

一、	测试总体	7	
1.	什么是软件测试?	7	
2.	软件测试的目的?	7	
3.	软件测试的目标	7	
4.	软件测试的原则	7	
5.	测试的职责	7	
6.	软件都有多少种分类?	7	
7.	软件的分类型	8	
8.	测试的主要方面	8	
9.	软件测试的对象	8	
10.	什么是“测试案例”?	8	
11.	怎么编写案例?	8	
12.	软件测试的两种方法	8	
13.	测试结束的标准是什么?	8	
14.	软件的生命周期	8	
15.	软件测试按过程分为三个步骤	9	
16.	面向对象的设计如何影响测试?	9	
17.	软件带来错误的原因很多。主要的原因有哪些?	9	
18.	软件测试的步骤是什么?	9	
19.	如何录制测试脚本?	9	
20.	完全测试是不可能的，必须对测试的各项进行等价划分。	9	
21.	应该考虑进行如何测试的测试方法	10	
22.	怎样估计测试工作量?	10	
23.	测试设计的问题	11	
24.	当测试过程发生错误时，有哪几种解决办法?	11	
25.	测试执行的问题	11	
26.	测试评估的目标	11	
27.	这些软件常识的理解和运用将有助于我们在进行软件测试时能够更好的把握软件测试的尺度。	11	11
28.	如何提高测试?	12	
29.	C/S 模式的优点和缺点	12	
30.	B/S 模式的优点和缺点	12	
31.	对面向过程的系统蔡依林个的集成策略有哪些?	12	
32.	测试结束的标准是什么?	12	
33.	怎么才能够全面的测试到每一个点?	13	
34.	开发与测试的关系	13	
35.	项目的入口标准	13	
36.	测试过程	13	
37.	测试出口准则	13	
38.	测试完成标准	13	
39.	测试活动中统计了哪些数据?	13	
40.	如何选择用户测试的工作产品?	13	
41.	测试环境描述在哪儿?	13	
42.	进行测试时产生了哪些文档或记录?	13	
43.	测试人员需要何时参加需求分析?	13	
44.	产品测试完以后由谁来发布?	14	
45.	软件测试与调试的关系	14	
46.	质量的八大特性是什么? 各种特性的定义?	14	

47.	什么是软件的“质量”？	14
48.	软件质量应该从哪些方面来评价？	14
49.	什么是“软件质量保障”？	14
50.	为什么软件会有毛病？	14
51.	什么是验证、评价、预排、检查？	14
52.	什么是 UML？	15
53.	什么是 CMM？	15
54.	对面向对象的系统采用的集成策略有？	15
55.	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	15
56.	比较负载测试、压力测试，容量测试和强度测试区别	15
57.	测试执行过程的三个阶段	15
58.	测试的任务和发展目标——质量	15
59.	测试的组织形式——小组	15
60.	测试的运行方式——制度化并形成应用	16
61.	传统测试流程遇到的挑战和对策——问题发现得越早，解决的代价就越小	16
62.	软件生产的三个最重要的因素是：质量、进度和费用	16
二、	需求分析	16
1.	需求人员需要何时参加需求分析？	16
2.	如果需求一直在变化怎么办？	16
三、	测试计划	16
1.	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？	16
2.	制定测试计划之前需要了解什么问题？	17
3.	测试计划都包括哪些项？	17
4.	怎样做好测试计划？	17
5.	测试资源	17
6.	风险和问题	17
四、	测试策略	18
1.	什么是“测试策略”？	18
2.	测试策略包括哪些？	18
五、	测试用例	18
1.	测试用例制定的原则？	18
2.	测试用例是否纳入测试基线管理？测试用例发生变更的流程？测试用例如何进行标识？	18
3.	什么时候编写测试用例？依据是什么？如何保证测试用例与需求的一致性？需要同行评审吗？	18
4.	测试用例如何设计的？	18
5.	测试用例包括哪些项？	错误！未定义书签。
6.	如何保证用例覆盖到罕见缺陷？	19
7.	什么时候编写测试用例？依据是什么？如何保证测试用例与需求的一致性？需要同行评审吗？	19
六、	缺陷报告	19
1.	缺陷报告的优先级别	19
2.	简单概述缺陷报告	19
3.	缺陷报告包括哪些项？	19
七、	测试总结报告	19
1.	测试总结报告包括哪些项？	19
八、	缺陷 (Bug)	19
1.	软件缺陷的原则	19
2.	软件缺陷的特征。	19
3.	什么是 Bug？	19

4. 缺陷处理流程.....	20
5. 缺陷的等级划分.....	20
6. 开发人员修复缺陷后，如何保证不影响其他功能？	20
7. 状态为已修改的缺陷，实际没有修改怎么办？	20
8. 生产软件的最终目的是为了客户需求，我们以客户需求作为评判软件质量的标准，认为软件缺陷（Software Bug）的具体含义包括下面几个因素	20
九、 白盒测试.....	20
1. 白盒测试有哪几种方法？	20
十、 单元测试.....	21
1. 单元测试主要是做一下几个事情：	21
十一、 集成测试.....	21
1.简述集成测试的过程	21
2.集成测试进入的准则？退出的准则？	21
3.集成测试通常都有那些策略？	21
十二、 系统测试.....	21
1. 系统的性能.....	21
2. 系统的开发.....	21
3. 系统的升级维护.....	22
4. 系统测试的策略有：	22
5.设计系统测试计划需要参考的项目文档有：	22
6.系统测试计划是否需要同行审批，为什么.....	22
7.Alpha 测试与 beta 的区别.....	22
8.系统测试阶段低级缺陷较多 怎么办？	22
10.系统测试的进入和退出准则？	22
11.系统测试阶段低级缺陷较多 怎么办？	22
12.系统测试包含哪些方面？	22
十三、 验收测试.....	22
1. 软件验收测试包括：	22
2.验收测试：	23
十四、 功能测试.....	23
1. 什么是功能测试？	23
2. 请问功能测试和性能测试的区别是什么？(只总结了两个方面，有其他的自己补充) 23	
十五、 兼容性测试.....	23
1. 兼容性测试.....	23
十六、 易用性测试（可用性测试）	23
1. 什么是易用性测试？	23
十七、 文档测试.....	23
1.文档测试.....	23
2. 怎么做好文档测试	23
3.文档测试要注意什么？	23
十八、 可靠性测试.....	23
1. 什么是可靠性测试？	23
十九、 安全测试.....	23
1. 什么是安全测试？	23
二十、 自动化测试.....	24
1. 什么时候适用自动化测试？	24
2. 不宜使用自动化的情况	24

二十一、	性能测试.....	24
1.	您在从事性能测试工作时,是否使用过一些测试工具?如果有,请试述该工具的工作原理,并以一个具体的工作中的例子描述该工具是如何在实际工作中应用的。.....	24
2.	您认为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?.....	24
3.	性能测试.....	24
4.	性能测试什么时候开始最合适.....	24
5.	并发性能测试的目的主要体现在三个方面?.....	24
二十二、	软件开发.....	24
1.	测试在开发阶段的作用.....	24
2.	完整的开发流程.....	25
3.	测试与开发各阶段的关系.....	25
4.	在软件开发过程中5个常见的问题是什么?.....	25
5.	针对软件开发过程中的问题,有5个解决办法:.....	25
6.	测试开发.....	25
7.	测试开发的问题:.....	26
二十三、	测试案例.....	26
1.	给你一个网站,你应该如何测试?.....	26
	(一个网站基本完工后,需要通过下面三步测试才可以交活).....	26
2.	游戏测试与软件测试的区别.....	26
二十四、	测试工具.....	27
1.	Quicktestprofessional.....	27
2.	TestDirect.....	27
3.	Loadrunner.....	27
二十五、	面试文本.....	28
1.	好的测试工程师应具备的素质?.....	28
2.	软件测试给你带来什么样的快乐?.....	28
3.	为什么要在一个团队中开展测试工作?.....	28
4.	你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?.....	28
5.	你们以前测试的流程是怎样的?.....	28
6.	为什么选择测试这行?.....	28
7.	如果时间不够,无法进行充分的测试怎么办?.....	28
8.	你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?.....	29
9.	你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.....).....	29
10.	你自认为测试的优势在哪里?.....	29
11.	你在测试中发现了一个bug,但是开发经理认为这不是一个bug。你应该怎么做?.....	29
12.	你是如何制定时间进度表的?.....	29
13.	介绍一下整体项目流程.....	29
14.	你是如何制定测试过程中的时间进度表的?.....	29
15.	测试工作进行到一半时,发现时间不够,你是如何处理的?.....	29
16.	怎样保证你所负责的模块通过了测试?.....	30
17.	软件测试人员和测试组长的职责分工.....	30
18.	如果你是测试组长你是如何对项目及组员进行管理的?.....	30
19.	什么时候开始搭建测试环境?由谁搭建?如何进行产品的集成?.....	30
20.	你所做的项目中采用了哪些测试方法?进行回归测试吗?.....	30
21.	上级如何检查你的工作?.....	30

22.	QA 是如何检查你的工作的?	30
23.	你参加了哪些培训?	30
24.	在你所做的项目中有需要测试的项目过程吗? 有, 请介绍。	30
25.	怎样保障你所负责的模块通过了测试?	30
26.	你是如何了解到你说项目中的成员?	31
27.	是否成立了独立的测试组? 测试人员在项目中测试的职责?	31
28.	测试结果分析如何? 如何产生和被记录?	31
29.	你认为软件测试过程中较常见的困难是什么? 如何有效克服这些困难? (根据自己实际测试中遇到的情况来写的)	31
30.	在实际项目中你是如何做测试计划?	31
31.	你什么时候开始制定测试计划?是否发生过变更? 如何进行变更?	31
32.	你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。	31
33.	你认为做好测试用例设计工作的关键是什么?	32
34.	在你以往的工作中,一条软件缺陷(或者叫 Bug)记录都包括哪些内容? 如何提交高质量的软件缺陷(Bug)记录?	32
35.	你在五年内的个人目标和职业目标分别是什么?	32
36.	你怎样做出自己的职业选择?	32
37.	是否了解公司的组织过程财富库?	32
二十六、	网络题库	32
1.	避免死锁的方法	32
2.	内存有哪几种存储组织结构, 请分别加以说明。	32
3.	操作系统中的同步和互斥解决了什么问题?	33
4.	什么是数据的独立性? 有几个层次?	33
5.	网状、层次数据模型与关系数据模型的最大区别是什么?	33
6.	简述 ODBC 的设置过程	33
7.	局域网对等网的含义及一般配置连接方法	33
8.	两台计算机连接的方法有哪些?	33
9.	介绍 plc 或板卡的大致功能和其在行业中的一般应用场合。	33
10.	组态软件	33
11.	写出下列端口号: ftp、dns、http、pop3、smtp、telnet、windows 终端服务	33
12.	写出 ISO 模型, tcp/udp 在那层? 他们有什么区别	34
13.	写出 HUB、tch、Router 是 ISO 的第几层设备, 叫什么?	34
14.	在防火墙中有哪些有效的设置方法	34
15.	ms-dos6.22 支持的分区最大为多少	34
16.	ms-dos6.22 能管理的硬盘容量最大为多大	34
17.	一台电脑由哪些部分组成,最小化硬件系统指的是需要哪些硬件	34
18.	目前一台普通的 pc(主板)最多能接几个硬盘	34
19.	请说出下列 dos 命令的作用	34
20.	在 windows98 启动前按 f5 或 f8 分别有什么作用	34
21.	你知道的微软的操作系统有哪些,你使用的是哪个,能熟练使用哪个	35
22.	为什么有些分区在 windows98 中看不到而在 windows xp 中却可以看到	35
23.	请简述 ip 地址,网关,子网掩码的含义	35
24.	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?	35
25.	中间件的含义, 意义, 主要产品名称?	35
26.	简述一下 c/s 模式或者 b/s 模式	35
27.	一台计算机的 IP 是 192.168.10.71 子网掩码 255.255.255.64 与 192.168.10.201 是同一局域网吗?	35

28.	internet 中 e-mail 协议,IE 的协议,NAT 是什么,有什么好处,能带来什么问题?DNS 是什么,它是如何工作的?	35
29.	PROXY 是如何工作的?	35
30.	win2k 系统内 AT 命令完成什么功能,Messenger 服务是做什么,怎么使用?	35
31.	进程,线程的定义及区别	36
32.	32 位操作系统内,1 进程地址空间多大,进程空间与物理内存有什么关系?	36
33.	网络攻击常用的手段,防火墙如何保证安全	36
34.	如何配静态 IP,如何测网络内 2 台计算机通不通,PING 一次返几个数据包?	36
35.	WIN9X 与 WINNT 以上操作系统有"服务"吗,服务是什么,如何停止服务?	36
36.	AD 在 WIN2KSERVER 上建需什么文件格式,AD 是什么?XP 多用户下"注销"与"切换"的区别.	36
37.	UDP 可以跨网段发送吗?	36
38.	最简单的确认远程计算机(win2K 以上)某个监听端口是正常建立的?	36
39.	winrunner,loadrunner 是什么,区别	36
40.	磁盘分区如何分类,请举例说明安装操作系统的注意事项	36
41.	在网络应用测试中能够,网络延迟是一个重要指标。以下关于网络延迟的理解,正确的是 (D)	37
42.	模块的耦合度描述了 (D)	37
43.	耦合度,内聚	37
44.	网络七层协议具体描述	37
45.	路由器,交换机的功能是什么,子网掩码如何分配	37
46.	winXP 下局域网不通,怎么办	38
47.	对 RUP.CMM,CMMI,XP,PSP.TSP 的认识	38
48.	Ghost 的主要用途和常用方法	38
49.	写出 Weblogic 控制台访问地址(端口缺省)	38
	格式如下:	38
50.	手工修改 Tomcat 端口,在那个文件里?	38
51.	WinRunner 分几种录制模式,有什么特点?	38
52.	WinRunner 运行测试脚本时,有哪些模式?在什么情况下使用	38
53.	怎么把 C:\test 目录配制成 FTP 目录,供远程端使用	38
54.	怎么配置 Weblogic 域?	38
55.	文件格式系统有哪几种类型?分别说说 win95、win98、winMe、w2k、winNT、winXP 分别支持那些文件系统。	38
56.	我现在有个程序,发现在 WIN98 上运行得很慢,怎么判别是程序存在问题还是软硬件系统存在问题?	39
57.	FAT16/FAT32/NTFS 哪个的安全性最好,为什么?	39
58.	Internet 采用哪种网络协议?该协议的主要层次结构?	39
59.	请问交换机和路由器各自的实现原理是什么?分别在哪个层次上面实现的?	39
60.	请你分别画出 OSI 的七层网络结构图和 TCP/IP 的四层结构图。	39
61.	请你详细地解释一下 IP 协议的定义,在哪个层上面?主要有什么作用?TCP 与 UDP 呢?	39
62.	问:在 windows 环境下,遇到用杀毒软件都无法杀的病毒除了重作系统应该怎么办,如何关闭后台进程,msn 上不了都可能是什么问题,怎样解决。	39
63.	Pop3 与 IMAP 有什么区别?	39
64.	PIN、PIN2、PUK 是什么	40
65.	MVC 三层架构:	40
二十七、	暂时不需的测试题目	40
1.	代码会审是什么?	40
2.	你参与了哪些方面的同行评审?同行评审流程?	40
3.	分析工作组和域的区别	40

一、 测试总体

1. 什么是软件测试?

答：为了发现程序中的错误而执行程序的过程

2. 软件测试的目的?

