

【软考达人】

# 软考资料免费获取

- 1、最新软考题库
- 2、软考备考资料
- 3、考前压轴题



**微信扫一扫，立马获取**



**6W+ 免费题库**



**免费备考资料**

PC版题库: [ruankaodaren.com](http://ruankaodaren.com)

# 全国计算机技术与软件专业技术资格（水平）考试

## 2022 年下半年 软件设计师 上午试卷

（考试时间 9:00~11:30 共 150 分钟）

请按下述要求正确填写答题卡

1. 在答题卡的指定位置上正确写入你的姓名和准考证号，并用正规 2B 铅笔在你写入的准考证号下填涂准考证号。
2. 本试卷的试题中共有 75 个空格，需要全部解答，每个空格 1 分，满分 75 分。
3. 每个空格对应一个序号，有 A、B、C、D 四个选项，请选择一个最恰当的选项作为解答，在答题卡相应序号下填涂该选项。
4. 解答前务必阅读例题和答题卡上的例题填涂样式及填涂注意事项。解答时用正规 2B 铅笔正确填涂选项，如需修改，请用橡皮擦干净，否则会导致不能正确评分。

### 例题

2022 年下半年全国计算机技术与软件专业技术资格（水平）考试日期是 (88) 月 (89) 日。

- |           |       |       |       |
|-----------|-------|-------|-------|
| (88) A. 9 | B. 10 | C. 11 | D. 12 |
| (89) A. 4 | B. 5  | C. 6  | D. 7  |

因为考试日期是“11 月 5 日”，故 (88) 选 C，(89) 选 B，应在答题卡序号 88 下对 C 填涂，在序号 89 下对 B 填涂（参看答题卡）。

以下关于 RISC（精简指令集计算机）特点的叙述中，错误的是\_\_（1）\_\_。

- (1) A. 对存储器操作进行限制，使控制简单化  
B. 指令种类多，指令功能强  
C. 设置大量通用寄存器  
D. 选取使用频率较高的一些指令，提高执行速度

CPU（中央处理单元）的基本组成部件不包括\_\_（2）\_\_。

- (2) A. 算逻运算单元 B. 系统总线 C. 控制单元 D. 寄存器组

某种部件用在 2000 台计算机系统中，运行工作 1000 小时后，其中有 4 台计算机的这种部件失效，则该部件的千小时可靠度 R 为\_\_（3）\_\_。

- (3) A. 0.990 B. 0.992 C. 0.996 D. 0.998

以下存储器中，\_\_（4）\_\_使用电容存储信息且需要周期性地进行刷新。

- (4) A. DRAM B. EPROM C. SRAM D. EEPROM

对于长度相同但格式不同的两种浮点数，假设前者阶码长、尾数短，后者阶码短、尾数长，其它规定都相同，则二者可以表示数值的范围和精度情况为\_\_（5）\_\_。

- (5) A. 二者可表示的数的范围和精度相同  
B. 前者所表示的数的范围更大且精度更高  
C. 前者所表示的数的范围更大但精度更低  
D. 前者所表示的数的范围更小但精度更高

计算机系统中采用补码表示有符号的数值，\_\_（6）\_\_。

- (6) A. 可以保持加法和减法运算过程与手工运算方式一致  
B. 可以提高运算过程和结果的精准程度  
C. 可以提高加法和减法运算的速度  
D. 可以将减法运算转换为加法运算从而简化运算器的设计

下列认证方式安全性较低的是\_\_（7）\_\_。

- （7） A. 生物认证      B. 多因子认证      C. 口令认证      D. U 盾认证

X.509 数字证书标准推荐的密码算法是\_\_（8）\_\_，而国密 SM2 数字证书采用的公钥密码算法是\_\_（9）\_\_。

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

某单位网站首页被恶意篡改，应部署\_\_（10）\_\_设备阻止恶意攻击。

- （10） A. 数据库审计      B. 包过滤防火墙      C. Web 应用防火墙      D. 入侵检测

使用漏洞扫描系统对信息系统和服务器进行定期扫描可以\_\_（11）\_\_。

- （11） A. 发现高危风险和安全漏洞      B. 修复高危风险和安全漏洞  
C. 获取系统受攻击的日志信息      D. 关闭非必要的网络端口和服务

