

21世纪高等学校规划教材 | 软件工程



软件质量保证 及测试基础

李晓红 主编

唐晓君 王海文 副主编



清华大学出版社

21世纪高等学校规划教材 | 软件工程

软件质量保证 及测试基础

李晓红 主编

唐晓君 王海文 副主编

清华大学出版社

北京

内 容 简 介

本书内容系统、全面,叙述简练,实践性和实用性强。全书共分 11 章,主要包括:软件质量、软件质量保证概要、软件质量保证过程、软件测试基础、软件测试过程、静态测试、动态测试、各级别的测试、面向对象软件测试、软件缺陷管理和测试评估和软件测试自动化。

本书可作为高等院校教学研究型或教学应用型的计算机专业、软件工程专业的软件测试课程教材使用,还可供软件测试从业人员理论和实践参考。因编写参考了 ISTQB 的软件测试初级认证大纲,还可供想获取 ISTQB 基础级认证的人员参考使用。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

软件质量保证及测试基础/李晓红主编.--北京:清华大学出版社,2015

21 世纪高等学校规划教材·软件工程

ISBN 978-7-302-39616-1

I. ①软… II. ①李… III. ①软件质量—质量管理—研究 ②软件—测试—研究 IV. ①TP311.5

中国版本图书馆 CIP 数据核字(2015)第 049503 号

责任编辑:黄 芝 李 晔

封面设计:傅瑞学

责任校对:时翠兰

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 刷 者:三河市君旺印务有限公司

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:16.5 字 数:402 千字

版 次:2015 年 7 月第 1 版 印 次:2015 年 7 月第 1 次印刷

印 数:1~2000

定 价:34.00 元

产品编号:058318-01

出版说明

随着我国改革开放的进一步深化,高等教育也得到了快速发展,各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程(简称‘质量工程’)”,通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建议,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作,提高教学质量的若干意见》精神,紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”,在有关专家、教授的倡议和有关部门的大力支持下,我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”),旨在配合教育部制定精品课程教材的出版规划,讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师,其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求,“编委会”一致认为,精品课程的建设工作从开始就要坚持高标准、严要求,处于一个比较高的起点上;精品课程教材应该能够反映各高校教学改革与课程建设的需要,要有特色风格、有创新性(新体系、新内容、新手段、新思路,教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学科体系有实质性的改革和发展,顺应并符合21世纪教学发展的规律,代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版

社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。推出的特色精品教材包括:

(1) 21 世纪高等学校规划教材·计算机应用——高等学校各类专业,特别是非计算机专业的计算机应用类教材。

(2) 21 世纪高等学校规划教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 21 世纪高等学校规划教材·电子信息——高等学校电子信息相关专业的教材。

(4) 21 世纪高等学校规划教材·软件工程——高等学校软件工程相关专业的教材。

(5) 21 世纪高等学校规划教材·信息管理与信息系统。

(6) 21 世纪高等学校规划教材·财经管理与应用。

(7) 21 世纪高等学校规划教材·电子商务。

(8) 21 世纪高等学校规划教材·物联网。

清华大学出版社经过三十多年的努力,在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌,为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格,这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

联系人:魏江江

E-mail: weijj@tup. tsinghua. edu. cn

前言

计算机技术的发展使得计算机软件规模和软件复杂度不断提升,软件企业对软件的质量也越来越重视,软件质量的可靠性变得尤为重要。软件测试作为软件质量保证的重要手段,也得到了业界的普遍认可和实施,软件测试行业得以迅速发展。

随着软件测试行业的发展,软件质量保证和测试的理论、知识、技术和工具都在不断更新。同时,软件测试相关资料和书籍大量涌现,读者可以找到很多书籍,有关于特定应用领域的,如 Web 应用测试、嵌入式应用测试;有关于测试基础知识的,如软件测试基础、软件测试原理等;有关于测试工具实践的;还有关于测试管理的等。

本书致力于为软件质量管理人员及测试人员提供一个坚实的理论和实践基础,注重基础知识的理解同时强调实践的培养。书中重点介绍基本的质量保证知识,质量保证过程,软件测试基本技术、方法以及自动化测试工具的使用,能够为即将选择软件测试作为职业发展方向的读者提供一个起步学习参考。与同类的教材相比,本书具有如下特点:

