

## 2005年下半年软件设计师上午试题

●阵列处理机属于\_\_(1)\_\_\_计算机。

- (1) A.SISD      B SIMD      C.MISD      D.MIMD

●采用\_\_(2)\_\_\_不能将多个处理机互连构成多处理机系统。

- (2) A.STD 总线      B.交叉开关      C.PCI 总线      D.Centronic 总线

●某计算机系统的可靠性结构是如下图所示的双重串并联结构，若所构成系统的每个部件的可靠度为 0.9，即  $R=0.9$ ，则系统的可靠度为\_\_(3)\_\_\_。

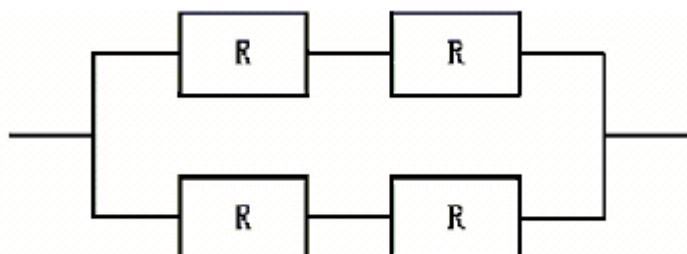


图 1

- (3) A.0.9997      B.0.9276      C.0.9739      D.0.6561

●若每一条指令都可以分解为取指、分析和执行三步。已知取指时间  $t_{\text{取指}}=5\Delta t$ , 分析时间  $t_{\text{分析}}=2\Delta t$ , 执行时间  $t_{\text{执行}}=5\Delta t$ 。如果按顺序方式从头到尾执行完 500 条指令需\_\_(4)\_\_\_ $\Delta t$ 。如果按照[执行] $k$ 、[分析] $k+1$ 、[取指] $k+2$  重叠的流水线方式执行指令，从头到尾执行完 500 条指令需\_\_(5)\_\_\_ $\Delta t$ 。

- (4) A.5590      B.5595      C.6000      D.6007

- (5) A.2492      B.2500      C.2510      D.2515

●在开发一个系统时，如果用户对系统的目标是不很清楚，难以定义需求，这时最好使用\_\_(6)\_\_\_。

- (6) A.原型法      B.瀑布模型      C. V-模型      D.螺旋模型

●应该在\_\_(7)阶段制定系统测试计划。

- (7) A.需求分析    B.概要设计    C.详细设计    D.系统测试

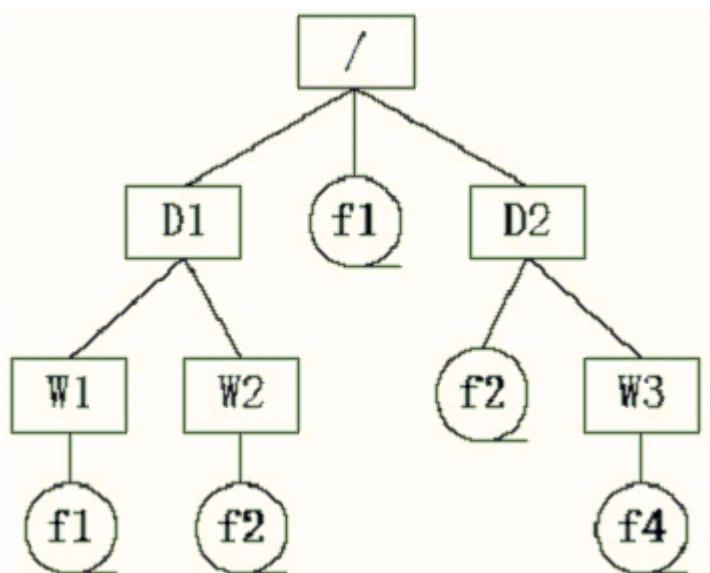
●已经发布实施的标准（包括已确认或修改补充的标准），经过实施一定时期后，对其内容再次审查，以确保其有效性和先进性，其周期一般不超过\_\_(8)\_\_年。

