

希赛网, 专注于软考、PMP、通信考试的专业 IT 知识库和在线教育平台。希赛网在线题库, 提供历年考试真题、模拟试题、章节练习、知识点练习、错题本练习等在线做题服务, 更有能力评估报告, 让你告别盲目做题, 针对性地攻破自己的薄弱点, 更高效的备考。

希赛网官网: <http://www.educity.cn/>

希赛网软件水平考试网: <http://www.educity.cn/rk/>

希赛网在线题库: <http://www.educity.cn/tiku/>

2014 年下半年软设综合知识真题答案与解析: <http://www.educity.cn/tiku/tp18944.html>

## 2014 年下半年软件设计师考试上午真题 (参考答案)

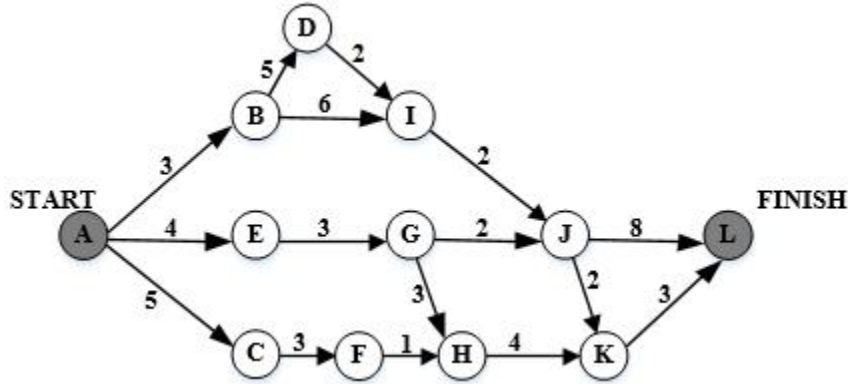
- 三总线结构的计算机总线系统由\_\_ (1) \_\_组成。
  - (1) A. CPU 总线、内存总线和 IO 总线
  - B. 数据总线、地址总线和控制总线
  - C. 系统总线、内部总线和外部总线
  - D. 串行总线、并行总线和 PCI 总线
- 计算机采用分级存储体系的主要目的是为了解决\_\_ (2) \_\_问题。
  - (2) A. 主存容量不足
  - B. 存储器读写可靠性
  - C. 外设访问效率
  - D. 存储容量、成本和速度之间的矛盾
- 属于 CPU 中算术逻辑单元的部件是\_\_ (3) \_\_。
  - (3) A. 程序计数器
  - B. 加法器
  - C. 指令寄存器
  - D. 指令译码器
- 内存按字节编址从 A5000H 到 DCFFFH 的区域其存储容量为\_\_ (4) \_\_。
  - (4) A. 123KB
  - B. 180KB
  - C. 223KB
  - D. 224KB
- 以下关于 RISC 和 CISC 的叙述中, 不正确的是\_\_ (5) \_\_。

- (5) A. RISC 通常比 CISC 的指令系统更复杂  
B. RISC 通常会比 CISC 配置更多的寄存器  
C. RISC 编译器的子程序库通常要比 CISC 编译器的子程序库大得多  
D. RISC 比 CISC 更加适合 VLSI 工艺的规整性要求
- Flynn 分类法基于信息流特征将计算机分成 4 类, 其中\_\_(6)\_\_只有理论意义而无实例。
- (6) A. SISD  
B. MISD  
C. SIMD  
D. MIMD
- 网络系统中, 通常把\_\_(7)\_\_置于 DMZ 区。
- (7) A. 网络管理服务器  
B. Web 服务器  
C. 入侵检测服务器  
D. 财务管理服务器
- 以下关于拒绝服务攻击的叙述中, 不正确的是\_\_(8)\_\_。
- (8) A. 拒绝服务攻击的目的是使计算机或者网络无法提供正常的服务  
B. 拒绝服务攻击是不断向计算机发起请求来实现的  
C. 拒绝服务攻击会造成用户密码的泄漏  
D. DDoS 是一种拒绝服务攻击形式
- \_\_(9)\_\_不是蠕虫病毒。
- (9) A. 熊猫烧香  
B. 红色代码  
C. 冰河  
D. 爱虫病毒
- 甲公司接受乙公司委托开发了一项应用软件, 双方没有订立任何书面合同。在此情形下\_\_(10)\_\_享有该软件的著作权。
- (10) A. 甲公司  
B. 甲、乙公司共同  
C. 乙公司  
D. 甲、乙公司均不
- 甲、乙软件公司于 2013 年 9 月 12 日就其财务软件产品分别申请“大堂”和“大唐”商标注册。两财务软件相似, 且经协商双方均不同意放弃使用其申请注册的商标标识。此情形下, \_\_(11)\_\_获准注册。
- (11) A. “大堂”  
B. “大堂”与“大唐”都能  
C. “大唐”

