

## 1. 软件测试工程师的定义？

软件开发过程中的质量检测者和保障者，负责质量的把关工作。主要工作职责是：理解产品的功能要求，并对其进行测试，检查软件有没有错误，决定软件是否具有稳定性，写出相应的测试规范和测试用例。简而言之，软件测试工程师在一家软件企业中担当的是“质量管理”的角色，及时纠错及时改正，确保产品的正常运作。

## 2. 工作内容？

编写软件测试计划，设计软件测试脚本和用例，搭建软件测试用例环境； 2.执行软件项目测试，包括功能测试、性能测试、易用性测试等； 3.整理、分析、报告并追踪软件缺陷，并确认软件测试问题得以解决； 4.编写软件测试结果分析报告，预先评估项目的风险，编写其他相关文档； 5.结合研发软件产品项目情况，制定相应的软件、项目版本控制制度。

管理之路：测试主管 测试管理者 项目主管 产品发布主管

## 3. 测试工具

Test platform 软件测试平台，简称 TP。能够全面管理软件质量工作，具有高度集成性，能够完成多款其他各类相关质量管理工具集成在一起才能完成的软件质量管理工作。它集成了需求跟踪、静态测试、动态测试、测试人员管理、软件环境管理、测试计划管理、测试用例管理、缺陷管理、缺陷分析等软件质量相关流程！

AutoRunner 是国内第一款自动化测试工具，可以用来完成功能测试、回归测试、每日构建测试与自动回归测试等工作，是一款支持 IE 测试和 Windows native 测试的自动化测试工具。

TestCenter 是一款功能强大测试管理工具，实现测试用例的过程管理，对测试需求过程、测试用例设计过程、业务组件设计实现过程等整个测试过程进行管理。实现测试用例的标准化及每个测试人员都能够理解并使用标准化后的测试用例，降低了测试用例对个人的依赖；提供测试用例复用，以保护测试人员的资产；提供可伸缩的测试执行框架，提供自动测试支持；提供测试数据管理，帮助用户统一管理测试数据，降低测试数据和测试脚本之间的耦合度。

LoadRunner 是一种预测系统行为和性能的工业标准级负载测试工具。通过成千上万用户实施并发负载及实时性能监测的方式来确认和查找问题；它能对整个企业架构进行测试；通过这款工具，企业能最大限度地缩短测试时间，优化性能和加速应用系统的发布周期。

TestDirector 是全球最大的软件测试工具提供商 MI 公司生产的企业级测试管理工具，也是业界第一个基于 web 的测试管理工具；它可以在公司内部或外部进行全球范围内测试的管理。通过在一个整体的应用系统中集成了测试管理的各个部分，包括需求管理，测试计划，测试执行以及错误跟踪等功能，它极大地加速了测试过程。

#### 4. 测试用例

- 目的：统一测试用例编写的规范，以保证使用最有效的测试用例，保证测试质量。
- 范围：适用于公司对产品的业务流程、功能测试测试用例的编写。
- 业务流程测试用例编写原则：系统性、连贯性
- 测试用例设计的方法：等价类划分法、边界值分析法
- 测试用例设计的原则：全面性、正确性、符合正常业务惯例、仿真性、可操作性
- 测试用例编写格式细则：测试用例内容、测试用例表格格式
- 测试用例优先级：测试用例优先级 A 重要 B 比较重要 C 次重要 D 不重要 E 系统小单元、系统容错功能  
对 AB 应重点考虑 9.BUG 级别：

#### 5. 测试人员的个人素质

- 要提高自己的分层测试思维，在划分测试点时不仅能从业务的角度考虑，还要充分考虑数据的特殊性
- 应具有一定的市场意识和风险意识，能够站在不同的角度，尽可能地分析系统可能存在的风险场景
- 测试负责人制定测试计划粒度要小，周期要短；测试组成员对于自己的工作都有明确的目标和子目标；特别是存在交叉测试的模块，测试人员一定要责任明确，确保没有漏测的场景

