

软件测试

软件测试

- 软件测试是通过有限的测试用例集对程序行为进行动态验证的过程。
- 软件测试的目的不是为了证明程序是正确的，而是发现程序中的错误。

软件测试的原则

1. 尽早且不断地进行软件测试。
2. 测试用例应包含测试输入数据和预期输出结果。
3. 开发和测试队伍应分开建立。
4. 测试用例应包括合理和不合理的输入条件。
5. 注意测试中的群集现象。
6. 严格执行测试计划，排除随意性。
7. 对每个测试结果进行全面检查。
8. 修改程序后进行回归测试。
9. 保存测试计划和测试用例，为维护提供方便。

白盒测试

- 白盒测试（结构测试）基于程序内部逻辑结构设计测试用例。
- 方法包括逻辑覆盖测试、基本路径覆盖测试、数据流测试和循环测试。

黑盒测试

- 黑盒测试（行为测试）基于程序需求规约说明书设计测试用例。
- 方法包括等价类划分、边界值分析、比较测试、错误猜测和因果图。

测试策略

- 测试分为单元测试、集成测试、确认测试和系统测试。
- V模型描述了软件开发各阶段与测试策略之间的对应关系。

测试完成标准

- 无法绝对判定软件中是否还有错误，因此需要实用的测试完成标准。
- 标准包括基于统计的方法、使用指定测试用例设计方法、植入缺陷观察发现数量等。

补充内容

- Rice's theorem：所有非平凡的程序语义属性都是不可判定的。
- 形式化方法：用数学技术证明程序是绝对正确的方法。
- 检查和审查：由软件开发人员团队对代码或中间文档进行审查。
- 桌面检查和代码检查：详细的检查过程和注意事项。
- 走查：文件作者向一小组人员解释软件产品，参与者提出可能的错误和问题。