- (8) A. 1 B. 3 C. 5 D. 7

●\_\_(9)\_\_不需要登记或标注版权标记就能得到保护

- (9) A.专利权    B.商标权    C.著作权    D.财产权

●如图 2 所示的树型文件中，方框表示目录，圆圈表示文件，“/”表示路径的分隔符，“/”路径之首表示根目录。图 2 中，\_\_(10)\_\_。



假设当前目录是 D1，进程 A 以如下两种方式打开文件 f1：方式① fd1 = open("\_\_(11)\_\_/f1", o\_RDONLY); 方式② fd1 = open("/D1/W1/f1", o\_RDONLY);

其中，方式 1 的工作效率比方式 2 的工作效率高，因为采用方式 1 的文件系统 \_\_\_\_(12)\_\_。

- (10) A. 子目录 W2 中文件 f2 和子目录 D2 中文件 f2 是完全相同的  
 B. 子目录 W2 中文件 f2 和子目录 D2 中文件 f2 是不相同的  
 C. 子目录 W2 中文件 f2 和子目录 D2 中文件 f2 是可能相同也可能不相同  
 D. 树型文件系统中不允许出现相同名字的文件

- (11) A./D1/W1    B.D1/W1    C.W1    D.f1

- (12) A.可以直接访问根目录下的文件 f1  
B.可用从当前路径开始查找需要访问的文件 f1  
C.只需要访问一次磁盘，就可以读取文件 f1，而方式 2 需要两次  
D.只需要访问一次磁盘，就可以读取文件 f1，而方式 2 需要三次

●正在开发的软件项目可能存在一个未被发现的错误，这个错误出现的概率是 0.5%，给公司造成的损失将是 1 000 000 元，那么这个错误的风险曝光度(risk exposure) 是\_\_\_\_(13)\_\_\_\_ 元。

- (13) A.5 000 000      B.50 000      C.5 000      D.500

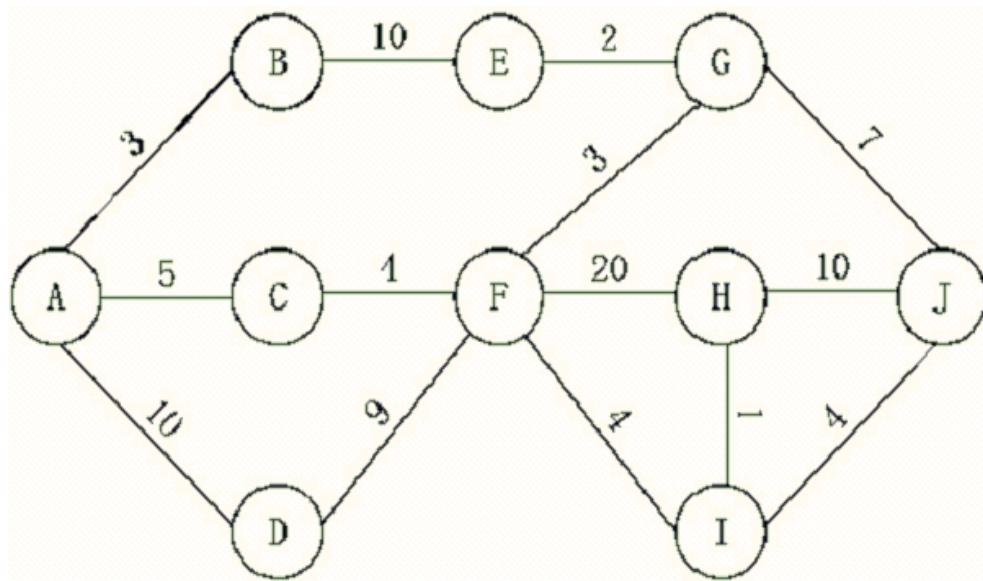
●代码走查(code walkthrough) 和代码审查(code inspection) 是两种不同的代码评审方法，这两种方法的主要区别是\_\_\_\_(14)\_\_\_\_。