- 一定要有自我推销意识，主动加强与开发人员、需求人员及客户的沟通
- 测试人员应像质量管理人员一样，在项目中起到过程监控点的作用。能够从全局角度反映项目存在的问题。
- 提高测试效率很重要的一点就是对可复用资产的充分利用；可复用资产中比较重要的一部分就是测试用例的复用；所以在以后的测试中一定要充分利用可复用的测试用例提高测试用例的编写效率。
- 经验的浪费是最大的浪费，测试人员一定要善意总结，把自己的测试经验记录下来，这样不仅能完善自己的测试思想还能为同事提供系统的学习经验。
- 软件测试涉及的心理学问题 1.程序测试的过程具有破坏性 2.程序员应避免测试自己的程序。

#### 6. 什么是兼容性测试？兼容性测试侧重哪些方面？

- (1) 兼容测试主要是检查软件在不同的硬件平台、软件平台是否可以正常的运行，即是通常说的软件的可移植性。
- (2) 兼容的类型，有平台的兼容，网络兼容，数据库兼容，以及数据格式的兼容。
- (3) 兼容的重点是，对兼容环境的分析。

#### 7. 单元测试的策略有哪些？

逻辑覆盖、循环覆盖、同行评审、桌前检查、代码走查、代码评审

#### 8. 测试的策略有哪些？

黑盒/白盒，静态/动态，手动/自动，冒烟测试，回归测试，公测（Beta 测试的策略）。

功能测试是测试的类型，黑盒测试是测试的手段；功能测试用的最多的是黑盒测试

- 按照测试方法划：

黑盒测试：已知产品的功能设计规格，可以通过测试证明每个实现了的功能是否符合要求。

白盒测试：已知产品的内部工作过程，可以进行测试证明每种内部操作是否符合设计规格要求，所有内部成分是否已通过检查。

灰盒测试：介于白盒和黑盒测试之间，关注输出对于输入的正确性，也关注程序的内部结构，但没有白盒测试那么详细完整。

- 按照测试方向划分

功能测试：能不能做

性能测试：能够做多好。分为 3 个小方向：压力测试：测试软件最大承载量，发现软件的性能瓶颈；负载测试：在峰值 80%~90% 负荷内能运行多久；并发测试：主要指测试多用户并发访问同一个应用、模块、数据时是否产生隐藏的并发问题，如内存泄漏、线程锁、资源争用问题，几乎所有的性能测试都会涉及并发测试。

安全测试：主要防黑客。

- 按照测试阶段划分

单元测试（模块测试）：是开发者编写的一小段代码，用于检验被测代码的一个很小的、很明确的功能是否正确。通常由程序员自己编写单元测试，执行单元测试。

集成测试：也叫组装测试、联合测试，两个已经测试过的单元组合成一个组件，并且测试它们之间的接口，最后将构成进程的所有模块一起测试

系统测试：将经过测试的子系统装配成一个完整系统来测试。是检查系统能否确实能提供系统方案说明书中指定功能的有效方法。目的是对最终软件系统进行全面测试，确保最终软件系统满足产品需求并且遵循系统设计

验收测试：是部署软件之前的最后一个测试操作，为了向用户表明系统能够按照预期的要求进行工作。验收测试的目的是确保软件准备就绪，并且可以让最终用户将其用于执行软件的既定功能和任务。

9. 测试计划工作的目的是什么？测试计划工作的内容都包括什么？其中哪些是最重要的？

- 可以明确测试任务和测试方法，保持测试实施过程的顺畅沟通，跟踪和控制测试进度，应对测试过程中的各种变更。
- 包含了产品概述、测试策略、测试方法、测试区域、测试配置、测试周期、测试资源、测试交流、风险评估等。
- 测试计划主要从宏观上规划测试活动的范围、方法和资源配置。最重要的是测试策略和测试方法（最好是能先评审）

#### 10. 你以前参加工作时的测试流程？

需求评审（开发人员、产品经理、测试人员、项目经理） --> 需求确定（出一份明确的需求文档） --> 开发设计文档（开发人员在开始写代码前就能输出设计文档） --> 想好测试策略，写出测试用例 --> 发给开发人员和测试经理查看（非正式的评审用例） --> 接到测试版本 --> 执行测试用例（中间可能会补充用例） --> 提交 bug（有些 bug 需要开发人员的确定（严重级别的，或突然发现的在测试用例范围之外的难以重现的问题点，有些可以直接录制进 TD） --> 开发人员修改（可以在测试过程中快速的修改） --> 回归测试（可能又会发现新问题，再按流程开始跑）。

