

# 第一章 软件测试常见面试题

## 1 软件的含义

程序、数据及相关文档的完整集合。

## 2 测试与调试的区别是什么？

测试是由测试人员进行，主要目标是发现、报告和跟踪缺陷。

调试是由开发人员进行，主要目标是定位缺陷位置，分析缺陷原因，修复缺陷。

## 3 IEEE 是什么意思？

国际电气电子工程师协会

## 4 GB 是什么意思？

国家标准

## 5 软件测试的含义

简单讲，软件测试是发现缺陷的过程；IEEE 中的定义是，软件测试是使用人工或自动手段来运行或测定某个系统的过程，目的在于检验它是否满足规定的需求或弄清预期结果与实际结果之间的差别。

## 6 软件测试的目的

- (1) 验证软件是否满足各类文档说明书等规定的软件质量要求
- (2) 找出软件缺陷
- (3) 为软件产品的质量测量和评价提供依据
- (4) 帮助开发改进开发流程

## 7 什么是功能、性能、兼容性

功能代表一个软件能做什么；性能反映软件运行的速度或效率、占用资源的多少等指标；兼容性表示一个软件与其所在运行环境的依赖程度，包括与硬件、操作平台、其他软件的依赖。

## 8 测试分为哪几个阶段？每个阶段的测试目的是什么？

测试分为单元测试、集成测试、系统测试、验收测试四个阶段。前三个阶段的目的是尽可能多的发现缺陷，而验收测试是要验证软件满足了用户需求，帮助用户建立系统可以正常使用的信心，发现缺陷不是此阶段的目标。

## 9 解释 QA 及其职责

QA 的含义是软件质量保证（人员）。

主要职责是制定和加强促进软件开发并防止软件缺陷的标准和方法，并监督标准和过程被正确的遵循。

## 10 测试工程师与软件质量保证的区别

测试工程师的主要任务是在最短的时间内发现尽可能多的缺陷，并确保这些缺陷得以修复。

软件质量保证的主要职责是制定和加强促进软件开发并防止软件缺陷的标准和方法，并监督标准和过程被正确的遵循。

## 11 测试应该由什么人来进行？

测试应该由独立的第三方来进行，第三方表示测试人员不参与程序的开发。

## 12 pareto 法则、帕累托法则、28 原则、82 原则

一般情况下 80% 的缺陷聚集在 20% 的关键核心业务模块中，这个原则至少告诉我们在做测试时，应该重点分析和测试 20% 的核心业务，具体说要做好需求分析。

## 13 杀虫剂怪事

杀虫剂怪事用于描述软件测试越多，其对测试的免疫力越强的现象。这个现象告诉我们，测试时，应尝试新方法、不同的测试程序，对程序进行测试，以找出更多软件缺陷。

## 14 木桶原理

木桶原理在软件方面的主要含义是全面质量管理，另外还告诉我们测试时要关注团队中较弱的人。

## 15 Good-enough 原则

Good-enough 原则告诉我们做测试的时候既不要做过多测试，也不做不充分的测试。至于多少测试合适，需要我们不断积累经验，在项目中可以指定最低测试通过标准和测试内容，然后具体问题具体分析。

## 16 群集效应

群集效应的含义是发现的缺陷越多，证明软件存在的缺陷越多。群集效应指导我们在找到软件缺陷的地方要继续找找。

## 17 什么是确认测试？回归测试？

确认测试也称再测试：缺陷修复以后，验证缺陷是否真正修复

回归测试：缺陷修复以后，确保对程序的修改没有给软件其他未改变部分带来新的缺陷。

## 18 测试人员应该具备哪些素质？

要有责任心，要有破坏的态度，对事不对人，三心二意（细心、信心、耐心、缺陷预防意识、沟通意识），具有一定的开发技能，善于思考。

## 19 如果测试提交的缺陷开发人员不认可，该怎么办？

首先分析与开发沟通开发不认可的原因。

如果拒绝原因是提交的不是缺陷，而且自己分析后，的确不是缺陷，则应该注意以后再做测试时要做好复现，认真研读需求，提高自己找缺陷的能力。

如果拒绝原因是提交的不是缺陷 但自己分析时认为缺陷应该是存在的，则再次研读需求并做好复现，拿出确实是缺陷的证据，然后与开发沟通。

如果拒绝原因是认可缺陷，但不予修复，如果自己觉得必须修复，则拿出充分理由和证据和不修复的不利影响和影响范围，再与开发沟通。

注意沟通技巧，合理的论述，向开发说明自己的判断的理由，注意客观、严谨，不掺杂个人情绪。

把问题交给测试经理，等待测试经理做出最终决定，如果仍然存在争议，可以通过公司政策所提供的渠道，向上级反映，并由上级做出决定。

## 20 如何解决开发和测试的矛盾？

- （1）以沟通和合作的方式开展工作
- （2）提高开发技能
- （3）换位思考
- （4）进行有效沟通

## 21 测试团队中都有哪些角色？各负责什么任务？各有多少人？

测试负责人：制定测试计划，监督安排任务，进行测试总结，1

测试工程师：进行测试需求分析、设计用例、搭建环境、执行用例、提交并跟踪缺陷，3

技术支持：负责环境维护，1

配置管理员：维护版本架构，维护版本库，文档配置，1

质量保证人员：负责软件质量方面的工作，1

## 22 什么是软件开发生命周期？

从软件最初构思到公开发行的过程。瀑布模型的过程是计划、需求、设计、编码、测试、运行、维护循环。

瀑布模型有严格的开发步骤，每个阶段是按顺序进行的，每个阶段都必须编写完整的文档，每个阶段完成后必须经过审查才能进入下一步。

瀑布模型不能迭代、不能反复；测试在编码之后，测试太晚；测试的只是程序。

## 23 软件开发有什么模型？软件测试主要有哪些模型？

软件开发模型：大爆炸模型、边写边改模型、瀑布模型、螺旋模型、敏捷开发模型

软件测试模型：V 模型、W 模型、H 模型、X 模型、前置测试模型、敏捷测试模型

## 24 简述 V 模型。

V 模型的过程：用户需求→需求分析→概要设计→详细设计→编码→单元测试→集成测试→系统测试→验收测试。

优点：

(1) V 的左端表示传统的瀑布开发模型，V 的右端明确地将测试分为不同的级别或阶段，测试过程更为具体；

(2) 测试各个阶段和开发的各个阶段相对应；

(3) V 模型的测试策略包括低层测试和高层测试，低层测试是为了源代码的正确性，高层测试是为了整个系统满足用户的需求。

缺点：

(1) 测试的对象就是程序本身。忽视了测试活动对需求分析，系统设计等活动的验证和确认的功能，直到后期的验收测试才被发现。

(2) 测试是开发之后的一个阶段。实际应用中容易导致需求阶段的错误一直到最后系统测试阶段才被发现。

## 25 简述 W 模型。

W 模型的过程：左边 V 是需求分析→概要设计→详细设计→编码实现→模块集成→系统构建→系统安装；右边 V 是需求测试→概要设计测试→详细设计测试→单元测试→集成测试→系统测试→验收测试。

优点：

(1) W 模型体现了尽早和不断测试的原则，既强调测试方案设计，也强调测试执行。

(2) 左侧 V 是开发，右侧 V 是与开发并行的测试，相对于 V 模型，W 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 明确表示出了测试与开发的并行关系。测试与开发是同步进行的，有利于尽早地全面的发现问题。

(3) 测试伴随整个软件开发周期，且测试的对象不仅仅是程序，需求、设计等同样要测试。

缺点：

在 W 模型中，需求、设计、编码等活动被视为串行的，测试和开发活动也保持着一种线性的前后关系，上一阶段完全结束，才可正式开始下一个阶段工作。这样就无法支持迭代的开发模型，不利于当前软件开发复杂多变的情况。

## 26 简述 H 模型。

H 模型将测试活动完全独立出来，形成一个完全独立的流程，将测试准备活动和测试执行活动清晰地体现出来。H 模型的测试流程是只要测试准备工作完成，达到测试就绪点，测试就可以执行了。

优点：

- (1) 软件测试不仅仅指测试的执行，还包括很多其他的活动。
- (2) 软件测试是一个独立的流程，贯穿产品整个生命周期，与其他流程并发地进行。当某个测试时间点就绪时，软件测试即从测试准备阶段进入测试执行阶段。
- (3) H 模型反映出软件测试要尽早准备，尽早执行。
- (4) 软件测试可以进行迭代、反复进行。

## 27 敏捷开发

敏捷开发的核心思想是：以人为本，适应变化。

具体讲：

- (1) 认为个体和交互重于过程和工具，强调通过过程和工具理解个人和交流的作用；
- (2) 认为可用软件重于完备文档，强调通过全面的文档理解运行的软件；
- (3) 认为客户协作重于合同谈判，强调通过合同和谈判得到客户的协作；
- (4) 认为响应变化重于遵循计划，强调在计划的执行中做出对变更的响应。

特点：

- (1) 敏捷开发提倡迭代式和增量式的开发模式，并强调测试在其中的重要作用。
- (2) 敏捷开发是以用户为中心、以客户需求为导向的开发过程，在此过程中随时做好“迎接变化”的准备，客户是敏捷的关键环节。
- (3) 敏捷开发没有单一固定的开发方法或过程，敏捷开发有三个共同点：依赖客户的参与、测试驱动以及紧凑的迭代开发周期。

## 28 敏捷测试

(1) 敏捷测试是协同测试的一种形式，程序员结对编程，程序员分饰测试员角色，敏捷测试是连续测试。

(2) 敏捷测试侧重单元测试和验收测试。单元测试的过程是先设计单元测试用例，然后进行编码，之后执行测试。

(3) 敏捷测试强调客户参与，单元测试通过之后代码集成到代码库中，再由客户进行验收测试，验收测试的结论反馈给开发人员，缺陷得以迅速修复。

## 29 软件质量要求有哪些？

功能要求和非功能要求。

## 30 软件非功能要求有哪些？

性能要求（负载测试、压力测试、容量测试、可靠性测试）、界面测试、兼容性测试、易用性测试、文档测试、可用性测试、安装测试、安全测试、灾难恢复测试等。

## 31 简述测试的基本过程

- (1) 测试人员进行测试需求分析。
  - (2) 测试负责人编写测试计划。
  - (3) 测试人员根据测试需求分析设计和编写测试用例。
  - (4) 测试人员搭建测试环境、创建测试数据、执行测试用例、提交缺陷报告并进行跟踪、记录测试事件。
  - (5) 进行测试评估和总结。
- 每一分步工作完成后都进行评审。

## 32 拿到一个软件后，应该怎样开始工作？

编写需求分析并评审→编写测试计划并评审→设计测试用例并评审→搭建测试环境、执行测试用例、提交缺陷报告→进行评估和总结

## 33 怎么做测试？

编写需求分析并评审→编写测试计划并评审→设计测试用例并评审→搭建测试环境、执行测试用例、提交缺陷报告→进行评估和总结

## 34 简介测试流程

编写需求分析并评审→编写测试计划并评审→设计测试用例并评审→搭建测试环境、执行测试用例、提交缺陷报告→进行评估和总结。

## 35 怎么进行测试需求分析？

- (1) 收集各类文档，仔细阅读文档，提出问题，分析问题或沟通解决，整理需求信息。
- (2) 编写测试需求分析说明书：功能分解，编写检查点和测试点。
- (3) 需求评审。

