

<<软件测试>>

图书基本信息

书名：<<软件测试>>

13位ISBN编号：9787040244342

10位ISBN编号：7040244349

出版时间：2008-7

出版时间：赵瑞莲 高等教育出版社 (2008-07出版)

作者：赵瑞莲 编

页数：266

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>>

前言

随着计算机应用领域的日益扩大，计算机系统的规模和复杂性急剧增加，软件开发成本及由于软件故障而造成的经济损失也正在增加，软件质量问题已成为人们关注的焦点。

软件测试是对软件需求分析、设计规格说明和编码的最终评审，是保证软件质量的关键步骤，是为了发现故障而执行程序的过程。

软件系统规模和复杂性的增加，使得软件企业必须重视和加强软件测试工作。

据统计，在欧美的软件项目中，软件测试的工作量已占到项目总工作量的40%以上。

国外的成熟软件企业中，软件开发人员与测试人员的比例约为1:2；而国内软件企业中，平均8个软件开发工程师才配备1个软件测试工程师，比例严重失衡，软件测试人才紧缺的现状已经凸显出来。

为适应我国软件产业的飞速发展，培养更多、更专业的软件测试技术人员，我们在多年从事软件测试技术研究、实践和教学的基础上，结合当前软件测试技术的最新发展动态，编写了本书，旨在为普及我国的软件测试技术尽自己一份微薄的力量。

本书根据“软件测试”课程的教学需要，结合未来的职业需要和定位，综合考虑了软件故障分类、软件测试的停止标准、软件测试与开发的关系、测试方法的选择、测试环境的搭建、测试计划的制订、软件生存周期各阶段的测试任务和可交付的文档、主流测试工具的选择和使用、软件测试职业指导等方面，力求突出软件测试技术的基本概念和基本方法；同时注重理论与实践结合，重点讲解如何选择有效的软件测试方法，如何快速找出软件中存在的故障，如何清楚地报告发现的软件问题，以有助于读者掌握软件测试的基本技术并能应用到具体实践中。

此外，我们将多年在软件测试技术领域积累的资源，整理成附录，给出了主要的：IEEE / ANSI软件测试标准、一些著名的软件测试国际 / 国内会议和期刊，为读者提供了大量的软件测试信息资源。

全书共分为12章。

第1章介绍软件测试的发展历史和现状；第2章介绍软件测试的基本概念及涉及的关键问题；第3章介绍基本的软件测试策略；第4章介绍常用的黑盒测试方法；第5章介绍常用的白盒测试方法；第6章介绍单元测试、集成测试和系统测试的任务和测试用例设计思路；第7章介绍基本的验证方法和确认测试策略；第8章介绍软件生存周期各阶段的测试任务与可交付的文档；第9章简单介绍面向对象软件测试与传统软件测试的区别，以及面向对象软件的测试技术；第10章通过两个实例说明如何针对具体问题，选择测试方法和设计测试用例；第11章简单介绍目前市场上的主流测试工具，详细介绍了一款业界常用的白金测试工具——DevPartner；第12章简单介绍软件质量保证及优秀软件测试工程师应具备的素质。

本书的建议学时为56-64学时，其中实践教学建议为16学时左右，带星号(，-)的章节为可选内容。

具体学时分配见下表。

<<软件测试>>

内容概要

《普通高等教育“十一五”国家级规划教材：软件测试》系统地介绍了软件测试的基本概念、常用方法、基本流程、测试工具及测试过程中所涉及的文档资料，重点介绍了黑盒测试和白盒测试方法，并以实例说明了测试方法的选取及测试用例的设计思想，最后简单介绍了面向对象的软件测试、软件质量保证以及软件测试职业指导等。

《普通高等教育“十一五”国家级规划教材：软件测试》力求突出软件测试技术的基本概念和基本方法，注重理论与实践结合，重点讲解如何选择有效的软件测试方法，如何快速地找出软件中存在的故障，如何清楚地报告发现的软件问题，以有助于读者掌握软件测试的基本技术并能应用到具体实践中。

