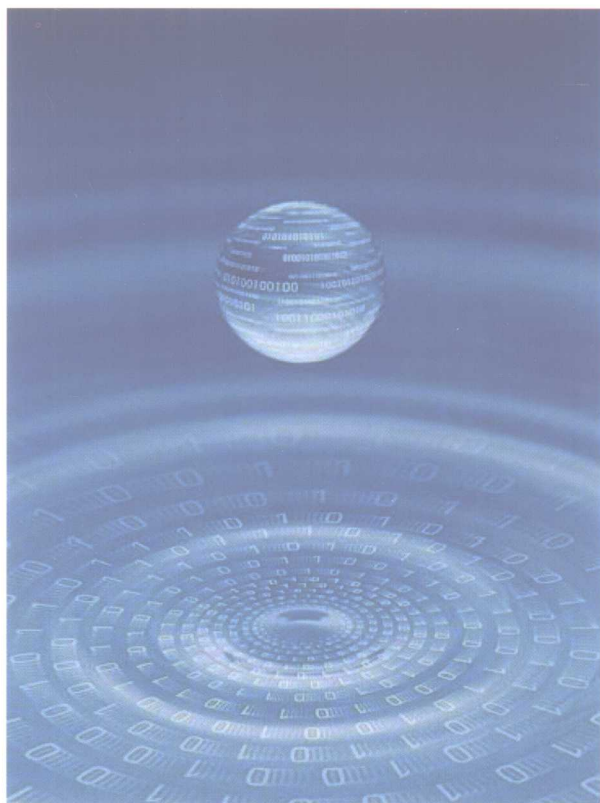


高等学校计算机应用规划教材

中国工程院院士何新贵作序推荐

软件测试大学教程

- ◆ 软件与软件危机
- ◆ 软件缺陷与缺陷管理
- ◆ 软件测试基本概念
- ◆ 生命周期软件测试方法
- ◆ 基于软件配置项的测试种类
- ◆ 软件静态测试
- ◆ 软件动态测试
- ◆ 软件测试自动化
- ◆ 软件可靠性测试
- ◆ 软件本地化测试
- ◆ 面向对象软件测试
- ◆ C/S应用软件测试
- ◆ 基于B/S架构的Web应用软件测试
- ◆ 游戏软件测试
- ◆ 嵌入式软件测试



蔡建平 编著



清华大学出版社

高等学校计算机应用规划教材

软件测试大学教程

蔡建平 编著

清华大学出版社

北 京

内 容 简 介

本书是作者多年从事软件测试研究和教学经验的总结。全书共 16 章,分为 3 部分:第 1 部分(第 1~6 章)是软件测试基础,涉及软件测试的一些基本概念和基础知识,如软件与软件危机、软件缺陷与缺陷管理、软件测试基本概念、生命周期软件测试方法、软件测试过程和基于软件配置项的测试种类;第 2 部分(第 7~12 章)是本书的重点,讲述软件测试方法,包括软件静态测试、动态测试、软件测试自动化、软件可靠性测试、软件本地化测试以及面向对象软件测试;第 3 部分(第 13~16 章)是典型应用的软件测试,重点介绍 C/S 应用软件测试、B/S 架构的 Web 应用软件测试(包括信息系统的测试)、游戏软件测试以及嵌入式软件测试。

另外,本书几乎在各个章节对支撑该章节软件测试方法和技术应用的测试工具进行了详细介绍,特别是对开源软件测试工具进行了较全面的介绍,这些工具对于很好地支持高校软件测试课程实践教学是非常有现实意义的。

本书取材新颖、内容翔实、通俗易懂、技术实用、覆盖面广、指导性强,既可作为软件测试相关课程的研究生与高年级本科生的教材,同时也可作为软件测试培训和软件测试人员的自学书籍。

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

软件测试大学教程/蔡建平 编著. —北京:清华大学出版社, 2009.9

ISBN 978-7-302-20655-2

I. 软… II. 蔡… III. 软件—测试—高等学校—教材 IV. TP311.5

中国版本图书馆 CIP 数据核字(2009)第 124578 号

责任编辑:王 军 鲍 芳

装帧设计:康 博

责任校对:胡雁翎

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

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

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

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185×260 印 张:24.75 字 数:571 千字

