

软件测试

一、概念题（熟记和辨析常见术语）

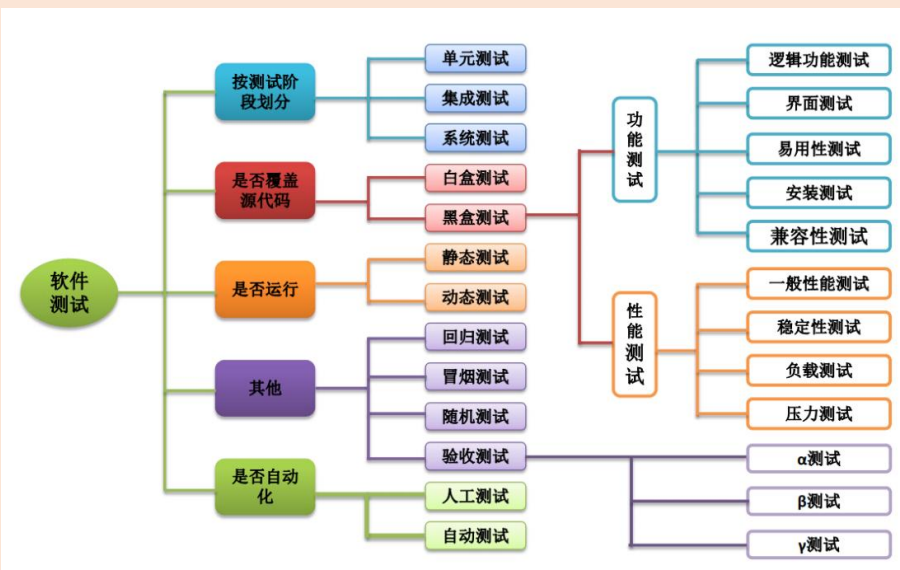
请你简述下你认为的软件测试？软件测试都有哪些方法？一般流程是什么。

在我看来，**软件测试**就是在**规定条件**下对**程序**进行操作，来发现程序潜在的**错误**，从而**保证软件质量的过程**。按照**开发阶段**来分，**软件测试**包括：**单元测试**、**集成测试**、**系统测试**和**验收测试**。按照**测试技术**来分，软件测试包括：**黑盒测试**、**白盒测试**和**灰盒测试**。

软件测试不是等软件开发完成才介入，而是**应当尽早地从需求阶段就介入软件开发过程**。所以软件测试的一般流程（W 模型）是：

- （1）**分析测试需求**：阅读策划需求，分析需求点
- （2）**准备测试计划**：参考软件设计说明书等编写测试计划，指定整体测试策略
- （3）**编写测试用例**：针对需求设计不同的**测试用例**，一个**测试用例**包含编号、日期、测试人员、优先级、标题、目标、环境、输入数据、预期结果等
- （4）**执行测试计划**：**单元测试->集成测试->系统测试->验收测试**
- （5）**输出测试报告**：输出测试整体结果，确认是否可以上线

请你简述下软件测试的分类。



请你简述下常见的测试模型。

测试模型是测试人员在测试过程中的经验汇总，是测试过程管理的重要参考依据。			
	特点	优点	缺点
<div>V模型 </div>	V 模型 是开发阶段完成后再进入测试阶段	包含了单元测试和系统测试，测试阶段分工明确	很多测试阶段发现的错误已经为时已晚，很可能无法再修改；返工量大，灵活性低

	V模型 是开发阶段和测试阶段并行进行	测试伴随软件的开发周期，可以发现早期缺陷，缺陷修复成本低	技术复杂度高，实践起来比较困难，不适合软件的反复迭代
	H模型 将测试活动完全独立出来，不和开发流程绑定在一起，可以根据自身需要提前准备各种测试工具	测试过程完全独立，和其他流程并发执行，具有很强的灵活性，也是可迭代的	管理难度大，对测试人员的技能要求高

请你简述下单元测试、集成测试、系统测试和验收测试。

单元测试是指对软件中的**最小可执行单元**进行测试。这个“单元”可以是一个**函数、类、功能模块**或者**子系统**，一般来说，单元越小越好，**单元测试**可以由**开发人员**也可以由**测试人员**来做，各有利弊。**单元测试和白盒测试紧密相关**，开发人员一般更了解自己开发的单元，写测试代码更容易，但开发人员不一定有意愿或者时间来单独做单元测试，而且很可能测试数据不够全面；测试人员一般有系统的测试思想，能够保证测试数据的覆盖率。