此外，书中还给出了主要的IEEE/ANSI软件测试标准、主要的软件测试国际会议和期刊，为读者提供了大量的软件测试信息资源。

《普通高等教育“十一五”国家级规划教材：软件测试》结构清晰，内容丰富，可作为普通高等院校、高职高专院校计算机相关专业“软件测试”课程的教学用书，也适合对软件测试感兴趣的读者使用，同时对希望增强知识并了解软件测试的开发人员、管理人员和质量保证人员也具有一定的指导意义。

<<软件测试>>

书籍目录

开始之前 第1章 绪论 1.1 计算机系统中软件的可靠性问题 1.2 软件测试与软件可靠性 1.3 软件测试的发展历史、现状和展望 小结 习题 第2章 软件测试基础 2.1 软件测试的基本概念 2.1.1 软件测试的目的 2.1.2 软件测试的定义 2.1.3 软件测试涉及的关键问题 2.1.4 软件测试与软件质量保证 2.2 软件故障 2.2.1 故障的定义 2.2.2 软件故障的分类 2.2.3 软件故障的修复代价 2.3 测试的复杂性与经济性 2.4 测试的充分性问题 2.5 测试原则 2.6 停止测试的标准 2.6.1 五类常用的停止测试标准 2.6.2 第四类停止测试标准 小结 习题 第3章 软件测试策略 3.1 软件开发过程及模型 3.1.1 软件开发过程 3.1.2 软件开发模型 3.2 软件测试过程 3.2.1 单元测试 3.2.2 集成测试 3.2.3 确认测试 3.2.4 系统测试 3.2.5 验收测试 3.3 软件测试与软件开发的关系 3.3.1 软件测试过程模型 3.3.2 软件测试生存周期 3.4 黑盒测试与白盒测试 3.4.1 黑盒测试 3.4.2 白盒测试 3.4.3 黑盒测试与白盒测试的比较 3.4.4 灰盒测试 3.5 静态测试与动态测试 3.6 验证测试与确认测试 3.7 回归测试和随机测试 3.7.1 回归测试 3.7.2 随机测试 3.8 软件测试方法的分类及选择 3.8.1 软件测试方法的分类 3.8.2 软件测试方法的选择 3.9 测试环境的搭建 小结 习题 第4章 黑盒测试 4.1 3个被测程序 4.1.1 三角形问题 4.1.2 NextDate函数 4.1.3 佣金问题 4.2 等价类划分测试 4.2.1 等价类划分 4.2.2 常见的等价类划分测试形式 4.2.3 等价类划分测试举例 4.2.4 等价类划分测试的指导方针 4.3 边界值分析 4.3.1 边界条件 4.3.2 次边界条件 4.3.3 边界值分析测试 4.3.4 稳健性测试 4.3.5 边界值分析举例 4.3.6 边界值分析的局限性 4.4 决策表测试 4.4.1 决策表 4.4.2 决策表在黑盒测试中的应用 · 4.4.3 决策表测试的指导方针 4.5 其他黑盒测试方法 4.5.1 因果图 4.5.2 特殊值测试 4.5.3 故障猜测法 4.6 黑盒测试的效率 小结 习题 第5章 白盒测试 5.1 程序控制流图 5.2 逻辑覆盖测试 5.2.1 语句覆盖 5.2.2 判定覆盖 5.2.3 条件覆盖 5.2.4 判定 / 条件覆盖 5.2.5 路径覆盖 5.3 路径分析与测试 5.3.1 程序路径表示 5.3.2 程序中线性独立路径数的计算 5.3.3 Z路径覆盖 5.3.4 独立路径测试及测试用例生成 5.4 数据流测试 5.4.1 数据流分析 5.4.2 定义和使用测试 5.5 符号测试 5.6 域测试策略 5.7 程序变异 5.7.1 程序强变异 5.7.2 程序弱变异 5.8 程序插装 小结 习题 第6章 软件测试技术 6.1 单元测试 6.1.1 单元测试的任务 6.1.2 单元测试的环境 6.1.3 单元测试的策略 6.1.4 单元测试用例的设计 6.2 集成测试 6.2.1 集成测试概述 6.2.2 集成测试与单元测试的区别 6.2.3 集成测试的策略 6.2.4 集成测试用例的设计 6.3 系统测试 6.3.1 系统测试的任务 第7章 验证测试和确认测试 第8章 测试计划与测试文档 第9章 面向对象的软件测试 第10章 软件测试实例分析 第11章 软件测试自动化和测试工具 第12章 软件质量保证及测试职业指导 附录A 软件工程和软件测试标准 附录B 软件测试会议和期刊 参考文献 参考网站