版 次:2009 年 9 月第 1 版 印 次:2009 年 9 月第 1 次印刷

印 数:1~4000

定 价:35.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:033841-01

序

作为软件工程的重要分支，软件测试在自提出至今的几十年中，已经发生了很大的变化。它从传统软件工程瀑布模型中的一个测试阶段发展为覆盖包括系统分析与设计、软件需求分析、设计、编码、测试、系统确认验收以及维护的全生命周期测试；从过去朴素的测试概念发展到包括静态分析、质量度量与评价在内的评测结合的软件评测思想；从传统的测试内容分类到基于质量特性、子特性的测试内容分类；从传统的结构化程序测试到面向对象等新型软件测试方法；从早期的单机或桌面测试到网络软件测试及嵌入式软件测试；从以简单的用例测试发展到具有各种测试工具支持软件测试及管理。事实上，随着软件越来越复杂，软件测试也就成为一件费力、耗时的复杂工作，这对测试人员提出了高素质、专业化的要求。软件测试人员不但要精通各种软件测试技术和方法，有软件测试工程实践经验；还要求他们熟悉软件开发技术和流程，具有快速学习专业知识或领域知识，掌握新技术和应用新工具的能力；此外，软件测试人员还应有团队合作精神，善于和人沟通与交流，并能承受被人误解和指责的心理素质。

随着计算机技术的快速发展，软件越来越普遍地应用到各个领域和方面，应用规模越来越大，应用模式越来越复杂，对软件质量的要求越来越高，从而使得与软件质量密切相关的软件测试越来越重要。当前高素质的软件测试专业人才奇缺，软件测试人才的培养十分紧迫。国家示范性软件学院的一个重要职责就是要在教学研究、教学实践以及教学改革方面进行大胆的探索和实践。因此，在完善已有课程体系及授课内容的同时，应该充分利用优秀的教学资源，总结教学经验和科研成果，编写专业教材，力争探出一条为国家快速培养高素质软件工程人才之路。

北京工业大学软件学院蔡建平教授长期从事软件工程、软件测试以及软件质量保证的研究、实践和教学，并已为编写此书做了长时间的辛勤工作。从已成的书稿看，该教材具有几个主要特点：

- ◆ 教材十分重视突出重要知识点的组织和讲述，并且做到了软件测试知识覆盖的全面性，能够较好满足国内企业，特别是国内各种评测机构或组织对现代软件测试人才培养的要求；
- ◆ 在讲述传统软件测试技术和方法的基础上，教材强调了软件测试要与软件质量度量和评价相结合，并在软件工程全生命周期内实施软件测试相关活动，要充分重视软件开发方法和应用方式对软件测试的影响，要十分重视软件测试工具对软件测试的重要支撑作用等；
- ◆ 作者根据自己和他人的实际工程经验，在教材中给出了软件测试在几个典型应用领域具体实施的要点和注意事项，这对缺乏实践经验的培养对象而言具有极好的引领作用，对开阔软件测试人员的眼界、思路和具体实践有很大帮助。

- ◆ 书中不少内容取材于互联网，一定程度上体现了软件测试技术的最新发展，具有较强的新颖性和现代性。

蔡建平教授的书是一部好教材，它在现代软件测试技术的教学、普及、推广和软件测试人才培养以及软件测试教学知识体系的建立等方面进行了很好的探索。教材内容全面、详实，实用性强。它的出版必将有助于国内软件测试人员和计算机相关专业的本科生及研究生的培养，有益于推动现代软件测试技术和方法的研究、教学和实践的进一步发展，同时对我国软件测试业的发展起到积极的促进作用。

