### 软测简答

**1、代码评审有哪些方法？**  
 （1）代码互查  
 （2）代码走查  
 （3）正式会议审查

**2、单元测试的任务主要包括哪些？**  
 （1）单元中所有独立执行路径测试 （2）单元局部数据结构测试  
 （3）单元接口测试 （4）单元边界条件测试  
 （5）单元的各条错误处理通路测试 （6）内存分析

**3、集成测试、单元测试与系统测试的联系与区别是什么。**  
 （1）联系：①单元测试是对软件基本组成单元进行正确性检验的测试工作  
 ②集成测试是在单元测试的基础上，将所有模块按照设计要求组装成为子   
 系统或系统，验证组装后功能以及模块间接口是否正确的测试工作  
 ③系统测试是将已经集成好的软件系统，在实际使用环境下，对计算机系  
 统进行一系列的组装测试和确认测试的工作  
 （2）区别：采用的测试方式不同：  
 ①单元测试：白盒测试  
 ②集成测试：灰盒测试  
 ③系统测试：黑盒测试

**4、简述测试用例的作用。**  
 （1）是测试人员在测试过程中的重要参考依据  
 （2）可以帮助实施有效的测试  
 （3）使测试过程事半功倍  
 （4）是一个知识积累的过程

**5、简述测试自动化普遍存在的问题。**  
 （1）不正确的观念或不现实的期望  
 （2）缺乏具有良好素质、经验的测试人才  
 （3）测试工具本身的问题影响测试的质量  
 （4）没有进行有效的、充分的培训  
 （5）没有考虑到公司的实际情况，盲目引入测试工具  
 （6）没有形成一个良好的使用测试工具的环境  
 （7）其他技术问题和组织问题

**6、简述测试总结报告的目的及内容。**  
 （1）目的：对测试对象有一个清楚的描述，对测试记录、测试结果如实汇总分析  
 （2）内容：①产品标识   
 ②用于测试的计算机系统   
 ③使用的文档及其标识  
 ④产品描述、用户文档、程序和数据的测试结果   
 ⑤与要求不符的清单  
 ⑥针对建议的要求不符的清单，产品未做符合性测试的说明  
 ⑦测试结束日期

**7、简述单元测试的目标和任务。**  
 （1）目标：①确保各单元模块被正确地编码  
 ②确保代码在结构上可靠且健全  
 （2）任务：①单元中所有独立执行路径测试 ②单元局部数据结构测试  
 ③单元接口测试 ④单元边界条件测试  
 ⑤单元的各条错误处理通路测试 ⑥内存分析

**8、简述单元测试过程中的文档管理。**  
 （1）《软件需求规格说明书》、《软件详细设计说明书》、《单元测试计划》  
 （2）《单元测试计划》、《软件详细设计说明书》、《单元测试计划》  
 （3）《单元测试用例》文档及《软件需求规格说明书》、《软件详细设计说明书》、  
 《缺陷跟踪报告》、《缺陷检查表》  
 （4）《单元测试用例》、《缺陷跟踪报告》、《缺陷检查表》、《单元测试检查表》  
 （5）《单元测试报告》

**9、简述单元测试和代码调试的区别**。  
 （1）单元测试：为了发现软件缺陷  
 （2）代码调试：为了修复软件缺陷

**10、简述好的用户界面所具备的要素。**  
 （1）符合标准和规范 （2）直观性 （3）一致性 （4）灵活性  
 （5）舒适性 （6）正确性 （7）实用性

**11、简述黑盒测试方法中常用的具体方法。**  
 （1）等价类划分法 （2）边界值分析法 （3）错误推测法 （4）因果图法  
 （5）决策表法 （6）正交试验发 （7）场景法 （8）功能图法

**12、简述集成测试的过程。**  
 （1）构建的确认 （2）补丁的确认 （3）系统集成测试测试组提交  
 （4）测试用例设计 （5）测试代码编写 （6）Bug的报告  
 （7）每周/每两周的构建 （8）点对点的测试 （9）组内培训

**13、简述集成测试中自顶向下和自底向上集成方法各自的优缺点。**  
 （1）自顶向下：①优点：不需要驱动程序，早期能发现上层模块的接口错误  
 ②缺点：需要桩程序，早期不能充分展开人力  
 （2）自底向上：①优点：不需要桩程序，早期能充分展开人力  
 ②缺点：需要驱动程序，早期不能发现上层模块的接口错误

