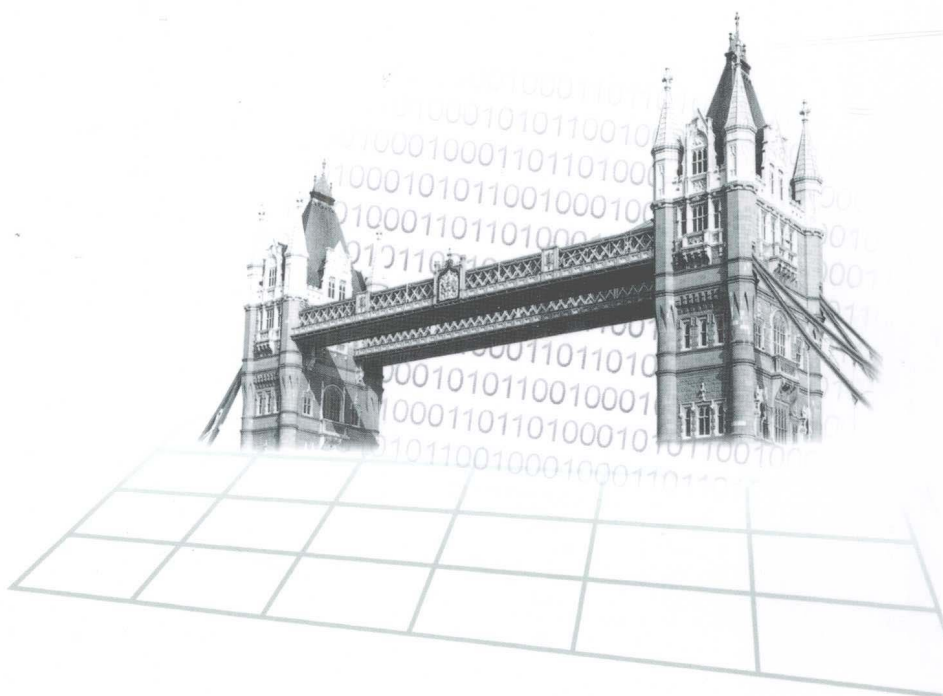


软件测试实用技术与常用模板

Software Test
with Cases and

Common Use

李龙 李向函 冯海宁 李向平 等编著



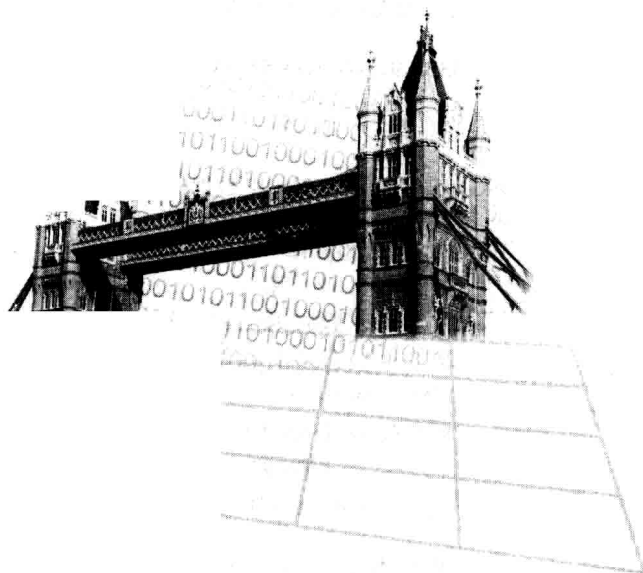
机械工业出版社
China Machine Press

开发人员专业技术丛书

软件测试实用技术与常用模板

Software Test
with Cases and Templates in Common Use

李龙 李向函 冯海宁 李向平 等编著



机械工业出版社
China Machine Press

本书从软件测试实用理论和实战技术的角度出发,以通俗易懂的语言讲述软件测试所需要的知识,并且以“手机信息管理系统模型”为测试软件平台,系统、全面地对测试实用理论和实训技术所需要的基本知识及高级技能进行介绍。本书突出实用性和实训的特点,是作者的软件测试经验和实践体会的总结。

本书主要讲解测试理论和各种测试方法,并且给出大量实际工作中用到的测试用例、测试方法和测试写作模板等内容。本书附赠的光盘中包含一些实用工具和文档。

本书可作为软件测试、软件开发等专业方向工程类硕士、项目管理类硕士和本科生的教材,也可作为软件测试理论与实践工作者进行研究、培训与应用实践的参考资料,同时还可供从事软件测试和项目决策的中高级管理人员阅读使用。

封底无防伪标均为盗版

版权所有,侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目(CIP)数据

软件测试实用技术与常用模板/李龙等编著. —北京:机械工业出版社, 2010. 10
(开发人员专业技术丛书)

ISBN 978-7-111-31950-4

I. 软… II. 李… 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(2010)第 185015 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:李 荣

北京京北印刷有限公司印刷

2010 年 10 月第 1 版第 1 次印刷

186mm × 240mm · 22 印张

标准书号: ISBN 978-7-111-31950-4

ISBN 978-7-89451-701-2 (光盘)