北京大学教授、院士 何新贵
2009.6.6 于北京

前 言

软件测试是软件工程的一个重要分支，是软件质量保证的重要基础。现代软件测试与传统的软件测试不一样，它贯穿软件工程整个软件生命周期(从需求分析、软件设计、代码开发，一直到最后测试、验收和维护)，涉及各种软件开发技术、应用技术以及测试技术，覆盖软件各种应用领域(桌面软件、网络应用、信息系统、电子服务、嵌入式系统、数字娱乐等)，面临不同的专业领域知识，所要求的投入与软件开发相比规模同等甚至更高。

软件测试是一项需具备较强专业技术、学习和创新能力的工作，软件测试人员必须要具有缜密的逻辑思维能力、全面的技术能力、敢想敢干的创新能力，要有较强的责任心和团队合作精神以及出色的沟通能力等专业素质，要成为一名合格的软件测试人才需要经过严格系统化的专业培养，并且在专业培养过程中一定要结合软件测试专业化的特殊要求，强调学生实践动手能力和工程应用能力。

近年来我国政府强调要加大工业化与信息化融合的力度，促进我国工业的发展和转型，这一措施有助于我国软件产业的升级，软件开发模式的升级。实际上，软件作为工业化的产品，软件测试是软件开发企业必不可少的质量监控环节，它在整个软件开发的系统工程中占据相当大的比重。在软件产业发达国家，软件企业一般是把 40%的工作花在测试上，测试人员和开发人员之比平均在 1:1 以上，软件测试费用占整体开发费用的 30—50%，对于要求高可靠性、高安全性的软件，测试费用则相当于整个软件项目开发所有费用的 3 至 5 倍。这些说明软件测试的重要性。

软件测试的上述特点和重要性表明软件测试职业专业性极强，知识面极广、对人员综合素质(包括学习能力、观察能力、沟通能力及团队合作能力)要求极高。加上软件测试新技术、新需求、新观念的发展和变化，传统的软件测试思想、方法以及与之对应的教材难以满足现代教学要求。因此，本人在多年从事软件工程、软件测试以及软件质量保证的研究和多年讲授软件测试课程经验和体会的基础上，对软件测试课程教学内容的知识点进行调整、补充和完善等方面的改革，特别针对我国国防工业如航空、航天、船舶、电子、通讯等大量国防、军用软件开发和应用的实际情况以及对软件质量评价和软件高可靠性的要求，针对国内软件研发和测试外包的本地化要求，以及针对软件测试用于各种应用领域的要求选择授课的知识点，并取得了较好的效果。

软件测试与软件开发、应用以及质量保证密切相关，而大多数软件测试人员更多地关注和应用传统的软件测试技术，而缺乏现代的软件测试知识。本书是目前国内软件测试教材中难得的全面、系统地介绍包括现代软件测试的书籍。

本书以现代软件测试需求为背景，以现代软件测试技术和方法为基础，以当前软件测试通常应用为典型实例，全面地介绍了软件测试的基本概念，软件测试的技术、方法和工具应用，以及软件测试在当前主流应用中的具体开展和实施。与国内常见的软件测试教材

重点讲授一般软件测试技术和方法不同，本书特色在于：

- ◆ 以现代软件测试思想为指导，除了全面讲述传统软件测试技术和方法外，还通过突出全生命周期软件测试概念、软件质量分析手段、现代软件测试技术、主流测试工具应用以及典型应用测试方法等，帮助学生了解和掌握现代软件测试的各种原理、方法和技术，并能够选择合适的软件测试工具进行相关测试。为培养学生今后成为高素质、专业化的软件测试人才打下基础。
- ◆ 针对软件开发方法和技术的发展变化，针对我国软件外包服务的蓬勃兴起，针对我国国防工业如航空、航天、船舶、电子、通讯等大量重要软件或关键软件的实际应用情况和测试要求，特别是对软件高可靠性的要求，选择教材的知识点；
- ◆ 本书对支撑现代软件测试技术应用的测试工具进行了全面地介绍，特别是对开源软件测试工具的介绍，这对高校开设软件测试实验课程是非常有意义的。
- ◆ 本书内容全面、条理清晰、结构严谨、可用性强，对重点、难点阐述透彻，使其既符合现代软件测试技术发展的潮流，又具有相对的稳定性，还易于剪裁，以满足各类软件测试课程的教学需要和各类软件测试人员的学习需要。