答：首先，测试并不仅仅是为了要找出错误。通过分析错误产生的原因和错误的分布特征，可以帮助项目管理者发现当前所采用的软件过程的缺陷，以便改进。同时，这种分析也能帮助我们设计出有针对性地检测方法，改善测试的有效性。

其次，没有发现错误的测试也是有价值的，完整的测试是评定测试质量的一种方法。详细而严谨的可靠性增长模型可以证明这一点。

测试的目的是按照用户所需软件的质量，检查开发软件过程出现的 bug，使得开发人员及时修改，可以避免在开发结束的时候发现软件存在质量问题，避免公司不必要的损失。赢得用户对公司产品的认可。

测试的目的是以最少人力、物力和时间找出软件中潜在各种错误和缺陷，通过修正种错误和缺陷提高软件质量，回避软件发布后由于潜在的软件缺陷和错误造成的隐患带来的商业风险。

测试的附带收获是，它能够证明软件的功能和性能与需求说明相符合。

实施测试收集到的测试结果数据为可靠性分析提供了依据。

测试不能表明软件中不存在错误，它只能说明软件中存在错误。

3. 软件测试的目标

答：发现尽可能多的错误

测试是一个为了寻找错误而运行程序的过程。

一个好的测试用例是指很可能找到迄今为止尚未发现的错误的用例。

一个成功的测试是指揭示了迄今为止尚未发现的错误的测试。

4. 软件测试的原则

1) 应当把“尽早地和不断地进行软件测试”作为软件开发者的座右铭。

2) 测试用例应由测试输入数据和对应的预期输出结果这两部分组成。

3) 程序员应避免检查自己的程序。

4) 在设计测试用例时，应包括合理的输入条件和不合理的输入条件。

5) 软件测试的原则

6) 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。

7) 严格执行测试计划，排除测试的随意性。

8) 应当对每一个测试结果做全面检查。

9) 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。

5. 测试的职责

测试经理：

1、制定测试计划。

2、确保测试过程正常进行。

测试工程师

1、编写测试用例

2、搭建测试环境

3、执行测试

6. 软件都有多少种分类?

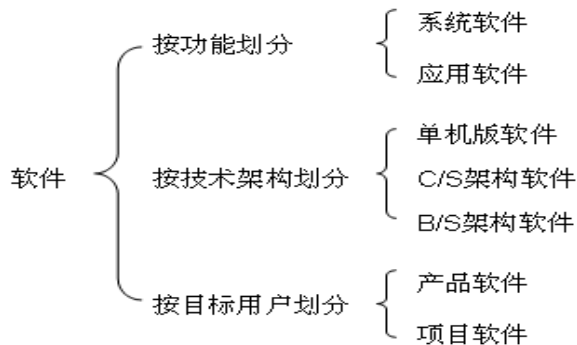
答：根据功能的不同，电脑软件可以粗略地分成四个层次：

最贴近电脑硬件的是一些小巧的软件。它们实现一些最基本的功能，通常“固化”在只读存储器芯片中，因此称为固件。

系统软件包括操作系统和编译器软件等。系统软件和硬件一起提供一个“平台”。它们管理和优化电脑硬件资源的使用。

支持软件。包括图形用户界面、软件开发工具、软件评测工具、数据库管理系统、中间件等。
应用软件种类最多，包括办公软件、电子商务软件、通信软件、行业软件、游戏软件等等。

7. 软件的分类



8. 测试的主要方面

答：A、功能测试：a、链接测试 b、表单测试 c、Cookies 测试 d、设计语言测试 e、数据库测试

B、性能测试：a、连接速度测试 b、负载测试 c、压力测试

C、接口测试：a、服务器接口 b、外部接口 c、错误处理

D、可用性测试：a、导航测试 b、图形测试 c、内容测试 d、整体界面测试

E、兼容性测试：a、平台测试 b、浏览器测试 c、视频测试 d、Modem/连接速率测试 f、打印机测试 g、组合测试

F、安全测试：a、目录设置 b、登录 c、Session d、日志文件 e、加密 f、安全漏洞

G、代码合法性测试：a、程序代码合法性检查 b、显示代码合法性检查

H、文档测试：

9. 软件测试的对象

答：软件测试并不等于程序测试。软件测试应贯穿于软件定义与开发的整个期间。

需求分析、概要设计、详细设计以及程序编码等各阶段所得到的文档，包括需求规格说明、概要设计规格说明、详细设计规格说明以及源程序，都应成为软件测试的对象

10. 什么是“测试案例”？

答：测试案例是一份文档，它描述了一个输入、反应、或者是与其相应的预期的响应，以便来判断应用软件的工作是否正常。测试案例应当包括测试标识、测试案例的名称、目标、测试条件/设置、输入数据要求、步骤、以及预期的结果。

注：开发一个应用软件的测试案例的过程，需要全面、深入地考虑该软件的操作，所以有助于发现在其需求或设计里面的问题。因此，如果有可能，在开发周期中应当尽早准备测试案例。

11. 怎么编写案例？

答：案例的编写与测试阶段的定义有很大的关系。系统测试和 unit 测试的案例可能不同。总体而言测试案例根据系统的需求而定。

12. 软件测试的两种方法

答：黑盒测试和白盒测试

黑盒：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。

黑盒测试又叫做功能测试或数据驱动测试。

白盒：此方法把测试对象看做一个透明的盒子，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的状态，确定实际的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。

13. 测试结束的标准是什么？

答：1.用例全部执行。2.覆盖率达到标准。3.缺陷率达到标准。4.其他指标达到质量标准

14. 软件的生命周期

答：软件生命周期是指一个计算机软件从功能确定、设计，到开发成功投入使用，并在使用中不断地修改、增

补和完善，直到停止该软件的使用的全过程（从酝酿到废弃的过程）

什么是软件的生命周期？

生命周期从收到应用软件开始算起，到该软件不再使用为止。它有如下各方面的内容：初始构思、需求分析、功能设计、内部设计、文档计划、测试计划、文档准备、集成、测试、维护、升级、再测试、逐步淘汰 (phase-out)、等等。

15. 软件测试按过程分为三个步骤

答：单元测试：单元测试又称模块测试，是针对软件设计的最小单位——程序模块，进行正确性检验的测试工作。其目的在于发现各模块内部可能存在的各种差错。

单元测试需要从程序的内部结构出发设计测试用例。多个模块可以平行地独立进行单元测试。

集成测试：在运行（可能是不完整）的应用中保证软件单元被结合后能正常操作的测试执行的阶段

系统测试：当应用作为整体运行时的测试执行阶段

16. 面向对象的设计如何影响测试？

答：好的面向对象的工程设计使得从代码追溯内部设计、再到功能测试，最后追溯到需求，成为一件容易的事。因为它对黑盒测试的影响很少（不需要了解应用软件的内部设计），而白盒测试只需针对该应用软件的对象。如果该应用软件设计得好，就可简化测试设计。

17. 软件带来错误的原因很多。主要的原因有哪些？

- 1) 交流不够、交流上有误解或者根本不进行交流
- 2) 软件复杂性
- 3) 程序设计错误
- 4) 需求变化
- 5) 时间压力
- 6) 代码文档贫乏
- 7) 软件开发工具

18. 软件测试的步骤是什么？

- 1) 测试过程按 4 个步骤进行，即单元测试 (Unit Testing)、集成测试 (Integrated Testing)、确认测试 (Validation Testing) 和系统测试 (System Testing) 及发版测试。
- 2) 开始是单元测试，集中对用源代码实现的每一个程序单元进行测试，检查各个程序模块是否正确地实现了规定的功能。
- 3) 集成测试把已测试过的模块组装起来，主要对与设计相关的软件体系结构的构造进行测试。
- 4) 确认测试则是要检查已实现的软件是否满足了需求规格说明中确定的各种需求，以及软件配置是否完全、正确。

19. 如何录制测试脚本？

- 1) 新建一个脚本 (Web/HTML 协议)
- 2) 点击录制按钮，在弹出的对话框的 URL 中输入 "about:blank"。
- 3) 在打开的浏览器中进行正常操作流程后，结束录制。
- 4) 调试脚本并保存。可能要注意到字符集的关联。
- 5) 设置测试场景
- 6) 针对性能设置测试场景，主要判断在正常情况下，系统的平均事务响应时间是否达标
- 7) 针对压力负载设置测试场景，主要判断在长时间处于满负荷或者超出系统承载能力的条件下，系统是否会崩溃

20. 完全测试是不可能的，必须对测试的各项进行等价划分。

1. 概念

等价分配：软件有无限的测试案例，我们要想办法把软件的相似输入、输出、操作分成一组，来使无限的测试案例减小到同样有效的小范围，这个过程称为等价分配。

边界条件：软件计划的操作界限所在的边缘条件，即如果超出这个边界条件，就可能会引出错误。

2 原因

输入量太大

输出结果太多

软件实现途径太多

软件说明书没有客观标准。从不同的角度看，软件缺陷的标准不同。

3 方法

(1) 数据测试:

1) 确定输入的边界条件，对边界线上的及边界线两边的数据进行测试；

2) 边界线可能是 2 的乘方，默认值、空白值、零值等；每一个软件测试问题各不相同，可能包含格式各异边界的不同数据。

(2) 状态测试（软件的状态是指软件当前所处的情况或者模式）

1) 每种状态至少访问一次；

2) 测试看起来最常见最普遍的状态转换；

3) 测试状态之间最不常用的分支；

4) 测试所有错误状态及其返回值；

5) 测试随机状态转换。

21. 应该考虑进行如何测试的测试方法

黑盒测试 (Black box testing) —— 不考虑内部设计和代码，根据需求和功能进行测试。

白盒测试 (White box testing) —— 根据应用软件的代码的内部逻辑，按照代码的语句、分支、路径和条件进行测试。

功能测试 (functional testing) —— 对一个应用软件的功能模块进行黑盒测试。这种测试应当由测试人员进行。但这并不意味着程序员在推出软件之前不进行代码检查。（这一原则适用于所有的测试阶段。）

系统测试 —— 针对全部需求说明进行黑盒测试，包括系统中所有的部件。

回归测试 (regression testing) —— 每当软件经过了整理、修改、或者其环境发生变化，都重复进行测试。很难说需要进行多少次回归测试，特别是到了开发周期的最后阶段。进行此种测试，特别适于使用自动测试工具。

负荷试验 (load testing) —— 在大负荷条件下对应用软件进行测试。例如测试一个网站在不同负荷情况下的状况，以确定在什么情况下系统响应速度下降或是出现故障。

压力测试 (stress testing) —— 经常可以与“负荷测试”或“性能测试”相互代替。这种测试是用来检查系统在下列条件下的情况：在非正常的巨大负荷下、某些动作和输入大量重复、输入大数、对数据库进行非常复杂的查询，等等。

性能测试 (performance testing) —— 经常可以与“压力测试”或“负荷测试”相互代替。理想的“性能测试”（也包括其他任何类型的测试）都应在质量保障和测试计划的文档中予以规定。

可用性测试 (usability testing) —— 是专为“对用户友好”的特性进行测试。这是一种主观的感觉，取决于最终用户或顾客。可以进行用户会见、检查、对用户会议录像、或者使用其他技术。程序员和测试人员通常不参加可用性测试。

安装/卸载测试 (install/uninstall testing) —— 对安装/卸载进行测试（包括全部、部分、升级操作）。

安全测试 (security testing) —— 测试系统在应付非授权的内部/外部访问、故意的损坏时的防护情况。这需要精密复杂的测试技术。

兼容性测试 (compatibility testing) —— 测试在特殊的硬件/软件/操作系统/网络环境下的软件表现。

α 测试 (alpha testing) —— 在开发一个应用软件即将完成时所进行的测试。此时还允许有较小的设计修改。通常由最终用户或其他人进行这种测试，而不是由程序员和测试人员进行。

β 测试 (beta testing) —— 当开发和测试已基本完成，需要在正式发行之前后最后寻找毛病而进行的测试。通常由最终用户或其他人进行这种测试，而不是由程序员和测试人员进行。

22. 怎样估计测试工作量？

效率假设：即测试队伍的工作效率。对于功能测试，这主要依赖于应用的复杂度，窗口的个数，每个窗口中的动作数目。对容量测试，主要依赖于建立测试所需数据的工作量大小。

测试假设：为了验证一个测试需求所需测试动作数目。

应用的维数：应用的复杂度指标。例如要加入一个记录，测试需求的维数就是这个记录中域的数目。

所处测试周期的阶段：有些阶段主要工作都在设计，有些阶段主要是测试执行。

23. 测试设计的问题

- 1) 不做测试设计，测试过程也是胡乱建立的。
- 2) 测试设计不详细，不是基于可量度的测试策略，例如测试计划覆盖一个集合或者测试需求的一个子集。
- 3) 测试过程没有采用最好的技术来检验 Windows C/S 结构的测试需求
- 4) 测试用例的选择规则
- 5) 选择与测试需求的实质部分最相关的测试用例。
- 6) 选择的测试用例应该不容易应用程序的改变的影响。

24. 当测试过程发生错误时，有哪几种解决办法？

- 1) 跳转到别的测试过程
- 2) 调用一个能够清除错误的过程
- 3) 退出过程，启动另一个
- 4) 退出过程和应用程序，重新启动 Windows，在失败的地方重新开始测试

25. 测试执行的问题

测试执行的问题

- 1) 自动化测试没有有效的利用，使得手工测试太多。
- 2) 测试结果的捕获没有系统性，而且没有查看或调查
- 3) 缺陷报告必须用手工加入缺陷跟踪系统

错误分类

1、测试用例失败

正常错误

2、脚本命令失败

当测试过程不能执行录制过程中的某个功能时，回产生这种错误，如鼠标单击按钮或选择菜单项等。它也能指示是缺陷还是测试过程的设计问题。

3、致命错误

导致测试停止，这种情况最好重起 Windows。

具体步骤：

- 1) 建立测试系统
- 2) 准备测试过程
- 3) 运行初始化过程
- 4) 执行测试
- 5) 从终止的测试恢复
- 6) 验证预期结果
- 7) 调查突发结果
- 8) 记录缺陷日记

26. 测试评估的目标

- 1) 量化测试进程
- 2) 生成缺陷和测试覆盖率的总结报告
 1. 测试评估的问题
- 3) 没有把测试覆盖率作为报告测试进程的根据，使得不知测试是否结束；
- 4) 没有做缺陷评估，缺陷评估是量度软件可行性的重要指标；
- 5) 不使用专门的软件工具进行数据输入任务和相应的评估活动，使得这些任务变得繁重累人。

测试覆盖率——评估测试完成多少的标准

27. 这些软件常识的理解和运用将有助于我们在进行软件测试时能够更好的把握软件测试的尺度。

- 1) 测试是不完全的（测试不完全）
- 2) 测试具有免疫性（软件缺陷免疫性）
- 3) 测试是 “ 泛型概念 ” （全程测试）
- 4) 80-20 原则

- 5) 为效益而测试
- 6) 缺陷的必然性
- 7) 软件测试必须有预期结果
- 8) 软件测试的意义 - 事后分析

28. 如何提高测试?

提高测试需要从几个方面着手，其实只是自己的一些感觉，不一定就需要按部就班，需要找自己适合的点。

制定完备的测试计划

清楚的认识测试计划，测试计划是一个文档，能够保证整个研发过程中顺利执行的一个指导性文档，它描述了几个方面的问题。

- 1) 描述了项目的目的
- 2) 描述了项目的开发周期
- 3) 描述了在测试中遇到的技术
- 4) 描述了测试案例的设计周期
- 5) 描述测试案例的执行周期
- 6) 描述了测试过程中用到的工具或者技术
- 7) 描述了测试过程中用到的资源情况
- 8) 描述了测试过程中可能遇到的风险以及规避方法
- 9) 提高案例设计水平

29. C/S 模式的优点和缺点

29.1. C/S 模式的优点

- 1) 由于客户端实现与服务器的直接相连，没有中间环节，因此响应速度快。
- 2) 操作界面漂亮、形式多样，可以满足客户自身的个性化要求。
- 3) C/S 结构的管理信息系统具有较强的事务处理能力，能实现复杂的业务流程。

29.2. C/S 模式的缺点

- 1) 需要专门的客户端安装程序，分布功能弱，针对点多面广且不具备网络条件的用户群体，不能够实现快速部署安装和配置。
- 2) 兼容性差，对于不同的开发工具，具有较大的局限性。若采用不同工具，需要重新改写程序。
- 3) 开发成本较高，需要具有一定专业水准的技术人员才能完成。

30. B/S 模式的优点和缺点

30.1. B/S 模式的优点

- 1) 具有分布性特点，可以随时随地进行查询、浏览等业务处理。
- 2) 业务扩展简单方便，通过增加网页即可增加服务器功能。
- 3) 维护简单方便，只需要改变网页，即可实现所有用户的同步更新。
- 4) 开发简单，共享性强。

30.2. B/S 模式的缺点

- 1) 个性化特点明显降低，无法实现具有个性化的功能要求。
- 2) 操作是以鼠标为最基本的操作方式，无法满足快速操作的要求。
- 3) 页面动态刷新，响应速度明显降低。
- 4) 无法实现分页显示，给数据库访问造成较大的压力。
- 5) 功能弱化，难以实现传统模式下的特殊功能要求。

31. 对面向过程的系统蔡依林个的集成策略有哪些?

