|  |  |  |  |
| --- | --- | --- | --- |
| **Detected By** | **Summary** | **Description** | **Defect ID** |
| admin | 【138】 翻译：有关P2P点对点文件传输的原理 | 1、Related P2P point-to-point file transfer principle  2、The P2P principle of point-to-point file transfer | 154 |
| admin | 【139】 翻译：有关互联网的发展对商务、学习、交流的影响 | 1、Related Internet’s development to commercial, study, exchange influence  2、The development of the Internet business, study, the impact of exchange | 155 |
| admin | FAT16/FAT32/NTFS 哪个安全性最好，为什么？ | 肯定是NTFS了。NTFS可以进行权限设置，并且支持文件EFS加密,所以更安全一些。 | 156 |
| admin | c语言中主要的排序方法 | 目前所知：直接插入排序、冒泡排序、简单选择排序、希尔排序、快速排序、基数排序。  PS:可以网上搜索排序例题。好多··· | 157 |
| admin | 网管告诉你路由器可以与远程端进行通信，你的WIN98不能通信，应该检查什么配置？ | 协议配置：需要在TCP/IP协议上捆绑NETBIOS解析计算机名。（个人想法） | 158 |
| admin | 【147】 中间件的含义，意义，主要产品名称 | 中间件（middleware）是基础软件的一大类，属于可复用软件的范畴。顾名思义，中间件处于操作系统软件与用户的应用软件的中间。中间件在操作系统、网络和数据库之上，应用软件的下层，应用意义是为处于自己上层的应用软件提供运行与开发的环境，帮助用户灵活、高效地开发和集成复杂的应用软件。  在众多关于中间件的定义中，比较普遍被接受的是IDC表述的：中间件是一种独立的系统软件或服务程序，分布式应用软件借助这种软件在不同的技术之间共享资源，中间件位于客户机服务器的操作系统之上，管理计算资源和网络通信。  IDC对中间件的定义表明，中间件是一类软件，而非一种软件；中间件不仅仅实现互连，还要实现应用之间的互操作；中间件是基于分布式处理的软件，最突出的特点是其网络通信功能。  主要产品：应用服务器、消息中间件、报表生成工具 | 159 |
| admin | 【148】 简述一下c/s模式或者b/s模式 | C/S：网络连接模式的一种,即客户机/服务器，Client/Server。在客户机/服务器网络中，服务器是网络的核心，而客户机是网络的基础，客户机依靠服务器获得所需要的网络资源，而服务器为客户机提供网络必须的资源。  B/S结构，即Browser/Server(浏览器/服务器)结构，是随着Internet技术的兴起，对C/S结构的一种变化或者改进的结构。在这种结构下，用户界面完全通过WWW浏览器实现，一部分事务逻辑在前端实现，但是主要事务逻辑在服务器端实现。B/S结构利用不断成熟和普及的浏览器技术实现原来需要复杂专用软件才能实现的强大功能。 | 160 |
| admin | 【149】 SQL语句 | 1、    建表TEST，包含3个字段A1整形、A2时间形、A3字符型（20）  2、    分别写出2条能成功插入数据的语句和3条不能成功插入的语句  3、    找出TEST表中A3字段与TEST01表中B2字段相同的记录 | 161 |
| admin | 【150】 数据库： | Student表（Sno， Sname，Sage，Ssex）          Course表 （Cno，Cname，Ccredit）          SC表 （Sno，Cno，grade）  1、    建表student、course、SC  2、    查询年龄小于20的学生学号、年龄并显示  3、    查询年龄不在20-23（包括20，23）的学生的成绩 | 162 |
| admin | 【151】 如何测试ATM? | 【151】 如何测试ATM? | 163 |
| admin | 【152】 why do you want to be a tester? | 1、    To expand the computer field of knowledge Continuously  不断地充实自己在电脑领域的知识面  2、    A very rely on the experience and technical job, and teamwork, work is very challenging, meaningful  一份十分依靠经验和技术的工种,团队合作,工作时很具挑战性,很有意义  3、    High technology content, the significance of the work, to get a satisfaction of my own salaries  技术含量高,工作意义大,能领到一份自己满意的薪水 | 164 |
| admin | 【153】 测试的分类和过程。 | 基本可以从3个角度进行分类  1、    是否关注软件的结构和算法，可以分为白盒测试和黑盒测试  2、    是否执行被测试软件可以分为动态测试和静态测试  3、    基于测试的不同阶段可以分为：单元测试、集成测试、系统测试、验收测试  测软件试大致为3个基本步骤：1、测试计划 2、测试用例的设计和开发 3、执行测试 | 165 |
| admin | 【154】 如果你是测试组长，你将怎么样做计划，去测试一个ICEBOX。列出KEY | 从用户的需求入手。如冷藏、冷冻空间大小；冷藏、冷冻温度的上限是否能达到用户的需求。 | 166 |
| admin | 【154】如果你是测试组长，你将怎么样做计划，去测试一个ICEBOX。列出KEY |  | 167 |
| admin | 【155】 名词解释3G/PDA/HTML  【155】 名词解释3G/PDA/HTML | 3G 全称为3rd Generation，中文含义就是指第三代数字通信。第三代与前两代的主要区别是在传输声音和数据的速度上的提升，能够在全球范围内更好地实现无缝漫游，并处理图像、音乐、视频流等多种媒体形式，提供包括网页浏览、电话会议、电子商务等多种信息服务，同时也要考虑与已有第二代系统的良好兼容性。  PDA 英文全称Personal Digital Assistant，即个人数码助理，一般是指掌上电脑。相对于传统电脑，PDA的优点是轻便、小巧、可移动性强，同时又不失功能的强大，缺点是屏幕过小，且电池续航能力有限。PDA通常采用手写笔作为输入设备，而存储卡作为外部存储介质。在无线传输方面，大多数PDA具有红外和蓝牙接口，以保证无线传输的便利性。许多PDA还能够具备Wi-Fi连接以及GPS全球卫星定位系统。  HTML （HyperTextMark-upLanguage）即超文本标记语言或超文本链接标示语言，是WWW的描述语言。设计HTML语言的目的是为了能把存放在一台电脑中的文本或图形与另一台电脑中的文本或图形方便地联系在一起，形成有机的整体。HTML的结构包括头部（Head）、主体（Body）两大部分，其中头部描述浏览器所需的信息，而主体则包含所要说明的具体内容。 | 168 |
| admin | 【155】 名词解释3G/PDA/HTML  【155】 名词解释3G/PDA/HTML |  | 169 |
| admin | 【156】 什么是好的Bug？ | 发现的这个BUG给自己带来了不错的收获和学习的经验的话，应该算是一个好的BUG。 | 170 |
| admin | 【157】 对WINCE有了解么？ | Windows CE操作系统是Windows家族中最新的成员，专门设计给掌上型电脑所使用的电脑环境。这样的操作系统可使完整的可携式技术与现有的Windows桌面技术整合工作。 | 171 |
| admin | 【158】 “outer join” 是什么意思？ | 外连接。  left outer join 以左表为主，左表内容全显示，右表连接上的内容才显示  right outer join相反  full outer join 左右表未连接的内容全部显示  外联接可以是左向外联接、右向外联接或完整外部联接。 | 172 |
| pc1 | 1. 为什么要在一个团队中开展软件测试工作？见书P12 | ⑴.开发人员不太了解需求，不清楚应该“做什么“和“不做什么“，常常做不合需求的事情，因此产生了错误。  ⑵.软件系统越来越复杂，开发人员不太可能精通所有的技术，如果不能正确的使用技术，将产生错误。  ⑶.技术文档普遍比较糟糕，文档本身就有错误，导致使用者产生更多的错误。  ⑷.软件需求,设计报告,程序经常发生变更每次变更都可能产生新的错误。  ⑸.任何人在编程时都有可能犯错误，导致程序中有错误。  ⑹.人们长处于进度的压力之下，急忙之下容易产生错误，尤其是在期限临近之际。  ⑺.人们过于自信，喜欢说“没问题“，不真实的“没问题“将产生真实的问题。  …….软件错误的产生是不可避免的。很多公司都在测试过程中统计错误发生的原因，这样将有利于工作的改进。  ⑴.开发人员不太了解需求，不清楚应该“做什么“和“不做什么“，常常做不合需求的事情，因此产生了错误。  ⑵.软件系统越来越复杂，开发人员不太可能精通所有的技术，如果不能正确的使用技术，将产生错误。  ⑶.技术文档普遍比较糟糕，文档本身就有错误，导致使用者产生更多的错误。  ⑷.软件需求,设计报告,程序经常发生变更每次变更都可能产生新的错误。  ⑸.任何人在编程时都有可能犯错误，导致程序中有错误。  ⑹.人们长处于进度的压力之下，急忙之下容易产生错误，尤其是在期限临近之际。  ⑺.人们过于自信，喜欢说“没问题“，不真实的“没问题“将产生真实的问题。  …….软件错误的产生是不可避免的。很多公司都在测试过程中统计错误发生的原因，这样将有利于工作的改进。 | 13 |