- (14) A.在代码审查中由编写代码的程序员来组织讨论，而在代码走查中由高级管理人员来领导评审小组的活动  
B.在代码审查中只检查代码中是否有错误，而在代码走查中还要检查程序与设计文档的一致性  
C.在代码走查中只检查程序的正确性，而在代码审查中还要评审程序员的编程能力和工作业绩  
D.代码审查是一种正式的评审活动，而代码走查的讨论过程是非正式的

●在软件项目管理中可以使用各种图形工具来辅助决策，下面对 Gantt 图的描述不正确的是\_\_\_\_(15)\_\_\_\_。

- (15) A.Gantt 图表现各个活动的顺序和它们之间的因果关系  
B.Gantt 图表现哪些活动可以并行进行  
C.Gantt 图表现了各个活动的起始时间  
D.Gantt 图表现了各个活动完成的进度

●在活动图中，结点表示项目中各个工作阶段的里程碑，连接各个结点的边表示活动，边上的数字表示活动持续的时间。在下面的活动图中，从 A 到 J 的关键路径是\_\_\_\_(16)\_\_\_\_，关键路径长度是\_\_\_\_(17)\_\_\_\_，从 E 开始的活动启动的最早时间是\_\_\_\_(18)\_\_\_\_。



(16) A.ABEGJ      B.ADFHJ      C.ACFGJ      D.ADFIJ

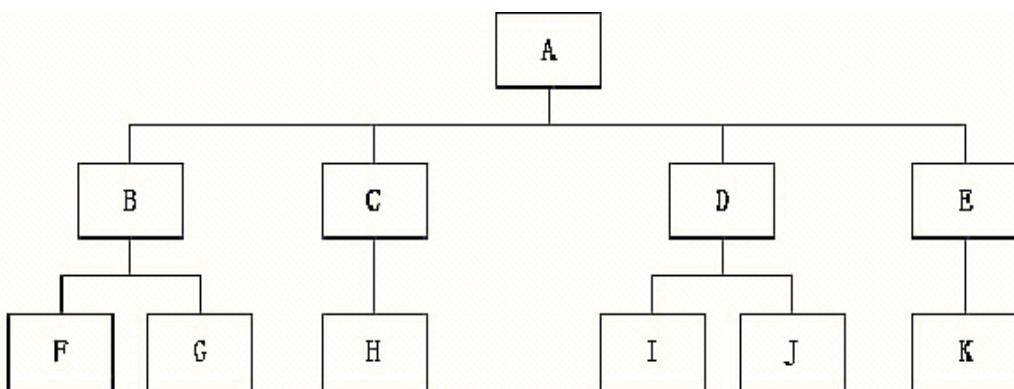
(17) A.22      B.49      C.19      D.35

(18) A.10      B.12      C.13      D.15

● 一个故障已经被发现，而且也被排除了，为了检查修改是否引起了其它故障，这时应该进行\_\_(19)\_\_。

(19) A.程序走查      B.退化测试      C.软件评审      D.接收测试

● 一个系统的模块结构图如下所示，用{×,×,×}表示这个系统的测试模块组合。下面的选项中\_\_(20)\_\_ 表示自顶向下的测试，\_\_(21)\_\_ 表示三明治式测试。



