

在程序执行过程中, Cache 与主存的地址映像由 (1)。

- (1) A. 硬件自动完成 B. 程序员调度
C. 操作系统管理 D. 程序员与操作系统协同完成

【答案】A

【解析】 本题考查计算机系统基础知识。

Cache 的工作是建立在程序与数据访问的局部性原理上。即经过对大量程序执行情况的结果分析：在一段较短的时间间隔内程序集中在某一较小的内存地址空间执行，这就是程序执行的局部性原理。同样，对数据的访问也存在局部性现象。为了提高系统处理速度才将主存部分存储空间中的内容复制到工作速度更快的 Cache 中，同样为了提高速度的原因，Cache 系统都是由硬件实现的。

指令寄存器的位数取决于(2)。

- (2) A. 存储器的容量 B. 指令字长 C. 数据总线的宽度 D. 地址总线的宽度

【答案】B

【解析】 本题考查计算机系统基础知识。

指令寄存器是 CPU 中的关键寄存器，其内容为正在执行的指令，显然其位数取决于指令字长。

若计算机存储数据采用的是双符号位(00 表示正号、11 表示负号)，两个符号相同的数相加时，如果运算结果的两个符号位经(3)运算得 1，则可断定这两个数相加的结果产生了溢出。

- (3) A. 逻辑与 B. 逻辑或 C. 逻辑同或 D. 逻辑异或

【答案】 D

【解析】 本题考查计算机系统基础知识。

当表示数据时规定了位数后，其能表示的数值的范围就确定了，在两个数进位相加运算的结果超出了该范围后，就发生了溢出。在二进制情况下，溢出时符号位将变反，即两个正数相加，结果的符号位是负数，或者两个负数相加，结果的符号位是正数。采用两个符号位时，溢出发生后两个符号位就不一致了，这两位进行异或的结果一定为 1。

某指令流水线由 4 段组成, 各段所需要的时间如下图所示。连续输入 8 条指令时的吞吐

率(单位时间内流水线所完成的任务数或输出的结果数)为(4)。



- (4) A. $8/56 \Delta t$ B. $8/32 \Delta t$ C. $8/28 \Delta t$ D. $8/24 \Delta t$

【答案】C

【解析】 本题考查计算机系统基础知识。

流水线的吞吐率指的是计算机中的流水线在特定的时间内可以处理的任务或输出数据的结果的数量。流水线的吞吐率可以进一步分为最大吞吐率和实际吞吐率。该题 0 中要求解的是实际吞吐率,以流水方式执行 8 条指令的指令行时间是 $28 \Delta t$,因此吞吐率为 $8/28 \Delta t$ 。

(5)不是 RISC 的特点。

- (5) A. 指令种类丰富 B. 高效的流水线操作
C. 寻址方式较少 D. 硬布线控制

【答案】A

【解析】 本题考查计算机系统基础知识。

RISC(Reduced Instruction Set Computer,精简指令集计算机)的主要特点是重叠寄存器窗口技术;优化编译技术。RISC 使用了大量的寄存器,如何合理分配寄存器、提高寄存器的使用效率及减少访存次数等,都应通过编译技术的优化来实现;超流水及超标量技术。为了进一步提高流水线速度而采用的技术;硬布线逻辑与微程序相结合在微程序技术中。

程序运行过程中常使用参数在函数(过程)间传递信息,引用调用传递的是实参的(6)。

- (6) A. 地址 B. 类型 C. 名称 D. 值

【答案】A

【解析】 本题考查程序语言基础知识。

进行函数调用时,常需要在调用环境中的数据传递给被调用函数,作为输入参数由被调用函数处理,基本的调用方式为值调用(或传值调用)和引用调用。其中,值调用方式下是将实参的值单向地传递给被调用函数的形参,引用调用方式下通过将实参的地址传递给形参,在被调用函数中通过指针实现对实参变量数据的间接访问和修改,从而达到将修改后的值“传回来”的效果。

(7)不是单元测试主要检查的内容。

- (7) A. 模块接口 B. 局部数据结构 C. 全局数据结构 D. 重要的执行路径

【答案】C

【解析】本题考查软件测试的基础知识。

单元测试又称为模块测试，是针对软件设计的最小单元—程序模块，进行正确性检验的测试。其目的在于发现各模块内不可能存在的各种问题和错误。单元测试需要从程序的内部结构出发设计测试用例。模块可以单独进行单元测试。单元测试测试以下几个方面：模块接口、局部数据结构、执行路径、错误处理和边界。

PKI 体制中，保证数字证书不被篡改的方法是(8)。

- (8) A. 用 CA 的私钥对数字证书签名 B. 用 CA 的公钥对数字证书签名
C. 用证书主人的私钥对数字证书签名 D. 用证书主人的公钥对数字证书签名

【答案】A

【解析】本题考查 PKI 体制。

PKI 体制中，为保障数字证书不被篡改而且要发送到证书主人手中，需要用 CA 的私钥对数字证书签名，防伪造，不可抵赖。

下列算法中，不属于公开密钥加密算法的是(9)。

- (9) A. ECC B. DSA C. RSA D. DES

【答案】D

【解析】本题考查加密算法的基础知识。

常用的加密算法依据所使用的密钥数分为单钥和双钥加密体制，也称私钥和公钥加密算法。ECC、DSA 和 RSA 都属于公开密钥加密算法，DES 是典型的私钥加密体制。

为说明某一问题，在学术论文中需要引用某些资料。以下叙述中，(10)是不正确的。

- (10) A. 既可引用发表的作品，也可引用未发表的作品
B. 只能限于介绍、评论作品
C. 只要不构成自己作品的主要部分，可适当引用资料
D. 不必征得原作者的同意，不需要向他支付报酬

【答案】A

【解析】 本题考查知识产权方面的基础知识。

选项 A “既可引用发表的作品，也可引用未发表的作品” 的说法显然是错误的。因为，为说明某一问题，在学术论文中需要引用某些资料必须是已发表的作品，但只能限于介绍、评论作品，只要不构成自己作品的主要部分，可适当引用资料，而不必征得原作者的同意，不需要向他支付报酬。

以下作品中，不适用或不受著作权法保护的是(11)。

- (11)A. 某教师在课堂上的讲课
B. 某作家的作品《红河谷》
C. 最高人民法院组织编写的《行政诉讼案例选编》
D. 国务院颁布的《计算机软件保护条例》

【答案】 D

【解析】 本题考查应试者知识产权方面的基础知识。

选项 D “国务院颁布的《计算机软件保护条例》” 的说法显然是错误的。因为，国务院颁布的《计算机软件保护条例》是国家为了管理需要制定的政策法规，故不适用著作权法保护。

已知文法 $G: S \rightarrow A0 \mid B1, A \rightarrow S1 \mid 1, B \rightarrow S0 \mid 0$, 其中 S 是开始符号。从 S 出发可以推导出(12)。

- (12)A. 所有由 0 构成的字符串
B. 所有由 1 构成的字符串
C. 某些 0 和 1 个数相等的字符串
D. 所有 0 和 1 个数不同的字符串

【答案】 C

【解析】 本题考查程序语言基础知识。

用文法表示语言的语法规则时，推导是产生语言句子的基本方式。以题目中的文法为例，推导出 1010 的过程为 $S \Rightarrow A0 \Rightarrow S10 \Rightarrow A010 \Rightarrow 10 \mid 0$ ，推导出 0110 的过程为 $S \Rightarrow A0 \Rightarrow S10 \Rightarrow B110 \Rightarrow 0110$ ，对于 0000、1111、1100、0011 等则推导不出。因为由 S 先推导出 A0 后，再去推导 A 则必然产生一个与 0 相邻(在 0 的左边)的 1，而由 S 先推导出 B1，则下一步必然要推导出一个与 1 相邻(在 1 的左边)的 0。这保证了当 1 出现时，马上就会出现 0，或者反之，且 0 和 1 的距离很近。分析更多的例子发现，仅有“某些 0 和 1 个数相等的字符串”是正确的。

算术表达式 $a+(b-c)*d$ 的后缀式是(13) (-、+、*表示算术的减、加、乘运算，运算符的优先级和结合性遵循惯例)。

(13) A. $b\ c\ -\ d\ *\ a\ +$

B. $a\ b\ c\ -\ d\ *\ +$

C. $a\ b\ +\ c\ -\ d\ *$

D. $a\ b\ c\ d\ -\ *\ +$

【答案】B

【解析】本题考查程序语言基础知识。

后缀式的特点是将运算符写在运算数的后面。对于表达式，其计算次序是相减、相乘、相加，其后缀式为“ $a\ b\ c\ -\ d\ *\ +$ ”。

将高级语言程序翻译为机器语言程序的过程中，常引入中间代码，其好处是(14)。

(14) A. 有利于进行反编译处理

B. 有利于进行与机器无关的优化处理

C. 尽早发现语法错误

D. 可以简化语法和语义分析

【答案】B

【解析】本题考查程序语言基础知识。

“中间代码”是一种简单且含义明确的记号系统，可以有若干种形式，它们的共同特征是与具体的机器无关，此时所作的优化一般建立在对程序的控制流和数据流分析的基础之上，与具体的机器无关。

假设某公司营销系统有营销点关系 S(营销点，负责人姓名，联系方式)、商品关系 P(商品名，条形码，型号，产地，数量，价格)，其中，营销点唯一标识 S 中的每一 a 元组。每个营销点可以销售多种商品，每一种商品可以由不同的营销点销售。关系 S 和 P 的主键分别为(15)，S 和 P 的之间联系类型属于(16)。

(15) A. 营销点、商品名

B. 营销点、条形码

C. 负责人姓名、商品名

D. 负责人姓名、条形码

(16) A. 1: 1

B. 1: n

C. n: 1

D. n: m

【答案】B D

【解析】本题考查对数据库基本概念和 SQL 语言的掌握。

(15) 根据题意，营销点唯一标识 S 中的每一个元组，所以营销点可以作为 S 的主键，商品关系 P(商品名，条形码，型号，产地，数量，价格)中的条形码属性可以作为该关系的主键，因为，条形码是由宽度不同、反射率不同的条和空，按照一定的编码规则(码制)编制成

的，用以表达一组数字或字母符号信息的图形标识符。利用条形码可以标出商品的生产国、制造厂家、商品名称、生产日期、图书分类号、由 P 件起止地点、类别、日期等信息，所以，条形码在商品流通、图书管理、邮电管理、银行系统等许多领域都得到了广泛的应用。显然，本题的正确答案是“营销点、条形码”。

(16) 根据题意“每个营销点可以销售多种商品,每一种商品可以由不同的营销点销售”,故 S 和 P 的之间联系类型属于 n:m。

若有关系 R(A, B, C, D, E)和 S(B, C, F, G), 则 R 与 S 自然联结运算后的属性列有 17 个, 与表达式 $\pi_{1, 3, 6, 7}(\sigma_{3 < 6} (R \bowtie S))$ 等价的 SQL 语句如下:

SELECT (18) FROM (19) WHERE (20);

- | | | | |
|---|------|---------------------------|---------|
| (17) A. 5 | B. 6 | C. 7 | D. 9 |
| (18) A. A, R, C, F, G | | B. A, C, S, B, S, F | |
| C. A, C, S, B, S, C | | D. R, A, R, C, S, B, S, C | |
| (19) A. R | B. S | C. RS | D. R, S |
| (20) A. R. B=S. B AND R. C=S. C AND R. C<S. B | | | |
| B. R. B=S. B AND R. C=S. C AND R. C<S. F | | | |
| C. R. B=S. B OR R. C=S. C OR R. C<S. B | | | |
| D. R. B=S. B OR R. C=S. C OR R. C<S. F | | | |

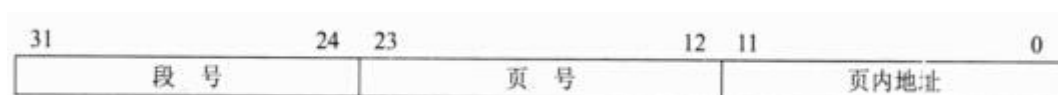
【答案】 C A D B

【解析】 本题考查关系代数运算与 SQL 查询方面的基础知识。

在 $\pi_{1, 3, 6, 7}(\sigma_{3 < 6}(R \bowtie S))$ 中, 自然联结 $R \bowtie S$ 运算后去掉右边重复的属性列名 S.B、S.C 后为: R.A、R.B、R.C、R.D、R.E、S.F 和 S.G, 空 (17) 的正确答案为 7。 $\pi_{1, 3, 6, 7}(\sigma_{3 < 6}(R \bowtie S))$ 的含义是从 $R \bowtie S$ 结果集中选取 R.C < S.F 的元组, 再进行 R.A、R.C、S.F 和 S.G 投影, 因此, 空 (18) 的正确答案为选项 A。显然, 空 (19) 的答案为 R, S。

空(20)的正确答案为选项 B。因为，自然联结 $R \bowtie S$ 需要用条件“WHERE $R.B=S.B$ AND $R.C=S.C$ ”来限定，选取运算 $\sigma_{3<6}$ 需要用条件“WHERE $R.C<S.F$ ”来限定。

假设段页式存储管理系统中的地址结构如下图所示, 则系统中(21)。



(21)A. 页的大小为 4K，每个段的大小均为 4096 个页，最多可有 255 个段

B. 页的大小为 4K，每个段最大允许有 4096 个页，最多可有 256 个段

C. 页的大小为 8K，每个段的大小均为 2048 个页，最多可有 128 个段

D. 页的大小为 8K，每个段最大允许有 2048 个页，最多可有 128 个段

【答案】B

【解析】本题考查操作系统页式存储管理方面的基础知识。

从图中可见，页内地址的长度是 12 位， $2^{12}=4096$ ，即 4K；页号部分的地址长度是 12 位，每个段最大允许有 4096 个页；段号部分的地址长度是 8 位， $2^8=256$ ，最多可有 256 个段。

假设系统中有三类互斥资源 R1、R2 和 R3，可用资源数分别为 10、5 和 3。在 T0 时刻系统中有 P1、P2、P3、P4 和 P5 五个进程，这些进程对资源的最大需求量和已分配资源数如下表所示，此时系统剩余的可用资源数分别为(22)。如果进程按(23)序列执行，那么系统状态是安全的。

进程 \ 资源	最大需求量			已分配资源数		
	R1	R2	R3	R1	R2	R3
P1	5	3	1	1	1	1
P2	3	2	0	2	1	0
P3	6	1	1	3	1	0
P4	3	3	2	1	1	1
P5	2	1	1	1	1	0

(22)A. 1、1 和 0

B. 1、1 和 1

C. 2、1 和 0

D. 2、0 和 1

(23)A. P1→P2→P4→P5→P3

B. P5→P2→P4→P3→P1

C. P4→P2→P1→P5→P3

D. P5→P1→P4→P2→P3

【答案】D B

【解析】

试题(22)的正确答案是 D。因为，初始时系统的可用资源数分别为 10、5 和 3。在 T0 时刻已分配资源数分别为 8、5 和 2，因此系统剩余的可用资源数分别为 2、0 和 1。

试题(23)的正确答案是 B。安全状态是指系统能按某种进程顺序(P1, P2, ..., Pn)，来为每个进程 Pi 分配其所需的资源，直到满足每个进程对资源的最大需求，使每个进程都可以顺利完成。如果无法找到这样的安全序列，则称系统处于不安全状态。

本题进程的执行序列已经给出，我们只需将四个选项按其顺序执行一遍，便可以判断出现死锁的三个序列。

资源 进程	最大需求量 R1 R2 R3			已分配资源数 R1 R2 R3			尚需资源数 R1 R2 R3		
P1	5	3	1	1	1	1	4	2	0
P2	3	2	1	2	1	0	1	1	1
P3	6	1	1	3	1	0	3	0	1
P4	3	3	2	1	1	1	2	2	1
P5	2	1	1	1	1	0	1	0	1

P1→P2→P4→P5→P3 是不安全的序列。因为在该序列中，进程 P1 先运行，P1 尚需资源数为(4, 2, 0)，假设将资源 R1 分配 2 台给进程 P1，则系统剩余的可用资源数为(0, 0, 1)，将导致系统所有的进程都不能作上能完成标志“True”。

P5→P2→P4→P3→P1 是安全的序列。因为所有的进程都能作上能完成标志“True”，如下表所示。

资源 进程	可用资源数 R1 R2 R3			已分配资源数 R1 R2 R3			尚需资源数 R1 R2 R3			可用+已分 R1 R2 R3			能否完成标志
P5	2	0	1	1	1	0	1	0	1	3	1	1	True
P2	3	1	1	2	1	0	1	1	1	5	2	1	True
P4	5	2	1	1	1	1	2	2	1	6	3	2	True
P3	6	3	2	3	1	0	3	0	1	9	4	2	True
P1	9	4	2	1	1	1	4	2	0	10	5	3	True

P5→P2→P4→P3→P1 具体分析如下：

①进程 P5 运行，系统剩余的可用资源数为(2, 0, 1)，P5 尚需资源数为(1, 0, 1)，系统可进行分配，故进程 P5 能作上能完成标志“True”，释放 P5 占有的资源数(1, 1, 0)，系统可用资源数为(3, 1, 1)。

②进程 P2 运行，系统剩余的可用资源数为(3, 1, 1)，P2 尚需资源数为(1, 1, 1)，系统可进行分配，故进程 P2 能作上能完成标志“True”，释放 P2 占有的资源数(2, 1, 0)，系统可用资源数为(5, 2, 1)。

③进程 P4 运行，系统剩余的可用资源数为(5, 2, 1)，P4 尚需资源数为(2, 2, 1)，系统可进行分配，故进程 P4 能作上能完成标志“True”，释放 P4 占有的资源数(1, 1, 1)，系统可用资源数为(6, 3, 2)。

④进程 P3 运行，系统剩余的可用资源数为(6, 3, 2)，P3 尚需资源数为(3, 0, 1)，系统可进行分配，故进程 P3 能作上能完成标志“True”，释放 P3 占有的资源数(3, 1, 0)，系统可用资源数为(9, 4, 2)。

⑤进程 P1 运行，系统剩余的可用资源数为(9, 4, 2)，P1 尚需资源数为(4, 2, 0)，系统可进行分配，故进程 P1 能作上能完成标志“True”，释放 P1 占有的资源数(1, 1, 1)，系统可

用资源数为(10, 5, 3)。

P4→P2→P1→P5→P3 是不安全的序列。因为在该序列中，进程 P4 先运行，P4 尚需资源数为(2, 2, 1)，假设将资源 R1 分配 2 台给进程 P4, 则系统剩余的可用资源数为(0, 0, 1)，将导致系统所有的进程都不能作上能完成标志 “True”。

P5→P1→P4→P2→P3 是不安全的序列。因为在该序列中，进程 P5 先运行，系统剩余的可用资源数为(2, 0, 1), P5 尚需资源数为(1, 0, 1)，系统可进行分配，故进程 P5 能作上能完成标志 “True”，释放 P5 占有的资源数(1, 1, 0)，系统可用资源数为(3, 1, 1)。进程 P1 运行，P1 尚需资源数为(4, 2, 0)，假设将资源 R1 分配 3 台给进程 P1，则系统剩余的可用资源数为(0, 1, 1)，将导致系统中的进程 P1、P2、P3 和 P4 都不能作上能完成标志 “True”。

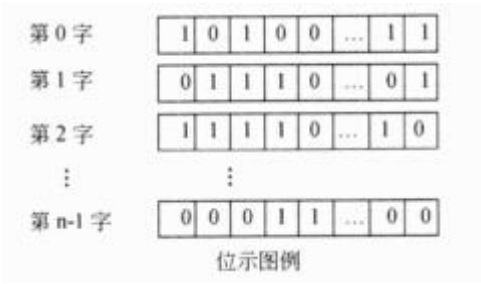
某文件管理系统采用位示图(bitmap)记录磁盘的使用情况。如果系统的字长为 32 位，磁盘物理块的大小为 4MB，物理块依次编号为：0、1、2、…，位示图字依次编号为：0、1、2、…，那么 16385 号物理块的使用情况在位示图中的第(24)个字中描述:如果磁盘的容量为 1000GB，那么位示图需要(25)个字来表示。

- (24)A. 128 B. 256 C. 512 D. 1024
- (25)A. 1200 B. 3200 C. 6400 D. 8000

【答案】C D

【解析】本题考查操作系统文件管理方面的基础知识。

文件管理系统是在外存上建立一张位示图(bitmap)，记录文件存储器的使用情况。每一位对应文件存储器上的一个物理块，取值 0 和 1 分别表示空闲和占用，如下图所示。



由于系统中字长为 32 位，所以每个字可以表示 32 个物理块的使用情况。又因为文件存储器上的物理块依次编号为：0、1、2、…，位示图表示物理块的情况如下，从下图可见，16385 号物理块应该在位示图的第 512 个字中描述。

如果要使得两个 IPv6 结点可以通过现有的 IPv4 网络进行通信，则应该使用隧道技术；如果要使得纯 IPv6 结点可以与纯 IPv4 结点进行通信，则需要使用翻译技术。

POP3 协议采用(29)模式进行通信，当客户机需要服务时，客户端软件与 POP3 服务器建立(30)连接。

- (29)A. Browser/Server

B. Client/Server

C. PeertoPeer

D. PeertoServer
- (30)A. TCP

B. UDP

C. PHP

D. IF

【答案】B A

【解析】

POP3 协议采用 C/S 模式进行通信，POP3 需要 TCP 连接的支持，当客户机需要服务时，客户端软件与 POP3 服务器建立 TCP 连接。

采用插入排序算法对 n 个整数排序，其基本思想是：在插入第 i 个整数时，前 i-1 个整数已经排好序，将第 i 个整数依次和第 i-1, i-2, …个整数进行比较，找到应该插入的位置。现采用插入排序算法对 6 个整数 {5, 2, 4, 6, 1, 3} 进行从小到大排序，则需要进行(31)次整数之间的比较。对于该排序算法，输入数据具有(32)特点时，对整数进行从小到大排序，所需的比较次数最多。