集成测试就是将经过**单元测试**后的**各个单元逐个集成后进行测试**，该测试的主要目的是检查不同单元之间的接口是否被正确使用等。在实际的开发过程中，**不同的单元模块很可能是不同的开发人员或者团队做的**，所以在将不同单元组合在一起时就需要进行全面测试，**集成测试和灰盒测试紧密相关**。

系统测试就是将**整个集成好的软件系统**进行**彻底地测试**。系统测试应该按照测试计划进行，其输入、输出应该要和需求文档进行对比。**系统测试和黑盒测试相关**，方法也很多，例如**功能测试（黑盒测试）、性能测试、随机测试**等。

验收测试也叫**交付测试**，是软件在部署前的最后一步测试。一般是**客户和测试人员**来共同测试，表明软件符合客户需求。

请你简述下什么是黑盒测试、白盒测试和灰盒测试？

黑盒测试就是**功能测试（或者数据驱动测试）**。简单点说，**黑盒测试**就是把**测试对象**看做一个封闭的黑盒子，测试人员完全不关心盒子内部的逻辑结构和特性，只需要**根据程序的需求文档**来依次检查，这个程序的功能是否都符合预定的要求。

白盒测试就是**结构测试（或者逻辑驱动测试）**。简单点说，**白盒测试**就是把**测试对象**看做一个打开的白盒子，测试人员根据盒子内部的逻辑结构和特性来设计**测试用例**，**检查程序在不同节点是否与预期一致**。

灰盒测试是介于**白盒测试**与**黑盒测试**之间的一种测试，**灰盒测试多用于集成测试阶段**，不仅关注输出、输入的正确性，同时也关注程序内部的情况。**灰盒测试**不像白盒测试那样详细、完整，但又比**黑盒测试**更关注程序的内部逻辑，常常是通过一些表征性的现象、事件、标志来判断内部的运行状态。

单元测试有哪些方法？你知道哪些单元测试框架？

单元测试的方法一般就是**人工静态检查**（自己读代码）和**动态执行跟踪**（运行测试脚本来测试）。为了更好的管理维护测试脚本，一般会采用**单元测试框架**来管理，不同的编程语言有不同的单元测试框架。例如：

- Java: JUnit、TestNG
- Python: unittest、pytest

单元测试的方法可以等价于**白盒测试**的方法。

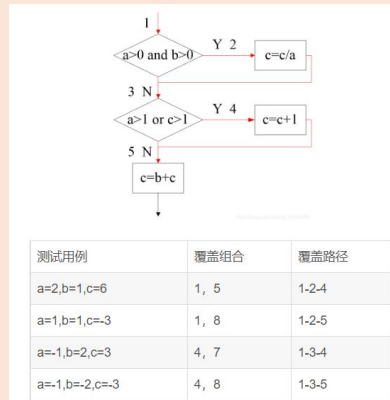
请你说说白盒测试有哪些方法？

白盒测试简单说就是考虑程序内部运行，通过分析程序内部的运行方式来设计各种输入情况。**白盒测试**方法可以分为两大类：

（1）**逻辑覆盖**：以程序的内部逻辑结构为基础，分为：**语句覆盖**（每行代码都尽可能至少被测试一次）、**判定覆盖**（每个 if 语句都至少测试一次）、**条件覆盖**（if 里面可能存在多个条件，尽可能让每个条件都至少满足 1 次）、

条件组合覆盖（if 的各种组合条件都尽可能测试一次）等

（2）**基本路径覆盖**：在程序控制流程的基础上，分析控制构造的环路复杂性，导出基本可执行路径集合，从而设计测试用例，比如一个程序中存在各种 if 分支，将其编号覆盖所有路径。