32. 测试结束的标准是什么?

- 1) 第一类标准：测试超过了预定时间，则停止测试。
- 2) 第二类标准：执行了所有的测试用例，但并没有发现故障，则停止测试。
- 3) 第三类标准：使用特定的测试用例设计方案作为判断测试停止的基础。
- 4) 第四类标准：正面指出停止测试的具体要求，即停止测试的标准可定义为查出某一预订数目的故障。

5) 第五类标准：根据单位时间内查出故障的数量决定是否停止测试。

33. 怎么才能够全面的测试到每一个点？

答：测试的全面性主要需要在设计测试计划的时候考虑，从测试策略，产品需求等等多个角度考虑从而定义全部的测试点

34. 开发与测试的关系

答：开发和测试是一个有机的整体！在产品的发布之前，开发和测试是循环进行的，测出的缺陷要经开发人员修改后继续测试。在开发的同时测试经理开始编写测试用例，测试文档要参考开发文档，所以开发和测试是不可分割的，少了任何一个都不能开发出产品。

从角色方面看，像理论和实验的关系，开发人员通过自己的想象创造出一套思想，之后测试人员再对它进行检查、证伪，开发人员再修改的过程从而不断丰富产品。从方法方面看，是演绎和归纳的关系，一个要掌握大量的技术，一个要不断的从实例中学习。因这两方面的不同，所以开发和测试看上去做的工作很不一样。

开发与测试是相辅相承、密不可分的，开发人员开发出新的产品后要通过测试判断产品是否完全满足用户的需求。如果发现缺陷，提交给开发人员进行修复，然后再转交测试人员进行回归测试，直到产品符合需求规格说明。一个符合用户需求的产品是开发和测试共同努力的成果。

35. 项目的入口标准

答：项目需求规格说明书完成。

36. 测试过程

- 1) 制定系统测试计划
- 2) 编写系统测试用例
- 3) 执行系统测试用例
- 4) 跟踪管理缺陷
- 5) 总结测试

37. 测试出口准则

- 1) 所有的缺陷已经解决
- 2) 项目规定测试阶段时间结束
- 3) 执行完成测试计划中的系统测试内容，修正了所发现的错误，未修正的错误被项目经理允许留到下一版本
- 4) 高级经理和项目经理均同意结束测试
- 5) 测试结果经过了专门的评审

38. 测试完成标准

- 1) 系统功能与用户需求说明书一致
- 2) 功能性测试用例通过率达到 100%
- 3) 非功能性测试用例通过率达到 95%
- 4) 一、二级错误修复率应达到 100%。
- 5) 三、四级错误修复率应达到 80%以上。
- 6) 五级错误修复率应达到 60%以上。

39. 测试活动中统计了哪些数据？

答：工作量 busaa

40. 如何选择用户测试的工作产品？

答：在用户有需求得到签字确认之后，我们选择用户测试的工作产品。我们几乎所有的项目都进行了测试，我们是在项目立项公告中得知需要对工作产品进行测试。

41. 测试环境描述在哪儿？

答：测试环境在测试计划里面进行描述，测试计划是由测试经理编写，我们在测试计划中了解到自己是此次项目组的测试工程师。

42. 进行测试时产生了哪些文档或记录？

答：测试的整个过程有系统测试计划、系统测试用例、系统测试报告、缺陷报告、产品发布说明

在执行测试的过程中 只有缺陷报告，这个还是用在缺陷管理工具中进行的，最后在工具中导出缺陷报告

43. 测试人员需要何时参加需求分析？

答：如果条件循序 原则上来说 是越早介入需求分析越好 因为测试人员对需求理解越深刻 对测试工作的开展

越有利 可以尽早的确定测试思路 减少与开发人员的交互 减少对需求理解上的偏差

44. 产品测试完以后由谁来发布?

答: 这个不定 开发发布 还是技术支持发布都有可能

45. 软件测试与调试的关系

- 1) 测试条件已知, 规程可定义, 结果可预知
- 2) 测试可以计划, 过程可控
- 3) 测试是检验, 调试是推理过程
- 4) 测试表明程序失败, 调试表明正确
- 5) 测试可不了解设计细节
- 6) 测试由非设计人员完成
- 7) 测试有理论依据
- 8) 测试可自动化

46. 质量的八大特性是什么? 各种特性的定义?

1) 功能性: 软件所实现的功能达到它的设计规范和满足用户需求的程度 2) 性能: 在规定条件下, 实现软件功能所需的响应时间和计算机资源(CPU、内存、磁盘空间和数据吞吐量)的使用程度 3) 可靠性: 在满足一定条件的应用环境中, 软件能够正常维持其工作的能力, 在出现一些错误操作时, 软件可以具有容错性, 如果软件意外退出, 重新启动后可以恢复最近的软件数据 4) 安全性: 为了防止意外或人为的破坏, 软件应具备的自身保护能力 5) 使用性: 用户在理解、学习和操作软件的过程中的付出的努力的难易程度 6) 维护性: 软件在运行维护过程中, 如果出现了运行故障或者扩展新功能和性能, 软件系统是否具有可分析性和良好的扩展性, 重新设计后的软件的稳定性和可测试性 7) 移植性: 软件从现有运行平台向另一个运行平台过度的适应程度和平台可替换性 8) 重用性: 整个软件或其中一部分能作为软件包而被再利用的程度

47. 什么是软件的“质量”?

答: 高质量的软件是适当的、无错误的, 能在预算内按时交货, 满足需求/或期望, 并且是可维护的。所以, 质量是一个主观的术语。它取决于谁是客户以及客户对项目计划的影响。对一个软件开发项目来说, “客户”的范围很广, 包括最终用户、客户所接受的测试者、与客户合同有关的官员、客户管理、开发机构的管理者/会计/测试人员/销售人员、未来的软件维护工程师、股票持有者、杂志专栏记者, 等等。每一类客户对“质量”都有自己的倾向性 - 会计部门判断质量会从其收益来考虑, 而最终用户则重视友好的用户界面和没有错误。

48. 软件质量应该从哪些方面来评价?

答: 可靠性、安全性、性能、易用性、外观、稳定性

49. 什么是“软件质量保障”?

答: 软件质量保障涉及到整个软件开发过程, 包括监视和改善过程、确保任何经过认可的标准和步骤都被遵循、并且保证问题被发现和被处理。从本质上说, 软件质量保障是“预防”。

50. 为什么软件会有毛病?

1. 交流错误或者没有进行交流,
2. 软件的复杂性 编程错误
3. 需求变更 客户恐怕不明白改变需求的影响, 也许是知道但依然需要变更 —— 会导致重新设计、重订工程进度表、对其他项目的影响、已完成的工作需要重做或者放弃、对硬件需求的影响等等。如果在项目中出现许多小的改变或一个大的改变, 在项目各部分中出现已知或未知的相关的问题, 可能会相互影响并导致出现问题。而且, 不断地变更也会增加软件的复杂性, 可能会导致错误的出现。这样就会影响技术人员的积极性。在一些快速变化的商业环境里, 持续变更需求的影响是致命的。在这种情况下, 管理者必须知道它的危险性。质量保障和测试工程师必须与此相适应, 并安排持续的广泛的测试, 以克服不可避免产生的问题。

4. 时间压力

因为有许多猜测成分, 软件开发项目的进度很难安排得理想。当最后期限快到的时候, 压力逐渐增大, 错误随之产生

5. 自负心理、代码文档质量差、软件开发工具

51. 什么是验证、评价、预排、检查?

答: 验证 (verification) 涉及了回顾和会议, 以评估文档、计划、代码、需求和说明书。可以通过检查表、调查表、排练、和检查会来进行。

评价 (validation) 则指在检查完成之后的实际测试。术语“IV”和“V”分别代表验证和评价。

“预排”是一个非正式的会议，用来进行评估和信息交流。通常不需要或者只需很少一点准备。

检查比预排更正式一点，通常有 3-8 个人参加会议，包括一个仲裁者 (moderator)、读者 (可以是作者或者任何评论者)、一个记录员作记录。典型的检查对象是一个文件，例如需求说明或者测试计划，目的在于发现问题和查找遗漏，而不是去对任何东西进行实际的修改。会议的参加者应当有准备，应当通读文件，大多数的错误会在准备的过程中被发现。检查会的结果应写成书面报告。对检查会进行全面准备是困难而艰苦的工作，但它是保证质量最有用的方法。在检查过程中，最有经验的雇员的作用就向‘大哥哥’一样，他们的技能也许不大显眼，但对任何软件开发机构是最重要的，这是因为预防错误要比发现错误在费用方面更加有效。

52. 什么是 UML?

答：Unified Modeling Language 它是一种用于描述，构造软件系统以及商业建模的语言。简单的理解就是它可以以一种直观的方式表示出一个系统的各项内容

53. 什么是 CMM?

CMM: Capability Maturity Model, 即“能力成熟度模型”。它是一个分 5 级的、可以描述结构完善程度的模型，用它来说明所交付的软件的性能。

54. 对面向对象的系统采用的集成策略有?

答：自顶向下，自底向上两种。

55. 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：把测试对象当成一个黑盒子，测试人员完全不考虑逻辑结构和内部特性，只依据程式的需求说明书来检查程式的功能是否满足它的功能说明。

白盒测试：把测试对象当成一个透明的盒子，允许测试人员利用程序内部逻辑结构及相关信息，设计或选择测试用例，对程式所有逻辑路径进行测试。

单元测试：白盒测试的一种，对软件设计中的单元模块进行测试。

集成测试：在单元测试的基础上，对单元模块之间的连接和组装进行测试。

系统测试：在所有都考虑的情况下，对系统进行测试。

验收测试：第三方进行的确认软件满足需求的测试。

56. 比较负载测试、压力测试、容量测试和强度测试区别

负载测试：在一定的工作负荷下，系统的负荷及响应时间。通过逐步增加系统负载，最终确定在满足性能指标的情况下，系统能承受的最大负载量的测试。

强度测试：又称疲劳强度测试，在系统稳定运行的情况下能够支持的最大并发用户数，持续执行一段时间业务，通过综合分析，确定系统处理最大工作量强度性能的过程。一定负荷条件下，在较长时间跨度内的系统连续运行给系统性能所造成的影响。

容量测试：容量测试目的是通过测试预先分析出反映软件系统应用特征的某项指标的极限值（如最大并发用户数、数据库记录数等），系统在其极限值状态下没有出现任何软件故障或还能保持主要功能正常运行。容量测试还将确定测试对象在给定时间内能够持续处理的最大负载或工作量。容量测试的目的是使系统承受超额的数据容量来发现它是否能够正确处理。容量测试是面向数据的，并且目的是显示系统可以处理目标内确定的数据容量。

压力测试：通过逐步增加系统负载，最终确定在什么负载条件下系统性能将处于崩溃状态，以此获得系统能提供的最大服务级别的测试。

57. 测试执行过程的三个阶段

(1) 初测期

——测试主要功能和关键的执行路径，排除主要障碍。

(2) 细测期

——依据测试计划和测试大纲、测试用例，逐一测试大大小小的功能、方方面面的特性、性能、用户界面、兼容性、可用性等等；预期可发现大量不同性质、不同严重程度的错误和问题。

(3) 回归测试期

——系统已达到稳定，在一轮测试中发现的错误已十分有限；复查已知错误的纠正情况，确认未引发任何新的错误时，终结回归测试。

58. 测试的任务和发展目标——质量

59. 测试的组织形式——小组

60. 测试的运行方式——制度化并形成应用

61. 传统测试流程遇到的挑战和对策——问题发现得越早，解决的代价就越小

62. 软件生产的三个最重要的因素是：质量、进度和费用

二、需求分析

1. 需求人员需要何时参加需求分析？

答：如果条件循序 原则上来说 是越早介入需求分析越好 因为测试人员对需求理解越深刻 对测试工作的开展越有利 可以尽早的确定测试思路 减少与开发人员的交互 减少对需求理解上的偏差

原则上，测试人员对需求了解得越深入对测试工作越有利，所以最好一开始就应该参加需求分析工作。这样做可以带来如下好处：

- 1) 测试人员全程参与需求分析，对需求了解得很深入，减少了很多与开发人员的交互，节省了时间。测试人员参与前期开发讨论，直接掌握了不清晰的需求点。
- 2) 早期确定测试用例的编写思路，为测试打好基础
- 3) 可以获取一些测试数据，为测试用例设计提供帮助
- 4) 可以发现需求不合理的地方，降低了测试成本。
- 5) 测试人员主要的工作之一就是确认系统是否正确实现了需求。

2. 如果需求一直在变化怎么办？

答：这是一个常见的令人头疼的问题。

- 1) 如果可能，尽早与承担该项目风险的人接触，以便了解需求会怎样改变，从而可以尽早地改变测试计划和策略。
- 2) 如果在对应用程序进行初始设计时多考虑一些适应性，那么以后在发生需求的改变时，就不需要再为改变做很多事情了。
- 3) 好的代码注释和好的文档有助于开发人员作出相应的改变。
- 4) 只要有可能，就应使用快速原型 (rapid prototyping)，以帮助用户确认他们的需求，从而减少变更。
- 5) 在项目的时间表中应当留出余量，以应付可能出现的变更。
- 6) 尽量把新的需求纳入应用程序的“下一版”，而把原始需求作为“第一版”。
- 7) 通过谈判，把易于实现的新的变更列入项目，而把难于实现的新需求列入该应用程序的以后的版本。
- 8) 要确保让客户和管理人员了解变更对进度表的影响、所带来的风险、以及因变更所引起的大量资金消耗。
- 9) 在应付改变时，应在为建立自动测试而作的努力和重新进行测试所做的努力之间取得平衡。
- 10) 在设计自动测试剧本时，试图使其有一些灵活性。
- 11) 在对应用程序进行自动测试时，要把注意力集中在看来不大会改变的部分。
- 12) 对变更进行适当的风险分析，以减少回归测试的要求。
- 13) 在设计测试案例时要有一定的灵活性。做到这一点并不容易，所以要降低测试案例的详细程度，或者只建立高级的通用型的测试计划。
- 14) 少注意详细的测试计划和测试案例，要把重点放在专门的测试 (ad hoc testing) 上。

三、测试计划

1. 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？

答：软件测试计划是指导测试过程的纲领性文件，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。所以其中最重要的是测试策略和测试方法（最好是能先评审）

答：测试计划工作是对测试工作内容的有效的组织和规划，能保证测试工作有效的展开。测试计划工作包括测试目标，测试范围的定义，测试方法的选择，测试进度里程碑，测试资源的有效配置和管理。

测试计划工作也称为测试策略，主要描述测试工程的总体方法和目标，描述目前在进行那一阶段的测试（单元测试，集成测试，系统测试）以及每一阶段内进行的测试种类（功能测试，性能测试等）确定测试范围，生成测试数据等。

