一、 你对测试最大的兴趣在哪里？为什么？  
最大的兴趣就是测试有难度，有挑战性！做测试越久越能感觉到做好测试有多难。曾经在无忧测试网上看到一篇文章，是关于如何做好一名测试工程师。一共罗列了11，12点，有部分是和人的性格有关，有部分需要后天的努力。但除了性格有关的1，2点我没有把握，其他点我都很有信心做好它。  
刚开始进入测试行业时，对测试的认识是从无忧测试网上了解到的一些资料，当时是冲着做测试需要很多技能才能做的好，虽然入门容易，但做好很难，比开发更难，虽然当时我很想做开发（学校专业课我基本上不缺席，因为我喜欢我的专业），但看到测试比开发更难更有挑战性，想做好测试的意志就更坚定了。  
我觉得做测试整个过程中有2点让我觉得很有难度（对我来说，有难度的东西我就非常感兴趣），第一是测试用例的设计，因为测试的精华就在测试用例的设计上了，要在版本出来之前，把用例写好，用什么测试方法写？（也就是测试计划或测试策略），如果你刚测试一个新任务时，你得花一定的时间去消化业务需求和技术基础，业务需求很好理解（多和产品经理和开发人员沟通就能达到目的），而技术基础可就没那么简单了，这需要你自觉的学习能力，比如说网站吧，最基本的技术知识你要知道网站内部是怎么运作的的，后台是怎么响应用户请求的？测试环境如何搭建？这些都需要最早的学好。至少在开始测试之前能做好基本的准备，可能会遇到什么难题？需求细节是不是没有确定好？这些问题都能在设计用例的时候发现。  
第二是发现BUG的时候了，这应该是测试人员最基本的任务了，一般按测试用例开始测试就能发现大部分的bug，还有一部分bug需要测试的过程中更了解所测版本的情况获得更多信息，补充测试用例，测试出bug。还有如何发现bug？这就需要在测试用例有效的情况下，通过细心和耐心去发现bug了，每个用例都有可能发现bug，每个地方都有可能出错，所以测试过程中思维要清晰（测试过程数据流及结果都得看仔细了，bug都在里面发现的）。如何描述bug也很有讲究，bug在什么情况下会产生，如果条件变化一点点，就不会有这个bug，以哪些最少的操作步骤就能重现这个bug，这个bug产生的规律是什么？如果你够厉害的话，可以帮开发人员初步定位问题。  
问题十五：你的测试职业发展目标是什么？  
测试经验越多，测试能力越高。所以我的职业发展是需要时间累积的，一步步向着高级测试工程师奔去。而且我也有初步的职业规划，前3年累积测试经验，按如何做好测试工程师的11，12点要求自己，不断的更新自己改正自己，做好测试任务。

二、您所熟悉的软件测试类型都有哪些？请试着分别比较这些不同的测试类型的区别与联系（如功能测试、性能测试……）  
测试类型有：功能测试，性能测试，界面测试。  
功能测试在测试工作中占的比例最大，功能测试也叫黑盒测试。是把测试对象看作一个黑盒子。利用黑盒测试法进行动态测试时，需要测试软件产品的功能，不需测试软件产品的内部结构和处理过程。采用黑盒技术设计测试用例的方法有：等价类划分、边界值分析、错误推测、因果图和综合策略。  
性能测试是通过自动化的测试工具模拟多种正常、峰值以及异常负载条件来对系统的各项性能指标进行测试。负载测试和压力测试都属于性能测试，两者可以结合进行。通过负载测试，确定在各种工作负载下系统的性能，目标是测试当负载逐渐增加时，系统各项性能指标的变化情况。压力测试是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服务级别的测试。  
界面测试，界面是软件与用户交互的最直接的层，界面的好坏决定用户对软件的第一印象。而且设计良好的界面能够引导用户自己完成相应的操作，起到向导的作用。同时界面如同人的面孔，具有吸引用户的直接优势。设计合理的界面能给用户带来轻松愉悦的感受和成功的感觉，相反由于界面设计的失败，让用户有挫败感，再实用强大的功能都可能在用户的畏惧与放弃中付诸东流。  
区别在于，功能测试关注产品的所有功能上，要考虑到每个细节功能，每个可能存在的功能问题。性能测试主要关注于产品整体的多用户并发下的稳定性和健壮性。界面测试更关注于用户体验上，用户使用该产品的时候是否易用，是否易懂，是否规范（快捷键之类的），是否美观（能否吸引用户的注意力），是否安全（尽量在前台避免用户无意输入无效的数据，当然考虑到体验性，不能太粗鲁的弹出警告）？做某个性能测试的时候，首先它可能是个功能点，首先要保证它的功能是没问题的，然后再考虑该功能点的性能测试