## 36 拿到项目后，需要分析或咨询软件哪些方面的问题？

软件主要的功能、流程、开发环境（开发语言<含数据类型>、数据库、中间件）、运行环境（硬件、软件、网络、软件架构）、用户群、测试范围、测试优先级。

## 37 什么时候提交发现的缺陷？

测试执行发现缺陷时立即提交缺陷。

## 38 什么是入口准则、出口准则？

入口准则是进行一项测试工作前需要准备好的前提条件。

出口准则是一项测试工作可以结束的前提条件。

### 39 需求评审都有哪些人参与？

项目经理、开发经理、测试经理、测试人员、开发人员、市场经理、客户等。

### 40 怎么做需求评审或者说需求评审需要评审哪些方面？

编写或设计需求评审检查单，比如可以检查有无错别字、病句，标点符号使用是否正确，格式是否一致，是否还有多余需求，是否有错误需求，是否有遗漏需求等。

### 41 测试资源需求有哪些方面？

人力资源、硬件资源、软件资源。

### 42 什么是测试策略？什么是测试范围？

测试策略主要指如何进行某种测试（如功能测试、性能测试、兼容性测试、可用性测试、易用性测试等），用于说明测试方法以及如何使用测试方法。测试范围有时候等价于测试策略，有时候可以表示要进行测试的某个软件部位。

### 43 什么是 BVT？冒烟测试？版本验证测试？怎么测？

也称冒烟测试、版本验证测试、小版本验证测试、版本构建测试。冒烟测试用例是一组想先运行以确定这个给出的小版本是否可以测试的测试用例。冒烟测试主要测试软件的基本功能，以判断整个软件值不值得进行大规模测试。通常由一个人进行 1-2 小时的测试，一般不测试次要功能和各种错误。

### 44 测试计划的内容和目的是什么？

包含了产品概述、测试区域/测试策略/测试范围/测试目标（测试项、被测特征）、测试配置/测试资源、测试周期、进度安排（测试任务、人员安排）、测试方法/途径、测试交流、风险分析等内容。目的是指导测试过程，规定测试活动的范围、方法、资源和进度；明确正在测试的项目、要测试的特性、要执行的测试任务、每个任务的责任人以及与计划相关的风险。

### 45 怎么判断是不是软件缺陷？

- （1）软件未达到产品说明书标明的功能；
- （2）软件出现了产品说明书指明不会出现的错误；
- （3）软件功能超出产品说明书指明范围；
- （4）软件未达到产品说明书虽未指出但应达到的目标；
- （5）软件测试员具体问题具体分析，认为软件难以理解、不易使用、运行速度缓慢，或者最终用户认为不好。

## 46 缺陷的产生主要有哪些原因？最主要的原因是什么？

需求频繁变更、沟通不良、不了解客户的需求、实现新功能或很酷的功能、追求新技术、项目期限的压力、需求分析或设计投入的时间和精力不够、产品的复杂度、开发人员疲劳、压力过大或受到干扰、缺乏足够的知识、技能和经验、缺乏动力等。

最主要的原因：需求方面的原因

## 47 当你发现一个缺陷时，应该怎么确认的确是一个缺陷？

根据缺陷的判断原则来甄别发现的问题是不是一个缺陷，发现缺陷后，应该做好分离和再现（3次），然后才能提交。

## 48 在正式提交一个缺陷前，你应该做些什么？

分离缺陷、再现缺陷（3次），然后才能提交。

## 49 怎么处理无法再现的缺陷？

首先，应当对这样的缺陷进行详细的记录，并尽快提交给开发人员。

其次，对于寻找难以再现的缺陷要合理地安排时间，对一时难以再现的缺陷可以暂时搁置，以保证项目的正常进度。

最后，在测试过程中对未再现缺陷予以关注。

## 50 什么是重复缺陷？怎么避免重复缺陷？

提交了一个缺陷库中存在或者开发人员已经知道的缺陷。

1、如果缺陷是跟同事提交的重复，任务分工解决，也可以在提交之前查询下库缺陷是否存在。

2、如果缺陷是与自己提交的缺陷重复，则需要提高发现缺陷的能力，通过提高开发能力来理解两个缺陷本质上是一个缺陷。

## 51 什么是无效缺陷？怎么避免无效缺陷？

提交的缺陷不是真正的缺陷。

充分了解需求、提高自己识别缺陷的能力、提高缺陷写作能力

## 52 缺陷报告的写作准则是什么？

Correct（准确）：每个组成部分的描述准确，不会引起误解；

Clear（清晰）：每个组成部分的描述清晰，易于理解；

Concise（简洁）：只包含必不可少的信息，不包括任何多余的内容；

Complete（完整）：包含复现该缺陷的完整步骤和其他本质信息；

Consistent（一致）：按照一致的格式书写全部缺陷报告。



## 53 缺陷报告的内容有哪些？

缺陷标题（或者说缺陷摘要、缺陷概述、缺陷基本信息）

预处理

复现步骤

预期结果

实际结果

严重程度

优先级

测试环境

测试版本

测试执行人

注释

## 54 缺陷报告的组织结构是什么？

缺陷标题（或者说缺陷摘要、缺陷概述、缺陷基本信息）

预处理

复现步骤

预期结果

实际结果

严重程度

优先级

测试环境

测试版本

测试执行人

注释

## 55 缺陷报告的写作需要注意什么问题？

不要使用我、你、他等字眼，不要使用情绪化的语言和强调符号、不要使用“似乎”、看上去可能等不确定性内容、不要使用认为比较幽默的内容、不要使用不确定的测试问题(不确定是否是缺陷)、不要人身攻击。

