

题 目 如何保证软件测试的质量

学 号 14126106

学生姓名 雷文雅

班 级 三班

专业名称 软件工程

所在系（院） 软件学院

指导教师 田思源

2015 年 5 月 23 日

如何保证软件测试的质量

提高软件测试的质量对保证软件质量起到举足轻重的作用。说到如何保证软件测试的质量,不得不先解释和说明下软件测试和软件质量保证,最后来说明如何保证软件测试的质量。

1. 软件测试

软件测试是为使产品满足质量要求所采取的作业技术和活动,它包括检验、纠正和反馈。比如软件测试进行检验发现不良品后将其剔除,然后将不良信息反馈给相关部门采取改善措施。因此软件测试的控制范围主要是在工厂内部,其目的是防止不合格品投入、转序、出厂。确保产品满足质量要求及只有合格品才能交付给客户。

在开发大型软件系统的漫长过程中,面对极其复杂的问题,人的主观认识不可能完全符合客观现实,与工程密切相关的各类人员之间的通信和配合也不可能完全符合客观现实,因此在软件生命周期的每个阶段都不可避免地会产生差错。如果在软件投入生产性运行之前,没有发现并纠正软件中的大部分差错,则这写差错迟早会在生产过程中暴露出来,那时不仅改正这些错误的代价更高,而且往往会造成恶劣的后果。测试的目的就是在软件投入生产性运行之前,尽可能多地发现软件中的错误。目前软件测试仍然是保证软件质量的关键步骤,它是对软件规格说明、设计和编码的最后复审。

软件测试是软件开发过程中最艰巨最繁重的任务,大型软件的测试应该分阶段地进行,通常至少分为单元测试、集成测试和验收测试 3 个基本阶段。

白盒测试和黑盒测试是软件测试的两类基本方法,通常,在测试过程的早期阶段主要使用白盒方法,而在测试过程的后期阶段主要使用黑盒方法。

仅就测试而言,它的目标是发现软件中的错误,但是发现错误并不是最终目的。软件工程的根本目标是开发出高质量的完全符合用户需要的软件,因此,通过测试发现错误之后还必须诊断并改正错误,这就是调试的目的。测试和调试是软件测试阶段的两个关系非常密切的过程,它们往往交替进行。在对测试结果进行收集和评价的时候,软件所达到的可靠性也开始明朗了。

2. 软件质量保证

软件质量保证是为满足顾客要求提供信任,即顾客确信你提供的产品能满足他的要求。软件质量保证的目的不是为了保证产品质量,保证产品质量是软件测试的任务。

软件质量就是“软件与明确地和隐含地定义的需求相一致的程度”。更具体地说,软件质量是软件与明确地叙述的功能和性能需求、文档中明确描述的开发标准以及任何专业开发的软件产品都应该具有的隐含特征相一致的程度。软件需求是软件质量的基础,与需求不一致就是质量不高。指定的开发标准定义了一组指导软件开发的准则,如果没有遵守这些准则,肯定会导致软件质量不高。通常,有一组没有显式描述的隐含需求。如果软件满足明确描述的需求,但却不满足隐含的需求,那么软件的质量仍然是值得怀疑的。

从管理角度看,影响软件质量的主要因素有三组:产品运行(正确性、健壮性、效率、完整性、可用性、风险)、产品修改(可理解性、可维修性、灵活性、可测试性)和产品转移(可移植性、可再用性、可运行性)。

软件质量保证的措施主要有:基于非执行的测试,基于执行的测试和程序的正确性证明。基于非执行的测试主要用来保证在编码之前各阶段产生的文档的质量;基于执行的测试需要

在程序编写出来之后进行，它是保证软件质量的最后一道防线；程序正确性证明使用数学方法严格验证程序是否对它的说明完全一致。

3.保证软件测试的质量

软件产品的开发过程大体分为需求分析、设计、编码和测试四个阶段，而最终的软件质量与这四个阶段的各自质量之间的关系应当是：

最终的软件质量 = 需求分析质量 && 设计质量 && 编码质量 && 测试质量

最终的质量来自于各阶段质量之“与”，而不是“或”的关系只要其中一个环节质量是差，则产品的整体质量都将是差。显而易见。每一个阶段的质量都起着决定性的作用。

随着软件开发规模的增大、复杂程度的增加，以寻找软件中的错误为目的的测试工作就显得更加困难。统计表明，开发规模较大的软件，有 40%以上的精力是耗费在测试上的，即使富有经验的程序员，也难免在编码中发生错误，何况有些错误在设计甚至分析阶段就早已埋下祸根，无论是早期潜伏下的错误或者编码中新引入的错误，若是不及时排除，轻者降低了软件的可靠性，重者导致整个系统的识别。为了尽可能多地找出程序中的错误，生产出高质量的软件产品，加强对测试工作的组织和管理就显得尤为重要。

