**选择2\*10**

1、以下哪个情况最能够代表出现了性能问题（D ）

A：网络延迟达到15ms以上

B：DNS没有完成解析

C：WEB服务器的可用内存降到了1GB以下

D：用户体验超过了预期的系统响应时间

2、关于C语法规则中下面那个说法是正确的（ A ）：

 A：在C语言中，允许用一个变量来存放指针

 B：分号“；”代表一段程序语句的结束

 C：/t后面的内容都是注释

 D：C语言是不区分大小写的

3、LoadRunner实现合并图的过程中一般不包括（D ）

A：叠加

B：平铺

C： 关联

D： 替换

4、影响WEB前端页面性能一般不包括下面那个（ C ）

A. 服务器数据返回延迟

B. 网络传输速率

C. 磁盘空间不够

D. 页面渲染

5、选出下列那个不是系统性能监控的指标（C ）

A：CPU利用率 B：磁盘空间大小 C： 内存空间使用率 D： 网络吞吐量

6、下面哪个LoadRunner的组件生成运行Vuser的负载？（ D ）

A: VuGen

B: Controller

C: Analysis

D: Load Generator

7、在用LoadRunner进行性能测试过程中Run-Time Setting常用的超时设置不包括（ B ）

A：HTTP-request connect timeout(sec)

B：Call to Copy of Action

C：HTTP-request receive timeout(sec)

D：Step download timeout

8、C语言数据类型不能遵循下面那个规则（C ）：

 A：char指的是字符型数据

 B：int指的是基本整型

 C：float指的是双精度实数

 D：指针是一种特殊的同时又是具有重要作用的数据类型

9、通过疲劳强度测试，最容易发现问题的问题是（ B）

A．并发用户数 B．内存泄露 C．系统安全性 D．功能错误

10、如下哪些测试场景不属于负载压力测试: （A ）

A．恢复测试 B．疲劳强度测试 C．大数据量测试 D．并发性能测试

11、下列关于软件性能测试的说法中，正确的是（ B）

A．性能测试的目的不是为了发现软件缺陷

B．压力测试与负载测试的目的都是为了探测软件在满足预定性能需求的情况下所能负担的最大压力

C．性能测试通常要对测试结果进行分析才能获得测试结论

D．在性能下降曲线上，最大建议用户数通常处于性能轻微下降区与性能急剧下降区的交界处

12、你是一个LoadRunner的专家顾问，并已分配给一个客户需要的性能测试，尚未发布的应用程序。你怎么能获得有关应用程序的预期负载的信息？（ D ）

A：个人估计应用程序的预期负载

B：从Web日志中获取必要的信息

C： 根据应用程序的数据库，以确定预期的负载

D： 与业务专家协商，以确定预期的负载

13、选出下列不需要LoadRunner关注的Linux进程类型（B ）

A：交互进程 B：独立进程 C：批处理进程 D： 监控进程

14、通过对于场景分析你发现：Vuser每秒http点击率持续增加，突然变得平坦。可能是什么原因造成的？（ C ）

 A. 带宽问题

B. 数据库服务器的问题

C. Web服务器的连接问题

D. 磁盘空间不足问题

**填空：1\*10=10**

1. 脚本的参数化时数据更新的方式为 每次出现、 每次迭代和 一次。
2. 场景的设计时有基于手动的场景设计和面向目标的场景设计两种方式。
3. 突破参数的百条限制要修改config文件中的Max VisibleLines=200。
4. 用户在执行两个连续操作期间等待的时间称为思考时间。

5、LoadRunner主要包含3个方面的应用：HP Virtual User Generator用于创建脚本、HP Virtual Controller用于设计并运行场景和HP Virtual Analysis用于分析运行后的场景结果。

1. 脚本的参数化时数据分配的方法为 顺序、 随机和 唯一。
2. 突破Controller可用脚本50条限制修改注册表中的max\_num\_of\_scripts=100。