(20) A.{A}{A,B,C,D,E}{A,B,C,D,E,F,G,H,I,J,K}

B.{F}{G}{H}{I}{J}{K}{B,F,G}{C,H}{D,I,J}{E,K}{A,B,C,D,E,F,G,H,I,J,K}

C.{K}{J}{I}{H}{G}{F}{B}{C}{D}{E}{A,B,C,D,E,F,G,H,I,J,K}

D.{A}{F}{G}{H}{I}{J}{K}{B,F,G}{C,H}{D,I,J}{E,K}{A,B,C,D,E,F,G,H,I,J,K}

- (21) A.{A}{A,B,C,D,E}{A,B,C,D,E,F,G,H,I,J,K}  
B.{F}{G}{H}{I}{J}{K}{B,F,G}{C,H}{D,I,J}{E,K}{A,B,C,D,E,F,G,H,I,J,K}  
C.{K}{J}{I}{H}{G}{F}{B}{C}{D}{E}{A,B,C,D,E,F,G,H,I,J,K }  
D.{A}{F}{G}{H}{I}{J}{K}{B,F,G}{C,H}{D,I,J}{E,K}{ A,B,C,D,E,F,G,H,I,J,K }

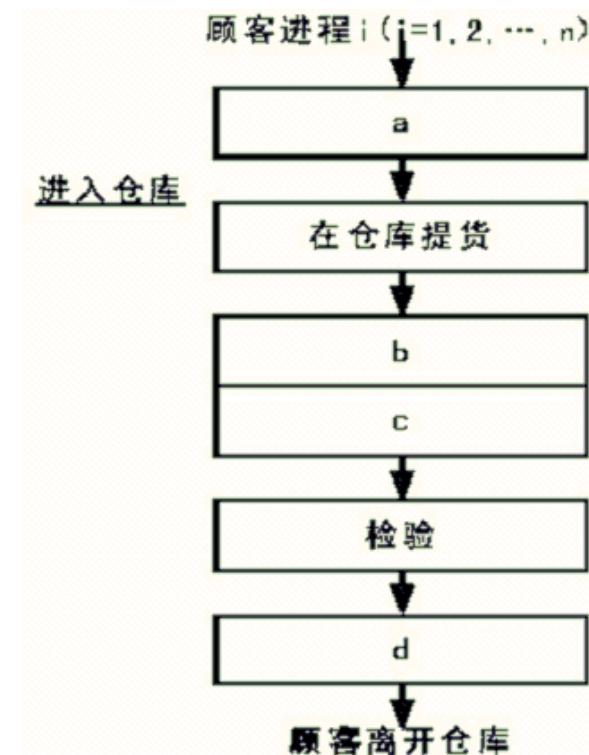
● 某公司服务器上存储了大量的数据，员工使用服务器前首先必须登录。为了保证安全，使用认证技术\_\_\_\_(22)\_\_\_\_。为保证传输效率，使用\_\_\_\_(23)\_\_\_\_加密算法对传输的数据进行加密。

- (22) A.对登录系统的用户身份进行认可      B.保护数据在传输中的机密性  
C.保证数据在传输中不被非法修改      D.防止登录信息被泄漏出去  
(23) A.RSA      B.ECC      C.MD4      D.3DES

● \_\_\_\_ (24)\_\_\_\_ 不能保障公司内部网络边界的安全

- (24) A.在公司网络与 Internet 或外界其它接口处设置防火墙  
B.公司以外网络上用户要访问公司网时，使用认证授权系统  
C.禁止公司员工使用公司外部的电子邮件服务器  
D.禁止公司内部网络的用户私自设置拨号上网

● 某仓库有两名发货员，一名审核员。当顾客提货时，只要发货员空闲，允许顾客进入仓库提货，顾客离开时，审核员检验顾客提货是否正确。其工作流程如下图所示。为了利用 PV 操作正确地协调他们之间的操作，设置了两个信号量 S1 和 S2，且 S1 的初值为 2，S2 的初值为 1。图中的 a 应填写\_\_\_\_(25)\_\_\_\_；图中的 b、c 和 d 应分别填写\_\_\_\_(26)\_\_\_\_。



(25) A.P(S1) B.P(S2) C.V(S1) D.V(S2)