其中软件计划中的测试目标最重要，他的软件测试的所需要达成的最终结果。

2. 制定测试计划之前需要了解什么问题？

答：1.软件测试计划的目的是什么？是否所有人都知道？他们同意这个测试计划过程吗？

2.测试的是什么产品？是新程序还是维护升级的？是独立程序还是由多个小程序组成的？

3.产品的质量目标是什么？产品的功能需求和性能指标必须得到所有人的一致认可。

3. 测试计划都包括哪些项？

答：测试计划的主体部分是应该包括：对时间的安排、人力物力的分配、总体的测试策略以及对风险的评估和相应的措施！还有项目的相关简介、测试范围、测试的参考文档和测试提交的文档、测试时间的安排、人力资源的分配、系统风险的评估和优先级的定义、缺陷严重级别标准以及在接下来测试工作中的编写测试用例和缺陷报告的模板！

4. 怎样做好测试计划？

答：1.理解系统。从整个系统的高度了解被测系统必须满足的功能和非功能性需求。利用涉及整个系统的文档，形成对系统的整体了解。

2.及早介入。为了深入了解项目，测试人员应该在系统的开始阶段介入，可以增加对客户需求，客户问题，潜在风险，以及最重要的功能方面的理解

3.测试期望。程序员的期望是什么？客户的期望是什么？销售对测试的期望又是什么？测试目标必须是绝对的，以免说不清楚是否达到目标。

4.吸取教训。把以前工作中学习到的经验教训运用过来，对确定测试策略很有作用。

5.工作量大小。完成测试需要多少工作量？需要多少人员？

6.技术选择。系统会采取什么技术？系统会采用什么架构？这些信息有助于确定测试策略和测试工具。

7.时间表。系统开发和测试分配的时间有多长？截止日期是什么时候？

5. 测试资源

答：计划资源需求是确定测试策略必备条件的过程。在软件测试之前，要制定一个项目资源计划，包含每一个阶段的任务，所需要的资源，当发生类似到了使用期限或资源共享的事情时，要更新这个计划，在计划中，项目期间可能用到的任何资源都要考虑到，例如：

- 1) 人员：人数，经验和专长，全职还是兼职。
- 2) 设备：计算机，测试硬件，测试工具。
- 3) 软件：应用程序，数据库程序和自定义工具。
- 4) 其它供应：软盘，电话，参考书，培训资料。

6. 风险和问题

答：市场的压力

- 1) 测试时间不够
- 2) 测试资源的及时到位
- 3) 测试人员的技能需求
- 4) 开发进度的变化，需求的变更
- 5) 开发部门的版本控制
- 6) 短时间上线。这个是已经定好的，没有参考测试人员的意见。时间短往往不能得到充分的测试，测试策略必须根据可用的时间进行调整。尽快指出这样的问题非常重要，只有这样才能调整时间表，确定快速开发的风险并制定降低风险的策略。
- 7) 新的设计过程。引入新的设计过程会增加风险，新的设计过程包括新的工具和设计技术。如果采用新的技术，能否像我们预期的那样运转，都存在很大的风险
- 8) 复杂性。我们应该进行一些分析工作来确定哪个功能最复杂，哪个功能最容易出错，错误会对系统的哪些地方造成重大的影响。
- 9) 使用频率。软件最常用功能中隐藏的问题可能给用户造成严重的损失。
- 10) 不可测试的需求。不可测试的需求会对系统的成功造成巨大的威胁。如果测试组在需求阶段就验证了需求

的可测试性，对需求进行了评审，那么此类问题会减少很多

四、测试策略

1. 什么是“测试策略”？

答：测试策略描述测试工程的总体方法和目标 主要包括以下三个方面：

- 1 确定的测试技术和工具
- 2 制定测试启动 停止 完成标准
- 3 风险分析和应对方案

其目的 是为我们更好的写出高质量的用例 提供支撑

2. 测试策略包括哪些？

答：测试策略包括：

- 1、要使用的测试技术和工具；
- 2、测试完成标准；
- 3、影响资源分配的特殊考虑例如测试与外部接口或者模拟物理损坏、安全性威胁。

五、测试用例

1. 测试用例制定的原则？

答：测试用例要包括欲测试的功能、应输入的数据和预期的输出结果。测试数据应该选用少量、高效的测试数据进行尽可能完备的测试；基本目标是：设计一组发现某个错误或某类错误的测试数据，测试用例应覆盖方面：

- 1、 正确性测试：输入用户实际数据以验证系统是满足需求规格说明书的要求；测试用例中的测试点应首先保证要至少覆盖需求规格说明书中的各项功能，并且正常。
- 2、 容错性（健壮性）测试：程序能够接收正确数据输入并且产生正确（预期）的输出，输入非法数据（非法类型、不符合要求的数据、溢出数据等），程序应能给出提示 并进行相应处理。把自己想象成一名对产品操作一点也不懂的客户，在进行任意操作。
- 3、 完整（安全）性测试：对未经授权的人使用软件系统或数据的企图，系统能够控制的程度，程序的数据处理能够保持外部信息（数据库或文件）的完整。
- 4、 接口间测试：测试各个模块相互间的协调和通信情况，数据输入输出的一致性和正确性。
- 5、 数据库测试：依据数据库设计规范对软件系统的数据库结构、数据表及其之间的数据调用关系进行测试。
- 6、 边界值分析法：确定边界情况（刚好等于、稍小于和稍大于和刚刚大于等价类边界值），针对我们的系统在测试过程中主要输入一些合法数据/非法数据，主要在边界值附近选取。
- 7、 压力测试：输入 10 条记录运行各个功能，输入 30 条记录运行，输入 50 条记录运行。。。进行测试。
- 8、 等价划分：将所有可能的输入数据（有效的和无效的）划分成若干个等价类。
- 9、 错误推测：主要是根据测试经验和直觉，参照以往的软件系统出现错误之处。
- 10、 效率：完成预定的功能，系统的运行时间（主要是针对数据库而言）。
- 11、 可理解（操作）性：理解和使用该系统的难易程度（界面友好性）。
- 12、 可移植性：在不同操作系统及硬件配置情况下的运行性。
- 13、 回归测试：按照测试用例将所有的测试点测试完毕，测试中发现的问题开发人员 已经解决，进行下一轮的测试。
- 14、 比较测试：将已经发版的类似产品或原有的老产品与测试的产品同时运行比较，或与已往的测试结果比较。

2. 测试用例是否纳入测试基线管理？测试用例发生变更的流程？测试用例如何进行标识？

答：是。测试用例没有变更流程

测试用例的标识为 ST-001 这种格式标识

3. 什么时候编写测试用例？依据是什么？如何保证测试用例与需求的一致性？需要同行评审吗？

答：在测试计划完成之后，按照计划进度编写测试用例。

依据是软件需求规格说明书

通过同行评审来对用例进行评审，需要同行评审

4. 测试用例如何设计的？

答：在测试用例的设计之前首先要仔细阅读开发的详细设计文档，充分了解产品的详细功能，不清的地方与开发人员进行沟通，搞懂每个功能，尽量详细到输入框、按钮等小功能，功能点清楚之后按照功能模块分类进行用例编写。在具体的用例设计中会运用到等价类 边界值等黑盒测试方法

5. 如何保证用例覆盖到罕见缺陷?

答: 充足的设计时间 充分的需求分析 每一个功能点都有用例覆盖
严格的评审流程 保障输出都是有效的
在测试执行过程中 会根据实际的项目情况 对用例做增加和修改

6. 什么时候编写测试用例? 依据是什么? 如何保证测试用例与需求的一致性? 需要同行评审吗?

答: 在测试计划完成之后, 按照计划进度编写测试用例。
依据是软件需求规格说明书
通过同行评审来对用例进行评审, 需要同行评审

六、 缺陷报告

1. 缺陷报告的优先级别

答: 最高优先级: 立即修复, 停止进一步测试
次高优先级: 在产品发布之前必须修复
中等优先级: 如果时间允许应该修复
最低优先级: 可能会修复, 但是也能发布

2. 简单概述缺陷报告

答: 现在缺陷报告一般不再使用纸质档文档编写, 而是专用测试管理工具 (如 TestDirector), 这样便于缺陷管理。在这些工具中, 每个缺陷作为一条记录输入指定的缺陷管理系统中。

3. 缺陷报告包括哪些项?

答: 缺陷报告包括: 软件名称、版本号、功能模块、缺陷编号、对应的用例编号、编写时间、编写人、测试员、预期结果、实际结果、缺陷描述、严重级别、优先级别

七、 测试总结报告

1. 测试总结报告包括哪些项?

答: 主要是: 测试过程的总结 版本质量的评估、

八、 缺陷 (Bug)

1. 软件缺陷的原则

答: A、软件缺陷区别于软件 bug, 它是在测试过程中出现的对系统有影响的, 但是在设计中没有的或者对修改后的 bug 测试和开发人员有不同意见等

- B、软件未达到产品说明书标明的功能。
- C、软件出现了产品说明书指明不会出现的错误。
- D、软件功能超出产品说明书指明范围。
- E、软件未达到产品说明书虽未指出但应达到的目标。
- F、软件测试员认为软件难以理解、不易使用、运行速度缓慢, 或者最终用户认为不好。

2. 软件缺陷的特征。

- 1) “看不到”
- 2) ——软件的特殊性决定了缺陷不易看到
- 3) “看到但是抓不到”
- 4) ——发现了缺陷, 但不易找到问题发生的原因所在

软件缺陷产生的原因

- 1) 软件产品说明书 (需求) ——56%
- 2) 设计——27%
- 3) 编写代码——7%
- 4) 其他——10%

3. 什么是 Bug?

答: 软件的 Bug 指的是软件中 (包括程序和文档) 不符合用户需求的问题。

常见的软件 Bug 分为以下三类:

没有实现的功能

完成了用户需求的功能, 但是运行时会出现一些功能或性能上的问题

实现了用户不需要的多余的功能

4. 缺陷处理流程

答：测试人员提交新的 Bug 入库，错误状态为 New。高级测试人员验证错误，如果确认是错误，分配给相应的开发人员，设置状态为 Open。如果不是错误，则拒绝，设置为 Declined(拒绝)状态。开发人员查询状态为 Open 的 Bug，如果不是错误，则置状态为 Declined；如果是 Bug 则修复并置状态为 Fixed。不能解决的 Bug，要留下文字说明及保持 Bug 为 Open 状态。对于不能解决和延期解决的 Bug，不能由开发人员自己决定，一般要通过某种会议（评审会）通过才能认可。测试人员查询状态为 Fixed 的 Bug，然后验证 Bug 是否已解决，如解决置 Bug 的状态为 Closed，如没有解决置状态为 Reopen。

5. 缺陷的等级划分

答：严重：系统崩溃 数据丢失 数据毁坏
较严重：操作性失误 错误结果 遗漏功能
一般：小问题 错别字 UI 布局 罕见故障
建议：不影响使用的瑕疵或更好的实现

6. 开发人员修复缺陷后，如何保证不影响其他功能？

答：重新执行用例、看是否出现错误结果。并对周围的一些相关功能点追加新的测试用例。

7. 状态为已修改的缺陷，实际没有修改怎么办？

答：加强项目质量管理，提高项目执行能力。如果测试人员发现了这样的问题，首先要弄清楚是什么原因导致这种情况，最终还是要督促开发人员，修改掉这些问题。如果是不能重现的问题或者是老版本中遗留下来的问题不能修改的 要做好标示。

8. 生产软件的最终目的是为了客户需求，我们以客户需求作为评判软件质量的标准，认为软件缺陷（Software Bug）的具体含义包括下面几个因素

答：

- 1) 软件未达到客户需求的功能和性能；
- 2) 软件超出客户需求的范围；
- 3) 软件出现客户需求不能容忍的错误；
- 4) 软件的使用未能符合客户的习惯和工作环境。

缺陷评估

评估软件质量的重要指标，通常评估模型假设缺陷的发现是呈泊松分布的；严格的缺陷评估要考察在测试过程中发现缺陷的间隔时间长短。评估要估计软件当前的可靠性并预测随着测试的继续进行，软件可靠性会怎样提高。

SQA Suite 提供四种形式进行缺陷评估：

- 1) 缺陷分布报告可以生成缺陷数量与缺陷属性的函数。如测试需求和状态。
- 2) 缺陷趋势报告可以看出缺陷增长和减少的趋势；
- 3) 缺陷年龄报告展示一个缺陷处于某种状态的时间长短
- 4) 测试结果进度报告展示测试过程在被测应用的几个版本中的执行结果以
- 5) 测试周期。

具体步骤

- 1) 回顾测试日记
- 2) 评估测试需求的覆盖率
- 3) 分析缺陷
- 4) 决定是否达到完成测试的标准，没有满足标准时
- 5) 再测试
- 6) 降低标准
- 7) 确定软件的一个满足标准的子集，看是否可以发布。

九、白盒测试

1. 白盒测试有哪几种方法？

答：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。

白盒测试:测试证明每种内部操作和过程是否符合设计规格和要求,基于程序结构的逻辑驱动测试.静态的举例:代码走读,代码检视.

十、 单元测试

1. 单元测试主要是做一下几个事情:

- 1) 模块或者函数的设计稿
- 2) 代码规范, 其中包含代码书写规范, 对齐方式
- 3) 代码的注释。非常重要
- 4) 参数类型, 数据长度, 指针, 数组长度大小
- 5) 输入输出参数和结果
- 6) 创建对象后是否删除了, 如果在这里没有删除, 请注明在那里
- 7) 删除
- 8) 是否应用了没有初试化的变量, 如果是, 请指明该变量在那里
- 9) 初始化
- 10) 变量是否声明, 声明是否按照要求进行
- 11) 调用此函数需要的满足条件需要注明
- 12) 在此函数或者模块中用到了系统或者其他第三方插件函数, 需要满足的系统条件

十一、 集成测试

1. 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。

集成测试过程:集成测试计划->集成测试设计->集成测试实现->集成测试执行

2. 集成测试进入的准则? 退出的准则?

集成测试由开发工程师完成

进入准则: 集成完成、报告完成之后

退出准则: 按照集成构件计划及增量集成策略完成了整个系统的集成测试

达到了测试计划中关于集成测试所规定的覆盖率的要求

集成工作版本满足设计定义的各项功能、性能要求

在集成测试中发现的错误已经得到修改, 各级缺陷修复率达到标准

3. 集成测试通常都有那些策略?

