ACM and SWEBOK?
- What are core contents of software engineering?
- Which key elements are included in classic software engineering process?
- How to combine main software engineering theories and practices?
- How to create a global view and world view of analysis, design, testing and maintenance in software engineering?
- What are commonly used software development tools and coding standards?

This book accumulates experience of the teaching practice and reform during past years, collects feedbacks from students, combines with the advantages of the current software engineering books and takes into account the study characteristics of students. Key features of this book include but not limited to the followings:

(1) Enhancing the integration of theory and practice through case studies: teaching cases from actual development projects are selected, all the concepts, methods will be applied to the one or two cases. All teaching materials and case-studies are carefully chosen for students so that there are not too much or too less contents. Teaching by case-study is popular abroad, this approach can help students the ability to analyze and solve problems.

(2) Highlighting the logic of the organization and increasing students' interests to learn: through years usage of materials and the classic textbook for teaching, the students generally reflect that

there are too many principles or concepts in them; content organization and logical thinking is not very clear; too many technical issues are described boringly.

(3) Introducing popular and hot technical knowledge in the software industry: the appropriate introduction to popular software platforms, such as software as a service to broaden students horizons and prepare for their employment.

This book includes introduction to software engineering (Chapter 1), process models (Chapter 2), software requirements (Chapter 3), software design engineering (Chapter 4), software productivity and effort metrics (Chapter 5), software quality management (Chapter 6), testing techniques (Chapter 7), test strategies (Chapter 8), software maintenance (Chapter 9), software project management (Chapter 10), software risk analysis and management (Chapter 11), software engineering development tools (Chapter 12), and an appendix of “General Style And Coding Standards For Software Projects.”

This book is good for senior undergraduate students and graduate students and can be a reference guide to researchers.

This book is a product of accumulation of large number of theories and practices, and edited by a group members through extensive teaching and research experience. Due to time constraints, there may be something missing or mistake, please correct and email to **SE_Coevolutionary@gmail.com**

Contents

目 录

Unit 1 Introduction to Software Engineering	1
软件工程介绍	
Learning Objectives	1
学习目标	
1.1 A Brief Introduction to Software Engineering	1
软件工程简介	
1.1.1 The Pioneering Era	2
开创时期	
1.1.2 1945 to 1965: The Origins	2
1945 到 1965: 发起	
1.1.3 1965 to 1985: The Software Crisis	2
1965 到 1985: 软件危机	
1.1.4 1985 to 1989: No Silver Bullet	3
1985 到 1989: 没有银弹	
1.1.5 1990 to 1999: Prominence of the Internet	4
1990 到 1999: Internet 盛行	
1.1.6 2000 to Present: Lightweight Methodologies	4
2000 年至今: 轻量级方法	
1.1.7 Current Trends in Software Engineering	4
软件工程的发展趋势	
1.1.8 Software Engineering Today	5
今天的软件工程	
1.2 Software	5
软件	
1.2.1 A Formal Definition of Software	5
软件的正式定义	
1.2.2 Characteristics of Software	5
软件的特征	
1.2.3 The Dual Roles of Software	7
软件的双重角色	
1.2.4 Classification of Software	7
软件分类	
1.2.5 Why does Software Change Very Fast?	7
为什么软件变化快?	
1.3 Why Do We Need Software Engineering?	8
为什么需要软件工程?	
1.3.1 Software Crisis	8
软件危机	
1.3.2 An Economist's View of Software and Software Engineering	8
一个经济学家对软件和软件工程的看法	
1.3.3 The Formal Definition of Software Engineering	9
软件工程的正式定义	

1.4	Major Concerns and Objectives of Software Engineering	9
	软件工程的目标和担忧	
1.5	Seven Principles for Software Engineering	9
	软件工程的七个原则	
1.5.1	Manage Using a Phased Lifecycle Plan	9
	使用阶段性的生命周期管理计划	
1.5.2	Perform Continuous Validation	10
	进行持续验证	
1.5.3	Maintain Disciplined Product Control	10
	维持有效的产品控制	
1.5.4	Use Modern Programming/Engineering Practices	10
	运用现代编程/工程实践	
1.5.5	Maintain Clear Accountability for Results	10
	维护清晰的问责机制	
1.5.6	Use Better and Fewer People	10
	用更好更少的人	
1.5.7	Maintain a Commitment to Improve the Process	11
	保持过程改进	
1.6	Major Topics in Software Engineering	11
	软件工程中的主要课题	
1.7	Relationship to Other Disciplines	13
	其他相关学科	
1.8	Some Misunderstandings about Software Engineering	14
	一些对软件工程的错误认识	
1.9	Software Engineering Code of Professional and Ethical Responsibility	15
	软件工程的专业和道德责任	
	Summary	16
	总结	
	Notes	16
	重点注释	
	Glossary of this unit	17
	术语	
	Quiz	17
	习题	
	References	18
	参考文献	
	Further Reading	18
	扩展阅读	
Unit 2 Process Models		21
	过程模型	
	Learning Objectives	21
	学习目标	
2.1	Introduction	21
	简介	
2.2	What is Software Process?	22
	什么是软件过程?	
2.3	Maturity Models	23
	成熟度模型	
2.4	Software Process Models	24
	软件过程模型	