测试应当包含更多的内涵，比如求证软件需求是用户所希望的就是其中的一种。对于验证的理解仍需要拿房屋的建造作为一个比方，以便加深理解。在房屋的建造过程中，当建筑材料到了工地以后，需要对其进行检验，以保证它的质量是合格的，否则不能用于建造。对应于软件开发，这个阶段就是单元测试。当软件工程师编写了代码以后如何保证代码的行为是其所希望的呢？那只能通过单元测试去验证。房子建造好了以后，还得对房子进行整体的验收以确保其最终是合格的。比如抽查墙壁所使用的水泥与沙的配比是合适的。虽然水泥和沙在进入工地时都经过了质检且是合格的，但在建造的过程中需要按一定的比例混合它们以作建筑粘合剂，而混合比例将确定粘合强度。在软件开发过程中，软件集成测试就如同房子在建造好了以后的验收。

从上面的比方能得出几个结论。第一，在软件开发过程中单元测试是必不可少的。它的缺少如同将没有检验过的建筑材料用于建造一样。第二，单元测试应当在集成测试之前完成。有的项目在一开始时并没有单元测试流程，但后来发现需要增加这个环节，于是出现了集成测试完成了以后，再进行单元测试这种情形。这种情形还是有点怪怪的，这如同房子已造好了，再将墙打掉去检查里面的砖是否是好的一样。“将墙打掉检查砖”这种行为的勇气虽然可佳，但是如果尽早地在项目中部署单元测试就能避免这种怪现象的发生。集成（包括开发集成和系统集成）测试在软件行业被广泛采用以保证软件质量，但单元测试对于软件质量保证的重要性在整个行业还缺乏广泛的、深刻的认识，其更多地被当作是负担而不是一种有效的质量保证手段。

总之，软件测试是从大量的测试用例中选择有限的测试用例，发现软件中的大部分缺陷的一种技术。它是目前用来验证软件是否能够完成所期望的功能的唯一有效方法。软件测试通常要在不同层次上执行分为三大阶段：单元测试、集成测试、系统测试。

单元测试用于验证独立软件模块的功能与《详细设计说明书》的符合程度。软件模块可以是独立子程序，也可以是由紧密相关的数个单元组成的较大构件。单元测试一般需要对被测代码进行访问和借助测试工具的支持，并且可能需要被测代码编程人员的介入。

集成测试在单元测试的基础上，检验系统各模块间的交互性与《概要设计说明书》的符合程度。通常采用自顶向下（需要提供桩模块）或自底向上（不需要提供桩模块）的集成方法，用于传统的、分级的结构化软件系统。现代集成测试策略更多是结构驱动的，这意味着对软件模块或子系统的集成是基于确定的功能线程，因此集成测试是一个连续活动，在

每一阶段测试人员必须抽象出低一级的情况并集中于正在处理的这一级的状况。

系统测试 检验整个系统是否满足《需求规格说明书》所提出的所有需求。它需要将系统与功能性系统需求进行比较，非功能性系统需求指系统的安全性、速率、精确性、可靠性等。系统与其它软件、应用程序、硬件设备或操作环境的外部接口评估也在系统测试中进行。系统测试的类别有：功能测试；性能测试（包括负载测试、压力测试等）；外部接口测试；人机界面测试；安全性测试；可靠性测试（包括强度测试、恢复测试等）。

作好测试工作，首先可以制定合适的测试过程规范；制定合理的测试计划；设计合适的测试方案；编写的测试用例覆盖到所有的需求；对测试执行过程进行监控；使用工具管理测试发现的缺陷；对缺陷进行统计分析，指导过程改进。其次可以验证软件是否满足软件科研任务书、需求规格说明书和软件设计所规定的技术要求；再次，通过测试，人们可以尽早发现软件缺陷，并确保其得以修复；最后，完善的测试为软件可靠性与安全性评估提供了重要依据。