- 1) 在把各个模块连接起来的时候, 穿越模块接口的数据是否会丢失;
- 2) 各个子功能组合起来, 能否达到预期要求的父功能;
- 3) 一个模块的功能是否会对另一个模块的功能产生不利的影响;
- 4) 全局数据结构是否有问题;
- 5) 单个模块的误差积累起来, 是否会放大, 从而达到不可接受的程度。

十二、 系统测试

1. 系统的性能

答: 在系统的性能方面, B/S 占有优势的是其异地浏览和信息采集的灵活性。任何时间、任何地点、任何系统, 只要可以使用浏览器上网, 就可以使用 B/S 系统的终端。

不过, 采用 B/S 结构, 客户端只能完成浏览、查询、数据输入等简单功能, 绝大部分工作由服务器承担, 这使得服务器的负担很重。采用 C/S 结构时, 客户端和服务端都能够处理任务, 这虽然对客户机的要求较高, 但因此可以减轻服务器的压力。而且, 由于客户端使用浏览器, 使得网上发布的信息必须是以 HTML 格式为主, 其它格式文件多半是以附件的形式存放。而 HTML 格式文件 (也就是 Web 页面) 不便于编辑修改, 给文件管理带来了许多不便。

2. 系统的开发

答：C/S 结构是建立在中间件产品基础之上的，要求应用开发者自己去处理事务管理、消息队列、数据的复制和同步、通信安全等系统级的问题。这对应用开发者提出了较高的要求，而且迫使应用开发者投入很多精力来解决应用程序以外的问题。这使得应用程序的维护、移植和互操作变得复杂。如果客户端是在不同的操作系统上，C/S 结构的软件需要开发不同版本的客户端软件。但是，与 B/S 结构相比，C/S 技术发展历史更为“悠久”。从技术成熟度及软件设计、开发人员的掌握水平来看，C/S 技术应是更成熟、更可靠的。

3. 系统的升级维护

答：C/S 系统的各部分模块中有一部分改变，就要关联到其它模块的变动，使系统升级成本比较大。B/S 与 C/S 处理模式相比，则大大简化了客户端，只要客户端机器能上网就可以。对于 B/S 而言，开发、维护等几乎所有工作也都集中在服务器端，当企业对网络应用进行升级时，只需更新服务器端的软件就可以，这减轻了异地用户系统维护与升级的成本。如果客户端的软件系统升级比较频繁，那么 B/S 架构的产品优势明显——所有的升级操作只需要针对服务器进行，这对那些点多面广的应用是很有价值的，例如一些招聘网站就需要采用 B/S 模式，客户端分散，且应用简单，只需要进行简单的浏览和少量信息的录入。

4. 系统测试的策略有：

功能测试，性能测试，可靠性测试，负载测试，易用性测试，强度测试，安全测试，配置测试，安装测试，卸载测试，文档测试，故障恢复测试，界面测试，容量测试，兼容性测试，分布测试，可用性测试，

5. 设计系统测试计划需要参考的项目文档有：

软件测试计划，软件需求工件和迭代计划。系统测试计划的依据是：软件需求规格说明书

6. 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。

7. Alpha 测试与 beta 的区别

Alpha 测试 在系统开发接近完成时对应用系统的测试；测试后仍然会有少量的设计变更。这种测试一般由最终用户或其它人员完成，不能由程序或测试员完成。

Beta 测试 当开发和测试根本完成时所做的测试，最终的错误和问题需要在最终发行前找到。这种测试一般由最终用户或其它人员完成，不能由程序员或测试员完成。

Alpha 测试：用户在接近正式环境下的测试，开发人员在用户旁记录错误情况和使用中的问题。

Beta 测试：多个用户在实际使用环境下的测试，开发人员不在测试现场，用户通过发送报告的形式把发现的问题反馈给开发人员。

α 测试是由一个用户在开发环境下进行的测试，也可以是公司内部的用户在模拟实际操作环境下进行的测试。

β 测试是由软件的多个用户在实际使用环境下进行的测试。这些用户返回有关错误信息给开发者。

测试时，开发者通常不在测试现场。因而， β 测试是在开发者无法控制的环境下进行的软件现场应用。

8. 系统测试阶段低级缺陷较多 怎么办？

公司有预测试这个流程 会在开展测试活动之前对主要功能点的正常流程做一个测试 以判断这个版本是不是可测试版本 如果低级缺陷比较多 严重阻碍测试执行的话 我们会打回开发部 不执行测试

10. 系统测试的进入和退出准则？

进入：（产品集成完成，打成安装包，纳入配置库）需求确认完后

退出：系统功能与用户需求说明书一致

测试计划规定的时间结束

所有确认缺陷都已修复

功能性测试用例通过率达到 100%

非功能性测试用例通过率达到 95%

11. 系统测试阶段低级缺陷较多 怎么办？

公司有预测试这个流程 会在开展测试活动之前对主要功能点的正常流程做一个测试 以判断这个版本是不是可测试版本 如果低级缺陷比较多 严重阻碍测试执行的话 我们会打回开发部 不执行测试

12. 系统测试包含哪些方面？

1. 恢复测试、2. 安全测试、3. 强度测试、4. 性能测试

十三、 验收测试

1. 软件验收测试包括：

正式验收测试，alpha 测试，beta 测试。

2.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。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。

十四、 功能测试

1. 什么是功能测试?

答: 功能测试是在规定的一段时间内运行软件系统的所有功能, 以验证这个软件系统有无严重错误

2. 请问功能测试和性能测试的区别是什么? (只总结了两个方面, 有其他的自己补充)

答: ①测试目的:

功能测试: 检查实际软件的功能是否符合用户的需求, 测功能是不是全部实现, 某个实现是不是有 BUG。主要为了发现以下几类错误: A、是否有不正确或遗漏的功能? B、功能实现是否满足用户需求和系统设计的隐藏需求? C、能否正确接收输入? 能否正确输出结果?

性能测试: 验证软件质量的三个质量特性, 可靠性, 正确性和效率。主要是测试产品的健壮性

②测试方式:

功能测试: 按照系统需求说明书和测试用例, 对产品的功能一步步进行测试。找出产品功能是否全部实现

性能测试: 一般都使用性能工具对产品的健壮性进行评估。通过创建场景和虚拟用户来模拟真是环境, 进行压力测试和负载测试。

十五、 兼容性测试

1. 兼容性测试

答: 这类测试主要想验证软件产品在不同环境之间的兼容性 主要有操作系统和浏览器

十六、 易用性测试 (可用性测试)

1. 什么是易用性测试?

答: 可使用性测试主要从使用的合理性和方便性等角度对软件系统进行检查, 发现人为因素或使用上的问题。

要保证在足够详细的程度下, 用户界面便于使用; 对输入量可容错、响应时间和响应方式合理可行、输出信息有意义、正确并前后一致; 出错信息能够引导用户去解决问题; 软件文档全面、正规、确切。

十七、 文档测试

1.文档测试

这种测试是检查用户文档(如用户手册)的清晰性和精确性。

用户文档中所使用的例子必须在测试中一一试过, 确保叙述正确无误。

2. 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

3.文档测试要注意什么?

文档的读者群、文档的术语、文档的正确性、文档的完整性、文档的一致性、文档的易用性、样例与示例、文档的语言

十八、 可靠性测试

1. 什么是可靠性测试?

答: 如果系统需求说明书中有对可靠性的要求, 则需进行可靠性测试。

平均失效间隔时间 MTBF (Mean Time Between Failures) 是否超过规定时限?

② 因故障而停机的时间 MTTR (Mean Time To Repairs) 在一年中应不超过多少时间。

十九、 安全测试

1. 什么是安全测试?

答: 安全性测试是要检验在系统中已经存在的系统安全性、保密性措施是否发挥作用, 有无漏洞。

力图破坏系统的保护机构以进入系统的主要方法有以下几种:

正面攻击或从侧面、背面攻击系统中易受损坏的那些部分；
以系统输入为突破口，利用输入的容错性进行正面攻击

二十、 自动化测试

1. 什么时候适用自动化测试？

- 1) 可重复的、不知疲倦地运动,对于数据能进行精确的大批量的比较的;
- 2) 回归测试
- 3) 在机械化的执行和比较

2. 不宜使用自动化的情况

- 1) 周期短并且一次性的项目
- 2) 进度非常紧张的项目
- 3) 需求非常不稳定的项目
- 4) 界面尚未确定
- 5) 使用了很多第三方或自定义控件的项目

二十一、 性能测试

1. 您在从事性能测试工作时，是否使用过一些测试工具？如果有，请试述该工具的工作原理，并以一个具体的工作中的例子描述该工具是如何在实际工作中应用的。

答：有使用过 LoadRunner,该工具能够录制测试人员的操作步骤，然后对这个操作步骤模拟出多个用户来播放出来。

1. Visual User Genertor 创建脚本，选择协议，录制操作，编辑操作。
2. 中央控制器 (Controller) 调度虚拟用户。创建场景，选择脚本，建立虚拟用户，设计 shedual，设置 ip spoofer。
3. 运行脚本。分析 shedual。

4. 分析测试结果。

2. 您认为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？

答：性能测试工作的目的是检查系统是否满足在需求说明书中规定的性能，性能测试常常需要和强度测试结合起来，并常常要求同时进行软件和硬件的检测。

性能测试主要的关注对象是响应时间，吞吐量，占用内存大小（辅助存储区），处理精度等。

3. 性能测试

答：性能测试是要检查系统是否满足在需求说明书中规定的性能。特别是对于实时系统或嵌入式系统。

性能测试常常需要与强度测试结合起来进行，并常常要求同时进行硬件和软件检测。

通常，对软件性能的检测表现在以下几个方面：响应时间、吞吐量、辅助存储区，例如缓冲区，工作区的大小等、处理精度，等等。

4. 性能测试什么时候开始最合适

答：一般在功能测试的最后阶段执行 因为功能走通了 性能才有意义 总之性能测试要根据用户的实际性能指标来操作 是一个很重要的测试活动 要根据软件的属性以及它的实际情况来制定策略

5. 并发性能测试的目的主要体现在三个方面？

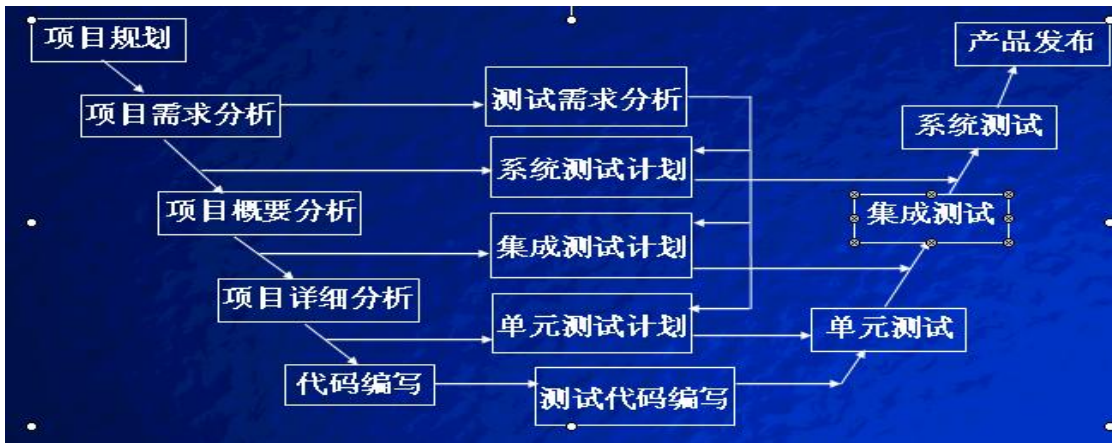
答：以真实的业务为依据，选择有代表性的、关键的业务操作设计测试案例，以评价系统的当前性能；当扩展应用程序的功能或者新的应用程序将要被部署时，负载测试会帮助确定系统是否还能够处理期望的用户负载，以预测系统的未来性能；通过模拟成百上千个用户，重复执行和运行测试，可以确认性能瓶颈并优化和调整应用，目的在于寻找瓶颈问题。

二十二、 软件开发

1. 测试在开发阶段的作用

- 1) 项目规划阶段：负责从单元测试到系统测试的整个测试阶段的监控。
 - 2) 需求分析阶段：确定测试需求分析、系统测试计划的制定，评审后成为管理项目。
 - 3) 详细设计和概要设计阶段：确保集成测试计划和单元测试计划完成。
 - 4) 编码阶段：由开发人员进行自己负责部分的测试代码。在项目较大时，由专人进行编码阶段的测试任务。
- 测试阶段（单元、集成、系统测试）：依据测试代码进行测试，并提交相应的测试状态报告和测试结束报告。

2. 完整的开发流程



3. 测试与开发各阶段的关系



4. 在软件开发过程中 5 个常见的问题是什么？

- ◆ 需求说明差 (poor requirements) —— 需求不清楚、不完整、太概括、或者不可测试，都会造成问题。
 - ◆ 不切实际的时间表 (unrealistic schedule) —— 如果在很短的时间里要求做许多事，出现错误是不可避免的。
 - ◆ 测试不充分 (inadequate testing) —— 只能根据客户意见或者系统崩溃来判断系统质量的高低。
 - ◆ 不断增加功能 (featuritis) —— 在开发正在进行过程中要求增加许多新的功能。这是常见的问题。
 - ◆ 交流问题 (miscommunication) —— 如果开发人员对客户的要求不了解，或者客户有不恰当的期望，必然会导致错误。
- (需求说明差 不切实际的时间表 测试不充分 不断增加功能 交流问题)

5. 针对软件开发过程中的问题，有 5 个解决办法：

- ◆ 可靠的需求 (solid requirements) —— 应当有一个经各方一致同意的、清楚的、完整的、详细的、整体的、可实现的、可测试的需求。为帮助确定需求，可使用模型 (prototypes)。
- ◆ 合理的时间表 (realistic schedules) —— 为计划、设计、测试、改错、再测试、变更、以及编制文档留出足够的时间。不应使用突击的办法来完成项目。
- ◆ 适当的测试 (adequate testing) —— 尽早开始测试；每次改错或变更之后，都应重新测试。项目计划中要为测试和改错留出足够的时间。
- ◆ 尽可能坚持最初的需求 (stick to initial requirements as much as possible) —— 一旦开发工作开始，要准备防止修改需求和新增功能。要说明这样作的后果。如果必须进行变更，必须在时间表上有相应的反映。如果可能，在设计阶段使用快速的模型，以便使客户了解将会得到的东西。这将会使他们对他们的需求有较高的信心，减少以后的变更。
- ◆ 沟通 (communication) —— 在适当时机进行预排和检查；充分利用团组通信工具 —— 电子邮件、群件 (groupware)、网络故障跟踪工具、变更管理工具、以及因特网的功能。要确保文件是可用的和最新的。优选电子版文档，避免纸介质文档；进行远距离联合作业及协作；尽早使用模型，使得客户的预想是清楚的。

6. 测试开发

输入：被测软件、基于测试需求的测试设计

输出：测试过程和测试用例

目标：

- ◆ 创建可以重用的测试过程和测试用例
- ◆ 维护测试过程、测试用例与相关测试需求的一一对应。

7.测试开发的问题：

- ◆ 测试开发很乱，与测试需求或测试策略没有对应性
- ◆ 测试过程不可重复或不可重用
- ◆ 测试过程被作为一个编程任务来执行，导致脚本太长，不能满足软件移植性的要求。

◆ 错误处理

二十三、 测试案例

1. 给你一个网站，你应该如何测试？

答：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等。

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

对于必须但为安装的空间，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、强度测试、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。

如果需要高级的安全性测试，确定获得专业安全公司的帮助，外包测试，或者获取支持

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。

（一个网站基本完工后，需要通过下面三步测试才可以交活）

a、制作者测试，包括美工测试页面、程序员测试功能。在做完后第一时间内有制作者本人进行测试。

b 全面测试 根据交工标准和客户要求，由专人进行全面测试

c 发布测试 网站发布到主服务器之后的测试，主要是防止环境不同导致的错误

2. 游戏测试与软件测试的区别

答：游戏测试作为软件测试的一部分，它具备了软件测试所有的一切共同的特性：

2) 测试的目的是发现软件中存在的缺陷。

- 3) 测试都是需要测试人员按照产品行为描述来实施。产品行为描述可以是书面的规格说明书，需求文档，产品文件，或是用户手册，源代码，或是工作的可执行程序。
- 4) 每一种测试都需要产品运行于真实的或是模拟环境之下。
- 5) 每一种测试都要求以系统方法展示产品功能，以证明测试结果是否有效，以及发现其中出错的原因，从而让程序人员进行改进。

游戏世界测试，主要有以下几个特性：

- 1) 游戏情节的测试，主要指游戏世界中的任务系统的组成，有人也称为游戏世界的事件驱动，我喜欢称为游戏情感世界的测试。
- 2) 游戏世界的平衡测试，主要表现在经济平衡，能力平衡（包含技能，属性等等），保证游戏世界竞争公平。
- 3) 游戏文化的测试，比如整个游戏世界的风格，是中国文化主导，还是日韩风格等等，大到游戏整体，小到 NPC（游戏世界人物）对话，比如一个书生，他的对话就必需斯文，不可以用江湖语言 J。

二十四、 测试工具

1.Quicktestprofessional

1 准备录制

打开你要对其进行测试的应用程序，并检查 QuickTest 中的各项设置是否适合当前的要求。

2 进行录制

打开 QuickTest 的录制功能，按测试用例中的描述，操作被测试应用程序。

3 编辑测试脚本

通过加入检测点、参数化测试，以及添加分支、循环等控制语句，来增强测试脚本的功能，使将来的回归测试真正能够自动化。

4 调试脚本

调试脚本，检查脚本是否存在错误。

5 在回归测试中运行测试

在对应用程序的回归测试中，通过 QuickTest 回放对应用程序的操作，检验软件正确性，实现测试的自动化进行。

6 分析结果，报告问题

查看 QuickTest 记录的运行结果，记录问题，报告测试结

2.TestDirect

安装好后，先进入站点管理

1 创建域及工程

2 添加用户

3 编辑 licenses 及本服务器

4 编辑数据库

TD

1 选择新建的工程进行定制(列表，用户，组，版本等)

2 在 require 中增加需求

3 把需求转化为 plan

4 在 testlab 中由计划新建测试具体用例与执行

5 发现 bug，在 defect 中提交 bug

（每一部分都可以相对独立地使用）

3.Loadrunner

1 制定负载测试计划

（分析应用程序，确定测试目标，计划怎样执行 LoadRunner）

2 开发测试脚本

（录制基本的用户脚本，完善测试脚本）

3 创建运行场景

（选择场景类型为 Manual Scenario，选择场景类型，理解各种类型场景的类型转化）

4 运行测试

5 监视场景

(MEMORY 相关, PROCESSOR 相关, 网络吞量以及带宽, 磁盘相关
WEB 应用程序, IIS5.0, SQL SERVER, NETWORK DELAY 等)

5 分析测试结果

6 (分析实时监视图表, 分析事务的响应时间, 分解页面, 确
WEB SERVER 的问题, 其他有用的功能)

二十五、 面试文本

1. 好的测试工程师应具备的素质?