(1) 内容简洁、完整、系统性强,便于读者全面掌握软件测试基础知识。

(2) 与 ISTQB(国际软件测试认证委员会)软件测试初级认证大纲要求一致,便于读者所学知识与国际接轨。

(3) 实践性强,以实例的形式介绍多种测试工具的使用。

(4) 实用性强,提供软件测试文档模板,为测试实践工作提供参考和帮助。

全书共分为 11 章,内容安排如下。

第 1 章 软件质量。讲述软件质量的概念和软件质量模型。

第 2 章 软件质量保证概要。讲述软件质量保证概念,软件质量保证目标和任务以及软件质量保证活动。

第 3 章 软件质量保证过程。讲述计划阶段、需求分析阶段、设计阶段、编码阶段、测试阶段、系统交付和安装阶段的质量保证内容。

第 4 章 软件测试基础。讲述软件测试的概念,软件测试的类型以及软件测试的原则。

第 5 章 软件测试过程。讲述常见的软件测试过程模型以及基本的软件测试过程。

第 6 章 静态测试。讲述评审和静态结构分析。

第 7 章 动态测试——测试用例设计技术。讲述白盒测试用例设计技术和黑盒测试用例设计技术。

第 8 章 各级别测试。讲述单元测试、集成测试、确认测试、系统测试、验收测试和回归测试内容。

第 9 章 面向对象软件测试。讲述面向对象测试基础、面向对象测试模型及面向对象单元测试和集成测试。

第 10 章 软件缺陷管理和测试评估。讲述软件缺陷概述、软件缺陷相关信息、软件缺陷的描述和软件测试评估。

第 11 章 软件测试自动化。讲述自动化软件测试的定义、自动化测试技术和自动化测试工具的使用。

附录内容列出了软件测试相关文档模板。

本书作者一直在高校从事软件质量管理及测试的教学和实践工作,具有丰富的理论知识和实践经验。全书共 11 章,由大连工业大学李晓红、唐晓君和王海文编写。其中,李晓红编写第 1、4、5、7、8、10、11 章,唐晓君编写第 2、3 章,王海文编写第 6、9 章和附录,全书由李晓红负责统稿定稿。

本书在编写过程中除书中列出参考文献外,还部分参考了 51testing、ISTQB 中国官网及互联网上其他文档资料,在此对网络中的各位知识分享者表示由衷的感谢。

希望通过对本书内容的学习,读者能够掌握软件质量保证的基础知识、软件测试的理论知识以及自动化软件测试工具的使用,为有志投身软件测试行业的专业人士和学生提供有益帮助。

软件测试发展迅速,加之作者水平和时间有限,书中难免有错误和不妥之处,恳请读者批评指正。

编 者

2015 年 3 月

目 录

第 1 章 软件质量	1
1.1 软件质量的概念	1
1.2 软件质量模型	2
1.2.1 McCall 质量模型	2
1.2.2 Boehm 质量模型	4
1.2.3 ISO/IEC 9126 质量模型	5
1.2.4 Perry 模型	6
本章小结	7
本章习题	7
第 2 章 软件质量保证概要	8
2.1 软件质量保证	8
2.1.1 软件质量保证定义	9
2.1.2 软件质量保证内容	9
2.1.3 软件质量保证要素	10
2.1.4 软件质量保证计划	10
2.1.5 软件质量保证素质	11
2.1.6 软件质量保证的组织结构	11
2.1.7 软件质量保证的岗位职责	12
2.2 软件质量控制	12
2.2.1 软件质量控制的基本概念	12
2.2.2 软件质量控制的基本方法	14
2.3 软件质量保证目标和任务	15
2.3.1 软件质量保证的目标	15
2.3.2 软件质量保证的任务	17
2.4 软件质量保证活动	19
2.5 全面软件质量管理	20
2.5.1 全面软件质量管理定义	20
2.5.2 全面软件质量管理四个要素	21
2.5.3 全面软件质量管理三个原则	21
2.5.4 全面软件质量管理方法	22
2.5.5 全面软件质量控制模型	25