- D. 由甲、乙抽签结果确定谁能
- 以下媒体中\_\_(12)\_\_是表示媒体, \_\_(13)\_\_是表现媒体。
    - (12) A. 声音  
B. 声音编码  
C. 超声波  
D. 喇叭
    - (13) A. 声音  
B. 声音编码  
C. 超声波  
D. 喇叭
  - 显示深度、图像深度是图像显示的重要指标。当\_\_(14)\_\_时, 显示器不能完全反映数字图像电使用的全部颜色。
    - (14) A. 显示深度=图像深度  
B. 显示深度>图像深度  
C. 显示深度≥图像深度  
D. 显示深度<图像深度
  - 以下关于结构化开发方法的叙述中, 不正确的是\_\_(15)\_\_。
    - (15) A. 总的指导思想是自顶向下, 逐层分解  
B. 基本原则是功能的分解与抽象  
C. 与面向对象开发方法相比, 更适合于大规模、特别复杂的项目  
D. 特别适合于数据处理领域的项目
  - 模块 A、B 和 C 都包含相同的 5 个语句, 这些语句之间没有联系。为了避免重复把这 5 个语句抽取出来组成一个模块 D, 则模块 D 的内聚类型为\_\_(16)\_\_内聚。
    - (16) A. 功能  
B. 通信  
C. 逻辑  
D. 巧合
  - 某个项目在开发时采用了不成熟的前沿技术, 由此而带来的风险属于\_\_(17)\_\_风险。
    - (17) A. 市场  
B. 技术  
C. 经济  
D. 商业
  - 属于面向对象、解释型程序设计语言的是\_\_(18)\_\_。
    - (18) A. XML  
B. Python  
C. Prolog

D. C++

- 下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示活动，边的权重表示活动的持续时间，则里程碑\_\_(19)\_\_在关键路径上。活动 GH 的松弛时间是\_\_(20)\_\_。



- (19) A. B  
 B. E  
 C. C  
 D. K
- (20) A. 0  
 B. 1  
 C. 2  
 D. 3

- 算术表达式“(a-b)\*(c+d)”的后缀式是\_\_(21)\_\_。

- (21) A. ab-cd+\*  
 B. abcd-\*+  
 C. ab-\*cd+  
 D. ab-c+d\*

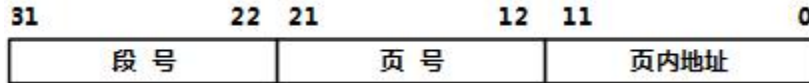
- 将高级语言源程序翻译成机器语言程序的过程中，常引入中间代码。以下关于中间代码的叙述中，不正确的是\_\_(22)\_\_。

- (22) A. 中间代码不依赖于具体的机器。  
 B. 使用中间代码可提高编译程序的可移植性  
 C. 中间代码可以用树或图表示  
 D. 中间代码可以用栈和队列表示

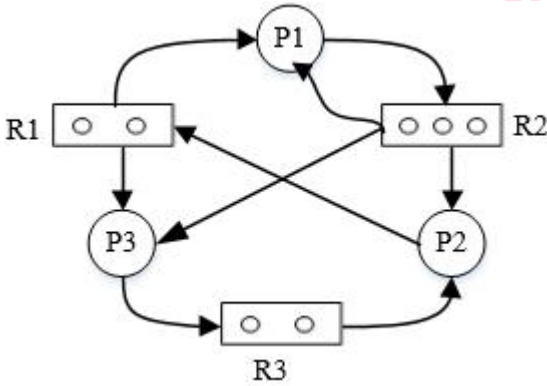
- 假设系统采用 PV 操作实现进程同步与互斥。若 n 个进程共享两台打印机，那么信号量 S 的取值范围为\_\_(23)\_\_。