## 56 简述缺陷报告的处理流程

软件测试人员提交缺陷报告；

测试负责人审核后将缺陷报告分配给相关的开发人员修改；

缺陷被修改后由测试人员根据缺陷报告中的修改记录进行返测

返测通过的缺陷报告由负责人关闭；

返测未通过的缺陷报告直接返回开发人员重新修改，然后再由测试人员返测，直到测试和开发达成一致处理意见。

## 57 简述缺陷的生命周期

软件测试人员提交缺陷报告；

测试负责人审核后将缺陷报告分配给相关的开发人员修改；

缺陷被修改后由测试人员根据缺陷报告中的修改记录进行返测

返测通过的缺陷报告由负责人关闭；

返测未通过的缺陷报告直接返回开发人员重新修改，然后再由测试人员返测，直到测试和开发达成一致处理意见。

## 58 简述重复缺陷的处理流程

提交缺陷→分配缺陷→是重复缺陷→置为无效缺陷。

## 59 缺陷按照严重程度可以分为哪些类型？

致命缺陷、严重缺陷、一般缺陷、较小错误、意见建议等

## 60 缺陷按照优先级可以分为哪些类型？

缺陷必须立即解决；

缺陷需要正常排队等待修复或列入软件发布清单；

缺陷可以在方便时被纠正；

下一个版本修复；

不修复。

## 61 缺陷的状态有哪些？

新建/已提交

打开

已拒绝

已解决/已修复

已验证

已关闭

## 62 测试有哪些级别？

单元测试、集成测试、系统测试、验收测试

## 63 测试有哪些阶段？

单元测试、集成测试、系统测试、验收测试

## 64 什么是单元测试？单元测试谁来做？

针对一个软件单元的测试。开发人员或懂开发的测试人员

## 65 什么是桩模块、驱动模块？

桩模块：被被测模块调用的模块。

驱动模块：调用被测模块的模块。

## 66 什么时候可以进行组件测试？

完成编译的测试对象，测试环境，开发工具，测试对象的规范说明书。

## 67 单元测试使用技术？测试重点是什么？测试条件是什么？

单元测试的技术：黑盒白盒技术，但是白盒居多，黑盒居少，一般先做黑盒再做白盒。

单元测试重点：功能性测试，健壮性（逆向测试：无效值），性能。

单元测试前提条件：完成编译的测试对象，测试环境，开发工具，测试对象的规范说明书。

## 68 什么是集成测试？

组件间的接口与交互的测试。

## 69 集成测试的测试重点是什么？测试条件是什么？使用什么技术？

接口和系统内不同部分的相互作用（交互）。

测试条件是完成集成的被测系统，测试台，有关组件间交互的文档。

测试技术包括白盒技术、黑盒技术，白盒居多，黑盒居少，对比单元测试，白盒下降，一般先做黑盒再做白盒。

## 70 集成测试有哪些策略？

自顶向下集成

自底向上集成

## 71 什么是系统测试？

对整个系统能不能满足用户需求的测试。

## 72 系统测试的目的是什么？

检查软件是否满足需求。

## 73 系统测试能够发现哪些缺陷？会遗留哪些缺陷？

发现：非功能性缺陷、涉及整个系统的问题。

遗漏：对用户的需求的错误理解、没有实现或者没有完全实现用户的隐性需求。

## 74 什么是验收测试？

一般由用户/客户进行的确认是否可以接受一个系统的验证性测试。验收测试根据用户需求，业务流程进行的正式测试以确保系统符合所有验收的准则。

## 75 验收测试有哪些人进行？

客户或用户，测试人员可以介入。

## 76 验收测试的目标是什么？

对系统或子系统建立信心、对系统非功能性的特性赢得信任。

## 77 什么是 alpha、beta 测试？有何区别？

Alpha 测试：潜在的客户/用户在开发场地进行的测试。

Beta 测试：由潜在客户/用户在自己的环境下测试软件系统。

## 78 什么是维护测试？

软件正常使用后，对软件的变更、更新进行测试

## 79 什么是性能测试？负载测试？压力测试？有什么区别？

性能表现处理速度、响应时间、CPU 使用、内存使用、硬盘使用等。

负载测试：通过不断增加负载来测试一个系统的性能。

压力测试：通过增加负载超过系统正常工作能力来考察系统能否在异常情况下正常工作

## 80 什么是功能测试？

测试一个软件能做什么，是不是做了应该做的工作，没做不该做的工作。

## 81 什么是结构测试？

白盒测试也称结构测试、逻辑驱动测试、基于程序本身的测试，是对程序结构进行的测试。

## 82 什么是与变更相关的测试？有哪些具体类型？

与变更相关的测试是对修改过的程序进行的测试。

确认测试（再测试）和回归测试。

## 83 什么是静态测试？动态测试？如何区分二者？

静态测试：不执行程序的测试。针对文档和不需执行的代码。

动态测试需要执行程序，方法一般采用黑盒测试方法和白盒测试方法。

## 84 圈复杂度怎么计算？

不重叠的闭合环数+1

## 85 什么是黑盒测试？白盒测试？

黑盒测试也称功能测试，基于规格说明书的测试，关注输入数据到程序中，输出结果是否正确，侧重于测试软件能做什么

白盒测试也称结构测试、逻辑驱动测试，是对程序内部逻辑结构进行的测试

## 86 白盒测试有哪些方法？具体解释每种方法？

白盒测试主要使用逻辑覆盖测试方法，包括语句覆盖、判定覆盖、条件覆盖、判定-条件覆盖、条件组合覆盖、路径覆盖等。

语句覆盖：程序中的每个可执行语句至少被执行一次。能发现语句错误，但不能发现逻辑错误。

判定覆盖：也称分支覆盖，程序中的每个判定的取真分支和取假分支至少执行一次。能发现逻辑错误，但不能发现组合判断中的条件错误。

条件覆盖：程序每个判定中每个条件的可能取值至少满足一次。能发现条件错误，但不能发现逻辑错误。

判定-条件覆盖：每个条件中的所有可能取值至少执行一次，同时，每个判定的可能结果至少执行一次。

条件组合覆盖：每个判定中的所有的条件取值组合至少执行一次。

路径覆盖：用例覆盖程序中的所有可能的执行路径。如果路径数很多，会变得不切实际。

## 87 什么是配置测试？

不同配置环境下进行测试。

## 88 什么是文档测试？

不仅包括测试文档校对，还有文档和软件不一致

## 89 什么是国际化测试？本地化测试？

国际性的软件

翻译成本国语言的，测试是否符合本国的语言习惯，是否符合本国法律，是否符合本国的国情。

## 90 测试用例的内容是什么？

用例编号，测试概述或用例标题、测试步骤，预期结果，输入数据，优先级，前置条件等

## 91 测试用例有哪些元素？

用例编号，测试概述或用例标题、测试步骤，预期结果，输入数据，优先级，前置条件等

或者说测试目标 Why、测试对象 What、测试环境要求 Where、测试前提 When，输入数据

## 92 什么是 UI、GUI？UI 测试什么意思？

界面

图形界面

界面测试

## 93 测试用例的优先级如何？

冒烟测试

高

中

低

## 94 解释测试目标、测试环境、测试对象、前置条件、测试策略、测试范围的含义？

测试目标：功能测试、性能测试、界面测试、易用性测试、兼容性测试、安全性测试

测试策略：某类别测试的过程、方法以及方法如何应用，测试的注意事项等

测试环境：硬件环境、软件环境、网络环境

前置条件：进行某些测试工作需要做好的准备条件

测试范围：软件需要测试的某个部位

## 95 用例评审一般使用什么方式？哪些人参与评审？

检查单。一般由测试人员进行

## 96 测试计划由谁编写？测试需求说明书由谁编写？测试用例谁编写？测试总结谁编写？

测试负责人。测试人员（测试需求分析人员）。测试人员（测试设计工程师）。测试负责人

## 97 软件投入运行后还需要测试吗？需要哪些测试？

需要测试。维护测试（含升级测试）、数据迁移测试、备份恢复测试、灾难恢复测试等

## 98 SP2 什么意思？

第 2 个版本的服务包或补丁包

## 99 给你一个网站，你如何测试？

- 首先，查找需求说明、网站设计等相关文档，分析测试需求。
- 制定测试计划，确定测试范围和测试策略，一般包括以下几个部分：功能性测试、界面测试、性能测试、数据库测试、安全性测试、兼容性测试。
- 设计测试用例：
  - ✓ 功能性测试可以包括，但不限于以下几个方面：
    - ✧ 链接测试。链接是否正确跳转，是否存在空页面和无效页面，是否有不正确的出错信息返回。
    - ✧ 提交功能的测试。
    - ✧ 多媒体元素是否可以正确加载和显示。
    - ✧ 多语言支持是否能够正确显示选择的语言等。
  - ✓ 界面测试可以包括但不限于一下几个方面：
    - ✧ 页面是否风格统一，美观
    - ✧ 页面布局是否合理，重点内容和热点内容是否突出
    - ✧ 控件是否正常使用
    - ✧ 对于必须但未安装的控制，是否提供自动下载并安装的功能
    - ✧ 文字检查
  - ✓ 性能测试一般从以下两个方面考虑：
    - ✧ 压力测试、负载测试
  - ✓ 数据库测试要具体决定是否需要开展。
    - ✧ 数据库一般需要考虑连结性，对数据的存取操作，数据内容的验证等方面。
  - ✓ 安全性测试：
    - ✧ 基本的登录功能的检查
    - ✧ 是否存在溢出错误，导致系统崩溃或者权限泄露
    - ✧ 相关开发语言的常见安全性问题检查，例如 SQL 注入等
  - ✓ 兼容性测试，根据需求说明的内容，确定支持的平台组合：
    - ✧ 浏览器的兼容性；
    - ✧ 操作系统的兼容性；
    - ✧ 软件平台的兼容性；
    - ✧ 数据库的兼容性
- 开展测试，并记录缺陷。
  - ✓ 合理的安排调整测试进度，提前获取测试所需的资源，建立管理体系（例如，需求变更、风险、配置、测试文档、缺陷报告、人力资源等内容）。
- 定期评审，对测试进行评估和总结，调整测试的内容。

## 100 一台客户端有三百个客户与三百个客户端有三百个客户对服务器施压，有什么区别？

- 300 个用户在一个客户端上
  - ✓ 会占用客户机更多的资源，而影响测试的结果。线程之间可能发生干扰，而产生一些异常。
  - ✓ 需要更大的带宽。
  - ✓ IP 地址的问题，可能需要使用 IP 欺骗来绕过服务器对于单一 IP 地址最大连接数的限制。
  - ✓ 不必考虑分布式管理的问题。
- 用户分布在不同的客户端上
  - ✓ 需要考虑使用控制器来整体调配不同客户机上的用户。
  - ✓ 需要给予相应的权限配置和防火墙设置。

## 101 试述软件的概念和特点？软件复用的含义？构件包括哪些？

- 软件是计算机系统中与硬件相互依存的另一部分，与计算机系统操作有关的计算机程序、规程、规则，以及可能有的文件、文档及数据。
- 软件复用(SoftWare Reuse)是将已有软件的各种有关知识用于建立新的软件，以缩减软件开发和维护的花费。软件复用是提高软件生产力和质量的一种重要技术。早期的软件复用主要是代码级复用，被复用的知识专指程序，后来扩大到包括领域知识、开发经验、设计决定、体系结构、需求、设计、代码和文档等一切有关方面。
- 可以被复用的软件成分一般称作可复用构件。

## 102 软件配置管理的作用？软件配置包括什么？

- 软件配置管理（Software Configuration Management，SCM）是一种标识、组织和控制修改的技术。
- 软件配置管理应用于整个软件工程过程。
- 在软件建立时变更是不可避免的，而变更加剧了项目中软件开发开发者之间的混乱。SCM 活动的目标就是为了标识变更、控制变更、确保变更正确实现并向其他有关人员报告变更。从某种角度讲，SCM 是一种标识、组织和控制修改的技术，目的是使错误降为最小并最有效地提高生产效率。
- 软件配置包括如下内容：配置项识别、工作空间管理、版本控制、变更控制、状态报告、配置审计。

## 103 什么是软件质量？

- 概括地说，软件质量就是“软件与明确的和隐含的定義的需求相一致的程度”。
- 具体地说，软件质量是软件符合明确叙述的功能和性能需求、文档中明确描述的开发标准、以及所有专业开发的软件都应具有的隐含特征的程度。
- 软件质量包括正确性、健壮性、效率、完整性、可用性、风险（产品运行）；可理解性、可维修性、灵活性、可测试性（产品修改）；可移植性、可再用性、互运行性（产



品转移)。

## 104 目前主要的测试用例设计方法是什么？

- 白盒测试：逻辑覆盖（语句覆盖、判定/分支覆盖、条件覆盖、条件-判定覆盖、多条件组合覆盖）、基本路径覆盖
- 黑盒测试：测试大纲法、场景法、等价类划分、边界值分析法、错误猜测法、判定表法、随机测试、探索性测试

## 105 软件的安全性应从哪几个方面去测试？

- 软件安全性测试包括程序、数据库安全性测试。
- 根据系统安全指标不同测试策略也不同。
- 用户认证安全的测试要考虑的问题
  - ✓ 明确区分系统中不同用户权限、系统中会不会出现用户冲突、系统会不会因用户的权限的改变造成混乱、用户登陆密码是否是可见、可复制、是否可以通过绝对途径登陆系统（拷贝用户登陆后的链接直接进入系统）、用户退出系统后是否删除了所有鉴权标记，是否可以使用后退键而不通过输入口令进入系统
- 系统网络安全的测试要考虑的问题
  - ✓ 测试采取的防护措施是否正确装配好，有关系统的补丁是否打上、模拟非授权攻击，看防护系统是否坚固、采用成熟的网络漏洞检查工具检查系统相关漏洞、采用各种木马检查工具检查系统木马情况、采用各种防外挂工具检查系统各组程序的外挂漏洞
- 数据库安全考虑的问题
  - ✓ 系统数据是否机密（比如对银行系统，这一点就特别重要，一般的网站就没有太高要求）、系统数据的完整性、系统数据可管理性、系统数据的独立性、系统数据可备份和恢复能力（数据备份是否完整，可否恢复，恢复是否可以完整）

## 106 什么是测试用例，什么是测试脚本，两者的关系是什么？

- 测试用例是为实施测试而向被测试系统提供的输入数据、操作或各种环境设置以及期望结果的一个特定的集合。
- 测试脚本是为了进行自动化测试而编写的脚本。
- 测试脚本的编写一般都需要对应相应的测试用例。

## 107 软件测试各个阶段通常完成什么工作？各个阶段的结果文件是什么？包括什么内容？

- 单元测试阶段：各独立单元模块在与系统地其他部分相隔离的情况下进行测试，单元测试针对每一个程序模块进行正确性校验，检查各个程序模块是否正确地实现了规定的功能。生成单元测试报告，提交缺陷报告。
- 集成测试阶段：集成测试是在单元测试的基础上，测试在将所有的软件单元按照概要设计规格说明的要求组装成模块、子系统或系统的过程中各部分工作是否达到或实现相应技术指标及要求的活动。该阶段生成集成测试报告，提交缺陷报告。

- 系统测试阶段：将通过确认测试的软件，作为整个给予计算机系统的一个元素，与计算机硬件、外设、某些支持软件、数据和人员等其他系统元素结合在一起，在实际运行环境下，对计算机系统进行全面的覆盖。该阶段需要提交测试总结和缺陷报告。
- 验收测试阶段：一般由用户进行测试，或者是用户委托第三方进行测试，主要验证软件是否满足用户的使用需求，提升用户的信心，出具验收测试报告。

## 108 测试人员在软件开发过程中的任务是什么？

- 尽可能早的找出系统中的 Bug；
- 避免软件开发过程中缺陷的出现；
- 衡量软件的品质，保证系统的质量；
- 关注用户的需求，并保证系统符合用户需求。

## 109 如何测试一个纸杯？

- 功能：用水杯装水看漏不漏；水能不能被喝到
- 安全性：杯子有没有毒或细菌
- 可靠性：杯子从不同高度落下的损坏程度
- 可移植性：杯子在不同的地方、温度等环境下是否都可以正常使用
- 兼容性：杯子是否能够容纳果汁、白水、酒精、汽油等
- 易用性：杯子是否烫手、是否有防滑措施、是否方便饮用
- 用户文档：使用手册是否对杯子的用法、限制、使用条件等有详细描述
- 疲劳测试：将杯子盛上水放 24 小时检查泄漏时间和情况；盛上汽油放 24 小时检查泄漏时间和情况等
- 压力测试：用根针并在针上面不断加重量，看压强多大时会穿透

## 110 测试计划工作的目的是什么？测试计划文档的内容应该包括什么？其中哪些是最重要的？

- 软件测试计划是指导测试过程的纲领性文件：
  - ✓ 领导能够根据测试计划进行宏观调控，进行相应资源配置等。
  - ✓ 测试人员能够了解整个项目测试情况以及项目测试不同阶段的所要进行的工作等。
  - ✓ 便于其他人员了解测试人员的工作内容，进行有关配合工作
- 测试计划包含了产品概述、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、风险分析等内容。借助软件测试计划，参与测试的项目成员，尤其是测试管理人员，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。
- 测试计划编写 6 要素（5W1H）：
  - ✓ why→为什么要进行这些测试；
  - ✓ what→测试哪些方面，不同阶段的工作内容；
  - ✓ when→测试不同阶段的起止时间；
  - ✓ where→相应文档，缺陷的存放位置，测试环境等；
  - ✓ who→项目有关人员组成，安排哪些测试人员进行测试；