#### 11. 软件的评审一般由哪些人参加？其目的是什么？

- 用户、客户或有关部门开发人员，测试人员，需求分析师
- 找出可能影响软件产品质量、开发过程、维护工作的适用性和环境方面的设计缺陷，并采取补救措施，以及找出在性能、安全性和经济方面的有效改进。

#### 12. 测试活动中，如果发现需求文档不完善或者不准确，怎么处理？

测试需求分析发现需求文档不完善或者不准确，应该立即和相关人员进行协调交流。

#### 13. 阶段评审与项目评审有什么区别？

阶段评审对项目各阶段评审：对阶段成果和工作；项目评审对项目总体评审：对工作和产品

14. 描述测试用例设计的完整过程？

需求分析+需求变更的维护工作； 根据需求得出测试需求； 设计测试方案，评审测试方案； 方案评审通过后，设计测试用例，再对测试用例进行评审；

15. 你认为做好测试计划工作的关键是什么？

软件测试计划是在软件测试工作正式实施之前明确测试的对象，并且通过对资源、时间、风险、测试范围和预算等方面的综合分析和规划，保证有效的实施软件测试； 做好测试计划工作的关键：目的，管理，规范

(1) .明确测试的目标，增强测试计划的实用性；

(2) .坚持 5W 规则，明确内容和过程； What（做什么（明确测试的范围和内容））， Why（为什么做（可以帮助测试团队理解测试的目的））， When（何时做（确定测试的开始和结束日期））， How（如何做（指出测试的方法和工具））， Where（在哪里（给出测试文档和软件的存放位置））

(3) .采用评审和更新机制，保证测试计划满足实际需求

(4) .分别创建测试计划与测试详细规格、测试用例

16. 做好测试用例工作的关键是什么？

需求和设计文档的理解程度，对系统的熟悉程度

17. 简述一下缺陷的生命周期？

一个软件缺陷被发现、报告到这个缺陷被修复、验证直至最后关闭的完整过程。

18. 软件的安全性应该从哪几个方面去测试？

1) 用户认证机制：数据证书、智能卡、双重认证、安全电子交易协议

2) 加密机制

- 3) 安全防护策略：安全日志、入侵检测、隔离防护、漏洞扫描
- 4) 数据备份与恢复手段：存储设备、存储优化、存储保护、存储管理
- 5) 防病毒系统

19. 一套完整的测试应该由哪些阶段组成？

测试计划、测试设计与开发、测试实施、测试评审与测试结论

20. 测试的文档应当贯穿于软件生命周期的全过程，其中用户文档是文档测试的重点。那么软件系统的用户文档包括哪些？

- 1.用户手册 2.安装和设置指导 3.联机帮助 4.指南、向导
- 5.样例、示例和模板 6.授权/注册登记表 7.最终用户许可协议

21. 软件系统除用户文档外，文档测试还应该关注哪些文档？

开发文档、软件需求说明书、数据库设计说明书、概要设计说明书、详细设计说明书、可行性研究报告、管理文档、项目开发计划、测试计划、测试报告、开发进度月报、开发总结报告

22. 简述软件系统中用户文档的测试要点？

读者群、术语、正确性、完整性、一致性、易用性、图表与界面截图、样例与示例、语言、印刷与包装。

23. 没有产品说明书和需求文档地情况下能够进行黑盒测试吗？

这个问题是测试工程师经常遇到的问题，根源就是国内的软件开发文档管理不规范，对变更的管理方法就更不合理了。

没有文档的时候，测试人员也是可以进行黑盒测试的，称之为探索测试。做法是：测试工程师通过自己的专业技能、领域知识等不断的深入了解测试对象、理解软件功能，进而发现缺陷。这种做法是把软件当成了产品说

说明书，测试过程中要和开发人员不断的进行交流。尤其在做项目的时候，进度压力比较大，可以作为加急测试方案。最大的风险就是不知道有些特性是否会被遗漏。

24. 测试中的“杀虫剂怪事”指的是？

用于描述测试人员对同一测试对象进行的测试越多，发现的缺陷越少。根本原因在于测试人员对测试软件过于熟悉，形成思维定势。为了克服这个现象，测试人员需要不断编写新的测试程序或者测试用例，对程序的不同部分进行测试，以发现更多的缺陷。也可以引用新人，刚刚进来的新手往往会发现一些意想不到的问题。