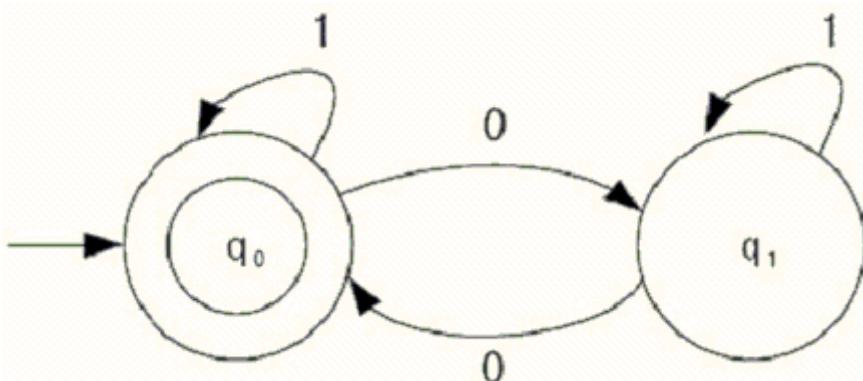
(26) A.P(S2)、V(S2)和 V(S1) B.P(S1)、V(S1)和 V(S2)

C.V(S1)、P(S2)和 V(S2) D.V(S2)、P(S1)和 V(S1)

● 编译程序进行词法分析时不能\_\_\_\_(27)\_\_\_\_。

- (27) A.过滤源程序中的注释      B. 扫描源程序并识别记号  
 C. 指出出错行号      D. 查出拼错的保留字（关键字）

● 某一确定有限自动机(DFA) 的状态转换图如下图所示，该 DFA 接受的字符串集是\_\_\_\_(28)\_\_\_\_，与之等价的正规式是\_\_\_\_(29)\_\_\_\_。



- (28) A.以 1 开头的二进制代码串组成的集合  
 B.以 1 结尾的二进制代码串组成的集合  
 C.包含偶数个 0 的二进制代码串组成的集合  
 D.包含奇数个 0 的二进制代码串组成的集合

(29) A. $1^*0(0|1)^*$       B. $((0|1^*)^*1^*)^*$       C. $1^*((0|1)0)^*$       D. $(1^*(01^*0))^*$

●下列关于静态多态实现重载的描述中正确的是\_\_\_\_(30)\_\_\_\_。

- (30) A.重载的方法（函数）的方法名（函数名）可以不同  
 B.重载的方法（函数）必须在返回值类型以及参数类型上有所不同  
 C.重载的方法（函数）必须在参数顺序或参数类型上有所不同  
 D.重载的方法（函数）只需在返回值类型上有所不同

●程序设计语言提供了基本类型及其相关的操作，而\_\_\_\_(31)\_\_\_\_ 则允许开发者自定义一种新的类型及其相关操作。

- (31) A.对象      B.实例      C.类      D.引用

● \_\_\_\_ (32) \_\_\_\_ 表示了对象间"is-a" 的关系。

- (32) A.组合      B.引用      C.聚合      D.继承

●在关系代数运算中，关系 S、SP 和 R 如下表所示。若先\_\_\_\_(33)\_\_\_\_，则可以从 S 和 SP 获得 R。其对应的关系表达式为\_\_\_\_(34)\_\_\_\_。如下的 SQL 语句可以查询销售总量大于 1000 的部门号。 Select 部门名 From S。Where 部门号 in (Select 部门号 From SP Group by \_\_\_\_ (35) \_\_\_\_ )关系表 S 关系表 SP 关系表 R

部门号	部门名
010	家电部
021	百货部
030	食品部

部门号	商品号	销售量
010	01020210	500
010	01020211	780
010	01020212	990

部门名	商品号	销售量
家电部	01020210	500
家电部	01020211	780
家电部	01020212	990

035	五金部
-----	-----

021	02110200	580
025	02520100	1290
035	03530311	4680

- (33) A.对 S 进行选择运算，再与 SP 进行自然连接运算  
 B.对 S 进行选择运算，再与 SP 进行自然连接运算，最后进行投影运算  
 C.对 S 和 SP 进行笛卡尔积运算，再对运算结果进行投影运算  
 D.分别对 S 和 SP 进行投影运算，再对运算结果进行笛卡尔积运算