事实上，撰写软件测试方面的教材想法来源于我国软件工程领域资深专家何新贵院士很早的一个建议(在这里我首先对何院士表示感谢)。三年前，何院士指导北京工业大学软件学院办学时特别指出学院要将软件工程学科和专业建设的重点放在软件质量保证和软件评测方向上，以提升我国软件质量保证和软件评测的专业方向建设水平，解决我国这方面人才严重匮乏的问题。这样，本人就有了编写软件测试课程教材的计划，也与有关出版社签订了编写协议，由于工作太过繁忙，无法按期完成编写任务而最终放弃。为此，我在这里表示歉意。

本书撰写的再一次启动并最终完稿首先要感谢学校质量工程建设任务的安排，作为工作我没有后退之路，必须完成；同时要感谢清华大学出版社对我编写本书过程中发生的一些变化所给予的理解和通融。本书的完成得益于许多教师和学生的积极参与，如早期的何坚老师和石振莲教师，以及王毅同学等。在这里我要特别感谢我指导的研究生路翠同学，本书的撰写和完稿她花费了很多心血，如资料的收集，大部分章节的起草，图表的制作，以及习题的编写等。没有她的积极参与，本书的完稿时间可能还会一拖再拖。另外，本书很多内容是对互联网资料进行收集、整理和改编的结果，包括软件测试专业网站，软件测试专家或工程师的博客，以及软件测试人员在论坛上论坛的经验之谈。我认为这些素材是很有意义的，尽管他们中的很多人可能很平常——不是学者或专家，但他们是从实际工作或项目中总结出的经验、体会或感想，对于软件测试教学特别是案例驱动教学很有帮助。因此，在对原则性错误、语法错误和文字错误等进行简单地改编外，我基本上原汁原味地奉上。当然，很多网上资料由于转载或引用频繁找不到原创处，在参考文献中无法标注，我在这里表示遗憾。最后，我要感谢我的家人，本书的撰写全部是利用寒暑假、节假日、周末和晚上的时间，没有家人的支持和照顾，本书也是很难完成的。

后续我们将组织教师和学生编写软件测试课程设计实验教材、嵌入式软件测试书籍，但愿我们能按计划如期完成。

目前国内关于软件测试方面的书较多，其中的很多是翻译的、为培训用的或介绍软件测试技术和方法的，作为教材或满足各类测试人员的学习还是有一定距离。本书是在本人长达二十几年软件工程、软件测试和软件质量保证实践经验和教学经验的基础上借鉴前人的成果，参考当前软件测试方法和技术应用实践案例进行编写的。尽管如此，由于软件测试覆盖面太大，应用周期太长，涉及领域太多，测试方法和技术太繁杂，加之时间紧、水平有限，一定有许多不周到、不准确或存在错误之处，恳请读者提出批评和建议，并争取再版时修正。

北京工业大学软件学院 蔡建平
2009年5月20日 于北京工业大学

作者简介

蔡建平，教授，北京工业大学软件学院院长助理、软件与网络工程系主任。在原总装备部某研究所工作十余年，一直从事该所负责的全军军用共性软件系统项目的论证和研究，并在其中发挥重要作用，得到有关领导和专家的认可和好评。获军队科技进步一等奖、二等奖、三等奖多项；发表各类学术论文 30 多篇，与他人合作著书一部。

在北京奥吉通科技有限公司任技术总监期间，除负责国防有关单位的软件工程、软件质量保证、软件测试以及嵌入式开发的技术咨询、提供解决方案和技术支撑外，还主持开发了科锐时系列软件测试工具，并成功用于国防软件的测试。

2005 年 9 月在北京工业大学软件学院任职教授以来，除了多年讲授软件测试课程和多次对外开展嵌入式软件测试技术培训外，在学院的学科建设、专业建设，如实验室建设、实践教学改革与创新、数字艺术方向和数字媒体技术专业建设、“211 工程”建设、教育部和北京市特色专业建设，以及科研基地——科技创新平台建设等方面做了大量的工作，取得了突破性成果。

