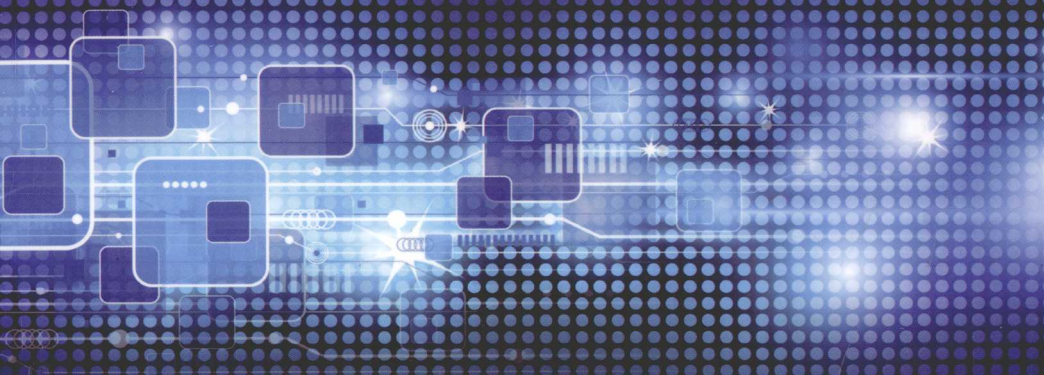


WILEY



Software Testing: Concepts and Operations

软件测试 概念与实践

[美] Ali Mili
Fairouz Tchier

颜炯

著
译



清华大学出版社

软件测试概念与实践

[美] Ali Mili 著
Fairouz Tchier

颜 炯 译

清华大学出版社

北 京

Ali Mili, Fairouz Tchier
Software Testing: Concepts and Operations
EISBN: 978-1-118-66287-8
Copyright © 2015 by John Wiley & Sons, Inc.
All Rights Reserved. This translation published under license.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2015-7461

Copies of this book sold without a Wiley sticker on the cover are unauthorized and illegal.

本书封面贴有 Wiley 公司防伪标签, 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

软件测试概念与实践 / (美) 阿里·迈利(Ali Mili) 等著; 颜炯 译.
—北京: 清华大学出版社, 2016
书名原文: Software Testing: Concepts and Operations
ISBN 978-7-302-44149-6

I. ①软… II. ①阿… ②颜 III. ①软件—测试 IV. ①TP311.5

中国版本图书馆 CIP 数据核字(2016)第 147625 号

责任编辑: 王 军 于 平
封面设计: 牛艳敏
版式设计: 思创景点
责任校对: 牛艳敏
责任印制: 刘海龙

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

印 装 者: 北京密云胶印厂

经 销: 全国新华书店

开 本: 148mm×210mm 印 张: 17 字 数: 397 千字

版 次: 2016 年 8 月第 1 版 印 次: 2016 年 8 月第 1 次印刷

印 数: 1~4200

定 价: 49.80 元

产品编号: 066971-01

译者序

随着软件的规模和复杂性不断增加，软件质量得到了越来越多的重视，软件产品测试得到了广泛关注，并且在软件开发成本(时间成本和人力成本)中占有重要地位。

本书主要分为五大部分，对软件测试进行了比较深入的研究。第 I 部分在软件工程的语境下概要介绍了软件测试，第 II 部分介绍了软件测试的数学基础，第 III 部分研究了软件测试数据生成，第 IV 部分讨论了软件测试预言设计、测试驱动设计、测试输出分析，第 V 部分探讨了软件测试管理方面的内容。

通过翻译本书，我们认为本书的最大特点如下：

- 与一般关注软件测试实践和介绍软件测试工具使用技巧的书籍不同，本书强调了软件测试的数学基础，包括软件规约、程序正确性和程序验证、软件可依赖性概念以及软件测试分类学。一般的软件测试书籍很少像本书这样讨论规约、验证、可依赖性等内容。确实，在没有深入理解软件规约的前提下，是不可能研究软件测试的，因为软件规约规定了被测试程序的功能属性，而这些功能属性也是测试预言设计的基础。
- 在研究基于测试数据生成准则的测试数据生成过程中，不仅讨论了能够直接从初始状态映射到终止状态的简

单程序，还研究了与状态历史相关的程序，也就是输出依赖于输入历史的程序。这个特点在模型驱动的软件开发和面向对象编程成为软件开发主流技术的当今，尤其具有实际价值。

- 系统地讨论了测试结果分析和软件测试度量。特别是对测试结果分析中与缺陷和失效有关的内容进行了相当详细的分析，系统地介绍了软件缺陷对软件测试的影响，这些内容对在软件工程语境下系统理解软件测试工作和管理软件测试工作提供了有力支持。

