

高等学校教材

王庆育 编著

软件工程



清华大学出版社

高等学校教材

软 件 工 程

王庆育 编著

清华大学出版社
北 京

内 容 简 介

本书全面系统地讲授软件工程的理论与实践,力求从不同的角度帮助读者迅速掌握软件工程的理论、方法和思路。全书共分软件工程基础、软件项目管理、传统的软件工程、面向对象的软件工程、软件工程的最新技术等五个部分。内容主要包括:软件工程概论、软件过程、软件建模语言、软件项目、团队管理、软件配置管理、系统设计、软件测试、面向对象的分析与设计、编码实现、基于构件的软件工程、客户服务器与 Web 软件工程、设计模式等。

本书理论与实践相结合,采用案例教学并附有大量习题便于读者学习掌握。本书既可作为高等学校相关专业的教材或教学参考书,也可作为软件设计人员的自学用书。

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

软件工程/王庆育编著. —北京:清华大学出版社,2004.7

ISBN 7-302-08763-6

I. 软… II. 王… III. 软件工程 IV. TP311.5

中国版本图书馆 CIP 数据核字(2004)第 053658 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客 户 服 务: 010-62776969

责任编辑: 宋 韬

封面设计: 付剑飞

印 装 者: 北京市清华园胶印厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印 张: 31.5 字 数: 716 千字

版 次: 2004 年 7 月第 1 版 2004 年 7 月第 1 次印刷

书 号: ISBN 7-302-08763-6/TP·6246

印 数: 1~5000

定 价: 47.00 元

前 言

软件工程的飞速发展促使软件工程理论不断地更新。实际上软件工程的发展大大超过了软件工程理论的发展。摩尔定律驱动着硬件同时也在带动着软件产品的更新换代。当人们在争论网络互联协议标准的时候,基于 TCP/IP 协议的 Internet 几乎一夜之间普及全世界,使这种争论瞬时变得毫无价值。当人们苦苦研究着面向对象的数据库理论时,XML 数据库迅速占领了面向对象数据库市场……。这样的例子举不胜举,它们充分说明了软件工程的理论大大滞后于软件工程实践。正如在软件工程中技术标准的制定常常滞后于新技术的使用那样,许多实践经验还来不及总结。因此,软件工程的理论往往是不完备的,它虽然是实践经验的总结,但是这种总结不一定像数学定律那样完全定量、得到充分的证明和普遍适用。它往往采用有限样本证明,可能部分是定性的、当前公认的,就像 TCP/IP 协议是当前大家公认的网络互联协议标准那样。

所以学习软件工程显然不能像学习高等数学那样。首先它是一门工程学。必须投身于软件开发的实践,才能真正领悟软件工程的精髓。软件工程学适合于有一定软件开发经历的学生学习。其次,它是一门年轻而不断发展的学科。在软件工程里没有“公理和定理”,只有“原理和准则”。软件工程学教科书需要不断更新,抛弃事实上已经不采用的软件工程理论、方法和技术,比如经典系统分析设计的一些方法,如 Jackson 方法,所谓“事务设计和变换设计”等。要紧密结合当前软件技术的发展,这正是当前国内教科书迫切要做的,否则就不能对当前软件工程实践起指导作用。

本书共分软件工程基础、软件项目管理、传统的软件工程、面向对象的软件工程、软件工程的最新技术五个部分。内容主要包括:软件工程概论、软件过程、软件工程建模语言、软件项目、团队管理、项目计划、风险分析和管理、项目进度计划与控制、软件质量保证、软件配置管理、需求工程、系统设计、软件测试、软件技术度量、面向对象的分析与设计、编码实现、基于构件的软件工程、客户服务器与 Web 软件工程、设计模式等。

软件工程是一门应用学科,软件教育的教育必须与实践相结合,对软件工程概念的理解只有身临其境才能得到。因此必须采用案例教学和案例习题才能取得好的教学效果。本书尽量用实例来解释概念,用案例来演绎方法和原理,同时提供大量的练习题,以便让读者能够思考并尝试解决实际的问题。

