



SOFTWARE TESTING

苏临之

sulinzhi029@nwu.edu.cn



Debugging

- ❖ 调试指的是定位错误和修改错误的技术。某种程度上属于灰盒测试。
- ❖ 调试步骤
 - 第一步：定位错误（大概95%）。
 - 第二步：修改错误。



Five Debugging Techniques

- Brutal Force Debugging
- Inclusive Debugging
- Deductive Debugging
- Back-Tracking Debugging
- Test Debugging



A Mock Task

- 假设要求每人必须写出一篇5000字以上的论文，写完并检验后以后发给每人500元作为奖励。
- 但要求不得抄袭，且不得出现任何错别字，语句要通顺。否则重复率大于5%要倒扣2000元作为惩罚；出现错别字每个字扣50元，扣光为止；语句不通顺每句扣100元，扣光为止；字数不够者每缺一个字扣10元，扣光为止。
- 要想尽可能不受罚地完成该论文，应该怎么做？



A Mock Task

- 首先，在该论文完成后，必须先进行自我的检查。重复率和字数限制的检测可以通过第三方软件来检查，即通过个人检查即可。
- 错别字和语句问题很难通过第三方软件一一查清，这时候即使是写完后反复检查和阅读，还是有可能有漏网之鱼，受到损失可能性也很大。



Disadvantages of Self-Testing

- ❖ 同理，自我进行软件测试时不可避免的会遗漏一些，这属于正常现象，原因如下：
 - 杀虫剂现象，即缺陷越来越难找。
 - 自我的认知角度偏于单一，难以多角度思考问题。
 - 注意力长期集中会导致疲劳，进而产生烦躁情绪，反而会妨碍缺陷的发现。
 - 长期自我测试软件无法做到和新的思想沟通与学习。
- ❖ 因此，软件行业不提倡个人英雄主义，而需要注重团队合作（Teamwork）。人们很容易落入希望自己单独负责测试软件的陷阱中，因此一定要避免落入这样的思维陷阱。



Test of Independence

- Test of independence corresponds to an independent group of people who participate the testing process. They do not develop the software.
- 优势：1) 测试人员从中立的角度看待每个缺陷；2)测试人员完全没有偏见，即不带主观人至偏颇；3)测试人员对质量没有任何假设。
- 劣势：1) 与开发团队的隔离有时会导致过时的文档或版本引用；2) 独立的测试执行通常是最后一个阶段，在此过程中任何延迟都会影响受到版本或产品的发布；3) 开发人员可能对质量不负责任，因为他们可能认为独立测试团队的测试系统中有问题；4) 独立测试一定会受到通信阻碍。



Shared Testing

- ❖ 测试共享：在一定时间内互换测试任务，可理解为“你执行我的测试，我执行你的测试”。
- ❖ 这样可以让他他人想出其他的测试用例来测试，如在动态黑盒测试里至少可以帮助审查等价类划分和测试用例，为测试提供新思路。
- ❖ 请求协助寻找软件缺陷的最佳伙伴是产品支持或客户服务小组。这些人对软件缺陷非常敏感（特别是易用性缺陷）。



Bug Bash

- ❖ 缺陷轰炸指的是在一段时间（一般为若干个小时）内整个测试小组停下指定的常规测试任务，选择软件中某一区域，所有测试员集中测试某个区域或者某组特性，仿佛对这个区域进行“轰炸”。
- ❖ 选择轰炸的区域可能是软件缺陷聚集之处，看是否还有更多的潜伏问题；也可能是怀疑不存在软件缺陷的区域。



α -Testing (Alpha-Testing)

- ❖ α 测试是由一个用户在开发环境下进行的测试，也可以是公司内部的用户在模拟实际操作环境下进行的受控测试，不能由程序员或测试员完成。
- ❖ α 测试发现的错误，可以在测试现场立刻反馈给开发人员，由开发人员及时分析和处理。其目的是评价软件产品的功能、可使用性、可靠性、性能和支持。尤其注重产品的界面和特色。 α 测试可以从软件产品编码结束之后开始，或在模块测试完成后开始，也可以在确认测试过程中产品达到一定的稳定和可靠程度之后再开始。有关的手册（草稿）等应该在 α 测试前准备好。