- ✓ how→如何去做，使用哪些测试工具以及测试方法进行测试
- 测试计划和测试详细规格、测试用例之间是战略和战术的关系，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，而测试详细规格、测试用例是完成测试任务的具体战术，其中最重要的是测试策略和测试方法（最好是对计划先评审）。

## 111 您认为在测试人员同开发人员的沟通过程中，如何提高沟通的效率和改善沟通的效果？维持测试人员同开发团队中其他成员良好的人际关系的关键是什么？

- 尽量面对面的沟通，其次是能直接通过电话沟通，如果只能通过 Email 等非及时沟通工具的话，必须对要沟通的主题理解深刻以及能表达清楚。
- 运用一些测试管理工具进行管理也是较有效的方法，同时要注意在工具中对 BUG 有准确的描述。
- 在团队中建立测试人员与开发人员良好沟通中注意以下几点：
  - ✓ 一真诚、二是团队精神、三是在专业上有共同语言、四是要对事不对人，工作上
- 当然也可以通过直接指出一些小问题，而不是进入 BUG 跟踪系统来增加对方的好感。

## 112 你对测试最大的兴趣在哪里？为什么？

- 回答这个面试题，没有固定统一的答案，但可能是许多企业都会问到的。提供以下答案供考：
  - ✓ 最大的兴趣，感觉这是一个有挑战性的工作；
  - ✓ 测试是一个经验行业，工作越久越能感觉到做好测试的难度和乐趣
  - ✓ 通过自己的工作，能使软件产品越来越完善，从中体会到乐趣
- 回答此类问题注意以下几个方面：
  - ✓ 尽可能的切合招聘企业的技术路线来表达你的兴趣，例如该企业是数据库应用的企业，那么表示你的兴趣在数据库的测试，并且希望通过测试提升自己的数据库掌握能力。
  - ✓ 表明你做测试的目的是为了提升能力，也是为了更好的做好测试；提升能力不是为了以后转开发或其他，除非用人企业有这样的安排。
- 不要过多的表达你的兴趣在招聘企业的范畴这外。
  - ✓ 比如招聘企业是做财务软件的，可是你表现出来的是对游戏软件的兴趣；或招聘是做 JAVA 开发的，而你的兴趣是在 C 类语言程序的开发。

## 113 什么是测试覆盖率

- 是指完成的测试工作目标量占总目标量的百分比，有很多分类。
- 软件测试覆盖率常用的计算公式：
  - ✓ 功能覆盖率=至少被执行一次的测试功能点数/测试功能点总数（功能点）
  - ✓ 需求覆盖率=被验证到的需求数量/总的需求数量（需求）
  - ✓ (用例)覆盖率=至少被执行一次的测试用例数/应执行的测试用例总数
  - ✓ 语句覆盖率=至少被执行一次的语句数量/有效的程序代码行数

- ✓ 判定覆盖率=判定结果被评价的次数/判定结果总数
- ✓ 条件覆盖率=条件操作数值至少被评价一次的数量/条件操作数值的总数

## 114 一个好的测试用例，有哪些特点

- 用例要完整、简洁、一致
  - ✓ 至少含有编号、标题、操作步骤和预期结果。
- 用例要表明测试目的
- 用例覆盖程度要高
- 用例能够使工作量最小化
- 用例描述正确、规范
  - ✓ 含有正确的、规范的测试标题和编号
- 用例的分类以及描述要足够清晰
- 用例要具有可测试性
- 测试用例易于维护
  - ✓ 如果被测对象有所升级，测试用例的说明或者脚本是不是容易维护呢？
- 可复用
- 可重复性
  - ✓ 不管谁执行此用例，结果一样。
- 可追踪性
  - ✓ 用例能追踪到一个具体的需求。

## 115 测试结束的标准是什么

- 全部测试用例都执行完成。
- 未修改 bug 都被确认或置为应有状态，暂缓修改的问题都有详尽的解释。
- 测试报告编写完成。
- 测试收尾工作结束。
- 测试总结完成。
- 项目处于试运行或上线阶段
- 在测试计划中定义结束标准
  - ✓ 如计划中规定：系统在一定性能下平稳运行 72 小时，本版本中没有严重的 BUG，普通 BUG 的数量在 3 以下，BUG 修复率 90%以上
  - ✓ 实际测试达到上述要求，然后由开发经理，测试经理，项目经理共同签字，认同测试结束，版本即可发布。

## 116 如何全面测试一款产品，请以手机短信功能为例来辅助说明，前

提是手机自带的短信功能，并非微信，QQ 这种软件

- 能正常打开或进入短信界面
- 短信可以正常编辑、修改、删除
- 短信可以正常发送和接收
- 短信页面字体、颜色显示正常
- 短信字体可调整

- 给多人同时发短信
- 给特殊号码发送短信
  - ✓ 如运营商（手机号所属运营商、其他运营商）
  - ✓ 不存在手机号
  - ✓ 服务号（收费、不收费）
- 接收验证码
- 短信耗电量测试
- 短信不耗流（联网、不联网）
- 短信干扰测试
  - ✓ 编辑短信期间（已经编辑好、正在打字），电话进来
  - ✓ 编辑短信期间（已经编辑好、正在打字），收到短信
  - ✓ 隐藏到后台（已经编辑好、正在打字），进行其他操作，再返回

## 117 你手中的这支笔有多少用途，请发挥你的想象力

- 写、画（纸上、墙上、桌子上、地上、其他位置）
- 染色
- 承重
- 当书签
- 当筷子
- 用来扎人
- 用来掏掉进笔记本键盘的小东西
- 用来去除缝隙（比如手机上的缝隙）里的灰尘
- 当作燃料
- 拿在手里转着玩，消遣一下
- 用来碰电门
- 塑料笔管用来当吸管

## 118 判断字符串回文

回文序列是指正序和反序都相同的字符串，如“A”，“BAB”，“BFFB”。实现一个函数，判断输入的字符串是否为回文，并写出测试用例。如有可能请使用你最熟悉的编程语言实现这个函数。

需求分析：回文是一个字符串，正反都一样

举例子找规律：

奇数个字符：	A	BAB	BADAB
		012	01 234
		[0] = [2]	[0] = [4]、[1] = [3]
	总共 n 个字符，第 i 个 = 第 n-1-i 个，需要判断 [n/2 下取整] 次		
	下一个是 1，结束      下一个是 2，结束		
偶数个字符：	AA	ABBA	ABCCBA
	01	0123	012 345
	[0] = [1]	[0] = [3]、[1] = [2]	[0] = [5]、[1] = [4]、[2] = [3]
	总共 n 个字符，第 i 个 = 第 n-1-i 个，需要判断 n/2 次		

总体的规律：

总共  $n$  个字符，第  $i$  个=第  $n-1-i$  个，需要判断  $n/2$  下取整次  
思路：

需要一个字符串参数，表示回文

循环判断，循环次数= $n/2$

如果第  $i$  个=第  $n-1-i$  个，下一个是  $n/2$  结束

否则，直接退出

测试：

空串、一个字符、两个字符（对称、不对称）、五个字符（对称、不对称）

AA

AB

ABCBA、ABCBW、ABCWA、ABCDE

java 代码：

```
public static boolean message(String s){
    int n=s.length();
    boolean jg=false;
    if(n!=0)
        for(int i=0;i<=n/2;i++){
            if(s.charAt(i)==s.charAt(n-1-i)){
                //System.out.println(i+" "+s.charAt(i)+" "+s.charAt(n-1-i));
                if(i==n/2)
                    jg=true;
            }
            else{
                jg=false;
                break;
            }
        }
    return jg;
}
```

白盒角度：

这里的条件覆盖等价于判定覆盖， $n!=0$  成立与不成立均测到

for: 循环 0 次不现实，所以不需要测，1 次以上

$s.charAt(i)==s.charAt(n-1-i)$  的条件覆盖等价于判定覆盖

$i==n/2$  的条件覆盖等价于判定覆盖

python 代码：

```
s="ABCVE"
n=len(s)
if(n==0):
    print("空字符串不是回文")
else:
    #print(int(n/2))
    for i in range(0,int(n/2)+1):
        if(s[i]==s[n-1-i]):
            if(i==int(n/2)):
                print(s+"是回文")
    else:
        print((s+"不是回文"))
```

break

**119 Function (A, B, C)** 是一个判断三角形形状的函数，其中 A, B, C 是三个变量，分别代表三角形的三个边，请设计一组测试用例，来测试这个函数是否正确。如有可能，请使用你最熟悉的编程语言实现这个函数。

思路：

三个参数 (double)

使用异常解决输入的不是数的问题

三个边都是正的

三边符合关系

三角形类型 (等边、等腰、直角、普通)

java 代码：

```
public static String function(double a,double b,double c){
    String jg;
    if(a>0 && b>0 && c>0)
        if(a+b>c && a+c>b && b+c>a)
            if(a==b && a==c)
                jg="等边三角形";
            else if(a==b || a==c || b==c)
                jg="等腰三角形";
            else if(a*a+b*b==c*c || a*a+c*c==b*b || c*c+b*b==a*a)
                jg="直角三角形";
            else
                jg="普通三角形";
        else
            jg="不能构成三角形";
    else
        jg="三边必须都是正数";
    return jg;
}
```

用例设计：

场景：

三边非全正 (1、2、3 为 0 或负数)

三边关系不满足 (相等, 小于)

三边等

两边等 (3 条)

直角 (3 条)

普通

基本流用例：

3 3 3  
3 3 4  
3 4 3  
4 3 3  
3 4 5  
3 5 4  
5 3 4  
3 4 6  
1 2 3  
1 2 4  
1 3 2  
1 4 2  
3 1 2  
4 1 2

备选流用例:

1 1 0  
1 1 -1  
1 0 1  
1 -1 1  
0 1 1  
-1 1 1  
1 0 0  
1 0 -1  
1 -1 0  
1 -1 -1  
0 0 1  
-1 0 1  
0 -1 1  
-1 -1 1  
0 1 0  
0 1 -1  
-1 1 0  
-1 1 -1

## 120 常见的性能测试策略有哪些

负载测试、压力测试、并发测试、内存泄露测试、基准测试、配置测试、数据容量测试、疲劳强度测试

## 121 请查询出 s\_dept 表中区域（region\_id）为 1、3 的部门信息。

```
select * from s_dept where region_id in('1','3')
```



**122 请查询出 s\_emp 表中工资 (salary) 在 1500 到 2000 之间的员工信息。**

```
select * from s_emp where salary between 1500 and 2000
```

**123 请查询出 s\_emp 表中姓名 (name) 中含有字母 a 的员工信息。**

```
select * from s_emp where name like '%a%'
```

**124 请从 t\_contrast\_master 保单表和 t\_policy\_prem 费用表中查询出满足以下条件的保单：费用表的 fee\_tpye=24, fee\_status=0, 且 due\_time 日期小于 2012 年 10 月 29 日，保单表的 organ\_id 以 111 开头的且 suspend="N"、两张表用 policy\_id 关联。**

```
select * from t_contrast_master M join t_policy_prem P on M.policy_id=P.policy_id where fee_type=24 and fee_status=0 and due_time<'2012/10/29' and organ_id like '111%' suspend='N'
```