在翻译本书的过程中，我们感到作者对软件测试理论在指导软件测试实践方面的认识极为深刻，特别是，作者的讨论从未偏离软件测试活动背后最为重要的概念，即软件规约和测试数据生成。作者不仅深入浅出地讲解了相关的理论基础，还强调了基于测试理论的系统化测试实践，包括测试数据生成、测试预言设计、测试驱动设计、测试输出分析、软件度量、软件测试工具以及软件产品线测试。我们相信读者在阅读本书之后一定会得到同样的结论。

本书所面向的主要读者是那些需要进一步系统化地深入掌握测试理论的开发人员和软件测试人员，也适合系统学习测试理论和系统实践的软件工程专业学生。为了阅读本书，读者应该具备一些离散数学方面的基础知识。通过掌握本书内容，有助于读者将测试实践活动系统化、规范化，进而能够对各项测试技术问题应付自如。

在本书的翻译过程中，我们本着对读者认真负责的精神，力求做到技术内涵的准确无误和专业术语的规范统一，也力求做到翻译的准确性和灵活性的有效结合。

本书由颜炯翻译，Be Flying 工作室负责人肖国尊负责本书翻译质量和进度的控制与管理。由于我们的水平有限，错误与不到位之处在所难免。敬请广大读者提供反馈意见，读者可以将意见 e-mail 至 215565222@qq.com，我们会仔细阅读读者发来的每一封邮件，以求进一步提高今后译著的质量。

译者

前 言

软件工程是唯一一种产品测试成为主要技术，同时产品测试还得到了开发机构关注，并成为一项具有重要成本因素的工程原则。这种现状的产生来源于以下几点原因：

- 软件测试获得了极大关注的首要原因是软件产品的规模和复杂性，其导致软件产品设计成为一种高风险且容易出错的工作。
- 第二个原因是软件产品缺少一种标准开发过程，这就意味着产品质量无法通过过程控制进行保障，因此只能通过产品控制来保证产品质量。
- 第三个原因是缺少能够通过静态产品分析来保证产品质量并能适应产品规模变化的实用方法，因此只能通过动态方法来保证产品质量。
- 其他原因还包括：缺少一种通用的重用规则，缺少适用于不同产品规模且能够保证产品正确性的开发方法，以及在产品开发、维护、演化过程中无处不在的规约变更，等等。

本书的主题是研究软件测试；与当前其他同一主题的众多书籍相比，本书在以下几个方面具有突出的特点：

- 将软件测试视为软件质量保障的有机组成部分。我们将软件测试作为软件质量保障总体策略中的一个组成部分，而软件质量保障总体策略还包含了多项其他技术。边际效用递减规律(The law of diminishing returns)支持使用多种不同的互补性技术，而且需要将不同技术应用于能够得到最大投入回报的场合。这样，软件测试才能在一个包含了其他方法、更为广泛的语境下得到更深入的研究，而不仅仅是作为一个孤立的技术集进行研究。
- 将软件测试视为静态分析的一种补充技术。在软件工程专业发展早期，针对有效性、可伸缩性、易用性等方面内容，人们对软件测试和静态程序分析的优势进行了丰富多彩的讨论。我们的立场是：某种技术在某些场合下能够发挥最大的效能，而在其他场合下可能就没有那么有效了；另一种常见的情况是，当我们感觉某种技术很难应用的时候，实际上往往不是因为技术本身存在问题，而是因为基于错误的规约类型使用了这项技术。为了保证产品正确性，不应该执着于选择某一种规约。实际上，我们应该将一种复杂的规约分解为不同的组成部分，然后将不同的组成部分映射到最为合适的技术。本书的第 6 章讨论了这个问题。
- 将软件测试视为一个系统化的逐步完成的过程。早期的软件测试被视为一种证明程序缺陷存在的方法，而不是一种证明程序缺陷不存在的方法。实际上，这种看法是不恰当的，因为测试可以用于多种目的，这一点我们将在第 7 章讨论。无论这种看法是否合适，它都带来了两种后果：第一，人们认为软件测试的唯一目的是暴露缺陷、诊断程序和去除缺陷；第二，结果是，人们认为测

试仅仅是生成测试数据的过程，特别是生成那些能够最大可能地暴露程序缺陷的测试数据。与这些看法不同的是，我们认为测试是一个多步骤的过程，包括识别与分析测试目标、测试数据生成、测试预言设计、测试驱动程序设计、测试部署、测试输出分析。我们将用不同的章节分别讲解这些内容。