<<软件测试>>

章节摘录

版权页：插图：5.7.2程序弱变异 程序弱变异也是一种变异方法，其目标仍然是要查出某一类故障。其主要思想是，假设P是一个程序，C是P的一个简单组成部分。

若有一变异变换作用于C而生成 C' ，如果 P' 是含有 C' 的P的变异体，弱变异是要设计测试数据，使得当P在该测试数据下运行时，C被执行，且至少在一次执行中，C产生的值与 C' 不同。

由此可以看出，弱变异是一种只对被测程序进行测试的变异方法，它强调变动程序的组成部分，根据弱变异准则，只要能够设计出使C与 C' 产生不同值的测试数据，就可将被测程序在这些测试数据上运行，并不需要实际产生其变异体。

实现弱变异的关键是，确定程序P的组成部分以及与其相关的变换。

组成部分可以是程序中的计算结构、变量定义与引用、算术表达式、关系表达式及布尔表达式等。

其中一个组成部分可以是另一个组成部分的一部分。

最基本的程序组成部分有：变量引用、变量定义、算术表达式、关系表达式和布尔表达式。

其中布尔表达式可以包含关系表达式，关系表达式又可以包含算术表达式，布尔表达式、关系表达式和算术表达式可以包含变量定义和变量引用。

下面分别进行简单讨论。

(1) 变量定义 变量定义是指给变量赋以新值。

例如，在语句 $A=B$ 中，变量A被赋予新值B，该语句被称为变量定义组成部分。

变量定义的弱变异使被定义的变量变为另一变量。

如在语句 $A=B$ 中，使变量A变为另一程序变量D，则语句变为 $D=B$ 。

假设V是程序P中变量定义组成部分C的一个被定义变量， C' 是C的一个变换。

要使C和 C' 产生不同的值，必须设计测试数据，使程序执行到C前V的值与C执行后V的值不一致。

否则，即便C的定义是错误定义，因v的值没有变化，因而不容易查出这一错误。

(2) 变量引用 变量引用是指变量值的使用。

如对于语句： $A=B+A$ 变量B被引用，变量A则先被引用，继而又被定义。

再如语句 $IF(A>B)$ 中，A、B的值均被引用，这些语句是变量引用组成部分。

变量引用的弱变异使被引用的变量变为另一变量。

例如，在语句 $A=B+A$ 中，使B变为另一程序变量D，则语句变为 $A=D+A$ 。

假设V是程序P中变量引用部分C的一个被引用变量， C' 是C的一个变换。

要保证C和 C' 在测试数据上执行时产生的值不同，要求程序执行到C时，所有程序变量的值都与V的值不同。

这是因为，若组成部分C为 $A=V$ ，变换C为 $A=W$ 。

如果执行到C时，W的值与V相等，那么这组测试数据就查不出此变量的引用错误。

<<软件测试>>

编辑推荐

《普通高等教育"十一五"国家级规划教材:软件测试》结构清晰,内容丰富,可作为普通高等院校、高职高专院校计算机相关专业“软件测试”课程的教学用书,也适合对软件测试感兴趣的读者使用,同时对希望增强知识并了解软件测试的开发人员、管理人员和质量保证人员也具有一定的指导意义。

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>