**14、简述软件评审的目的。**  
 （1）尽早地发现产品中的缺陷  
 （2）更早地发现需求工程、软件设计等各个方面的问题  
 （3）将问题记录下来，使其具有可追溯性

**15、简述软件缺陷产生的原因。**  
 （1）软件需求说明书编写的不全面、不完整、不准确，而且经常更改  
 （2）软件设计说明书不准确或者有偏差  
 （3）软件设计人员的水平有欠缺  
 （4）开发人员不能很好的理解需求说明书和沟通不足

**16、简述系统的性能指标有哪些。**  
 （1）延迟时间 （2）上升时间 （3）峰值时间  
 （4）调节时间 （5）超调量 （6）振荡次数

**17、简述制定测试计划时要面对的问题。**  
 （1）系统的测试范围 （2）测试的方法 （3）质量标准  
 （4）测试软件的编写 （5）自动化测试工具的选择

**18、如何划分等价类？**  
 （1）输入条件规定取值范围或值的个数，确立一个有效等价类和两个无效等价类  
 （2）输入条件规定输入值的集合，确立一个有效等价类和一个无效等价类  
 （3）输入条件是一个布尔量，确立一个有效等价类和一个无效等价类  
 （4）规定输入数据的一组值，并且程序对每一个值分别处理，确立n个有效等价类  
 和 一个无效等价类  
 （5）规定输入数据必须遵守的规则，确立一个有效等价类和若干个无效等价类  
 （6）确知已划分的等价类中各元素在程序处理中的方式不同，应在将该类等价类进  
 一步的划分为更小的等价类

**19、软件测试按照测试层次划分可分为几层，分别是什么？**  
 （1）底层层次：单元测试  
 （2）接口层次：集成测试  
 （3）系统层次：系统测试  
 （4）用户层次：验收测试

**20、软件验收测试应完成哪些主要测试工作？**  
 （1）文档资料的审查验收 （2）功能测试 （3）性能测试   
 （4）强化测试 （5）性能降级执行方式测试 （6）检查系统的余量要求   
 （7）安装测试 （8）用户操作测试

**21、什么是测试用例？为什么要设计测试用例？**  
 （1）测试用例：将软件测试的行为活动，做一个科学化的组织归纳。  
 （2）使用测试用例的好处：  
 ①避免盲目测试并提高测试效率  
 ②使软件测试的重点突出、目的明确  
 ③降低工作强度、缩短项目周期  
 ④使软件测试易于开展

**22、软件产品质量可以归纳为哪些属性？**  
 ①功能性 ②可用性 ③可靠性 ④性能 ⑤容量   
 ⑥可测试性 ⑦可维护性 ⑧兼容性 ⑨可扩展性

**23、什么是白盒测试？**  
 白盒测试，也称结构化测试或逻辑驱动测试，也就是已知产品的内部工作过程，清楚最终生成软件产品的计算机程序结构及其语句，按照程序内部的结构测试程序，测试程序内部的变量状态、逻辑结构、运行路径等，检验程序中的每条通路是否都能按预定要求正确工作，检查程序内部动作或运行是否符合设计规格要求，所有内部成分是否按规定正常进行。

**24、什么是黑盒测试？**  
 黑盒测试，也称数据驱动测试，在测试时，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，测试人员针对软件直接进行测试，检查系统功能是否按照需求规格说明书的规定正常使用、是否能适当地接收输入数据而输出正确的结果等，检查相应的文档是否采用了正确的模板、是否满足规范要求。