三、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：已知产品的功能设计规格，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：已知产品的内部工作过程，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。

软件的黑盒测试意味着测试要在软件的接口处进行。这种方法是把测试对象看做一个黑盒子，测试人员完全不考虑程序内部的逻辑结构和内部特性，只依据程序的需求规格说明书，检查程序的功能是否符合它的功能说明。因此黑盒测试又叫功能测试或数据驱动测试。黑盒测试主要是为了发现以下几类错误：  
1、是否有不正确或遗漏的功能？  
2、在接口上，输入是否能正确的接受？能否输出正确的结果？  
3、是否有数据结构错误或外部信息（例如数据文件）访问错误？  
4、性能上是否能够满足要求？  
5、是否有初始化或终止性错误？

软件的白盒测试是对软件的过程性细节做细致的检查。这种方法是把测试对象看做一个打开的盒子，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。通过在不同点检查程序状态，确定实际状态是否与预期的状态一致。因此白盒测试又称为结构测试或逻辑驱动测试。白盒测试主要是想对程序模块进行如下检查：  
1、对程序模块的所有独立的执行路径至少测试一遍。  
2、对所有的逻辑判定，取“真”与取“假”的两种情况都能至少测一遍。  
3、在循环的边界和运行的界限内执行循环体。  
4、测试内部数据结构的有效性，等等。

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。通常而言，一个单元测试是用于判断某个特定条件（或者场景）下某个特定函数的行为。

单元测试是由程序员自己来完成，最终受益的也是程序员自己。可以这么说，程序员有责任编写功能代码，同时也就有责任为自己的代码编写单元测试。执行单元测试，就是为了证明这段代码的行为和我们期望的一致。

集成测试（也叫组装测试，联合测试）是单元测试的逻辑扩展。它的最简单的形式是：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。从这一层意义上讲，组件是指多个单元的集成聚合。在现实方案中，许多单元组合成组件，而这些组件又聚合成程序的更大部分。方法是测试片段的组合，并最终扩展进程，将您的模块与其他组的模块一起测试。最后，将构成进程的所有模块一起测试。

系统测试是将经过测试的子系统装配成一个完整系统来测试。它是检验系统是否确实能提供系统方案说明书中指定功能的有效方法。（常见的联调测试）

系统测试的目的是对最终软件系统进行全面的测试，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。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。  
验收测试是向未来的用户表明系统能够像预定要求那样工作。经集成测试后，已经按照设计把所有的模块组装成一个完整的软件系统，接口错误也已经基本排除了，接着就应该进一步验证软件的有效性，这就是验收测试的任务，即软件的功能和性能如同用户所合理期待的那样。

四、当开发人员说不是BUG时，你如何应付？  
开发人员说不是bug，有2种情况，一是需求没有确定，所以我可以这么做，这个时候可以找来产品经理进行确认，需不需要改动，3方商量确定好后再看要不要改。二是这种情况不可能发生，所以不需要修改，这个时候，我可以先尽可能的说出是BUG的依据是什么？如果被用户发现或出了问题，会有什么不良结果？程序员可能会给你很多理由，你可以对他的解释进行反驳。如果还是不行，那我可以给这个问题提出来,跟开发经理和测试经理进行确认,如果要修改就改,如果不要修改就不改。其实有些真的不是bug，我也只是建议的方式写进TD中，如果开发人员不修改也没有大问题。如果确定是bug的话，一定要坚持自己的立场，让问题得到最后的确认。

五、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。

六、如果有机会转成开发人员，你会去做开发工作吗？  
如果公司确实需要我可以从事开发，但我还是喜欢做测试，我认为我更适合做测试。

七、软件测试分哪些阶段？各阶段的含义？  
分为单元测试、集成测试、确认测试、系统测试、验收测试。单元测试是最小单位的测试，测试独立模块；集成测试主要测试模块之间的接口是否正常，确认测试类似于冒烟测试通常在大规模系统测试之前验证版本主要功能是否实现，版本的稳定性是否可以进入系统测试，系统测试是全面测试验证系统是否满足用户需求包括功能、性能、兼容性等等。验收测试是用户参与的测试。

八、一份测试计划应该包括哪些内容？  
背景、项目简介、目的、测试范围、测试策略、人员分工、资源要求、进度计划、参考文档、常用术语、提交文档、风险分析。

九、针对于软件的行业背景，你如何理解软件的业务？  
1、阅读用户手册了解软件的功能和操作流程；  
2、看一些业务的专业书籍补充业务知识；  
3、如果有用户实际的数据，可以拿实际的数据进行参考；  
4、参考以前的用例和BUG报告；  
5、在使用软件的过程中多思考；  
6、多与产品经理交流。

十、测试用例应包括哪些内容？  
编号、模块名称、编写人、日期、操作说明、输入数据、预期结果等。