

# 软件测试技术 实战汇编

王柳人 © 编著



清华大学出版社



# 软件测试技术 实战汇编

王柳人 © 编著



清华大学出版社  
北京

## 内 容 简 介

本书面向软件测试的实际应用,系统阐述了软件测试所涉及的基础理论、常用技术、过程管理和质量控制;重点讲解软件测试中的白盒测试技术、黑盒测试技术以及自动化测试技术;全面剖析了软件测试从单元测试阶段、集成测试阶段、系统测试阶段以及验收测试阶段等各个不同阶段比较成熟的技术及方法。

本书内容丰富,深入浅出,实用性强,可作为大中专院校计算机、软件工程、测试等相关专业师生自学的教材,也可作为有关软件测试的培训教材,对于从事软件测试工作的相关技术人员也有一定的参考价值。

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

软件测试技术及实战汇编/王柳人编著. —北京:清华大学出版社,2017  
ISBN 978-7-302-48286-4

I. ①软… II. ①王… III. ①软件—测试 IV. ①TP311.55

中国版本图书馆 CIP 数据核字(2017)第 209866 号

责任编辑:闫红梅 李 晔

封面设计:刘 键

责任校对:白 蕾

责任印制:李红英

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](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

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

印 装 者:清华大学印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:26 字 数:630千字

版 次:2017年9月第1版 印 次:2017年9月第1次印刷

印 数:1~2000

定 价:59.00元

产品编号:069304-01

# 前言

随着软件业的发展,测试的需求也越来越大,加上测试过程越来越智能化,软件测试由原来的人工测试向自动化测试方向发展,不仅可以大大地提高测试效率,还能使测试人员从反复枯燥的测试工作中解放出来,使得测试人员可以把精力放在系统测试的大局上。相信在未来的软件业,人工测试将逐渐消失,取而代之的是自动化测试,测试的智能化和测试的效率也将越来越高。软件产品的质量控制与质量管理正逐渐成为软件企业生存与发展的核心。

软件测试课程涉及软件测试的理论、方法、技术和工具,是一门实践性和技巧性很强的课程。本书从理论、方法、技术和工具方面进行了详尽的阐述。由基础理论到质量控制,到文档、工具的使用和案例分析,循序渐进地介绍软件测试涉及的各个方面的知识。从基本理论出发,逐步讲解软件测试的各类技术,随后从测试过程的角度介绍测试各个环节涉及的方法和技术,最后介绍有关软件质量及质量保证的知识。