**125 为以下程序段设计一组测试用例，要求分别满足语句覆盖、判定覆盖、条件覆盖。int test (int A, int B) { if ((A>1) AND (B<10)) then X=A-B; if ((A=2) OR (B>20)) then X=A+B; returnx; }**

```
int test (int A, int B) {  
    if (A>1 AND B<10) then  
        X=A-B;  
    if (A=2 OR B>20) then  
        X=A+B;  
    returnx;  
}
```

语句覆盖：

满足要求 A>1 AND B<10 且 A=2 OR B>20

用例：A=2, B=5

判定覆盖：

满足要求：

A>1 AND B<10 真假各一次

A=2, B=5, 判定为真

A=0, B=5, 判定为假

A=2 OR B>20 真假各一次

A=2, B=5, 判定为真

A=0, B=5, 判定为假

条件覆盖:

满足要求: A>1 真假各一个, B<10 真假各一次, A=2 真假各一次, B>20 真假各一次

用例:

A=2, B=5	A>1 真, B<10 真	A=2 真, B>20 假
A=0, B=21	A>1 假, B<10 假	A=2 假, B>20 真

**126 请通过编写 sql 或编写存储过程的方式完成如下工作。表 a 中有保单信息, 包括保单 id、保费等信息, 现在对保单进行分红, 分红金额为保费的 3%, 如果分红金额大于 5000 元, 则取 5000 元进行分红。**

假设分红有上限 5000:

```
select id,5000 from 保单 where id in(select id from 保单 where 保费*0.03>=5000)
```

假设分红允许是 5000 的倍数:

```
select id,floor( 保费 *0.03/5000)*5000 from 保单 where id in(select id from 保单 where 保费*0.03>=5000)
```

**127 你自认为测试的优势在哪里?**

- 该面试题也没有固定不变的答案, 但可参考以下几点, 并结合自身特点:
  - ✓ 有韧性、有耐心、做事有条理性、喜欢面对挑战、有信心做好每一件事情、较强的沟通能力、从以前的经理处都得到了很好的评价表明我做的很好

**128 简述你在以前的工作中做过哪些事情, 比较熟悉什么。**

- 参考答案如下
  - ✓ 我过去的主要工作是系统测试和自动化测试。
    - ✧ 在系统测试中, 主要是对\*\*系统的业务逻辑功能, 以及系统的\*\*特性进行测试。
    - ✧ 性能测试中, 主要是进行的压力测试, 在各个不同数量请求的情况下, 获取系统响应时间以及系统资源消耗情况。
    - ✧ 自动化测试主要是通过自己写脚本以及一些第三方工具的结合来测试\*\*的特性测试。
  - ✓ 在测试中, 我感觉对用户需求的完全准确的理解非常重要。另外, 就是对 BUG 的管理, 要以需求为依据, 并不是所有 BUG 均需要修改。
  - ✓ 测试工作需要耐心和细致, 因为在新版本中, 虽然多数原来发现的 BUG 得到了修复, 但原来正确的功能也可能变得不正确。因此要注重迭代测试和回归测试。

## 129 Internet 采用哪种网络协议？该协议的主要层次结构？Internet

### 物理地址和 IP 地址转换采用什么协议？

- TCP/IP 协议
  - ✓ TCP：传输控制协议
  - ✓ IP：网络协议/网际协议
  - ✓ 主要层次结构为：应用层/传输层/网络层/数据链路层
- ARP (Address Resolution Protocol)
  - ✓ 地址解析协议

## 130 说说你对集成测试中自顶向下集成和自底向上集成两个策略的理解，要谈出它们各自的优缺点和主要适应于哪种类型测试

- 自顶向下集成
  - ✓ 优点：较早地验证了主要控制和判断点；按深度优先可以首先实现和验证一个完整的软件功能；功能较早证实，带来信心；只需一个驱动，减少驱动器开发的费用；支持故障隔离。
  - ✓ 缺点：桩的开发量大；底层验证被推迟；底层组件测试不充分。
  - ✓ 适应于产品控制结构比较清晰和稳定；高层接口变化较小；底层接口未定义或经常可能被修改；产品控制组件具有较大的技术风险，需要尽早被验证；希望尽早能看到产品的系统功能行为。
- 自底向上集成
  - ✓ 优点：对底层组件行为较早验证；工作最初可以并行集成，比自顶向下效率高；减少了桩的工作量；支持故障隔离。
  - ✓ 缺点：驱动的开发工作量大；对高层的验证被推迟，设计上的错误不能被及时发现。
  - ✓ 适应于底层接口比较稳定；高层接口变化比较频繁；底层组件较早被完成。

## 131 系统测试的策略有哪些

有功能测试、性能测试、负载测试、强度/压力测试、易用性测试、安全测试、配置测试、安装测试、文档测试、故障恢复测试、用户界面测试、可用性测试。

## 132 请说出这些测试最好由那些人员完成，测试的是什么？

- 代码、函数级测试
  - ✓ 一般由白盒测试人员完成，他们针对每段代码或函数进行正确性检验，检查其是否正确的实现了规定的功能和性能。
- 模块、组件级测试
  - ✓ 主要依据是程序结构设计测试模块间的集成和调用关系，一般由白盒测试人员完成，接口测试可以由测试人员完成。
- 系统测试
  - ✓ 在于模块测试与单元测试的基础上进行测试。了解系统功能与性能，根据测试用例

进行全面的测试。由系统测试人员完成。

### 133 在 windows 下保存一个文本文件时会弹出保存对话框，如果为文件名建立测试用例，等价类应该怎样划分？

- 分析需求
  - ✓ Windows 中文件名长度除去盘符，最长 255 字符，不区分中英文；不能使用 \: \* ? " < > | , \_ . com0-com9、lpt0-lpt9、lpt、nul、aux 等字符，不允许文件重名（询问替换还是附加(1)）
- 有效类
  - ✓ 25 个字符
- 无效类
  - ✓ 空、含有非法字符、输入 256 字符

### 134 假设有一个文本框要求输入 10 个字符的邮政编码，对于该文本框应该怎样划分等价类？

- 分析需求
  - ✓ 允许使用哪些字符（大写字母、小写字母、数字、标点符号），哪些字符具有特殊用途（保留字符），字符可以混合使用吗，是否可以留空
- 有效
  - ✓ 英文字母，如 ABCDefghik、数字和其他混合，如 123AAAAAAA、数字：1234567890
- 无效
  - ✓ 特殊字符，如 10 个\*或¥，小于十个字符，如 123；大于十个字符，如 11111111111；空字符；保留字符

### 135 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趋势，缺陷发现的越晚，修复它所花费的成本就越大。

### 136 怎么理解回归测试？

- 回归测试: (regression testing)有两类：用例回归和错误回归；
  - ✓ 用例回归是过一段时间以后再回头对以前使用过的用例在重新进行测试，看看会不会重新发现问题。
  - ✓ 错误回归，就是在新版本中，对以前版本中出现并修复的缺陷进行再次验证，并以缺陷为核心，对相关修改的部分进行测试的方法。

### 137 你认为做好测试计划工作的关键是什么？

- 明确测试的目标，增强测试计划的实用性
- 软件测试计划中的测试范围必须高度覆盖功能需求，测试方法必须切实可行，测试工具

并且具有较高的实用性，便于使用，生成的测试结果直观、准确

- 坚持“5W”规则，明确内容与过程
  - ✓ “5W”规则指的是“What（做什么）”、“Why（为什么做）”、“When（何时做）”、“Where（在哪里）”、“How（如何做）”。利用“5W”规则创建软件测试计划，可以帮助测试团队理解测试的目的（Why），明确测试的范围和内容（What），确定测试的开始和结束日期（When），指出测试的方法和工具（How），给出测试文档和软件的存放位置（Where）。
- 采用评审和更新机制，保证测试计划满足实际需求
- 分别创建测试计划与测试详细规格、测试用例

**138 您是否了解以往所工作的企业的软件开发过程？如果了解，请试述一个完整的开发过程需要完成哪些工作？分别由哪些不同的角色来完成这些工作？您在以往的测试工作中都曾经具体从事过哪些工作？其中最擅长哪部分工作？**

- 开发过程
  - ✓ 需求调研（需求人员）、需求分析（需求人员）、概要设计（设计人员）、详细设计（设计人员）、编码（开发人员）
- 测试过程
  - ✓ 需求评审、系统测试设计、概要设计评审、集成测试设计、详细设计评审、单元测试设计、测试执行
- 测试工作的整个过程都做过，擅长做测试设计...

**139 LoadRunner 分为哪三个模块？请简述各模块的主要功能。**

- Virtual User Generator
  - ✓ 虚拟用户发生器
  - ✓ 用于录制脚本、调试脚本、增强脚本、运行脚本
- Controller
  - ✓ 控制器
  - ✓ 用于创建、运行和监控场景
- Analysis
  - ✓ 分析
  - ✓ 用于分析测试结果

**140 针对于软件的行业背景，你如何理解软件的业务？**

- 阅读用户手册了解软件的功能和操作流程；
- 看一些业务的专业书籍补充业务知识；
- 如果有用户实际的数据，可以拿实际的数据进行参考；
- 参考以前的用例和 BUG 报告；
- 在使用软件的过程中多思考；

- 多与产品经理交流。

## 141 如何定位测试用例的作用？

- 测试用例是测试人员在测试过程中的重要参考依据。
- 测试用例可以帮助实施有效的测试，所有被执行的测试都是有意义的，不要执行毫无意义的测试操作。
- 良好的测试用例不断地被重复使用，使得测试过程事半功倍。
- 测试用例是一个知识积累的过程。
- 测试用例是一个知识传递的过程，能保持一致、稳定的测试质量
- 从项目管理的角度来说，测试用例的通过率是检验代码质量保证效果最主要的指标之一。
- 测试用例也可以作为评估测试人员进度、工作量以及跟踪/管理测试的工作效率的主要因素，从而更加合理地做出测试安排或调整。

## 142 需求测试的注意事项有哪些？

- 是否使用了公司的模板
- 文档内容是否符合规范
- 所有的需求分级是否清晰适当
- 所有的需求是否具有一致性
- 需求是否可行
- 需求是否足够
- 所有的其它需求是交叉引用是否正确
- 用户描述是否清楚
- 是否用客户的语言来描述需求
- 每个需求描述是否清楚没有歧义
- 是否所有的需求都是可验证
- 是否每条需求都具有独立性，即使发生了变化也不会影响其它需求
- 性能指标是否明确
- 非功能性需求是否得到充分表现
- 是否完整列出适用的标准或协议、标准和协议之间是否存在冲突等

## 143 主键、外键的作用，索引的优点与不足？

- 主键
  - ✓ 是表中的唯一标示键。
  - ✓ 作用：保证实体的完整性，加快数据库的操作速度，增加新的表记录时，数据库会自动检索新记录的主键值，不允许该值与其他表中记录的主键重复，数据库会按主键值的顺序显示记录，如果没有设定主键，则按输入的顺序显示记录。
- 外键
  - ✓ 是主键的从属，表示了两个表之间的联系。
  - ✓ 作用：使用外键可以建立数据之间的关联，还可以避免冗余。
- 索引
  - ✓ 通过创建唯一性的索引，可以保证表中数据的唯一性
  - ✓ 加速数据的检索速度

- ✓ 加快表与表之间的连接
- ✓ 在使用分组与排序数据检索时，可以显著检索分组与排序的时间
- ✓ 在查询的过程中使用优化隐藏器，提供系统性能。
- ✓ 缺点
  - ◆ 创建索引需要时间，且随着数据量的增加而增加
  - ◆ 索引需要占用物理空间
  - ◆ 当对表中数据进行修改时，索引也要动态维护，降低了数据的维护速度