| pc1 | 2. 您是否了解以往所工作的企业的软件测试过程？如果了解，请试述在这个过程中都有哪些工作要做？分别由哪些不同的角色来完成这些工作？ | 测试工作的基本步骤：测试计划、测试设计与开发、执行测试。按照不同的阶段软件测试要经过单元测试，集成测试，系统测试和验收测试阶段。  测试计划中测试人员首先对需求进行分析，最终定义一个测试集合。测试设计与开发阶段测试人员根据软件需求、软件设计说明书完成用例设计并编写必要的测试驱动程序。在执行测试中要建立测试环境、运行测试用例、记录测试结果、报告软件缺陷、跟踪软件缺陷直到被处理、分析测试结果。  单元测试和集成测试由设计人员和程序员完成，系统测试由软件测试小组根据上面的三个基本步骤完成，验收测试由用户完成。  测试工作的基本步骤：测试计划、测试设计与开发、执行测试。按照不同的阶段软件测试要经过单元测试，集成测试，系统测试和验收测试阶段。  测试计划中测试人员首先对需求进行分析，最终定义一个测试集合。测试设计与开发阶段测试人员根据软件需求、软件设计说明书完成用例设计并编写必要的测试驱动程序。在执行测试中要建立测试环境、运行测试用例、记录测试结果、报告软件缺陷、跟踪软件缺陷直到被处理、分析测试结果。  单元测试和集成测试由设计人员和程序员完成，系统测试由软件测试小组根据上面的三个基本步骤完成，验收测试由用户完成。 | 15 |
| pc1 | 3. 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？（对于软件测试部分，可以简述） | 如果只说过程的话，那就是：  1。问题定义 2。软件计划 3。需求分析 4。软件设计（概要设计+详细设计）  5。编码 6。测试 7。维护  一个真正的软件开发过程应该包括这些内容：  1。参与者：比如项目经理、系统分析员、架构设计师、程序员、测试设计师、测试员、writer、SQA，要清晰地规定每个角色的能力要求和职责。  2。过程的产品：比如RFP、需求规格说明、ROSE的设计模型、源代码、软件的发布版、相关的文档、比如项目计划、用户手册等。  3。过程的活动：比如需求分析、系统分析/设计、编码、测试、部署等，要规定每个活动由谁负责、输入是什么、输出是什么。  更复杂的过程还规定了采用的方法学，比如面向对象的方法学、结构化的设计、快速开发  还可以规定过程的生命周期,比如瀑布模型、迭代模型。  任何过程都离不开这样的框架！ | 26 |
| pc1 | 4. 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？ | 搭建测试测试环境，测试计划与测试用例的编写，执行测试用例等  比较擅长编写测试用例 | 37 |
| pc1 | 6. 请试着比较一下黑盒测试、白盒测试、单元测试、集成测试、系统测试、验收测试的区别与联系。 | 黑盒测试：不知道程序内部结构，只知道程序规格的情况下采用的测试技术与策略。可以进行测试证明每个实现了的功能是否符合要求。  白盒测试：知道程序的内部结构情况下采用的测试技术与策略。可以通过测试证明每种内部操作是否符合设计规格要求，所有内部成分是否以经过检查。  单元测试：检查程序的最小单元，模块的正确性，在编码完成后首先进行的测试。通常由开发工程师完成，多采用静态测试、白盒测试。  集成测试：当单元集成在一起，检查组成系统的模块接口是否正确进行的测试。通常由开发工程师与测试工程师一起完成，多采用黑盒测试与白盒测试相结合的方式。  系统测试：软件开发完毕，多数集成测试完后，检查系统是否符合软件需求。由测试工程师完成，采用黑盒测试方法。  验收测试：与系统测试的内容相同，在系统测试完成后，测试人员中加入用户，使用户再次验证系统是否符合软件需求。 | 39 |
| pc1 | 7. 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？ | 测试计划工作的目的是：为了使测试团队成员对测试的意图、期望以及将要执行的任务有充分理解，可以更好地其他相关部门交流，协调完成工作。  测试计划工作的内容包括：  1、 测试目标：对测试目标进行简要的描述。  2、 测试概要：摘要说明所需测试的软件、名词解释、以及提及所参考的相关文档。  3、 测试范围：测试计划所包含的测试软件需测试的范围和优先级，哪些需要重点测试、哪些无需测试或无法测试或推迟测试。  4、 重点事项：列出需要测试的软件的所有的主要功能和测试重点，这部分应该能和测试案例设计相对应和互相检查。  5、 质量目标：制定测试软件的产品质量目标和软件测试目标。  6、 资源需求：进行测试所需要的软硬件、测试工具、必要的技术资源、培训、文档等。  7、 人员组织：需要多少人进行测试，各自的角色和责任，他们是否需要进行相关的学习和培训，什么时候他们需要开始，并将持续多长时间。  8、 测试策略：制定测试整体策略、所使用的测试技术和方法。  9、 发布提交：在按照测试计划进行测试发布后需要交付的软件产品、测试案例、测试数据及相关文档。  10、测试进度和任务人员安排：将测试的计划合理的分配到不同的测试人员，并注意先后顺序.如果开发的Release不确定，可以给出测试的时间段.对于长期大型的测试计划，可以使用里程碑来表示进度的变化。  11、测试开始/完成/延迟/继续的标准：制定测试开始和完成的标准；某些时候，测试计划会因某种原因(过多阻塞性的Bug)而导致延迟，问题解决后测试继续。  12、 风险分析：需要考虑测试计划中可能的风险和解决方法。  重要的有：测试需求、测试策略、测试资源、风险和意外。 | 47 |
| pc1 | 8. 您认为做好测试计划工作的关键是什么？ | 关键是对软件需求进行充分的了解和分析，详细描述软件测试工作的目标、范围、策略、方法和重点。 | 52 |
| pc1 | 9. 您所熟悉的测试用例设计方法都有哪些？请分别以具体的例子来说明这些方法在测试用例设计工作中的应用。 | 黑盒测试的测试用例设计方法有：等价类划分、边界值划分、因果图分析和错误猜测.  白盒测试的测试用例设计方法有：语句覆盖、判定覆盖、条件覆盖、判定/条件覆盖、多重条件覆盖。在这里，主要讨论的是黑盒测试的测试用例的设计方法。 | 57 |
| pc1 | 10.您认为做好测试用例设计工作的关键是什么？ | 关键是对系统的熟悉程度，需求的理解，设计文档的了解情况。 | 64 |
| pc1 | 12. 您以往的工作中是否曾开展过测试用例的评审工作？如果有，请描述测试用例评审的过程和评审的内容。 | 有!  1、在项目开始时，所有测试测试人员必须参与需求分析，这样做的目的是解决测试需求的问题。写好软件需求不仅是一种能力，更是一种艺术。但真正能做到这点的人太少了。所以最好让测试设计人员尽早参与需求分析。  2、有一个明确的测试过程  3、建立测试案例模板，先对模板进行评审，避免对案例评审时又高出模板的问题。  4、评审应该分阶段进行，不要一次评审太多，要重点突出。  5、评审前应该提前告知被评审人及评审组要评审的内容和范围。以便参与评审的成员有所准备。不要请对需求不清楚的人参与评审，他们一般情况下是不说话，要么讲话不能切中要害，耽误时间。  6、评审开始，首先由被评审人介绍测试需求、测试设计的原则，然后，在进入详细案例的评审。  7、评审主要检查测试案例是否覆盖了所有的测试需求（正常的和异常的），测试案例是否有相应的环境来执行，测试案例是否有重复。  8、评审过程要有评审记录  9、评审不要超过2小时，否则，评审很可能成为一种形式。  10、评审者在评审过程中尽量不要打断被评审者的讲述，应该在评审结束后发表自己的意见。最后形成评审结论。 | 71 |
| pc1 | 13. 您以往是否曾经从事过性能测试工作？如果有，请尽可能的详细描述您以往的性能测试工作的完整过程。 | 是的，曾经从事过性能测试工作。  主要是通过使用Loadrunner自动化性能测试工具完成任务。比如模拟多个虚拟用户使用软件，并发执行软件某项功能等 | 77 |
| pc11 | 121.写出HUB、tch、Router是ISO的第几层设备，叫什么？ | HUB是ISO的第一层物理层，叫集线器  Switch在第二层数据链路层，叫交换机  Router在第三层网络层 ，叫路由器 | 1 |
| pc11 | 122.怎么测试ATM？ | 测试ATM基本流程  开端是ATM处于准备就绪状态。  1.准备提款：客户将银行卡插入ATM机的读卡机。  2.验证银行卡：ATM机从银行卡的磁条中读取帐户代码，并检查它是否属于可以接收的银行卡。  3.输入PIN：ATM要求客户输入PIN码（4位）  4.验证帐户代码和PIN： 验证帐户代码和PIN以确定该帐户是否有效以及所输入的PIN对该帐户来说是否正确。对于此事件流，帐户是有效的而且PIN对此帐户来说正确无误。  5.ATM选项：ATM显示在本机上可用的各种选项。在此事件流中，银行客户通常选择“提款“。  6.输入金额：要从ATM中提取的金额。对于此事件流，客户需选择预设的金额（10 美元、20美元、50美元或100美元）。  7.授权：ATM通过将卡ID、PIN、金额以及帐户信息作为一笔交易发送给银行系统来启动验证过程。对于此事件流，银行系统处于联机状态，而且对授权请求给予答复，批准完成提款过程，并且据此更新帐户余额。  8.出钞：提供现金。  9.返回银行卡：银行卡被返还。  10.收据：打印收据并提供给客户。ATM还相应地更新内部记录。  用例结束时ATM又回到准备就绪状态。  银行卡无效在基本流步骤2：验证银行卡，如果卡是无效的，则卡被退回，同时会通知相关消息。  ATM内没有现金在基本流步骤5：ATM选项，如果ATM内没有现金，则“提款“选项将无法使用。  ATM内现金不足在基本流步骤6：输入金额，如果ATM机内金额少于请求提取的金额，则将显示一则适当的消息，并且在步骤6：输入金额处重新加入基本流。  在基本流步骤4：验证帐户和PIN，客户有三次机会输入PIN。如果PIN输入有误，ATM将显示适当的消息；如果还存在输入机会，则此事件流在步骤3-输入PIN 处重新加入基本流。如果最后一次尝试输入的PIN 码仍然错误，则该卡将被ATM 机保留，同时ATM返回到准备就绪状态，本用例终止。  帐户不存在在基本流步骤4：验证帐户和PIN，如果银行系统返回的代码表明找不到该帐户或禁止从该帐户中提款，则ATM显示适当的消息并且在步骤9：返回银行卡处重新加入基本流。  帐面金额不足在基本流步骤7：授权中，银行系统返回代码表明帐户余额少于在基本流步骤6：输入金额内输入的金额，则ATM 显示适当的消息并且在步骤6：输入金额处重新加入基本流。  达到每日最大的提款金额在基本流步骤7：授权中，银行系统返回的代码表明包括本提款请求在内，客户已经或将超过在24小时内允许提取的最多金额，则ATM显示适当的消息并在步骤6：输入金额上重新加入基本流。  记录错误如果在基本流步骤10：收据中，记录无法更新，则ATM进入“安全模式“，在此模式下所有功能都将暂停使用。同时向银行系统发送一条适当的警报信息表明ATM 已经暂停工作。  退出客户可随时决定终止交易（退出）。交易终止，银行卡随之退出。  ATM包含大量的传感器，用以监控各种功能，如电源检测器、不同的门和出入口处的测压器以及动作检测器等。在任一时刻，如果某个传感器被激活，则警报信号将发送给警方而且ATM进入“安全模式“，在此模式下所有功能都暂停使用，直到采取适当的重启/重新初始化的措施。    基础数据  PIN:4987  账号：809－498  账户余额：自由假设  ATM内现金：自由假设样式  TC（测试用例）ID 号  场景/条件PIN帐号输入的金额（或选择的金额）  帐面金额ATM内的金额  预期结果  CW1. 场景1 – 成功的提款  4987 809 – 498 50.00 500.00 2,000 成功的提款。帐户余额被更新为450.00 | 3 |
