

<<软件测试教程>>

图书基本信息

书名：<<软件测试教程>>

13位ISBN编号：9787302307792

10位ISBN编号：7302307792

出版时间：杜文洁、景秀丽 清华大学出版社 (2013-01出版)

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件测试教程>>

书籍目录

第1部分软件测试入门 第1章软件测试概述 1.1软件测试背景 1.1.1软件缺陷 1.1.2软件测试技术的发展历史和现状 1.2软件测试的基本理论 1.2.1软件测试的定义和目标 1.2.2软件测试标准 1.2.3软件测试原则 1.2.4软件测试分类 1.3软件测试与软件开发 小结 习题 第2章软件测试方法 2.1静态测试 2.2动态测试 2.3黑盒测试 2.4白盒测试 小结 习题 第2部分软件测试提高 第3章软件测试过程与策略 3.1软件测试的复杂性与经济性分析 3.1.1软件测试的复杂性 3.1.2软件测试的经济性 3.1.3软件测试的充分性准则 3.1.4软件测试的误区 3.2软件测试流程 3.2.1软件开发的V模型 3.2.2单元测试 3.2.3集成测试 3.2.4确认测试 3.2.5系统测试 3.2.6验收测试 小结 习题 第4章软件测试环境搭建 4.1测试环境的要素 4.2搭建测试实验室步骤 4.3测试环境的管理与维护 4.4测试环境搭建举例 4.4.1JSP站点测试环境的搭建 4.4.2VMware workstation搭建NAT测试环境 小结 习题 第5章软件测试计划与文档 5.1测试计划的制订 5.1.1测试计划 5.1.2制订测试计划 5.2测试文档 5.2.1测试文档概述 5.2.2软件生存周期各阶段的测试任务与可交付的文档 5.3测试用例的设计 5.4测试总结报告 小结 习题 第6章软件自动化测试 6.1软件自动化测试概述 6.1.1自动化测试能做什么 6.1.2软件自动化测试的误区 6.1.3不适合自动化测试的情况 6.1.4国内软件自动化测试实施现状分析 6.1.5软件自动化测试的引入条件 6.2 自动化测试的策略与运用 6.2.1自动化测试策略 6.2.2自动测试的运用步骤 6.2.3测试工具的运用及作用 6.2.4自动化测试产生的问题 6.3常用自动化测试工具简介 6.3.1功能测试类 6.3.2性能 / 负载 / 压力测试类 6.3.3测试管理工具 小结 习题 第7章面向对象的软件测试 7.1面向对象软件测试概述 7.1.1面向对象软件测试的特殊性 7.1.2面向对象技术的特点及其对测试的影响 7.2面向对象测试的内容与范围 7.2.1面向对象分析的测试 7.2.2面向对象设计的测试 7.2.3面向对象编程的测试 7.2.4面向对象的单元测试 7.2.5面向对象的集成测试 7.2.6面向对象的系统测试 7.3面向对象软件测试技术与方法 7.3.1分析和设计模型测试技术 7.3.2类测试技术 7.3.3类层次结构测试技术 7.3.4对象交互测试技术 7.4面向对象软件测试用例设计 7.5面向对象测试基本步骤 7.6面向对象测试工具JUnit 小结 习题 第3部分软件测试实战 第8章黑盒测试实例设计 8.1等价类划分法 8.2边界值分析法 8.3决策表法 8.4因果图法 8.5黑盒测试综合用例 小结 习题 第9章 白盒测试实例设计 9.1逻辑覆盖测试 9.2路径分析测试 9.2.1控制流图 9.2.2独立路径测试 9.2.3 2路径覆盖测试 9.3其他白盒测试方法 9.3.1循环测试 9.3.2变异测试 9.3.3程序插装 9.4白盒测试综合用例 小结 习题 第10章Web网站测试 10.1 Web网站的测试 10.2功能测试 10.2.1页面内容测试 10.2.2页面链接测试 10.2.3表单测试 10.2.4Cookies测试

<<软件测试教程>>

章节摘录

版权页：插图：黑盒测试的基本观点：任何程序都可以看做是从输入定义域映射到输出值域的函数过程，被测程序被认为是一个打不开的黑盒子，黑盒中的内容（实现过程）完全不知道，只明确要做到什么。

黑盒测试作为软件功能的测试手段，是重要的测试方法。

它主要根据规格说明设计测试用例，并不涉及程序内部结构和内部特性，只依靠被测程序输入和输出之间的关系或程序的功能设计测试用例。

黑盒测试有两种基本方法，即通过测试和失败测试。

在进行通过测试时，实际上是确认软件能做什么，而不会去考验其能力如何，软件测试人员只运用最简单、最直观的测试用例。

在设计和执行测试用例时，总是先要进行通过测试，验证软件的基本功能是否都已实现。

在确信了软件正确运行之后，就可以采取各种手段通过搞垮软件来找出缺陷。

纯粹为了破坏软件而设计和执行的测试用例，被称为失败测试或迫使出错测试。

黑盒测试有两个显著的特点。

（1）黑盒测试不考虑软件的具体实现过程，在软件实现的过程发生变化时，测试用例仍然可以使用。

（2）黑盒测试用例的设计可以和软件实现同时进行，这样能够压缩总的开发时间。

在黑盒测试中，测试条件主要是基于程序或者系统的功能。

例如，测试人员需要有关输入数据的信息并观察输出数据，但是测试人员并不知道程序到底是如何工作的，这就好比一个人虽然会开车，但这个人并不知道汽车的内部工作方式。

在这里，运行一个程序并不需要理解其内部结构，测试人员只是根据产品应该实现的实际功能和已经定义好的产品规格，来验证产品所应该具有的功能是否能够实现，每个功能是否都能正常使用，是否满足用户的要求。

在测试时，测试人员将整个被测试的程序看成一个黑盒子，在完全不考虑程序或者系统的内部结构和内部特性的情况下，检查程序的功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收输入数据而产生正确的输出结果。

黑盒测试是以用户的观点，从输入数据与输出数据的对应关系出发进行测试的，它不涉及程序的内部结构。

很明显，如果外部特性本身有问题或规格说明书的规定有误，用黑盒测试方法是发现不了的。

黑盒测试方法着重测试软件的功能需求，是在程序接口上进行测试，主要是为了发现以下错误。

- （1）是否有不正确的功能，是否有遗漏的功能。
- （2）在接口上，是否能够正确地接收输入数据并产生正确的输出结果。
- （3）是否有数据结构错误或外部信息访问错误。
- （4）性能上是否能够满足要求。
- （5）是否有程序初始化和终止方面的错误。

黑盒测试不仅能够找到大多数其他测试方法无法发现的错误，而且一些外购软件、参数化软件包以及某些生成的软件，由于无法得到源程序，在一些情况下只能选择黑盒测试。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>