- (31)A. 9

B. 10

C. 12

D. 13
- (32)A. 从小到大

B. 从大到小

C. 所有元素相同

D. 随机分布

【答案】C B

【解析】

采用插入排序算法对 6 个整数 {5, 2, 4, 6, 1, 3} 进行从小到大排序的过程如表所示。

序号	已排序部分	未排序部分	说 明
0	{5}	{2,4,6,1,3}	初始时
1	{2,5}	{4,6,1,3}	将 2 插入序列{5}时，需要将 2 与 5 比较 1 次，完成插入
2	{2,4,5}	{6,1,3}	将 4 插入序列{2,5}时，需要将 4 与 5、2 分别比较 1 次，完成插入
3	{2,4,5,6}	{1,3}	将 6 插入序列{2,4,5}时，需要将 6 与 5 比较 1 次，完成插入
4	{1,2,4,5,6}	{3}	将 1 插入序列{2,4,5,6}时，需要将 1 与 6、5、4、2 分别比较 1 次，完成插入
5	{1,2,3,4,5,6}	{}	将 3 插入序列{1,2,4,5,6}时，需要将 1 与 6、5、4、2 分别比较 1 次，完成插入

综上，元素间共比较 12 次。从上表中的第 4 步可看出，当待插入的元素比已排序部分的所有元素都要小时，需要比较和移动的元素最多，因此当输入数据序列正好从大到小排列，而需要将其从小到大排序时，元素间的比较次数最多。

软件工程的基本目标是(33)。

- (33)A. 消除软件固有的复杂性 B. 开发高质量的软件
C. 努力发挥开发人员的创造性潜能 D. 推动软件理论和技术的发展

【答案】B

【解析】本题考查软件工程的基础知识。

软件工程是指导计算机软件开发和维护的工程学科，其最根本的目标就是开发出高质量的软件并有效地维护它。

(34)过程模型明确地考虑了开发中的风险。

- (34)A. 瀑布 B. 快速原型 C. V D. 螺旋

【答案】D

【解析】本题考查软件过程模型的基础知识。

软件过程是软件生存周期中的一系列相关活动，即用于开发和维护软件及相关产品的一系列活动。瀑布模型从一种非常同层的角度描述了软件开发过程中进行的活动，并且提出了要求开发人员经过的事件序列。原型模型允许开发人员快速地构造整个系统或系统的一部分以理解或澄清问题。V 模型是瀑布模型的变种，它说明测试活动是如何与分析 and 设计相联系的。螺旋模型把开发活动和风险管理结合起来，以将风险减到最小并控制风险，在该过程模型中，风险被明确地提了出来。

在开发一个字处理软件时，首先快速发布了一个提供基本文件管理、编辑和文档生成功能的版本，接着发布提供更完善的编辑和文档生成功能的版本，最后发布提供拼写和语法检查功能的版本，这里采用了(35)过程模型。

- (35)A. 瀑布 B. 快速原型 C. 增量 D. 螺旋

【答案】C

【解析】本题考查软件过程模型的基础知识。

瀑布模型从一种非常高层的角度描述了软件开发过程中进行的活动，并且提出了要求开

发人员经过的事件序列。原型模型允许开发人员快速地构造整个系统或系统的一部分以理解或澄清问题。增量开发是把软件产品作为一系列的增量构件来设计、编码、集成和测试，系统功能在增量中不断完善或者增加。螺旋模型把开发活动和风险管理结合起来，以将风险减到最小并控制风险。

在各种不同的软件需求中，(36)描述了用户使用产品必须要完成的任务，可以用 UML 建模语言的(37)表示。

- | | | | |
|-------------|----------|---------|---------|
| (36)A. 功能需求 | B. 非功能需求 | C. 过程约束 | D. 设计约束 |
| (37)A. 用例图 | B. 类图 | C. 状态图 | D. 序列图 |

【答案】A A

【解析】本题考查软件需求的基础知识。

功能需求描述了用户使用产品必须要完成的任务，UML 的用例图可以用来对功能需求建模。非功能需求描述软件解决方案必须具有的质量特性，如性能、安全等。过程约束是对用于构建系统的技术和资源的限制。设计约束是已经做出的设计决策或限制问题解决方案集的设计决策。

在结构化分析方法中，数据流图描述数据在系统中如何被传送或变换，反映系统必须完成的逻辑功能，用于(38)建模。在绘制数据流图时，(39)。

- | | | | |
|-----------|-------|-------|-------|
| (38)A. 数据 | B. 功能 | C. 结构 | D. 行为 |
|-----------|-------|-------|-------|
- (39)A. 每个加工至少有一个输入数据流，可以没有输出数据流
- B. 不允许一个加工有多条数据流流向另一个加工
- C. 不允许一个加工有两个相同的输出数据流流向两个不同的加工
- D. 必须保持父图与子图平衡

【答案】B D

【解析】本题考查软件工程需求分析的基础知识。

在结构化分析方法中，存在几种重要的模型：数据流图描述数据在系统中如何被传送或变换，反映系统必须完成的逻辑功能，用于功能建模；实体-联系图用于数据建模；状态-迁移图用于行为建模。

数据流图包含外部实体(数据源)、加工、数据存储和数据流。每个加工必须既有输入流又有输出流；一个加工可以有多个数据流流向另一个加工，一个加工可以有两个相同的输出

数据流流向两个不同的加工。在将父图分解为子图时，必须要保持数据流的平衡。

确定采用哪种软件体系结构是在(40)阶段进行的。

- (40)A. 需求分析 B. 概要设计 C. 详细设计 D. 软件实现

【答案】B

【解析】本题考查软件设计的基础知识。

简单地说，软件体系结构就是描述软件的部分及其部分之间的关系。软件体系结构应该尽早确定，一般在软件开发的概要设计阶段进行。

以下关于模块化的叙述中，正确的是(41)。

- (41)A. 每个模块的规模越小越好，这样开发每个模块的成本就可以降低了
B. 每个模块的规模越大越好，这样模块之间的通信开销就会降低了
C. 应具有高内聚和低耦合的性质
D. 仅适用于结构化开发方法

【答案】C

【解析】本题考查软件设计的基础知识。

模块化设计是软件设计的一个重要方面，不仅仅适用于结构化开发方法，面向对象开发方法也适用。模块的规模应该适中，过小可能会增加耦合性，过大可能会降低内聚性。在分解模块时，应该充分考虑模块之间的，低耦合和模块内的高内聚的性质。

当一个模块直接使用另一个模块的内部数据，或者通过非正常入口转入另一个模块内部，这种模块之间的耦合为(42)。

- (42)A. 数据耦合 B. 标记耦合 C. 公共耦合 D. 内容耦合

【答案】D

【解析】本题考查软件设计的基础知识。

模块独立性是创建良好设计的一个重要原则，一般采用模块间的耦合和模块的内聚两个准则来进行度量。耦合是模块之间的相对独立性的度量，模块之间的连接越紧密，联系越多，耦合性就越高，而其模块独立性越弱。存在多种耦合关系：

数据耦合：若一个模块访问另一个模块时，彼此之间是通过数据参数(不是控制参数、公共数据结构或外部变量)来交换输入、输出信息的，则称这种耦合大数据耦合。

标记耦合：若一组模块通过参数表传递记录信息，则这种耦合称为标记耦合。

公共耦合:若一组模块都访问同一个公共数据环境,则它们之间的耦合就称为公共耦合。

内容耦合：若一个模块直接访问另一个模块内部数据；一个模块不通过正常入口转到另一模块内部；两个模块有一部分程序代码重叠；一个模块有多个入口。存在上述情况之一，则这种耦合称为内容耦合。

在面向对象技术中，(43)是一组具有相同结构、相同服务、共同关系和共同语义的(44)集合，其定义包括名称、属性和操作。

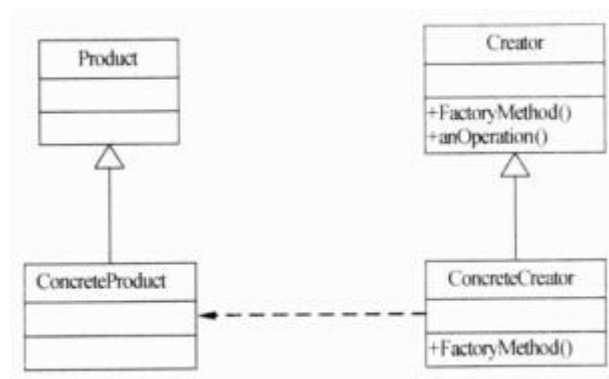
- (43) A. 类 B. 对象 C. 实例 D. 属性
- (44) A. 类 B. 对象 C. 实例 D. 属性

【答案】 A B

【解析】本题考查面向对象技术的基础知识。

几个选项都是面向对象技术中的基本概念，类是一组具有相同结构、相同服务、共同关系和共同语义的对象集合，其定义包括名称、属性和操作。类创建对象，对象是类的一个实例。属性用于刻画类和对象。

下图中, 类 Product 和 ConcreteProduct 的关系是(45), 类 ConcreteCreator 和 ConcreteProduct 的关系是(46)。



- (45) A. 继承 B. 关联 C. 组合 D. 依赖
- (46) A. 继承 B. 关联 C. 组合 D. 依赖

【答案】 A D

【解析】 本题考查面向对象技术的基础知识。

类之间的关系主要有几种：继承关系是指子类自动地具有其父类的全部属性与操作，也

称为父类对子类的泛化；关联关系是指两个或多个类之间的一种静态关系，表现为一个类是另一个类的成员变量。若一个类由若干个其他类作为其构成部分，则这样的关系是组合关系。依赖关系也是类之间的一种静态关系，表现为一个类是另一个类的局部变量。上图中，Product 与 ConcreteProduct 以及 Creator 与 ConcreteCreator 之间的关系是继承关系，在 UML 建模语言中，用空心三角形表示，从子类指向父类；ConcreteProduct 与 ConcreteCreator 之间是依赖关系，类 ConcreteCreator 依赖类 ConcreteProduct。

以下关于建立良好的程序设计风格的叙述中，正确的是(47)。

- (47) A. 程序应简单、清晰、可读性好 B. 变量的命名要符合语法
C. 充分考虑程序的执行效率 D. 程序的注释可有可无

【答案】A

【解析】本题考查软件实现的基础知识。

在编写程序时，变量的命名必须符合语法要求，这是正确性要求，不是程序设计风格的范畴。程序的执行效率是软件性能方面的要求，也不属于程序设计风格的范畴。良好的程序设计风格，包括程序的简单性、清晰性和可读性等方面，还包括有清晰的注释。

