

软件测试 Software Testing

01. 软件测试概论

程适

cheng@snnu.edu.cn

计算机科学学院

2016 年 12 月 23 日



陕西师范大学
SHAANXI NORMAL UNIVERSITY

关于本课程

■ 目标

- 理解软件测试
- 掌握常规测试方法，会撰写测试文档
- 了解软件测试工具，并学习使用

■ 课时及安排

- 理论：18 周 × 3 课时
- 实践：软件测试开源工具

关于本课程

■ 教材

- 软件测试方法和技术 朱少民
- 软件测试 Software Testing A Craftman's Approach Paul C. Jorgensen
- 软件测试经验与教训 Lessons Learned in Software Testing
- 有效软件测试 Effective Software Testing Elfriede Dustin
- Other Books + Web Resources

课程内容

- 先修课程：软件工程
 - 软件开发模型
 - 软件质量
- 软件测试的原理与方法
- 软件测试技术
 - 功能性测试与结构性测试
 - 集成测试与系统测试
- 软件测试实践
- 软件测试工具

软件测试

- 软件测试基础
- 软件测试引论
- 软件测试技术
- 软件测试实践

软件测试基础

- 软件测试概论
- 软件测试引论
- 软件测试基本概念
- 软件测试方法
- 软件测试依据和规范

Outline

- 软件测试概述
- 软件缺陷与软件质量
- 目的和原则
- 测试方法和测试用例
- 软件测试过程中表现出的特征

软件测试的发展历史

- 20 世纪 70 年代以前：Ad-hoc testing，与调试没有区分；
- 20 世纪 70 年代～ 80 年代中期：测试基础理论和实用技术形成，测试作为软件质量保证 (SQA) 的主要职能；
- 20 世纪 80 年代后期～ 90 年代中期：测试工具在质量和数量上不断增长，测试与 SQA(注重于过程和质量监督) 分离，注重于工具对测试效率的影响；
- 20 世纪 90 年代后期～目前：关于有效的过程管理对于软件测试的重要性，形成各种测试模型、测试能力成熟度模型。

软件测试概述

- 软件调试
- 专门的软件测试
- 专门的学科
- 与开发的融合

软件测试概述

■ 为什么软件测试发展比较缓慢？

- 软件开发环境、工具的改进
- 测试工具的局限性
- 企业对测试的投入

软件测试行业的现状和前景

- 国内软件测试仍处于相对初级的阶段
- 企业专职测试人员数量非常少，测试人员与开发人员的比例 1:3 1:5 左右
- 测试人员心态“浮躁”

软件测试的人员要求

■ 测试人员的素质要求

- 良好的心理素质
- 正确的测试态度
- 缜密的思维能力

软件测试的人员要求

- 测试人员的技能要求
 - 业务知识
 - 产品设计、产品架构知识
 - UML 语言
 - 测试工具
 - 开发工具
 - 用户心理学
 - 编程技能
 - ...

软件测试的目的

- 为什么要测试？
 - 对软件质量或可接受性做出判断
 - 发现问题
 - 人们很容易犯错误！

软件测试的目的

- “软件测试的目的是尽可能发现并改正被测试软件中的错误，提高软件的可靠性。”
- “测试的目的就是为了保证软件质量”
- IEEE 在 1983 年提出了软件测试的定义：“使用人工或自动手段来运行或测定某个系统的过程，其目的在于检验它是否满足规定的需求或是弄清预期结果与实际结果之间的差别。”

基本定义

- 1 错误 (error): mistake, bug
- 2 缺陷 (fault): 错误的结果, 过错缺陷、遗漏缺陷
- 3 失效 (failure): 缺陷执行时发生失效
- 4 事故 (incident):
- 5 测试 (test): 找出失效、演示正确的执行

研究内容

- 程序员犯了一个错误 (error/mistake)
- 这个错误在程序或软件中就表现为缺陷 (fault/defect)
- 运行带有缺陷的程序或软件，就有可能观察到失效 (failure)

测试用例 (test case)

■ 典型的测试用例信息：

- 测试用例 ID
- 目的
- 前提
- 输入
- 预期输出
- 后果
- 执行历史
- 日期 结果 版本 执行人

软件测试概述

- 思考：测试与调试的联系与区别？

软件缺陷与软件质量

- 软件缺陷
- 软件缺陷是对软件产品预期属性的偏离现象
 - 对产品规格说明的偏离
 - 对用户期望的偏离，即用户要求未体现在产品中（可能是规格说明有疏漏，也可能是实现中的问题）
- 软件缺陷不可能完全避免