- 将软件测试视为一个形式化或可形式化的过程。对被测试的软件产品或软件产品规约而言，测试所需的分析工作量比较小，因此测试往往被视为一项可以随意执行的工作，并且无须考虑形式化问题。但是，我们却认为测试应该与静态程序分析一样严格。为有效地进行测试，测试人员必须具备软件规约、程序正确性、相对正确性(relative correctness)、缺陷的含义、缺陷去除等知识。这些内容将在第 6 章详细讨论。
- 将软件测试视为一个面向目标的活动。我们认为，软件测试不仅是查找和去除缺陷，还可能具有更广泛的目标，包括诸如估计缺陷密度、估计可靠性、可靠性认证(certification)等。这些内容将在第 7 章详细讨论。

本书的内容来自软件测试及质量保障课程的讲义，因此主要用于课堂使用。当然，本书也可以满足软件工程师的实际工作需要和软件工程研究人员的研究需求。本书分为五个主要部分，每个部分包括了 3~4 章内容，全书共计 16 章。

- 第 I 部分在软件工程的语境下比较宽泛地介绍软件测试，并且将探讨软件测试希望达到或确定的质量目标，还会探讨软件测试的生命周期。
- 第 II 部分介绍软件测试的数学基础，包括软件规约、程序正确性和程序验证、软件可依赖性概念以及软件测试

分类学。一般的软件测试书籍很少像本书这样讨论规约、验证、可依赖性等内容。本书这样做是基于以下原因：

- 首先，我们相信，在没有深入理解软件规约的前提下，是不可能研究软件测试的，因为软件规约规定了被测试程序的功能属性，而这些功能属性也是测试预言设计的基础。
- 其次，当我们在产品认证(product certification)或者验收测试(acceptance testing)的语境下执行程序测试时，最受关注的是被测试的程序是否正确，而我们必须理解“正确”到底是指什么。
- 第三，如果打算协同使用动态程序测试和静态程序分析，并且希望获得比单独用一种方法更为全面的结论，那么就必须在同一个数学模型下使用这些方法。
- 第四，从一个程序中去除缺陷的行为是测试的核心任务，这个行为只能通过定义相对正确性(relative correctness)的属性来建立模型。相对正确性是指当程序中的缺陷被去除后，程序将更正确。因此，相对正确性只能在理解了(绝对)正确性属性的前提下进行定义。

软件测试技术分类学是指将软件测试技术根据一组准则进行分类，特别是软件测试的目的：在指导软件测试的过程中，识别不同的软件测试目的是很重要的，不同的软件测试目的将影响测试生命周期的全部阶段，包括测试数据生成、测试预言设计，乃至测试部署和测试输出分析。

- 第III部分研究软件测试过程中最为吸引研究人员和测试人员的步骤，这个步骤常常被视为软件测试过程中唯一有意义的工作：测试数据生成。在这一部分中，我们将简要讨论测试数据生成的一些通用概念，然后研究测试

数据生成的两种通用准则，也就是功能准则(第 9 章)和结构化准则(第 10 章)。我们在讨论测试数据生成时，不仅要考虑能够直接从初始状态映射到终止状态的简单程序，还要考虑与状态历史相关的程序，也就是输出依赖于输入历史的程序。

- 第IV部分讨论软件测试生命周期中除了测试数据生成之外的其他阶段，包括测试预言设计、测试驱动程序设计、测试输出分析。测试预言(第 11 章)是从目标规约导出的，而目标规约是根据正确性的定义来定义的，并且取决于我们研究的是与状态无关的简单程序，还是存在内部状态的程序。测试驱动程序设计(第 12 章)取决于测试数据是脱机生成的，还是来自一个现有的介质，或者测试数据是根据某些概率法则而随机生成的。测试输出分析(第 13 章)则依赖于测试目的，分析内容可以是可靠性估计、可靠性验证、缺陷密度估计、以及产品验收等。
- 第V部分探讨软件测试管理方面的内容，包括软件度量(第 14 章)、软件测试工具(第 15 章)，以及软件产品线测试(第 16 章)。

在编辑本书内容材料的过程中，我们集中关注针对软件测试的重要方面的分析和建模，而不仅是研究和综述最新的软件测试研究内容，主要原因如下：

- 本书主要是面向教学的，而非一部研究专著。
- 在软件测试这一类充满活力的研究领域中，学生最好要掌握那些将在一段很长的时期内都能够发挥作用的基础概念，而不应该过分关注最新的技术。这是因为，从较长的时期看来，最新的技术总会过时。

针对应该为学生在较短时间内提供操作技能还是应该为学

生打下坚实的基础，并使他们在未来工作中具备适应能力的问题，在学术上存在着长期的争论。我们决定采取第二种策略。

致谢

我们在此向已经去世的 Lawrence Bernstein 教授致以特别的谢意，他邀请我们将本书纳入了他编纂的系列丛书，这套系列丛书十分出色。

感谢学生对我们的不断支持，在我们多个学期不断精化和优化讲义内容的过程中，他们对我们给予了极大的包容。我们还要感谢 Slim Frikha，这是一位来自法国巴黎理工学院(ParisTech)的实习生，他认真审查和评估了本书第 15 章实用的软件测试工具。在一定程度上，本书能够出版还需要感谢卡塔尔国家研究基金(Qatar National Research Fund)的资助，资助编号为 NPRP 04-1109-1-174。本书作者对本书的内容负完全责任，其中的内容不代表 QNRF 的官方观点。