软件教育的教育必须基于问题的解决。软件的实质是人们以计算机编程语言为桥梁,将客观感知世界映射于计算机世界中去,以解决人们在客观感知世界中要解决的问题。而外部世界中存在一些我们可以感知却无法用语言来表达的东西。因此,应该让学生明白不能简单地用正确和错误来判别解决方案,要鼓励学生用自己的理解去解释概念,同时对标准方案进行评价和改进。本书在自测练习题中有意识让学生明白这一点,去选择比较适合的答案。同样本书案例的解决方案也不是惟一的,而是针对具体的需求有多种解决方案。

本书力求在全面系统讲解软件工程理论与实践的同时,引入更多的新知识,从不同的

角度帮助学生掌握软件工程的方法、思路和技术,帮助学生真正理解和灵活应用有关知识,并通过大量练习来增强解决问题的能力。

本书编写的目的就是试图根据最新软件工程实践来完善软件工程方法学体系。它是一本采用案例教学较为详细地讨论面向对象开发思想、方法和技术的教学参考书。

本书是作者根据多年软件工程的实践和教学而编写的。适合作为相关专业的大学生和研究生的教材或教学参考书。在学习本书前最好有软件开发的经历,并且能够使用一门编程语言或工具。这样学习的效果会更好。编者欢迎你的指正、建议和意见,请发E-mail给:qingyuw@sohu.com。感谢我的夫人席与华的支持。另外,史明生、王宝蓉、乔纪纯、王乃康、张一平、孙宝申、韩祥文、陈文章、王世英、刘世平、刘荫庭、王禹秋等参加了本书的讨论、编写和审校,在此一并致谢。

作者

2004年5月

目 录

第 1 部分 软件工程基础

第 1 章 软件工程概论	3
1.1 软件	3
1.1.1 软件的定义	3
1.1.2 软件的特征	4
1.1.3 软件的应用	4
1.2 软件工程	5
1.2.1 软件的发展历史	5
1.2.2 软件过程的概念	7
1.2.3 软件工程的概念	8
1.2.4 软件工程的要素	8
1.2.5 软件工程的目标	9
1.3 软件工程的活动	9
1.3.1 建模.....	10
1.3.2 问题求解.....	10
1.3.3 知识获取.....	11
1.3.4 决策知识.....	11
1.4 自测练习.....	11
1.5 自测练习参考答案.....	13
第 2 章 软件过程	16
2.1 软件过程框架.....	16
2.2 软件的生存周期和瀑布模型.....	17
2.3 原型实现模型.....	19
2.4 演化软件过程模型.....	20
2.4.1 增量模型.....	20
2.4.2 螺旋模型.....	21
2.5 微软解决框架过程模型.....	21
2.5.1 过程定义.....	22
2.5.2 目标驱动.....	22
2.5.3 基于风险管理的开发调度.....	23
2.5.4 按产品版本发布.....	23
2.5.5 支持项目管理.....	23
2.5.6 靠改进特性与固定资源来激发创造力的战略.....	23

2.5.7 同步——稳定开发法·····	26
2.6 基于构件的开发模型·····	26
2.7 极限编程·····	27
2.7.1 目标与活动·····	28
2.7.2 实践方法·····	29
2.7.3 XP——演化模型·····	31
2.7.4 XP 应用的限制·····	31
2.8 软件过程能力成熟度模型·····	32
2.8.1 CMM 简介·····	32
2.8.2 关键过程域·····	36
2.8.3 CMM 与 ISO·····	37
2.8.4 CMM 的应用·····	44
2.9 自测练习·····	47
2.10 自测练习参考答案·····	49
第3章 软件工程建模语言·····	51
3.1 建模的概念·····	51
3.1.1 系统及模型和视图·····	51
3.1.2 概念和现象·····	52
3.2 统一建模语言·····	52
3.2.1 为什么需要 UML·····	52
3.2.2 UML 简介·····	54
3.3 UML 图形符号·····	54
3.3.1 用例图·····	54
3.3.2 类图及对象图和包·····	57
3.3.3 构件图和配置图·····	64
3.3.4 消息·····	65
3.3.5 状态图·····	66
3.3.6 顺序图·····	67
3.3.7 协作图·····	68
3.3.8 活动图·····	69
3.3.9 四种图的运用·····	71
3.4 自测练习·····	71
3.5 自测练习参考答案·····	79