**25、从软件测试层次方面划分软件测试的级别。**  
 （1）单元测试：单元测试的单元是组成可靠系统的坚实基础，单元测试在质量保证活  
 动中举足轻重。  
 （2）集成测试：集成测试是在单元测试的基础之上，按照设计要求不断进行集成而进  
 行的测试。  
 （3）系统测试：系统测试是在集成测试完成之后进行，而且是针对应用系统进行测试  
 （4）验收测试：验收测试的目的是向未来的用户表明系统能够像预定要求那样工作，  
 验证软件的功能和性能及其他特性是否与用户的要求一致。

**26、测试计划书的主要内容集中在哪几个方面？**  
 （1）目标和范围 （2）项目估算 （3）风险计划  
 （4）进度安排 （5）资源配置 （6）跟踪和控制机制

**27、测试用例的设计需要遵守哪些流程？**  
 （1）制定测试用例设计的策略和思想，在测试计划中描述出来

（2）设计测试用例的框架，也就是测试用例的结构。

（3）细化结构，逐步设计出具体的测试用例。

（4）通过测试用例的评审，不断优化测试用例。

**28、优秀测试工程师应具备哪些素质？**  
 （1）责任感 （2）沟通能力 （3）技术能力 （4）自信心

（5）耐心 （6）怀疑精神 （7）适度的好奇心 （8）洞察力

（9）反向思维和发散思维能力 （10）记忆力

**29、什么是有效等价类和无效等价类？**  
 （1）有效等价类是指输入完全满足程序输入的规格说明、有意义的输入数据所构成的集合。利用有效等价类可以检验程序是否满足规格说明所规定的的功能和性能。

（2）无效等价类和有效等价类相反，即不满足程序输入要求或者无效的输入数据构成的集合。使用无效等价类，可以测试程序的容错性。

**30、什么是测试管理方法（TMap）？**  
 TMap是一种业务驱动的、基于风险策略的、结构化的测试方法体系，目的是更早地发现缺陷，以最小的成本，有效地，彻底地完成测试任务，以减少软件发布后的支持成本。由计划和控制、基础设施、准备、说明、执行和完成等阶段组成。  
**31、什么是敏捷测试？**  
 敏捷测试是符合敏捷测试宣言的思想，遵守敏捷开发原则，在敏捷开发环境下能够很好地和其整体开发流程融合的一系列测试时间。敏捷测试作为敏捷开发的组成部分，能够适应敏捷开发的流程，有效地帮助敏捷开发实现对质量的控制或促进软件产品的质量提升。  
**32、基于风险的测试过程分为哪些步骤？**  
 （1）列出软件的所有功能和特性  
 （2）确定每个功能出错的可能性

（3）如果某个功能出错或欠缺某个特征，需要评估对用户使用软件产品的影响程度

（4）根据上面连个步骤，计算风险度

（5）根据可能出错的迹象，来修改风险度

（6）决定测试的范围，编写测试方案

**33、正式会议审查有哪些主要工作？**  
 （1）由程序编写小组成员逐句阐述程序的逻辑，在此过程中可由程序员或测试小组成   
 员提出问题，追踪缺陷是否存在。

（2）利用通用缺陷检查表来分析讨论。主持人负责讨论沿着建设性方向进行，而其他  
 人则集中注意力发现缺陷。  
 （3）记录所有已确定的缺陷，在会议之后形成《评审报告》。  
 （4）审查小组根据代码审查的错误记录来评估程序，决定是否需要重新进行审议。

**34、什么是驱动模块？**  
 驱动模块是用以模拟被测模块的上级模块，能够调用被测模块。在测试过程中，驱动模块接收测试数据，调用被测模块并把相关的数据传送给被测模块。  
**35、什么是桩模块？**  
 桩模块是用以模拟被测模块工作过程中所调用的下层模块。桩模块由被测模块调用，它们一般只进行很少的数据处理，以便于检验被测模块与其下级模块的接口。  
**36、集成测试的分类？**  
 （1）非渐增式测试模式：先分别测试每个模块，再把所有模块按设计要求放在一起结   
 合成所要的程序。