| pc11 | 123.文件格式系统有哪几种类型？分别说说WIN95、Win98、WinMe、W2k、WinNT、WinXP分别支持哪些文件系统。 | 文件系统格式有FAT12、Fat16、Fat32、NTFS、NTFS5.0、WinFS  Win95支持Fat16，  Win98支持Fat16、Fat32，  WinMe支持Fat16、Fat32，  W2k支持Fat16、Fat32、NTFS、NTFS5.0，  WinNT支持Fat16和NTFS，  WinXP支持Fat16、Fat32、NTFS、NTFS5.0. | 9 |
| pc11 | 124.各种测试的概念。 | 黑盒测试：不基于内部设计和代码的任何知识，而是基于需求和功能性。  白盒测试：基于一个应用代码的内部逻辑知识，测试是基于覆盖全部代码、分支、路径、条件。  单元测试：有程序员做的最微小规模的测试，以测试某个功能或代码块  集成测试：开发工程师与测试工程师进行的检验多个单元组成在一起接口是否耦合的测试  功能测试：用于测试应用系统的功能需求的黑盒测试方法。  系统测试：基于系统整体需求说明书的黑盒类测试；应覆盖系统所有联合的部件  健全测试：软件主要功能成分的简单测试以保证它是否能进行基本的测试  衰竭测试：软件或环境的修复或更正后的“再测试“。  接受测试：基于客户或最终用户的规格书的最终测试，或基于用户一段时间的使用后，看软件是否满足客户要求。  负载测试：测试软件系统的最大负载，超出此负载软件可能会失常。  强迫测试：在交替进行负荷和性能测试时常用的术语。也用于描述象在异乎寻常的重载下的系统功能测试之类的测试，如某个动作或输入大量的重复，大量数据的输入，对一个数据库系统大量的复杂查询等。  性能测试：测试软件在各种情况下的性能，如在正常或最大负载下的状况。  可用性测试：测试软件是否易用。  安装/卸载测试：测试软件在“全部、部分、升级“等状况下的安装/反安装过程。  恢复测试：测试该系统从故障中恢复过来的能力。  安全测试：测试该系统防止非法侵入的能力。  兼容测试：测试该系统与其他软件、硬件兼容的能力。  比较测试：通过与同类产品比较，考察该系统的优点缺点。  Alpha测试：一种先期的用户测试，此时系统刚刚开发完成。  Beta测试：一种后期的用户测试，稀释系统已经通过内部测试，大部分错误已经改正，即将正式发行。  灰盒测试：灰盒测试是介于黑盒测试和白盒测试之间的一种测试.这个阶段的测试重点是各个组件之间的逻辑,灰盒测试重点在核心模块，他投入相对黑盒测试的时间少，而且从维护量上和产出比上比黑盒测试得出的数据更加有效，相对于白盒测试，他需要付出的要少，而且白盒测试需要一些功力比较深的人才能从根本上发现和解决问题，他的结果可能会影响到算法，发现问题快，而且从根本上发现，所以更彻底，灰盒测试有自己的优势，投入少，见效快。 | 17 |
| pc11 | 125.说说软件危机 | 软件危机产生的原因是由于软件产品本身的特点以及开发软件的方式、方法、技术和人员引起的：  a.软件的规模越来越大，结构越来越复杂。  b.软件开发管理困难而复杂。  c.软件开发费用不断增加。  d.软件开发技术落后。  e.生产方式落后。  f.开发工具落后，生产率提高缓慢。  软件危机主要表现在以下几个方面：  a.经费预算经常突破，完成时间一再拖延。  b.开发的软件不通满足用户要求。  c.开发的软件可维护性差。  d. 开发的软件可靠性差。 | 56 |
| pc11 | 126.解释alpha测试与beta测试 | Alpha测试：一种先期的用户测试，此时系统刚刚开发完成。  Beta测试：一种后期的用户测试，稀释系统已经通过内部测试，大部分错误已经改正，即将正式发行。 | 66 |
| pc11 | 127.软件为什么会出现问题，或者说软件生产过程中常会出现的问题有哪些方面？ | 软件产生问题的原因是：  1、开发人员不太了解需求。  2、软件系统越来越复杂，开发人员不太可能精通左右的技术，如果不能正确的使用技术，将产生错误。  3、技术文档普遍比较糟糕。  4、软件需求、设计报告、程序经常发生变更，每次变更都可能产生新的错误。  5、编程时可能犯错误。  6、人们过于自信。 | 82 |
| pc11 | 128.软件测试的目的，从用户和开发者两个角度说 | 从用户的角度出发，普遍希望通过软件测试暴露软件中隐藏的错误和缺陷，以考虑是否可接受该产品。  从软件开发者的角度出发，则希望测试成为表明软件产品中不存在错误的过程，验证该软件已正确地实现了用户的要求，确立人们对软件质量的信心。 | 85 |
| pc11 | 129.怎么评价一组测试用例是比较好的。 | 好的测试用例应该：  1.容易发现软件错误（或者是能够发现以往还没有发现过的软件错误）。  2.要有可重复性，能够精确的重现失败的情景。  3.必须清晰的定义一个或多个期望结果以及测试通过和失败标准。  4.测试用例没有冗余。 | 88 |
| pc11 | 130.软件测试是从软件的那个阶段介入的，介入以后一般会做些什么？为什么从这个阶段介入？ | 软件测试是从软件的需求分析阶段介入的。  介入后一般：1.理解需求，参与审核需求文档。2.理解项目的目标、限制，了解用户应用背景。3.编写测试计划。4.准备资源。  从这个阶段介入有以下几个原因：1.测试人员需要进行前期准备工作，需要在了解需求的情况下编写测试计划、测试用例、准备测试环境。2.需求文档本身需要被测试。3.估算工作量，估算项目成本，编写开发计划。 | 95 |
| pc11 | 131.软件测试人员应具备的素质 | 测试人员应具备：责任心，学习能力，怀疑精神，沟通能力，专注力，洞察力，团队精神，注重积累。 | 98 |
| pc11 | 132.考官从办公室（面试现场）随意选取一个简单物品，假定是一个喝水的带广告图案的花纸杯，让应聘人对它设计出尽可能多的测试用例。 | 纸杯的测试用例：  基本功能测试：该纸杯装水漏不漏，渗水不，水能不能被喝到。  界面测试:查看杯子外观，看其形状、大小设计是否符合要求；外观是否吸引人，赏心悦目；广告图案沾水后是否掉色、模糊；广告图案是否和当地政治、宗教符合，没有冲突；广告图案是否做到了本地化和国际化。  安全性：杯子所用的材料（包括纸基、涂层和广告颜料）是否符合食品卫生标准， 在内外温度等环境因素下是否会与所盛各种饮料反应，而产生对人体有害的物质。所有与人的饮食有关的产品，这一条应该是头等重要的。  可靠性：杯子从不同高度落下的损坏程度。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。  可移植性：杯子再不同的地方、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。杯子是否符合国标、行标。  配置测试：杯子大小是否适应市面上常见的杯托。  易用性：杯子是否烫手、是否有防滑措施、是否方便饮用，杯子设计是否上大下小，在运输过程中可以套在一起有效利用空间，在使用时也容易拿开。  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。  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。  其他测试：杯子装水后震动,检查产品是否能应对的铁路\公路\航空运输；装水，并且放入汤匙，看杯子是否能平稳放置而不会倾倒在桌上。 | 108 |
| pc11 | 133.Web测试的要点有？ | 一、功能测试：1、链接测试2、表单测试3、Cookies测试4、设计语言测试5、数据库测试  二、性能测试：1、连接速度测试2、负载测试3、压力测试  三、可用性测试：1、导航测试2、图形测试3、内容测试4、整体界面测试  四、客户端兼容性测试：1、平台测试2、浏览器测试  五、安全性测试 | 112 |
| pc11 | 134.需求说明书测试：检查说明书书写准确性，给大家提供三个产品：1.手机，2.电饭锅，3.电梯。 | 手机说明书测试：  检查是否有错别字。  检查说明语句是否符合逻辑。  检查说明语句是否易懂。  检查图片是否与实物相符。  检查是否为多国语言。 | 116 |
| pc17 | 67.产品测试到什么时候就算是足够了？ | 1）基于测试用例的规则。          2）基于“测试期缺陷密度“的规则。          3）基于“运行期缺陷密度“的规则。 | 5 |
| pc17 | 68.软件测试应该划分几个阶段?简述各个阶段应重点测试的点?各个阶段的含义？ | 软件测试分为四个主要阶段：单元测试、集成测试、系统测试和验收测试。这是一个从小到大、循序渐进的的测试过程。      单元测试：主要测试软件的单元模块。一般由开发人员而非独立的测试人员来执行，因为测试者需要懂得该单元的设计与程序实现，测试者可能学要编写额外的测试驱动程序。  集成测试：将一些“构件“集成在一起，测试他们能否正常运行。这里“构件“可以是程序模块、客户机-服务器程序等等。  系统测试：测试软件系统是否符合所有需求，包括功能性需求与非功能性需求。一般由独立的测试人员执行，通常采用黑盒测试方式。  验收测试：与系统测试很相似，但由客户或最终用户执行，测试软件系统是否符合需求规格说明书。 | 8 |
| pc17 | 70.单元测试、集成测试、系统测试的侧重点是什么？ | 单元测试：主要测试软件的单元模块。一般由开发人员而非独立的测试人员来执行，因为测试者需要懂得该单元的设计与程序实现，测试者可能学要编写额外的测试驱动程序。  集成测试：将一些“构件“集成在一起，测试他们能否正常运行。这里“构件“可以是程序模块、客户机-服务器程序等等。  系统测试：测试软件系统是否符合所有需求，包括功能性需求与非功能性需求。一般由独立的测试人员执行，通常采用黑盒测试方式。 | 30 |
| pc17 | 71.设计用例的方法、依据有那些？ | 白盒测试用例的依据：以软件设计、软件结构和算法。  方法：逻辑覆盖法和基本路径法。  黑盒测试用例的依据：需求分析。  方法：等价类划分、边值分析、错误推测、因果图。 | 36 |