**名词解释：5\*4=20**：

1. 性能测试：是为描述测试对象与性能相关的特征并对其进行评价，而实施和执行的一类测试，它主要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。
2. 白盒测试：又称为结构测试、逻辑驱动册数或基于程序本身的测试，它着重于程序的内部结构及算法，通常不关心功能与性能指标，是对软件过程性细节做细致的检查。
3. 并发测试：测试多个用户同时访问同一个应用、同一个模块或者数据记录时是否存在死锁或者其他性能问题，几乎所有的性能测试都会涉及一些并发测试。
4. 磁盘数据分段：数据分散存储在几个硬盘上。数据的第一段放在硬盘0，第2段放在硬盘1……直至达到硬盘链中的最后一个硬盘，然后下一个逻辑段将放在硬盘0，再下一个逻辑段放在硬盘1，如此循环直至完成写操作。
5. 软件测试：是根据软件开发各个阶段的规格说明和程序的内部结构而精心设计的多组测试用例（即输入数据及其预期的输出结果），并利用这些测试用例运行程序以发现错误的过程。
6. 黑盒测试：也可以称为功能测试、数据驱动测试或基于规格说明的测试。测试者不了解程序的内部情况，只知道程序的输入、输出和系统的功能，这是从用户的角度对程序进行的测试。
7. 可靠性测试：通过给系统加载一定的业务压力的情况下，运行一段时间，检查系统是否稳定。因为运行时间较长，通常可以测试出系统是否有内存泄露等问题。
8. 负载均衡：是由多台服务器以对称的方式组成一个服务器集合，每台服务器都具有等价的地位，都可以单独对外提供服务而无须其他服务器的辅助。通过某种负载分担技术，将外部发送来的请求均匀分配到对称结构中的某一条服务器上，而接收到请求的服务器独立地回应客户的请求。

**简答题：5题 30分**

1. 测试自动化和手工测试的区别和联系？

手工测试是传统的测试方法，由测试人员手工编写[测试用例](http://www.so.com/s?q=%E6%B5%8B%E8%AF%95%E7%94%A8%E4%BE%8B&ie=utf-8&src=internal_wenda_recommend_textn)，缺点在于测试工作量大，重复多，[回归测试](http://www.so.com/s?q=%E5%9B%9E%E5%BD%92%E6%B5%8B%E8%AF%95&ie=utf-8&src=internal_wenda_recommend_textn)难以实现；

[自动化测试](http://www.so.com/s?q=%E8%87%AA%E5%8A%A8%E5%8C%96%E6%B5%8B%E8%AF%95&ie=utf-8&src=internal_wenda_recommend_textn)利用[软件测试工具](http://www.so.com/s?q=%E8%BD%AF%E4%BB%B6%E6%B5%8B%E8%AF%95%E5%B7%A5%E5%85%B7&ie=utf-8&src=internal_wenda_recommend_textn)自动实现全部或者部分测试工作：管理、设计、执行和报告，自动化测试节省大量的测试开销，并能够完成一些手工测试无法实现的测试。

自动化测试是对手工测试的一种补充，自动化测试不可能完全替代手工测试，因为很多数据的正确性、界面是否美观、[业务逻辑](http://www.so.com/s?q=%E4%B8%9A%E5%8A%A1%E9%80%BB%E8%BE%91&ie=utf-8&src=internal_wenda_recommend_textn)的满足程度等都离不开测试人员的人工判断。而仅仅依赖手工测试的话，则会让测试过于低效，尤其是回归测试的重复工作量对测试人员造成了巨大的压力。

1. 系统调优由易到难的先后顺序是什么？
* 硬件问题
* 网络问题
* 应用服务器、数据库等配置问题
* 源代码、数据库脚本问题
* 系统构架问题
1. 为什么要使用关联，使用关联可以带来哪些方便？

首先，它可以生成动态的数据；其次，我们可以将冗长的数据参数化，通过应用关联技术，有效地减少代码的大小，这样不仅代码量会减少，脚本层次看起来更清晰明了

1. 如何利用“拐点”分析法来定位系统的性能瓶颈？

（"拐点分析"方法是一种利用性能计数器曲线图上的拐点进行性能分析的方法。它的基本思想就是）

性能产生瓶颈的主要原因就是因为某个资源的使用达到了极限，此时表现为随着压力的增大，系统性能却出现急剧下降，这样就产生了"拐点"现象。当得到"拐点"附近的资源使用情况时，就能定位出系统的性能瓶颈。

（"拐点分析"方法举例，如系统随着用户的增多，事务响应时间缓慢增加，当用户数达到100个虚拟用户时，系统响应时间急剧增加，表现为一个明显的"折线"，这就说明了系统承载不了如此多的用户做这个事务，也就是存在性能瓶颈。

）

1. 如何发现性能测试的规律？

根据测试用例按照由轻到重，对系统逐渐施压的过程执行性能测试，发现性能测试模型。根据模型的重要性能指标如：响应时间、吞吐量、资源利用率、最大用户数找到性能测试的规律。

1. 测试用例设计的方法有哪几种？
* 等价类划分法
* 边界值分析法
* 判定表法
* 因果图法
* 错误推测法
* 状态转换测试法
* 功能图分析
* 正交实验设计
1. 测试场景的设计要注意什么？