全书共 18 章,两个附录,分 4 篇。

第一篇 软件测试基础。共 7 章,完整地介绍了软件测试基础知识概述、软件测试工作概述、黑盒测试技术、白盒测试技术、单元测试和集成测试、系统测试及验收测试。

第二篇 软件测试质量保证。共 4 章,介绍了软件过程能力评估和软件质量保证及软件缺陷和缺陷管理,软件配置管理等。

第三篇 软件测试工具。共 3 章,介绍了主流的软件测试工具,并进行对比。

第四篇 软件测试案例。共 4 章,介绍了 4 个经典案例,完整地介绍了测试的整个过程测试用例和测试方法的设计。

附录 A 软件测试文档。介绍测试文档的写作、软件测试所需的常用模板。

附录 B 软件测试习题及答案。

本书可作为高等院校计算机软件及测试相关专业软件测试课程教材,也可以作为软件测试人员和软件开发人员的参考书。

由于编写的时间仓促,教材中难免有漏洞和疏忽,敬请广大读者指正。

编者

2017 年 5 月

## 第 1 篇 软件测试基础

<b>第 1 章 软件测试概述</b> .....	3
1.1 软件测试的发展 .....	3
1.2 软件可靠性 .....	4
1.3 软件缺陷 .....	5
1.4 软件测试的定义及原则 .....	7
1.5 软件测试与软件开发 .....	9
1.5.1 需求分析阶段.....	9
1.5.2 设计阶段.....	9
1.5.3 实现阶段.....	9
1.5.4 测试阶段 .....	10
1.5.5 运行和维护阶段 .....	10
课后习题 .....	10
<b>第 2 章 软件测试过程与策略</b> .....	11
2.1 软件测试策略概述.....	11
2.2 软件测试分类.....	12
2.3 静态测试与动态测试.....	16
2.3.1 静态测试 .....	16
2.3.2 动态测试 .....	18
2.4 软件测试过程模型.....	21
2.4.1 V 模型 .....	21
2.4.2 W 模型 .....	22
2.4.3 H 模型 .....	22
2.4.4 X 模型 .....	23
2.5 测试用例的定义和特征.....	24
2.5.1 测试用例 .....	24
2.5.2 测试用例的特征 .....	25
2.5.3 测试用例设计原则 .....	25
2.5.4 测试用例的设计过程 .....	25
2.5.5 测试用例模板 .....	26

课后习题 .....	29
<b>第3章 黑盒测试 .....</b>	<b>30</b>
3.1 等价类划分法 .....	30
3.1.1 弱一般等价类测试 .....	32
3.1.2 强一般等价类测试 .....	33
3.1.3 弱健壮等价类测试 .....	33
3.1.4 强健壮等价类测试 .....	34
3.1.5 单元实践 .....	34
3.2 边界值测试 .....	40
3.2.1 边界值分析 .....	40
3.2.2 健壮性测试 .....	42
3.2.3 最坏情况测试 .....	42
3.2.4 单元实践 .....	43
3.2.5 随机测试 .....	51
3.2.6 边界值测试的指导方针 .....	53
3.3 决策表法 .....	53
3.3.1 决策表 .....	53
3.3.2 实例 .....	56
3.3.3 指导方针 .....	60
3.4 因果图法 .....	60
3.5 场景法 .....	63
3.6 正交实验法 .....	67
课后习题 .....	70
<b>第4章 白盒测试方法 .....</b>	<b>72</b>
4.1 逻辑覆盖法 .....	72
4.1.1 语句覆盖法 .....	73
4.1.2 判定覆盖法 .....	74
4.1.3 条件覆盖法 .....	74
4.1.4 条件/判定覆盖法 .....	74
4.1.5 组合覆盖法 .....	75
4.1.6 路径覆盖法 .....	75
4.1.7 实例分析 .....	75
4.2 基本路径法 .....	80
4.2.1 控制流图 .....	80
4.2.2 环形复杂度 .....	83
4.2.3 独立路径 .....	83
4.2.4 实例 .....	84

4.3	循环测试	86
4.4	Z 路径覆盖	87
4.5	程序插桩技术	88
4.6	域测试	90
4.7	符号测试	91
4.8	程序变异测试法	91
4.9	静态测试法	91
4.10	最少测试用例数计算	93
	课后习题	96
<b>第 5 章</b>	<b>单元测试和集成测试</b>	<b>98</b>
5.1	单元测试	98
5.1.1	单元测试的定义	98
5.1.2	单元测试的重要性	100
5.2	单元测试的内容与方法	101
5.2.1	单元测试的内容	101
5.2.2	单元测试的方法	103
5.3	单元测试的过程	104
5.3.1	计划单元测试	105
5.3.2	设计单元测试	106
5.3.3	执行单元测试	107
5.3.4	评估单元测试	108
5.3.5	实例分析	108
5.4	集成测试	111
5.4.1	集成测试概述	112
5.4.2	集成策略	115
5.4.3	面向对象的集成测试	116
5.4.4	集成测试流程	118
5.4.5	实例分析	120
	课后习题	122
<b>第 6 章</b>	<b>系统测试</b>	<b>123</b>
6.1	系统测试概述	123
6.1.1	系统测试基础	123
6.1.2	系统测试策略	124
6.2	功能测试	125
6.2.1	基本概念	125
6.2.2	分析方法	126
6.2.3	功能测试实例	126