由于硬件配置的变化，如机型、终端或打印机等导致软件系统需要进行修改维护，这类维护属于(48)。

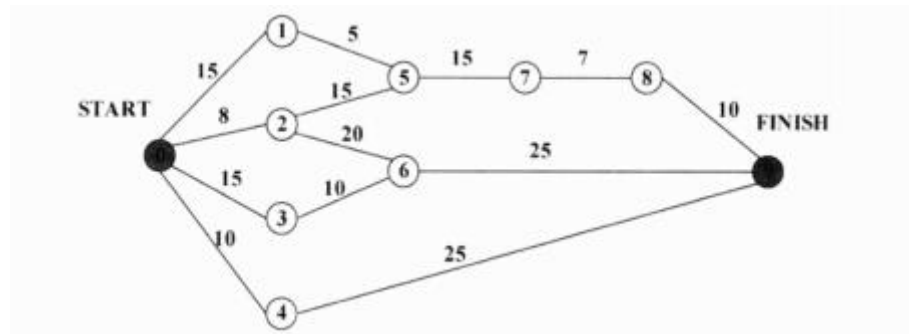
- (48) A. 改正性 B. 适应性 C. 完善性 D. 预防性

【答案】B

【解析】本题考查软件维护技术。

在软件开发完成交付用户使用后，就进入软件运行/维护阶段。软件维护活动根据其内容可以分为四种类型：改正性维护是指为了识别和纠正软件错误、改正软件性能上的缺陷、排除实施中的误使用，应进行的诊断和改正错误的过程；适应性维护是指由于信息技术飞速发展，软件运行的外部环境或数据环境可能发生变化，为了使软件适应这种变化，而修改软件的过程；完善性维护是指在软件使用过程中，用户往往会对软件提出新的功能与性能要求，为了满足这些要求，需要修改或再开发软件，以扩充软件功能、增强软件性能、改进加工效率、提高软件的可维护性而进行的维护活动；预防性维护是为了提高软件的可维护性和可靠性等，为以后进一步改进软件打下良好基础而进行的维护工作。

下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示包含的活动，则里程碑(49)没有按时完成会影响整个项目的进度。若活动 0→2 完成后，停止 3 天才开始活动 2→6, 则完成整个项目的最少时间是(50)天。



- (49) A. 1 B. 2 C. 3 D. 4
- (50) A. 53 B. 55 C. 56 D. 57

【答案】B C

【解析】本题考查软件项目管理的基础知识。

根据关键路径法，计算出关键路径为 0—2—5—7—8—9, 关键路径长度为 55。因此里程碑 2 在关键路径上，而里程碑 1、3 和 4 不在关键路径上。

若活动 0→2 完成后，停止 3 天才开始活动 2→6, 则相当于 0—2—6—9 是一个关键路径，此时，完成项目的最短时间应该是 56 天。

某软件系统无需在线容错，也不能采用冗余设计，如果对可靠性要求较高，故障有可能导致严重后果，一般采用(51)。

- (51) A. 恢复块设计 B. N 版本程序设计 C. 检错技术 D. 降低复杂度设计

【答案】C

【解析】本题考查软件的可靠性设计。

故障的恢复策略一般有前向恢复和后向恢复两种。所谓前向恢复是指使当前的计算继续下去，把系统恢复成连贯的正确状态，弥补当前状态的不连贯情况，这需要有错误的详细说明。所谓后向恢复是指系统恢复到前一个正确状态，继续执行。

恢复块方法采用后向恢复策略。它提供具有相同功能的主块和几个后备块，主块首先投入运行，结束后进行验收测试，如果没有通过验收测试，系统经现场恢复后由一后备块运行。这一过程可以重复到耗尽所有的后备块，或者某个程序故障行为超出了预料，从而导致不可恢复的后果。设计时应保证实现主块和后备块之间的独立性，避免相关错误的产生，使主块

和后备块之间的共性错误降到最低限度。验收测试程序完成故障检测功能，它本身的故障对恢复块方法而言是共性，因此，必须保证它的正确性。

N 版本程序设计是一种静态的故障屏蔽技术，采用前向恢复的策略，其设计思想是用 N 个具有相同功能的程序同时执行一项计算，结果通过多数表决来选择。

本题目中的故障可能导致严重后果，所以应采取检测技术预防故障的产生。

软件可靠性管理把软件可靠性活动贯穿于软件开发的全过程，成为软件工程管理的一部分。确定软件可靠性度量活动属于(52)阶段。

(52)A. 需求分析 B. 概要设计 C. 详细设计 D. 测试阶段

【答案】B

【解析】 本题考查软件可靠性概念。

对于软件可靠性度量活动的确定是概要设计阶段的工作。

软件测试的对象不包括(53)。

(53)A. 软件代码 B. 软件开发过程 C. 文档 D. 数据

【答案】B

【解析】 本题考查软件测试的对象。

根据软件的定义，软件包括程序、数据和文档。所以软件测试并不仅仅是程序测试，还应包括相应文档和数据的测试。

本题中选项 B 软件开发过程不属于上述三者之一，不是软件测试的对象。

以下关于测试计划的叙述中，不正确的是(54)。

(54)A. 测试计划能使测试工作顺利进行 B. 测试计划能使项目参与人员沟通顺畅
C. 测试计划无益于提高软件质量 D. 测试计划是软件过程规范化的要求

【答案】C

【解析】 本题考查软件测试计划相关知识。

《ANSI/IEEE 软件测试文档标准 829-1983》将测试计划定义为：“一个叙述了预定的测试活动的范围、途径、资源及进度安排的文档。它确认了测试项、被测特征、测试任务、人员安排，以及任何偶发事件的风险。”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。

根据上述描述，测试技术显然有益于提高软件质量。

以下关于软件测试原则的叙述中，正确的是(55)。

- (55) A. 测试用例只需选用合理的输入数据，不需要选择不合理的输入数据
- B. 应制定测试计划并严格执行，排除随意性
- C. 穷举测试是可能的
- D. 程序员应尽量测试自己的程序

【答案】B

【解析】本题考查软件测试的原则。

软件测试应遵循的原则包括：应当把“尽早和不断地测试”作为开发者的座右铭；程序员应该避免检查自己的程序，测试工作应该由独立的专业的软件测试机构来完成；设计测试用例时，应该考虑到合法的输入和不合法的输入，以及各种边界条件；一定要注意测试中的错误集中发生现象，应对错误群集的程序段进行重点测试；对测试错误结果一定要有一个确认的过程；制定严格的测试计划，并把测试时间安排得尽量宽松，不要希望在极短的时间内完成一个高水平的测试；回归测试的关联性一定要引起充分的注意，修改一个错误而引起更多错误出现的现象并不少见；妥善保存一切测试过程文档；穷举测试是不能实现的。

根据上述描述，测试过程中需要制定测试计划并严格执行。

以下关于测试时机的叙述中，不正确的是(56)。

- (56) A. 应该尽可能早地进行测试
- B. 软件中的错误暴露得越迟，则修复和改正错误所花费的代价就越高
- C. 应该在代码编写完成后开始测试
- D. 项目需求分析和设计阶段需要测试人员参与

【答案】C

【解析】本题考查软件测试一些基础知识。

在传统的瀑布模型中，软件项目主要有用户需求分析、概要设计、详细设计、编码和实现、测试以及运行维护几个阶段组成。因此人们一般认为，软件测试只是软件编码后的一个

阶段。但随着软件测试业的发展，人们越来越认识到：软件测试不应只是软件项目的收尾工作，而应该在软件生命周期的每一阶段中都包含测试。软件测试是贯穿于整个软件开发生命周期的过程活动，包括软件测试计划、软件测试需求分析、软件测试用例设计、软件测试执行、软件缺陷管理、软件测试风险管理以及其他的一些软件测试相关的活动等等组成。在软件项目的每个阶段，都需要进行不同目的和不同内容的测试活动，以保证各个阶段工作产品输出的正确性。在项目开发的后期，发现一些软件需求阶段和概要设计阶段的错误和问题，修改这些缺陷导致的成本将是非常高的。有资料表明：平均而言，如果在需求阶段修正一个错误的代价是 1，那么，在设计阶段就是它的 3~6 倍，在编程阶段是它的 10 倍，在内部测试阶段是它的 20~40 倍，在外部测试阶段是它的 30~70 倍，而到了产品发布出去，这个数字就是 40~1000 倍。

根据上述描述，测试活动贯穿于整个开发生命周期。

以下不属于软件测试工具的是(57)。

- (57) A. JMeter B. LoadRunner C. JTest D. JBuilder

【答案】D

【解析】本题考查软件工具的基本知识。

JMeter、LoadRunner 和 JTest 都是软件测试工具。

JBuilder 是一种软件开发工具，不属于软件测试工具。

软件的易用性包括(58)。

- ①理解性 ②易学习性 ③易操作性 ④吸引力 ⑤依从性

- (58) A. ①② B. ①②③ C. ①②③④ D. ①②③④⑤

【答案】D

【解析】本题考查易用性测试的基础知识。

软件的易用性包括易理解性、易学习性、易操作性、吸引力以及依从性。

黑盒测试不能发现(59)。

- (59) A. 功能错误或者遗漏 B. 输入输出错误
C. 执行不到的代码 D. 初始化和终止错误

【答案】C

【解析】 本题考查黑盒测试的基础知识。

黑盒测试是把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，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规定正常使用，程序是否能适当地接收输入数据而产生正确的输出信息。黑盒测试着眼于程序外部结构，不考虑内部逻辑结构，主要针对软件界面和软件功能进行测试。黑盒测试能发现功能错误或者遗漏、输入输出错误以及初始化和终止错误。

由于黑盒测试不考虑程序内部结构，所以该方法不能检测到执行不到的代码。

以下关于边界值测试法的叙述中，不正确的是(60)。

- (60) A. 边界值分析法不仅重视输入域边界，而且也必须考虑输出域边界
B. 边界值分析法是对等价类划分方法的补充
C. 发生在输入输出边界上的错误比发生在输入输出范围的内部的错误要少
D. 测试数据应尽可能选取边界上的值，而不是等价类中的典型值或任意值

【答案】 C

【解析】 本题考查黑盒测试中的边界值测试法。

边界值测试法就是对输入或输出的边界值进行测试的一种黑盒测试方法。边界值测试法作为对等价类划分法的补充，这种情况下，其测试用例来自等价类的边界。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。边界值测试不是从某等价类中随便挑一个作为代表，而是使这个等价类的每个边界都要作为测试条件。边界值测试不仅考虑输入条件，还要考虑输出空间产生的测试情况。用边界值测试方法设计测试用例，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。