第2部分 软件项目管理

第4章 软件项目·····	87
4.1 项目的历史及发展·····	88
4.1.1 项目的历史·····	88

4.1.2 项目管理的发展·····	88
4.1.3 项目管理的应用·····	89
4.1.4 软件项目管理的特点·····	90
4.2 软件项目的基本概念·····	90
4.2.1 基本概念·····	90
4.2.2 项目管理框架·····	91
4.2.3 人员·····	91
4.2.4 产品·····	92
4.2.5 过程·····	92
4.2.6 项目计划·····	92
4.3 项目生存周期·····	92
4.4 项目拥有者·····	93
4.5 关键管理技能·····	93
4.5.1 明白领导和管理的区别·····	93
4.5.2 交流技巧·····	94
4.5.3 谈判能力·····	94
4.5.4 解决问题的能力·····	94
4.5.5 影响组织·····	94
4.6 项目管理的基本思想和技术·····	94
4.6.1 成本/进度综合控制·····	94
4.6.2 蒙特卡罗模拟技术·····	95
4.6.3 项目进展评价技术·····	95
4.6.4 网络计划技术·····	96
4.6.5 项目管理的可视化技术·····	96
4.7 自测练习·····	97
4.8 自测练习参考答案·····	98
第5章 团队管理 ·····	100
5.1 团队模型·····	100
5.1.1 组织原则·····	100
5.1.2 微软解决方案框架团队模型·····	101
5.2 自测练习·····	104
5.3 自测练习参考答案·····	105
第6章 项目计划 ·····	106
6.1 项目计划简介·····	106
6.1.1 影响项目估算的因素·····	106
6.1.2 软件范围的确定·····	107
6.1.3 项目所需资源·····	107
6.2 项目估算·····	108
6.2.1 项目估算的方法·····	108

6.2.2 软件规模估算	109
6.2.3 经验估算模型	110
6.3 项目计划的制定与提交	112
6.4 自测练习	113
6.5 自测练习参考答案	114
第7章 风险分析和管理	116
7.1 软件风险	116
7.1.1 风险的概念	116
7.1.2 风险策略	116
7.1.3 软件风险的类别	117
7.2 风险识别	117
7.3 风险预测和评估	118
7.3.1 建立风险表	118
7.3.2 评估风险影响	120
7.3.3 风险评估	121
7.4 风险的缓解和监控与管理	121
7.4.1 建立有效处理风险策略	121
7.4.2 RMMM 计划	123
7.4.3 交流风险	124
7.5 自测练习	124
7.6 自测练习参考答案	124
第8章 项目进度计划与控制	127
8.1 项目进度计划	127
8.1.1 进度计划	127
8.1.2 自顶向下与自底向上的计划	127
8.1.3 基本步骤	128
8.2 项目跟踪与控制	129
8.2.1 进度安排	129
8.2.2 关键路径	130
8.2.3 跟踪进度	130
8.2.4 百分比值分析法	131
8.2.5 赢得值法	132
8.3 Microsoft Project 2000	136
8.3.1 简介	136
8.3.2 功能	138
8.4 自测练习	140
8.5 自测练习参考答案	142
第9章 软件质量保证	144
9.1 软件质量	144