- (23) A. -2~n  
 B. -(n-1)~1  
 C. -(n-1)~2  
 D. -(n-2)~2

- 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\_\_(24)\_\_\_。



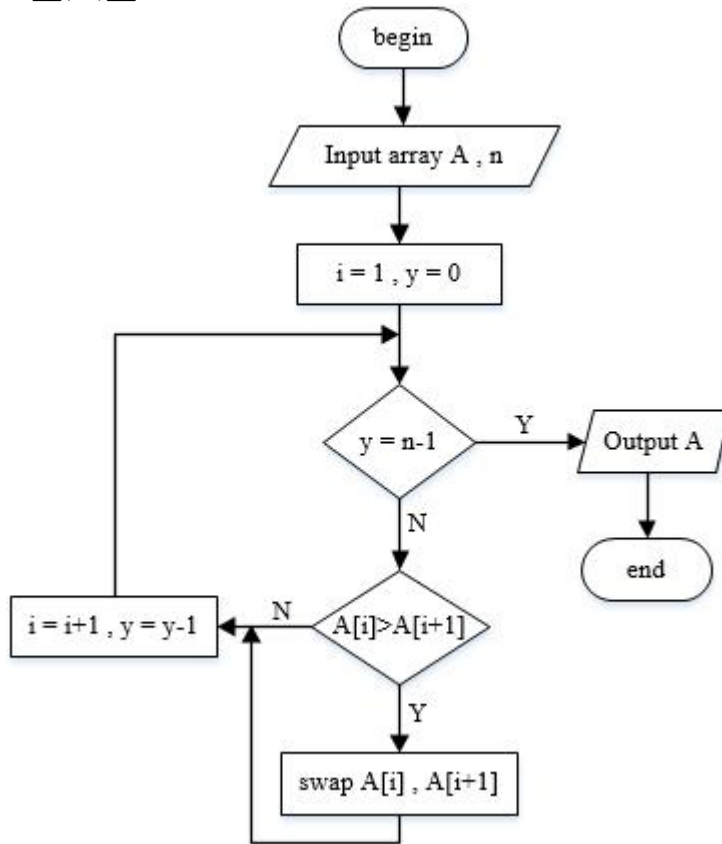
- (24) A. 最多可有 2048 个段, 每个段的大小均为 2048 个页, 页的大小为 2K  
 B. 最多可有 2048 个段, 每个段最大允许有 2048 个页, 页的大小为 2K  
 C. 最多可有 1024 个段, 每个段的大小均为 1024 个页, 页的大小为 4K  
 D. 最多可有 1024 个段, 每个段最大允许有 1024 个页, 页的大小为 4K
- 假设磁盘块与缓冲区大小相同, 每个盘块读入缓冲区的时间为 10 $\mu$ s, 由缓冲区送至用户区的时间是 5 $\mu$ s, 系统对每个磁盘块数据的处理时间为 2 $\mu$ s。若用户需要将大小为 10 个磁盘块的 Doc1 文件逐块从磁盘读入缓冲区, 并送至用户区进行处理, 那么采用单缓冲区需要花费的时间为\_\_(25)\_\_\_ $\mu$ s; 采用双缓冲区需要花费的时间为\_\_(26)\_\_\_ $\mu$ s。
- (25) A. 100  
 B. 107  
 C. 152  
 D. 170
- (26) A. 100  
 B. 107  
 C. 152  
 D. 170
- 在如下所示的进程资源图中, \_\_ (27) \_\_; 该进程资源图是\_\_(28)\_\_\_。