（2）渐增式测试模式：把下一个要测试的模块同已经测试好的模块结合起来进行测试，测试是在模块一个一个的扩展下进行，其测试的范围逐步增大。

**37、非渐增式测试模式和渐增式测试模式各自的优缺点？**  
 （1）渐增式测试模式需要编写的软件较多，工作量较大，而非渐增式测试开销小。

（2）渐增式测试模式发现模块间接口错误早；而非渐增式测试模式晚。

（3）非渐增式测试模式发现错误，较难诊断；而使用渐增式测试模式，如果发生错误  
则往往和最近加进来的那个模块有关。

（4）渐增式测试模式测试更彻底。

（5）渐增式测试模式需要较多的机器时间。

（6）使用非渐增式测试模式，可以并行测试。

**38、自顶向下测试方法有哪些步骤？**  
 （1）对主控模块进行测试，测试时用桩程序代替所有直接附属于主控模块的模块。

（2）根据选定的结合策略，每次用一个实际模块代替一个桩程序。

（3）在结合下一个模块的同时进行测试。

（4）为了保证加入模块没有引进新的错误，可能需要进行回归测试。

从第2步开始不断地重复进行上述过程，直至完成。

**39、自底向上测试方法有哪些策略？**  
 （1）把底层模块组合成实现某个特定的软件子功能的族。

（2）写一个驱动程序，协调测试数据的输入和输出。

（3）对由模块组成的子功能族进行测试。

（4）去掉驱动程序，沿软件结构自下向上移动，把子功能族组合起来形成更大的子功能族。

从第2步开始不断地重复进行上述过程，直至完成。

1. **什么是性能测试？**  
    性能测试就是为了发现系统性能问题或获取系统性能相关指标(运行速度、响应时间、资源使用率等)而进行的测试。一般在真实环境、特定负载条件下，通工具模拟实际软件系统的运行及其操作，同时监控性能各项指标，最后对测试结果进行分析，确定系统的性能状况。  
   **41、系统行为的性能指标主要有哪些？**  
    （1）请求响应时间：客户端浏览器向Wb服务器提交一个请求到收到响应之同的同隔时间。  
    （2）事务响应时间：事务可能由一系列请求组成，事务的响应时间就是这些请求完成处  
    理所花费的时间。

（3）数据吞吐量：单位时间内客户端和服务器之间网络上传输的数据量。

**42、什么是压力测试?**  
 压力测试也称为强度测试、负载测试。压力测试是模拟实际应用的软硬件环境及用户使用过程的系统负荷，长时间或超大负荷地运行测试软件，来测试被测系统的性能、可靠性，稳定性等。压力测试的目的就是在软件投入使用以前或软件负载达到极限以前，通过执行可重复的负载测试，了解系统可靠性、性能瓶颈等，以提高软件系统的可靠性、稳定性，减少系统的宕机时间和因此带来的损失。  
**43、什么是安全性测试？**  
 安全性是一个复杂的主题，涉及部署系统的各个级别。安全性要求分析，包括确定可能的或潜在的各类安全威胁和找到处理这些威胁的策略，即：

（1）确定关键（有形的和无形的）资产，并找到对这些资产的威胁。

（2）确定使组织暴露于可能带来风险威胁的薄弱环节。

（3）开发减轻组织风险的安全策略。

**44、什么是容错性测试？**  
 容错性测试主要检查系统的容错能力，检查软件在异常条件下自身是否具有防护性的措施或者某种灾难性恢复的手段。容错性测试可以看作可靠性测试和健壮性测试的组成部分，容错测试首先要通过各种手段，让软件强制性地发生故障，然后验证系统是否能尽快恢复。  
**45、软件可靠性包含哪些要素？**  
 （1）规定的时间  
 （2）规定的环境条件  
 （3）规定的功能