目 录

第 I 部分 软件测试基础

第 1 章 软件与软件危机	3
1.1 软件概念	3
1.2 软件特性	3
1.3 软件种类	5
1.4 软件危机及其原因	5
1.5 软件工程	9
1.6 软件生命周期	12
1.6.1 什么是软件生命周期	12
1.6.2 软件生命周期的阶段划分	12
1.6.3 软件生命周期模型	13
习题和思考题	15
第 2 章 软件缺陷与缺陷管理	16
2.1 缺陷定义与分类	16
2.1.1 软件缺陷	16
2.1.2 软件缺陷的描述和分类	19
2.2 缺陷管理流程	25
2.3 缺陷度量 and 缺陷报告	27
2.3.1 缺陷度量	27
2.3.2 缺陷报告	29
2.4 缺陷管理工具	32
2.4.1 BugRat(开源)	33
2.4.2 TrackRecord(商用)	33
2.4.3 ClearQuest(商用)	34
2.4.4 Bugzilla(开源)	34
2.4.5 QAMonitor(国内商业软件)	35
2.4.6 BMS(国内商业软件)	36
2.4.7 其他	36
习题和思考题	37

第 3 章 软件测试基本概念	38
3.1 软件测试发展史	38
3.2 软件测试的定义	39
3.3 软件测试的目的	40
3.4 软件测试的原则	41
3.5 软件测试人员的素质要求	43
3.6 软件测试与软件开发 各阶段的关系	45
3.7 软件测试的就业前景	45
3.8 有关软件测试的几个误区	46
习题和思考题	50
第 4 章 生命周期软件测试方法	52
4.1 生命周期测试任务	53
4.2 需求阶段测试	58
4.3 设计阶段测试	59
4.4 编码阶段测试	60
4.5 测试阶段	61
4.6 安装阶段测试	62
4.7 验收阶段测试	63
4.8 维护阶段	63
4.9 全生命周期质量管理 平台 Panorama++	63
习题和思考题	68
第 5 章 软件测试过程	69
5.1 软件测试过程模型	70
5.1.1 V 模型	70
5.1.2 W 模型	71
5.1.3 H 模型	72
5.1.4 其他测试模型	73

5.1.5 测试模型的使用	73	7.3 程序代码质量检查	122
5.2 软件测试过程管理	74	7.3.1 代码分析	123
5.2.1 测试计划阶段	75	7.3.2 代码审查	124
5.2.2 软件测试设计和开发	78	7.3.3 桌面检查	126
5.2.3 测试执行阶段	81	7.3.4 代码走查	126
5.2.4 测试执行结束和测试总结	83	7.3.5 技术评审	126
5.2.5 测试文档与测试过程 的关系	83	7.4 软件复杂性分析	126
5.3 测试过程改进	84	7.4.1 Line Count 复杂度	127
5.3.1 软件测试过程改进的概念	85	7.4.2 Halstead 复杂度	128
5.3.2 软件测试过程改进的 具体方法	86	7.4.3 McCabe 复杂度	130
5.4 测试过程辅助设计 与管理工具	87	7.4.4 软件复杂性度量与控制	133
5.4.1 HP-Mercury TestDirector	87	7.5 软件质量模型	137
5.4.2 北航软件所 QESuite	90	7.5.1 软件质量概念	138
5.4.3 TestLink(开源免费)	92	7.5.2 软件质量分层模型	140
习题和思考题	92	7.5.3 软件质量度量与评价	147
第 6 章 基于软件配置项的测试种类	93	7.6 静态分析工具	154
6.1 计算机软件配置项	93	7.6.1 静态分析工具 Logiscope	
6.2 软件测试类别和测试种类	94	RuleCheck/Audit	155
6.2.1 测试种类介绍	97	7.6.2 源代码分析工具	
6.2.2 测试种类的选择	101	Klocwork	161
习题和思考题	101	习题和思考题	169
第 II 部分 软件测试方法		第 8 章 动态测试	170
第 7 章 软件静态测试	105	8.1 “白盒”测试	170
7.1 编程规范	105	8.1.1 逻辑覆盖	172
7.2 程序理解	110	8.1.2 路径测试	174
7.2.1 程序理解的概念	111	8.1.3 数据流测试	179
7.2.2 程序理解的任务	112	8.1.4 覆盖率分析及测试 覆盖准则	180
7.2.3 程序理解的内容	112	8.2 “黑盒”测试	182
7.2.4 程序理解的方法	113	8.2.1 等价类划分	184
7.2.5 程序理解的相关技术	114	8.2.2 边界值分析	187
7.2.6 影响程序理解的因素 及对策	121	8.2.3 因果图	188
		8.2.4 随机测试	191
		8.2.5 猜错法	191
		8.3 “灰盒”测试	192
		8.4 测试用例设计	194