25. 完全测试程序是可能的吗？

实际上完全测试是不可能的，主要有以下原因：

- 1) 完全测试比较耗时，时间上不允许；
- 2) 完全测试通常意味着较多资源投入，这在现实中往往是行不通的；
- 3) 输入量太大，不能一一测试；
- 4) 输出结果太多，只能分类进行验证；
- 5) 软件实现途径太多；
- 6) 软件产品说明书没有客观标准，从不同的角度看，软件缺陷的标准不同；因此测试的程度要根据实际情况而定。

26. 软件测试的风险主要体现在哪里？

测试没有做好，客户碰到 bug 就是昂贵的后果。所以我们要选择最合适的测试量，把风险降低到最小。

27. 发现缺陷越多，说明软件缺陷越多吗？

如果软件编写的好，这种现象就不常见了。测试人员只要严肃认真的测试程序就可以了。

28. 软件测试人员就是 QA 吗？



软件测试人员的职责是尽可能早的找出软件缺陷，确保得以修复。主要工作是测试，测试工作也是测试保证人员的工作对象。质量保证人员的职责是创建或者制定标准和方法，提高促进软件开发能力和减少软件缺陷；质量保证人员日常的工作重要内容是检查与评审。

29. 如何减少测试人员跳槽带来的损失？

- 1) 加强部门内员工之间的互相学习，互相学习是建立学习型组织的基本要求，是知识互相转移的过程。
- 2) 企业能够给员工足够大的发展空间时，如果不是待遇特别低，员工不会主动离开。
- 3) 管理者应当把员工的个人成长和企业发展联系起来，为员工设立合理的发展规划并付诸实现。不过这个要有好的企业文化依托。

30. 测试产品与测试项目的区别是什么？

- 1) 质量要求不同，通常产品的质量要高一些，修复发布后产品的缺陷成本较高，甚至会带来很多负面的影响。而做项目通常面向某一用户，虽然质量越高越好，但是一般只要满足用户要求就可以了
- 2) 测试资源投入多少不同。

31. 为什么要在一个团队中开展软件测试工作？

因为没有经过测试的软件很难在发布之前知道该软件的质量，就好比 ISO 质量认证一样，测试同样也需要质量的保证，这个时候就需要在团队中开展软件测试的工作。在测试的过程发现软件中存在的问题，及时让开发人员得知并修改问题，在即将发布时，从测试报告中得出软件的质量情况。

软件测试的基本流程：测试计划、测试需求分析，测试用例的编写，测试执行，测试结果的分析，测试报告的编写。开发人员修改了 bug 后，一般要进行回归测试这样来进行测试的迭代。这就是大致的一个流程。

32. 常用的测试用例设计的方法有哪些？

等价类，因果图，边界值，正交实验法，错误推测法。白盒测试的测试手段有：语句覆盖、判定覆盖、条件覆盖、路径覆盖、条件组合覆盖。

软件测试的分类：（1）基于是否关注软件结构与算法：黑盒测试、白盒测试；（2）基于是否执行被测试软件：静态测试、动态测试；（3）基于测试的不同阶段：单元测试、集成测试、系统测试、验收测试；（4）软件测试的内容：回归测试：是指修改了旧代码后，重新进行测试，以确认修改没有引发新的错误或导致其他代码产生错误。自动回归测试将大幅降低系统测试、维护升级等阶段的成本。功能测试：是对产品的各功能进行验证，根据功能测试用例，逐步测试，检查产品是否达到用户要求的功能。负载测试：是一种性能测试，指数数据在超负荷环境中运行，程序是否负担。压力测试：通过确定一个系统的瓶颈或者不能接收的性能点，来获得系统能提供的最大的服务级别的测试。通俗的讲，压力测试是为了发现在什么条件下您的运用程序的性能会变得不可接受。性能测试：测试软件测试的性能，包括负载测试，强度测试，数据库容量测试、基准测试。强度测试：强度测试是一种性能测试，他在系统资源特别低的情况下软件系统运行情况。这类测试往往可以书写系统要求的软硬件要求水平。易用性测试：是指用户使用软件时是否感觉方便，比如是否最多点击鼠标三次就可以达到用户的目的。