根据上述描述，发生在输入输出边界上的错误比发生在输入输出范围的内部的错误要多。

白盒测试不能发现(61)。

- (61) A. 代码路径中的错误 B. 死循环 C. 逻辑错误 D. 功能错误

【答案】 D

【解析】 本题考查白盒测试的基础知识。

白盒测试也称结构测试或逻辑驱动测试，它是知道产品内部工作过程，按照程序内部的结构测试程序，检验程序中的每条通路是否都有能按预定要求正确工作，而不考虑其功能，白盒测试的主要方法有静态结构分析法、逻辑覆盖法、基本路径测试法等。白盒测试完全基于代码，能发现代码路径中的错误、程序中的死循环以及逻辑错误。白盒测试不关心功能需求，因此无法确定功能正确与否。

对于逻辑表达式 $((a \& b) \mid \mid c)$ ，需要(62)个测试用例才能完成条件组合覆盖。

(62) A. 2 B. 4 C. 8 D. 16

【答案】C

【解析】本题考查白盒测试中逻辑覆盖法的条件组合覆盖。

条件组合覆盖的含义是：选择足够的测试用例，使得每个判定中条件的各种可能组合都至少出现一次。

本题中有三个条件，组合之后需要的用例数是 8。

为检测系统在长时间运行下是否存在性能瓶颈，应进行(63)。

(63) A. 负载测试 B. 压力测试 C. 疲劳强度测试 D. 大数据量测试

【答案】C

【解析】本题考查负载测试、压力测试、疲劳强度测试、大数据量测试的基本知识。

负载测试是通过逐步增加系统负载，测试系统性能的变化，并最终确定在满足性能指标的情况下，系统所能承受的最大负载量的情况。压力测试是通过逐步增项系统负载，测试系统性能的变化，并最终确定在什么负载条件下系统性能处于失效状态，并以此来获得系统能提供的最大服务级别的测试。疲劳强度测试是采用系统稳定运行情况 F 能够支持的最大并发用户数，或者日常运行用户数，持续执行一段时间业务，保证达到系统疲劳强度需求的业务量，通过综合分析交易执行指标和资源监控指标，来确定系统处理最大工作量强度性能的过程。大数据量测试包括独立的数据量测试和综合数据量测试，独立数据量测试是指针对系统存储、传输、统计、查询等业务进行的大数据量测试；综合数据量测试是指和压力测试、负载测试、疲劳强度测试相结合的综合测试。

本题的目标是检测系统在长时间运行下的表现，应进行疲劳强度测试。

以下关于负载压力测试的叙述中，不正确的是(64)。

- (64) A. 负载压力测试用于确认系统是否支持性能需求
B. 负载压力测试能得到系统可承受的业务量增长
C. 负载压力测试是在一定约束条件下测试系统所能承受的最大负载压力
D. 负载压力测试不用于发现不同负载场景下的速度变慢、内存泄露等问题

【答案】D

【解析】本题考查负载压力测试的基础知识。

负载压力测试是在一定约束条件下测试系统所能承受的并发用户量、运行时间、数据量，以确定系统所能承受的最大负载压力。负载压力测试有助于确认被测系统是否能够支持性能需求，以及预期的负载增长等。负载压力测试不只是关注不同负载场景下的响应时间等指标，它也要通过测试来发现在不同负载场景下会出现的，例如速度变慢、内存泄漏等问题。负载压力测试是性能测试的重要组成部分。

根据上述描述，负载压力测试可以发现不同负载场景下的速度变慢、内存泄露等问题。

测试过程中，正确的测试顺序应该是(65)。

①单元测试 ②集成测试 ③系统测试

- (65) A. ①②③ B. ③①② C. ②③① D. ③②①

【答案】A

【解析】本题考查软件测试的过程。

测试过程应该是从模块层开始，然后扩大延伸到整个基于计算机的系统集中。因此正确的测试顺序应该是从针对单个模块的单元测试开始，然后逐步集成各个单元，最后进行系统测试。

以下属于静态测试方法的是(66)。

- (66) A. 分支覆盖率分析 B. 复杂度分析 C. 系统压力测试 D. 路径覆盖分析

【答案】B

【解析】本题考查软件静态测试方法。

静态测试不实际运行被测软件，主要是对软件的编程格式、结构等方面过行评估。本题中分支覆盖率分析、系统压力测试和路径覆盖分析都需要实际运行被测软件才能得到，只有复杂度分析可通过静态分析程序代码实现。

以下关于集成测试的叙述中，不正确的是(67)。

- (67)A. 在完成软件的概要设计后，即开始制定集成测试计划
- B. 实施集成测试时需要设计所需驱动和桩
- C. 桩函数是所测函数的主程序，它接收测试数据并把数据传送给所测试函数
- D. 常见的集成测试方法包括自顶向下、自底向上、Big-Bang 等

【答案】C

【解析】本题考查集成测试的基础知识。

集成测试是在单元测试的基础上，测试在将所有的软件单元按照概要设计规格说明的要求组装成模块、子系统或系统的过程中各部分工作是否达到或实现相应技术指标及要求的活动。集成测试的依据是概要设计规格说明，因此在概要设计完成后即开始制定集成测试计划。集成测试需要设计所需的驱动模块和桩模块。驱动模块用来模拟被测试模块的上一级模块，相当于被测模块的主程序，它接收数据，将相关数据传送给被测模块，启动被测模块，并打印出相应的结果。桩模块用来模拟被测模块工作过程中所调用的模块，它们一般只进行很少的数据处理。常见的集成测试方法包括自顶向下、自底向上、Big-Bang 等。

本题中选项 C 混淆了驱动和桩的概念。

对一段信息生成消息摘要防止信息在网络传输及存储过程中被篡改的基本手段，(68)不属于生成消息摘要的基本算法。

- (68)A. MD5 B. SHA-1 C. RSA D. SHA-256

【答案】C

【解析】本题考查生成消息摘要的基本算法。

对信息生成消息摘要防止信息在网络传输及存储过程中被篡改的基本手段，生成消息摘要的算法应是不可逆的，即基于消息摘要无法计算出原始消息内容。MD5、SHA-1、SHA-256 均属于典型的生成消息摘要的算法，而 RSA 是常用的公钥加密算法，其加密产生的数据可以被还原，因此不属于生成消息摘要的算法。

软件系统的安全性是信息安全的一个重要组成部分，针对程序和数据的安全性测试与评估是软件安全性测试的重要内容，(69)不属于安全件测试与评估的基本内容。

- (69)A. 用户认证机制 B. 加密机制
- C. 系统能承受的并发用户量 D. 数据备份与恢复手段

【答案】C

【解析】本题考查安全性测试与评估的基本内容。

安全性测试与评估是软件安全性测试的重要内容,其测试与评估的基本内容包括用户认证机制、加密机制、安全防护策略、数据备份与恢复手段、防病毒系统等,而系统能承受的并发用户量属于基本的应用负载压力测试内容。

模拟攻击试验是一种基本的软件安全性测试方法,以下关于模拟攻击试验的叙述中,正确的是(70)。

(70)A. 模拟攻击试验必须借助于特定的漏洞扫描器才能完成

B. 对安全测试来说,模拟攻击试验是一组特殊的白盒测试案例,必须在充分了解系统安全机制的软件组成基础上,才能进行相应攻击试验的设计与试验

C. 缓冲区溢出攻击是一种常见的模拟攻击试验,此类攻击者通常通过截获含有身份鉴别信息或授权请求的有效消息,将该消息进行重演

D. 服务拒绝攻击是一种常见的模拟攻击试验,此类攻击者通常通过向服务器发送大量虚假请求,使得服务器功能不能正常执行

【答案】D

【解析】本题考查模拟攻击试验的基本原理。

模拟攻击试验是一组特殊的墨盒测试案例,以模拟攻击来验证软件或信眉系统的安全防护能力,因为是墨盒测试,所以通常可在不借助漏洞扫描器的前提下完成,因此选项 A 与选项 B 是均是错误的;而选项 C 描述的应为模拟重演攻击的基本原理,因此也是错误的;选项 D 描述的是服务拒绝攻击的基本原理,因此本题正确答案应为 D。

There is nothing in this world constant but inconstancy. —SWIFT

Project after project designs a set of algorithms and then plunges into construction of customer-deliverable software on a schedule that demands delivery of the first thing built.

In most projects, the first system built is (71) usable. It may be too slow, too big, awkward to use, or all three. There is no (72) but to start again, smarting but smarter, and build a redesigned version in which these problems are solved. The discard and (73) may be done in one lump, or it may be done piece-by-piece. But all

large-system experience shows that it will be done. Where a new system concept or new technology is used, one has to build a system to throw away, for even the best planning is not so omniscient (全知的) as to get it right the first time.

The management question, therefore, is not whether to build a pilot system and throw it away. You will do that. The only question is whether to plan in advance to build a (74), or to promise to deliver the throwaway to customers. Seen this way, the answer is much clearer. Delivering that throwaway to customers buys time, but it does so only at the (75) of agony (极大痛苦) for the user, distraction for the builders while they do the redesign, and a bad reputation for the product that the best redesign will find hard to live down.

Hence plan to throw one away; you will, anyhow.

- | | | | |
|--------------------|-------------|-------------|---------------|
| (71)A. almost | B. often | C. usually | D. barely |
| (72)A. alternative | B. need | C. plan | D. build |
| (73)A. design | B. redesign | C. plan | D. build |
| (74)A. throwaway | B. system | C. software | D. product |
| (75)A. worth | B. value | C. cost | D. invaluable |

【答案】D A B A C

【解析】

不变只是愿望，变化才是永恒。——SWIFT

一个接一个的软件项目都是一开始设计算法，然后将算法应用到待发布的软件中，接着根据时间进度把第一次开发的产品发布给客户。

对于大多数项目，第一个开发的系统并不合用。它可能太慢、太大、难以使用，或者三者兼有。要解决所有的问题，除了重新开始以外，没有其他的办法——即开发一个更灵巧或者更好的系统。系统的丢弃和重新设计可以一步完成，也可以一块块地实现。所有大型系统的经验都显示，这是必须完成的步骤。而且，新的系统概念或新技术会不断出现，因此开发的系统必须被抛弃，但即使是最优秀的项目计划也不能无所不知地在最开始就解决这些问题。因此，管理上的问题不再是“是否构建一个实验性的系统，然后抛弃它”，你必须这样做。现在的问题是“是否预先计划抛弃原型的开发，或者是否将该原型发布给用户”。从这个角度看待问题，答案更加清晰。将原型发布给用户，虽然可以获得时间，但是其代价高昂——对于用户，使用极度痛苦；对于重新开发的人员，分散了精力；对于产品，影响了声誉，即