2.4.1	The Waterfall Model 瀑布模型	24
2.4.2	The Evolutionary Model 演化模型	26
2.4.3	Component-based Model 基于组件的模型	28
2.4.4	Incremental Process Models 增量模型	29
2.4.5	Some New Process 一些新模型	33
2.5	How to Choose Process Models? 如何选择过程模型?	33
2.6	The Relationship between Process and Product 过程和产品的关系	33
	Summary 总结	34
	Notes 重点注释	34
	Glossary of this unit 术语	35
	Quiz 习题	35
	References 参考文献	36

Unit 3 Software Requirements 37

软件需求

	Learning Objectives 学习目标	37
3.1	Software Requirements Fundamentals 软件需求基础	37
3.1.1	Definition of a Software Requirement 软件需求定义	38
3.1.2	Product and Process Requirements 产品和过程需求	38
3.1.3	Functional and Non-functional Requirements 功能和非功能需求	39
3.1.4	Quantifiable Requirements 可量化的需求	39
3.1.5	System and Software Requirements 系统需求和软件需求	39
3.2	Requirements Process 需求过程	40
3.2.1	Process Models 过程模型	40
3.2.2	Process Actors 过程角色	41
3.3	Requirements Elicitation 需求获取	41
3.3.1	Requirements Sources 需求来源	42

3.3.2	Elicitation Techniques	42
	需求获取方法	
3.4	Requirements Analysis	43
	需求分析	
3.4.1	Requirements Classification	44
	需求分类	
3.4.2	Conceptual Modeling	45
	概念建模	
3.4.3	Architectural Design and Requirements Allocation	46
	结构设计和需求配置	
3.4.4	Requirements Negotiation	46
	需求谈判	
3.5	Requirements Specification	47
	需求规格	
3.5.1	The System Definition Document	47
	系统定义文档	
3.5.2	The Software Requirements Specification	48
	需求规格说明书	
3.6	Requirements Validation	52
	需求验证	
3.6.1	Requirements Reviews	52
	需求审查	
3.6.2	Prototyping	53
	原型	
3.6.3	Model Validation	53
	模型验证	
3.6.4	Acceptance Tests	54
	验收测试	
3.7	Practical Considerations	54
	实际考虑因素	
3.7.1	Iterative Nature of the Requirements Process	54
	需求过程的迭代	
3.7.2	Change Management	55
	变更管理	
3.8	Example: Online Recruiting System Requirement Specification	55
	例: 网上招聘系统的需求规格说明书	
3.8.1	Introduction	55
	引言	
3.8.2	System Definitions	55
	系统定义	
3.8.3	Running Environment	56
	应用环境	
3.8.4	Function Specifications	57
	功能规格	
3.8.5	Performance Requirements	59
	性能需求	
3.8.6	Products Submitting	59
	产品提交	
3.8.7	Implementation Constrains	59
	实现约束	
	Summary	59
	总结	

Notes	59
重点注释	
Glossary of this unit	60
术语	
Quiz	60
习题	
References	60
参考文献	
Unit 4 Software Design Engineering	62
软件设计工程	
Learning Objectives	62
学习目标	
4.1 Introduction to Design Engineering	62
设计工程简介	
4.2 Design Process and Quality	63
设计过程和质量	
4.3 Design Technologies	64
设计技术	
4.3.1 Abstraction	64
抽象	
4.3.2 Design Patterns	65
设计模式	
4.3.3 Modularization	65
模块化	
4.3.4 Information Hiding	67
信息隐藏	
4.3.5 Function Independence	67
功能独立	
4.3.6 Refinement	68
改进	
4.3.7 Refactoring	68
重构	
4.4 Design Models	68
设计模型	
4.4.1 Data Design	70
数据结构设计	
4.4.2 Architecture Design	72
架构设计	
4.4.3 Interface Design	76
接口设计	
4.4.4 Component Design	76
组件设计	
4.4.5 Deploy Design	77
部署设计	
4.5 Design Case Study	80
设计实例	
Summary	82
总结	
Notes	83
重点注释	