9.1.1 软件质量的定义	144
9.1.2 软件质量控制的含义	145
9.1.3 软件质量保证的含义	145
9.1.4 软件质量成本的含义	145
9.2 软件质量保证	146
9.2.1 软件质量保证的历史	146
9.2.2 SQA 活动	147
9.2.3 统计软件质量保证	147
9.2.4 开发正确的产品	148
9.3 软件评审	148
9.3.1 阶段评审	149
9.3.2 正式评审	149
9.3.3 项目完成后评审	149
9.4 软件复杂性与可靠性	150
9.4.1 软件复杂性	150
9.4.2 软件可靠性	150
9.4.3 软件的容错技术	151
9.5 软件质量度量模型	152
9.5.1 不同的人对软件质量关心的重点不同	152
9.5.2 软件质量特性	152
9.6 自测练习	153
9.7 自测练习参考答案	155
第 10 章 软件配置管理	157
10.1 软件配置项及基线	157
10.1.1 软件配置项	157
10.1.2 基线	158
10.2 配置管理过程	159
10.2.1 配置对象标识	159
10.2.2 版本控制	160
10.2.3 变更控制	161
10.3 一个配置管理的工具——Visual Source Safe	162
10.4 自测练习	163
10.5 自测练习参考答案	164

第 3 部分 传统的软件工程

第 11 章 需求工程	169
11.1 基本概念	169
11.1.1 业务过程工程	170

11.1.2 产品工程	170
11.1.3 需求工程	170
11.2 需求获取	171
11.2.1 需求获取方式	172
11.2.2 提高获取的效率	173
11.2.3 原型与场景	176
11.3 需求分析建模	179
11.3.1 为什么需要建模	179
11.3.2 建模的方法	179
11.3.3 结构化分析	180
11.3.4 数据建模	180
11.3.5 功能建模	187
11.3.6 控制模型	192
11.3.7 数据字典	193
11.3.8 IDEF 方法	199
11.4 软件需求规约及评审	203
11.4.1 需求规约	203
11.4.2 规约的评审	203
11.5 自测练习	204
11.6 自测练习参考答案	208
第 12 章 系统设计	211
12.1 系统设计的概念	211
12.1.1 软件设计模型	211
12.1.2 设计目标和原则	212
12.1.3 设计基本方法	214
12.1.4 设计文档	218
12.2 数据设计	219
12.2.1 数据三模式的概念	219
12.2.2 数据库设计的原则	220
12.2.3 数据仓库	223
12.3 体系结构设计	224
12.3.1 体系结构是通用的	224
12.3.2 主机/终端结构	225
12.3.3 客户机/服务器结构	226
12.3.4 多层分布式结构	228
12.3.5 体系结构的选择	229
12.3.6 系统模块结构	231
12.4 人机界面设计	235
12.4.1 界面设计是综合艺术	235

12.4.2	界面设计的黄金规则	235
12.4.3	界面设计	237
12.4.4	界面设计的问题	240
12.4.5	界面设计原型技巧	241
12.5	过程设计	242
12.5.1	结构化程序设计	242
12.5.2	过程设计模板	243
12.6	自测练习	245
12.7	自测练习参考答案	249
第 13 章	软件测试	253
13.1	测试目标和原则	253
13.1.1	测试目标	253
13.1.2	测试规则	254
13.1.3	可测试性	255
13.1.4	测试方法	257
13.2	测试用例设计	257
13.2.1	白盒法	257
13.2.2	黑盒法	263
13.2.3	针对专门环境的测试	264
13.3	测试计划	265
13.4	软件测试过程	266
13.4.1	软件测试步骤	266
13.4.2	单元测试	268
13.4.3	集成测试	269
13.4.4	确认测试	272
13.4.5	一些常见的可用性问题	274
13.5	测试和调试	275
13.5.1	调试和测试的区别	275
13.5.2	调试的困难	276
13.5.3	调试的方法	276
13.5.4	测试准备	277
13.6	自测练习	277
13.7	自测练习参考答案	280
第 14 章	软件技术度量	284
14.1	软件技术度量的框架	284
14.1.1	度量准则	284
14.1.2	有效软件度量的属性	285
14.2	需求分析度量	285
14.2.1	基于功能的分析模型测量	285