定价: 45.00 元(附光盘)

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

序

软件测试与测试技术在目前已引起业界的重视。国内有关这方面的教材和参考书至少有三十多个版本，纵观各个版本的测试书籍的内容，本人认为李龙等编著的《软件测试实用技术与常用模板》一书具有以下特点：

1) 内容通俗实用，详细、全面地叙述了人工测试技术和自动化测试技术。本书讲的是实用技术，读者读后就能依据本书所讲的测试技术进行测试工作。

2) 抓住了测试主题，重点突出，内容全面。重点抓住了软件测试的基本知识、常用技术、测试用例、文档写作模板、数值计算等典型程序的测试要点以及静态测试的要点。书中以“手机信息管理系统模型”为主线，对人工测试（包括：静态测试、单元测试、功能测试、系统测试、验收测试、回归测试、配置测试等）进行了详细的介绍，并给出了具体的操作过程和测试用例。此外，还利用免费的测试工具给出了自动化测试（包括：单元测试、功能测试、性能测试、回归测试等）的具体操作。本书摒弃了一般书籍理论篇幅较大、具体操作较少的写法，本书写的就是测试，写的就是测试技术，写的就是测试操作，实用性很强。

3) 测试文档写作模板可以说是本书的另一大亮点。书中给出了 20 个常用的模板写作内容，不仅为测试人员编写文档提供了参考依据，同时也可供未来修订测试规范、标准作参考。

4) 本书可操作性强，依据测试原理和手机信息管理系统模型，编写了大量的测试用例，其中有大部分测试用例可供各种类型的应用系统参考使用。

5) 提供的资料全面。在光盘中提供了测试所需的各种资料，正如编著者在前言中所说的：这是一本非常好的、实用的参考书。

6) 作为一名研究者、实践者和老师，我认为在目前已出版的软件测试方面的书籍中，它是一本比较好的书籍，详细给出了各种测试用例、可操作性的测试方法（人工测试和自动化测试）、各种完整的测试写作模板。读者认真研读完本书后，就能从事测试工作。本书对需要学习软件测试和软件开发的人员会有很大的帮助。

最后，愿本书在软件测试的百花园中，绽放出艳丽夺目的光彩。

黎连业

2010 年 6 月

前 言

软件测试已受到许多软件开发公司的重视，越来越多的软件开发人员投入到了软件测试的行业中。如何保证软件测试的质量？如何适应软件测试行业的技术需求？软件开发人员如何快速加入到测试行业？这些都是我们所关心的问题。为此，我们在中国信息大学信息工程学院的“软件测试技术研究组”的指导和帮助下，从实际的角度出发，编写了本书，供从事软件测试的技术人员阅读和使用。编者认为，这是一本非常好的、实用的参考书。

我们认为软件测试技术中有四个非常重要的内容，它们分别是：软件测试的基本知识、软件测试的核心技术、软件测试的实用技术以及软件测试文档写作模板。因此，我们围绕这四个重点，经过不断努力，完成了这本书。

本书由4个部分、15章内容组成，系统、全面地讨论了软件测试的思想、流程和方法。本书重实践、重应用，适合软件公司的测试经理、工程师和想进入软件测试行业的人员学习。

第一部分（第1章）主要介绍软件测试的基本概念和测试的相关知识，构造软件测试的基本框架。

第二部分（第2章至第4章）详细介绍软件测试的核心技术，强调白盒测试、黑盒测试的重点知识和相关技能以及测试用例的设计方法，让读者在一个较高的层次上全面理解软件测试的实际测试方法。

第三部分（第5章至第14章）具体介绍软件测试的实用技术，包括单元测试、功能测试、集成测试、性能测试、系统测试、验收测试、回归测试和配置测试的实用技术，以及其他的测试技术（包括可靠性测试、Web测试等13个测试技术）。对于这一部分内容，我们先讲解实用的理论技术和测试用例的编写方法、注意的要点，然后利用本书提供的“手机信息管理系统模型”软件进行实际的软件测试实训。我们努力做到：介绍的软件测试实用技术达到当前的实际应用水平，进行的实训操作完全符合正规软件测试公司的验收要求，力争让读者研读本书后能够进入测试行业，独立完成测试的相关工作。

第四部分（第15章）重点介绍软件测试过程中常用的文档写作模板。测试文档写作模板非常重要，是测试人员必须掌握的，因此，我们给出了常用的模板，希望对从事软件测试工作的人员有一定的帮助。

本书附赠一张光盘，光盘里的内容丰富、实用，包括：两个“手机信息管理系统模型”的源码和安装程序（一个是本书实训操作用到的软件系统，另一个作为读者自行测试的练习软件）、本书的课件（PPT）、本书用到的开源自动化测试工具安装包、软件测试常用工具汇总文档、测试常用术语的中英文对照和释义、各章节中用到的人工和自动化测试的实训测试用例的源码和测试脚本等。