Glossary of this unit	84
术语	
Quiz	84
习题	
References	84
参考文献	
Unit 5 Software Productivity and Effort Metrics	86
软件生产率和工作量度量	
Learning Objectives	86
学习目标	
5.1 Introduction to Software Productivity and Project Effort Estimation	86
简介	
5.2 Software Productivity Estimation	88
软件生产率估计	
5.2.1 Size-based Metrics—Direct Measurement	89
基于规模的度量——直接度量	
5.2.2 Function-points Metrics—Indirect Measurement	90
功能点度量——间接度量	
5.2.3 The Relationship Between LOC and FP Based Metrics	92
LOC 和 FP 度量的关系	
5.2.4 How to Use These Metrics for Productivity Estimation?	92
如何运用这些度量进行生产率估计?	
5.3 Estimation for Project Effort	93
项目工作量估计	
5.3.1 Algorithmic Cost Modeling —Empirical Measurement	93
算法成本模型——基于经验的度量	
5.3.2 The COCOMO Model	94
COCOMO 模型	
5.3.3 Estimation of Effort in Project Planning	95
项目计划中的工作量度量	
5.3.4 Estimation of Effort by Tasks Breakdown	95
通过任务分解来估算工作量	
5.3.5 Estimation of Effort by Current Available Resources	96
通过目前可用的资源来估算工作量	
Summary	96
总结	
Notes	96
重点注释	
Glossary of this unit	98
术语	
Quiz	98
习题	
References	98
参考文献	
Unit 6 Software Quality Management	99
软件质量管理	
Learning Objectives	99
学习目标	
6.1 An Introduction to Software Quality Management	99
简介	

6.2	Software Quality	100
	软件质量	
6.2.1	Definition of Software Quality	100
	软件质量定义	
6.2.2	Quality Control	101
	质量控制	
6.2.3	Quality Assurance	101
	质量保证	
6.2.4	Quality Costs	102
	质量成本	
6.3	Software Quality Assurance	102
	软件质量保证	
6.3.1	Background Introduction	103
	背景介绍	
6.3.2	SQA Activities	104
	SQA 活动	
6.4	Software Review	106
	软件审查	
6.4.1	Impact of Software Bug Costs	106
	软件错误的代价	
6.4.2	Software Defect Indicator and Removal	107
	软件缺陷的发现和消除	
6.5	Formal Technique Review	107
	正式技术审查	
6.5.1	Review Report	107
	审查报告	
6.5.2	Review Tutorial	108
	审查指南	
6.6	Software Reliability	109
	软件可靠性	
6.6.1	Measurement of Reliability and Availability	109
	可靠性和可用性度量	
6.6.2	Software Security	109
	软件安全	
6.7	ISO 9000 Family of Standards	109
	ISO 9000 标准	
	Summary	113
	总结	
	Notes	113
	重点注释	
	Glossary of this unit	114
	术语	
	Quiz	114
	习题	
	References	114
	参考文献	
Unit 7	Testing Techniques	116
	测试方法	
	Learning Objectives	116
	学习目标	
7.1	Testing Introduction	116
	测试简介	

7.2	Testing Objective	117
	测试目标	
7.3	Testing Evaluation	117
	测试评估	
7.3.1	Coverage/Thoroughness Measures	117
	覆盖性/彻底性度量	
7.3.2	Fault Seeding	117
	故障植入	
7.4	Testing Techniques	117
	测试方法	
7.4.1	White-box Testing	117
	白盒测试	
7.4.2	Black-box Testing	122
	黑盒测试	
7.5	Example: Testing Design of Online Recruiting System	123
	例: 网上招聘系统的测试设计	
7.5.1	Introduction	123
	简介	
7.5.2	Testing Design	124
	测试设计	
7.5.3	Testing Cases	124
	测试用例	
	Summary	124
	总结	
	Notes	125
	重点注释	
	Glossary of this unit	125
	术语	
	Quiz	126
	习题	
	References	126
	参考文献	
Unit 8	Testing Strategies	127
	测试策略	
	Learning Objectives	127
	学习目标	
8.1	Testing Strategies Introduction	127
	测试策略简介	
8.2	Unit Testing	127
	单元测试	
8.2.1	Unit Testing Considerations	128
	单元测试考虑因素	
8.2.2	Unit Testing Procedures	129
	单元测试过程	
8.3	Integration Testing	130
	集成测试	
8.3.1	Top-down Integration	130
	自顶向下集成	
8.3.2	Bottom-up Integration	131
	自底向上集成	