HZ BOOKS
华章教育

计 算 机 科 学 丛 书

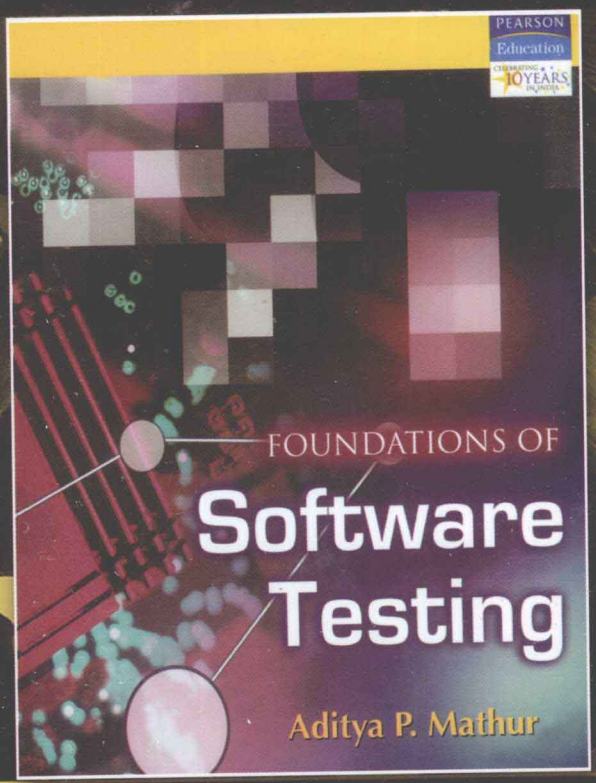
PEARSON

软件测试基础教程

(美) Aditya P. Mathur 著
普度大学

王峰 郭长国 陈振华 等译　　王峰 宗建建 施寅生 校

Foundations of Software Testing



机械工业出版社
China Machine Press

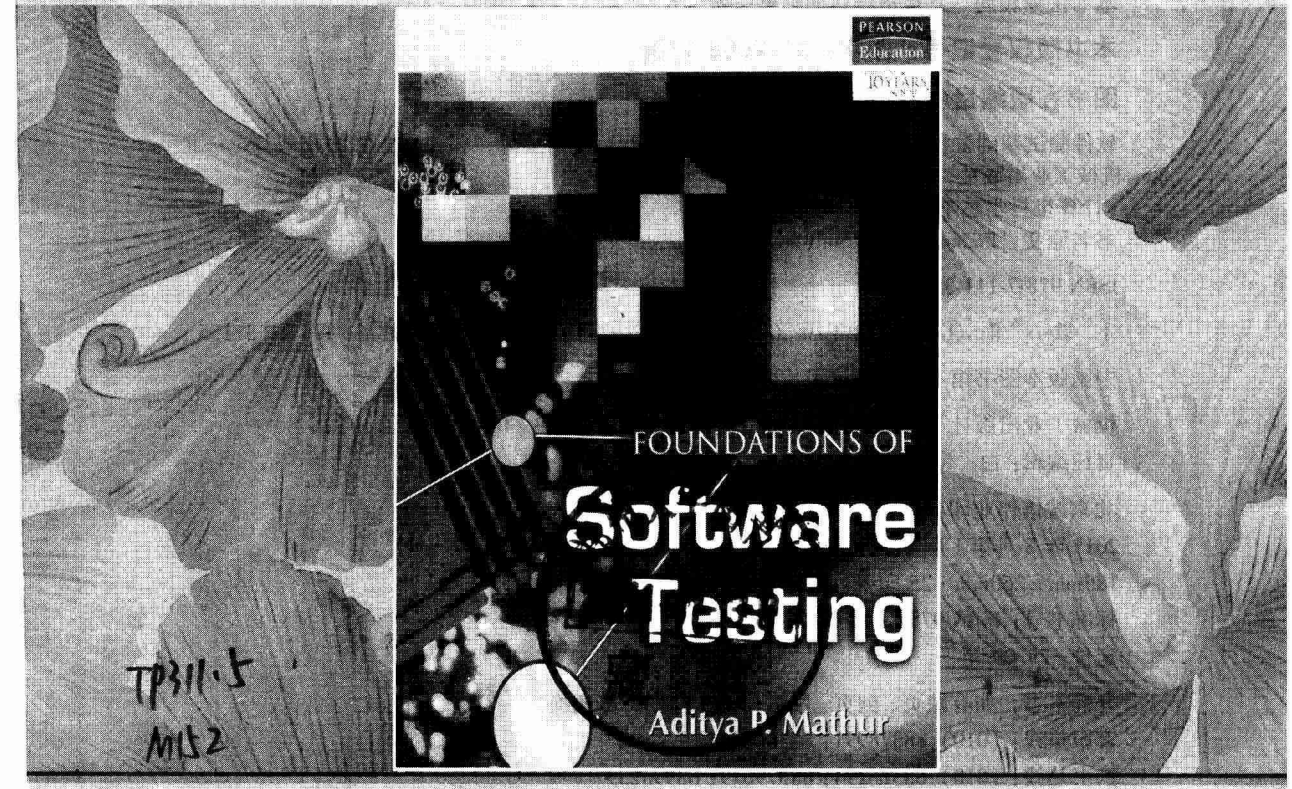
算 机 科 学 丛 书

软件测试基础教程

(美) Aditya P. Mathur 著
普度大学

王峰 郭长国 陈振华 等译　　王峰 宗建建 施寅生 校

Foundations of Software Testing



机械工业出版社
China Machine Press

本书全面介绍了软件测试的相关理论、测试方法、测试生成技术等内容。全书分为三个部分：第一部分是预备知识，介绍软件测试技术的相关术语等基础知识；第二部分介绍软件测试的生成技术，不仅包括基本的等价类划分、边界值分析、因果图、谓词测试等技术，还涵盖了从有穷状态模型自动生成测试的技术、基于组合设计的测试生成技术，以及用于回归测试中测试选择、优先级排序、最小化的一些基本技术；第三部分介绍软件测试中既重要又广泛适用的理论，即通过测试充分性的度量来加强测试，包括基于控制流、数据流的代码覆盖标准，以及最有效的基于程序变异的测试充分性度量标准。每章的结尾都有参考文献注释和练习题，帮助读者深入体会软件测试的过程，并熟练掌握测试生成的方法。

本书适合作为计算机、软件工程及相关专业软件测试课程的教材，也可作为软件测试技术人员的参考书。

本书版权登记号：图字：01-2009-1732

图书在版编目(CIP)数据

软件测试基础教程/（美）马瑟（Mathur, A. P.）著；王峰，郭长国，陈振华等译. —北京：机械工业出版社，2011.8

（计算机科学丛书）

书名原文：Foundations of Software Testing

ISBN 978-7-111-35188-7

Ⅰ. 软… Ⅱ. ①马… ②王… ③郭… ④陈… Ⅲ. 软件－测试－教材 Ⅳ. TP 311.5

中国版本图书馆 CIP 数据核字（2011）第 127150 号

机械工业出版社（北京市西城区百万庄大街 22 号　邮政编码　100037）

责任编辑：白　宇

北京诚信伟业印刷有限公司印刷

2011 年 8 月第 1 版第 1 次印刷

185mm×260mm · 25 印张

标准书号：ISBN 978-7-111-35188-7