14.2.2 分析规约质量的度量	287
14.3 设计模型度量	287
14.3.1 体系结构度量	287
14.3.2 过程设计度量	288
14.3.3 人机界面的度量	289
14.4 源代码度量	289
14.5 对测试的度量	290
14.6 对维护的度量	290
14.7 自测练习	290
14.8 自测练习参考答案	292

第 4 部分 面向对象的软件工程

第 15 章 面向对象的概念	295
15.1 面向对象的概念化	295
15.1.1 项目概念化	295
15.1.2 面向对象的范型	297
15.1.3 面向对象的软件工程	298
15.1.4 面向对象的管理	298
15.2 类与对象	299
15.2.1 对象的识别	299
15.2.2 对象和类	301
15.3 面向对象的要素	301
15.3.1 对象的关系	301
15.3.2 面向对象的特性	303
15.4 面向对象的建模	304
15.4.1 目标	305
15.4.2 面向对象的模型	305
15.4.3 面向对象的开发方法	307
15.5 自测练习	308
15.6 自测练习参考答案	310
第 16 章 面向对象的分析	311
16.1 面向对象的分析与设计	311
16.2 案例分析——图书管理系统	312
16.2.1 需求规格说明	312
16.2.2 需求规格说明的细化	314
16.2.3 需求规格说明的验证	315
16.3 需求分析	316
16.3.1 需求分析的过程	316

16.3.2 用 UML 建模	322
16.4 以用例为中心的需求分析方法	326
16.5 自测练习	327
16.6 自测练习参考答案	329
第 17 章 面向对象的设计	331
17.1 面向对象的设计概述	331
17.1.1 面向对象的设计目标	331
17.1.2 类设计与产品设计	331
17.2 对象的存储	332
17.2.1 对象存储方法	332
17.2.2 对象序列化	333
17.2.3 对象属性的永久保留	334
17.2.4 对象存储的评价	335
17.2.5 图书管理系统的对象存储	335
17.3 进程体系结构	337
17.3.1 多节点建模与进程间的通信	337
17.3.2 实例	339
17.4 用户界面	341
17.4.1 用户界面设计	341
17.4.2 用户界面的交互样式	341
17.4.3 图书管理系统的界面	344
17.5 类设计	345
17.5.1 类架构	346
17.5.2 系统分解	348
17.5.3 类图的附加符号	348
17.5.4 交互图	349
17.6 类设计的目标及其验证	354
17.6.1 代码重用	354
17.6.2 良好设计的类与方法	355
17.6.3 数据的完整性	355
17.6.4 类设计的验证	356
17.7 方法设计	357
17.7.1 确定方法	357
17.7.2 创建优质的方法	358
17.8 自测练习	358
17.9 自测练习参考答案	360
第 18 章 编码实现	363
18.1 程序设计的发展	363
18.1.1 程序设计的方法	363

18.1.2	软件开发环境的发展	364
18.1.3	软件开发模式的发展	364
18.2	开发策略	365
18.2.1	自顶向下的开发与自底向上的开发	365
18.2.2	自底向上和自顶向下的结合	367
18.2.3	线程方法	368
18.3	编码计划	368
18.4	编程风格	370
18.4.1	简单性	371
18.4.2	注释和内部文档	373
18.5	编码标准	376
18.5.1	一般规则	376
18.5.2	模块化规则	377
18.5.3	声明	377
18.5.4	可执行的命令	377
18.6	超类设计和超类对象的实现	378
18.6.1	识别超类	378
18.6.2	面向对象方法的超类实现	379
18.7	自测练习	379
18.8	自测练习参考答案	381
第 19 章	面向对象的测试	382
19.1	基本概念	382
19.1.1	测试的定义	382
19.1.2	面向对象测试	382
19.1.3	错误及故障和失效	383
19.2	测试计划	384
19.2.1	计划内容	384
19.2.2	预期结果	386
19.2.3	测试用例	386
19.3	面向对象的测试过程	387
19.3.1	面向对象的单元测试	387
19.3.2	面向对象的集成测试	389
19.3.3	面向对象的系统测试	391
19.3.4	面向对象的测试步骤	392
19.3.5	软件内部测试的自动工具	392
19.4	面向对象系统与面向过程系统的测试	393
19.4.1	面向对象系统测试的复杂性	393
19.4.2	有关测试的一些问题	395
19.5	案例分析——测试图书馆管理系统	398