| pc17 | 72.集成测试通常都有那些策略？ | 1）把所有模块按设计要求一次全部组装起来，然后整体测试，这称为非增量式集成。  2）增量式集成方法，程序一段一段地扩展，测试的范围一步一步地增大，错误易于定位和纠正。包括两种：自顶向下集成、自底向上集成。 | 38 |
| pc17 | 73.你用过的测试工具的主要功能、性能及其他？ | 缺陷管理工具：bugzilla mantis.  bugzilla的特点：  1）基于Web方式，安装简单，运行方便快捷，管理安全。  2）有利于缺陷的清楚表达。  3）系统灵活，强大的可配置能力。  4）自动发送E-MAIL通知相关人员。  功能测试工具：  Winrunner是企业级的功能测试工具，用于检测应用程序是否达到预期的功能及正常运行。 qtp是自动化测试工具  压力测试工具：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。通过以模拟成千上万用户实施并发负载及实习性能检测的方式来确认和查找问题。可以最大限度的缩短测试时间，优化性能和加速应用系统的发布周期。 | 46 |
| pc17 | 的自动化测试工具。 | 1）错误编号（每个软件错误都需要有一个唯一的编号）。      2）软件名称和版本号（错误所属的软件名称、版本号）。      3）错误的严重程度（用1-5或者公司规定的其他形式来表示）。      4）错误概要（描述错误的标题）。      5）报告人。      6）发现错误的时间。      7）承办人（由项目经理制定相关程序员修改错误）。      8）错误的优先级（一般由项目经理来指定）。      9）错误状态（描述错误所处的处理状态）。      10）注释。 | 48 |
| pc17 | 75.基于WEB信息管理系统测试时应考虑的因素有哪些？ | 1）有租好的性能，不咬在并发用户很多是响应速度很慢。  2）有足好的兼容性，当我们用IE之外的浏览器访问时，仍能够正常地使用该网站。  3）有足够强的安全性。登陆系统的用户名和密码不能够被他人轻易的获得。  4）有好的正确性。当我们点击添加按钮时，不希望出现的是删除的界面。 | 49 |
| pc17 | 76.软件本地化测试比功能测试都有哪些方面需要注意？ | 本地化测试的对象是软件的本地化版本。本地化测试的目的是测试特定目标区域设置的软件本地化质量。本地化测试的环境是在本地化的操作系统上安装本地化的软件。从测试方法上可以分为基本功能测试，安装/卸载测试，当地区域的软硬件兼容性测试。测试的内容主要包括软件本地化后的界面布局和软件翻译的语言质量，包含软件、文档和联机帮助等部分 | 53 |
| pc17 | 77.测试生命周期、测试过程分为几个阶段，以及各阶段的含义？ | 同68。  软件测试生命周期包括6个阶段（大体上）：  1）计划 2）分析，3）设计，4）构建，5）测试周期，6）最后测试和实施， 7）实施后。 | 58 |
| pc17 | 78.引入测试管理的含义？ | 测试过程中所涉及到的人、活动和工具都是很多的（特别是大型软件的测试中），在制定测试计划时，要对这些因素加以管理。在测试管理阶段，需要考虑的主要问题包括：      选择缺陷管理工具和测试管理工具。      定义工作进度。      建立风险管理计划。 | 62 |
| pc17 | 79.什么是软件测试静态分析？ | 不通过执行来测试一个系统。如代码检查，文档检查和评审等。 | 63 |
| pc17 | 80.需求测试注意事项有哪些？ | 需求测试的内容：          需求文档是否符合公司的格式要求？          需求是否正确？          要保证需求文档中所描述的内容是真实可靠的？          这是“真正的“需求吗?描述的产品是就是要开发的产品?          需求是否完备?第一个版本是否需要更多的功能?          需求是否兼容?需求有时候是矛盾的.  需求是否可实现?需求设想的硬件是否比实际运行的要快?他们要求的内存、I/O设备是否太多 ，要求的输入或输出设备的粪便率是否过高？  需求是否合理？在开发进度、开发费用、产品性能、可靠性和内存使用之间存在着平衡关系 。  需求是否可测？ | 78 |
| pc17 | 69.简述你对测试工作的认识过程、在以后的工作的一些建议。（6 分） | 69、简述你对测试工作的认识过程、在以后的工作的一些建议。（6 分） | 81 |
| pc2 | 16(1).缺陷记录包含的内容 | 1.基本信息  (1)编号 (2)版本号 (3)严重程度 (4)提交人 (5)日期等  2.错误描述  3.测试环境说明  4.其他附件(如屏幕截图) | 11 |
| pc2 | 16(2)如何提交高质量的缺陷报告 | 1.重点说明问题重现的步骤和方法  2.分析错误,用最少的错误描述  3.报告应完备,易读,且没有敌意  4.不轻易猜测错误原因  5.进行演示和使用文件  6.立即报告 | 25 |
| pc2 | 21.主流的软件工程方法论和思想 | RUP（Rational Unified Process，统一软件开发过程，统一软件过程)是一个面向对象且基于网络的程序开发方法论。  ＣＭＭ（Ｃａｐａｂｉｌｉｔｙ Ｍａｔｕｒｉｔｙ Ｍｏｄｅｌ，简称ＣＭＭ）是由美国软件工程学会制定的一套专门针对软件产品的质量管理和质量保证标准。  CMMI全称是Capbility Maturity Model Integration，即集成的能力成熟度模型，是由美国国防部与卡内基-梅隆大学和美国国防工业协会共同开发和研制，与2002年4月推出了系统工程和软件工程的集成成熟度模型。CMMI是一套融合多学科的、可扩充的产品集合，同时也是工程实践与管理方法。 | 43 |
| pc2 | 25.软件的概念和特点?软件复用的含义?软件包括哪些? | 软件是一系列按照特定顺序组织的计算机数据和指令的集合。  软件复用是将已有的软件及其有效成分用于构造新的软件或系统。它不仅是对软件程序的复用，还包括对软件生产过程中其它劳动成果的复用，如项目计划书、可行性报告、需求分析、概要设计、详细设计、编码(源程序)、测试用例、文档与使用手册等等。因此，软件复用包括软件产品复用和软件过程复用两部分的内容。 | 68 |
| pc2 | 26.瀑布模型和螺旋模型的主要区别是什么? | 瀑布模型有六个阶段:计划,需求分析,设计,编码,测试,运行维护.同时测试阶段比较集中,是一种线性组织形式.  螺旋模型有四个阶段:计划,风险分析,实施工程,评审.它将瀑布模型和原型模型结合起来,呈螺旋周期性组织形式. | 90 |
| pc2 | 15.性能测试工作的目的是什么?做好性能测试工作的关键是什么? | 性能测试的目的：通过测试确认软件是否满足产品的性能需求，同时发现系统中存在的性能瓶颈，起到优化系统 的目的。  性能测试的关键:关键是测试脚本的录制，测试时候测试环境的干净。 | 132 |
| pc21 | 81 测试分析测试用例注意事项 | 1）一个测试用例应当有完整的信息，例如：测试用例ID号，测试用例名字，测试用例的目的，测试条件，输入数据需求，步骤和期望结果，  2）一个好的测试用例应该容易发现软件错误，好的测试用例要有可重复性，好的测试用例必须清晰地定义一个或多个期望的结果以及测试通过和失败标准。好的测试用例的另外一个特征是测试用例是没有冗余。  3）测试用例的作用：指导测试的实施，作为编写测试脚本的“设计规格说明书“自动测试的主要任务是编写测试脚本，评估测试结果的度量基准，分析缺陷的标准/ | 20 |
| pc21 | 82怎么评价软件测试工程师？ | 作为软件测试工程师，其职业素质要求包括；  责任心，学习能力，怀疑精神，沟通能力，专注力，团队精神，注重积累  转移情绪能力，技术能力，自信心，外交能力，幽默感，很强的记忆力，耐心，自我督促，洞察力。 | 59 |
| pc21 | 84按瀑布模型软件开发分哪几个阶段？对应的测试环节又分哪几个阶段？ | 按瀑布模型软件测试分为  一：定义阶段：（1）计划（2）需求分析，  二：开发阶段：（1）设计（2）编码 （3）测试  三：维护阶段；（1）运行 （2）维护 | 75 |
| pc21 | 85测试都包括哪些内容？ | 测试计划内容和要点：  测试活动进度综述，可供项目经理产生项目进度的参考。  测试方法，包括测试工具的使用。  测试工具，包括如何和何时获取工具。  实施测试和报告结果的过程。  系统测试进入和结束准则。  设计、开发和执行测试所需的人员。  设备资源，需要什么机器和测试基准。  恰当的测试覆盖率目标。  测试所需的特殊软件和硬件配置。  测试应用程序策略。  测试哪些特性，不测试哪些特性。  风险和意外情况计划。 | 83 |
| pc21 | 86什么是兼容性测试？兼容性测试侧重哪些方面，请按照优先级用矩阵图表列出 | 兼容性测试是为了检验程序与很多可能的硬件或系统软件配置之间的兼容情况 | 84 |
| pc21 | 93你在测试过程中从事哪些工作？最擅长哪些工作？ | 测试计划，测试设计与开发，执行测试  完成测试用例设计并编写必要的测试 驱动程序 | 99 |
| pc21 | 92描述软件开发的过程，完整的开发过程需要完成哪些工作？ | 软件开发的过程：1）定义：明确软件开发的目标，软件的需求  2）计划：制订软件开发所涉及的各种计划。  3）实现：进行设计，编码，文档编写工作，完成所要求开发的软件特征。  4）稳定化：以测试和缺陷修复工作为主，确保将提交的软件具有良好的质量  5）部署： 安装，提交开发完成的软件，建立可供用户使用环境。 | 102 |
| pc21 | 91描述软件测试的过程，其中有哪些工作要做？ | 软件测试过程； 测试计划，测试设计与开发，执行测试  测试计划中测试人员的任务是首先对需求进行分析，最终定义一个测试集合。  测试设计与开发中软件测试人员根据软件需求，软件设计说明书，完成测试用例设计并编写必要的测试 驱动程序  执行测试中主要是建立测试环境，根据前面编写的测试计划和测试用例运行测试，记录测试结果，报告软件缺陷，跟踪软件缺陷，直至被处理，分析测试结果。 | 103 |
| pc21 | 88写出你常用的测试工具？ | 功能测试工具： winrunner qtp  性能测试工具：loadrunner  测试管理工具：bugzilla mantis bugs bugreee TD | 119 |
| pc21 | 89 LORDRUNNER分哪三部分？ | Analysis.lnk    controller.lnk    virtual user Generator.lnk | 126 |