- (27) A. P1、P2、P3 都是阻塞节点  
 B. P1 是阻塞节点、P2、P3 是非阻塞节点  
 C. P1、P2 是阻塞节点、P3 是非阻塞节点  
 D. P1、P2 是非阻塞节点、P3 是阻塞节点
- (28) A. 可以化简的, 其化简顺序为 P1→P2→P3  
 B. 可以化简的, 其化简顺序为 P3→P1→P2  
 C. 可以化简的, 其化简顺序为 P2→P1→P3  
 D. 不可以化简的, 因为 P1、P2、P3 申请的资源都不能得到满足
- 以下关于增量模型的叙述中, 正确的是\_\_(29)\_\_\_。
- (29) A. 需求被清晰定义  
 B. 可以快速构造核心产品

- C. 每个增量必须要进行风险评估
  - D. 不适宜商业产品的开发
- 以下关于 CMM 的叙述中, 不正确的是\_\_(30)\_\_\_。
- (30) A. CMM 是指软件过程能力成熟度模型  
B. CMM 根据软件过程的不同成熟度划分了 5 个等级, 其中, 1 级被认为成熟度最高, 5 级被认为成熟度最低  
C. CMMI 的任务是将已有的几个 CMM 模型结合在一起, 使之构造成为“集成模型”  
D. 采用更成熟的 CMM 模型, 一般来说可以提高最终产品的质量
- 在 ISO/IEC 软件质量模型中, 可靠性是指在规定的二段时间内和规定的条件下, 软件维持在其性能水平的能力; 其子特性不包括\_\_(31)\_\_\_。
- (31) A. 成熟性  
B. 容错性  
C. 易恢复  
D. 可移植性
- 在软件开发过程中, 系统测试阶段的测试目标来自于\_\_(32)\_\_\_阶段。
- (32) A. 需求分析  
B. 概要设计  
C. 详细设计  
D. 软件实现
- 以下关于文档的叙述中, 不正确的是\_\_(33)\_\_\_。
- (33) A. 项目相关人员可以通过文档进行沟通  
B. 编写文档会降低软件开发的效率  
C. 编写高质量文档可以提高软件开发的质量  
D. 文档是软件的不可或缺的部分

- 下图所示的程序流程图中有\_\_(34)\_\_\_条不同的简单路径。采用 McCabe 度量法计算该程序图



的环路复杂性为\_\_(35)\_\_\_。

- (34) A. 3
- B. 4
- C. 5
- D. 6
- (35) A. 3
- B. 4
- C. 5
- D. 6
- 以下关于软件维护和可维护性的叙述中，不正确的是\_\_(36)\_\_\_。
- (36) A. 软件维护要解决软件产品交付用户之后运行中发生的各种问题
- B. 软件的维护期通常比开发期长得多，其投入也大得多
- C. 进行质量保证审查可以提高软件产品的可维护性
- D. 提高可维护性是在软件维护阶段考虑的问题
- 类\_\_(37)\_\_\_之间存在着一般和特殊的关系。
- (37) A. 汽车与轮船
- B. 交通工具与飞机
- C. 轮船与飞机
- D. 汽车与飞机

● 多态分为参数多态、包含多态、过载多态和强制多态四种不同形式, 其中\_\_(38)\_\_多态在许多语言中都存在, 最常见的例子就是子类型化。

- (38) A. 参数  
B. 包含  
C. 过载  
D. 强制

● 在面向对象程序设计语言中, 对象之间通过\_\_(39)\_\_方式进行通信。以下关于好的面向对象程序设计语言的叙述中, 不正确的是\_\_(40)\_\_。

- (39) A. 消息传递  
B. 继承  
C. 引用  
D. 多态
- (40) A. 应该支持被封装的对象  
B. 应该支持类写实例的概念  
C. 应该支持通过指针进行引用  
D. 应该支持继承和多态

● UML 中有 4 种事物: 结构事物、行为事物、分组事物和注释事物。类、接口、构建属于\_\_(41)\_\_事物; 依附于一个元素或一组元素之上对其进行约束或解释的简单符号为\_\_(42)\_\_事物。