请你说说黑盒测试有哪些方法？

黑盒测试简单说就是不考虑程序内部运行，**只考虑输入输出**。输入各种情况，能否达到预期的输出。**黑盒测试**方法包括：

（1）**等价类划分**：将所有 的 输入划分为若干不相交的子集叫做等价类，然后从每个等价类中选取少量具有代表性的数据进行测试。

（2）**边界值分析**：在等价类划分时，往往要考虑划分的边界，尽可能地使用一些特殊边界值来测试。比如除法测试中使用 0 来做除数等，使用最大值做除数等。

测试用例	所属等价类	输入数据		预期输出	实际结果
		加数1	加数2	和	提示“请输入1~100间的整数”
1	1 有效等价类	1	1	2	
2	1 有效等价类	100	100	200	
3	2 无效等价类	0	0	提示“请输入1~100间的整数”	
4	3 无效等价类	101	101	提示“请输入1~100间的整数”	
5	4 无效等价类	1.2	3.2	提示“请输入1~100间的整数”	
6	5 无效等价类	A	B	提示“请输入1~100间的整数”	
7	6 无效等价类	@	#	提示“请输入1~100间的整数”	
8	7 无效等价类	空格	空格	提示“请输入1~100间的整数”	
9	8 无效等价类			提示“请输入1~100间的整数”	

（3）**判定表法**：判定表就是借助表格对**输入条件**进行**组合设计**，已到达尽可能覆盖的测试效果。

	ID	项目名称	R1	R2	R3	R4	R5
条件项	C1	此商品在经营范围	N	Y	Y	Y	Y
	C2	此商品可以发货		Y	Y	N	N
	C3	此客户没有拖欠过付款		Y	N	Y	N
动作项	A1	货到后允许客户转账		1			
	A2	货到客户必须立即付款			1		
	A3	重新组织货源				1	1
	A4	电话通知				1	
	A5	书面通知	1				1

（4）**因果图分析**：因果图是借助图形来分析各种**输入条件**的组合，**每种组合条件就是因**，**输出结果就是果**，因果图分析的结果就是产生一张判定表。

请说一下功能测试与非功能测试都包含哪些方法？

按照**测试的目标**，**软件测试**可以分为**功能测试**和**非功能测试**。**功能测试**主要是测试程序的**预期行为是否满足对应功能**，着重测试软件的输出符不符合预期；**非功能测试**是测试程序的**性能和稳定性**，即在各种极端条件下可能发生的情况。

功能测试分为：**单元测试**、**集成测试**、**系统测试**、**回归测试**等。

非功能测试分为：**性能测试**、**压力测试**、**安全性测试**、**可靠性测试**、**恢复性测试**、**兼容性测试**等。

你知道什么是回归测试吗？

回归测试是指在**程序发生修改**之后**重新测试先前的测试用例**以保证修改的正确性。理论上，**软件产生新版本**，**都需要进行回归测试**，验证以前发现和修复的错误是否在新软件版本上再次出现。根据修复好了的缺陷再重新进行测试。回归测试的目的在于验证以前出现过但已经修复好的缺陷不再重新出现。一般指对某已知修正的缺陷再次围绕它原来出现时的步骤重新测试。

二、开放题（无标准回答，只供参考）

请你说一下如何写测试用例

测试用例就是一个测试需求，是指导测试人员进行测试的依据，测试人员使用各种**白盒测试**或者**黑盒测试**的方法的最终目的，就是编写出尽可能全的**测试用例**，然后对照**测试用例**进行实际测试。一个**测试用例**一般写在**Excel 表中**，包含编号、日期、测试人员、优先级、标题、目标、环境、输入数据、预期结果等详细记录，需要有经验的测试人员来设计：

用例编号	用例标题	测试项目	用例级别	前置条件	测试输入	执行步骤	预期结果
英文+数字表示唯一性	验证某个功能的实现，和用例编号一一对应	该用例所属测试的范围，一般通过英文简称描述	priority: 优先级的范围，一般通过英文简称描述	执行的前提条件	测试过程中具体输入的数据，没有可以不写	验证操作步骤按照顺序地输入	希望得到的结果

编写出好的**测试用例**需要以下几个要求：

- 1、**从用户角度出发**。要将用户当做一个可能会犯各种错误的人，考虑各种可能的输入，尤其是异常输入。
- 2、**测试理论的支撑**。比如使用思维导图、流程图、因果图、判定表等工具辅助自己分析测试场景。
- 3、**测试经验的总结**。对于相同的需求要善于复盘总结经验，那么下次遇到相同的需求就可能复用上次的经验。