2.5.6 全面软件质量控制技术	27
2.6 软件质量管理体系结构	30
2.6.1 CMM/CMMI	31
2.6.2 ISO 9000	34
本章小结	35
本章习题	35
第3章 软件质量保证过程	36
3.1 计划阶段	36
3.2 需求分析阶段	38
3.3 设计阶段	40
3.4 编码阶段	41
3.5 测试阶段	45
3.6 系统交付和安装阶段	47
本章小结	48
本章习题	48
第4章 软件测试基础	49
4.1 软件测试的概念	49
4.1.1 软件缺陷	49
4.1.2 验证和确认	53
4.1.3 软件测试的定义	54
4.1.4 软件测试的目的	55
4.1.5 测试用例	56
4.2 软件测试的分类	57
4.2.1 按技术分类	57
4.2.2 按测试方式分类	57
4.2.3 按测试阶段分类	58
4.2.4 按测试内容分类	59
4.3 软件测试的误区	60
4.4 软件测试的原则	61
本章小结	62
本章习题	62
第5章 软件测试过程	63
5.1 常见测试过程模型	63
5.1.1 V模型	63
5.1.2 W模型	64
5.1.3 H模型	65

5.1.4	X 模型	65
5.1.5	前置模型	66
5.1.6	测试模型总结	69
5.2	基本测试过程	70
5.2.1	测试计划和控制	71
5.2.2	测试分析和设计	75
5.2.3	测试实现和执行	75
5.2.4	测试评估和报告	77
	本章小结	77
	本章习题	78
第 6 章	静态测试	79
6.1	静态测试概述	79
6.1.1	为什么要进行静态测试	80
6.1.2	静态测试的重要性	81
6.2	评审	82
6.2.1	评审成功的因素及基本术语	83
6.2.2	评审的分类	83
6.2.3	非正式评审	84
6.2.4	正式评审及其基本过程	85
6.3	技术评审	86
6.3.1	技术评审的目的和内容	86
6.3.2	技术评审团队	86
6.3.3	技术评审会议	87
6.4	审查	87
6.4.1	审查的目的和内容	87
6.4.2	审查团队	88
6.4.3	审查的前提条件	88
6.4.4	审查会议过程	89
6.4.5	审查输出	91
6.4.6	数据收集	92
6.4.7	审查的注意事项	92
6.5	代码审查	93
6.5.1	代码审查的测试内容及组成	93
6.5.2	代码审查的步骤	93
6.5.3	代码审查单	93
6.5.4	阅读的方法	94
6.6	走查	95
6.6.1	走查的目的和内容	95

6.6.2	走查团队	95
6.6.3	走查会议	96
6.6.4	走查与审查	96
6.7	静态分析	97
6.7.1	数据流分析	97
6.7.2	控制流分析	99
	本章小结	103
	本章习题	104
第7章	动态测试——测试用例设计技术	105
7.1	白盒测试用例设计技术	105
7.1.1	逻辑覆盖	106
7.1.2	逻辑覆盖准则	112
7.1.3	路径测试	114
7.1.4	其他白盒测试技术	120
7.1.5	白盒测试技术讨论	122
7.2	黑盒测试用例设计技术	123
7.2.1	等价类划分	124
7.2.2	边界值分析	132
7.2.3	决策表	135
7.2.4	因果图	139
7.2.5	状态转换测试	142
7.2.6	其他黑盒测试技术	144
7.2.7	黑盒测试技术讨论	145
	本章小结	146
	本章习题	146
第8章	各级别的测试	149
8.1	单元测试	149
8.1.1	单元测试的概念	149
8.1.2	单元测试的目的	150
8.1.3	单元测试的内容	150
8.1.4	单元测试的原则	152
8.1.5	单元测试的策略	153
8.1.6	单元测试停止的条件	153
8.2	集成测试	154
8.2.1	集成测试的概念	154
8.2.2	集成测试的必要性	154
8.2.3	集成测试的内容	155

8.2.4	集成测试的原则	155
8.2.5	集成测试策略	156
8.2.6	集成测试的停止条件	159
8.2.7	集成测试与单元测试的区别	160
8.3	确认测试	160
8.4	系统测试	160
8.4.1	系统测试的定义	160
8.4.2	系统测试的类型	161
8.4.3	系统测试的停止条件	163
8.4.4	系统测试与单元测试、集成测试的区别	163
8.5	验收测试	163
8.5.1	验收测试的概念	163
8.5.2	Alpha 测试	164
8.5.3	Beta 测试	164
8.6	回归测试	165
8.6.1	回归测试前提	165
8.6.2	回归测试基本过程	166
8.6.3	回归测试用例的选择	167
8.6.4	回归测试与一般测试的比较	167
	本章小结	168
	本章习题	168
第 9 章	面向对象的软件测试	169
9.1	面向对象测试基础	169
9.1.1	面向对象测试层次	169
9.1.2	面向对象测试顺序	170
9.1.3	面向对象测试用例	170
9.2	面向对象测试模型	170
9.2.1	面向对象分析的测试	171
9.2.2	面向对象设计的测试	173
9.2.3	面向对象编程的测试	174
9.3	面向对象的单元测试	175
9.3.1	单元的定义	175
9.3.2	以方法为单元	177
9.3.3	以类为单元	178
9.3.4	面向对象单元测试的特殊性	179
9.4	面向对象的集成测试	180
9.4.1	面向对象集成测试基础	180
9.4.2	基于 UML 集成测试参考模型	182