本书主要由李龙、李向函、冯海宁、李向平、柴许伟、王传康、陈默涵、韩晨晨、孙平平

编著。

本书的编写经历了两个阶段。

第一阶段，编写本书实用测试理论：第5章、第14章和第15章由李龙编写；第4章和第13章由李向函编写；第1章和第8章由王传康编写；第2章和第6章由陈默涵编写；第10章和第11章由李向平编写；第3章由冯海宁编写；第7章由孙平平编写；第9章由韩晨晨编写；第12章由柴许伟编写。

第二阶段，编写本书实训操作技术：第6、7、8、10、12、13章这6章内容中的人工测试实训操作方法由李龙、冯海宁、李向平共同编写；第6、7、9、12、13章这5章内容中的自动化测试实训操作方法由李向函、柴许伟共同编写。本书中测试用到的两个“手机信息管理系统模型”分别由李龙和李向函个人开发。

同时，刘红力、郑恒健、温瑞、焦伟、范丰杰、程延斌等人帮助编写了本书的部分章节。马夺、王天恒、王占歌、王成刚、单江中、吴成超、袁立周、王君妍、王祥弟、孔繁奇、左家右、刘平、刘金辉、刘桐、刘鹏、孙若杰、孙晨、庄磊、曲瑞兴、朱文庆、朱会龙、张友强、张洁、张健、李建伟、苗龙、郑雪峰、施晓辉、徐庆磊、栗贝贝、秦唯一、谭尚成、魏升等人参与了本书内容的讨论和审定工作。在此特别感谢参加本书第一阶段和第二阶段写作的人员夜以继日的辛勤努力。

此外，我要特别感谢信息工程学院“软件测试技术研究组”的黎连业、马璐、赵克农、安灵慈、陈晨、吴丽丽老师的支持和鼎力帮助。同时，也特别感谢原洁（中软公司）、张晓冬（广东省气象局信息科 zhangxd115@live.cn）、张淑辉（中国武警工程学院）同志在百忙之中审阅了本书的全部书稿，并提出了宝贵的意见。

本书是软件测试专业的必修课，并且可以作为软件开发和软件工程类学科的选修课教材，可作为工程类硕士、项目管理类硕士和软件及软件测试本科生的教材，也可作为软件测试理论与实践工作者进行研究、培训与应用实践的参考资料，同时还可供那些从事软件测试和项目决策的中高级管理人员阅读使用。

由于作者水平有限，加上软件测试领域的发展日新月异，书中难免会有疏漏和不妥之处，敬请广大读者批评斧正。

李龙

2010年6月于北京

目 录

序 前言

第一部分 软件测试的基本知识

第 1 章 软件测试概述	1
1.1 软件的相关知识概述	1
1.1.1 软件的定义及分类方法	1
1.1.2 软件工程的主要内容	3
1.1.3 软件的生存期	3
1.2 软件测试的相关知识概述	4
1.2.1 软件测试的概念	4
1.2.2 软件测试的方法	4
1.2.3 软件测试的任务	6
1.3 软件测试的目的和原则	6
1.3.1 软件测试的目的	6
1.3.2 软件测试的原则	7
1.4 软件测试的流程	8
1.4.1 软件测试工作总体流程图	8
1.4.2 软件测试活动分布图	10
1.4.3 软件测试流程关系图	10
1.5 软件测试对组织结构和 人员的要求	10
1.5.1 对组织结构的要求	10
1.5.2 对人员的要求	10
1.6 自动化测试概述	14
1.6.1 自动化测试的特点	14
1.6.2 自动化测试的过程	16
1.6.3 自动化测试工具选择的原则	17
1.6.4 自动化测试技术	17
1.6.5 自动化测试技术的比较	19
1.6.6 自动化测试的级别	19

1.6.7 自动化测试方案选择需要 考虑的方面	20
1.6.8 自动化测试要点	21
1.6.9 自动化测试适用的测试阶段及其 执行的先决条件	21
1.6.10 自动化测试工作中的重点	22
1.6.11 自动化测试执行需要 解决的问题	24
1.6.12 自动化测试工具简介	25
1.7 软件测试的前景	27
习题	28

第二部分 软件测试的核心技术

第 2 章 白盒测试的实用技术	29
2.1 白盒测试概述	29
2.1.1 白盒测试的基本知识	29
2.1.2 白盒测试的原则	29
2.1.3 白盒测试的类别、依据和 流程	30
2.2 白盒测试的方法	31
2.2.1 代码检查	31
2.2.2 代码检查的方式	32
2.2.3 代码检查项目	33
2.2.4 静态结构分析	36
2.2.5 SQL 语句测试	36
2.2.6 代码检查的分析与评价	37
2.3 白盒测试的常用技术	37
2.3.1 逻辑覆盖法	37
2.3.2 插桩技术	39