使是最好的再设计也难以挽回名声。

因此，为舍弃而计划，无论如何，你一定要这样做。

试题一

【说明】

逻辑覆盖法是设计白盒测试用例的主要方法之一,通过对程序逻辑结构的遍历实现程序的覆盖。针对以下由 C 语言编写的程序,按要求回答问题。

```
int XOR(char * filename, unsigned long key){
    FILE * input = NULL , *output = NULL;           //1
    char * outfilename = NULL;
    int len = strlen(filename);
    unsigned char buffer;
    if( (filename[len-2] == '.') && (filename[len-1] == 'c') ){ //2,3
        outfilename = new char[len+1];               //4
        strcpy(outfilename,filename);
        outfilename[len-2] = '\\0';
    }
    else{                                             //5
        outfilename = new char[len+5];
        strcpy(outfilename,filename);
        strncat(outfilename, ".c",2);
    }
    input = fopen(filename,"rb");
    if( input == NULL){                             //6
        cout << "Error opening file " << filename << endl; //7
        delete [] outfilename;
        outfilename = NULL;
        return 1;
    }
    output = fopen(outfilename,"wb");
    if( output == NULL ){                           //8
        cout << "Error creating output file " << outfilename << endl; //9
        delete [] outfilename;
```

```

        outfilename = NULL;
        return 1;
    }
    while( ! feof(input) ){ //10
        if( fread(&buffer,sizeof(unsigned char),1,input) != 1 ){ //11
            if( ! feof(input) ){ //12
                delete [] outfilename; //13
                outfilename = NULL;
                fclose(input);
                fclose(output);
                return 1;
            }
        }
        else{ //14
            buffer ^= key;
            fwrite(&buffer,sizeof(unsigned char),1,output);
        }
    }
    fclose(input); //15
    fclose(output);
    delete [] outfilename;
    return 0;
}

```

【问题1】

请给出满足 100%DC(判定覆盖)所需的逻辑条件。

编 号	条 件	
1	(filename[len-2] == '.') && (filename[len-1] == 'c')	
2	!((filename[len-2] == '.') && (filename[len-1] == 'c'))	
3	input == NULL	
4	input != NULL	
5	output == NULL	
6	output != NULL	
7	!feof(input)	
8	feof(input)	
9	fread(&buffer,sizeof(unsigned char),1,input) != 1	
10	fread(&buffer,sizeof(unsigned char),1,input) == 1	
11	!feof(input)	
12	feof(input)	

本题考查白盒测试技术的应用。

本问题考查白盒测试用例设计方法：判定覆盖法。

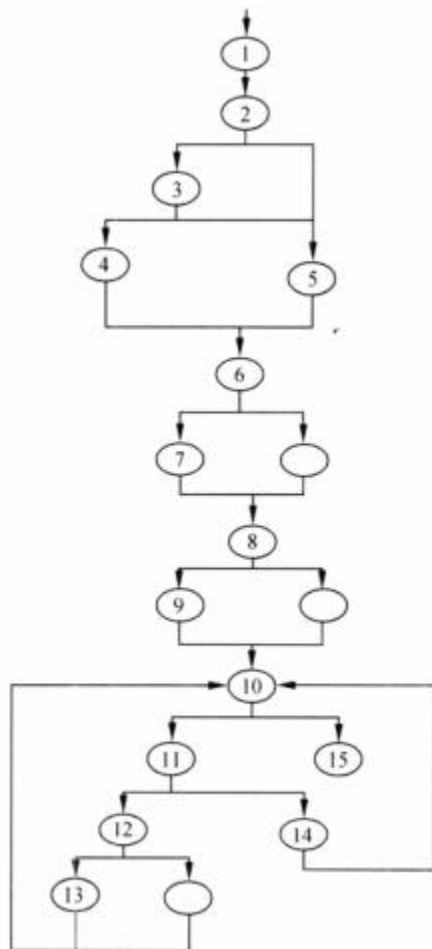
判定覆盖指设计足够的测试用例，使得被测程序中每个判定表达式至少获得一次真”值

和“假”值，从而使程序的每一个分支至少都通过一次。本题中程序有 6 个判定，所以满足判定覆盖一共需要 12 个逻辑条件。

【问题 2】

请画出上述程序的控制流图，并计算其控制流图的环路复杂度 $V(G)$ 。

环路复杂度 $V(G)=8$ 。



本问题考查白盒测试用例设计方法：基本路径法。涉及的知识点包括：根据代码绘制控制流图、计算环路复杂度。

控制流图是描述程序控制流的一种图示方式，它由节点和定向边构成。控制流图的节点代表一个基本块，定向边代表控制流的方向。其中要特别注意的是，如果判断中的条件表达式是复合条件，即条件表达式是由一个或多个逻辑运算符连接的逻辑表达式，则需要改变复合条件的判断为一系列之单个条件的嵌套的判断。本题程序中，`if((filename[len-2]='.') && (filename[len-1] = 'c'))` 这条判断语句中的判定由两个条件组成，因此在画控制流图的时候需要拆分成两条判断语句。程序的环路复杂度等于控制流图中判定节点的个数加 1，本题控制流图中判定节点个数为 7，所以 $V(G)=8$ 。

【问题 3】

请给出问题 2 中控制流图的线性无关路径。

线性无关路径：

1. 1-2-3-4-6-7-8-9-10-11-12-13-10...
2. 1-2-5-6-7-8-9-10-11-12-13-10...
3. 1-2-3-5-6-7-8-9-10-11-12-13-10...
4. 1-2-3-4-6-8-9-10-11-12-13-10...
5. 1-2-3-4-6-7-8-10-11-12-13-10...
6. 1-2-3-4-6-7-8-9-10-15
7. 1-2-3-4-6-7-8-9-10-11-14-10...
8. 1-2-3-4-6-7-8-9-10-11-12-10...

本问题考查白盒测试用例设计方法：基本路径法。涉及的知识点包括：根据控制流图和环路复杂度给出线性无关路径。

线性无关路径是指包含一组以前没有处理的语句或条件的路径。从控制流图上来看，一条线性无关路径是至少包含一条在其他线性无关路径中从未有过的边的路径。程序的环路复杂度等于线性无关路径的条数，所以本题中应该有 8 条线性无关路径。

试题二

【说明】

某软件公司为某银行设计开发了一套网上银行系统，该系统提供基本的支付、查询、转账和信息修改等功能。银行对网上银行系统提出了初步的性能指标：

- (1) 交易响应时间不超过 2s
- (2) 并发用户数 ≥ 1000
- (3) CPU 利用率不超过 80%
- (4) 系统需要 7*24 小时不间断的稳定运行
- (5) 每秒事务数为 7
- (6) 交易成功率为 100%

现需要对该软件进行性能测试。

【问题 1】

常见的性能测试包括负载测试、压力测试、并发性能测试、疲劳强度测试和大数据量测试等。针对题目中所述的 6 个性能指标，并发性能测试和疲劳强度测试所涉及的指标各有哪些？

并发性能测试涉及的指标有 (1)、(2)、(5)

疲劳强度测试涉及的指标有 (4)

本题主要考查负载压力测试相关知识及应用。

负载压力测试是指在一定约束条件下测试系统所能承受的并发用户量、运行时间、数据量等，以确定系统所能承受的最大负载压力。负载压力测试是性能测试的重要组成部分，包括负载测试、压力测试、并发性能测试、疲劳强度测试、大数据量测试等内容。

本问题考查负载压力测试的基本概念。

负载测试是通过逐步增加系统负载，测试系统性能的变化，并最终确定在满足性能指标的情况下，系统所能承受的最大负载量所进行的测试。

压力测试是通过逐步增加系统负载，测试系统性能的变化，并最终确定在什么负载条件下，系统性能处于失效状态，以此来获得系统能提供的最大服务级别的测试。所以说，压力测试是一种特定类型的负载测试。

并发性能测试，包含了负载测试和压力测试。通过逐渐增加并发用户数负载，直到系统的瓶颈或者不能接收的性能点，通过综合分析交易执行指标、系统资源监控指标来确定系统并发性能。并发性能测试是负载压力测试中的重要内容。

疲劳强度测试，通常是采用系统稳定运行情况下能够支持的最大并发用户数，或者日常运行用户数，持续执行一段时间业务，保证达到系统疲劳强度需求的业务量，通过综合分析交易执行指标、系统资源监控指标，确定系统处理最大工作量强度 1 生能的过程。一般情况下利用疲劳强度测试老模拟系统日常业务操作。

大数据量测试，包括独立的数据量测试和综合数据量测试。独立的数据量测试是针对某些系统存储、传输、统计、查询等业务进行的大数据量测试；综合数据量测试是指和压力测试、负载性能测试、疲劳强度测试相结合的综合测试。

从题干列举的 6 种性能指标看，根据各类测试的定义，可以得出(1)、(2)、(5)属于并发性能测试，(4)属于疲劳强度测试。(3)属于一种系统资源监控的指标，而(6)的定义不明确，因为交易成功率取决多种因素，包括业务数据、用户请求数据、基础软件、网络环境等等。

【问题 2】

性能测试中，针对一个单独的性能指标，往往需要采用多种不同的测试方法。该软件公司需要测试性能指标(1)和(5)，设计了如下的测试方案：

测试常规情况下的并发用户数，逐步增加并发用户数，分别测试：

(1)在响应时间为 2s 时，系统所能承受的最大并发访问用户的数量；

(2)系统在多大的并发访问用户数量下，响应时间不可接受(例如超过 2s)。

请指出这两项测试分别属于哪种类型的测试，并分别解释这两种测试类型的基本概念。

第一种测试属于负载测试，第二种属于压力测试。

负载测试是通过逐步增加系统负载，测试系统性能的变化，并最终确定在满足性能指标的情况下，系统所能承受的最大负载量的测试。

压力测试是通过逐步增加系统负载，测试系统性能变化，并最终确定在什么负载条件下系统性能处于失效状态，以此来获得系统能提供的最大服务级别的测试。

本问题考查负载测试和压力测试的区别。

从问题 1 的分析中可以看出，负载测试强调的是在满足性能指标的情况下，系统所能承受的最大负载量的测试，前提是系统正常工作；而压力测试则是要使得系统性能处于失效状态，即压力测试是为了发现在什么条件下系统的性能会变得不可接受。

因此，通过对本问题描述的分析可以得出，“在响应时间为 2s 时，系统所能承受的最大并发访问用户的数量”属于负载测试，而“系统在多大的并发访问用户数量下，响应时间不可接受(例如超过 2s)”属于压力测试。

【问题 3】

在测试性能指标(5)时，该软件公司在客户端模拟大量并发用户来执行业务操作，统计平均的每秒事务数。该软件公司认为客户端接收响应信息与该性能指标的测试无关，因此在模拟客户端上发起正常业务申请，接收系统响应后直接丢弃响应信息，没有进行功能校验。请说明该软件公司的做法正确与否，并简要说明原因以及执行功能校验的副作用。

该软件公司的做法是错误的。

该软件公司在负载压力测试中没有进行功能校验，忽略了负载压力情况下的功能不稳定问题。没有正确的功能保证，负载压力性能测试就失去了意义。

在测试过程中进行功能校验，需要记录业务操作结果，会导致资源消耗、操作行为增加以及产生大量日志等问题。

本问题考查负载压力测试中的功能内容校验。

负载压力测试中的功能内容校验，指的是在进行负载压力测试时，需要核查功能的执行是否正常，在功能异常或错误时，需要记录所产生的错误。如果没有功能执行结果正确的保证，负载压力性能测试失去了意义。对本为题而言，如果不能保证业务操作的正确，则平均每秒事务数的指标是不正确的。