定价：75.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991；88361066

购书热线：(010) 68326294；88379649；68995259

投稿热线：(010) 88379604

读者信箱：hzjsj@hzbook.com

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭示了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自 1998 年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与 Pearson，McGraw-Hill，Elsevier，MIT，John Wiley & Sons，Cengage 等世界著名出版公司建立了良好的合作关系，从他们现有的数百种教材中甄选出 Andrew S. Tanenbaum，Bjarne Stroustrup，Brain W. Kernighan，Dennis Ritchie，Jim Gray，Afred V. Aho，John E. Hopcroft，Jeffrey D. Ullman，Abraham Silberschatz，William Stallings，Donald E. Knuth，John L. Hennessy，Larry L. Peterson 等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专程为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近两百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段，我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下：

华章网站：www. hzbook. com
电子邮件：hzjsj@ hzbook. com
联系电话：（010）88379604
联系地址：北京市西城区百万庄南街 1 号
邮政编码：100037

华章教育
华章科技图书出版中心

译者序

Foundations of Software Testing

本书的翻译、审校工作基本结束，三年半来一直压在心头的石头终于可以落地了。

2007 年 12 月 24 日下午，我应温莉芳编审、姚蕾编辑之约，前往机械工业出版社华章公司讨论本书的翻译工作。首次浏览原版书目录，顿感又是一本难得的好书，很是兴奋，当时就答应下来。