2.3.3	基本路径测试法	40
2.3.4	域测试法	42
2.3.5	符号测试	42
2.3.6	Z 路径覆盖法	42
2.3.7	程序变异测试法	43
习题		43
第3章 黑盒测试的实用技术		44
3.1	黑盒测试的基本概念	44
3.1.1	黑盒测试的优点和缺点	44
3.1.2	对黑盒测试人员的要求	45
3.1.3	黑盒测试的意义	45
3.2	黑盒测试的方法	45
3.2.1	等价类划分	45
3.2.2	边界值分析法	47
3.2.3	因果图法	48
3.2.4	判定表驱动法	50
3.2.5	场景法	50
3.2.6	功能图法	51
3.2.7	错误推测法	51
3.2.8	正交试验设计法	52
3.3	黑盒测试的原则和策略	52
3.3.1	黑盒测试的原则	52
3.3.2	黑盒测试的策略	53
3.4	黑盒测试的步骤	53
3.4.1	测试计划	53
3.4.2	测试设计	53
3.4.3	测试执行的步骤	54
3.4.4	测试评估	54
习题		54
第4章 测试用例设计方法		55
4.1	测试用例编写概述	55
4.1.1	什么是测试用例	55
4.1.2	为什么要编写测试用例	55
4.1.3	测试用例主要包括哪些内容	55
4.1.4	设计测试用例所需的	

文档资料	56
4.2 测试用例的作用	56
4.2.1 实施测试指导的作用	56
4.2.2 指导测试数据规划的作用	56
4.2.3 指导脚本编写的作用	57
4.2.4 作为评判基准的作用	57
4.2.5 作为分析缺陷的基准的作用	57
4.3 测试用例的设计	57
4.3.1 白盒测试用例的设计	57
4.3.2 黑盒测试用例的设计	58
4.3.3 综合设计方法	58
4.3.4 测试用例设计的原则和 注意事项	58
4.3.5 测试用例的管理	61
4.4 测试用例主要内容的编写	61
4.4.1 测试目标的描述	61
4.4.2 测试环境的描述	61
4.4.3 输入数据/动作的编写	62
4.4.4 测试的步骤	62
4.4.5 测试预期的结果	63
4.4.6 测试脚本的编写	64
习题	64

第三部分 软件测试的实用技术

第5章 对本书测试模型的介绍	65
5.1 引言	65
5.1.1 编写目的	65
5.1.2 背景	66
5.1.3 定义	66
5.1.4 参考资料	67
5.2 程序系统的结构	67
5.3 对密码修改模块的设计说明	70
5.3.1 程序描述	70
5.3.2 功能	70
5.3.3 输入项	71
5.3.4 输出项	71

5.3.5 算法	71	6.8.3 自动化单元测试工作重点	84
5.3.6 流程逻辑	71	6.8.4 自动化单元测试案例	84
5.3.7 接口	72	习题	89
5.4 对详细查询功能模块的设计说明	72	第7章 功能测试的实用技术	90
5.4.1 程序描述	72	7.1 功能测试概述	90
5.4.2 功能	72	7.1.1 功能测试的基本概念	90
5.4.3 输入项	73	7.1.2 功能测试的基本要求	90
5.4.4 输出项	73	7.1.3 功能测试需要的参考文档	90
5.4.5 算法	73	7.2 功能测试的重点	91
5.4.6 流程逻辑	73	7.3 功能测试的常见方法	91
5.4.7 接口	73	7.3.1 安装测试	91
习题	73	7.3.2 卸载测试	94
第6章 单元测试的实用技术	74	7.3.3 运行与关闭测试	96
6.1 单元测试概述	74	7.3.4 菜单功能测试	98
6.1.1 单元测试的目的	74	7.3.5 浏览功能测试	98
6.1.2 单元测试所需文档资料	74	7.3.6 选择功能测试	98
6.2 单元测试的主要任务	75	7.3.7 搜查功能测试	99
6.2.1 程序语法检查	75	7.3.8 统计功能测试	99
6.2.2 程序逻辑检查	75	7.3.9 Web 功能测试	100
6.2.3 模块接口测试	75	7.3.10 翻页功能测试	101
6.2.4 局部数据结构测试	76	7.4 功能测试的常见流程	101
6.2.5 路径测试	76	7.5 功能测试的人工测试实训和	
6.2.6 边界条件测试	76	操作方法	102
6.2.7 错误处理测试	76	7.5.1 安装卸载测试	102
6.2.8 代码书写规范检查	77	7.5.2 手机信息管理系统模型的	
6.3 单元测试的步骤	77	登录测试	105
6.4 单元测试需要填写的表格	79	7.6 功能测试的自动化测试实训和	
6.5 单元测试提交的软件 Bug 清单	79	操作方法	108
6.6 单元测试的测试用例	79	7.6.1 自动化功能测试应该	
6.7 单元测试的人工测试实训和		何时开始	108
操作方法	81	7.6.2 自动化功能测试实例	108
6.8 单元测试的自动化测试实训和		习题	113
操作方法	83	第8章 集成测试的实用技术	114
6.8.1 选择适用的单元测试工具	83	8.1 集成测试概述	114
6.8.2 自动化单元测试中需要		8.1.1 集成测试基本情况	114
注意的问题	84		