8.4.1	测试用例设计原则	195	9.2	软件测试自动化框架	244
8.4.2	测试用例要素	195	9.2.1	TestHarness	244
8.4.3	测试用例的设计步骤	196	9.2.2	常用的自动化测试框架	245
8.5	单元测试	198	9.2.3	基于 API 测试的分布式 测试框架	247
8.5.1	单元测试的意义	199	9.3	测试自动化技术	251
8.5.2	单元测试的内容	201	9.3.1	脚本技术	252
8.5.3	单元测试方法和步骤	204	9.3.2	录制/回放技术	254
8.5.4	国产单元测试工具 Visual Unit	206	9.3.3	基于测试预期的结果分析	257
8.6	集成测试	208	9.3.4	第三方测试工具集成	257
8.6.1	一次性组装方式	209	9.4	自动化测试工具应用举例	263
8.6.2	渐增式测试	210	9.4.1	测试中常用的自动化 测试工具	263
8.6.3	组装测试的组织和实施	214	9.4.2	基于 STAF/STAX 的 自动化测试框架	264
8.7	确认测试	214	习题和思考题	269	
8.7.1	有效性测试	214	第 10 章	软件可靠性测试	270
8.7.2	软件配置复查	215	10.1	操作剖面与统计测试	270
8.7.3	α 和 β 测试	216	10.2	基于操作剖面的软件 可靠性测试	271
8.8	系统测试	217	10.2.1	基于操作剖面的 统计测试	272
8.8.1	系统测试的目的与意义	217	10.2.2	操作剖面的构造	273
8.8.2	系统测试中关注的 重要问题	217	10.2.3	操作剖面的使用	278
8.8.3	系统测试的要求 和主要内容	221	10.2.4	结论	279
8.8.4	系统测试设计	227	10.3	基于操作剖面的软件 可靠性疲劳测试	279
8.8.5	系统测试中的软件攻击	230	习题和思考题	280	
8.8.6	系统测试中的典型 测试类型介绍	233	第 11 章	软件本地化测试	281
8.8.7	系统测试工具举例	236	11.1	软件本地化测试的目的 和特点	282
习题和思考题		239	11.1.1	软件本地化 测试的目的	282
第 9 章	软件测试自动化	240	11.1.2	软件本地化 测试的特点	282
9.1	软件测试自动化概念	240			
9.1.1	自动化测试的定义	240			
9.1.2	适合于自动化测试的 相关活动	241			
9.1.3	自动化测试的优点	242			
9.1.4	自动化测试的局限性	242			

- 11.2 软件本地化错误类型及其原因..... 283
 - 11.2.1 翻译错误.....283
 - 11.2.2 功能错误.....283
 - 11.2.3 国际化错误.....284
 - 11.2.4 本地化错误.....284
- 11.3 软件本地化测试的类型..... 285
- 11.4 软件本地化测试的原则..... 288
- 习题和思考题..... 288
- 第 12 章 面向对象软件测试..... 289**
 - 12.1 面向对象程序设计语言对软件测试的影响 290
 - 12.1.1 信息隐蔽对测试的影响.....290
 - 12.1.2 封装和继承对测试的影响.....290
 - 12.1.3 集成测试.....291
 - 12.1.4 多态性和动态绑定对测试的影响.....292
 - 12.2 面向对象测试模型..... 293
 - 12.2.1 面向对象分析测试(OOA Test)294
 - 12.2.2 面向对象设计的测试(OOD Test)297
 - 12.2.3 面向对象编程的测试(OOP Test)298
 - 12.2.4 面向对象的单元测试(OO Unit Test)300
 - 12.2.5 面向对象的集成测试(OO Integrate Test)302
 - 12.2.6 面向对象的系统测试(OO System Test).....303
 - 12.3 面向对象软件的回归测试..... 304
 - 12.4 UML 图和面向对象软件测试 304