| pc21 | 90希望以后的软件测试是一个怎么样的行业？ | 1）    该角色在整个软件开发中不可或缺  2）    该角色需要面对巨大的技术挑战，可以不断地积累经验和技术，从而做得更好  3）    该角色所需要掌握的技术内容是无止境的，永远都需要面对新的问题，需要进行新的学习  4）    要成为一名优秀的测试人员，需要有对特定行业的了解，相对全面的技术能力，对软件开发过程的深入了解。 | 131 |
| pc21 | 83软件测试是一个什么样的行业 | 软件测试是软件质量保证的重要手段，软件测试是为了发现程序中的错误分析或执行程序的过程。软件测试有两个基本的功能；验证和确认。软件测试的目标是发现软件错误。 | 138 |
| pc21 | 87写出你的职业生涯 | 无 | 144 |
| pc4 | 54.什么是软件测试 | 为了发现程序中的错误而执行程序的过程 | 12 |
| pc4 | 55简述集成测试的过程 | 系统集成测试主要包括以下过程：  1. 构建的确认过程。  2. 补丁的确认过程。  3. 系统集成测试测试组提交过程。  4. 测试用例设计过程。  5. 测试代码编写过程。  6. Bug的报告过程。  7. 每周/每两周的构建过程。  8. 点对点的测试过程。  9. 组内培训过程。 | 14 |
| pc4 | 56 怎么做好文档测试 | 仔细阅读，跟随每个步骤，检查每个图形，尝试每个示例。P142  检查文档的编写是否满足文档编写的目的  内容是否齐全，正确  内容是否完善  标记是否正确 | 16 |
| pc4 | 57 白盒测试有几种方法 | 总体上分为静态方法和动态方法两大类。  静态：关键功能是检查软件的表示和描述是否一致,没有冲突或者没有歧义  动态：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。 | 18 |
| pc4 | 58系统测试计划是否需要同行审批，为什么 | 需要，系统测试计划属于项目阶段性关键文档，因此需要评审。 | 67 |
| pc4 | 59 比较负载测试，容量测试和强度测试的区别 |  | 133 |
| pc4 | 60测试结束的标准是什么？ | (一)    基于测试用例的规则  (1)    先构造测试用例  (2)    在测试过程中，当测试用例的不通过率达到20%时，则拒绝继续测试，待开发人员  修正软件后再进行测试。  (3)    当 测试用例通过率达到100%，非功能性测试用例通过率达到90%时，允许正常结束测试。  (二)    基于“测试期缺陷密度“的规则，反测试一个CPU小时发现的缺陷数称为“测试期缺陷密度“。如果在相邻n个CPU小时内“测试期缺陷密度“全部低于某个值m时，则允许正常结束测试。例如，n大于10，m小于等于1。该规则比较适用于系统测试阶段。  (三)    基于“运行期缺陷密度“的规则，反软件运行一个CPU小时发现的缺陷数称为“运行期缺陷密度“。如果在相邻n个CPU小时内“运行期缺陷密度“全部低于某个值m时，则允许正常结束测试。例如，n大于100，m小于等于1。该规则比较适用于验收测试阶段。 | 136 |
| pc4 | 62 请描述软件测试活动的生命周期。. | 软件生命周期包括了定义、计划、实现、稳定化、部署、运行与维护这样六个阶段的过程。 | 140 |
| pc4 | 66．如果能够执行完美的黑盒测试，还需要进行白盒测试吗？为什么？. | 需要，因为在进行-黑盒测试时，并不关注程序的内部结构本身，而是注重测试软件的功能需求和性能需求。因此，黑盒测试也称为功能测试或数据驱动测试，它是在以知产品所具有的功能，通过测试来检测每个功能是否都能正常使用。黑盒测试完全不考虑程序内部结构和内部特征的情况下，测试者在程序接口进行测试，它只检查程序功能是否按照需求规格说明书的正常使用，程序是否能适当地接收输入数据而产生正确的输出信息，并且保持外部信息的完整。  白盒测试是对软件的过程性细节做细致的检查。它允许测试人员利用程序内部的逻辑结构及有关信息，设计或选择测试用例，对程序所有逻辑路径进行测试 | 142 |
| pc4 | 67.产品测试到什么时候就算足够? | (一)    基于测试用例的规则  (1)    先构造测试用例  (2)    在测试过程中，当测试用例的不通过率达到20%时，则拒绝继续测试，待开发人员  修正软件后再进行测试。  (3)    当 测试用例通过率达到100%，非功能性测试用例通过率达到90%时，允许正常结束测试。  (二)    基于“测试期缺陷密度“的规则，反测试一个CPU小时发现的缺陷数称为“测试期缺陷密度“。如果在相邻n个CPU小时内“测试期缺陷密度“全部低于某个值m时，则允许正常结束测试。例如，n大于10，m小于等于1。该规则比较适用于系统测试阶段。  (三)    基于“运行期缺陷密度“的规则，反软件运行一个CPU小时发现的缺陷数称为“运行期缺陷密度“。如果在相邻n个CPU小时内“运行期缺陷密度“全部低于某个值m时，则允许正常结束测试。例如，n大于100，m小于等于1。该规则比较适用于验收测试阶段。 | 143 |
| pc4 | 63.什么是测试评估,测试评估的范围是什么? | 测试评估摘要用于整理并展示复审与评估的测试结果及测试的主要评测方法。此外，测试评估摘要包括测试员和测试设计员对这方面信息的评估，还要包括他们对将来工作的建议。 | 146 |
| pc5 | 27．软件生存周期及其模型是什么？ | 软件生命周期是从软件的诞生到最后被停止使用的完整周期可以包括定义、计划、实现、稳定化、部署、运行与维护六个阶段  模型包括：瀑布模型 原型模型 增量模型 螺旋模型 | 24 |
| pc5 | 28.什么是软件测试？软件测试的目的和原则？ | 软件测试就是为了发现程序中的错误而分析或执行程序的过程。  软件测试的目的：(1)测试是为了发现程序中的错误而执行程序的过程；  　　　　　　　　(2)好的测试方案是极可能发现迄今为止尚未发现的错误的测试方案；  　　　　　　　　(3)成功的测试是发现了至今为止尚未发现的错误的  软件测试的原则：1) 所有的测试都应追溯到用户需求。2) 应尽早地和不断地进行软件测试3) 在有限的时间和资源下进行测试5) 充分关注测试中的集群现象。6) 程序员应避免检查自己的程序。7) 尽量避免测试的随意性 | 29 |
| pc5 | 29.软件配置管理的作用？软件配置包括什么？ | 配置管理是通过技术或行政手段对软件产品及其开发过程和生命周期进行控制、规范的一系列措施。配置管理的目标是记录软件产品的演化过程，确保软件开发者在软件生命周期中各个阶段都能得到精确的产品配置。  软件配置包括：它包括控制、记录、追踪对软件的修改和每个修改生成的软件组成部件来实现对软件产品的管理功能 | 31 |
| pc5 | 30.简述需求分析的过程和意义？ | 需求分析的过程：（1）收集用户的需求 （2）编写需求定义文档（3）编写软件功能说明（4）编写软件需求跟踪矩阵（5）审核软件需求文档  需求分析的意义：明确软件具体要实现哪些功能、以便在接下来的概要设计中进一步分析，同时也是对用户需求的一个总结，通过它可以和用户更好的交流 | 51 |
| pc5 | 31.数据库中什么是数据的对立性？有几个层次？ | 31.数据库中什么是数据的对立性？有几个层次？ | 65 |
| pc5 | 32.数据库中网状、层次数据模型与关系模型的最大区别是什么？ | 数据之间的联系的表示方式不同(记录型之间的联系方式不同)。 | 93 |
| pc5 | 33.数据库dbms读取一条记录时发生哪些事件 ？ | 33.数据库dbms读取一条记录时发生哪些事件 ？ | 94 |
| pc5 | 34.什么是软件质量？软件包是什么？ | 软件质量是指适当的、无错误的，满足用户的需求，能在预算内按时完成，满足所定的目标期望，并且是可维护的。  软件包是指具有特定的功能，用来完成特定任务的一个程序或一组程序。 | 96 |
| pc5 | 35.什么是软件质量的保证?其主要任务是什么？ | 软件质量保证是为了确保软件开发过程和结果符合预期的要求，而建立的一系列规程，以及依照规程和计划采取的一系列活动及其结果评价。  主要任务（１）用户要求定义(2)力争不重复劳动(3)掌握开发新软件的方法(4)组织外部力量协作(5)排除无效劳动  (6)发挥每个开发者的能力（７)提高计划和管理质量 | 111 |
| pc5 | 36.软件质量保证体系是什么？国家标准中与质量保证管理相关的几个标准是什么？他们的编号和全称是什么？ | 36.软件质量保证体系是什么？国家标准中与质量保证管理相关的几个标准是什么？他们的编号和全称是什么？ | 113 |
| pc5 | 37.软件测试的原则与策略是什么? | 测试的原则：严格依据测试需求分析文档进行软件测试  软件测试的策略：（１）测试的范围（２）测试的方法（３）质量的标准（４）自动化测试的工具（５）测试软件测得编写 | 117 |
| pc5 | 38.什么是测试用例？什么是测试脚本？两者的关系是什么？ | 测试用例是为特定目标开发的测试输入、执行条件和与预期结果的集合。  测试脚本是在执行测试用例时，自动化工具生成的脚本进行的操作  两者的关系:测试脚本是依据测试用例设计的测试脚本，它可以多个有一个脚本对应多用例 | 122 |
| pc5 | 39.测试问题的严重性分为几级？如何区分？ | （1）致命错误 包括：由于程序所引起的死机,非法退出、死循环、数据库发生死锁、因错误操作导致的程序中断、功能错误、与数据库连接错误、数据通讯错误  （2）严重错误 包括：规定的功能没有实现或不完整或产生错误结果；设计不合理造成性能低下，影响系统的运营；是系统不稳定或破坏数据。  （3）一般错误 包括：通常指不影响业务的功能使用。如操作界面错误；打印内容、格式错误；简单的输入限制未放在前台进行控制；删除错作为给出提示；数据库表中有过多的空字段等。  （4）轻微错误 包括：通常指界面的拼写错误或用户使用不方便等小问题或需要完善的问题。如耗比较长的操作为给出提示；可输入和只读区没有明显区分。  （5）改进建议 改进建议一般只软件中值得改良的地方。 | 124 |
| pc5 | 40.测试用例设计的原则是什么？目前主要的测试用例设计方法是什么？ | 测试用例设计的原则：  　　　　　　　充分注意测试中的群集现象。经验表明，测试后程序中残存的错误数目与该程序中已发现的错误　　　　　数目成正比。  　　　　　　　严格执行测试计划，排除测试的随意性。  　　　　　　　应当对每一个测试结果做全面检查。  　　　　　　　妥善保存测试计划，测试用例，出错统计和最终分析报告，为维护提供方便。  测试用例的设计方法：对于白盒测试有逻辑覆盖法、基本路径法、辅助模块设计  对于黑盒测试有等价类划分、边界值分析、因果图、错误推测 | 127 |