答: 沟通能力、移情能力、技术能力、自信心、外交能力、幽默感、很强的记忆力、耐心、怀疑精神、自我督促、洞察力。

2. 软件测试给你带来什么样的快乐?

答: 测试可以给我带来很多快乐, 如果测试出一个项目缺少东西, 我会很高兴, 因为我对自己的工作有了新的认识, 也为公司做了效益; 如果测试出一个项目没有问题, 我也很高兴, 因为同事们都在努力, 大家都希望为公司做贡献, 这就是一个很强大的团队, 这是一件多么另人振奋的事情啊!

3. 为什么要在一个团队中开展测试工作?

答: 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。

软件测试在整个一个团队中占有非常重要的地位, 具体来说就是测试是一个发现软件错误的过程, 执行软件测试会以最少的人力和时间, 系统的找到软件存在的缺陷和错误, 建立起开发人员和使用者对软件的信心。

4. 你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?

答: 测试从事过 web 测试, 后台测试, 客户端软件, 进行功能测试, 性能测试, 编写测试工具, 文档的管理等, 比较擅长编写测试用例和进行功能测试。

5. 你们以前测试的流程是怎样的?

答: 测试计划—测试用例设计—测试执行—测试分析报告

6. 为什么选择测试这行?

答: 它是一个新兴的行业, 有发展潜力, 而且很锻炼人, 需要掌握更多的技能, 比做开发要更难

7. 如果时间不够, 无法进行充分的测试怎么办?

使用风险分析, 确定测试的重点。

由于很少有机会对一个应用软件进行所有可能的测试 (包括所有可能的事件组合、所有的相关性、或者一切可能出错的东西), 对大多数软件开发项目来说, 利用风险分析是适当的。这需要判断技能、常识、感觉和经验。如果有正当理由, 也可采用正式的方法。需要考虑下列因素:

- 1) 对于该项目的用途而言, 哪种功能最重要?
- 2) 哪种功能对用户最明显?
- 3) 哪种功能对安全影响最大?
- 4) 哪种功能对用户最有用?
- 5) 对客户来说, 该应用软件的哪个部分最重要?
- 6) 在开发过程中, 该应用软件的哪个部分可以最先测试?
- 7) 哪一部分代码最复杂, 容易导致出现错误?
- 8) 哪一部分的应用程序是在急迫或在惊恐的情况下开发出来的?
- 9) 哪一部分程序与过去项目中引起问题的部分相类似/有关?
- 10) 哪一部分程序与过去项目中需要大量维护的部分相类似/有关?
- 11) 需求和设计的那些部分不清楚或不容易读?
- 12) 开发人员认为在应用软件中哪些部分是高风险的?
- 13) 哪些问题能造成最差的发行?
- 14) 哪些问题最能引起用户抱怨?
- 15) 哪些测试可以容易地覆盖多种功能?

16) 哪些测试在覆盖高风险部分的测试时使用时间最少?

8. 你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?

答: 软件测试部门配合系统分析人员软件需求分析讨论, 并根据需求说明书制定《项目测试计划》, 编写测试用例, 建立测试环境。

软件测试人员负责软件开发部门的新产品测试及原有产品的升级测试, 负责软件问题解决过程跟踪, 负责软件开发文档开发工作的规范化及管理开发部门的产品文档, 制作用户手册及操作手册, 负责产品的上线测试, 监督软件开发过程的执行, 提高产品质量。

需求人员连同系统分析人员&测试人员开会讨论需求。系统分析人员写出需求分析说明, 并连同系统分析人员&测试人员&需求人员开会讨论可行性。系统分析人员写出详细设计说明书, 程式人员编码, 给出系统流程图。交与测试人员, 测试人员给出 Bug 统计表。

9. 你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……)

答: 有功能测试, 性能测试, 可靠性测试, 安全性测试, 负载测试, 压力测试, 安装/卸载测试, 启动/停止测试, 兼容性测试, 互连测试, 文档测试, 恢复测试, 回归测试, 可使用性测试, 容量测试。

功能测试只对软件的功能是否满足用户需求来做测试。性能测试需要和压力和负载测试联合起来。

10. 你自认为测试的优势在哪里?

答: 优势在于我对测试坚定不移的信心和热情, 虽然经验还不够, 但测试需要的基本技能我有信心在工作中得以发挥。

11. 你在测试中发现了一个 bug, 但是开发经理认为这不是一个 bug。你应该怎么做?

答: 首先, 将问题提交到缺陷管理库里面进行备案。

然后, 要获取判断的依据和标准:

根据需求说明书、产品说明、设计文档等, 确认实际结果是否与计划有不一致的地方, 提供缺陷是否确认的直接依据;

如果没有文档依据, 可以根据类似软件的一般特性来说明是否存在不一致的地方, 来确认是否是缺陷;

根据用户的一般使用习惯, 来确认是否是缺陷;

与设计人员、开发人员和客户代表等相关人员探讨, 确认是否是缺陷;

合理的论述, 向测试经理说明自己的判断的理由, 注意客观、严谨, 不参杂个人情绪。

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。

12. 你是如何制定时间进度表的?

答: 首先确定三个大的时间段 项目开始时间 项目结束时间 开发转系统测试时间, 在根据测试各个阶段的工作量和项目资源制定计划、设计、执行、评估、验收阶段的时间。设计和执行的时间一般较多。

13. 介绍一下整体项目流程

答: 公司将测试分为了五个阶段 计划 设计 执行 验收和评估。在测试计划阶段 我们主要工作是编写测试计划 对项目做一个整体的规划 其中进度的安排 人力物力的分配 总体测试策略的制定和风险的评估比较重要。设计阶段我们主要是编写测试策略和测试用例。在执行阶段, 当我们拿到开发转过来的版本以后, 首先是安装测试环境, 然后执行用例。发现缺陷我们会提交缺陷报告, 然后交给开发人员进行修改, 之后进行循环测试! 至于说具体系统测试会循环多少轮, 是根据项目实际情况和版本的质量来决定的。在评估阶段我们主要是编写测试总结报告, 其中对测试过程的总结和版本质量的评价要体现在测试总结报告里面 最后就是软件的验收, 验收阶段我们会和客户一同进行产品的最后检查!

14. 你是如何制定测试过程中的时间进度表的?

答: 确定项目开始 截止和转测试的时间 然后根据我们的经验值 去合理划分这五个阶段的时间 在设计和执行阶段 比重比较大

15. 测试工作进行到一半时, 发现时间不够, 你是如何处理的?

1) 先看加班 加人能不能解决 如果不行就找重点 提出优先级高的用例执行

2) 向上级申请加派测试员

3) 和客户协商, 推迟产品的发布时间

16. 怎样保证你所负责的模块通过了测试?

答：首先是了解用户的需求，设计好的测试用例，严格的进行用例的评审，认真的执行测试用例，对自己提交的 Bug 进行详细的描述。

反复测试增强测试的准确性，通过冒烟回归随机测试挖掘缺陷提高测试工作质量，把各个模块整体运行发现未曾出现的错误，完善测试用例。

17. 软件测试人员和测试组长的职责分工

普通测试人员：

创作相关的测试计划和测试案例

识别可自动测试的区域

参与组内的测试计划和测试案例以及测试脚本分析工作

手动或自动测试

按照需求规格说明查证并验证各项功能

发现并报告 bug，跟踪其状态

初步评估 bug 对产品其他部分的影响

测试组长：

确定测试的策略

参与对整个产品的完整测试计划的制定

参与并管理测试

评估 bug 对用户的影响

跟踪关键 bug 状态

管理测试工作和对象的资源

参与面试新人

交流状态和存在问题，并驱动问题的解决

促进组内的交流

18. 如果你是测试组长你是如何对项目及组员进行管理的?

答：首先要充分了解要测试的项目，参考开发文档同时与开发人员及时沟通，要对产品十分的熟悉。员工方面多与员工沟通，了解员工的擅长的工作，根据员工擅长的工作进行分配，能力强的多分配，这样可以使测试工作快速稳定的进行！最终的测试工作就会一帆风顺！

19. 什么时候开始搭建测试环境？由谁搭建？如何进行产品的集成？

答：测试开始之前搭建测试环境

由测试工程师搭建

产品的集成由开发人员完成

20. 你所做的项目中采用了哪些测试方法？进行回归测试吗？

答：功能测试 界面测试 安装测试 性能测试

进行了回归测试 一般在缺陷修复之后进行验证的过程中进行回归测试

21. 上级如何检查你的工作？

答：查看项目周报 开周例会

22. QA 是如何检查你的工作的？

答：检查项目过程及文档 参与周例会

23. 你参加了哪些培训？

答：参加了测试工具 TestDirector 的培训

参加了 CMMI 过程改进的培训

24. 在你所做的项目中有需要测试的项目过程吗？有，请介绍。

答：有啊，我们在奇瑞，华师，网校项目中都进行了测试。在测试过程中，我们先根据用户需求进行测试计划，编写测试用例，在开发部完成项目并进行产品集成，交由配置管理员纳入配置库之后，我们从配置库签出安装包，搭建测试环境，开始进行测试，测试过程中我们提交缺陷，开发人员修复缺陷，直到所有缺陷修改完毕，做测试总结。最终完成测试过程。

25. 怎样保障你所负责的模块通过了测试？

答：仔细分析需求，制定测试计划与策略，详细编写测试用例因实际情况合理修改并执行，经过严格的评审，总结出真实测试报告。

首先在设计阶段要保证所设计的用例的覆盖率，发现缺陷的能力；在提交缺陷报告时要对缺陷进行详细确切的描述，方便开发人员进行修复；修复后认真进行回归测试，直到产品符合用户需求。

- 1.通过反复测试来增强测试的准确性
- 2.还可以通过冒烟、回归、随机测试来挖掘冒烟发现的缺陷，提高测试工作的质量
- 3.把各个模块整体运行时来发现未曾出现的错误
- 4.完善测试用例

26. 你是如何了解到你说项目中的成员？

答：在项目立项公告中

27. 是否成立了独立的测试组？测试人员在项目中测试的职责？

答：是。验证需求是否正确实现 发现软件中存在的缺陷 确认软件缺陷被修复。

28. 测试结果分析如何？如何产生和被记录？

答：在项目测试之后，我们对缺陷进行了统计分析，并生成了测试报告文档。在此次项目中所有的缺陷都已修复并关闭。所有的缺陷都记录在缺陷管理工具中，并导出了缺陷报告

29. 你认为软件测试过程中较常见的困难是什么？如何有效克服这些困难？（根据自己实际测试中遇到的情况来写的）

答：①Bug 的重现问题：有些 Bug 只是偶尔出现的，根本就不知道具体需要什么条件才能重现 Bug.

解决方法：将不能重现的 Bug，利用截图的方式记录下来。并说明一系列的操作步骤

②Bug 的更新：旧的 Bug 修改好之后，很多时候会引发更多 Bug 的出现。

解决方法：对更新的功能模块重点的测试之后，再重新测试和更新的功能密切的模块，会不会产生新的 Bug.

③与开发人员的沟通和对业务流程理解的分歧，经常缺少需求文档

解决方法：根据需求说明书和 Bug 情况，多多和开发人员进行交流

30. 在实际项目中你是如何做测试计划？

答：在做计划之前 我们要先了解这个项目的大致情况 比如测试的是什么产品？是新程序还是维护升级的？是独立程序还是由多个小程序组成的？

产品的质量目标是什么？产品的功能需求和性能指标必须得到所有人的一致认可。

为了深入了解项目 测试人员应该及早介入项目 对工作量的大小 时间点的安排作出了解。然后我们的测试计划包括。。。。。。

31. 你什么时候开始制定测试计划？是否发生过变更？如何进行变更？

答：在需求完成并得到确认之后开始制定测试计划 没有发生变更

变更流程：

项目经理向部门经理提交《计划变更申请单》。其中应当说明：变更原因、变更内容及变更可能对项目造成的影响和影响范围

由项目管理部组织对项目经理提交的《计划变更申请单》进行评审，评审人中应包括高层经理、CCB 成员，如果评审结论是不同意变更项目计划，则退回《计划变更申请单》，项目要按照原计划执行。如果评审结论是同意变更项目计划，则执行变更

修改项目计划

项目计划及其下属计划的评审和审批

工作产品纳入配置库

32. 你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。

答：1. 等价类划分 2. 边界值分析法 3. 错误推测法 4. 因果图方法

有黑盒和白盒两种测试种类，黑盒有等价类划分法，边界分析法，因果图法和错误猜测法。白盒有逻辑覆盖法，循环测试路径选择，基本路径测试。

例子：在一次输入多个条件的完整性查询中。利用等价类划分法则和边界分析法则，首先利用等价类划分法，可以一个或多个结果是 OK 的测试用例，然后确认多个 NG 的测试用例，然后利用边界值分析法，可以对结果分别是 OK 和 NG 的测试用例进行扩展和补充。

请详述设计测试用例的方法？（只是列出一个测试用例思考的方向，具体设计靠经验）

①黑盒测试用例根据业务需求说明书来设计，分为：

等价划分法 边界值分析法 错误推测法 因果图法 逻辑覆盖法

②白盒测试用例通过研究代码与程序结构可以分为以下两种方式：

静态测试：通过静态的检查程序代码、界面、文档中可能存在的错误的过程。

|-测试代码编写的规范性 |-测试界面 |-测试相关需求说明和用户手册是否符合实际要求

动态测试：通过路径和分支测试。测试用例主要根据以下六种覆盖测试方法设计

|-语句覆盖 |-判定覆盖 |-条件覆盖 |-判定/条件覆盖 |-组合覆盖 |-路径覆盖

33. 你认为做好测试用例设计工作的关键是什么？

- 1) 明确测试的目标，增强测试计划的实用性
- 2) 坚持“5W”规则，明确内容与过程，'what' 'why' 'when' 'where' 'how'
- 3) 采用评审和更新机制，保证测试计划满足实际需求
- 4) 分别创建测试计划与测试详细规格、测试用例