2008 年元旦过后，开始组建翻译小组并进行分工。由我本人负责第一部分、陈振华负责第二部分、郭长国负责第三部分，并由我负责全书的审校和统稿。翻译工作自当年春节后正式开始。

陈振华于 2008 年 4 月初提交了第 2 章 2.4 节之前各节的译稿。但天有不测风云，就在我返回他修改稿之后不久，陈振华因公出差不幸在湖北省襄樊市发生特大车祸，严重受伤，后辗转襄樊、北京、天津多家医院，身体每况愈下，自 2008 年 9 月底昏迷以后到现在尚未苏醒过来。原由他负责的第二部分翻译工作改由郭长国和我共同承担。在本书即将出版之际，我们衷心祝愿陈振华能够早日苏醒过来，最终康复。

本书翻译工作异常艰辛。第一、没有时间，所有的翻译工作皆在业余时间完成。因本职工作太忙，很少有空余时间，下班之后也没多少精力再去翻译。但既然答应承担翻译任务，心里就放不下。因此，很多次外出开会、出差，甚至在“5.12”地震之后回灾区老家，我都在行囊里背着原版书和电脑，我也明知在外时不可能翻译，只是出于对翻译任务的尊重，以求得心里的安慰。由于没有整块的时间翻译，很多内容是“炒夹生饭”，刚一读懂就放下，过几天甚至一两个月再来，反反复复，第 7 章个别小节内容审校不下五次。第二、原版书排版质量太差。初步统计，原版书中明显错误不下 200 处。为了保证图书质量，我们花费了很多时间纠正原版书中出现的错误，尤其是一些出现在算法和数学公式中的错误，必须推导验证才能排除。特别是第 3 章，我们根据自己对算法的理解，增补了大量测试用例。翻译早期，我就原书第 2 章的错误与作者 Aditya P. Mathur 教授交流过，他全部接受，称我们的纠错工作为 excellent。

关于本书的内容，读者一看目录就清楚了。需要强调的是，这是一本到目前为止我本人读过的关于测试技术的最全面、最深入的书。我以前曾翻译过两本关于软件测试的书，Cem Kanner 的 *Testing Computer Software* 第 2 版和 Glenford J. Myers 的 *The Art of Software Testing* 第 2 版，前者偏重测试技术管理，后者就是一本测试普及小册子。我也常去北京的新华书店，浏览有关软件测试的图书，包括翻译书和原创书，估计不下 30 种。我可以负责任地讲，尽管不敢夸口说本书是关于测试技术最好的书，但只要你读过本书之后，很多其他的书真的没必要看了。虽然看完书中的内容之后不可能马上就学会软件测试，但这些知识却是作为一个软件测试（甚至是软件）从业人员必须掌握的。等价类划分、边界值分析、因果图分析、有穷状态机、组合设计、程序切片、控制流、数据流、程序变异等，我们在软件工程课程中都能学到这些知识，但如何应用于软件测试，还是本书第一次系统地做了介绍。尤其是其后面所列的 500 多篇参考文献，几乎囊括了到 2006 年所有有价值的软件测试文献，对从事软件测试研究的人员来说，如获至宝。

本书由王峰、郭长国、陈振华翻译，其中王峰主要负责前言、致谢、第 1 章、第 2 章 2.5 节以后各节、第 3 章、封底；郭长国负责第 4 ~ 7 章；陈振华负责第 2 章 2.4 节之前各节。由

王峰、宗建建、施寅生审校，其中宗建建负责第5章第一、第二次审校；施寅生负责第4章第一次审校；王峰负责第2章、第4章第二次、第5章第三次、第6章、第7章审校。郭长国校读了前言、致谢、第1~4章和第6章，施寅生校读了第7章7.10.7节以后各节。此外，参加翻译工作的还有赵志强、郑彦兴、李海龙、齐璇、齐超、苏晓艳、喻琳、衣双辉、房友园、谷天阳、刘宇、包阳、李冬红、杨广华、战茅、张鲁靖，全书由王峰统稿。

虽然我们尽力纠正原版中的错误，但书中可能仍然存在疏漏与错误，诚恳地希望各位读者批评指正。

王　峰

2011年6月17日晚于北京

本书翻译工作得到国家863计划课题2009AA01Z146的资助，特此感谢。

前 言
Foundations of Software Testing

欢迎您阅读《软件测试基础教程》！希望本书对您有所帮助！

文如其名，本书将向您讲述什么是软件测试。

对于那些准备从事 IT 行业的学生来说，选择一门软件测试方面的课程是很重要的。这门课程应该为学生提供一个获取他们职业生涯中永远有用的知识的机会，这也是很重要的，因为他们的工作将涉及大量的软件应用系统、软件产品以及不断变化的外部环境。