### 33. 测试用例的目的？

编写测试用例最终目的是发现更多软件中的 bug。测试用例是测试人员进行测试操作的依据，它定义了测试的前提，测试环境，测试步骤，测试预期结果等要素，测试人员主要根据它来进行测试。

### 34. 软件测试的目的？

为软件的质量评估提供依据 为软件质量改进和管理提供帮助

### 35. 什么是软件测试？

广义：指软件生存周期中所有的检查、评审和确认工作，其中包括了对分析、设计阶段，以及完成开发后维护阶段的各类文档、代码的审查和确认 狭义：识别软件缺陷的过程，且实际结果与预期结果的不一致。

### 36. 测试用例的重要性?

测试用例的设计是整个软件测试工作的核心。测试用例反映对被测对象的质量要求，决定对测试对象的质量评估

### 37. 测试中要编写的文档?

测试计划，测试方案，测试用例，缺陷报告，测试报告，进度报告，测试记录等

### 38. 基于 WEB 信息管理系统测试时应该考虑的因素有哪些?

界面设计、界面易用性、功能、规范性、合理性、美观与协调性、菜单设置、独特性、快捷方式的组合、容错性考虑（这个是从恢复测试来考虑）、多窗口的应用与系统资源

### 39. 软件测试项目从什么时候开始? 为什么?

软件测试应该在需求分析阶段就介入，因为测试的对象不仅仅是程序编码，应该对软件开发过程中产生的所有产品都测试；并且软件缺陷存在放大蓄势，缺陷发现越晚，修复他的成本越大

### 40. 需求测试注意事项有哪些?

- 1) 要与客户有良好的沟通；
- 2) 确立需求规格说明书，需求规格说明书尽量做到完整、详尽，并与客户实际要求保持一致

### 41. 软件的生命周期?

软件的生命周期一般是开发生命周期+测试生命周期。

开发生命周期：需求分析->概要和详细设计->编码->维护

测试生命周期：测试计划->测试设计->测试执行->测试评估

### 42. 怎么划分缺陷的等级?

极高：在测试的过程中出现死机，数据丢失、系统崩溃、功能没有实现等此类现象的级别； 高：此类缺陷导致系统功能不稳定，或功能实现错误，流程错误等； 中：校验错误、罕见故障、错别字等不会影响系统功能，但会影响用户易用性的错误； 低：对系统没影响的一些小问题。

43. 软件工程工程师的基本素质？

1、良好的编码能力；2、自觉规范意识和团队精神；3、认识和运用数据库的能力；4、较强的英语阅读能力和写作能力；5、具有软件工程的概念；6、具有求知欲和上进心

44. 一个缺陷测试报告的组成？

缺陷的标题、缺陷的基本信息、复现缺陷的操作步骤，缺陷的实际结果描述，期望的正确结果

45. 需求测试注意事项有哪些？

完整性、正确性、可行性、一致性、无二义性、健壮性、必要性、可测试性、可修改性、可跟踪性

46. 当面试官最后问你有没有什么问题要问的时候？

1、贵公司的近期和远期发展目标是什么？ 2、贵公司的主要竞争对手有哪些？ 3、贵公司有多少开发人员有多少测试人员？ 4、贵公司又进一步扩充测试人员的计划吗？ 5、如果我有幸能够进入贵公司的话，我有怎么样的发展？ 6、测试人员的沟通能力很重要，贵公司有规范的沟通渠道吗？ 8、请问我什么时候能够知道结果？

47. 一个测试工程师应当具备那些素质和技能？

1、掌握基本的测试基础理论；2、本着找出软件存在的问题的态度进行测试，即保持客观，不要以挑刺形象出现；3、可熟练阅读需求规格说明书等文档；4、以用户的观点看待问题；5、有着强烈的质量意识；6、细心和责任心；7、良好有效的沟通方式（与开发人员及客户）； 8、能够及时准确判断出高危险区在哪；

1) 沟通能力：和用户谈话的重点必须放在系统可以正确地处理什么和不可以处理什么上，和开发人员谈相同信息时，就必须把这些话重新组织语言以另一种方式表达出来，测试小组的人员必须能够同等地同用户和开发者沟通