软件缺陷

■ 根据缺陷后果的严重程度对缺陷的一种分类：

- 轻微
- 中等
- 使人不悦
- 影响使用
- 严重
- 非常严重
- 极为严重
- 无法忍受
- 灾难性
- 容易传染

软件质量

- 软件需求是衡量软件质量的基础
- 规定的标准是软件开发必须遵循的准则
- 如果已开发的软件已经满足了那些明文规定的需求，却没有满足隐含的需求，软件产品的质量仍然是有问题的

软件测试的心理学

- 软件测试就是证明软件不存在错误的过程
- 软件测试的目的在于证明软件能够正确完成其预定的功能
- 软件测试就是一个“软件做了其应该做的”信心的过程
- 软件测试是为发现错误而执行程序的过程

软件测试的心理学

- 软件测试更适宜被视为试图发现程序中错误的破坏性的过程。
- 一个成功的测试用例，通过诱发程序发生错误，可以再这个方向上促进软件质量的改进。

软件测试的经济学

- 即使是规模很小的程序，也不可能发现“所有”的错误
- 测试的策略

软件测试的目的和原则

■ 测试目的（J. Myers）

- 测试是程序执行的过程，目的在于发现错误（缺陷）
- 好的测试用例能有效地发现别的测试用例未发现的错误（缺陷）
- 成功的测试是发现了未曾发现的错误

软件测试的目的和原则

- 确保软件的功能符合用户的需求，把尽可能多的问题在发布或交付前发现并改正：
 - 确保软件完成了它所承诺或公布的功能
 - 确保软件满足性能的要求
 - 确保软件是健壮的和适应用户环境的

软件测试的目的和原则

■ 一些原则：

- 一个好的测试用例具有较高的发现过去未被发现过的错误的概率；
- 自己不能测试自己编写的程序；
- 对期望结果的描述是每个测试用例的必要组成部分；
- 杜绝不能重现或匆忙的测试；
- 既要编写使用有效输入条件的测试用例，也要编写使用非法输入条件的测试用例；
- 深入细致地审查测试结果

软件测试的目的和原则

■ 一些原则：

- 充分注意测试中的集群现象：测试后程序中残存的错误数目与该程序中已发现的错误数目成正比；
- 让最优秀的人员去完成测试；
- 保证软件的可测试性是软件设计的一个重要目标；
- 不要为了测试方便而修改程序；
- 测试工作必须在任务建立之初就确定目标。
- Good enough: 一种权衡投入/产出比的原则；
- 保证测试的覆盖程度，但穷举测试是不可能的；

软件测试的目的和原则

■ 一些原则：

- 所有的测试都应该追溯到用户需求；
- 越早测试越好，测试过程与开发过程应该是相结合的；
- 测试的规模由小而大，从单元测试到系统测试；
- 为了尽可能多的发现错误，应该由独立的第三方来测试；
- 既应该测试软件该做什么，也应该测试软件不该做什么

测试方法分类

- 根据软件测试的策略分类：
 - 黑盒测试与白盒测试（功能性测试和结构性测试）
 - 静态测试与动态测试
 - 手工测试与自动测试
 - ...

测试方法分类

■ 根据测试的阶段分类：

- 单元测试
- 集成测试
- 系统测试

软件测试过程中表现出的特征

- 软件测试具有一定的风险
- 软件缺陷的“寄生虫性”
- 软件测试的“杀虫剂现象”
- 软件测试的不修复原则
- Pareto principle（Pareto 原则，80/20 法则）

完全测试程序是不可能的

■ 原因：

- 1 输入量太大
- 2 输出结果太多
- 3 软件实现途径太多
- 4 软件说明书没有客观标准

软件缺陷的寄生虫性

- 找到的软件缺陷越多，就说明软件缺陷越多
- 原因：
 - 程序员的疲倦
 - 程序员往往犯同样的错误
 - 某些软件的缺陷其实是大灾难的征兆

软件测试的杀虫剂现象

- 软件测试越多，其免疫力越强的现象
- 克服方法：
 - 不断编写不同的新的测试程序
 - 对程序的不同部分进行测试

软件测试的不修复原则

- 并非所有软件缺陷都能修复
- 不需要修复软件缺陷的原因：
 - 没有足够的时间
 - 不算真正的软件缺陷
 - 修复的风险太大
 - 不值得修复

Pareto 原则，80/20 法则

- 测试发现的错误中的 80% 很可能起源于程序模块中的 20%。

小结

- 软件测试概述
- 软件缺陷与软件质量
- 目的和原则
- 测试方法和测试用例
- 软件测试过程中表现出的特征

小结

- 软件测试是为发现错误而执行程序的过程
- 一个好的测试用例具有较高的发现某个尚未发现的错误的可能性
- 一个成功的测试用例能够发现某个尚未发现的错误

致谢

谢谢，欢迎提问！