| pc6 | 41.(答案不确定)结构化系统测试和功能性系统测试分别采用了哪些方法和技术 | 结构化系统测试一般采用白盒测试方法；功能性系统测试一般采用黑盒测试方法进行 | 23 |
| pc6 | 43.在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 | a 测试需求分析（需求文档）  测试需求是整个测试过程的基础；确定测试对象以及测试工作的范围和作用。用来确定整个测试工作（如安排时间表、测试设计等）并作为测试覆盖的基础。而且被确定的测试需求项必须是可核实的。  ·测试需求是制订测试计划的基本依据，确定了测试需求能够为测试计划提供客观依据；  ·测试需求是设计测试用例的指导，确定了要测什么、测哪些方面后才能有针对性的设计测试用例；  ·测试需求是计算测试覆盖的分母，没有测试需求就无法有效地进行测试覆盖。  b　测试计划（测试计划文档）  对需求进行分析，定义测试集合，测试策略（包括确定测试范围，选择测试方法，确定测试标准，测试工具的选择等），测试环境及如何配置测试环境，测试管理并定义工作进度。  c 　测试设计与开发（测试设计文档和评审过的测试用例）  测试总体设计（包括定义设计目标，定义数据文件、配置文件等的相关输入说明，定义并配置测试环境，提供测试设计文档），开发测试用例（定义详细测试过程，黑盒、白盒测试用例编写），评审测试用例。  d　 执行测试（报告测试结果，提交Error report，提交测试总结报告）  环境配置好了，新的版本也收到了，人员也都培训好了，可按计划开始执行测试，单元测试、集成测试、系统测试  e　 测试评价：对版本测试覆盖率，测试质量，人员测试工作以及前期的一些工作制定情况进行评价，评估，测试结果的统计信息 | 28 |
| pc6 | 42.面向对象的测试用例设计有几种方法 如何实现 | 面向对象的开发模型突破了传统的瀑布模型，将开发分为面向对象分析（OOA）面向对象设计（OOD）和面向对象编程（OOP）三个阶段。针对这种开发模型，结合传统的测试步骤的划分，我们把面向对象的软件测试分为：面向对象分析的测试、面向对象设计的测试、面向对象编程的测试、面向对象单元测试、面向对象集成测试、面向对象确认和系统测试。建议是一种整个软件开发过程中不断测试的测试模型，使开发阶段的测试与编码完成后的单元测试、集成测试、系统测试成为一个整体。 | 33 |
| pc6 | 47.测试计划的目的是什么？ 内容,最重要的是哪部分,关键是什么 | 答：为了更好的开展测试工作，需要设计与编码小组，市场部门，客户支持部门，项目管理部门知道测试要做什么，需要什么资源，并且还更多争取他们的认可和支持。此外要更好的完成测试工作还需要考虑更多的细则问题，写测试计划能够整理自己的思维想法和记忆。计划的最终目的是为了交流。  内容：测试需求、测试策略、测试资源、风险和意外。重要的是要很好的分析需求，从而定义良好的测试策略。 | 35 |
| pc6 | 48.软件测试应该划分几个阶段?简述各个阶段应重点测试的点?各个阶段的含义？ | 单元测试  单元测试是对软件中的基本组成单位进行的测试，如一个模块、一个过程等等。它是软件动态测试的最基本的部分，也是最重要的部分之一，其目的是检验软件基本组成单位的正确性。一个软件单元的正确性是相对于该单元的规约而言的。因此，单元测试以被测试单位的规约为基准。单元测试的主要方法有控制流测试、数据流测试、排错测试、分域测试等等。  集成测试  集成测试是在软件系统集成过程中所进行的测试，其主要目的是检查软件单位之间的接口是否正确。它根据集成测试计划，一边将模块或其他软件单位组合成越来越大的系统，一边运行该系统，以分析所组成的系统是否正确，各组成部分是否合拍。集成测试的策略主要有自顶向下和自底向上两种。  系统测试  系统测试是对已经集成好的软件系统进行彻底的测试，以验证软件系统的正确性和性能等满足其规约所指定的要求，检查软件的行为和输出是否正确并非一项简单的任务，它被称为测试的“先知者问题“。因此，系统测试应该按照测试计划进行，其输入、输出和其他动态运行行为应该与软件规约进行对比。软件系统测试方法很多，主要有功能测试、性能测试、随机测试等等。  验收测试  验收测试旨在向软件的购买者展示该软件系统满足其用户的需求。它的测试数据通常是系统测试的测试数据的子集。所不同的是，验收测试常常有软件系统的购买者代表在现场，甚至是在软件安装使用的现场。这是软件在投入使用之前的最后测试。  回归测试  回归测试是在软件维护阶段，对软件进行修改之后进行的测试。其目的是检验对软件进行的修改是否正确。这里，修改的正确性有两重含义：一是所作的修改达到了预定目的，如错误得到改正，能够适应新的运行环境等等；二是不影响软件的其他功能的正确性。 | 87 |
| pc6 | 45.如果能够执行完美的黑盒测试，还需要进行白盒测试吗？为什么？ | 答：当然需要，因为黑盒只能完成功能方面的测试 | 91 |
| pc6 | 44.软件的安全性应从哪几个方面去测试 | 答：安全性测试，这里的安全性主要包括了两部分：数据的安全性和操作的安全性。核实只有规格规定的数据才可以访问系统，其他不符合规格的数据不能够访问系统；核实只有规格规定的操作权限才可以访问系统，其他不符合规格的操作权限不能够访问系统  用户认证安全的测试要考虑问题：  1. 明确区分系统中不同用户权限  2. 系统中会不会出现用户冲突  3. 系统会不会因用户的权限的改变造成混乱  4. 用户登陆密码是否是可见、可复制  5. 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）  6. 用户推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问题  1. 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  2. 模拟非授权攻击，看防护系统是否坚固  3. 采用成熟的网络漏洞检查工具检查系统相关漏洞（即用最专业的黑客攻击工具攻击试一下，现在最常用的是 NBSI 系列和 IPhacker IP ）  4. 采用各种木马检查工具检查系统木马情况  5. 采用各种防外挂工具检查系统各组程序的客外挂漏洞  数据库安全考虑问题：  1. 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）  2. 系统数据的完整性（我刚刚结束的企业实名核查服务系统中就曾存在数据的不完整，对于这个系统的功能实现有了障碍）  3. 系统数据可管理性  4. 系统数据的独立性  5. 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） | 97 |
| pc6 | 46.产品测试到什么时候就算是足够了？ | 答：规定测试策略和应达到的目标：  1）白盒测试法：一般可规定以完全覆盖即语句和分支覆盖必须达到100%  2）黑盒测试法：根据情况设计测试用例集，当测试用例完全被测试完即可结束。  规定至少要查出的错误数：如果经验丰富事先规定需查出的错误数，当规定的错误数量的错误测出后即可终止。 | 100 |
| pc6 | 49.针对缺陷采取怎样的管理措施？ | 答：  1.要更好的管理缺陷，必须引入缺陷管理工具，商用的或者开源的都可。  2.根据缺陷的生命周期，考虑缺陷提交的管理、缺陷状态的管理和缺陷分析的管理。  3.所有发现的缺陷（不管是测试发现的还是走读代码发现的）都必须全部即时的、准确的提交到缺陷管理工具中，这是缺陷提交的管理。  4.缺陷提交后，需要即时的指派给相应的开发人员，提交缺陷的人需要密切注意缺陷的状态，帮助缺陷的尽快解决。缺陷解决后需要即时对缺陷的修复进行验证。这样的目的有两个：一个是让缺陷尽快解决；二是方便后面缺陷的分析（保证缺陷相关的信息准确，如龄期等），这是缺陷状态的管理。  5.为了更好的改进开发过程和测试过程，需要对缺陷进行分析，总结如缺陷的类别、缺陷的龄期分布等信息，这是缺陷分析的管理。 | 115 |
| pc6 | 50.试叙述对一个软件项目测试的全过程。 | 见43题 | 125 |
| pc6 | 51.简述你对测试工作的认识过程、在以后的工作的一些建议。 | 开始只知道测试就是找bug，随着认识的深入逐渐了解了测试在软件项目中的作用，测试的整个流程，在测试的各个阶段需要什么文档、工具来进行测试，测试时采用的手段、方法和技术等， | 128 |
| pc6 | 52.简述静态测试和动态测试的区别？ | 答： 静态测试：又称做文档测试，代码测试，不运行程序，测试人员用眼睛看文档或程序，用头脑分析文档或源程序，用笔或电子编辑工具批注所发现的错误。这种测试发现60%以上的错误。  动态测试：又称做运行代码测试。是指在在计算机或网络运行上运行被测试的系统。按事先规定的测试计划，运行事先准备的测试用例。取得运行后的数据再将此数据与测试计划中的期望结果相互比较，若二者一致则通过；若二者不一致，则表明程序有错误。此种测试只发现40%以下的错误。  1。按照是否运行被测软件区分为动态测试和静态测试。  静态测试：不运行被测软件，比如文档评审，代码走读等，其实就是我们用眼睛看用脑子想；  动态测试：是要运行被测软件的。  2。依据被测对象信息不同，分为黑盒测试和白盒测试。  黑盒测试：依据被测对象的整体、外在信息来进行测试，把被测软件当成个黑盒子就行了，只要完成它的特定功能阿，不需要管它内部结构；  白盒测试：依据被测对象的具体内部构造来进行测试，要了解程序编码的程度。  灰盒测试：依据被测对象的整体功能信息和内部接口信息进行测试。 | 134 |
| pc6 | 53.阶段评审与同行评审的区别 | 答：在软件开发过程中的适当阶段对软件阶段产品进行评审，是确保软件产品最终质量的重要方法。阶段评审可以对某个开发阶段的阶段产品进行评审，也可以对某几个开发阶段的阶段产品进行综合评审。  在同行评审中，由软件工作产品创建者的同行们检查该工作产品，识别产品的缺陷，改进产品的不足。包括了执行评审的步骤和两种非正式的同行评审，走查和轮查，以及对每个评审选择适当方法的指南。 | 135 |
| pc7 | 108、什么是测试评估，测试评估的范围是什么？ | 定义：为了确定一特定的软件模块、软件包或软件产品是否验收合格或发布而把特定的评估准则应用到该软件模块、软件包或软件产品上去的活动。软件评估（Softwareassessment）的实质是对软件质量的度量与评价。其目的是“确定被评对象是否验收合格或发布“。  范围：软件评估的对象是“软件模块、软件包或软件产品“ | 2 |
| pc7 | 109、阐述工作版本的定义 | Build工作版本，软件开发过程中用于内部测试的功能和性能等不完善的软件版本。工作版本既可以是系统的可操作版本，也可以是展示要在最终产品中提供的部分功能的部分系统。 | 4 |