白盒测试用例设计的关键是以较少的用例覆盖尽可能多的内部程序逻辑结果

黑盒法用例设计的关键同样也是以较少的用例覆盖模块输出和输入接口。不可能做到完全测试，以最少的用例在合理的时间内发现最多的问题

34. 在你以往的工作中，一条软件缺陷（或者叫 Bug）记录都包括哪些内容？如何提交高质量的软件缺陷（Bug）记录？

答：检测时间，系统环境，硬体环境，严重程度，程式版本，确认人，功能模块，问题描述，详细操作步骤，是否会重现。

问题描述和详细操作步骤要尽可能的详细。Bug 应该尽量用书面语，对与严重程度比较高的缺陷要在相同环境下在测试一遍。

在 C/S 模式下，如果条件满足可以使用替换法来确认是 client 端的问题还是 server 端的问题。

35. 你在五年内的个人目标和职业目标分别是什么？

答：从现在起的五年之内，我希望能够在在一个很好的职位上待几年，而且最好有一次晋升，然后就期待着下一步。不管是向上提升，还是在企业内横向调动，对我个人来说，我希望找到一家企业——一家愿意做相互投入的企业——待上一段时间。

36. 你怎样做出自己的职业选择？

答：在上大学四年级前的那个夏天，我决定集中精力在某一领域谋求发展。尽管我是学商业的，但是我不知道自己最终会从事哪一行业的工作。我花了一定的时间考虑自己的目标，想清楚了自己擅长做的事情以及想从工作中得到的东西，最后我得出了一个坚定的结论，那就是这个行业是最适合我的。

37. 是否了解公司的组织过程财富库？

答：了解

组织财富库里面包含：项目角色说明、工作环境指南、组织度量库、质量方针、评审报告、过程裁剪指南、组织过程改进方针、风险数据库、体系综述、软件生命周期说明、过程资产创建与维护规程、培训数据库等等……

二十六、 网络题库

1. 避免死锁的方法

- (1) 互斥条件：一个资源每次只能被一个进程使用。
- (2) 请求与保持条件：一个进程因请求资源而阻塞时，对已获得的资源保持不放。
- (3) 不剥夺条件：进程已获得的资源，在未使用完之前，不能强行剥夺。
- (4) 循环等待条件：若干进程之间形成一种头尾相接的循环等待资源关系。

这四个条件是死锁的必要条件，只要系统发生死锁，这些条件必然成立，而只要上述条件之一不满足，就不会发生死锁。

2. 内存有哪几种存储组织结构，请分别加以说明。

ROM 相当于 PC 机的 BIOS，Cisco 路由器运行时首先运行 ROM 中的程序。该程序主要进行加电自检，对路由器的硬件进行检测。其次含引导程序及 IOS 的一个最小子集。ROM 为一种只读存储器，系统掉电程序也不会丢失。

FLASH 是一种可擦写、可编程的 ROM，FLASH 包含 IOS 及微代码。可以把它想象和 PC 机的硬盘功能一样，

但其速度快得多。可以通过写入新版本和 OS 对路由器进行软件升级。FLASH 中的程序，在系统掉电时不会丢失。

DRAM：动态内存。该内存中的内容在系统掉电时会完全丢失。DRAM 中主要包含路由表，ARP 缓存，fast-switch 缓存，数据包缓存等。DRAM 中也包含有正在执行的路由器配置文件。

NVRAM：NVRAM 中包含有路由器配置文件，NVRAM 中的内容在系统掉电时不会丢失。

3. 操作系统中的同步和互斥解决了什么问题？

同步：各个进程不知对方名字，但通过某些对象（如 I/O 缓冲区）的共同存取来协同完成一项任务。

互斥：互斥跟临界资源有关，因为计算机的某些资源有限，所以必须通过互斥操作防止进程之间竞争临界资源而发生死锁，互斥操作用 PV 原语实现。

4. 什么是数据的独立性？有几个层次？

数据独立性是数据库系统的一个最重要的目标之一，它使数据能独立于应用程序。

数据独立性包括数据的物理独立性和数据的逻辑独立性。

5. 网状、层次数据模型与关系数据模型的最大区别是什么？

ODBC(Open Database Connectivity, 开放数据库互连)

创建 ODBC 连接之前，请先安装和配置数据库客户机。请咨询数据库管理员以获知适当的客户机配置。安装和配置数据库客户机后，设置 ODBC 数据源连接从运行 Tivoli Risk Manager Crystal Reports 的系统访问 Tivoli Risk Manager 数据库。

6. 简述 ODBC 的设置过程

层次模型与网状模型类似，分别用记录和链接来表示数据和数据间的联系。与网状模型不同的是：层次模型中的记录只能组织成树的集合而不能是任意图的集合。

关系模型用表的集合来表示数据和数据间的联系

7. 局域网对等网的含义及一般配置连接方法

局域网 (Local Area Network, LAN) 是指在某一区域内由多台计算机互联成的计算机组。一般是方圆几千米以内

对等网通常是由很少几台计算机组成的工作组。对等网采用分散管理的方式，网络中的每台计算机既作为客户机又可作为服务器来工作，每个用户都管理自己机器上的资源，是一种投资少、见效快、高性价比的实用型小型网络系统

8. 两台计算机连接的方法有哪些？

- 1、交叉线连接
- 2、通过交换接连接
- 3、USB 对接线连接、蓝牙连接
- 4、红外（比较简单的连接，不过对距离要求很苛刻）
- 5、无线局域网（需要两个电脑都有无线网卡）

9. 介绍 plc 或板卡的大致功能和其在行业中的一般应用场合。

PLC：是一种以微处理技术为基础的数字运算的电子系统，专为在工业环境下应用而设计。将控制处理规则存储于存储器中，应用于以控制开关量为主或包括控制参量在内的逻辑控制、机电运动控制或过程控制等工业控制领域的新型工业控制装置。

板卡：计算机中的一种印制电路板，可以插入计算机的主电路板，控制一定硬件的运行，比如显示器及另外一些元件等

10. 组态软件

组态软件，又称监控组态软件，译自英文 SCADA,即 Supervision,Control and Data Aquisition(数据采集与监视控制),组态软件的应用领域很广，它可以应用于电力系统、给水系统、石油、化工等领域的数据采集与监视控制以及过程控制等诸多领域。在电力系统以及电气化铁道上又称远动系统(RTU System,Remote Terminal Unit)。简单的讲，组态就是用应用软件中提供的工具、方法、完成工程中某一具体任务的过程。与硬件生产相对照，组态与组装类似。

几种常用组态软件介绍：InTouch、MCGS、iFIX、WinCC、EcHmi、组态王

11. 写出下列端口号：ftp、dns、http、pop3、smtp、telnet、windows 终端服务

ftp: 21 dns:53 http: 80 pop3: 110
 smtp:25 snmp: 161 telnet: 23 QQ 使用的 8000 和 4000 端口

12. 写出 ISO 模型, tcp/udp 在那层? 他们有什么区别

OSI 模型: 物理层, 数据链路层, 网络层, 传输层, 会话层, 表示层, 应用层

tcp/udp 在 TCP/IP 的传输层

应用层—应用层是所有用户所面向的应用程序的统称。

传输层—这一层的的功能主要是提供应用程序间的通信

网络层—是 TCP/IP 协议族中非常关键的一层, 主要定义了 IP 地址格式, 从而能够使得不同应用类型的数据在 Internet 上通畅地传输, IP 协议就是一个网络层协议。

13. 写出 HUB、tch、Router 是 ISO 的第几层设备, 叫什么?

OSI 模型一共 7 层, 1-7 层分别是: 物理层 数据链路层 网络层 传输层 会话层 表示层 应用层

HUB 是 ISO 的第一层(物理层)设备, 集线器

Tch 是 ISO 的第二层(数据链路层)设备, 交换机

Router 是 ISO 的第三层(网络层)设备, 路由器

14. 在防火墙中有哪些有效的设置方法

所谓防火墙指的是一个有软件和硬件设备组合而成、在内部网和外部网之间、专用网与公网之间的界面上构造的保护屏障.是一种获取安全性方法的形象说法, 它是一种计算机硬件和软件的结合, 使 Internet 与 Intranet 之间建立起一个安全网关 (Security Gateway), 从而保护内部网免受非法用户的侵入, 防火墙主要由服务访问政策、验证工具、包过滤和应用网关 4 个部分组成,

防火墙就是一个位于计算机和它所连接的网络之间的软件或硬件(其中硬件防火墙用的很少只有国防部等地才用,因为它价格昂贵)。该计算机流入流出的所有网络通信均要经过此防火墙。

防火墙对流经它的网络通信进行扫描, 这样能够过滤掉一些攻击, 以免其在目标计算机上被执行。防火墙还可以关闭不使用的端口。而且它还能禁止特定端口的流出通信, 封锁特洛伊木马。最后, 它可以禁止来自特殊站点的访问, 从而防止来自不明入侵者的所有通信。

防火墙具有很好的保护作用。入侵者必须首先穿越防火墙的安全防线, 才能接触目标计算机。你可以将防火墙配置成许多不同保护级别。高级别的保护可能会禁止一些服务, 如视频流等, 但至少这是你自己的保护选择。防火墙有不同类型。一个防火墙可以是硬件自身的一部分, 你可以将因特网连接和计算机都插入其中。防火墙也可以在一个独立的机器上运行, 该机器作为它背后网络中所有计算机的代理和防火墙。最后, 直接连在因特网的机器可以使用个人防火墙。

15. ms-dos6.22 支持的分区最大为多少

4G

16. ms-dos6.22 能管理的硬盘容量最大为多大

8G

17. 一台电脑由哪些部分组成,最小化硬件系统指的是需要哪些硬件

基本由主机、输入设备和输出设备组成的。最小化硬件系统: 电源、主板和 CPU

18. 目前一台普通的 pc(主板)最多能接几个硬盘

不接光驱的话 IDE 上接 4 块, SATA 也能接 4 块, USB 如果最低有两个的话再接 2 块移动硬盘,一共 10 块,前提电源要带动,主能认出这些容量

19. 请说出下列 dos 命令的作用

dir/ad/od xcopy a:/s scandisk.exe disk.exe 并简述命令的过程

format a:/s 并说出执行命令后 a:中有哪些文件

dir/ad/od按分类顺序列出文件

xcopy a:/s 复制 a:/s 文件夹和文件夹内所有内容

scandisk.exe 检测、修理磁盘

fdisk.exe 分区命令

format a:/s 格式化 a 盘, 并且格式化后 a 盘只能读写

20. 在 windows98 启动前按 f5 或 f8 分别有什么作用

F5 键进入 Windows 98 的安全模式

F8 键显示启动菜单，可以从菜单中选择启动方式

21. 你知道的微软的操作系统有哪些,你使用的是哪个,能熟练使用哪个

Windows xp, windows server 2003, windows7, windows98, Vista……

我使用的是 Windows xp, 能熟练使用 Windows xp

22. 为什么有些分区在 windows98 中看不到而在 windows xp 中却可以看到

因为 Windows 98 不能识别 NTFS 分区

23. 请简述 ip 地址,网关,子网掩码的含义

IP 地址是 TCP/IP 网络中的主机(或称为节点)的惟一地址。IP 地址是网络层的逻辑地址

缺省网关(Default Gateway)是指缺省的路由器。只有在不同子网之间通信时,才需要配置缺省网关的 IP 地址

子网(Subnet)是在 TCP/IP 网络上,用路由器连接的网段,子网掩码(Subnet Mask)用来确定 IP 地址中的网络地址部分。其格式与 IP 地址相同,也是一组 32 位的二进制数。

24. 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?

首先检查 IP 与路由器是否在同一网段,桌面-网上邻居-属性,设置 ip 和 dns、网关等项。

win98 的默认安装中没有安装宽带上网的插件,添加了 windows 组件后运行 winipcfg 后获取 ip 就可以了。

25. 中间件的含义,意义,主要产品名称?

中间件是一种计算机软件连接了软件部件或者应用程序。这种软件有一组服务构成,这些服务包括允许多进程运行在一个或者多个机器上以达到在网络中互相交互的目的。

26. 简述一下 c/s 模式或者 b/s 模式

B/S 是 Browser/Server 就是用浏览器(如 ie)为应用程序客户端操作服务器。这样用浏览器来操作简单易用,但是对输入没有很好的验证。逻辑实现不多。本地只是用于获取数据 然后大部分验证需要提交服务器来完成。

C/S 是 Client/Server 客户端软件则是一个拥有很多逻辑验证的软件,本地验证成功后 才提交服务器。

27. 一台计算机的 IP 是 192.168.10.71 子网掩码 255.255.255.64 与 192.168.10.201 是同一局域网吗?

子网掩码不对。不可能出现 255.255.255.64 的子网掩码。

另外,也不能说成“同一局域网”,局域网是针对物理的拓扑结构而言。事实上,我们研究的是否在同一子网的一些 IP,往往都是同一个局域网内。

28. internet 中 e-mail 协议,IE 的协议,NAT 是什么,有什么好处,能带来什么问题?DNS 是什么,它是如何工作的?

NAT 全称 Network Address Translation, 中文解释为“网络地址转换”。NAT 是一种 IETF (Internet Engineering Task Force) 的标准,简单描述其功能就是让处于内网的计算机能够通过 NAT 的作用透明的访问外网的互联网资源。NAT 的功能一般集成在路由器、防火墙或者单独的 NAT 设备中。

DNS 全名是 Domain Name System, 透过 DNS 系统,我们可以由一部机器的 domain name 查其 IP,也可以由机器的 IP 反查它的 domain name,除此之外 DNS 还与 Mail System 结合,提供 Mail routing 的功能。

DNS 分为 Client 和 Server,Client 扮演发问的角色,也就是问 Server 一个 Domain Name,而 Server 必须要回答此 Domain Name 的真正 IP 地址。而当地的 DNS 先会

29. PROXY 是如何工作的?

Proxy 是什么呢,是代理。普通的因特网访问是一个典型的客户机与服务器结构:用户利用计算机上的客户端程序,如浏览器发出请求,远端 WWW 服务器程序响应请求并提供相应的数据。而 Proxy 处于客户机与服务器之间,对于服务器来说,Proxy 是客户机,Proxy 提出请求,服务器响应;对于客户机来说,Proxy 是服务器,它接受客户机的请求,并将服务器上传来的数据转给客户机。它的作用很象现实生活中的代理商。因此 Proxy Server 的中文名称就是代理服务器。

Proxy Server 的工作原理是:当客户在浏览器中设置好 Proxy Server 后,你使用浏览器访问所有 WWW 站点的请求都不会直接发给目的主机,而是先发给代理服务器,代理服务器接受了客户的请求以后,由代理服务器向目的主机发出请求,并接受目的主机的数据,存于代理服务器的硬盘中,然后再由代理服务器将客户要求的数据发给客户。

30. win2k 系统内 AT 命令完成什么功能,Messenger 服务是做什么,怎么使用?

AT 命令可在指定时间和日期、在指定计算机上运行命令和程序。

Messenger 服务:发送和接收系统管理员或者“警报器”服务传递的消息。

31. 进程,线程的定义及区别

进程是具有一定独立功能的程序关于某个数据集合上的一次运行活动,进程是系统进行资源分配和调度的一个独立单位。

线程是进程的一个实体,是 CPU 调度和分派的基本单位,它是比进程更小的能独立运行的基本单位.线程自己基本上不拥有系统资源,只拥有一点在运行中必不可少的资源(如程序计数器,一组寄存器和栈),但是它可与同属一个进程的其他的线程共享进程所拥有的全部资源.

一个线程可以创建和撤销另一个线程;同一个进程中的多个线程之间可以并发执行

32. 32 位操作系统内,1 进程地址空间多大,进程空间与物理内存有什么关系?

4GB, 物理内存是真正的内存, 在应用中顾名思义, 物理上, 真实的插在板子上的内存是多大就是多大了。看机器配置的时候, 看的就是这个物理内存。

虚拟内存是为了满足系统对超出物理内存容量的需求时在外存(如硬盘)上开辟的存储空间。由于虚拟内存其实是放在外存上, 因而与物理内存相比读写速度都非常慢

33. 网络攻击常用的手段,防火墙如何保证安全.

口令入侵

放置特洛伊木马程序

WWW 欺骗技术

电子邮件攻击

通过一个节点攻击其他节点

网络监听

利用黑客软件攻击

利用安全漏洞攻击

端口扫描攻击

(一) 内部网络和外部网络之间的所有网络数据流都必须经过防火墙

(二) 只有符合安全策略的数据流才能通过防火墙

(三) 防火墙自身应具有非常强的抗攻击免疫力

34. 如何配静态 IP,如何测网络内 2 台计算机通不通,PING 一次返几个数据包?