- 2) 移情能力：和系统开发有关的所有人员都处在一种既关心又担心的状态之中。用户担心将来使用不符合要求、开发者担心由于系统要求不正确使他不得不重新开发整个系统、管理部门则担心这个系统突然崩溃而使他名誉受损。需要测试人员对他们每个人都有足够的理解和同情，具备了这种能力可以将测试人员与相关人员之间的冲突和对抗降到最低程度
- 3) 技术能力：测试者必须既明白被测试软件的概念又要会使用工程中的那些工具、丰富的开发经验可以帮助对开发过程有较深入的理解；从开发的人员角度正确的评价测试者，简化自动测试工具编程的学习曲线
- 4) 自信心：开发者指责测试者出了错是常有的事，测试者必须对自己的观点有足够的自信心。如果容许别人对自己指东指西，就不能完成什么更多的事情了
- 5) 外交能力：机智老练和外交手法有助于维护与开发人员的协作关系，不可过硬，避免影响双方关系。
- 6) 幽默感：在遇到狡辩的情况下，一个幽默的批评将是很有帮助的
- 7) 记忆力：一个理想的测试者应该有能力将以前曾经遇到类似的错误从记忆深处挖掘出来，这一能力在测试过程中的价值无法衡量的。许多新问题与旧问题没差别
- 8) 耐心：一些质量保证工作需要难以置信的耐心。有时你需要花费惊人的时间去分离、识别和分派一个错误。这个工作是那些坐不住的人无法完成的。
- 9) 怀疑精神：可以预料，开发者会尽他们最大的努力将所有的错误解释过去。测试者必须听每个人的说明，但他必须保持怀疑直到他自己看过以后，
- 10) 自我督促：干测试工作很容易使你变得懒散。只有那些具有自我督促能力的人才能够使自己每天正常地工作。
- 11) 洞察力：一个好的测试工程师具有“测试是为了破坏”的观点，捕获用户观点的能力，强烈的质量追求，对细节的关注能力。应用的高风险区的判断能力以便将有限的测试针对重点环节。

48. 你自认为测试的优势在哪里？

优势在于我对测试坚定不移的信心和热情，虽然经验还不够，但测试需要的基本技能我有信心在工作中得以发挥。

49. 黑盒测试的优点与缺点

- 1) 比较简单，不需要了解程序内部的代码及实现；
- 2) 与软件的内部实现无关；
- 3) 从用户角度出发，能很容易的知道用户会用到哪些功能，会遇到哪些问题；
- 4) 基于软件开发文档，所以也能知道软件实现了文档中的哪些功能；

在做软件的自动化测试时较为方便 黑盒测试的缺点有：不可能覆盖所有的代码，覆盖率较低，大概只能达到总代码量的 30%；自动化测试的复用性。

50. 白盒测试的优点与缺点

优点：帮助软件测试人员增大代码的覆盖率，提高代码的质量，发现代码中隐藏的问题

缺点：

- 1) 程序运行会有很多不同的路径，不可能测试所有的运行路径
- 2) 测试基于代码，只能测试开发人员做的对不对，而不能知道设计的正确与否，可能会漏掉一些功能需求
- 3) 系统庞大时，测试开销非常大

51. 软件测试的主要任务

保证软件的可靠性。 开发和测试的关系：二者相辅相成，需要软件测试人员、程序员和系统分析师等密切联系，需要更多的交流协商，以便提高测试效率。

52. 准确的说出测试误区的原因：

1、软件开发完成后进行软件测试；2、对软件测试的轻视；3、软件测试要求不高，随便找个人就行；4、软件测试是测试人员的事情与开发无关；5、项目进度吃紧时少做些测试，时间富裕时多做测试。

53. 软件测试的目的？简单的说是什么？

目的在于发现错误，一个成功的测试是发现至今未发现的错误测试。

简单的说，测试的根本目的就是保证最终交给用户的产品符合用户的需求，在产品交给用户前尽可能多的发现并改正问题。

具体讲，确保产品完成了它所承诺或公布的功能，并且用户可以访问到的功能都有明确的书面说明，确保产品满足性能和效率的要求，确保产品是健壮性的和适应用户环境的。

54. 什么是 Bug?它是如何产生的？

1、产品说明书中规定要做的事情，而软件没有实现；2、产品说明书中规定不要做的事情，而软件却实现了；3、产品说明书没有提到的事情，而软件却实现了；4、产品说明书中没有提到但是必须要做的事情，软件却没有实现