6.3	性能测试 .....	127
6.3.1	性能测试基础 .....	127
6.3.2	性能测试实例 .....	129
6.4	本地化测试 .....	131
6.4.1	本地化测试基础 .....	131
6.4.2	关于 Java 用户界面本地化实例 .....	133
6.4.3	本地化测试的错误分类 .....	136
6.5	可用性测试 .....	137
6.5.1	可用性测试基础 .....	137
6.5.2	可用性测试方法 .....	138
6.5.3	传统 ATM 可用性测试报告 .....	139
6.6	配置测试 .....	145
6.6.1	配置测试基础 .....	145
6.6.2	3D 游戏的配置测试 .....	146
6.7	压力测试 .....	147
	课后习题 .....	149
<b>第 7 章 验收测试 .....</b>		<b>150</b>
7.1	验收测试概述 .....	150
7.1.1	基本概念 .....	150
7.1.2	验收测试的总体思路 .....	151
7.1.3	$\alpha$ 、 $\beta$ 测试简介 .....	152
7.2	验收测试的常用策略 .....	152
7.2.1	正式验收测试 .....	152
7.2.2	非正式验收测试 .....	153
7.2.3	$\beta$ 测试 .....	153
7.3	验收测试过程 .....	154
7.4	用户验收测试实施 .....	155
7.5	验收测试实例 .....	156
	课后习题 .....	157
<b>第 2 篇 软件质量保证</b>		
<b>第 8 章 软件过程能力评估 .....</b>		<b>161</b>
8.1	软件能力成熟度模型 .....	162
8.1.1	CMM/CMMI 的发展 .....	162
8.1.2	CMM/CMMI 应用领域 .....	162
8.1.3	CMM/CMMI 基本框架 .....	163
8.2	软件测试成熟度模型 .....	164



8.3	软件测试过程改进 .....	169
8.3.1	软件测试过程改进的概念 .....	169
8.3.2	组织的规划内容 .....	170
8.3.3	主要策略 .....	170
8.3.4	软件测试过程改进的具体方法 .....	171
	课后习题 .....	171
<b>第9章</b>	<b>软件缺陷及缺陷管理 .....</b>	<b>172</b>
9.1	软件缺陷 .....	172
9.1.1	软件缺陷的定义 .....	172
9.1.2	软件测试中的常用术语 .....	172
9.1.3	软件缺陷产生的原因 .....	173
9.1.4	软件缺陷的属性 .....	174
9.1.5	软件缺陷的类型 .....	176
9.1.6	缺陷严重程度 .....	178
9.1.7	软件缺陷修复的代价 .....	179
9.1.8	缺陷优先级 .....	180
9.1.9	软件缺陷的生命周期 .....	180
9.1.10	报告软件缺陷 .....	181
9.1.11	分离和再现软件缺陷 .....	182
9.2	测试总结报告 .....	182
9.3	软件缺陷跟踪管理 .....	183
9.4	缺陷管理工具 .....	184
9.4.1	常见缺陷管理工具 .....	185
9.4.2	Bugzilla 缺陷管理工具 .....	186
	课后习题 .....	207
<b>第10章</b>	<b>软件质量保证 .....</b>	<b>208</b>
10.1	软件质量 .....	208
10.1.1	软件质量的定义 .....	208
10.1.2	影响软件质量的因素 .....	209
10.1.3	软件质量评价应遵守的原则 .....	209
10.1.4	软件质量模型 .....	210
10.2	软件质量保证 .....	212
10.2.1	软件质量保证策略 .....	213
10.2.2	软件质量保证活动 .....	214
	课后习题 .....	223

<b>第 11 章 配置管理</b> .....	224
11.1 配置管理的基本概念 .....	224
11.2 配置管理活动 .....	225
11.2.1 识别配置项 .....	225
11.2.2 基于配置项版本控制 .....	226
11.2.3 变更控制 .....	226
11.2.4 发布配置状态报告 .....	227
11.2.5 配置审计 .....	227
11.2.6 发布管理 .....	227
11.3 项目经理的配置管理流程 .....	228
11.3.1 配备人员 .....	228
11.3.2 配置经理 .....	228
11.4 软件配置管理工具 .....	228
11.4.1 CVS .....	228
11.4.2 VSS .....	230
11.4.3 SVN .....	230
11.4.4 ClearCase .....	230
11.5 SVN 配置管理工具的使用 .....	230
11.5.1 服务器 SVN(Subversion)的安装和配置 .....	230
11.5.2 基本操作 .....	232
课后习题 .....	235