执行功能内容校验也会带来一些副作用，包括资源消耗、操作行为增加，以及产生大量日志等问题。

试题三

【说明】

某高校开发了一套基于 Web 的教务管理系统，实现教务管理人员课程设置、学生选课和成绩查询、教师上传成绩以及特殊情况下教务处对成绩进行修改等功能。系统基于 JavaEE 平台实现，采用表单 (Form) 实现用户数据的提交并与用户交互。

系统要支持：

- (1) 在特定时期内 100 个用户并发时，主要功能的处理能力至少要达到 10 个请求/秒，平均数据量 8KB/请求；
- (2) 用户可以通过不同的移动设备、操作系统和浏览器进行访问。

【问题 1】

简要叙述教务管理系统表单测试的主要测试内容。

表单测试是 Web 应用功能测试的重要内容，教务管理系统主要测试如下内容：

- ①每个字段的验证；
- ②字段的缺省值；
- ③表单中的输入；
- ④提交操作的完整性。

本题考查 Web 应用测试相关内容。Web 应用测试除了类似传统软件系统测试性能测试、压力测试等之外，还需要测试链接、浏览器、表单和可用性等多个方面。

本问题考查表单测试的主要内容。表单是 Web 应用的重要组成部分，用于获取用户的信息并和用户进行交互。因此，表单测试是 Web 应用功能测试的重要内容，需要测试：

- ①首先检查每个字段的所有验证；
- ②检查字段的缺省值；
- ③表单中的错误输入；
- ④如果有创建、删除、查看和修改表单，要进行测试。

【问题 2】

简要叙述为了达到系统要支持的 (2)，需要进行哪些兼容性测试，并设计一个兼容性测试矩阵。

“教务管理系统”的兼容性测试：

①平台兼容性和浏览器兼容性。

②兼容性测试矩阵如下：

浏览器 平台	IE(7、8、9、10)	Firefox 12	Google Chrome	Android browser	Safari
Windows XP						
Windows 8						
iOS						
Android						
.....						

本问题考查 Web 应用兼容性测试的内容。Web 应用的兼容性是测试工作的重要方面，包括：浏览器兼容性、操作系统平台兼容性、移动浏览、打印选项等。本系统用户可以通过不同的移动设备、操作系统和浏览器进行访问，因此需要针对普通设备和移动设备，进行操作系统平台和浏览器的兼容性测试。

【问题 3】

在满足系统要支持的(1)时，计算系统的通信吞吐量。

通信吞吐量： $P=N(\text{并发用户的数量}=100) \times T(\text{每单位时间的在线事务数量}=10) \times D(\text{事务服务器每次处理的数据负载}=8\text{KB/s})=100 \times 10 \times 8=8000\text{KB/s}$ 。

本问题考查 Web 应用系统的性能指标计算。通信吞吐量，设定如下指标参数：

N：并发用户的数量；

T：每单位时间的在线事务数量；

D：事务服务器每次处理的数据负载；

P：系统的通信吞吐量。

有如下计算公式：

$$P=N \times T \times D$$

本题中系统要求支持的(1)中给出 100 个用户并发，即 $N=100$ ；主要功能的处理能力至少要达到 10 个请求/秒，即 $T=10$ ；平均数据量 8KB/请求，即 $D=10\text{KB/s}$ 。

则：通信吞吐量 $P=100 \times 10 \times 8=8000\text{KB/S}$

【问题 4】

系统实现时，对成绩更新所用的 SQL 语句如下：

```
“ UPDATE StudentScore SET score = ” + intClientSubmitScore + “ WHERE  
Stuent_ID=+strStudentID + “’;”
```

设计 1 个测试用例，以测试该 SQL 语句是否能防止 SQL 注入，并说明该语句是否能防止 SQL 注入，以及如何防止 SQL 注入。

设计如下测试：【注：设计类似如下用例的一个即可，其中包含 SQL 功能符号使 SQL 变为不符合设计意图即可，如包含’， DROP 等】。

(1) intClientSubmitScore: 100—， strStudentID: 20130002, 则该 SQL 变为：

```
UPDATE StudentScore SET score = 100 -- WHERE Student_ID='20130002';
```

(2) intClientSubmitScore: 100, strStudentID: 20130002’; DROPTABLE StudentScore—，
则该 SQL 语句变为：

从测试用例所拼接处的 SQL 可以看出，该 SQL 语句不安全，容易造成 SQL 注入。防止 SQL 注入的方法主要有：拼接 SQL 之前对特殊符号进行转义，使其不作为 SQL 语句的功能符号。

```
UPDATE StudentScore SET score = 100 WHERE Student_ID='20130002'; DROP TABLE  
StudentScore -';
```

本问题考查 Web 应用安全性方面的 SQL 注入，SQL 注入是 Web 应用安全性测试的重要方面。

许多 Web 应用系统采用某种数据库，接收用户从 Web 页面中输入，完成展示相关存储的数据(如检查用户登录信息)、将输入数据存储到数据库(如用户输入表单中数据域并点击提交后，系统将信息存入数据库)等操作。在有些情况下，将用户输入的数据和设计好的 SQL 框架拼接后提交给数据库执行，就可能存在用户输入的数据并非设计的正确格式，从而给恶意用户提供了破坏的机会。即 SQL 注入。恶意用户输入不期望的数据，拼接后提交给数据库执行，造成可能使用其他用户身份、查看其他用户的私密信息，还可能修改数据库的结构，甚至是删除应用的数据库表等严重后果。因此需要在测试阶段进行认真严格的测试。

本系统实现时，对成绩更新所用的如下 SQL 语句：

```
"UPDATE StudentScore SET score=" + intClientSubmitScore + " WHERE Stuent_ID=  
'" + strStudentID + "';"
```

采用拼接字符串方式，无法防止 SQL 注入。例如 intClientSubmitScore: 100—，
strStudentID: 20130002, 则该 SQL 变为：

```
UPDATE StudentScore SET score = 100 -- WHERE Stuent_ID='20130002';
```

--是 SQL 中注释符号，其后的内容为注释，这样上述语句中--之后的内容变为注释，只要 StudentScore 表中所有的记录的 score 都变为 100, 而没有受到 WHERE 子句后的学号限制。

```
UPDATE StudentScore SET score = 100 WHERE Stuent_ID='20130002' or 'a'='a';
```

因为 'a'='a' 条件总是成立，因此，SQL 执行结果包括学生成绩表中所有行的 score 都更新为 100 分。

更为严重的情况下，用户输入 DROP 等功能性命令，会造成数据库表的永久删除等严重后果，如 strStudentID:20130002'; DROPTABLEStudentScore--，则该 SQL 语句：

```
UPDATE StudentScore SET score = 100 WHERE Stuent_ID='20130002'; DROP TABLE  
StudentScore --';
```

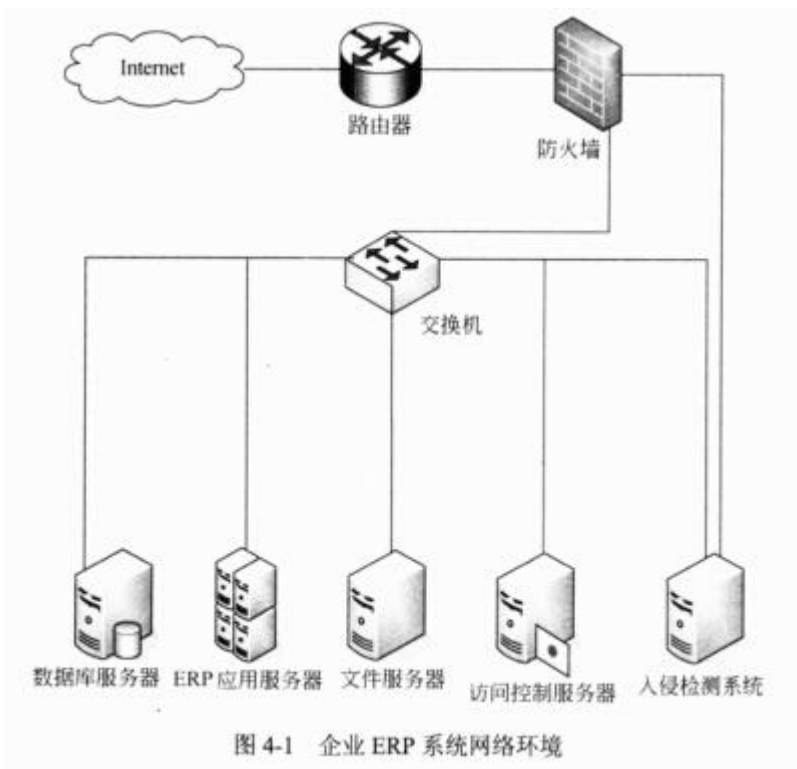
防止 SQL 注入的方法主要有：拼接 SQL 之前对特殊符号进行转义或者等价方式，使其不作为 SQL 语句的功能符号。验证所有输入数据能从输入层面防止 SQL 注入。

SQL 注入在使用 SSL 的应用中仍然存在，甚至是防火墙也无法防止 SQL 注入。因此，在测试 Web 应用时，需要认真仔细设计测试用例，采用 Web 漏洞扫描工具等进行检查，以保证不存在 SQL 注入机会。

试题四

【说明】

某企业最近上线了 ERP 系统，该系统运行的网络环境如图 4-1 所示。企业信息中心目前拟对该系统相关安全防护体系进行全面的安全性测试，以提供全面的安全测评报告。



【问题 1】

企业 ERP 系统上线后，企业主要业务的日常运作都高度依赖该系统的正常运行，因此 ERP 系统的稳定性与可靠性对企业至关重要。故障恢复与容灾备份措施是提高系统稳定性与可靠性的重要因素。对于故障恢复与容灾备份措施，参与测试的王工认为应从故障恢复、数据备份和容灾备份等三个方面进行测试。请用 300 字以内文字，对这三方面的测试内容进行简要说明。

- 对于故障恢复与容灾备份措施，应从以下三个方面进行测试：
- ①故障恢复：测试整个 ERP 系统是否存在单点故障；任何一台设备失效时，能否按照预定义的规则实现快速切换；是否采用磁盘镜像技术，实现主机系统到磁盘系统的高速连接。
 - ②数据备份：ERP 系统关键业务是否具备必要的双机热备或磁盘镜像等热备份机制；对于整个 ERP 业务，是否提供外部存储器备份和恢复机制，保证系统能够根据备份策略恢复到指定时间的状态。

③容灾备份：ERP 系统是否建立异地容灾备份中心，当主中心发生灾难性事件时，可由备份中心接管所有业务；备份中心是否有足够的带宽确保与主中心的数据同步，是否有足够处理能力来接管主中心的业务，能否确保快速可靠地与主中心的应用切换。