测试场景的设计应该注意：集合点、集合点策略、迭代次数、思考时间、参数化取值方式等性能指标。在进行场景设计时，如果在脚本中应用了集合点函数，要确认在Controller应用中启用了集合点并设置了对应的策略。同时关注运行时设置、相关的迭代、思考时间等是否与预期的设置的一致。

1. 比较常见的负载均衡技术有哪几种？
* 基于DNS的负载均衡
* 反向代理负载均衡
* 基于NAT的负载均衡技术
1. 如何结合企业特点进行性能测试？
* 大型企业、大型项目的应用策略。

解决方案一：构建自己的性能测试团队。

解决方案二：专业性能测试机构为系统测试。

* 中型企业、中型项目的应用策略。

解决方案一：临时组建性能测试团队。

解决方案二：自行编写测试程序。

* 小型企业、小型项目的应用策略。

解决方案一：临时组建性能测试团队。

解决方案二：专业性能测试机构为系统测试。

**综合应用题：3题 20分**

1. 如何实现ping ip 的功能

在Vugen中选择“web（http/html）”协议

#include “web\_api.h”

Action()

{

 System(“ping 192.168.4.236”);

Return 0;

}

1. 如何设置用户名和口令

直接弹出对话框

Vnser\_init()

{

 web\_set\_user(“tony”,”for you”,”barton:8080”);

 web\_url(“web\_url”,

“url=http://www.bintonx.com/auth/index.jsp”,

“TargetFrame=”,

“Resource=0”,

“Referer=”,

LAST);

return 0;

}

1. 系统性能改进意见

系统主要存在CPU利用率过高的问题以及用户在进行业务处理时登录失败的问题

Connection reset by peer的原因主要原因

* 服务器的并发连接数超过了其承载量，服务器会将其中的一些连接断掉
* 客户关掉了浏览器，服务器仍在给客户端发送数据

以上问题的解决

* 调整服务器的应用配置，应用连接池、设置更多的连接数
* 设定一些思考时间，留给服务器一些处理时间

CPU利用率过高的问题

* 查找是否系统在启动同时开启了多个与本系统无关的应用程序，导致测试前被占用很多内存和CPU利用率
* 如果CPU不能满足测试需要，考虑更换频率更高的CPU
* 对相关程序代码、数据库语句进行改良，减少对CPU利用率

内存的问题

* 查看在运行期间是否系统启动了一些其他的服务或者应用程序，致使可用内存数量减少
* 可用监控内存是否存在泄露情况
1. 如何手动转换字符串编码

Action（）

{

int rc=0;

rc=lr\_convert\_string\_encoding(“hello word”,LR\_EC\_SYSTEM\_LOCALE,LR\_ENC\_UNICODE,”strUnicode”);

if(rc<0)

{

 lr\_output\_message(“转换\”hello word\”失败”!”);

}

rc=lr\_convert\_string\_encoding(“love LR”,LR\_EC\_SYSTEM\_LOCALE,LR\_ENC\_UNICODE,”strUnicode”);

if(rc<0)

{

 lr\_output\_message(“转换\”love LR\”失败”!”);

}

return 0;

}

1. 如何使用C函数进行脚本跟踪

Action()

{

LPCSTR l=”hello”;

ci\_set\_debug(ci\_this\_connext,1,1);

lr\_output\_message(“%s”,1);

ci\_set\_debug(ci\_this\_connext,0,0);

return 0;

}

1. 为何无法与load generator通信

检查以下两项

* TCP/IP连接。

首先，使用ping命令确保Controller和Load Generator是可以在网络上相互ping通的，这里假设本机为一个负载生成器机器（即：Load Generator）ping主控机（Controller）同时也要保证主控机可以ping通负载机。

如果ping命令无响应或超时失败，说明无法识别计算机名。要解决此问题，编辑Windows系统目录下边的hosts文件，编辑该文件添加一行包含IP地址和名称的信息

* Load Generator 连接。

要验证Load Generator连接，单击"Connect"按钮确保能够连接到每个远程Load Generator。如果连接成功，状态将变更为"Ready"，否则将显示"Failed"。如果场景使用多个域，那么Controller在与Load Generator通信时可能会产生问题。发生这种问题的原因是Controller默认使用了Load Generator简短名称（不包含域）。要解决此问题，必须指示Controller确定Load Generator的全名（包含域）。

请您修改主控机（Controller）的"C:\Program Files\HP\LoadRunner\dat\miccomm.ini"文件，将"LocalHostNameType=1"变更为"LocalHostNameType=0" （"0"表示尝试使用完整计算机名、"1"表示使用简短计算机名，这是默认值，"2"代表IP地址）