| pc7 | 110、请画出软件测试活动的流程图 | 测试过程：  用户需求———————–验收测试            回  规格定义—————–系统测试            归  设计——————集成测试            测  编码————单元测试            试  配置管理—————————————缺陷跟踪 | 6 |
| pc7 | 111、软件测试主要有哪几个阶段，每个阶段都应用什么测试方法？ | 主要有单元测试、集成测试、系统测试、验收测试四个阶段：  单元测试：白盒测试、静态测试———-开发工程师  集成测试：黑盒测试————————-开发工程师、测试工程师  系统测试：黑盒测试————————-测试工程师  验收测试：黑盒测试——-开发工程师、测试工程师、产品经理或其它高级经理、用户 | 10 |
| pc7 | 112、输入3个整数作为三角形的边，为判断三角形是否有效、等腰、等边设计用例？ | 黑盒法：  设三角形的三条边分别为A, B, C。如果它们能够构成三角形的三条边，必需满足：  A > 0，B > 0，C > 0，且A + B > C，B + C > A，A + C > B。  如果是等腰的，还要判断是否A = B，或B = C，或A = C。  对于等边的，则需判断是否A = B，且B = C，且A = C。  列出等价类表：  输入条件     有效等价类     无效等价类  是否三角形的三条边 (A>0) (1), (B>0) (2),(C>0) (3), (A+B>C)(4)，(B+C>A) (5), (A+C>B) (6)      A￡0 (7), B￡0 (8), C￡0 (9),A +B￡C (10), A +C￡B (11)B+C￡A (12)  是否等腰三角形    (A = B) (13), (B = C) (14),(A = C) (15) (A |B) and (B|C) and (A |C) (16)  是否等边三角形    (A = B) and (B = C) and (A = C) (17) (A|B) (18), (B|C) (19),(A |C) (20)  设计测试用例：输入顺序是〖A，B，C〗  §〖3，4，5〗覆盖等价类 (1), (2), (3), (4), (5), (6)。满足即为一般三角形。  §〖0，1，2〗覆盖等价类 (7)。不能构成三角形。 若不考虑特定A, B, C，  §〖1，0，2〗覆盖等价类 (8)。同上。         三者取一即可  §〖1，2，0〗覆盖等价类 (9)。同上。  §〖1，2，3〗覆盖等价类 (10)。同上。 若不考虑特定A, B, C，  §〖1，3，2〗覆盖等价类 (11)。同上。 三者取一即可  §〖3，1，2〗覆盖等价类 (12)。同上。  §〖3，3，4〗覆盖等价类 (1), (2), (3), (4), (5), (6), (13)。 满足即为等腰三角形，  §〖3，4，4〗覆盖等价类 (1), (2), (3), (4), (5), (6), (14)。 若不考虑特定A, B, C，  §〖3，4，3〗覆盖等价类 (1), (2), (3), (4), (5), (6), (15)。 三者取一即可  §〖3，4，5〗覆盖等价类 (1), (2), (3), (4), (5), (6), (16)。 不是等腰三角形。  §〖3，3，3〗覆盖等价类 (1), (2), (3), (4), (5), (6), (17)。 是等边三角形  §〖3，4，4〗覆盖等价类 (1), (2), (3), (4), (5), (6), (14), (18)。 不是等边三角形，  §〖3，4，3〗覆盖等价类 (1), (2), (3), (4), (5), (6), (15), (19)。 若不考虑特定A, B, C，  §〖3，3，4〗覆盖等价类 (1), (2), (3), (4), (5), (6), (13), (20)。 三者取一即可  白盒法：  序号 测试内容        测试数据     预期结果  a          b      c  1 等边      5,5,5 4,4,4          等边  2 等腰 4,4,5 5,4,4          等腰  3 任意      3,4,5          任意  4 非三角形      9,4,4 4,9,4 4,4,9 No  5 退化三角形 8,4,4 4,8,4 4,4,8 No  6 零数据 0,4,5 4,0,5 4,5,0 No  7 零数据 0,0,0 No  8 负数据      -3,4,5 3,-4,5 3,4-5 运行出错  9 负数据         -3,-4,-5 运行出错  10 遗漏数据 3,4             运行出错  11 非整数 3.3,4,5         运行出错  12 非数字符          A,4,5         类型不符 | 32 |
| pc7 | 113、你了解的自动化测试工具？都有什么特点？ | WinRunner、QuickTest Professional  WinRunner：WR会用一种类C的测试脚本语言TSL生成一个测试脚本，1）可以轻松的创建测试，只需用鼠标键盘完成一个标准的业务操作流程，WR自动记录用户的操作并生成所需要的代码。2）插入检查点，在记录测试过程中可插入检查点，检查某时刻/状态下，应用程序是否运行正常。3）检验数据，WR可以验证数据库的数值，确保业务交易的准确性。4）增强测试 WR提供了数据驱动向导，可以将业务流程测试转化为数据驱动测试，反映多个用户各自的真实行为。5）运行测试并分析结果 WR会自动操作应用程序，并通过交互式的报告工具来提供详尽、易读的报告。6）维护测试 WR可以创建在整个应用程序生命周期内都能重复使用而不必重新创建测试。  QuickTest Professional：在建置（build）完新版本的应用程序或网站后，可以重复使用测试脚本进行测试；QuickTest会仿真鼠标的动作与键盘的输入。具有快速(Fast)、可靠(Reliable)、可重复(Repeatable)、可程序化(Programmable)、广泛的(Comprehensive)、可再使用(Reusable)等特点。 | 45 |
| pc7 | 114、写出下列端口号 | ftp—————-21  dns—————-53  http————–80  pop3————-110  smtp————–25  telnet————23  windows终端服务—————-3389 | 54 |
| pc7 | 115、写出下面的中文 | VLAN：虚拟局域网  WLAN：无限局域网  WAN：广域网  VPN：虚拟专网 | 60 |
| pc7 | 116、写出下列Linux命令 | kill 删除执行中的程序或工作  mv 移动或更名现有的文件或目录  rmdir 删除目录  ls 列出目录内容  df 显示磁盘的相关信息 -a 包含全部的文件系统 | 72 |
| pc7 | 117、抽象类和接口的区别 | 抽象类一般做为公共的父类，为子类的扩展提供基础，为属性和行为上的扩展，  接口不考虑属性，一般只考虑方法，使子类可以自由填补或扩展接口所定义的方法。 | 80 |
| pc7 | 118、编程链表反置 | fp:前指针;p:当前指针;np后指针  fp=head; /\*将前指针指向头指针，即fp指向头结点\*/  p=fp-> next; /\*当前指针p指向第二个结点 \*/  fp-> next=null;          /\*释放头结点 \*/  if (p!=null) /\*如果当前指针不为空\*/  {  while(p-> next!=null) /\*如果当前指针的下一个结点不为空\*/  {  np=p-> next; /\*将后指针指向当前结点的下一个结点 \*/  p-> next=fp; /\*将当前结点的next指针指向fp \*/  fp=p; /\*移动前指针，使其指向当前结点\*/  p=np; /\*将当前指针指向下一个结点\*/  }  if (p-> next=null) /\*当当前结点为最后一个结点\*/  p-> next=fp; /\*将最后一个结点的next指针指向当前的前一个结点\*/  }  head=p; /\*将头指针指向最后一个结点，成为新的头结点\*/ | 86 |
| pc7 | 119、防火墙中有哪些有效的设置方法？ | 如果我们的计算机是采用拨号上网的方式，那么就不存在防火墙的设置问题。但是如果是采用LAN或者WAN上网，出于安全性的考虑，就非常有必要设置防火墙，以隔离外来的未经授权的连接。　如果我们上网的LAN或WAN支持FTP代理协议，那么我需要在CuteFTP中填写下面的信息：  1 防火墙主机的IP地址  2 作为FTP代理服务器的代理端口  3 用户名和密码 | 89 |
| pc7 | 120、写ISO模型，tcp/udp在哪一层？他们有什么区别？ | ISO模型：应用层（Application layer）  表示层（Presentation layer）      会话层（Session layer）  传输层（Transport layer）      网络层（Network layer）  数据链路层（Data link layer）      物理层（Physical layer）  Tcp/udp是传输层，Tcp（传输控制协议）和 udp（用户数据报协议）  两者的主要区别在于udp不一定提供可靠的数据传输，Tcp的目的是提供可靠的数据传输，并在相互进行通信的设备或服务之间保持一个虚拟连接。Tcp在数据包接收无序、丢失或在交付期间被破坏时，负责数据恢复。当某个程序的目标是尽快地传输尽可能多的信息时（其中任意给定数据的重要性相对较低），可使用 udp。 | 92 |
| pc8hga | 94 软件测试计划工作的目的，内容、最主要的是哪部分，关键是什么？ | 目的：  软件测试计划是测试的基础，是整个测试工作的框架结构，可以有效地克服测试的盲目性,缩短测试周期,提高测试效率,并且起到测试文档与开发文档互查的作用。    内容：  产品基本情况调研、测试需求说明、测试策略和记录、测试资源配置、计划表、问题跟踪报告、测试计划的评审、结果等等。    重点：  测试的策略和记录这是整个测试计划的重点所在，既要描述如何公正客观地开展测试，又要考虑模块、功能、整体、系统、版本、压力、性能、配置和安装等各个因素的影响，要尽可能的考虑到细节，越详细越好，并制作测试记录文档的模板，为即将开始的测试做准备。    关键：  问题跟踪报告是整个测试的计划关键，这里面包括了如何准备去做一个问题报告以及如何去界义一个问题的性质，问题报告又包括问题的发现者和修改者、问题发生的频率、用了什么样的测试案例测出该问题的，以及明确问题产生时的测试环境等。 | 145 |