β -Testing (Beta-Testing)

- ❖ β 测试是软件的多个用户在一个或多个用户的实际使用环境下进行的测试。开发者通常不在测试现场，不能由程序员或测试员完成。所以， β 测试是在开发者无法控制的环境下进行的软件现场应用。
- ❖ 在 β 测试中，由用户记下遇到的所有问题，包括真实的以及主管认定的，定期向开发者报告，开发者在综合用户的报告后，做出修改，最后将软件产品交付给全体用户使用。 β 测试着重于产品的支持性，包括文档、客户培训和支持产品的生产能力。只有当 α 测试达到一定的可靠程度后，才能开始 β 测试。由于 β 测试的主要目标是测试可支持性，所以 β 测试应该尽可能由主持产品发行的人员来管理。



β -Testing (Beta-Testing)

- ❖ β 测试实际上是一种描述外部测试的术语，让他人验证和确认软件的常用过程。
- ❖ 在 β 测试中，软件分发给选定的客户群，让他们在实际环境中使用软件，一般在产品开发周期行将结束时进行。这种测试可以是免费的，也有可能是要付费的。
- ❖ β 测试的目标：
 - 让媒体报道软件初期使用印象
 - 用户界面确认
 - 最后一步寻找软件缺陷



Problems to Plan the β -Testing

- ❖ 基础问题：谁是 β 测试者？不同类型的测试者会有不同的收获，比如他们是更关心底层操作呢还是更关心易用性呢？
- ❖ 怎样知道 β 测试者使用过软件呢？如1000个 β 测试者拿到软件一个月后报告没有发现问题，那么是没有缺陷，还是看到软件缺陷没有报告，抑或是记录报告的硬件遗失了？为了消除这些问题，一定要对 β 测试者进行跟踪。
- ❖ β 测试可以成为寻找配置和兼容性软件缺陷的好方法。如果 β 测试者挑选的明智，能够代表目标客户，那么就能帮大忙。



Problems to Plan a β -Testing

- ❖ 易用性测试是 β 测试能有所作为的另一个领域。条件是精心挑选参加者，在这里需要把有经验的用户和无经验的用户完美结合。
- ❖ 除去配置、兼容性和易用性测试之外， β 测试往往并不是很有效。由于一般参与者只能找出明显的大问题，而且测试时间点较晚，很有可能很多没有足够时间修复。因此不要试图依靠 β 测试来代替实际测试。



Differences

- ❖ 两者主要区别是测试的场所、环境和时序不同：
 - α 测试可以是指把用户请到开发方的场所来测试， β 测试是指在一个或多个用户的场所进行的测试。
 - α 测试可以的环境是受开发方控制的，用户的数量相对比较少，时间比较集中。而 β 测试的环境是不受开发方控制的，且用户数量相对比较多，时间不集中。
 - 一般 α 测试可以先于 β 测试执行。



Exercise

- 写出以下英文术语的中文名称。

1. functional testing

2. shared testing

3. alpha-testing

4. dynamic testing

5. specification

6. assertions

7. business logic layer

8. bottom-up testing

9. test-to-pass

10. code-based testing

11. threat modeling

12. compatibility testing

13. condition coverage

14. documentation testing

15. stub module

16. incremental testing

17. test-to-fail

18. top-down testing



Exercise

- 19. equivalence partitioning
- 20. software failure
- 21. system testing
- 22. load testing
- 23. stress testing
- 24. statement coverage
- 25. usability testing
- 26. configuration testing
- 27. driver module
- 28. cause-effect diagram
- 29. software defect
- 30. deductive debugging
- 31. test of independence
- 32. localization
- 33. static testing
- 34. black-box testing
- 35. inclusive constraint
- 36. multiple condition coverage
- 37. backward compatibility
- 38. spiral model
- 39. decision table
- 40. cyclomatic complexity
- 41. basic path set
- 42. acceptance testing



THANK YOU!