编写具体的测试用例可以根据不同的阶段使用不同的方法，例如在**单元测试阶段**可以使用常见的**白盒测试**方法；在**系统测试阶段**可以使用常见的**黑盒测试**方法。

你工作中使用到了哪些测试方法？简单说明一下。

- （1）**单元测试**方法。在平时的开发过程中 **Debug** 时基本也就使用了一些单元测试的思想。比如设计某一个线程池时，先将线程池的类封装好，然后写个简单代码测试这个类的 **API** 接口，测试结果没问题再放入整个项目中使用。
- （2）**压力测试**方法。在 **Web** 服务器项目开发中使用 **Web Bench** 工具测试服务器的并发性能的访问。在跳表项目中模仿 **Web Bench** 工具开发 **Stress Bench** 测试跳表的并发性能。

请你说一下，你印象中最深的一个 **Bug**，你是怎么定位这个 **Bug** 的，如何解决的？

我影响最深刻的 **BUG** 可能是在开发这个 **Web** 服务器过程中遇到的一个 **BUG**，这个 **BUG** 是我有一天新打开电脑，发现我编写到一半的项目刷新网页后连接不到服务器了。

我当时的第一反应是可能是我新编写的代码出了问题，所以导致无法连接到服务器，因为之前没写这些代码时页面还是正常的，所以我就进行了版本回退，将新添加的代码注释掉，然后测试发现依然连接不到。

然后我就继续注释掉我之前写的一些模块，保留下最开始的联网的套接字模块，测试后依然不行。我怀疑是电脑联网的问题，因为我服务器程序在一个远程主机上，套接字绑定远程主机的一个静态 IP，浏览器在我本地的电脑上，但是很快发现也不对，因为我的电脑可以访问到服务器，我可以通过 **ToDesk** 软件来访问远程主机。

然后我就测试了下我本机的浏览器，发现这个浏览器访问其他网站也出现了问题，这个时候我就知道问题应该出在本地的浏览器配置上。我试着重启浏览器、重置浏览器设置都不行，然后猜测是代理服务器 **DNS** 配置出了问题，我就试着关闭本机的代理服务器，然后关闭后发现它又很快重启了，我搜索了一些方法让代理服务器不重启后也还是不行，接着我按照一些博客的方法，通过命令行重置 **DNS** 缓存，设置注册表等也不行，这个 **BUG** 让我反反复复调试了快一个小时，而且不解决还不行，因为这个 **BUG** 导致我的浏览器访问不了任何网页。

我试着安装其他的浏览器发现连安装都安装不了，重装操作系统成本又太高，最后我猜测可能 **TCP** 配置除了问题，因为我连其他浏览器的安装包都没法成功运行，说明操作系统内部的 **TCP** 程序根本没法联网，然后我试着按照网络文档重新设置了 **Windows** 的 **TCP/IP** 设置中的本地 **DNS** 地址，然后发现成功了。

至于为什么会出现这个 **BUG**，后来我发现是因为我之前关电脑的时候没有提前关掉我的梯子（**cfw**，**Clash For Windows**），这个梯子可能修改了我的 **DNS** 地址开启重启后梯子又没有自动启动，导致我的 **TCP** 设置出现了问题。

从这件事中我发现的经验教训是：遇到 **BUG** 一定要思路清晰，不要烦躁，通过排除法一步步定位 **BUG** 的位置就能解决 **BUG**。就像网页无法显示这件事，问题可能出在服务器、浏览器、梯子、**DNS** 配置、网络连接多个地方，需要一步步手动排除才能解决。