### 第 3 篇 软件测试工具

<b>第 12 章 自动化测试工具</b> .....	239
12.1 自动化测试 .....	239
12.2 自动化测试工具 .....	243
12.3 自动化测试工具详细介绍 .....	245
12.3.1 LoadRunner .....	245
12.3.2 QTP .....	257
12.3.3 WinRunnerMercury .....	268
12.3.4 Rational Robot .....	270
12.3.5 AdventNet QEngineAdventNet QEngine .....	271
12.3.6 SilkTest .....	271
12.3.7 QARun .....	273
12.3.8 TestPartner .....	275
课后习题 .....	276

第 13 章 测试框架 .....	277
13.1 JUnit 单元测试框架 .....	277
13.2 NUnit 单元测试框架 .....	287
课后习题 .....	293
第 14 章 其他测试工具 .....	294
14.1 一般测试工具 .....	294
14.2 主要厂商及产品介绍 .....	295
14.2.1 HP .....	295
14.2.2 IBM .....	295
14.2.3 Micro Focus .....	296
14.3 流行的开源测试工具 .....	297
14.3.1 单元测试工具 .....	297
14.3.2 功能测试工具 .....	299
14.3.3 性能测试工具 .....	299
14.3.4 测试管理工具 .....	300
课后习题 .....	302

## 第 4 篇 软件测试案例

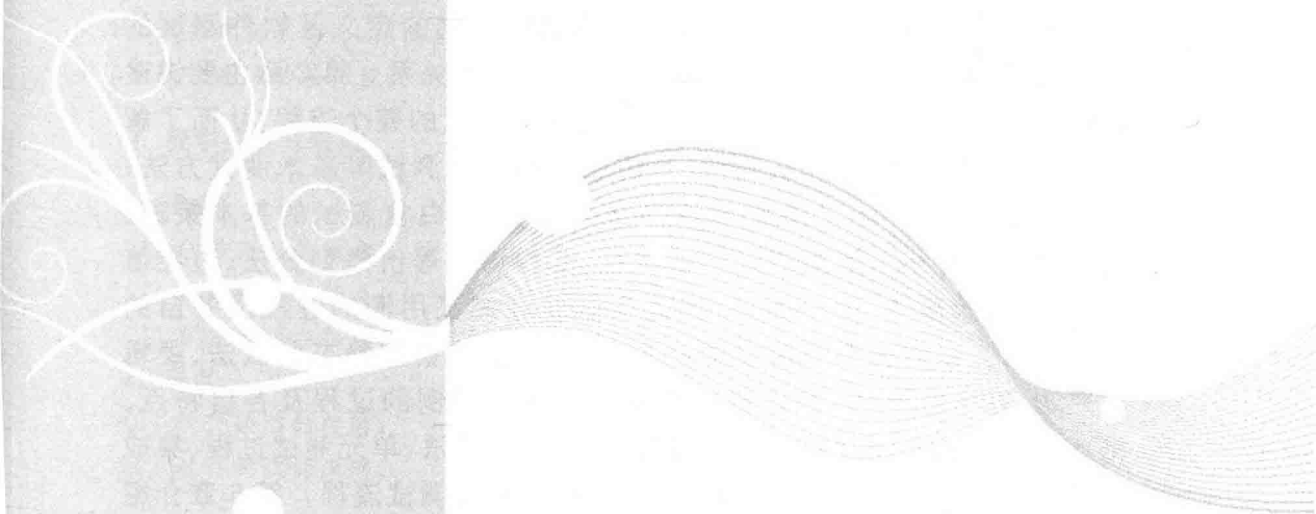
第 15 章 成人教务管理系统 .....	305
15.1 测试计划 .....	305
15.2 测试项目说明 .....	306
15.3 评价 .....	311
第 16 章 图书管理系统 .....	317
16.1 软件测试计划 .....	317
16.1.1 引言 .....	317
16.1.2 任务概述 .....	318
16.1.3 计划 .....	318
16.1.4 测试项目说明 .....	320
16.2 软件测试分析报告 .....	326
16.2.1 测试结果 .....	326
16.2.2 文档检查 .....	329
16.2.3 功能性测试定性评价 .....	329
16.2.4 非功能性评价 .....	330