- (41) A. 结构  
B. 行为  
C. 分组  
D. 注释
- (42) A. 结构  
B. 行为  
C. 分组  
D. 注释

● 一组对象以定义良好但是复杂的方式进行通信, 产生的相互依赖关系结构混乱且难以理解。采用\_\_(43)\_\_模式, 用一个中介对象来封装一系列的对象交互, 从而使各对象不需要显式地相互引用, 使其耦合松散。而且可以独立地改变它们之间的交互。此模式与\_\_(44)\_\_模式是相互竞争的模式, 主要差别是: 前者的中介对象封装了其它对象间的通信, 而后者通过引入其它对象来分布通信。

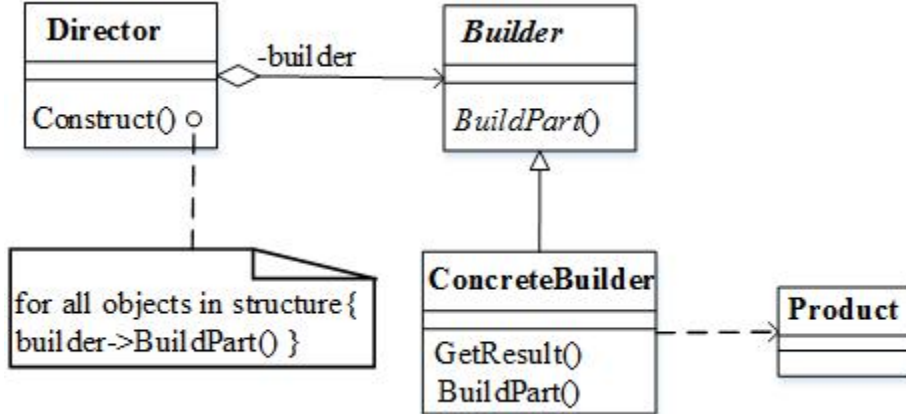
- (43) A. 解释器(Interpreter)  
B. 策略(Strategy)  
C. 中介者(Mediator)  
D. 观察者(Observer)
- (44) A. 解释器(Interpreter)  
B. 策略(Strategy)  
C. 中介者(Mediator)  
D. 观察者(Observer)



● UML 图中，一张交互图显示一个交互。由一组对象及其之间的关系组成，包含它们之间可能传递的消息。\_\_(45)\_\_不是交互图。

- (45) A. 序列图  
 B. 对象图  
 C. 通信图  
 D. 时序图

● 图所示为\_\_(46)\_\_设计模式，适用于\_\_(47)\_\_。

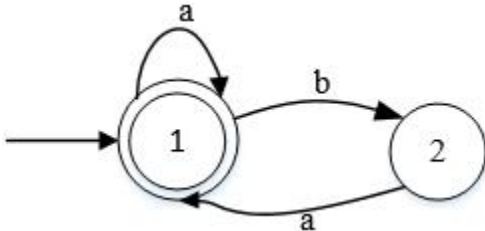


- (46) A. 抽象工厂 (Abstract Factory)  
 B. 生成器 (Builder)  
 C. 工厂方法 (Factory Method).  
 D. 原型 (Prototype)
- (47) A. 一个系统要由多个产品系列中的一个来配置时  
 B. 当一个类希望由它的子类来指定它所创建的对象时  
 C. 当创建复杂对象的算法应该独立于该对象的组成部分及其装配方式时  
 D. 当一个系统应该独立于它的产品创建、构成和表示时