8.1.2 集成测试重点考虑的内容	114	9.6 性能测试常用的测试用例	131
8.1.3 集成测试过程	115	9.7 性能测试的自动化测试实训和 操作方法	134
8.2 集成测试的方法	115	9.7.1 待测测试模型介绍	135
8.2.1 自底向上集成测试	116	9.7.2 使用 LoadRunner 进行负载/ 压力测试	136
8.2.2 自顶向下集成测试	117	习题	140
8.2.3 核心系统先行集成测试	117	第 10 章 系统测试的实用技术	141
8.2.4 高频集成测试	117	10.1 系统测试概述	141
8.2.5 桩驱动测试	118	10.1.1 系统测试的目的	141
8.3 集成测试阶段的工作	119	10.1.2 系统测试的意义	141
8.3.1 计划阶段	119	10.1.3 系统测试的主要内容	141
8.3.2 设计阶段	119	10.1.4 系统测试的目标	143
8.3.3 实现阶段	119	10.1.5 系统测试的测试类型	143
8.3.4 执行阶段	120	10.1.6 系统测试过程	144
8.4 集成测试的步骤和测试用例	120	10.1.7 系统测试策略	144
8.5 集成测试中的注意事项	120	10.1.8 系统测试环境	145
8.6 集成测试完成的标志	121	10.2 系统测试的方法	145
8.7 集成测试的人工测试实训和 操作方法	122	10.2.1 人工系统测试	145
习题	124	10.2.2 自动化工具系统测试	149
第 9 章 性能测试的实用技术	125	10.2.3 多任务测试	149
9.1 性能测试概述	125	10.2.4 临界测试	150
9.1.1 性能测试的目的	125	10.2.5 中断测试	150
9.1.2 性能测试的先决条件	126	10.2.6 系统测试策略	150
9.1.3 性能测试的准备工作	126	10.3 系统测试的过程	151
9.2 性能测试的基本要求	127	10.4 系统测试的结果分析	151
9.2.1 性能测试所需相关文档	127	10.5 系统测试的文档资料	154
9.2.2 对性能测试相关文档的 要求	127	10.6 系统测试的人工测试实训和 操作方法	154
9.2.3 对进行性能测试的 人员的要求	127	10.6.1 对登录界面的测试	154
9.3 性能测试的内容	127	10.6.2 对修改用户名和密码 模块的测试	155
9.3.1 性能测试的分类	128	10.6.3 手机信息管理系统 界面测试	158
9.3.2 性能测试的方法	128	10.6.4 易用性测试	161
9.3.3 性能测试的调优	129	习题	161
9.4 检查测试环境	130		
9.5 性能测试的执行流程	130		

第 11 章 验收测试的实用技术	162	12. 1. 9 自动化回归测试流程	178
11. 1 验收测试的条件、目的和 涉及的文档	162	12. 1. 10 自动化回归测试框架、作用和 框架的技术特点	178
11. 1. 1 验收测试的首要条件	162	12. 2 回归测试用例库的维护方法	180
11. 1. 2 验收测试的目的	163	12. 2. 1 删除过时的测试用例	180
11. 1. 3 验收测试的过程	163	12. 2. 2 改进不受控的测试用例	180
11. 1. 4 验收测试的需求文档	163	12. 2. 3 删除冗余的测试用例	180
11. 1. 5 验收测试的设计思路	164	12. 2. 4 增添新的测试用例	180
11. 2 应用系统验收测试	164	12. 3 回归测试的方法	180
11. 2. 1 验收工作	164	12. 3. 1 再测试全部用例	181
11. 2. 2 验收报告	164	12. 3. 2 基于风险进行测试	181
11. 2. 3 鉴定工作程序和文档资料	165	12. 3. 3 基于操作进行测试	181
11. 3 外包软件的验收测试	168	12. 3. 4 仅测试修改部分	181
11. 3. 1 外包软件验收测试的软件 配置和配置审核	168	12. 4 总结回归测试的结果	181
11. 3. 2 非正式验收测试	169	12. 5 回归测试自动化的问题	182
11. 3. 3 正式验收测试	170	12. 6 回归测试实践总结	182
11. 4 验收测试的组织过程	171	12. 7 回归测试文档	183
11. 4. 1 验收测试组织方案	171	12. 8 人工回归测试实训和操作方法	183
11. 4. 2 项目进度审核	171	12. 9 回归测试的自动化测试实训和 操作方法	185
11. 4. 3 验收测试计划	171	12. 9. 1 自动化回归测试工作中的 问题	185
11. 5 验收测试的内容	172	12. 9. 2 自动化回归测试解决方案	186
11. 5. 1 验收测试的要点	172	12. 9. 3 手机信息管理系统回归 测试案例	186
11. 5. 2 验收测试结果及缺陷分析	173	习题	187
11. 5. 3 项目验收情况汇总	173	第 13 章 配置测试的实用技术	188
习题	174	13. 1 配置测试概述	188
第 12 章 回归测试的实用技术	175	13. 1. 1 软件测试配置管理的概念	188
12. 1 回归测试概述	175	13. 1. 2 软件配置管理的定义	188
12. 1. 1 什么是回归测试	175	13. 1. 3 软件测试配置管理的任务	189
12. 1. 2 回归测试的目的	176	13. 1. 4 软件测试配置管理的目的	189
12. 1. 3 回归测试的范围	176	13. 1. 5 软件测试配置管理的 基本目标	189
12. 1. 4 回归测试的基本过程	176	13. 1. 6 软件测试配置管理的 执行约定	190
12. 1. 5 回归测试的策略	176		
12. 1. 6 回归测试人员	177		
12. 1. 7 选择有效的回归测试包	178		
12. 1. 8 人工回归测试流程	178		