本书旨在介绍软件测试的知识，是该领域比较合适的教材。本书浓缩了全世界数以百计测试研究人员、一线测试人员的经验，并以易理解的方式呈现给读者。

软件测试所有技术活动的基础，在于测试生成、选择、优先排序以及评价。合理应用这些基本技术，可以测试不同的软件应用系统，以及各种质量特性。应用系统涉及面向对象系统、Web 服务软件、图形用户界面（GUI）、嵌入式系统等，而软件质量特性包括安全性、可靠性、性能、可维护性等。

随着软件日益渗透到我们日常生活的方方面面，软件测试的重要性就愈加明显。不幸的是，现在只有少数几所大学有资格开设软件测试课程，而他们还在努力地为课程选择教材。所以，我希望本书有助于科研院所开展软件测试教学，对于那些已开设该门课程的院校，将不再为寻找一本合适的教材而苦恼了，或者不再简单地依赖于一些研究性出版物了。

通过与商用软件开发单位中的测试人员和项目经理接触，我发现，虽然软件测试被认为是一项重要的工作，但软件测试人员经常抱怨与系统开发人员和设计人员相比他们没有得到应有的重视。我相信，提高软件测试的技术水平，将有利于开展高水平的软件测试工作，保证生产出高质量的软件，同时，也会给软件测试这种职业带来正面效应。我希望，一个学生即使掌握本书中知识的一半也能树立起他对软件测试作为一门学科应有的信心，就像编译原理、数据库、算法、计算机网络等成熟学科一样。

本书的读者对象

很自然，有人会问：这本书是面向哪个层次的读者？根据我和一些使用过本书初稿的教师的经验，本书最适合高年级本科生和低年级研究生。虽然本书的表述风格是以学院或综合性大学在校学生为读者对象，但我相信它对一线测试人员和测试研究人员同样也是有用的。如果有耐心的话，一线测试人员会发现这本书提供了丰富的技术资源供他们学习，并能够应用到开发和测试工作中；测试研究人员很可能会发现本书是很好的参考资料。

本书涵盖的内容

软件测试涉及很多活动。从大的方面看，这些活动似乎一样，但从小的方面看，它们却大相径庭。例如，大多数软件开发环境都能进行测试，但是对操作系统的测试与对心脏起搏器的测试却大不一样——一个是开放式系统，一个是嵌入式系统，需要采用不同的测试执行方式。

软件测试活动中相似性和差异同时存在，这为作者以及教师出了个难题。一本书或者一门

课程是应该专注于具体的软件开发环境以及它们完成不同测试活动的方式，还是应该专注于具体的测试技术而对环境轻描淡写？任何一种做法都会招来批评，并且会让学生要么与测试应用环境脱节，要么完全不了解测试理论。

通过精心选择、组织书中的材料，我成功地解决了这个难题。本书分为三部分，主要讲述各种测试技术的基础理论。第一部分通过实例来介绍在不同的软件开发组织中软件测试过程的差别。第二部分介绍用预期的程序行为模型生成测试的技术。第三部分介绍测试充分性的度量与增强技术。

本书的组织

本书由三部分组成。

第一部分涵盖与测试相关的概念和预备知识。第 1 章，也是本部分唯一的一章，介绍软件测试中普遍涉及的术语和基本概念。一些采用该书早期初稿作为本科教材的教师通常在前两周或三周内讲完本章的内容。

第二部分涵盖不同的测试生成技术。第 2 章介绍广泛适用于几乎所有软件应用系统的最基本的测试生成技术，包括等价类划分、边界值分析、因果图、谓词测试等。第 3 章介绍从有穷状态模型自动生成测试的技术，包括 W、Wp 以及 UIO（Unique Input - Output）方法。有穷状态模型大量应用于诸如面向对象测试、安全性测试、GUI 测试。第 4 章介绍基于组合设计的测试生成技术。一旦软件版本升级或进行广泛维护时，回归测试是所有软件开发过程的一个组成部分。第 5 章介绍用于回归测试中测试选择、优先级排序、最小化的一些基本技术。