- 12.5 面向对象软件测试用例的设计.....307
 - 12.5.1 基于故障的测试.....307
 - 12.5.2 基于脚本的测试.....308
 - 12.5.3 面向对象类的随机测试.....308
- 习题和思考题.....309

第III部分 典型应用软件测试

- 第 13 章 C/S 应用软件测试 313**
 - 13.1 C/S 系统测试对传统测试的影响.....314
 - 13.2 C/S 系统测试的目标317
 - 13.3 C/S 系统测试的内容和步骤.....318
 - 13.3.1 C/S 系统测试的内容.....318
 - 13.3.2 C/S 系统测试的常见测试点.....320
 - 13.3.3 C/S 系统的性能测试.....321
 - 13.3.4 C/S 系统的测试步骤.....322
 - 13.4 C/S 系统的测试工具322
 - 习题和思考题324
- 第 14 章 基于 B/S 架构的 Web 应用软件测试 325**
 - 14.1 C/S 和 B/S 的比较326
 - 14.2 Web 应用测试327
 - 14.2.1 功能测试.....328
 - 14.2.2 性能测试.....329
 - 14.2.3 可用性测试.....330
 - 14.2.4 客户端兼容性测试.....332
 - 14.2.5 安全性测试.....333
 - 14.2.6 接口测试.....333
 - 14.3 Web 应用性能测试方法334
 - 14.4 Web 应用的测试模型337
 - 14.4.1 对象模型.....338
 - 14.4.2 行为模型.....338

14.4.3	结构模型	339	16.3.2	嵌入式“黑盒” 测试工具	369
14.4.4	基于统计信息的 Web 测试	340	16.3.3	嵌入式“灰盒” 测试工具	370
14.4.5	性能测试模型	340	16.3.4	嵌入式软件仿真工具	370
14.5	Web 信息系统测试	341	16.4	嵌入式软件测试工具举例	370
14.5.1	信息系统的概念	341	16.4.1	美国 Vector 公司的 VectorCAST ——单元测试工具	370
14.5.2	信息系统的分类	344	16.4.2	IBM Rational Test RealTime(RTRT)	370
14.5.3	Web 信息系统 测试方法	346	16.4.3	美国 FreeScale 公司的 CodeTest——“白盒” 测试工具	371
	习题和思考题	348	16.4.4	比利时 SPACEBEL 公司 产品——全数字仿真 测试工具	371
第 15 章	游戏软件测试	349	16.4.5	荷兰 CHESS 公司产品 SHAM——半实物仿真 测试工具	371
15.1	游戏软件测试的特性	350	16.4.6	北航的“黑盒”测试 工具 GESTE	372
15.2	常见的游戏软件错误	350	16.4.7	欧洲航天局实时仿真 产品 Eurosim	372
15.3	游戏软件测试与游戏 开发过程	351	16.4.8	网络资源透明访问 工具 SPINEware	373
15.3.1	游戏开发过程	351	16.5	传统测试工具的局限性	373
15.3.2	游戏测试与开发过程 的关系	352	16.6	全数字仿真测试方案	374
15.4	网络游戏测试	356	16.6.1	全数字仿真的概念	374
15.4.1	网络游戏的 平衡性测试	357	16.6.2	北京奥吉通的 CRESTS/ATAT 和 CRESTS/TESS 介绍	374
15.4.2	网络游戏的性能测试	359		习题和思考题	377
15.5	手机游戏的测试内容	360		参考文献	378
	习题和思考题	363			
第 16 章	嵌入式软件测试	364			
16.1	嵌入式软件测试概念	365			
16.2	嵌入式软件测试的 传统方法	366			
16.2.1	嵌入式软件测试各个 阶段的通用策略	367			
16.2.2	嵌入式软件测试 的流程	367			
16.3	嵌入式软件测试的 典型工具	369			
16.3.1	嵌入式“白盒” 测试工具	369			