本问题考查故障恢复与容灾备份措施测试内容。

针对故障恢复与容灾备份措施的安全性测试一般包括故障恢复、数据备份与容灾备份三个测试点，其包含的基本测试内容为：

①故障恢复：整个系统是否存在单点故障；对于关键应用系统，当任何一台设备失效时，能否按照预定义的规则实现快速切换；是否采用磁盘镜像技术，实现主机系统到磁盘系统的高速连接；

②数据备份：对于关键业务，是否具备必要的双机热备或磁盘镜像等热备份机制；对于所有业务，是否提供外部存储器备份和恢复机制，保证系统能够根据备份策略恢复到指定时间的状态；

③容灾备份：是否建立异地容灾备份中心，当主中心发生灾难性事件时，可由备份中心接管所有业务；备份中心是否有足够的带宽确保与主中心的数据同步，是否有足够处理能力来接管主中心的业务，能否确保快速可靠地与主中心的应用切换。

【问题 2】

数据库服务器中目前主要存储 ERP 系统业务数据，后续还需要存储企业网站相关数据，当前 ERP 系统的用户认证方式包含口令认证方式，相应的用户权限和口令也存储在数据库二维表中。针对上述实际情况，参与测试的李工认为在对数据库权限进行测试时，除数据库账号保护及权限设置相关的常规测试外，还必须对敏感数据加密保护及对数据库访问方式进行相应测试。请用 200 字以内文字，对敏感数据加密保护和数据库访问方式两个方面的测试内容进行简要说明。

敏感数据加密保护和数据库访问方式的测试内容为：

①敏感数据的加密保护：由于 ERP 系统的用户权限和口令存储在数据库内，因此需要测试相应敏感数据是否采用加密算法进行加密保护。

②数据库访问方式测试：是否为不同应用系统或业务设置不同的专门用户用于数据库访问，应杜绝在代码中使用超级用户及默认密码对数据库进行访问。

本题考查考生对数据库权限测试内容的理解。

根据本题说明，ERP 系统的用户权限和口令信息存储在数据库中，因此需要测试相应敏感数据是否采用加密算法进行加密保护；而数据库中还需要存储 ERP 系统之外的其他系统业务数据，因此应为不同应用系统或业务设置不同的专门用户用于数据库访问，且应杜绝在代码中使用超级用户及默认密码对数据库进行访问。

【问题 3】

为对抗来自外网或内网的主动攻击，系统通常会采用多种安全防护策略，请给出四种常见的安全防护策略并进行简要解释。结合一种在图 4-1 中明确标识出的安全防护策略机制，说明针对该机制的安全测试应包含哪些基本测试点。

安全防护策略是软件系统对抗攻击的主要手段，常见的安全防护策略有以四种：

①安全日志：安全日志用于记录非法用户的登录名称、操作时间及内容等信息，以便发现问题并提出解决措施；安全日志仅记录相关信息，不对非法行为做出主动反应，属于被动防护的策略；

②入侵检测系统：入侵检测系统是一种主动的网络安全防护措施，从系统内部或各种网络资源中主动采集信息，从中分析可能的网络入侵或攻击，通常入侵检测系统还应对入侵行为做出紧急响应；

③漏洞扫描：漏洞扫描是对软件系统及网络系统进行与安全相关的检测，以找出安全隐患和可被黑客利用的漏洞；

④隔离防护：隔离防护是将系统中的安全部分与非安全部分进行隔离的措施，主要的技术手段有防火墙和隔离网闸等，其中防火墙主要用于内网和外网的逻辑隔离，而网闸则主要用于实现内网和外网的物理隔离。

图 4-1 中明确标识出的安全防护策略机制为防火墙和入侵检测系统。（可选择防火墙或入侵检测系统中的任一机制描述相应安全测试的测试点）

针对防火墙的测试点：

- (1) 是否支持交换和路由两种工作模式；
- (2) 是否支持对 FTP、HTTP、SMTP 等服务类型的访问控制；
- (3) 是否考虑防火墙的冗余设计；

- (4) 是否支持对日志的统计分析功能，日志是否可存储在本地或网络数据库中；
- (5) 对防火墙或受保护内网的非法攻击，是否提供多种告警方式和多种级别的告警。针对入侵检测系统的测试点：
- ① 能否在检测到入侵事件时，自动执行切断服务、记录入侵过程、报警等动作；
 - ② 是否支持攻击特征信息的集中式发布和攻击取证信息的分布式上载；
 - ③ 能否提供多种方式对监视引擎和检测特征进行定期更新；
 - ④ 内置的网络能否使用状态监控工具或网络监听工具。

本题考查基本的安全防护策略的相关知识。

安全防护策略是软件系统对抗来自外网或内网的主动攻击的主要手段，常见的安全防护策略包括安全日志、入侵检测、隔离防护与漏洞扫描等，本题首先要求考生对常见安全防护策略的有基本的理解，了解其基本原理与主要措施。

图 4-1 中明确标识出的安全防护策略机制为防火墙和入侵检测系统，考生可自主选择其中一种，描述对其进行安全性测试时需要完成的测试点。

试题五

【说明】

软件在机载设备中的运用越来越广泛，驻留于机载设备中的嵌入式软件失效会产生灾难性后果，一般要求其具有较高的可靠性，因此，软件可靠性测试对机载软件至关重要。

【问题 1】

解释软件可靠性的含义及影响软件可靠性的主要因素。

软件可靠性是在规定的条件下，在规定的时间内，软件不引起系统失效的概率。影响软件可靠性的主要因素包括：运行剖面、软件规模、软件内部结构、软件的开发方法和开发环境、软件的可靠性投入等。

本题考查软件可靠性知识与应用。

软件可靠性 (software reliability) 是软件产品在规定的条件下和规定的时间区间完成规定功能的能力。规定的条件是指直接与软件运行相关的使用该软件的计算机系统的状态和软件的输入条件，或统称为软件运行时的外部输入条件；规定的时间区间是指软件的实际运行时间区间；规定功能是指为提供给定的服务，软件产品所必须具备的功能。软件可靠性不但与软件存在的缺陷和 (或) 差错有关，而且与系统输入和系统使用有关。软件可靠性的概率度量称软件可靠度。

现代软件工程认为，质量形成于过程，过程的因素是影响软件可靠性的主要因素。从软件开发的角，影响软件可靠性的因素有软件规模、运行剖面、软件内部结构、软件可靠性管理、软件可靠性测试与投入、软件可靠性设计技术、软件开发方法与软件开发环境等。一般地，软件规模越大，其可靠性问题就越多。

失效概率是软件从运行开始到某一时刻 t 为止，出现失效的概率；

可靠度是软件系统在规定的条件下，规定的时间内不发生失效的概率；

平均无失效时间指软件运行后，到下一次出现失效的平均时间。

软件可靠度 $R(t)$ 与软件失效概率之间的关系为 $R(t)=1-F(t)$ ，题目中 $t=1000$ ， $F(t)=0.0012$ ，因此 $R(t)=1-0.0012=0.9988$ ，显然不符合该嵌入式软件设计要求的 0.9999 无失效概率。

【问题 2】

可靠性评价时，经常使用的定量指标包括失效概率、可靠度和平均无失效时间 (MTTF)，请分别解释其含义。

失效概率是软件从运行开始到某一时刻 t 为止，出现失效的概率；

可靠度是软件系统在规定的条件下，规定的时间内不发生失效的概率；

平均无失效时间指软件运行后，到下一次出现失效的平均时间。

本题考查软件可靠性知识与应用。

软件可靠性 (software reliability) 是软件产品在规定的条件下和规定的时间区间完成规定功能的能力。规定的条件是指直接与软件运行相关的使用该软件的计算机系统的状态和软件的输入条件，或统称为软件运行时的外部输入条件；规定的时间区间是指软件的实际运行时间区间；规定功能是指为提供给定的服务，软件产品所必须具备的功能。软件可靠性不但与软件存在的缺陷和 (或) 差错有关，而且与系统输入和系统使用有关。软件可靠性的概率度量称软件可靠度。

现代软件工程认为，质量形成于过程，过程的因素是影响软件可靠性的主要因素。从软件开发的角度，影响软件可靠性的因素有软件规模、运行剖面、软件内部结构、软件可靠性管理、软件可靠性测试与投入、软件可靠性设计技术、软件开发方法与软件开发环境等。一般地，软件规模越大，其可靠性问题就越多。

失效概率是软件从运行开始到某一时刻 t 为止，出现失效的概率；

可靠度是软件系统在规定的条件下，规定的时间内不发生失效的概率；

平均无失效时间指软件运行后，到下一次出现失效的平均时间。

软件可靠度 $R(t)$ 与软件失效概率之间的关系为 $R(t)=1-F(t)$ ，题目中 $t=1000$ ， $F(t)=0.0012$ ，因此 $R(t)=1-0.0012=0.9988$ ，显然不符合该嵌入式软件设计要求的 0.9999 无失效概率。

【问题 3】

对某嵌入式软件，设计要求其可靠度为 1000 小时无失效概率 99.99%。经实测得出其失效概率函数 $F(1000)=0.0012$ ，问该软件是否符合设计可靠性要求，并说明原因。

不符合可靠度要求。

软件可靠度 $R(t)$ 和软件失效概率之间的关系为 $R(t)=1-F(t)$ 。
 $R(1000)=1-F(1000)=1-0.0012=0.9988=99.88\%$, $99.88\%<99.99\%$, 因此不符合设计软件可靠度要求。

本题考查软件可靠性知识与应用。

软件可靠性 (software reliability) 是软件产品在规定的条件下和规定的时间区间完成规定功能的能力。规定的条件是指直接与软件运行相关的使用该软件的计算机系统的状态和软件的输入条件, 或统称为软件运行时的外部输入条件; 规定的时间区间是指软件的实际运行时间区间; 规定功能是指为提供给定的服务, 软件产品所必须具备的功能。软件可靠性不但与软件存在的缺陷和(或)差错有关, 而且与系统输入和系统使用有关。软件可靠性的概率度量称软件可靠度。

现代软件工程认为, 质量形成于过程, 过程的因素是影响软件可靠性的主要因素。从软件开发的角度, 影响软件可靠性的因素有软件规模、运行剖面、软件内部结构、软件可靠性管理、软件可靠性测试与投入、软件可靠性设计技术、软件开发方法与软件开发环境等。一般地, 软件规模越大, 其可靠性问题就越多。

失效概率是软件从运行开始到某一时刻 t 为止, 出现失效的概率;

可靠度是软件系统在规定的条件下, 规定的时间内不发生失效的概率;

平均无失效时间指软件运行后, 到下一次出现失效的平均时间。

软件可靠度 $R(t)$ 与软件失效概率之间的关系为 $R(t)=1-F(t)$, 题目中 $t=1000$, $F(t)=0.0012$, 因此 $R(t)=1-0.0012=0.9988$, 显然不符合该嵌入式软件设计要求的 0.9999 无失效概率。