- (34) A.  $\sigma_{1=010^*}(S) \bowtie SP$       B.  $\pi_{1,3,4}(\sigma_{1=010^*}(S) \bowtie SP)$   
 C.  $\pi_{2,3,4}(\sigma_{1=010^*}(S) \bowtie SP)$       D.  $\pi_{2,3,4}(S \times SP)$

- (35) A.部门号 where sum( 销售量)>1000  
 B.部门号 having sum( 销售量)>1000  
 C.商品号 where sum( 销售量)>1000  
 D.商品号 having sum( 销售量)>1000

●给定关系 R (A1, A2, A3, A4) 上的函数依赖集 F={A1A3->A2, A2->A3}，R 候选关键字为\_\_\_\_(36)\_\_\_\_。分解  $\rho = \{(A1, A2), (A1, A3)\}$  \_\_\_\_ (37)\_\_\_\_。(36) A.A1A3 B.A1A2A3 C.A1A3A4 D.A2 和 A1A3

- (37) A.是无损联接的      B. 是保持函数依赖的  
 C.既是无损联接又保持函数依赖      D. 既是有损联接又不保持函数依赖

●已知某二叉树的中序、层序序列分别为 DBAFCE 、FDEBCA，则该二叉树的后序序列为\_\_\_\_(38)\_\_\_\_。

- (38) A.BCDEAF      B.ABDCEF      C.DBACEF      D.DABECF

●在二叉树的顺序存储中，每个结点的存储位置与其父结点、左右子树结点的位置都存在一个简单的映射关系，因此可与三叉链表对应。若某二叉树共有 n 个结点，采用三叉链表存储时，每个结点的数据域需要 d 个字节，每个指针域占用 4 个字节，若采用顺序存储，则最后一个结点下标为 k (起始下标为 1)，那么\_\_\_\_(39)\_\_\_\_时采用顺序存储更节省空间。

- (39) A.d<12n/(k-n)      B.d>12n/(k-n)      C.d<12n/(k+n)      D.d>12n/(k+n)

●简单无向图的邻接矩阵是对称的，可以对其进行压缩存储。若无向图 G 有 n 个节点。若无向图 G 有 n 个节点，其邻接矩阵为 A[1..n,1..n]，且压缩存储在 B[1..k] 中，则 k 的值至少为\_\_\_\_(40)\_\_\_\_。若按行压缩存储对称矩阵的上三角元素，则当 n 等于 10 时，边(V6, V3)的信息存储在 B[\_\_\_\_(41)\_\_\_\_] 中。

(40) A. $n(n+1)/2$       B. $n^2/2$       C. $(n-1)(n+1)/2$       D. $n(n-1)/2$

(41) A.18 B.19 C.20 D.21

●以下内容中，\_\_\_\_(42)\_\_\_\_ 应写入操作手册。

- (42) A.描述系统对各种输入数据的处理方法
- B.说明系统升级时厂商提供的服务
- C.描述系统处理过程的各个界面
- D.说明系统各部分之间的接口关系

● 某软件工程项目各开发阶段工作量的比例如下表所示

需求分析	概要设计	详细设计	编码	测试
0.29	0.13	0.17	0.10	0.31

假设当前已处于编码阶段，3000 行程序已完成了 1200 行，则该工程项目开发进度已完成的比例是\_\_\_\_(43)\_\_\_\_。

(43) A.29%      B.45%      C.59%      D.63%

●在 11 个元素的有序表 A[1..11] 中进行折半查找 ( L(low+high)/2 ) 查找元素 A[11] 时，被比较的元素的下标依次是\_\_\_\_(44)\_\_\_\_。