第 I 部分 软件测试基础

1947 年，计算机还是由机械式继电器和真空管驱动的、房间那么大的庞然大物，由哈佛大学制造的 MarkII 则是体现当时技术水平的计算机。在一次整机运行中，它突然停止了工作。技术人员爬到计算机上找原因，发现是一只飞蛾受光和热的吸引飞到了计算机内部一组继电器的触点之间，然后被高电压击死。虽然最后该问题被技术人员解决了，但是我们从此认识了它的缺陷。

如今软件已经渗透到了人们的日常生活中，从办公设备到家用电器，从通信工具到航空航天事业，软件无处不在，然而却又很难完美无缺。

1994 年秋天，迪斯尼公司发布了第一个面向儿童的多媒体光盘——狮子王动画故事书 (The Lion King Animated Storybook)。对此，迪斯尼公司做了大量的宣传。因此，销售额非常可观。然而圣诞节过后，公司接到了大量投诉电话，称游戏不能运行。经证实，造成这种后果的原因是迪斯尼公司未对市面上使用的许多不同类型的 PC 机型进行测试，软件只能在少数系统中运行。

同样是 1994 年，英特尔奔腾浮点除法缺陷事件，不仅使英特尔公司的形象受到严重影响，而且使其为自己处理软件缺陷的行为付出了 4 亿多美元的代价。

类似的还有美国航天局火星极地登陆者号探测器事件、爱国者导弹防御系统事件、千年虫问题等。这些事件的后果有的是带来不便，例如游戏不能运行，有可能是灾难性的——导致机毁人亡。它们的发生都是由于在软件中隐藏着错误。

软件为什么会频繁出问题，如何杜绝或将它们减至最少，影响降至最低呢？在论述软件测试概念之前先介绍一下软件、软件危机及软件工程等概念，然后再讲解软件测试的相关知识。



第1章 软件与软件危机

1.1 软件概念

随着计算机技术的发展，不同阶段有不同的认识。计算机发展的初期，硬件的设计和生产是主要问题，那时的所谓软件，就是程序，甚至是机器指令程序，它们处于从属的地位。软件的生产方式是个体的手工方式，设计是在一个人的头脑中完成的，程序的质量完全取决于个人的编程技巧。其后，人们认识到在机器上增加软件的功能会使计算机系统的能力大大提高，于是在研制计算机系统时既考虑硬件，又考虑软件，而且开始编制一些大型程序系统。这时的生产方式类似于互助合作的手工方式，所以人们认为软件就是程序加说明书。后来，社会需要对计算机提出了更高的要求，有的大型系统的设计和生产的工程量高达几千人/年，指令数百万条，有的达几千万条，如美国宇航飞船的软件系统有 4000 万条语句。现在，软件在计算机系统的比重越来越大，而且这种趋势还在增加。所以人们感到传统的软件生产方式已不适应发展的需要，于是提出把工程学的基本原理和方法引进到软件设计和生产中，就像机械产品一样，软件生产也被分成几个阶段，每个阶段都有严格的管理和质量检验，科学家们研制了软件设计和生产的方法和工具，并在设计和生产过程中用书面文件作为共同遵循的依据。这时软件的含义就成了文档加程序。文档是软件的“质”的部分，程序则是文档代码化的形式。

现在软件的正确含义应该是：

- 当运行时，能够提供所要求功能和性能的指令或计算机程序集合。
- 该程序能够具有满意地处理信息的数据结构。
- 描述程序功能需求以及程序如何操作和使用所要求的文档。

1.2 软件特性

软件是人通过智力劳动产生的，软件产品是人的思维结果，是一个逻辑部件，而不是一个物理部件。因此，软件生产水平最终在相当程度上取决于软件人员的教育、训练和经