第三部分涵盖软件测试中既重要又广泛适用的理论和技术，即通过测试充分性的度量来加强测试。第 6 章介绍了多种基于控制流、数据流的代码覆盖标准，以及如何将它们应用于实际测试工作。第 7 章介绍最有效的基于程序变异的测试充分性度量标准。尽管几乎每一个软件开发组织都有一些测试充分性度量方法，但是通过掌握本部分的知识，确实能帮您把测试充分性度量与增强技术提升到一个新的高度，从而可以明显地提高软件的可靠性。

一线测试人员常常抱怨，许多白盒测试的充分性标准在集成测试和系统测试时是不可用的，这种抱怨在大多数时候是正确的。本书讨论了一些最有效的测试充分性评价标准如何才能够、也应该应用于除单元测试之外的测试。当然，我的建议是以用商用工具进行测试充分性评价为前提的。

本书每一章的结尾都有详细的参考文献注释。在引用与该章内容相关的文献时，我努力做到尽可能全面、综合。希望教师和学生会发现每章的参考文献注释那一节有益于他们了解本书之外更丰富的知识。

本书未涵盖的内容

软件测试包含大量相关、交织的活动。一些活动是技术性的，一些是管理性的，而另一些活动只是些规程。技术性活动包括单元测试、子系统测试、集成测试、系统测试、回归测试中测试用例和测试预期结果的设计。管理性活动包括人员计划、成本预算和报告。计划活动包括测试计划、质量评估和人员分配。最好将如人员分配之类的计划活动划分为管理性的，而另一些计划活动，如测试计划，是与测试用例设计等技术性活动交织在一起的。

一些测试活动是与具体的产品相关的。例如，对设备驱动器的测试常常需要开发设备模拟器，而设备模拟器包括：测试心脏起搏器时用的心脏模拟器、测试 I/O 驱动器时用的 USB 端

口模拟器、测试飞机升空噪音控制软件时用的飞机升空噪音模拟器。这些活动对测试的有效性和自动化极其重要，因而常常需要做大量的开发工作。例如，开发一个设备模拟器并对其进行测试，这既是开发活动，又是测试活动。本书中描述的测试生成和评价技术适用于每一个与具体产品相关的测试活动，当然，这些测试活动只是通过实例来说明的，并不作详细描述，我的建议是，学生最好通过工业部门赞助的实践项目来学习这些知识。

给教师的建议

软件测试课程涵盖的主题有很多种，我尽量使本书涵盖大部分最重要的主题。表1、表2分别给出了完全基于本书的本科生、研究生测试课程教学大纲。

表1　典型的本科生软件测试课程

教　学　周	教 学 内 容	本书章节
第1周	课程目标与目的，实习项目安排，测试术语和概念	第1章
第2周	测试过程与管理	第1章
第3周	软件错误、软件故障、软件失效	第1章
第4周	边界值分析，等价类划分，判定表	第2章
第5、6周	基于谓词的测试生成	第2章
第7周	项目中期汇报 复习，期中考试	
第8周	测试充分性：控制流	第6章
第9周	测试充分性：数据流	第6章
第10、11周	测试充分性：程序变异	第7章
第12、13、14周	特殊专题，如面向对象测试、安全性测试	另一卷
第15、16周	复习，项目总结汇报	
第17周	期末考试	

表2　典型的研究生软件测试课程

教　学　周	教 学 内 容	本书章节
第1周	课程目标与目的，测试术语和概念	第1章
第2周	测试过程与管理 软件错误、软件故障、软件失效	第1章
第3周	边界值分析，等价类划分，判定表	第2章
第4周	基于谓词的测试生成	第2章
第5、6周	根据有穷状态模型设计测试	第3章
第7、8周	组合设计 复习，期中考试	第4章
第9周	测试充分性：控制流	第6章
第10周	测试充分性：数据流	第6章
第11、12周	测试充分性：程序变异	第7章
第13、14周	特殊专题，如实时系统测试和安全性测试	另一卷
第15、16周	复习，研究工作汇报	
第17周	期末考试	

典型的本科生软件测试课程

我们计划一学期的本科生软件测试课程为3个学分，每周2次课，每次课50分钟，总共有17周的时间用于上课、考试和项目汇报。这门课程每周有2小时的实习，要求学生3~4人组成一个小组共同完成一个实习项目，最后提交一份研究报告和一个测试工具原型。每两周对学生进行一次测验，需在4~6小时之内完成。