● 对高级语言源程序进行编译的过程可以分为多个阶段，分配寄存器的工作在\_\_(48)\_\_阶段进行。

- (48) A. 词法分析  
 B. 语法分析  
 C. 语义分析  
 D. 目标代码生成

● 以下关于下图所示有限自动机的叙述中，不正确的是\_\_(49)\_\_。



- (49) A. 该自动机识别的字符串中 a 不能连续出现

- B. 自动机识别的字符串中 b 不能连续出现  
 C. 该自动机识别的非空字符串必须以 a 结尾  
 D. 该自动机识别的字符串可以为空串
- 对于大多数通用程序设计语言, 用\_\_(50)\_\_描述其语法即可。
    - (50) A. 正规文法
    - B. 上下文无关文法
    - C. 上下文有关文法
    - D. 短语结构文法
  - 在数据库逻辑结构设计阶段, 需要\_\_(51)\_\_阶段形成的\_\_(52)\_\_作为设计依据。
    - (51) A. 需求分析
    - B. 概念结构设计
    - C. 物理结构设计
    - D. 数据库运行和维护
    - (52) A. 程序文档、数据字典和数据流图。
    - B. 需求说明文档、程序文档和数据流图
    - C. 需求说明文档、数据字典和数据流图
    - D. 需求说明文档、数据字典和程序文档
  - 给定关系模式 R(A,B,C,D)、S(C,D,E), 与  $\pi_{1,3,5}(\sigma_{2='软件工程'}(R \bowtie S))$  等价的 SQL 语句如下:  
 SELECT \_\_(53)\_\_ FROM R,S WHERE \_\_(54)\_\_;  
 下列查询 B="信息"且 E="北京"的 A、B、E 的关系代数表达式中, 查询效率最高的是  
 \_\_(55)\_\_.
    - (53) A. A,C,S.C
    - B. A,B,E
    - C. A,R,C,E
    - D. A,R,C,S.D
    - (54) A. B='软件工程' OR R.C=S.C AND R.D=S.D
    - B. B='软件工程' OR R.C=S.C AND R.D=S.D
    - C. B='软件工程' OR R.C=S.C OR R.D=S.D
    - D. B='软件工程' AND R.C=S.C AND R.D=S.D
    - (55) A.  $\pi_{1,2,7}(\sigma_{2='信息' \wedge 3=5 \wedge 4=6 \wedge 7='北京'}(R \times S))$
    - B.  $\pi_{1,2,7}(\sigma_{3=5 \wedge 4=6}(\sigma_{2='信息'}(R) \times \sigma_{3='北京'}(S)))$
    - C.  $\pi_{1,2,7}(\sigma_{3=5 \wedge 4=6 \wedge 2='信息'}(R \times \sigma_{7='北京'}(S)))$
    - D.  $\pi_{1,2,7}(\sigma_{3=5 \wedge 4=6 \wedge 7='北京'}(\sigma_{2='信息'}(R) \times S))$
  - 给定关系模式 R(U,F), U={A,B,C,D,E,H}, 函数依赖集 F={A→B,A→C,C→D,AE→H}。关系模式 R 的候选关键字为\_\_(56)\_\_.
    - (56) A. AC
    - B. AB
    - C. AE
    - D. DE

- 对于线性表，相对于顺序存储，采用链表存储的缺点是\_\_(57)\_\_\_。

(57) A. 数据元素之间的关系需要占用存储空间，导致存储密度不高  
 B. 表中结点必须占用地址连续的存储单元，存储密度不高  
 C. 插入新元素时需要遍历整个链表，运算的时间效率不高  
 D. 删除元素时需要遍历整个链表，运算的时间效率不高

- 若一个栈初始为空，其输入序列是 1, 2, 3, ..., n-1, n，其输出序列的第一个元素为 k( $1 \leq k \leq \lceil n/2 \rceil$ )，则输出序列的最后一个元素是\_\_(58)\_\_\_。

(58) A. 值为 n 的元素  
 B. 值为 1 的元素  
 C. 值为 n-k 的元素  
 D. 不确定的

- 某个二叉查找树（即二叉排序树）中进行查找时，效率最差的情形是该二叉查找树是\_\_(59)\_\_\_。

(59) A. 完全二叉树  
 B. 平衡二叉树  
 C. 单枝树  
 D. 满二叉树

- 在字符串的 KMP 模式匹配算法中，需先求解模式串的 next 函数值，其定义如下式所示，j 表示模式串中字符的序号（从 1 开始）。若模式串 p 为“abaac”，则其 next 函数值为\_\_(60)\_\_\_。

$$next[j] = \begin{cases} 0 & j=1 \\ \max \{k \mid 1 < k < j, 'p_1 p_2 \dots p_{k-1}' = 'p_{j-k+1} p_{j-k+2} \dots p_{j-1}'\} & \\ 1 & \text{其他情况} \end{cases}$$