13.1.7 软件测试配置管理的 执行能力	190	14.4.1 容错性测试的定义	209
13.1.8 软件测试配置管理组负责 协调的工作	191	14.4.2 容错性测试的内容	209
13.1.9 软件测试配置管理的流程	191	14.4.3 容错性测试需考虑的 特殊事项	209
13.1.10 软件测试配置管理的 关键活动	191	14.5 易用性测试	209
13.1.11 软件配置管理的要求	193	14.5.1 易用性测试的定义	210
13.1.12 配置测试的前提条件	194	14.5.2 易用性测试的内容	210
13.1.13 配置测试的范围	195	14.6 安全性测试	210
13.1.14 配置测试的目标	195	14.6.1 什么是软件安全	210
13.1.15 软件测试配置管理人员 工作的主要内容	195	14.6.2 安全性测试的内容和方法	210
13.2 配置测试技术	197	14.7 需求分析测试	211
13.2.1 配置测试设计	197	14.7.1 需求分析的定义和 测试任务	211
13.2.2 配置测试常用技术的使用	198	14.7.2 需求分析测试的内容	212
13.3 配置测试的主要内容	199	14.8 可靠性测试	212
13.3.1 硬件环境配置测试	199	14.8.1 可靠性测试概述	212
13.3.2 软件环境配置测试	201	14.8.2 软件可靠性测试中需 注意的问题	213
13.4 配置测试工作的文档	203	14.8.3 软件可靠性测试的流程	213
13.5 配置测试的人工测试实训和 操作方法	203	14.9 风险测试	216
习题	205	14.9.1 风险表现的形式	216
第14章 软件的其他测试技术	206	14.9.2 风险解决策略	216
14.1 可用性测试	206	14.9.3 风险测试步骤	216
14.1.1 可用性测试的定义	206	14.9.4 风险管理	217
14.1.2 可用性测试的方法	206	14.10 缺陷测试	217
14.1.3 可用性测试的目的	207	14.10.1 缺陷测试的定义 及其内容	217
14.2 压力测试	207	14.10.2 缺陷测试应注意的问题	218
14.2.1 压力测试的定义、特点和 核心原则	207	14.10.3 缺陷测试结论分析	218
14.2.2 压力测试的目标	208	14.11 Web 测试	218
14.3 确认测试	208	14.11.1 Web 的功能测试	219
14.3.1 确认测试的定义	208	14.11.2 Web 的性能测试	219
14.3.2 确认测试的内容	208	14.11.3 Web 的用户界面测试	220
14.4 容错性测试	209	14.11.4 Web 的兼容性测试	221
		14.11.5 Web 的安全性测试	221
		14.11.6 Web 的接口测试	221

14.12 接口测试	222	15.4 测试需求说明书写作模板	248
14.12.1 接口测试概述	222	15.5 单元测试写作模板	253
14.12.2 接口测试的内容	222	15.6 代码检查写作模板	257
14.12.3 接口测试的测试项目	223	15.7 程序错误报告写作模板	262
14.13 安装和反安装测试	223	15.8 程序设计写作模板	263
14.13.1 安装和反安装测试的 工作	223	15.9 测试用例写作模板	272
14.13.2 安装和反安装测试的 目标	224	15.10 软件测试写作模板	279
14.13.3 安装测试和反安装 测试的内容	224	15.11 功能测试写作模板	290
习题	224	15.12 性能测试写作模板	295
 第四部分 软件测试文档写作模板		15.13 可靠性测试写作模板	300
 第15章 软件测试所需的		15.14 集成测试写作模板	303
常用模板		15.15 系统测试写作模板	309
15.1 测试大纲写作模板		15.16 验收测试写作模板	313
15.2 测试计划写作模板		15.17 测试分析报告写作模板	320
15.3 测试任务说明书写作模板		15.18 测试总结写作模板	327
		15.19 Web 测试写作模板	329
		15.20 软件安全性测试写作模板	334
		习题	335
		参考文献	336

第一部分 软件测试的基本知识

第 1 章 软件测试概述

软件测试是软件开发中不可缺少的一个重要环节，随着软件的日益复杂，软件测试也变得越来越重要。掌握软件的基础知识和软件测试的概念（方法、目标和任务）是进行软件测试的基础。

本章主要讨论以下内容：

- 软件的相关知识概述。
- 软件测试的相关知识概述。
- 软件测试的目的和原则。
- 软件测试的流程。
- 对软件测试人员的要求。
- 自动化测试概述。
- 软件测试的前景。

1.1 软件的相关知识概述

从概念入手，才能少走弯路，才能对与此概念相关的问题有一个正确的认识，最终解决问题。软件测试的对象就是软件，为了进行软件测试，我们应了解什么是软件、软件的内容及其生存期。

1.1.1 软件的定义及分类方法

1. 软件的定义

软件是计算机系统中与硬件相互依存的一个部分，它是包括程序、数据及其相关文档的完整集合。其中，程序是按事先设计的功能和性能要求执行的指令序列；数据是使程序能正常操纵信息的数据结构；文档是与程序开发、维护和使用有关的图文材料。