表3是推荐的一个评价计划。精心设计的测验是本课程的重要内容。每次测验向学生提供一个采用测试工具完成测试任务的机会。例如，某个测验的目的可能就是让学生了解或熟悉测试执行工具JUnit或Web服务性能测试工具JMeter。教师可以根据前面完成的教学内容来设计测验内容。在学校的测试实验室中，有大量的商用和开源测试工具供学生使用。

表3　推荐的本科生、研究生软件测试课程评价要素

学生类别	评价要素	权　重	持续时间
本科生	期中考试	15分	90分钟
	期末考试	25分	120分钟
	课堂回答问题	10分	短时间
	测验	10分	10次测验
	实习项目	40分	一学期
研究生	期中考试	20分	90分钟
	期末考试	30分	120分钟
	测验	10分	5次测验
	研究/实习项目	40分	一学期

典型的研究生软件测试课程

我们计划一学期的研究生软件测试课程为3个学分。学生学习这门课时不必先修上面介绍的本科阶段的软件测试课程。除了考试之外，还要求学生阅读最新的研究材料并作读书报告。通过不定期的测验来使学生掌握测试工具。

测试工具

有大量的商用、免费或开源工具可供使用。表4列出了一小部分这样的工具。

表4　本科生、研究生软件测试课程中典型的测试工具

目　的	工　具	来　源
组合设计	AETG	
代码覆盖度量	TestManager™	JUnit CodeTest Suds
缺陷跟踪	Bugzilla	FogBugz
GUI测试	WebCoder	JfcUnit
变异测试	muJava	Proteum
性能测试	Performance Tester	JMeter

与时俱进

我希望本书能够随着时间的推移不断地完善。本书涵盖技术的任何进展，任何新出现的测试技术都将出现在本书的后续版本中。由我本人发现的或由读者指出的任何错误都将得到纠正。鼓励读者访问以下网站，以便获取有关本书的最新信息：

www. pearsoned. co. in/adityapmathur

虽然本书涵盖了软件测试的重要知识，但由于篇幅的限制，仍有一些先进技术未被包含进来。我计划再编写一本书，涵盖这些先进的测试技术，供那些想了解更多软件测试知识的学生以及业界的专业人士使用。

现金奖励

以前，我对仔细阅读本书并指出书中错误的学生进行现金奖励。现在，仍然采用这种方法，力求不断提高本书的质量。

致谢

在本书的写作过程中，很多人给予了重要帮助。对于那些在书中本应列出而未列出的人名，我在此深表歉意，虽然这种遗漏纯属偶然。

首先感谢 Rich DeMillo，是他将我引入软件测试这个领域，并资助了我早期的研究工作。Rich 指定的文献对我获取软件测试知识、增强对软件测试的理解很有帮助。衷心感谢 Bob Horgan，是他影响了并支持着我对软件测试、软件可靠性以及代码覆盖重要性之间关系的理解。Bob 向我免费提供了测试工具 χSuds 的早期和后期版本。我一直认为 χSuds 是到目前为止可用的、最好的测试充分性评价与增强工具。诚挚地感谢 Ronnie Martin，是他花费了大量时间修改我的技术报告。

热情地感谢 Donald Knuth 以及他的团队，他们向我提供了 T_EX 中错误的详细信息，并与我共享了 T_EX 的早期版本。感谢 Hiralal Agrawal 耐心地回答我关于动态切片的问题。感谢 Farokh Bastani、Fevzi Belli、Jim Berger、蔡开元、Ram Chillarege、Sid Dalal、Raymond DeCarlo、Marcio Delamaro、Phyllis Frankl、Arif Ghafoor、Amrit Goel、Dick Hamlet、Mats Heimdahl、Michael A. Hennell、Bill Howden、Ashish Jain、Pankaj Jalote、Rick Karcick、Bogdan Korel、Richard Lipton、Yashwant Malaiya、Josè Maldonado、Simanta Mitra、John Musa、Jeff Offutt、Tom Ostrand、Amit Paradkar、Alberto Pasquini、Ray Paul、C. V. Ramamoorthy、Vernon Rego、Nozer Singpurwalla、Mary – Lou Soffa、Rajesh Subramanian、Kishor Trivedi、JeffereyVoas、Mladen Vouk、Elaine Weyuker、Lee White 和 Martin Woodward，与他们的讨论及其提供的建设性意见改正了我许多关于软件测试和可靠性（甚至日常生活）原本错误、愚蠢的想法。