第 17 章 人事档案管理系统 .....	331
17.1 实验环境 .....	331
17.2 实验任务 .....	331
17.3 实验内容与步骤 .....	331
第 18 章 嵌入式软件系统——俄罗斯方块 .....	335
18.1 项目简介 .....	335
18.2 单元测试设计 .....	336
18.2.1 静态测试：代码走查 .....	336
18.2.2 动态测试 .....	338
18.2.3 驱动模块 .....	340
18.2.4 单元测试的 Bug 列表 .....	341
附录 A 软件测试文档 .....	342
A.1 概述 .....	342
A.2 模板 .....	346
A.2.1 测试大纲模板 .....	346
A.2.2 软件测试计划模板 .....	349
A.2.3 测试任务说明书模板 .....	352
A.2.4 测试需求说明书模板 .....	355
A.2.5 单元测试模板 .....	358
A.2.6 代码检查模板 .....	360
A.2.7 程序错误报告模板 .....	361
A.2.8 程序设计模板 .....	362
A.2.9 测试用例模板 .....	365
A.2.10 软件测评模板 .....	366
A.2.11 功能测试模板 .....	367
A.2.12 性能测试模板 .....	369
A.2.13 可靠性测试模板 .....	370
A.2.14 集成测试模板 .....	371
A.2.15 系统测试模板 .....	373
A.2.16 验收测试模板 .....	374
A.2.17 测试分析报告模板 .....	376
A.2.18 测试总结模板 .....	378
A.2.19 Web 测试模板 .....	379
A.2.20 软件安全性测试模板 .....	381

附录 B 软件测试习题及答案 .....	383
第一部分 软件测试习题 .....	383
第二部分 习题答案 .....	391
参考文献 .....	402

第 1 篇

# 软件测试基础



本篇分为7章,第1章软件测试概述,第2章软件测试过程与策略,第3章黑盒测试方法,第4章白盒测试方法,第5章单元测试和集成测试,第6章系统测试,第7章验收测试。第1章主要讲述软件测试的发展历史及软件可靠性问题,重点讲述了软件缺陷的定义及软件测试的定义及原则,简要总结了软件测试与软件开发的关系。第2章主要讲述软件测试的复杂性和经济性;通过讲述软件测试的整个流程,从而了解单元测试、集成测试、确认测试、系统测试和验收测试等基本测试方法;通过比较分析,介绍了静态与动态测试、黑盒与白盒测试的基本策略。第3章首先介绍了黑盒测试方法概况,其次讲解等价类划分法、边界值分析法、决策表法、因果图法、场景法等设计测试用例的过程及各自的特点。第4章首先介绍白盒测试方法概况,其次讲解静态测试法、逻辑覆盖法、基本路径法、循环测试法等设计测试用例的过程及各自特点。第5章介绍单元测试基础、单元测试的内容与方法、单元测试过程、集成测试概况、集成策略、面向对象的集成测试、集成测试流程。第6章介绍系统测试概况,选取系统测试范畴中的功能测试、性能测试、本地化测试、可用性测试、配置测试进行详细讲解,介绍各种测试方法及其应用。第7章介绍验收测试概况、验收测试的常用策略、验收测试过程。

# 第 1 章

## 软件测试概述

### 1. 概述

本单元主要讲述软件测试的发展历史及软件可靠性问题,重点讲述软件缺陷的定义及软件测试的定义及原则,简要总结软件测试与软件开发的关系。通过本章的学习,读者能初步了解软件测试的基础理论及测试涉及的相关问题。

### 2. 教学重点与难点

#### 1) 重点

- (1) 软件测试的概念;
- (2) 软件缺陷的定义;
- (3) 软件测试在软件开发中的地位。

#### 2) 难点

- (1) 软件缺陷的理解;
- (2) 软件测试的原则。

## 1.1 软件测试的发展

软件测试是伴随着软件的产生而产生的。在早期的软件开发过程中,软件规模都很小、复杂程度低,软件开发的过程混乱无序、相当随意,测试的含义比较狭窄,开发人员将测试等同于“调试”,目的是纠正软件中已经知道的故障,常常由开发人员自己完成部分的工作,对测试的投入极少,测试介入软件开发过程的时间也较晚,常常是等到形成代码,甚至产品已经基本完成时才进行测试。