三、场景题（无标准回答，只供参考）

请你根据微信的登录界面设计一些简单的测试用例/如何测试登录页面？

只有一个完成的登录页面，无法看到登录页面的代码逻辑时，那只能进行**黑盒测试**，具体可以分为**功能测试**和**非功能测试**两大类进行：

一、功能测试（列举仅供参考）

- 1、输入正确的用户名和密码，点击提交按钮，验证是否能正确登录，登录成功则需要记住用户名。（正常需求）
- 2、输入错误的用户名和密码，点击提交按钮，验证是否能正确登录，登录失败不需要记住用户名。（错误需求）
- 3、查看是否存在用户名和密码的限制，比如用户名的最大长度，限制字符等，密码等级的要求等，如果没满足这些限制是否会提醒并限制登录。
- 4、什么都不输入，点击提交按钮，检查提示信息。
- 5、检查能否选择不同登录方式进行登录，如使用手机号登录、使用微信号登录或扫码登录。
- 6、有验证码时，还要考虑文字是否扭曲过度导致辨认难度大，考虑颜色、刷新或换一个按钮是否好用
- 7.输入密码的时候，大写键盘开启的时候要有提示信息。

二、非功能测试（列举仅供参考）

1、界面测试

- （1）布局是否合理，文本框和按钮是否整齐
- （2）界面中的文字简洁易懂，没有错别字。
- （3）不同的屏幕分辨率下界面显示是否正常。

2、性能测试

- （1）检查登录成功后跳转到登录页面的响应时间是否在合理区间内。
- （2）模拟大量用户同时登陆，检查一定压力下能否正常登陆跳转。

3、兼容性测试

- （1）不同移动平台或 PC 环境下下能否显示正常且功能正常
- （2）多用户在同一机器上登录、同一用户能否在多台机器上登录是否正常。
- （3）不同语言环境（中文或英文）下，页面的显示是否正确。

4、安全性测试

- （1）通过抓包分析工具，检查用户名和密码是否通过加密的方式发送给 Web 服务器。
- （2）是否存在页面防脚本设计，用户名和密码的输入框，应该禁止通过脚本输入（防止 XSS 攻击和 SQL 注入攻击）。
- （3）用户名和密码的验证应该也用服务器端验证，而不能单单是在客户端用 javascript 验证。
- （4）防止暴力破解，检测是否限制错误登陆的次数。
- （5）登录成功后生成的 Cookie，是否能被脚本盗取是 httponly (否则容易被脚本盗取)。

如果对一个杯子进行测试，你要怎么做？

无论测试什么对象，测试成品时基本就只能进行黑盒测试，黑盒测试就要从功能测试和非功能测试上思考。

一、功能测试（列举仅供参考）

- （1）装水时是否会漏水，装少许水，装一半水，装满水的情况
- （2）装满水会有多重，是否超过人手承受范围
- （3）水杯有刻度的话，刻度是否符合标准
- （4）杯子是否需要携带方便，如果需要，可以放进多大的包里

二、非功能测试（列举仅供参考）

1、界面测试

- （1）外观是否达到标准，图案是否清晰完整、美观
- （2）材质表面是否细腻不刮手，放在手上是否舒服

2、压力测试

- （1）杯子材质能够承受的最大重力
- （2）杯子可容纳的最高和最低水温
- （3）杯子长时间放置是否漏水、保温能力如何

3、安全性测试

- (1) 杯子从多高的地方下坠会被摔碎
- (2) 材质在高温下是否有污染，是否易繁殖细菌等
- (3) 杯子长时间放置是否漏水、保温能力如何
- (4) 杯子是否可以使用微波炉加热，是否可以容纳酒精、碳酸饮料、果汁等
- (5) 杯子是否存在防滑设计

请你说一下 Web 测试、APP 测试和小程序测试的区别？

- 1、关注的对象不同。**Web 是 B/S（浏览器-服务器）架构**，通过浏览器访问服务器，Web 前端开发的各种页面也是存放在服务器上，所以只需要维护服务器资源即可；**App 是 C/S 架构**，通过客户端访问服务器，除了关注服务端，还需要关注客户端，所以修改了服务端时，很可能也要对客户端用户的所有核心版本进行回归测试。**微信小程序也是 C/S 架构**，但是客户端是依赖于微信环境的程序。
- 2、关注的性能不同。**Web 页面和微信小程序一般只关注页面的响应速度**。APP 的性能指标有**内存占用、网络流量、电量功耗、启动速度、界面切换速度、响应速度**等。
- 3、兼容性不同。**Web 页面**主要考虑**对不同主流浏览器的兼容**，APP 则需要考虑对不同厂商的**机型、操作系统的兼容**。**小程序**主要是指**微信小程序**，只要关注是否**兼容不同的微信版本**即可。