右键网上邻居, 点属性, 打开网络连接, 右键本地连接, 点属性, 打开本地连接属性, 双击 internet 协议 (TCP/IP), 配置 IP, ping 另外的机器通则正常, 否则不正常。4 个

35. WIN9X 与 WINNT 以上操作系统有"服务"吗,服务是什么,如何停止服务?

有服务, 可以在命令提示符下输入 net stop 服务名, 回车即可。或在控制面板--> 管理 工具--> 服务下启动或停止服务

36. AD 在 WIN2KSERVER 上建需什么文件格式,AD 是什么?XP 多用户下"注销"与"切换"的区别.

需要 NTFS 格式

AD: Active Directory 是指 Windows 2000 网络中的目录服务。

它有两个作用: 1.目录服务功能。2.集中式管理。

注销与切换的区别:

最大的区别在于利用“切换用户”的功能可以不中止当前用户所运行的程序甚至不必关闭已打开的文件, 而进入其他用户的工作桌面, 而“注销用户”则必须中止当前用户的一切工作。

37. UDP 可以跨网段发送吗?

udp 可以跨网段,只要你的 router 允许你跨相应的网段。还有, 如不能直接连出, 利用代理也可以。

38. 最简单的确认远程计算机(win2K 以上)某个监听端口是正常建立的?

用 telnet 远程 IP 地址 端口

39. winrunner,loadrunner 是什么,区别

应用对象不同

WR 主要是针对功能测试和回归测试

LR 主要针对性能测试

40. 磁盘分区如何分类,请举例说明安装操作系统的注意事项.

在 DOS 的系统架构下,分区有三种:主分区 (Primary Partition)、扩充分区 (Extended Partition)、逻辑磁盘 (Logical Drive)。一块硬盘上只能有一个主分区。

在已经存在一个主分区的硬盘上, DOS 不允许您再建立其他主分区, 不过您却可以再建立逻辑磁盘。但使用其他的分区程序, 如 OS/2 的 Boot Manager 便可以在一块硬盘中分区出至多四个主分区 (对 Boot Manager 来说, 扩充分区也算一个主分区), 然后选择要用来启动的分区 (假设该分区中存在有操作系统的引导代码)

41. 在网络应用测试中能够,网络延迟是一个重要指标。以下关于网络延迟的理解,正确的是 (D)

- A. 之响应时间
- B. 指报文从客户端发出到客户端收到服务器相应的间隔时间
- C. 指报文在网络上的传输时间
- D. 指从报文开始进入网络到它开始离开网络之间的时间

42. 模块的耦合度描述了 (D)

- A. 模块内各种元素结合的程度
- B. 模块内多个功能之间的接口
- C. 模块之间公共数据的数量
- D. 模块之间相互关联的程度

43. 耦合度, 内聚

耦合度是指模块之间联系的紧密程度

内聚 (Cohesion) 是一个模块内部各成分之间相关联程度的度量

44. 网络七层协议具体描述

第一层: 物理层 (PhysicalLayer), 规定通信设备的机械的、电气的、功能的和过程的特性, 用以建立、维护和拆除物理链路连接。

属于物理层定义的典型规范代表包括: EIA/TIA RS-232、EIA/TIA RS-449、V.35、RJ-45 等

第二层: 数据链路层 (DataLinkLayer): 在物理层提供比特流服务的基础上, 建立相邻结点之间的数据链路, 通过差错控制提供数据帧 (Frame) 在信道上无差错的传输, 并进行各电路上的动作系列。数据链路层在不可靠的物理介质上提供可靠的传输。该层的作用包括: 物理地址寻址、数据的成帧、流量控制、数据的检错、重发等。

数据链路层协议的代表包括: SDLC、HDLC、PPP、STP、帧中继等。

第三层是网络层, 在计算机网络中进行通信的两个计算机之间可能会经过很多个数据链路, 也可能还要经过很多通信子网。网络层的任务就是选择合适的网间路由和交换结点, 确保数据及时传送。网络层将数据链路层提供的帧组成数据包, 包中封装有网络层包头, 其中含有逻辑地址信息--源站点和目的站点地址的网络地址。网络层协议的代表包括: IP、IPX、RIP、OSPF 等。

第四层是处理信息的传输层。第 4 层的数据单元也称作数据包 (packets)。但是, 当你谈论 TCP 等具体的协议时又有特殊的叫法, TCP 的数据单元称为段 (segments) 而 UDP 协议的数据单元称为“数据报 (datagrams)”。这个层负责获取全部信息, 因此, 它必须跟踪数据单元碎片、乱序到达的数据包和其它在传输过程中可能发生的危险。第 4 层为上层提供端到端 (最终用户到最终用户) 的透明的、可靠的数据传输服务。所谓透明的传输是指在通信过程中传输层对上层屏蔽了通信传输系统的具体细节。

传输层协议的代表包括: TCP、UDP、SPX 等。

第五层是会话层, 这一层也可以称为会话层或对话层, 在会话层及以上的高层次中, 数据传送的单位不再另外命名, 统称为报文。会话层不参与具体的传输, 它提供包括访问验证和会话管理在内的建立和维护应用之间通信的机制。如服务器验证用户登录便是由会话层完成的。

第六层是表示层, 这一层主要解决拥护信息的语法表示问题。它将欲交换的数据从适合于某一用户的抽象语法, 转换为适合于 OSI 系统内部使用的传送语法。即提供格式化的表示和转换数据服务。数据的压缩和解压缩, 加密和解密等工作都由表示层负责

第七层应用层, 应用层为操作系统或网络应用程序提供访问网络服务的接口

应用层协议的代表包括: Telnet、FTP、HTTP、SNMP 等。

45. 路由器,交换机的功能是什么,子网掩码如何分配

交换机能把处于同一个网段的若干个终端设备连接起来, 而路由器则能把两个或者几个不同的网段连

接起来

子网掩码就是用来表示有多少位做网络号，多少位做主机号的，其全1和全0组成，并且全1在前。所以在没有划分子网的情况下C类的掩码就是24个1和8个0。

46. winXP下局域网不通,怎么办

- 1、查看网络配置是否正确(TCP/IP协议的配置数据，IP地址、子网掩码、默认网关)
- 2、查看网络接口有无故障

47. 对RUP,CMM,CMMI,XP,PSP,TSP的认识

RUP是最常用软件工程过程在片刻内，定义了四个阶段和九个核心工作流，RUP中每一个迭代都可能贯穿这四个阶段和九个核心工作流，但不是一定就会。

CMM是Capability Maturity Model for Software的简称，中文叫“软件能力成熟度模型”，是对组织软件过程能力的描述。

CMMI (Capability Maturity Model Integration——能力成熟度模式整合)的本质是软件管理工程的一个部分。

PSP全称PlayStation Portable，是一种由SONY开发的新型掌上游戏机。

48. Ghost的主要用途和常用方法

主要用途：将硬盘做磁区分割

分区备份、硬盘克隆与备份、备份还原

49. 写出Weblogic控制台访问地址(端口缺省)

格式如下：

[http://\[weblogic所在机器IP\]:\[weblogic端口\]/console](http://[weblogic所在机器IP]:[weblogic端口]/console)

若没有指定端口，且安装在本机上则为：(weblogic默认端口为7001)

<http://localhost:7001/console>

50. 手工修改Tomcat端口，在那个文件里？

Tomcat服务器的conf目录下的主配置文件server.xml

51. WinRunner分几种录制模式，有什么特点？

一般录制模式和低级录制模式

1.record-context sensitive 只记录到动作开始，一直到动作结束，例如画图工具，在录制过程中，只记录了画图的起点与结束的重点，是一条直线，不会记录绘图过程中的每一个动作过程，是经常用到的录制方式。

主要是以gui对象为基础，识别使用者点选的GUI对象(如窗口，菜单，按钮等)，以及执行的操作(如按下，移动，选取等)

2.record-analog 记录鼠标轨迹、鼠标的点选以及键盘的输入三种动作。例如画图工具，记录了画图的每一个动作过程，但是回放起来很慢，一般没有特殊情况不会使用到这个录制方式。

52. WinRunner 运行测试脚本时，有哪些模式？在什么情况下使用

在检验模式(Verify mode)下测试被测软件。WR在脚本运行中遇到检查点后，就把当前数据和前期捕捉的期望值进行比较。如果发现有不符合，就记录下来作为实测结果。

53. 怎么把C:\test目录配制成FTP目录，供远程端使用

先在C盘下建test文件夹，在IIS服务器上新建FTP站点，浏览路径：C:\test

54. 怎么配置Weblogic域？

1. 创建一个WTC服务
2. 创建一个本地Tuxedo访问点
3. 创建一个远程Tuxedo访问点
4. 导出服务
5. 导入服务
6. 为服务器分配新创建的WTC服务

55. 文件格式系统有哪几种类型？分别说说win95、win98、winMe、w2k、winNT、winXP分别支持那些文件系统。

FAT(File Allocation Table)是“文件分配表”的意思。

对我们来说，它的意义在于对硬盘分区的管理。FAT16、FAT32、NTFS是目前最常见的三种文件系统。

Win95: FAT16 和 FAT32

Win98: FAT16, FAT32

winMe: FAT16, FAT32

w2k: FAT(FAT16), FAT32, NTFS

winNT: FAT16/FAT32/NTFS

winXP: FAT16, FAT32, NTFS

56. 我现在有个程序，发现在 WIN98 上运行得很慢，怎么判别是程序存在问题还是软硬件系统存在问题？

可能是病毒或者恶意程序啊，如果是程序运行慢那肯定是软件问题，

硬件问题主要表现在点不亮机子和报警。

解决方法，安装杀毒软件和优化软件。

57. FAT16/FAT32/NTFS 哪个的安全性最好，为什么？

NTFS，NTFS 文件系统是一个基于安全性的文件系统，是 Windows NT 所采用的独特的文件系统结构，它是建立在保护文件和目录数据基础上，同时照顾节省存储资源、减少磁盘占用量的一种先进的文件系统。

58. Internet 采用哪种网络协议？该协议的主要层次结构？

答：Internet 采用 TCP/IP 协议 应用层/传输层/网络层/数据链路层/物理层

59. 请问交换机和路由器各自的实现原理是什么？分别在哪个层次上面实现的？

交换机是工作在 Data link layer(数据链路层)，处理帧

路由器工作在网络层，对数据进行分组。路由器中传的是 IP 数据报。

60. 请你分别画出 OSI 的七层网络结构图和 TCP/IP 的四层结构图。

TCP/IP 层次模型共分为四层：应用层、传输层、网络层、数据链路层、物理层。

OSI 层次模型共分为七层：应用层、表示层、会话层、传输层、网络层、数据链路层、物理层。

61. 请你详细地解释一下 IP 协议的定义，在哪个层上面？主要有什么作用？TCP 与 UDP 呢？

IP 地址是 TCP/IP 网络中的主机(或称为节点)的惟一地址。IP 地址是网络层的逻辑地址。由网络地址和主机地址两部分组成。为了便于寻址和层次化地构造网络。IP 在网络层，UDP，TCP 在传输层，

62. 问：在 windows 环境下，遇到用杀毒软件都无法杀的病毒除了重作系统应该怎么办，如何关闭后台进程，msn 上不了都可能是什么问题，怎样解决。

对于病毒，可以从以下几个方面进行考虑：

- 1、使用命令“msconfig”，开启系统配置实用工具，关闭陌生的启动项，然后重新启动，观察系统的运行情况。
- 2、使用任务管理器，杀掉不正常的进程，看系统的运行情况。
- 3、使用命令“regedit”，开启注册表配置工具，清除注册表中的表项。
- 4、如果杀毒工具提供了病毒的标识，可以利用搜索引擎，查找相关的手动删除方法。
- 5、利用专杀工具进行清除。
- 6、给系统打上相应的补丁。
- 7、使用系统还原，还原系统。

关闭后台进程：

- a) 使用“管理工具”中的“服务”，将对应的服务关闭掉。
- b) 使用 gpedit.msc 来对系统进行配置。
- c) 使用服务自带的管理工具关闭。
- d) 直接在任务管理器重杀死进程。

MSN 的问题：

- 1、首先检查网络连接是否正常连通。
- 2、检查防火墙规则是否正常，是否禁止了程序或者端口。
- 3、帐号是否被他人窃取，并且修改了密码。
- 4、根据返回的错误消息，上 MSN 网站的 Messenger 子站点，查找相应的帮助，根据帮助的向导解决问题。

63. Pop3 与 IMAP 有什么区别？

pop3 和 imap 都是收取邮件的服务协议，只不过是读取的方式不同，pop3 把邮件收到本地客户端(outlook, outlook express,...) 来读，而 imap 则是打开服务器上邮件的一个快照来读，也就是说邮件并没有下

载到本地，而是存贮在邮件服务器上。

64. PIN、PIN2、PUK 是什么

个人标识号 (PIN=personal identification number)，用于保护智能卡免受误用的秘密标识代码。

PIN2 码是为特定的手机功能提供第二层保护的 PIN 码，PIN2 码是设定手机计费时使用的，其作用方式与 PIN 码相同。

PUK 码 (个人解锁码)。PUK 码是由您的网络运营商或服务供应商提供的 8 位数密码，用户无法更改

65. MVC 三层架构:

(1) GUI 应用程序的界面显示层

(2) BLL 业务逻辑层

(3) DAL 数据访问层

二十七、暂时不需的测试题目

1. 代码会审是什么?

答：对代码的一个评审的过程 发现一些最基本的错误 方式是静态的代码走读方式 在一些大型软件的设计过程中 还是必不可少的

2. 你参与了哪些方面的同行评审? 同行评审流程?

答：软件需求规格说明书 概要设计说明书 系统测试计划 系统测试用例

以测试计划为例：我制定评审计划，同时我担任主持人，对评审内容进行确认，按评审计划安排相应的资源和评审人员进行评审前的准备。评审人员确认进行评审后，按照《同行评审检查列表》提出《预审问题表》。然后进行评审会议，评审员确定问题和定义问题的严重程度，记录员记录已达成共识和未达成共识的问题，形成《评审会议记录》，记录评审结果，评审人员签字确认，记录员记录《评审问题跟踪单》，给提出者解决，最后形成评审报告。

3. 分析工作组和域的区别

一、工作组

一个网络建成后，默认所有电脑都处在网上邻居中。当电脑比较多时，则显的非常的乱。因此可以对其进行分组进行管理。例如“销售部”、“服务部”等，属于这些部分的电脑分别加入相应的组中，这样我们打开网上邻居首先看到的是这些组，进入组后看到的才是一台台具体的计算机。

对此，我们某一台电脑要加入组中，只需要右击打开我的电脑属性窗口，在“计算机名”中单击“更改”按钮，然后将“隶属于”设为“工具组”，并输入自己所属的工作组名称，如果输入的是新名称，那么则默认创建该组并加入该组，设置完毕单击“确定”按钮重新启动后生效。

二、域

从上面介绍工作组时，可以看出工作组内的每一台电脑之间是平等的，每一个人都可以自由控制自己加入哪一个组，显的比较松散。因此我们又引入了“域”的概念，即由一台电脑充当域服务器，由它控制其它电脑能否加入域。

要加入到域，那么则必须在域服务器的管理工具中打开“Active Directory 用户和计算机”，然后在左侧选择“计算机”，并在右侧右击选择“新建”中的“计算机”，填入要加入域的客户机的名称，建议使用有一定规律的英文。

在域服务器上做好后，就可以打开系统属性窗口，然后在“计算机名”中单击“更改”按钮，将“隶属于”设为“域”，然后按提示输入域名以及域用户账户即可。

如果是宽松型的，对管理要求不是特别高的可以使用工作组；如果对资源的访问有较严格的管理，那么则应建立域。