## 144 性能测试的流程？

- 测试需求分析
- 测试计划制定与评审
- 测试用例设计与开发，编写测试脚本
- 测试执行与监控，开发场景
- 分析测试结果
- 编写性能测试报告
- 测试经验总结

## 145 你接触过正则表达式吗？怎么校验数字？

- 可以使用[0-9]表示 1 位数字，也可以使用\d 表示 1 位数字
- 使用多对[]可以表示多位，也可以使用{重复次数}来表示，有的软件需要使用\{进行转义
- 比如 6 位数字，检验写法有
  - ✓ [0-9][0-9][0-9][0-9][0-9][0-9]
  - ✓ \d{6}

## 146 什么是 BS 架构？什么是 CS 架构？

- BS 是浏览器/服务器架构，需要通用客户端，主要压力在服务器
- CS 是客户端/服务器架构，需要专用客户端，客户端承担一部分工作和压力

## 147 什么是 OO 思想？

- Object Oriented
- 面向对象的思想
  - ✓ 以数据为核心
  - ✓ 将问题分解为不同的事物或类和对象，考虑类和对象的特征和行为
  - ✓ 编程时，创建类，类包含属性和方法，属性反应所有对象的共同特征，方法反应所有对象的公共行为
  - ✓ 创建对象，调用方法

## 148 什么是 JRE？什么是 JDK？

- JRE 是 java 运行时环境，内含 java 虚拟机和类库，用于运行 java 程序，一般需要安装

到需要运行 java 程序的客户端

- JDK 是 java 开发包，内含 jre 和开发 java 程序的类库，一般用于开发 java 程序

## 149 Java 的三大特征分别是什么？

- 继承
  - ✓ 可以共享或分享其他类的代码，实现代码复用，一般使用 `extends` 关键字实现
- 多态
  - ✓ 表示类和对象有多种形态，以实现处理的多样化，多态可以使用类数组、方法重载、方法重写等实现
  - ✓ 类数组是定义数组时使用父类，而元素对象使用子类
  - ✓ 方法重载是方法名相同，但是参数的个数、类型、顺序不同
  - ✓ 方法重写是子类继承父类后，重新编写了与父类同名且参数完全相同的方法
- 封装
  - ✓ 是将类的属性进行的“保密性”处理，不允许属性的使用者通过属性来访问和操作属性，需要通过访问器和设置器才可以使用
  - ✓ 一般可以将属性使用 `public` 以外的修饰符修饰就可以实现不同程度的封装，封装力度最大的是 `private` 关键字，只允许本类可以直接使用

## 150 成员变量用 `static` 修饰和不用 `static` 修饰有什么区别？

- 使用 `static` 的成员变量被称为类变量，表示属性值是公共的，比如大家的学校、国籍等，占用的内存是一段所有对象共享的内存，仅此一份；如果一个对象更改了 `static` 变量的值，其他对象的此属性值也会发生变化，而无 `static` 的属性隶属于每个对象，占用内存独立，相互不影响

## 151 如果变量用 `final` 修饰，则会如何？如果方法用 `final` 修饰，则会如何？如果类用 `final` 修饰，则会如何？请举例说明你见过哪些异常？

- `final` 修饰变量，称为终态变量，表示这个变量是常量，后面的代码以及其他类和对象将不允许修改此变量的值
- `final` 修饰的方法表示这是一个终态方法，不允许子类去重写
- `final` 修饰的类为终态类，不允许其他类继承 `final` 修饰的类
- 常见的异常
  - ✓ `ArithmeticException`
    - ◆ 算数异常，比如除以 0
  - ✓ `InputMismatchException`
    - ◆ 数据类型不匹配异常
  - ✓ `FileNotFoundException`
    - ◆ 文件不存在、写文件时文件只读



## 152 请编码实现如下功能：将字符串” I Love China” 存储到硬盘上

- `FileWriter writer=new FileWriter(文件名,true)`
  - ✓ `true` 表示追加数据
- `writer.write(字符串)`
- `writer.close()`
  - ✓ `FileWriter file=new FileWriter("e:/a.txt",true);`
  - ✓ `String a="hello zhsan\r\nhow are you";`
  - ✓ `file.write(a);`
  - ✓ `file.close();`

## 153 你了解几种约束？

- 主键
  - ✓ 为了保证实体完整性，具体是为了保证表中的实体或行的唯一性，控制表中的行不重复
  - ✓ 不允许空值
  - ✓ 不允许重复
  - ✓ 一个表中只允许最多一个主键
- 唯一键
  - ✓ 为了保证实体完整性，具体是为了保证表中某个列下的行数据不重复
  - ✓ 有的数据库允许空值
  - ✓ 不允许重复
- check
  - ✓ 检查约束
  - ✓ 通常为了空值列的行数据必须符合的要求，比如范围要求，例如性别只能输入男或女，年龄只能输入 0-100
- 非空
  - ✓ 表示数据不能为空，必须输入数据
- 外键
  - ✓ 通常是为了与其它表建立关联
  - ✓ 一个表中设置了主键，这个表称为主表，外键所在的表（表称为外表、相关表）中外键列的行数据可以有重复，对应的外键值来源于主表中主键列下的数据
- 默认值
  - ✓ 当表中有大量重复数据时，或者是有些数据难以人为输入的时候，比如时间（尤其精确度要求很高）时，就可以默认值

## 154 你接触过哪些数据库对象？你使用过哪些系统函数？

- 数据库、表、查询、视图、存储过程、函数、触发器、索引
- `getdate()`可以获得日期时间（sqlserver 函数）、`sysdate` 获得系统日期时间的函数（oracle）
- `len` 或 `length` 获得字符串长度
- `left`、`right`、`substring/substr` 用于获得字符串的一部分（子串）
- `trim` 去掉字符串首尾的空格

- count 统计行数、sum 计算列下的和、avg 计算列下的均值、max 计算列下的最大值、min 计算列下的最小值

## 155 存储字符串时，使用 char 还是 nchar？使用 char 还是 varchar2？为什么？

- char 是定长字符串类型，nchar 双字节定长字符串类型，varchar2 是可变长字符串类型
- char 以字节形式存储，适合存储英文字符，一个汉字存储为 2 个字节，如果数据量小于指定的长度，则补空格
- nchar 是以字符形式处理，适合存储类似于汉字的这种字符，一个字符在内存或硬盘中也是以字节形式存储，如果数据量小于指定长度，仍然补空格
- varchar2 是以字节形式存储，如果数据量超过指定的长度，计算机会自动收回未用空间。

## 156 在 oracle 数据库中有如下的一张表：

表名：user 表结构

id:char (20)

name:varchar (50)

gender:char (1) (本字段的值为 F 或者 M)

email:varchar (50)

address:varchar (200)

- 共有 1000 条记录
- 现在要取出性别为女的第 100 至 110 条记录，请给出 sql 语句
  - ✓ select \* from (select rownum num,id,name,gender,email,address from user where gender='F') where num between 100 and 110

## 157 你接触过哪些标签？

- head、title、body
- table、tr、td
- a
- hr
- center
- font
- b、i、u
- form、input(type=text/password/checkbox/radio/submit/reset/button/textarea)
- span
- div
- img

## 158 你接触过哪些浏览器对象？你接触过哪些脚本对象？

- ie
- 火狐

- google chrome
- qq 浏览器
- uc
- 遨游
- ....
- vbs
- js

## 159 Form 标签的 action 属性可以赋什么值？有什么作用？

- <form action=?>
- 一般是写跳转到的页面，表示本页面给 action 后指定的页面发送数据，把表单数据发送给指定的 action 去处理
- 可以省略，表示发送数据给自己处理

## 160 Form 标签的 method 属性可以赋什么值？有什么作用？

- <form method=?>
- 可以 get、post、delete...
- 常用的是 get 和 post
- 用于指定发送数据的方式
- get 发送数据显示在地址栏中，数据量有限制，只能发送二进制形式，可以缓存
- post 发送数据不显示在地址栏，数据量无限制，可以发送出二进制以外的其它形式，数据不可以缓存，比 get 安全

## 161 js 如何编程获取名字为“game”的复选框所有选中的值？

- <form>
- <input type="checkbox" name="game" value=111>111
- <input type="checkbox" name="game" value=222>222
- <input type="checkbox" name="game" value=333>333
- <input type="checkbox" name="game" value=444>444
- <input type="checkbox" name="game" value=555>555
- <input type="button" value="click" onclick="show()">
- </form>
- 
- <script>
- function show(){
- items = document.getElementsByName("game");
- data = [];
- for(x in items){
- if(items[x].checked)
- data.push(items[x].value);
- }
- alert(data);

- }
- </script>

## 162 功能测试中常用的测试工具有哪些？做简要说明

- 缺陷管理工具：bugfree、mantis、zentao
- 测试管理工具：QC/ALM
- 版本控制工具：SVN
- 自动化测试工具：QTP（大型收费）、Selenium（免费开源）

## 163 功能测试中常用的测试方法有哪些？做简要说明

- 场景法
  - ✓ 主要对业务流程进行测试
  - ✓ 将业务流区分为基本流和备选流，基本流按照正确的流程设计用例，备选流测试各种错误流程
- 等价类划分
  - ✓ 将输入区域划分为若干个等价的类别，具体的分为有效等价类、无效等价类
  - ✓ 有效等价类是依据规格说明书中的要求设计用例
  - ✓ 无效等价类是违反规格说明书中的要求进行设计用例
- 边界值分析
  - ✓ 依据输入数据的范围进行用例设计，一般选择边界点上的 4 个或 6 个值进行用例设计
  - ✓ 通常选择比最小值正好小一点的值、最小值、最大值、比最大值刚刚大一点的数据作为用例
- 决策表
  - ✓ 分析不同输入的各种组合产生不同的输出
- 错误猜测
  - ✓ 考虑各种对软件的攻击

## 164 简单说一下对冒烟测试/集成测试的了解

- 冒烟测试
  - ✓ 也称为小版本验证测试
  - ✓ 对软件基本功能的验证
  - ✓ 通过对软件基本功能的测试，确定能否进行后续大规模的测试，如果冒烟测试通过，则后续可以进行大规模测试，否则将软件打回开发
  - ✓ 一般，冒烟测试由一个人进行，通常耗时 1-2 小时
  - ✓ 不同测试类型的冒烟测试可能会有不同，比如性能测试的冒烟测试需要的人力和时间也许会拉长
- 集成测试
  - ✓ 主要是测试软件多个单元合并到一起时需要的测试
  - ✓ 通常测试软件各部件之间的接口和交互
  - ✓ 一般由多个懂开发的测试人员进行

## 165 如何判断一个系统是否通过冒烟测试？

- 软件的基本功能或者重要功能已经正确实现

## 166 简单说一下 bug 等级是如何确定的？

- 致命
  - ✓ 一般会引发蓝屏、死机、人身生命安全威胁、非常重要的功能没有实现或未正确实现、数据库死锁、数据库连接失败、因为错误操作导致程序中断
- 严重
  - ✓ 重要功能没实现或者出现错误、接口出错
- 一般
  - ✓ 常用功能没实现或者出错，界面错误、出错不提示、输入限制不在前台实现
- 小
  - ✓ 界面不规范，长时间操作不提示、文字不使用专业术语、必填项与可输入项没有明确区分
- 意见和建议