9.5 面向对象的系统测试	183
本章小结	184
本章习题	184
第10章 软件缺陷管理和测试评估	185
10.1 软件缺陷概述	185
10.1.1 软件缺陷的类型	185
10.1.2 软件缺陷的等级及优先级	186
10.1.3 软件缺陷生命周期	187
10.2 软件缺陷相关信息	188
10.2.1 完整软件缺陷信息	188
10.2.2 软件缺陷记录	190
10.3 软件缺陷跟踪和分析	191
10.3.1 软件缺陷的处理	192
10.3.2 软件缺陷的分析	192
10.3.3 软件缺陷的跟踪	194
10.4 软件测试评估	195
10.4.1 软件测试评估概述	195
10.4.2 软件测试评估分类	196
本章小结	197
本章习题	197
第11章 软件测试自动化	198
11.1 自动化测试的定义	198
11.1.1 概念	198
11.1.2 自动化测试的优点	198
11.1.3 自动化测试的局限性	200
11.1.4 自动化测试的适用范围	200
11.2 自动化测试原理	201
11.2.1 代码分析	201
11.2.2 录制回放技术	202
11.2.3 脚本技术	202
11.2.4 虚拟用户技术	204
11.3 软件测试工具	204
11.3.1 软件测试工具类型	204
11.3.2 单元测试工具实例——JUnit	206
11.3.3 性能测试工具实例——LoadRunner	215
11.3.4 功能测试工具实例——QTP	225
本章小结	234

本章习题	234
附录A 软件测试相关文档模板	235
附录 A.1 代码审查单	238
附录 A.2 代码走查报告	240
附录 A.3 软件测试计划模板	241
附录 A.4 软件测试用例模板	244
附录 A.5 软件缺陷模板	245
附录 A.6 测试报告模板	246
参考文献	249

第1章

软件质量

本章学习目标：

- 掌握软件质量的概念。
- 掌握四种软件质量模型。

1.1 软件质量的概念

随着计算机应用越来越广泛、深入,软件也越来越复杂。软件质量问题可能导致经济损失甚至灾难性的后果,质量问题还会增加开发和维护软件产品的成本,使软件产品失去市场。本节重点讨论软件质量问题。

1. 质量

质量(Quality),决定产品存在的价值,是企业的生命。我们先来看一下一些权威机构对质量做出的解释。

在《辞海》中,对质量的解释为“产品或工作的优劣程度”。

国际标准化组织(ISO)给出的质量定义是:反映实体(可单独描述和研究的事物,如活动、过程、产品、组织、体系或人,以及它们各项的任何组合)满足明确和隐含需要能力的特性组合。

IEEE在“Standard Glossary of Software Engineering Terminology”中给出的质量定义是被普遍接受的概念,即质量是系统、部件或过程满足明确需求。

世界著名的质量管理专家Juran博士把“质量”定义为:产品在使用时能成功地满足用户目的的程度。

从众多的定义中,我们可以看到,质量是一个复杂、多层面的概念,如果站在不同的观点上从不同的层面或角度对质量就有着不同的理解。

- 从用户出发的质量观:质量是产品满足使用目的的程度。
- 以产品为中心的质量观:质量是软件的内在特征。
- 生产者的质量观:质量是产品性能符合规格要求的程度。
- 以价值为基准的质量观:质量依赖于顾客愿意付给产品报酬的数量。

因此,有一个很重要的概念和质量息息相关,这个概念就是“用户”,不同的用户对待质量的看法是不同的,质量和用户两者相对存在。

用户的定义至少存在两个范畴——内部的和外部的。

外部用户是产品的实际使用者或服务的对象,是传统意义上大家所认可的用户。

内部用户是更为广泛意义上的用户,用户可以被理解为下一道工序的接受者。在软件生产的环节中有关的人员都可被定义为这一类型的用户,软件的设计者是需求分析人员的用户,编程人员是设计者的用户,软件测试是编程人员的用户。