Fairouz Tchier

Ali Mili

目 录

第 I 部分 软件测试介绍

第 1 章 软件工程：一种与众不同的原则	3
1.1 年轻而动荡的原则	3
1.2 压力下的行业	6
1.3 规模庞大的复杂产品	7
1.4 昂贵的产品	10
1.5 缺乏重用实践	12
1.6 容易出错的设计	13
1.7 矛盾的经济学	14
1.7.1 劳动力密集的产业	15
1.7.2 缺少自动化	15
1.7.3 质量控制手段有限	16
1.7.4 不平衡的生命周期成本	16
1.7.5 不平衡的维护成本	17
1.8 本章小结	19
1.9 参考书目注释	19
第 2 章 软件质量属性	21
2.1 功能属性	22

2.1.1	布尔属性	22
2.1.2	统计属性	23
2.2	操作属性	25
2.3	可用性属性	27
2.4	业务属性	28
2.5	结构属性	30
2.6	本章小结	32
2.7	练习	32
2.8	参考书目注释	33
第 3 章	软件测试生命周期	35
3.1	软件工程生命周期	35
3.2	软件测试生命周期	41
3.3	软件测试的 V 模型	48
3.4	本章小结	51
3.5	参考书目注释	51

第 II 部分 软件测试基础

第 4 章	软件规约	55
4.1	坚实可靠规约的原则	57
4.2	关系数学	58
4.2.1	集合与关系	58
4.2.2	关系操作	59
4.2.3	关系的属性	61
4.3	简单的输入输出程序	63
4.3.1	表达规约	63
4.3.2	对规约排序	67

4.3.3	规约生成	69
4.3.4	规约确认	75
4.4	可靠性和安全性	83
4.5	基于状态的系统	86
4.5.1	关系模型	86
4.5.2	公理表达	89
4.5.3	规约确认	98
4.6	本章小结	101
4.7	练习	101
4.8	问题	106
4.9	参考书目注释	109
第 5 章	程序正确性和验证	111
5.1	正确性的定义	113
5.2	正确性: 命题	117
5.2.1	正确性和精化	118
5.2.2	集合论的特征	120
5.2.3	解释	121
5.3	验证	124
5.3.1	公式样例	126
5.3.2	推理系统	128
5.3.3	用于解释的示例	133
5.4	本章小结	138
5.5	练习	139
5.6	问题	141
5.7	参考书目注释	141

第 6 章 失效、错误和缺陷	143
6.1 失效、错误和缺陷	143
6.2 缺陷和相对正确性	146
6.2.1 缺陷, 一个含糊的概念	146
6.2.2 相对正确性	148
6.3 视情况而定的缺陷和确定性缺陷	151
6.3.1 视情况而定的缺陷	151
6.3.2 单调的缺陷去除	155
6.3.3 一个单调缺陷去除的框架	161
6.3.4 确定性缺陷	162
6.4 缺陷管理	164
6.4.1 多道防线	164
6.4.2 混合确认	166
6.5 本章小结	171
6.6 练习	172
6.7 问题	174
6.8 参考书目注释	174
第 7 章 软件测试分类	175
7.1 联合测试存在的问题	175
7.2 分类方案	176
7.2.1 主要属性	178
7.2.2 次要属性	183
7.3 测试分类	190
7.3.1 单元级测试	190
7.3.2 系统级测试	192
7.4 练习	194

7.5 参考书目注释	195
------------------	-----

第III部分 测试数据生成

第8章 测试生成概念	199
------------------	-----

8.1 测试生成和目标属性	199
8.2 测试输出	203
8.3 测试生成需求	206
8.4 测试生成准则	211
8.5 基于经验的充分性评价	216
8.6 本章小结	222
8.7 练习	223
8.8 参考书目注释	225
8.9 附录: 变异程序	226

第9章 功能准则	229
----------------	-----

9.1 域划分	229
9.2 使用表格表达式生成测试数据	237
9.3 基于状态的系统的测试生成	244
9.4 随机测试数据生成	255
9.5 将测试数据选择比喻为一次旅行	260
9.6 本章小结	263
9.7 练习	264
9.8 参考书目注释	267

第10章 结构化准则	269
------------------	-----

10.1 路径和路径条件	270
10.1.1 执行路径	270
10.1.2 路径函数	273