(44) A.6, 8, 10, 11      B.6, 9, 10, 11      C.6, 7, 9, 11      D.6, 8, 9, 11

●关于维护软件所需的成本，以下叙述正确的是\_\_\_\_(45)\_\_\_\_。

- (45) A.纠正外部和内部设计错误比纠正源代码错误需要更大的成本
- B.与需求定义相比，源代码的文字量大得多，所以源代码的维护成本更高
- C.用户文档需要经常更新，其维护成本超过了纠正设计错误的成本

D. 需要定义的错误会在设计时被发现并纠正，因此需求定义纠错的成本小于源代码纠错的成本

●由元素序列(27,16,75,38,51)构造平衡二叉树，则首次出现的最小不平衡子树的根（即离插入结点最近且平衡因子的绝对值为2的结点）为\_\_\_\_(46)\_\_\_\_。

- (46) A.27      B.38      C.51      D.75

●若排序前后关键字相同的两个元素相对位置不变，则称该排序方法是稳定的。\_\_\_\_(47)\_\_\_\_排序是稳定的。

- (47) A.归并      B.快速      C.希尔      D.堆

●若对象A可以给对象B发送消息，那么\_\_\_\_(48)\_\_\_\_。

- (48) A.对象B可以看见对象A      B.对象A可看见对象B  
C.对象A、B相互不可见      D.对象A、B相互可见

●类描述了一组对象共同的特性，下列叙述中正确的是\_\_\_\_(49)\_\_\_\_。

- (49) A.类本身不能具有变量  
B.对象具有类定义的所有变量的一份拷贝  
C.对象间不能共享类定义的变量  
D.可通过类名访问静态变量(类变量)

●\_\_\_\_(50)\_\_\_\_反映了类间的一种层次关系，而\_\_\_\_(51)\_\_\_\_反映了一种整体与部分的关系。

- (50) A.继承      B.组合      C.封装      D.多态  
(51) A.继承      B.组合      C.封装      D.多态

●下列叙述中正确的是\_\_\_\_(52)\_\_\_\_。

- (52) A.面向对象程序设计语言都不支持过程化的程序设计  
B.面向对象系统只可采用面向对象程序设计语言实现  
C.某些过程化程序设计语言也可实现面向对象系统  
D.面向对象程序设计语言不支持对成员变量的直接访问

● 设求解某问题的递归算法如下：

```

F(int n){
    if n==1 {
        Move(1)
    }else{
        F(n-1);
        Move(n);
        F(n-1);
    }
}

```

求解该算法的计算时间时,仅考虑算法 Move 所做的计算为主要计算,且 Move 为常数级算法。则算法 F 的计算时间 T(n)的递推关系式为\_\_\_\_(53)\_\_\_\_ ; 设算法 Move 的计算时间为 k, 当 n=4 时, 算法 F 的计算时间为\_\_\_\_(54)\_\_\_\_ 。

- (53) A.T(n)=T(n-1)+1      B.T(n)=2T(n-1)      C.T(n)=2T(n-1)+1      D.T(n)=2T(n+1)+1  
 (54) A.14k                    B.15k                    C.16k                    D.17k

●利用贪心法求解 0/1 背包问题时, \_\_\_\_ (55)\_\_\_\_ 能够确保获得最优解。用动态规划方法求解 0/1 背包问题时,将"用前 i 个物品来装容量是 X 的背包"的 0/1 背包问题记为 KNAP(1,i,X), 设  $f_i(X)$  是 KNAP(1,i,X) 最优解的效益值, 第 j 个物品的重量和放入背包后取得效益值分别为 W 和  $p(j=1 \sim n)$  。则依次求解  $f_0(X)$  、  $f_1(X)$  、 ... 、  $f_n(X)$  的过程中使用的递推关系式为\_\_\_\_(56)\_\_\_\_ 。

- (55) A.优先选取重量最小的物品      B. 优先选取效益最大的物品  
 C. 优先选取单位重量效益最大的物品      D.没有任何准则  
 (56) A. $f_i(X)=\min\{f_i(X), f_i(X)+p\}$       B. $f_i(X)=\max\{f_i(X), f_i(X-W_i)+p\}$   
 C. $f_i(X)=\min\{f_i(X-W_i), f_i(X-W_i)+p\}$       D. $f_i(X)=\max\{f_i(X-W_i), f_i(X)+p\}$