| pc8hga | 95、测试用例的设计方法有哪些，描述一具体实例。 | 黑盒测试  ·等价类划分方法  ·边界值分析方法  ·错误推测方法  ·因果图方法  等价类划分:  是把所有可能的输入数据,即程序的输入域划分成若干部分（子集）,然后从每一个子集中选取少数具有代表性的数据作为测试用例.该方法是一种重要的,常用的黑盒测试用例设计方法.  1) 划分等价类: 等价类是指某个输入域的子集合.在该子集合中,各个输入数据对于揭露程序中的错误都是等效的.并合理地假定:测试某等价类的代表值就等于对这一类其它值的测试.因此,可以把全部输入数据合理划分为若干等价类,在每一个等价类中取一个数据作为测试的输入条件,就可以用少量代表性的测试数据.取得较好的测试结果.等价类划分可有两种不同的情况:有效等价类和无效等价类.  有效等价类:是指对于程序的规格说明来说是合理的,有意义的输入数据构成的集合.利用有效等价类可检验程序是否实现了规格说明中所规定的功能和性能.  无效等价类:与有效等价类的定义恰巧相反.  设计测试用例时,要同时考虑这两种等价类.因为,软件不仅要能接收合理的数据,也要能经受意外的考验.这样的测试才能确保软件具有更高的可靠性.  2）划分等价类的方法:下面给出六条确定等价类的原则.  ①在输入条件规定了取值范围或值的个数的情况下,则可以确立一个有效等价类和两个无效等价类.  ②在输入条件规定了输入值的集合或者规定了”必须如何”的条件的情况下,可确立一个有效等价类和一个无效等价类.  ③在输入条件是一个布尔量的情况下,可确定一个有效等价类和一个无效等价类.  ④在规定了输入数据的一组值（假定n个）,并且程序要对每一个输入值分别处理的情况下,可确立n个有效等价类和一个无效等价类.  ⑤在规定了输入数据必须遵守的规则的情况下,可确立一个有效等价类（符合规则）和若干个无效等价类（从不同角度违反规则）.  ⑥在确知已划分的等价类中各元素在程序处理中的方式不同的情况下,则应再将该等价类进一步的划分为更小的等价类.  3）设计测试用例:在确立了等价类后,可建立等价类表,列出所有划分出的等价类:  输入条件 有效等价类 无效等价类  … … …  … … …  然后从划分出的等价类中按以下三个原则设计测试用例:  ①为每一个等价类规定一个唯一的编号.  ②设计一个新的测试用例,使其尽可能多地覆盖尚未被覆盖地有效等价类,重复这一步.直到所有的有效等价类都被覆盖为止.  ③设计一个新的测试用例,使其仅覆盖一个尚未被覆盖的无效等价类,重复这一步.直到所有的无效等价类都被覆盖为止.  边界值分析法  边界值分析方法是对等价类划分方法的补充.  （1）边界值分析方法的考虑:  长期的测试工作经验告诉我们,大量的错误是发生在输入或输出范围的边界上,而不是发生在输入输出范围的内部.因此针对各种边界情况设计测试用例,可以查出更多的错误.  使用边界值分析方法设计测试用例,首先应确定边界情况.通常输入和输出等价类的边界,就是应着重测试的边界情况.应当选取正好等于,刚刚大于或刚刚小于边界的值作为测试数据,而不是选取等价类中的典型值或任意值作为测试数据.  （2）基于边界值分析方法选择测试用例的原则:  1）如果输入条件规定了值的范围,则应取刚达到这个范围的边界的值,以及刚刚超越这个范围边界的值作为测试输入数据.  2）如果输入条件规定了值的个数,则用最大个数,最小个数,比最小个数少一,比最大个数多一的数作为测试数据.  3）根据规格说明的每个输出条件,使用前面的原则1）.  4）根据规格说明的每个输出条件,应用前面的原则2）.  5）如果程序的规格说明给出的输入域或输出域是有序集合,则应选取集合的第一个元素和最后一个元素作为测试用例.  6）如果程序中使用了一个内部数据结构,则应当选择这个内部数据结构的边界上的值作为测试用例.  7）分析规格说明,找出其它可能的边界条件.  错误推测法  错误推测法: 基于经验和直觉推测程序中所有可能存在的各种错误, 从而有针对性的设计测试用例的方法.  错误推测方法的基本思想: 列举出程序中所有可能有的错误和容易发生错误的特殊情况,根据他们选择测试用例. 例如, 在单元测试时曾列出的许多在模块中常见的错误. 以前产品测试中曾经发现的错误等, 这些就是经验的总结. 还有, 输入数据和输出数据为0的情况. 输入表格为空格或输入表格只有一行. 这些都是容易发生错误的情况. 可选择这些情况下的例子作为测试用例.  因果图方法  前面介绍的等价类划分方法和边界值分析方法,都是着重考虑输入条件,但未考虑输入条件之间的联系, 相互组合等. 考虑输入条件之间的相互组合,可能会产生一些新的情况. 但要检查输入条件的组合不是一件容易的事情, 即使把所有输入条件划分成等价类,他们之间的组合情况也相当多. 因此必须考虑采用一种适合于描述对于多种条件的组合,相应产生多个动作的形式来考虑设计测试用例. 这就需要利用因果图（逻辑模型）.  因果图方法最终生成的就是判定表. 它适合于检查程序输入条件的各种组合情况.  利用因果图生成测试用例的基本步骤:  (1) 分析软件规格说明描述中, 那些是原因(即输入条件或输入条件的等价类),那些是结果(即输出条件), 并给每个原因和结果赋予一个标识符.  (2) 分析软件规格说明描述中的语义.找出原因与结果之间, 原因与原因之间对应的关系. 根据这些关系,画出因果图.  (3) 由于语法或环境限制, 有些原因与原因之间,原因与结果之间的组合情况不不可能出现. 为表明这些特殊情况, 在因果图上用一些记号表明约束或限制条件.  (4) 把因果图转换为判定表.  (5) 把判定表的每一列拿出来作为依据,设计测试用例.  从因果图生成的测试用例（局部,组合关系下的）包括了所有输入数据的取TRUE与取FALSE的情况,构成的测试用例数目达到最少,且测试用例数目随输入数据数目的增加而线性地增加.  除了上述几种黑盒测试的测试用例设计方法之外其他方法还包括判定表驱动分析方法、正交实验设计方法、功能图分析方法等 | 147 |
| pc8hga | 96、测试用例的关键是什么？ | 对系统的熟悉程度，需求的理解及设计文档的了解情况 | 148 |
| pc8hga | 97、描述测试用例设计的完整过程 | 首先根据需求文档、概要设计、测试计划、测试方案 细分出各功能模块的测试项，再根据各测试项，按照概要设计、详细设计以及测试方案中测试的覆盖率 细分出测试子项，最后按照测试子项、根据测试用例的设计方法（因果图、边界值、等价类等的设计方法）书写测试用例。 | 149 |
| pc8hga | 98、性能测试工作的完整过程、目的、最关键的是什么？ | 性能高度过程大致可分为七步，分别为 制定目标和分析系统、选择测试度量的方法 、学习的相关技术和工具、制定评估标准、设计测试用例、运行测试用例、分析测试结果    目的是验证软件系统是否能够达到用户提出的性能指标，同时发现软件系统中存在的性能瓶颈，优化软件，最后起到优化系统的目的。  并发性能测试是关键，因为并发性能测试的过程是一个负载测试和压力测试的过程，即逐渐增加负载，直到系统的瓶颈或者不能接收的性能点，以通过综合分析交易执行指标和资源监控指标来确定系统并发性能的过程。 | 150 |
| pc8hga | 101、 单元测试和集成测试，描述工作实际开展的情况 | 单元测试是在软件开发过程中要进行的最低级别的测试活动，在单元测试活动中，软件的独立单元将在与程序的其他部分相隔离的情况下进行测试。集成测试，也叫组装测试或联合测试。在单元测试的基础上，将所有模块按照设计要求）如根据结构图〕组装成为子系统或系统，进行集成测试。 | 151 |
| pc8hga | 104 、对 RUP CMM CMMI XP PSP TSP 的认识 | RUP是Rational Unified Proces 的缩写，翻译成中文就是”统一软件过程”。  主要由 Ivar Jacobson的 The Objectory Approch 和 The Rational Approch 发展而来。同时，它既是文档化的软件工程产品，又是一套软件工程方法的框架，各个组织可根据自身的实际情况，以及项目规模对RUP进行裁剪和修改，以制定出合乎需要的软件工程过程。  RUP 吸收了多种开发模型的优点，具有很好的可操作性和实用性。从它一推出市场，凭借Booch、Ivar Jacobson、以及Rumbagh 在业界的领导地位以及与统一建模语言（Unified Model Language , 以下简称UML）的良好集成、多种CASE工具的支持、不断的升级与维护，迅速得到业界广泛的认同，越来越多的组织以它作为软件开发模型框架。    CMM是指”能力成熟度模型”，其英文全称为Capability Maturity Model for Software，英文缩写为SW-CMM，简称CMM。它是对于软件组织在定义、实施、度量、控制和改善其软件过程的实践中各个发展阶段的描述。CMM的核心是把软件开发视为一个过程，并根据这一原则对软件开发和维护进行过程监控和研究，以使其更加科学化、标准化、使企业能够更好地实现商业目标。  CMMI 有两种表示方法，和软件CMM 一样的阶段式表现方法，另一种是连续式的表现方法。这两种表现方法的区别是：阶段式表现方法仍然把CMMI 中的若干个过程区域分成了5 个成熟度级别，帮助实施CMMI 的组织建议一条比较容易实现的过程改进发展道路。而连续式表现方法则通过将CMMI 中过程区域分为四大类：过程管理、项目管理、工程以及支持。对于每个大类中的过程区域，又进一步分为基本的和高级的。这样，在按照连续式表示方法实施CMMI的时候，一个组织可以把项目管理或者其他某类的实践一直做到最好，而其他方面的过程区域可以完全不必考虑。  PSP 是SONY公司第一次投入掌上游戏机市场的第一件产品  TSP是Humphrey在SEI开发的一种软件团队开发过程，在针对个人的PSP与针对企业的CMMI之间架设了桥梁。 | 152 |
| pc8hga | 107、 请描述软件测试活动的生命周期 | 按照IEEE标准，测试的生命周期大致可分为计划、设计、实现、执行这四部分。  其中：计划：对整个测试周期中所有活动进行规划，估计工作量、风险，安排人力物力资源，安排进度等；  设计：完成测试方案，从技术层面上对测试进行规划；  实现：进行测试用例和测试规程设计；  执行：根据前期完成的计划、方案、用例、规程等文档，执行测试用例，记录测试结果，进行测试分析，完成测试报告 |  |