软件具有以下 8 个特点：

- 1) 软件是一种逻辑实体，而不是具体的物理实体，因而它具有抽象性。
- 2) 软件的生产与硬件不同，它没有明显的制造过程。要提高软件的质量，必须在软件开发方面下工夫。
- 3) 在软件的运行和使用期间，不会出现硬件中所出现的机械磨损、老化问题。然而它存在退化问题，必须要对其进行多次修改与维护。
- 4) 软件的开发和运行常常受到计算机系统的制约，它对计算机系统有着不同程度的依赖性。

为了解除这种依赖性，在软件开发中提出了软件移植的问题。

5) 软件的开发至今尚未完全摆脱人工的开发方式。

6) 软件本身是复杂的。软件的复杂性可能来自它所反映的实际问题的复杂性，也可能来自程序逻辑结构的复杂性。

7) 软件成本相当昂贵。软件的研制工作需要投入大量的、复杂的、高强度的脑力劳动，它的成本是比较高的。

8) 相当多的软件工作涉及社会因素。许多软件的开发和运行涉及机构、体制及管理方式等问题，它们直接决定项目的成败。

2. 软件的分类方法

软件的分类方法有 4 种，下面分别介绍。

(1) 按软件的功能分类

按软件的功能可以将软件分为：系统软件、支撑软件、应用软件。

- 系统软件：与计算机硬件紧密配合在一起，使计算机系统各个部件、相关的软件和数据协调、高效地工作的软件。例如，操作系统、数据库管理系统、设备驱动程序以及通信处理程序等。
- 支撑软件：是协助用户开发软件的工具性软件，其中包括帮助程序人员开发软件产品的工具，以及帮助管理人员控制开发进程的工具。
- 应用软件：是在特定领域内开发、为特定目的服务的一类软件，其中包括为特定目的进行的数据采集、加工、存储和分析服务的资源管理软件。

(2) 按软件的服务对象的范围分类

按软件的服务对象的范围可以将软件分为：项目软件、产品软件。

- 项目软件：也称定制软件，是受某个特定客户（或少数客户）的委托，由一个或多个软件开发机构在合同的约束下开发出来的软件。例如，军用防空指挥系统、卫星控制系统等。
- 产品软件：是由软件开发机构开发出来直接提供给市场，或为成百上千个用户服务的软件。例如，文字处理软件、财务处理软件、人事管理软件等。

(3) 按开发软件所需要的人力、时间以及完成的源程序行数分类按开发软件所需的人力、时间以及完成的源程序行数可以将软件分为 6 种不同规模的软件，如表 1-1 所示。

表 1-1 软件规模的分类

类别	参加人员数	研制期限	产品规模（源程序行数）
微型	1	1~4 周	500 行
小型	1	1~6 月	1000~2000 行
中型	2~5	1~2 年	5000~50 000 行
大型	5~20	2~3 年	50 000~100 000 行
甚大型	100~1000	4~5 年	1 000 000 行
极大型	2000~5000	5~10 年	1 000 000~10 000 000 行

不论是规模大、时间长、很多人参加的软件项目，还是规模小、时间短、参加人员少的软件项目，其开发工作必须要有软件工程的知识作为指导，遵循一定的开发规范，其基本原则是一样

的，只是对软件工程技术依赖的程度不同而已。

(4) 按软件的工作方式分类

按软件的工作方式可以将软件分为：实时处理软件、分时软件、交互式软件、批处理软件。

- 实时处理软件：指在事件或数据产生时，立即予以处理，并及时反馈信号，控制需要监测和控制的过程的软件，主要包括数据采集、分析、输出三部分。
- 分时软件：允许多个联机用户同时使用计算机的软件。
- 交互式软件：能实现人机通信的软件。
- 批处理软件：把一组输入作业或一批数据以成批处理的方式一次运行，按顺序逐个处理完的软件。

1.1.2 软件工程的主要内容

软件工程研究的主要内容是软件开发技术和软件开发管理两个方面。在软件开发技术中，主要研究软件工程方法、软件工程过程、软件开发工具和环境。

1) 软件工程方法为软件开发提供了“如何做”的技术。它包括了多方面的任务，如项目计划与估算、软件系统需求分析、数据结构、系统总体结构的设计、算法的设计、编码、测试以及维护等。软件工程方法常采用某种特殊的语言或图形的表达方法，以及一套质量保证标准。

2) 软件工程过程是将软件工程的方法和工具综合起来以达到合理、及时地进行计算机软件开发的目的。软件工程过程定义了软件工程方法使用的顺序、要求交付的文档资料，以及为保证质量和协调变更所需要的管理。

3) 软件开发工具和环境为软件工程方法提供了自动的或半自动的软件支撑环境。目前，人们已经开发出了许多软件工具来支持上述的软件工程方法。而且已经有人把诸多软件工具集成起来，使得一种工具产生的信息可以被其他的工具所使用，从而建立起一种称为计算机辅助软件工程（CASE）的软件开发支撑系统。CASE 将各种软件工具、开发机器和一个存放开发过程信息的工程数据库组合起来形成一个软件工程环境。