## 167 解释数据库中 Join 的用法

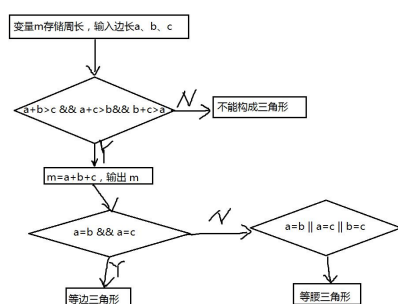
- join 用于连接多个表
- 一般需要指定连接条件
- 语法是
  - ✓ select 列名列表 from 表 1 join 表 2 on 表 1.列名=表 2.列名
- 包括内连接、外连接、自然连接、自连接
  - ✓ 内连接
    - ◆ 表示显示符合连接条件的数据
  - ✓ 外连接
    - ◆ 表示一个表显示全部数据，另一个表只显示符合条件的数据
    - ◆ 全连接：双方的表数据全部出现，符合连接条件的数据双方表合并为一行显示在结果中，不符合要求的数据对方列显示 null
  - ✓ 自然连接
    - ◆ 要求两个表必须具有同名列
    - ◆ 以同名列的值相等为条件，进行自动连接
  - ✓ 自连接
    - ◆ 需要自己写 sql 语句实现
    - ◆ 表示表自己与自己相连

## 168 Sql 中排序、编辑、包含的关键字是？

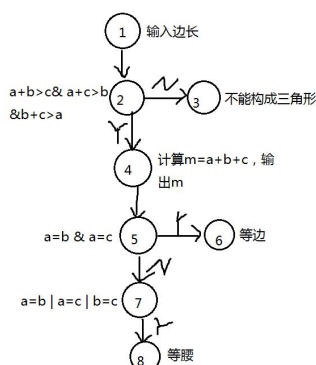
- 排序使用 order by
- 修改数据使用 update
- 如果表示数据集中有没有数据使用 in

169 在三角形计算中，要求三角形的三个边长：A，B 和 C。当三边不可能构成三角形时提示错误，可构成三角形时计算三角形周长。若是等腰三角形打印“等腰三角形”，若是等边三角形，则提示“等边三角形”。画出程序流程图、控制流程图、找出基本测试路径，对此设计一个测试用例。

➤ 程序流程图



➤ 控制流图



➤ 测试路径

✓ 语句覆盖/路径覆盖

◆ 124578

- 用例：3、3、4

◆ 123

- 用例：1、2、3

◆ 12456

- 用例：3、3、3

✓ 判定覆盖

◆ 用例同上

✓ 条件覆盖

◆ 条件  $a+b>c$ 、 $a+c>b$ 、 $b+c>a$ 、 $a=b$ 、 $a=c$ 、 $b=c$

◆ 都成立的用例：3、4、5

◆ 都不成立的用例：0、0、0

✓ 判定-条件覆盖

- ◆ 上边判定+条件
- ✓ 条件组合
  - ◆ 条件  $a+b>c$ 、 $a+c>b$ 、 $b+c>a$ 
    - $a+b>c$  真、 $a+c>b$  真,  $b+c>a$  真
      - 用例: 3、4、6
    - $a+b>c$  真、 $a+c>b$  真,  $b+c>a$  假
      - 用例: 2、1、0
    - $a+b>c$  真、 $a+c>b$  假,  $b+c>a$  真
      - 用例: 0、2、1
    - ...
  - ◆ 条件  $a=b$ 、 $a=c$ 、 $b=c$ 
    - $a=b$  真、 $a=c$  真
      - 用例: 3、3、3
    - $a=b$  真、 $a=c$  假
      - 用例: 3、3、4
    - $a=b$  假,  $a=c$  真
      - 用例: 3、4、3
    - $a=b$  假,  $a=c$  假
      - 用例: 3、4、6

## 170 编写一个程序显示 1-100 之间的所有素数（语言不限）

### ➤ 思路:

- ✓ 需求
  - ◆ 质数/素数: 一个正整数, 只能被 1 和自己整除 (整除:  $8/2$ ,  $9/2$ ), 1 不是素数
  - ◆ 注意: 任何数除以 1 都能除开, 任何数除以自己也都能除开
  - ◆ 根据定义, 2 是素数
- ✓ 举例子规律
  - ◆ 3:  $3\%2!=0$ , 输出 3
  - ◆ 4:  $4\%2=0$ , 下一个数
  - ◆ 5:  $5\%2!=0$ ,  $5\%3!=0$ ,  $5\%4!=0$ , 输出 5
- ✓ 算法
  - ◆ 输出 2
  - ◆ 循环: 左边 3~100
    - 循环: 右边 2~左边-1
      - 判断左边%右边=0
        - ◆ break
      - 否则
        - ◆ 判断右边=左边-1
          - 输出
        - ◆ 否则, 继续循环
- ✓ 代码
 

```
System.out.print(2+" ");
```

```

for(int zuo=3;zuo<=100;zuo++)
    for(int you=2;you<=zuo-1;you++)
        if(zuo%you==0)
            break;
        else
            if(you==zuo-1)
                System.out.print(zuo+" ");

```

## 171 请使用标准 SQL 语句解答下列问题

现有学生课程数据库的三张表，

学生表 Student (Sno, Sname, Ssex, Sage, Sdept) : Sno-学号, Sname-姓名, Ssex-性别, Sage-年龄, Sdept-所在系

课程表 Course (Cno, Cname, Cpno, Ccredit) : Cno-课程号, Cname-课程名, Cpno-选修课号, Ccredit-学分

学生选课表 SC (Sno, Cno, Grade) : Sno-学号, Cno-课程号, Grade-成绩

问题：

- 1、求各个课程号及其相应的选课人数，按照选课人数降序排列
- 2、求所有学生的姓名以及其选修课程的课程名。注意有部分学生未选任何课程
- 3、求与“张三”在同一个系学习的学生学号和姓名
- 4、查询没有选修 1 号课程的学生姓名
- 5、将计算机系 (Sdept= 'CS') 全体学生的选课成绩改为 100

➤ 参考答案

- ✓ 1、select cno,count(\*) rs from sc group by cno order by rs desc
- ✓ 2、select sname,cname from student left join sc on student.sno=course.sno left join course on course.cno=sc.cno
- ✓ 3、select sno,sname from student where sdept=(select sdept from student where sname='张三')
- ✓ 4、select sname from student where sno in(select sno from sc where cno!=1)
- ✓ 5、update sc set grade=100 where sno in(select sno from student where sdept='CS')

## 172 小明一家人要过一座桥，过桥是在黑夜，所以必须有灯。小明过

桥要 1 分钟，小明的弟弟要 3 分钟，小明的爸要 6 分钟，小明的妈妈要 8 分钟，小明的爷爷要 12 分钟。每次此桥最多可过两人，过桥的速度按过桥慢者而定。但是灯在点燃后 30 分钟就会熄灭。

问小明一家怎样过去？

➤ 需求

- ✓ 时间不能超过 30 分
- ✓ 不超过两人过桥，允许一人过桥
- ✓ 过桥人必须带灯



- ✓ 时间慢的为准
- 策略
  - ✓ 明 1+弟 3            3
  - ✓ 明回            1
  - ✓ 妈 8+爷 12    12
  - ✓ 弟回            3
  - ✓ 明 1+爸 6            6
  - ✓ 明回            1
  - ✓ 明 1+弟 3            3
- 总共需要 29 分钟

**173 烧一根不均匀的绳，从头烧到尾总共需要 1 个小时。现在有若干条材质相同的绳子，问如何用烧绳的方法来计时一个小时十五分钟呢？**

- 需求分析
  - ✓ 一根绳子 1 小时
  - ✓ 材质不均匀
- 策略
  - ✓ 烧一根绳子：1 小时
  - ✓ 绳子两头烧，烧完 30 分钟
  - ✓ 再找一部分，两头烧
- 参考答案
  - ✓ 找两根绳子，一根从头烧，一根两头烧
  - ✓ 当两头烧的绳子烧完时，用时 30 分钟，将另一根绳子熄灭
  - ✓ 将熄灭的绳子两头烧，开始计时，烧完用时 15 分钟
  - ✓ 等绳子熄灭后，再烧一根绳子，用时 60 分钟

**174 1 元钱一瓶汽水，喝完后两个空瓶换一瓶汽水，问：你有 20 元钱，最多可以喝到几瓶汽水？**

- 需求
  - ✓ 1 元 1 瓶
  - ✓ 2 瓶换 1 瓶
  - ✓ 总共 20 元
- 策略
  - ✓ 20 元，买 20 瓶，喝 20 瓶
  - ✓ 20 瓶换 10 瓶，喝 10 瓶
  - ✓ 10 瓶换 5 瓶，喝 5 瓶
  - ✓ 5 瓶中的 4 瓶换 2 瓶，喝 2 瓶，还剩 1 空瓶
  - ✓ 2 瓶换 1 瓶，喝 1 瓶，上一步还剩 1 空瓶，现在新剩 1 空瓶
  - ✓ 2 空瓶换 1 瓶，喝 1 瓶

✓  $20+10+5+2+1+1=39$

**175** 小明早上起床发现闹钟停了，把闹钟调到 7 点 10 分后，就去图书馆看书。当到那里时，他看墙上的闹钟是 8 点 50 分，又在那看了一个半小时书后，又用同样的时间回到家，这时家里闹钟显示为 11 点 50 分。请问此时小明该把时间调到几点？

➤ 需求

- ✓ 地点：家里、路上、图书馆
- ✓ 路上时间一致

➤ 思路

- ✓ 家里      路途      图书馆
- ✓ 7:10      8:50
- ✓      看书      1:30
- ✓           10:20
- ✓ 11:50
- ✓
- ✓ 标准一致：
- ✓ 家里：去回时长 4:40
- ✓ 图书馆：呆的时间 1:30
- ✓ 路上时间：3:10
- ✓ 单程时间：1:35

➤ 结论

- ✓ 家里 7:10+路上 1:35=按照家里的时间应该是 8:45，而图书馆是 8:50，说明家里比图书馆慢 5 分钟
- ✓ 家里表调成 11:55

**176** 小明和小华买了 10 斤蜂蜜，装在一个大瓶子里，要把蜂蜜平分，只有两个空瓶子，一个正好装 7 斤，另一个正好装 3 斤，怎样才能用最简单的方法把蜂蜜分出来。

➤ 需求

- ✓ 有一个大瓶子，有 10 斤蜂蜜
- ✓ 两个空瓶，一个 7 斤，一个 3 斤
- ✓ 平分蜂蜜

➤ 策略

- ✓ 10 斤瓶，倒入 3 斤瓶，3 斤瓶倒入 7 斤瓶（结果：10 斤有 7，7 斤有 3）
- ✓ 10 斤瓶，倒入 3 斤瓶，3 斤瓶倒入 7 斤瓶（结果：10 斤有 4，7 斤有 6）
- ✓ 10 斤瓶，倒入 3 斤瓶，3 斤瓶只能有 1 斤倒入 7 斤瓶（结果：10 斤有 1，7 斤有 7，3 斤瓶有 2）

- ✓ 7斤瓶倒入10斤瓶（结果：10斤有8，3斤有2），3斤瓶的2斤倒入7斤瓶（7斤有2），10斤瓶倒入3斤瓶，3斤瓶倒入7斤瓶，完成

**177 你让工人为你工作7天，给工人的回报是一根金条。金条平分成相连的7段，你必须在每天结束时给他们一段金条，如果只许你两次把金条弄断，你如何给你的工人付费？**

- 需求
  - ✓ 7天总共一根金条
  - ✓ 每天都要给一段
  - ✓ 可以分成3段
- 参考答案
  - ✓ 分成1、2、4三份
  - ✓ 第1天：给1
  - ✓ 第2天：给2，收回1
  - ✓ 第3天：给1
  - ✓ 第4天：给4
  - ✓ 第5天：给1
  - ✓ 第6天：给2，收回1
  - ✓ 第7天：给1