感谢 Jim Mapel、Marc Loos 以及其他几位在 Boston Scientific 公司（其前身是 Guidant 公司）工作的工程师，通过他们，我接触到了为确保心脏医疗设备高度可靠的复杂测试过程。感谢 Klaus Diaconu、Mario Garzia、Abdelsalam Heddaya、Jawad Khaki、Nar Ganapathy、Adam Shapiro、Peter Shier、Robin Smith、Amitabh Srivastava 以及许多在微软 Widows 可靠性和设备驱动器团队的工程师，通过他们，我了解了微软为确保向全球成千上万用户提供高可靠的操作系统而进行

的错综复杂的测试过程以及采用到的工具。

感谢本书的匿名审阅者所付出的辛苦劳动以及提出的有益建议。感谢 Muhammad Naeem Ayyaz、Abdeslam En – Nouaary、Joao Cangussu 和 Eric Wong，他们修改过本书的早期书稿以及用于本科生和研究生教学的相关材料。课堂上老师和学生的反馈意见对本书的修改起了重要作用。

感谢 Emine Gokce Aydal、Christine Ayers、Jordan Fleming、Nwokedi Idika、K. Jayaram、Yuanlu Jiang、Ashish Kundu、Yu Lei、Jung-Chi Lin、Shuo Lu、Ammar Masood、Kevin McCarthy、Roman Joel Pacheco、Tu Peng、Van Phan、James Roberts、Chetak Sirsat、Kevin Smith、Travis Steel、Yunlin Xu、Il-Chul Yoon、Hiroshi Yamauchi 和 Brandon Wuest，他们仔细阅读了本书早期书稿的有关章节，发现并纠正了其中的错误。我不会忘记 David Boardman、João Cangussu、Mei-Hwa Chen、Byoungju Choi、Praerit Garg、Sudipto Ghosh、Neelam Gupta、Vivek Khandelwal、Edward Krauser、Saileshwar Krishnamurthy、Tsanchi Li、Pietro Michielan、Scott Miller、Manuela Schiona、Baskar Sridharan 和 Brandon Wuest 与我在软件测试研究中的耐心合作。

感谢 T. S. K. V. Iyer 教授，他总是不停地问我本书的写作是否完成，这成为了我写完本书的巨大动力。衷心感谢 Raymond Miller、Nancy Griffith、Bill Griffith 和 Pranas Zunde 在我刚到一个新国家时给予的热情欢迎和帮助。感谢 John Rice 和 Elias Houstis 在实验设施和设备方面提供的帮助，很怀念与他们在一起的精彩共事岁月。感谢 Susanne Hambrusch 和 Ahmed Sameh 帮我在普渡大学开设软件工程和软件测试课程。感谢普渡大学计算机系的教职员工帮我安装用于本科生和研究生教学的实验设备和软件。感谢 Patricia Minniear 的辛勤劳动，是她及时拷贝本书以便学生使用。

衷心感谢 S. Venkateswaran 教授、L. K. Maheshwari 教授以及位于 Pilani 的 BITS 计算机系的教职员工，他们在我学术访问期间为我营造了友好融洽的工作环境。感谢我亲爱的朋友 Mohan Lal 及其家人多年来给我提供的帮助，尤其是我在 Pilani 期间，在那里我完成了本书的部分章节。感谢 BITS 招待所（VFAST）的全体雇员，他们的热情和友好对本书的质量产生了积极影响。

感谢 Hanna Lena Kovenock 为本书封面设计所作的贡献，她花了大量时间反复设计本书封面的卡通图案，描述了动物世界中的“chair development”团队。Hanna 是个伟大的艺术家，能得到她的帮助，我很荣幸。

感谢我的朋友 Ranjit Gulrajani、Pundi Narasimhan 以及他们的家人多年来给我的精神支持。

感谢我的父母和我的兄弟姐妹，他们坚定的爱和支持才是完成本书写作的根本。感谢我的孩子 Gitanjali、Ravishankar 和女婿 Abhishek Gangwal，他们总是问“什么时候这本书才能印刷啊?”

感谢我的科利狗 Raja 和 Shaan，它们陪我度过了宝贵的休息时光。

最后，但是最重要的，我要将最衷心的感谢献给我的爱妻 Jyoti Iyer Mather，感谢她对我无以撼动的爱和支持。

Aditya P. Mathur

目 录

Foundations of Software Testing