19.5.1 LMS 的测试计划	399
19.5.2 LMS 的单元测试	400
19.5.3 LMS 的测试用例	401
19.6 配置管理的测试	402
19.7 自测练习	403
19.8 自测练习参考答案	404

第 5 部分 软件工程的最新技术

第 20 章 基于构件的软件工程	409
20.1 构件	409
20.1.1 构件技术	409
20.1.2 构件属性	410
20.2 基于构件的系统开发	411
20.2.1 系统结构分层和逻辑与数据分离	411
20.2.2 选择和评价已有的构件	412
20.2.3 构件的使用	412
20.2.4 实现构件技术必须具备的条件	413
20.3 构件的开发	414
20.3.1 构件的设计	414
20.3.2 领域工程	415
20.4 构件和中间件	418
20.4.1 中间件是构件存在的基础	418
20.4.2 构件思想对中间件的作用	420
20.4.3 中间件的分类	421
20.4.4 基于中间件的开发系统模型	424
20.4.5 中间件面临的一些问题	427
20.5 软件构件的标准	427
20.5.1 OMG/CORBA	427
20.5.2 EJB 和 J2EE	428
20.5.3 Microsoft DCOM 及 Windows DNA	429
20.6 自测练习	431
20.7 自测练习参考答案	432
第 21 章 客户服务器与 Web 软件工程	434
21.1 计算机系统体系结构	434
21.1.1 计算机系统结构	434
21.1.2 C/S 系统结构	436
21.2 C/S 系统	437
21.2.1 C/S 系统构成	438

21.2.2 分布应用子系统的指导原则·····	441
21.2.3 分布应用举例·····	441
21.3 C/S 系统开发·····	442
21.3.1 C/S 系统设计·····	442
21.3.2 C/S 系统结构设计·····	443
21.3.3 C/S 系统数据库设计·····	445
21.3.4 提高 C/S 系统性能的方法·····	445
21.4 基于 Web 系统·····	449
21.4.1 C/S 与 B/S·····	449
21.4.2 B/S 工程·····	452
21.4.3 B/S 系统设计·····	454
21.4.4 B/S 系统测试·····	457
21.4.5 B/S 系统管理·····	458
21.5 自测练习·····	460
21.6 自测练习参考答案·····	461
第 22 章 设计模式 ·····	463
22.1 设计模式简介·····	463
22.1.1 设计模式产生于建筑学和人类学·····	463
22.1.2 再工程与模式化运动·····	464
22.1.3 软件模式与模式语言·····	465
22.1.4 面向模式的软件再工程·····	465
22.1.5 软件再工程首先面对反面模式·····	467
22.1.6 从再工程过程中抽象软件模式·····	467
22.2 基本设计模式·····	468
22.2.1 创建型模式·····	468
22.2.2 结构型模式·····	471
22.2.3 行为模式·····	474
22.2.4 学习设计模式的意义·····	479
22.3 应用设计模式·····	479
22.3.1 包装程序设计模式·····	479
22.3.2 桥接设计模式·····	480
22.3.3 迭代程序设计模式·····	483
22.3.4 单实例设计模式·····	484
22.4 自测练习·····	485
22.5 自测练习参考答案·····	485
参考文献·····	487