1.1.3 软件的生存期

软件生存期概念的出现可以帮助我们较为全面地认识软件开发。在 1988 年制定和公布的国家标准 GB8566-88《计算机软件开发规范》中，将软件生存期划分为八个阶段，即：可行性研究和计划、需求分析、概要设计、详细设计、实现、组装测试、确认测试、使用和维护。该标准为每个阶段规定了任务、实施步骤、实施要求以及完成的标志。将软件生存期划分为这八个阶段大致符合并满足瀑布模型。

20 世纪 90 年代初提出了软件工程过程的概念。软件工程过程规定了获取、供应、开发、操作和维护软件时所实施的过程、活动和任务。其目的是为各种人员提供一个公共的框架，以使用相同的语言进行交流。这个框架由几个重要的过程组成，这些主要过程含有用来获取、供应、开发、操作和维护软件所用的基本的、一致的要求。该框架还可以用来控制和管理软件过程，各种组织和开发机构可以根据具体情况进行选择和剪裁，可在一个机构的内部或外部实施。

1995 年国际标准化组织在此基础上对生存期过程作了调整，公布了新的国际标准，即 ISO/

IEC 12207《信息技术——软件生存期过程》。该标准全面、系统地阐述了软件生存期的过程、活动和任务。该标准定义的 17 个过程分别属于：主要过程、支持过程和组织过程。如图 1-1 所示。

1.2 软件测试的相关知识概述

软件测试是软件质量保证的一种手段，软件测试的目的就是发现错误以及避免这些错误的发生。那么我们就应知道什么是软件测试、软件测试的对象是什么以及用什么方法测试。

1.2.1 软件测试的概念

软件测试是软件工程中的一个环节，是开发项目中的一部分。软件测试是有计划、有组织的，是保证软件质量的一种手段，它是软件工程中的一个非常重要的环节。因此，可以认为它是伴随软件工程的诞生而诞生的。伴随着软件复杂程度的增加、规模的增大，软件测试作为一种能够保证软件质量的有效手段，越来越受到人们的重视，软件测试的最终目的是使产品更加完美。

软件测试方法没有完全标准化和统一化，因为软件产业的产品各式各样，本书中介绍的软件测试方法可用于多数应用软件的测试。

软件测试不是万能的，不可能发现全部的软件缺陷，而且软件的功能和性能不是由测试决定的，软件测试是有局限性的。

软件测试是在完成程序设计阶段工作后，经程序编码员测试已初步奠定了基础，需要进一步完成的测试工作。我们都知道软件要投入运行必须确保其正确无误，这一点极其重要，所以软件要在投入运行前施行测试。未经周密测试而将软件贸然投入运行，将会造成难以想象的后果。

尽管各个程序在设计完毕以后都一一做过调试，但能否发挥整个系统的功能（例如，某一程序的运行与后续程序的运行是否矛盾，程序能否顺利持续运行）尚不清楚，整个系统的总测试要等全部程序设计结束且能连续测试时才可进行。

1.2.2 软件测试的方法

从软件测试的技术上说，目前的软件测试的方法中交流软件测试技术的多，而探讨软件测试方法的少。作为测试方法，如白盒测试方法、黑盒测试方法是探讨软件测试的宏观的方法论。从

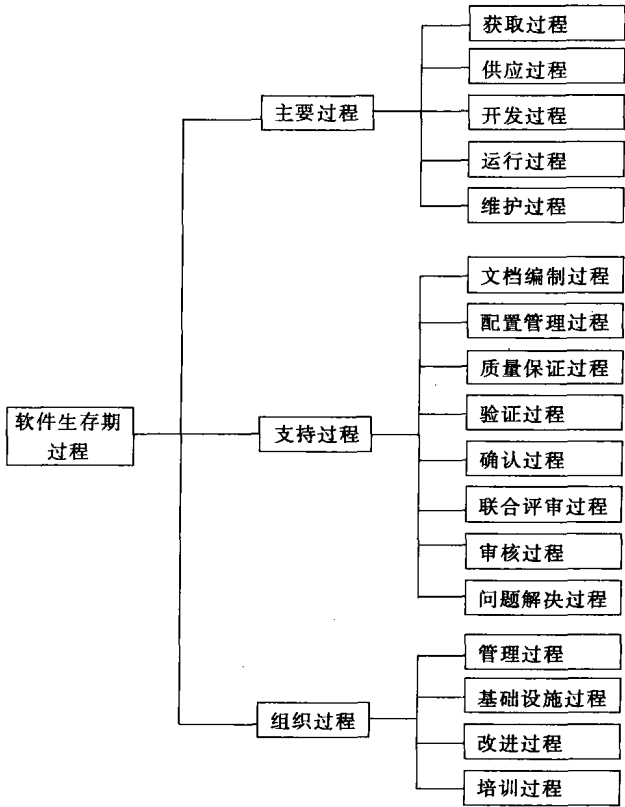


图 1-1 ISO/IEC 12207 信息技术——软件生存期过程