**178 12个球，一个天平，现知道只有一个和其它的重量不同，问怎样称才能用三次就找到那个球。**

- 需求
  - ✓ 12个球
  - ✓ 1个球重量与其他不同，其它重量都相同
  - ✓ 称3次
- 策略
  - ✓ 首先将4个球一组，分成3份
  - ✓ 随便2份上称，要么两份相同重量，则剩余一份超重，要么两份中有一份超重，中止能发现一份超重的
  - ✓ 将超重的4个球，2球一组分为2份，上称，一定会找到重的一份
  - ✓ 将重的一份上称，最终就可以找到超重的一个球

**179 门外三个开关分别对应室内三个灯泡，线路良好，在门外控制开关时候不能看到室内灯的情况，现在只允许进门一次，确定开关和灯的对应关系？**

- 需求

- ✓ 三个灯泡，灯泡会发热的，灯会亮的
- ✓ 只能进门一次
- 策略
  - ✓ 打开一个开关，持续 10 分钟，灯会发热，关掉
  - ✓ 再打开一个开关，进门
  - ✓ 摸一下灯泡，热的为第 1 个，亮的为第 2 个，另一个对应

**180 现有员工信息表 A，员工编号（codecode，PK），员工姓名（codename），员工性别（codesex），员工电话（codetel），员工住址（codeaddress），员工亲属信息表 B：员工编码（codecode，PK），亲属编码（recodecode，PK）、亲属姓名（recodename），联系电话（codetel）**

- 1、使用 SQL 语句，向员工信息表中插入一条数据：（001，张三，男，010-6257007，北京市海淀区）；向亲属信息表中插入 2 条数据（001，1，父亲，010-6257007）（001，2，母亲，010-6257007）
- 2、使用 SQL 变更 001 员工地址为：北京市朝阳区
- 3、用 SQL 查询出亲属数量大于 1 的员工编码、员工姓名、员工亲属数量
- 4、有部分员工的亲属信息录入重复，例如父亲信息录入两次，这类信息的特征是亲属姓名相同，但是亲属编码不同，请用 SQL 语句将这类情况的员工编码，重复的亲属编码，亲属姓名查询出来

- 参考答案
  - ✓ 1、insert into A values('001','张三','男','010-6257007','北京市海淀区')  
insert into B values ('001','1','父亲','010-6257007'),('001','2','母亲','010-6257007')
  - ✓ 2、update A set codeaddress='北京市朝阳区' where codecode='001'
  - ✓ 3、select A.codecode,codename,count(\*) from A join B on A.codecode=B.codecode group by A.codecode having count(\*)>1
  - ✓ 4、select C.codecode,C.recodecode,C.recodename from B C join B D on C.recodename=D.recodename and C.recodecode!=D.recodecode

**181 请使用你熟悉的编程语言完成数组 7、3、8、2、9、5、1 的排序**

- 两种方法：冒泡、选择
- 冒泡

```
int a[]={7,3,8,2,9,5,1};
int t;
for(int lun=1;lun<=a.length-1;lun++)
    for(int ci=1;ci<=a.length-lun;ci++)
        if(a[ci-1]>a[ci]){
            t=a[ci-1];
            a[ci-1]=a[ci];
```

```

        a[ci]=t;
    }
    for(int x:a)
        System.out.print(x+" ");
➤ 选择
    int a[]={7,3,8,2,9,5,1};
    int k,t;
    for(int lun=1;lun<=a.length-1;lun++){
        k=lun-1;
        for(int i=lun;i<=a.length-1;i++)
            if(a[i]<a[k])
                k=i;
        t=a[k];
        a[k]=a[lun-1];
        a[lun-1]=t;
    }
    for(int x:a)
        System.out.print(x+" ");

```

## 182 设计登录的测试用例

- | 账号  | 密码  | 预期结果      |
|-----|-----|-----------|
| 对   | 对   | 登录成功      |
| 对   | 太短  | 提示密码长度错误  |
| 对   | 太长  | 提示密码长度错误  |
| 对   | 内容错 | 提示账号或密码错误 |
| 太短  | 对   | 提示账号长度错误  |
| 太短  | 太短  | 提示账号长度错误  |
| 太短  | 太长  | 提示账号长度错误  |
| 太短  | 内容错 | 提示账号长度错误  |
| 太长  | 对   | 提示账号长度错误  |
| 太长  | 太短  | 提示账号长度错误  |
| 太长  | 太长  | 提示账号长度错误  |
| 太长  | 内容错 | 提示账号长度错误  |
| 内容错 | 对   | 提示账号错误    |
| 内容错 | 太短  | 提示账号错误    |
| 内容错 | 太长  | 提示账号错误    |
| 内容错 | 内容错 | 提示账号错误    |

## 183 存在一个数列 0 1 1 2 3 5 8 13 21 33 55 89 144 233 377...

- 描述数列中的规律，此数据存在一处错误，请找出
- 用任意语言实现该数列
- 参考答案
  - ✓ 规律：从第三个数起，后面的数是前面两个数之和

- ✓ 错误处: 33, 应该是 34
- ✓ Java 代码

```
int a=0,b=1,s=0;
System.out.print(s+" ");
while(s<400){
    s=a+b;
    a=b;
    b=s;
    System.out.print(s+" ");
}
```
- ✓ Python 代码

```
a,b=0,1
print(0,end=" ")
while b<400:
    a,b=b,a+b
    print(b,end=" ")
```

## 184 没时间写测试用例怎么办?

- 在日常测试过程中, 每天写日志时, 记录重要的需求
- 随时记录重要的测试点
- 测试过程中, 会发现一些缺陷, 把这些缺陷整理成测试点
- 重要数据也可以做记录
- 寻找空闲时间, 写出用例

## 185 表 A 有成绩 grade, 用 sql 写出 grade 大于 70 的人的数目

- `select count(*) from A where grade>70`

## 186 写出下列 linux 命令的作用。

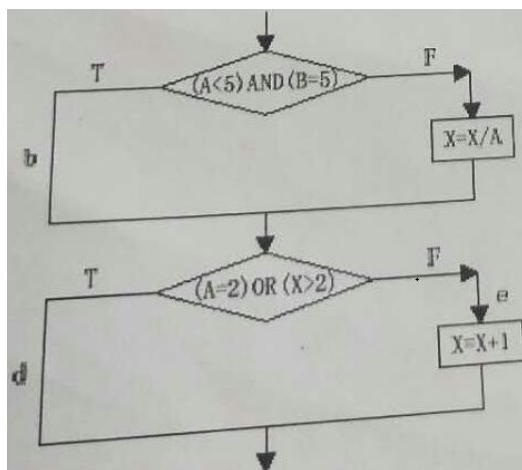
- top
- free
- du
- df
- cd
- ls
- cat
- vi
- rm
- mkdir
- touch
- cp
- 参考答案
  - ✓ top: 查看 cpu、内存密集使用的动态数据

- ✓ free: 查看内存使用情况
- ✓ du: 查看目录或文件的大小
- ✓ df: 查看磁盘分区的大小
- ✓ cd: 切换目录
- ✓ ls: 显示目录中的内容
- ✓ cat: 查看文件中的数据
- ✓ vi: 进入/打开、修改、保存文件
- ✓ rm: 删除目录或文件
- ✓ mkdir: 创建目录
- ✓ touch: 如果一个文件不存在, 则创建文件, 如果文件存在, 则修改文件最后的访问时间
- ✓ cp: 拷贝目录或文件

**187 在 Linux 中使用 Shell 修改/etc/snmp.conf 文件为所有用户都可读可写可执行。**

- chmod a+rwX /etc/snmp.conf
- chmod 777 /etc/snmp.conf

**188 对下面给出的程序控制图, 分别以各种不同的测试方法写出最少的测试用例。**



- 语句覆盖
  - ✓  $A < 5$  且  $B = 5$  不成立,  $A = 2$  或  $X > 2$  不成立
    - ◆ 用例:  $A = 5, B = 6, X = 0$
- 判定覆盖
  - ✓  $A < 5$  且  $B = 5$  真假各一次,  $A = 2$  或  $X > 2$  真假各一次
    - ◆ 用例:
      - $A = 2, B = 5, X = 3$
      - $A = 5, B = 6, X = 0$
- 条件覆盖

- ✓ A<5 真假各一次, B=5 真假各一次, A=2 真假各一次, X>2 真假各一次
  - ◆ 用例:
    - A=2, B=5, X=3
    - A=5, B=6, X=0
- 条件组合覆盖
  - ✓ A<5 真, B=5 真 第 1 条用例
  - ✓ A<5 真, B=5 假 3
  - ✓ A<5 假, B=5 真 4
  - ✓ A<5 假, B=5 假 2
  - ✓ A=2 真, X>2 真 1
  - ✓ A=2 真, X>2 假 3
  - ✓ A=2 假, X>2 真 4
  - ✓ A=2 假, X>2 假 1
  - ◆ 用例:
    - A=2, B=5, X=3
    - A=5, B=6, X=0
    - A=2, B=6, X=0
    - A=5, B=5, X=3
- 路径覆盖
  - ✓ 条件组合覆盖已经涵盖

## 189 解释图片中 sql 命令的含义



- 查找 18 到 24 岁以内没有男朋友的女孩信息

## 190 Win7 系统中, 使用 IE 浏览器中输入 www.sogou.com, 按回车,

1 分钟后, 显示为白页面, 请问有哪些原因会导致这个现象? 分别如何进行排查?

- 关闭浏览器, 重新打开浏览器并输入网址
- 清除浏览器缓存以后, 重新打开 sogou 网址
- 输入其它网址看能否打开
- 换成其它浏览器, 输入 sogou 网址



- 重启计算机，重新打开浏览器，输入 sogou 网址
- 更换计算机，还是 win7 系统，重新打开 sogou 网址
- 重新安装操作系统后，重新打开 sogou 网址
- 尝试在其它操作系统中打开 sogou 网址

**191 有一个在线购物网站，分别适配 PC 端和微信端，针对选择商品—提交订单—支付这个流程（默认已经成功登录），请列出你能想到的测试点。**

- 不选择商品，提交订单
- 选择一件商品
- 选择一件商品的多件
- 选择不同商品各一件
- 选择多种商品各多件
- 选择多件商品，有一件的，有多件的
- 修改商品的件数
  - ✓ 一件变多件
  - ✓ 多件变一件
  - ✓ 一件变 0
- 删除某类商品
- 清空所有商品
- 提交订单后，撤销订单
- 提交订单后，付款
- 提交订单后，不付款
- 分别在 pc 端和微信端作上述测试
- 查看 pc 段和微信端数据是否同步

**192 web 系统中，username 参数表为 file 类型，表中有 12 个值，分别 A、B、C、D、E、F、G、H、I、J、K、L。测试场景中虚拟并发用户数设为 4，迭代次数设为 3，参数中 Select next row 与 Update value on 分别为（Unique，Once）时，写出迭代 3 次的取值情况。**

- 第 1 个虚拟用户：3 次均选择 A
- 第 2 个虚拟用户：3 次均选择 B
- 第 3 个虚拟用户：3 次均选择 C
- 第 4 个虚拟用户：3 次均选择 D