● \_\_\_\_ (57)\_\_\_\_ 标准中定义了 MP3 音乐标准。

- (57) A.JPG      B.MPEG-1      C.MPEG-2      D.MPEG-3

● 使用 200DPI 的扫描分辨率扫描一幅  $2 \times 2.5$  英寸的黑白图像, 可以得到一幅\_\_(58)\_\_\_ 像素的图像。

- (58) A.  $200 \times 2$       B.  $2 \times 2.5$       C.  $400 \times 500$       D.  $800 \times 1000$

● 要在网络上发布彩色动画文件, 可以采用的存储格式是\_\_(59)\_\_\_。

- (59) A. BMP      B. JPEG      C. MP3      D. GIF

● 为保证用户在网络上边下载边观看视频信息, 需要采用\_\_(60)\_\_\_ 技术。

- (60) A. 流媒体      B. 数据库      C. 数据采集      D. 超链接

● 通过代理服务器使内部局域网中的客户机访问 Internet 时, \_\_(61)\_\_\_ 不属于代理服务器的功能。      (61) A. 共享 IP 地址      B. 信息缓存      C. 信息转发      D. 信息加密

● 下列(62)设备可以隔离 ARP 广播帧。

- (62) A. 路由器      B. 网桥      C. 以太网交换机      D. 集线器

● 在 Windows 系统中, \_\_(63)\_\_\_ 不是网络服务组件。

- (63) A. RAS      B. HTTP      C. IIS      D. DNS

● 在 OSI 参考模型中, 数据链路层处理的数据单位是\_\_(64)。

- (64) A. 比特      B. 帧      C. 分组      D. 报文

● 在 OGSA 标准中定义了\_\_(65) 的概念, 它提供一组遵守特定的约定并定义明确的接口, 是实体之间产生、管理和交换信息的机制。

- (65) A. Object      B. Grid Service      C. Web Service      D. XML

● MIDI enables people to use\_\_(66) computers and electronic musical instruments. There are actually three components to MIDI, the communications "\_\_(67)", the Hardware Interface and a distribution \_\_(68) called "Standard MIDI Files". In the context of the WWW, the most interesting component is the\_\_(69) Format. In principle, MIDI files contain sequences of MIDI

Protocol messages. However, when MIDI Protocol \_\_\_(70) are stored in MIDI files, the events are also time-stamped for playback in the proper sequence. Music delivered by MIDI files is the most common use of MIDI today.

- (66) A.personal      B.electronic      C. multimedia      D. network
- (67) A.device      B.protocol      C. network      D. controller
- (68) A.format      B.text      C. wave      D. center
- (69) A.Video      B.Faxmail      C. Graphic      D. Audio
- (70) A.messages      B.packets      C.frame      D.information

● Certificates are \_\_\_(71) documents attesting to the \_\_\_(72) of a public key to an individual or other entity. They allow verification of the claim that a given public key does in fact belong to a given individual. Certificates help prevent someone from using a phony key to \_\_\_(73) someone else. In their simplest form, Certificates contain a public key and a name. As commonly used, a certificate also contains an \_\_\_(74) date, the name of the CA that issued the certificate, a serial number, and perhaps other information. Most importantly, it contains the digital \_\_\_(75) of the certificate issuer. The most widely accepted format for certificates is X.509 , thus, Certificates can be read or written by any application complying with X.509.

- (71) A.text      B.data      C.digital      D.structured
- (72) A.connecting      B.binding      C.composing      D.conducting
- (73) A.impersonate      B.personate      C.damage      D.control
- (74) A.communication      B.computation      C.expectation      D.expiration
- (75) A.signature      B.mark      C.stamp      D.hypertext