(60) A. 01234  
 B. 01122  
 C. 01211  
 D. 01111

- 快速排序算法在排序过程中，在待排序数组中确定一个元素为基准元素，根据基准元素把待排序数组划分成两个部分，前面一部分元素值小于等于基准元素，而后面一部分元素值大于基准元素。然后再分别对前后两个部分进一步进行划分。根据上述描述，快速排序算法采用了\_\_(61)\_\_\_算法设计策略。日知确定基准元素操作的时间复杂度为  $\Theta(n)$ ，则快速排序算法的最好和最坏情况下的时间复杂度为\_\_(62)\_\_\_。

(61) A. 分治  
 B. 动态规划  
 C. 贪心  
 D. 回溯

(62) A.  $\Theta(n)$ 和 $\Theta(n \lg n)$   
 B.  $\Theta(n)$ 和 $\Theta(n^2)$

- C.  $\Theta(\text{nlgn})$ 和 $\Theta(\text{nlgn})$
- D.  $\Theta(\text{nlgn})$ 和 $\Theta(n^2)$

● 对一待排序序列分别进行直接插入排序和简单选择排序，若待排序序列中有两个元素的值相同，则\_\_ (63) \_\_保证这两个元素在排序前后的相对位置不变。

- (63) A. 直接插入排序和简单选择排序都可以  
 B. 直接插入排序和简单选择排序都不能  
 C. 只有直接插入排序可以  
 D. 只有简单选择排序可以

● 已知一个文件中出现的各字符及其对应的频率如下表所示。若采用定长编码，则该文件中字符的码长应为\_\_ (64) \_\_。若采用 Huffman 编码，则字符序列“face”的编码应为\_\_ (65) \_\_。

| 字符     | a  | b  | c  | d  | e | f |
|--------|----|----|----|----|---|---|
| 频率 (%) | 45 | 13 | 12 | 16 | 9 | 5 |

- (64) A. 2  
 B. 3  
 C. 4  
 D. 5
- (65) A. 110001001101  
 B. 001110110011  
 C. 101000010100  
 D. 010111101011

● PPP 中的安全认证协议是\_\_ (66) \_\_，它使用三次握手的会话过程传送密文。

- (66) A. MD5  
 B. PAP  
 C. CHAP  
 D. HASH

● ICMP 协议属于因特网中的\_\_ (67) \_\_协议，ICMP 协议数据单元封装在\_\_ (68) \_\_中传送。

- (67) A. 数据链路层  
 B. 网络层  
 C. 传输层  
 D. 会话层
- (68) A. 以太帧  
 B. TCP 段  
 C. UDP 数据报  
 D. IP 数据报

● DHCP 客户端可从 DHCP 服务器获得\_\_ (69) \_\_。

- (69) A. DHCP 服务器的地址和 Web 服务器的地址  
B. DNS 服务器的地址和 DHCP 服务器的地址  
C. 客户端地址和邮件服务器地址  
D. 默认网关的地址和邮件服务器地址

● 分配给某公司网络的地址块是 210.115.192.0/20, 该网络可以被划分为\_\_(70)\_\_个 C 类子网。

- (70) A. 4  
B. 8  
C. 16  
D. 32

● Teams are required for most engineering projects. Although some small hardware or software products can be developed by individuals, the scale and complexity of modern systems is such, and the demand for short schedules so great, that it is no longer \_\_(71)\_\_ for one person to do most engineering jobs. Systems development is a team \_\_(72)\_\_, and the effectiveness of the team largely determines the \_\_(73)\_\_ of the engineering. Development teams often behave much like baseball or basketball teams. Even though they may have multiple specialties, all the members work toward \_\_(74)\_\_. However, on systems maintenance and enhancement teams, the engineers often work relatively independently, much like wrestling and track teams. A team is \_\_(75)\_\_ just a group of people who happen to work together. Teamwork takes practice and it involves special skills. Teams require common processes; they need agreed-upon goals; and they need effective guidance and leadership. The methods for guiding and leading such teams are well known, but they are not obvious.

- (71) A. convenient  
B. existing  
C. practical  
D. real
- (72) A. activity  
B. job  
C. process  
D. application
- (73) A. size  
B. quality  
C. scale  
D. complexity
- (74) A. multiple objectives  
B. different objectives  
C. a single objective  
D. independent objectives
- (75) A. relatively  
B. /  
C. only  
D. more than