直到 1957 年,作为一种软件缺陷的测试活动,软件测试才开始与调试区别开来。人们在潜意识里仍将测试理解为“使自己确信产品能工作”,测试活动始终后于开发,主要依靠“错误推测(Error Guessing)”来寻找软件中的缺陷。因此,大量软件交付后,仍存在很多问题,软件产品的质量无法保证。

到了 20 世纪 80 年代初期,软件和 IT 行业进入了大发展阶段,软件趋向大型化、高复杂度,软件的质量越来越重要。这个时候,一些软件测试的基础理论和实用技术开始形成,人们为软件开发设计了各种流程和管理方法,软件开发的方式也逐渐由混乱无序的开发过程



过渡到结构化的开发过程,这一时期的工作以结构化分析与设计、结构化评审、结构化程序设计以及结构化测试为特征。人们还将“质量”的概念融入其中,软件测试定义发生了改变,测试不单纯是一个发现错误的过程,而且将测试作为软件质量保证(Software Quality Assurance SQA)的主要职能,包含软件质量评价的内容, Bill Hetzel 在《软件测试完全指南》(*Complete Guide of Software Testing*)一书中指出:“测试是以评价一个程序或者系统属性为目标的任何一种活动。测试是对软件质量的度量。”这个定义至今仍被引用。软件开发人员和测试人员开始坐在一起探讨软件工程和测试问题。1983年,IEEE提出的软件工程术语中给软件测试下的定义是:“使用人工或自动的手段来运行或测定某个软件系统的过程,其目的在于检验它是否满足规定的要求或弄清预期结果与实际结果之间的差别”。这个定义明确指出:软件测试的目的是为了检验软件系统是否满足需求。它再也不是一个一次性的,只是在开发后期实施的活动,而是与整个开发流程融合成一体。软件测试已成为一个专业,需要运用专门的方法和手段,需要专门人才和专家来承担。

20世纪90年代后期,自动化的测试工具开始盛行。随着计算机和软件技术的飞速发展,软件测试技术研究也取得了很大突破,人们关注有效的过程管理对于软件测试的重要性,形成了各种测试模型、测试能力成熟度模型。

21世纪初,软件测试的重要性越来越被人们接受,甚至出现了软件开发活动应该以测试为主导的思想,比如极限编程中倡导的测试驱动开发,随着软件测试分工的细化和成熟,软件企业注重自身核心竞争力的提升,促使大量的独立软件测试服务机构涌现出来,这些测试服务机构的运作机制日趋成熟,从单一的第三方认证评测逐步转向参与整个软件开发过程的测试服务,形成了一个成熟和广阔的市场区间。

## 1.2 软件可靠性

软件可靠性(software reliability)是软件产品在规定的条件下和规定的时间区间完成规定功能的能力。规定的条件是指直接与软件运行相关的使用该软件的计算机系统的状态和软件的输入条件,或统称为软件运行时的外部输入条件;规定的时间区间是指软件的实际运行时间区间;规定功能是指为提供给定的服务,软件产品所必须具备的功能。软件可靠性不但与软件存在的缺陷和(或)差错有关,而且与系统输入和系统使用有关。软件可靠性的概率度量称软件可靠度。

1983年美国IEEE计算机学会对“软件可靠性”做出了明确定义,此后该定义被美国标准化研究所接受为国家标准,1989年我国也接受该定义为国家标准。该定义包括两方面的含义:

- (1) 在规定的条件下,在规定的时间内,软件不引起系统失效的概率;
- (2) 在规定的周期内,在所述条件下程序执行所要求的功能的能力。

其中的概率是系统输入和系统使用的函数,也是软件中存在的故障的函数,系统输入将确定是否会遇到已存在的故障(如果故障存在)。