程序开发产生 Bug 的原因：1、需求不清晰：软件复杂性，程序编码错误；2、需求变化；3、时间压力；4、代码文档贫乏；5、开发工具自身错误；

测试导致 Bug 的原因：1、测试目标定义错误；2、在开发生命周期中，错误的选择了测试介入时期/3、选择了低效的测试技术；4、测试人员专业知识培训不够；5、工作低效：计划不够详细；6、测试随意性很大；7、测试人员同开发人员沟通困难。

55. 什么是静态测试？开始静态测试的前提、时机？

静态测试，指不运行被测程序本身，仅通过分析或检查源程序的文法，结构，过程，接口等来检查程序的正确性。理论上讲，静态测试应在项目立项开始，贯穿整个项目的始终，原则是越早越好。

## 56. 什么是动态测试？

动态测试，指运行被测程序，检查运行结果与预期结果的差异，并分析运行效率和健壮性等性能；

这种方法有三部分组成：编写测试用例，执行程序，分析程序的输出结果。 步骤：单元测试、集成测试、系统测试、验收测试、回归测试。

## 57. 静态测试与动测试的区别？

1、静态测试是用来预防的，动态测试是用来矫正的/2、多次的静态测试比动态测试要效率和效益高；3、静态测试综合测试程序代码；4、在相当短的时间内里，静态测试的覆盖率能达到 100%，而动态测试经常是只能达到 50%左右，原因动态测试发现的 bug 大部分只是测试实际执行的那部分代码；5、动态测试比静态测试更浪费时间；6、静态测试比动态测试更能够发现 bug，静态测试的执行可以在程序编码编译前、动态测试只能在编译后才能执行。

## 58. 用户需求分析

定义：在系统设计之前和设计、开发过程中对用户需求所做的调查与分析，是系统设计、系统完善和系统维护的依据。 过程：需求说明、需求确认、建立需求状态；例子：CLOSED：经过确认，双方认可并达成共识；

OPEN：双方确认，但没有达成共识的需求。 待定：客户提出需求，但双方没有经过沟通或确认；

## 59. 概要设计的主要任务

把需求分析得到的 DFG 转换为软件结构和数据结构；设计软件结构的具体任务是： 将一个复杂系统按功能进行模块划分、建立模块的层次结构及调用关系、确定模块间的接口及人机界面等。 数据结构设计包括数据特征的描述、确定数据的结构特性、以及数据库的设计。

## 60. 测试模型

V 模型是软件开发瀑布模型的一种， 它反映了测试活动与分析和设计的关系：用户需求 对应 验收测试；需求分析和系统设计 对应 确认测试和系统测试；概要设计 对应 集成测试；详细设计 对应 单元测试与编码



局限性：整个软件产品的过程质量保证完全依赖于开发人员的能力和工作的责任心，如果有环节出错，必将延缓整个工程的质量和预期进度。也就是把测试作为编码之后的最后一个活动，需求分析等前期产生的错误直到后期的验收测试才能发现！

W 模型：相对于 V 模型增加了软件各开发阶段中应同步进行的验证和确认活动。W 模型强调测试伴随着整个软件开发周期，而且测试的对象不仅仅是程序，需求、设计等同样要测试，也就是说，测试同开发要同时进行。W 模型有利于尽早地全面发现问题。

局限性：仍把开发活动看成是从需求开始到编码结束的串行活动，只有上一阶段完成后，才开始下一阶段的活动，不能支持迭代，自发性以及变更调整。

H 模型：软件测试过程活动完全独立，贯穿了产品的整个生命周期，与其他流程并发进行。某个测试点准备就绪时，就可以从测试准备阶段到测试执行阶段；软件测试要尽早准备，尽早执行。软件测试可以根据被测物的不同而分层次进行。X 模型的优点是能帮助有测试经验的人员在测试计划之外发现更多的软件错误；

缺点是对测试造成人力、物力、财力的浪费，对测试员的熟练程度要求比较高。