从质量的定义和不同的理解中可以看到,质量具有“满足用户需求的特征”这个核心含义。质量似乎不是客观的,因为没有什么科学仪器可以直接测出质量来;质量似乎也不是主观的,因为它不仅仅存在于人们的脑海中。其实,质量应该是客观存在的,但是测量它的方法却是主观的。

2. 软件质量

软件质量有多种定义。

《计算机软件质量保证计划规范》(GB/T12504—90)对软件质量的定义是,软件产品中能满足给定需求的各种特性的总和。这些特性称为质量特性,它包括功能性、可靠性、易用性、时间经济性、资源经济性、可维护性和可移植性等。

ANSI/IEEE std. 729 对软件质量的定义是,软件产品中能满足规定的和隐含的与需求能力有关的全部特征和特性,包括:

- 软件产品质量满足用户要求的程度。
- 软件各种属性的组合程度。
- 用户对软件产品的综合反映程度。
- 软件在使用过程中满足用户要求的程度。

从以上定义可以看出,软件质量是软件产品满足使用要求的程度。其中“程度”是由软件的特性或特征集组成的。

1.2 软件质量模型

从软件质量的定义得知软件质量是通过一定的属性集来表示其满足使用要求的程度,那么这些属性集包含的内容就显得很重要了。计算机界对软件质量的属性进行了较多的研究,得到了一些有效的质量模型,包括 McCall 质量模型、Boehm 质量模型、ISO/IEC 9126 质量模型和 Perry 模型。

1.2.1 McCall 质量模型

早期的 McCall 软件质量模型是 1977 年 McCall 和他的同事建立的,他们在这个模型中提出了影响质量因素的分类。软件质量因素按一定方法分成几组,每组反映软件质量的一个方面,称为质量要素。构成一个质量要素的诸因素是对该要素的衡量标准。每个衡量标准由一系列具体的度量构成。模型结构如图 1.1 所示。

如图 1.2 所示为 McCall 模型的示意图,11 个质量因素集中在软件产品的 3 个重要方面:产品操作、产品修改、产品改型。

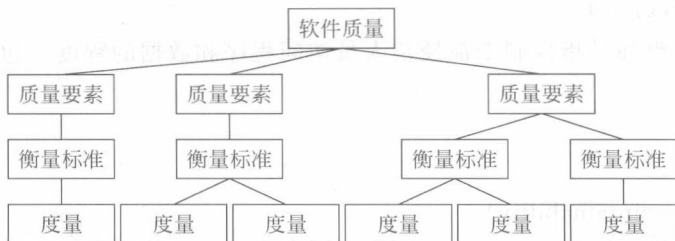


图 1.1 McCall 质量模型结构图

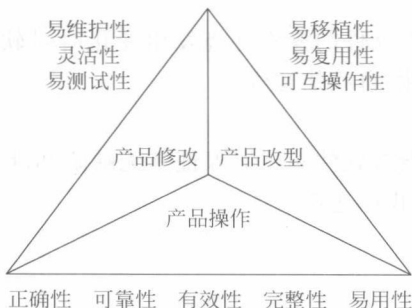


图 1.2 McCall 模型示意图

1. 产品操作

1) 正确性(correctness)

产品操作的正确性是指程序满足规约(specification)及完成用户目标的程度,主要包括易追溯性、一致性和完备性。

- 易追溯性是指在特定的软件开发与操作环境中,能够从软件的需求中找出其相应的实现能力与性质。
- 完备性是指软件实现了其全部所需功能的性质。
- 一致性是指在软件设计与实现中使用统一的技术和术语的性质。

2) 有效性(efficiency)

产品操作的有效性是指软件对计算机资源的使用效率,包括:

- 运行效率(execution efficiency)——软件使用最少的处理时间的性质。
- 存储效率(storage efficiency)——软件在操作中对存储空间的需求最小的性质。

3) 易使用性(usability)

产品操作的易使用性是指其学习使用的难易程度。包括操作软件、为软件准备输入数据、解释软件输出结果。

4) 可靠性(reliability)

产品操作的可靠性是指能够防止因为概念、设计和结构等方面的不完善造成的软件系统失效,具有挽回因为操作不当造成的软件系统失效的能力。包括容错性、一致性、准确性和简洁性。