以下关于某委托开发软件的著作权归属的叙述中，正确的是\_\_（12）\_\_。

- （12） A. 该软件的著作权归属仅依据委托人与受托人在书面合同中的约定来确定  
B. 无论是否有合同约定，该软件的著作权都由委托人和受托人共同享有  
C. 若无书面合同或合同中未明确约定，则该软件的著作权由受托人享有  
D. 若无书面合同或合同中未明确约定，则该软件的著作权由委托人享有

《计算机软件保护条例》第八条第一款第八项规定的软件著作权中的翻译权是将原软件由\_\_（13）\_\_的权利。

- （13） A. 源程序语言转换成目标程序语言  
B. 一种程序设计语言转换成另一种程序设计语言  
C. 一种汇编语言转换成一种自然语言  
D. 一种自然语言文字转换成另一种自然语言文字

M 公司将其开发的某软件产品注册商标为 S，为确保公司在市场竞争中占据优势地位，M 公司对员工进行了保密约束，此情形下，该公司不享有\_\_(14)\_\_。

- (14) A. 软件著作权 B. 专利权 C. 商业秘密权 D. 商标权

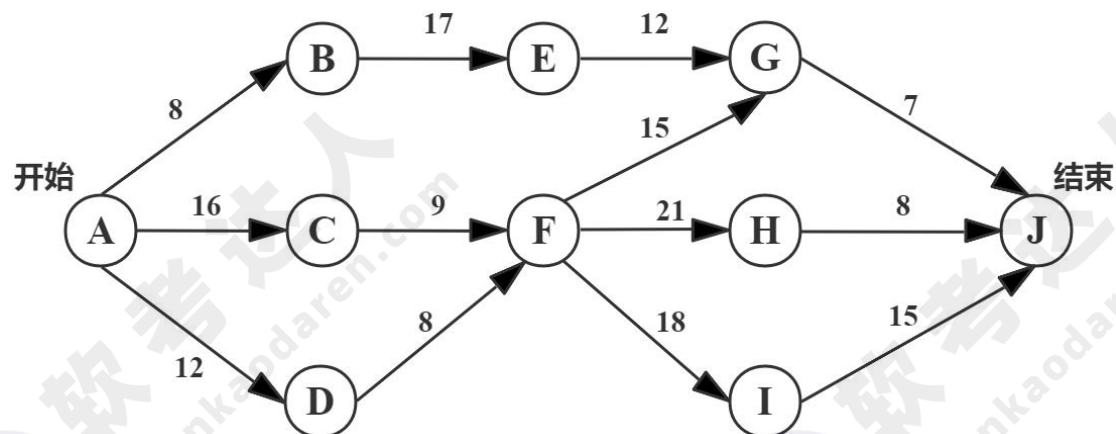
某零件厂商的信息系统中，一个基本加工根据客户类型、订单金额、客户信用等信息的不同采取不同的行为，此时最适宜采用\_\_(15)\_\_来描述该加工规格说明。

- (15) A. 自然语言 B. 流程图  
C. 判定表 D. 某程序设计语言

优化模块结构时，\_\_(16)\_\_不是适当的处理方法。

- (16) A. 使模块功能完整  
B. 消除重复功能，改善软件结构  
C. 只根据模块功能确定规模大小  
D. 避免或减少模块之间的病态连接

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，边上的数字表示完成该活动所需要的天数。则关键路径长度为\_\_(17)\_\_。若在实际项目进展中，在其他活动都能正常进行的前提下，活动\_\_(18)\_\_一旦延期就会影响项目的进度。



- (17) A. 34 B. 47 C. 54 D. 58  
(18) A. A→B B. C→F C. D→F D. F→H



以下关于风险管理的叙述中，不正确的是\_\_（19）\_\_。

- （19） A. 承认风险是客观存在的，不可能完全避免  
B. 同时管理所有的风险  
C. 风险管理应该贯穿整个项目管理过程  
D. 风险计划本身可能会带来新的风险

当函数调用执行时，在栈顶创建且用来支持被调用函数执行的一段存储空间称为活动记录或者栈帧，栈帧中不包括\_\_（20）\_\_。

- （20） A. 形参变量 B. 全局变量 C. 返回地址 D. 局部变量

编译器与解释器是程序语言翻译的两种基本形态，以下关于编译器工作方式及特点的叙述中，正确的是\_\_（21）\_\_。

- （21） A. 边翻译边执行，用户程序运行效率低且可移植性差  
B. 先翻译后执行，用户程序运行效率高且可移植性好  
C. 边翻译边执行，用户程序运行效率低且可移植性好  
D. 先翻译后执行，用户程序运行效率高且可移植性差

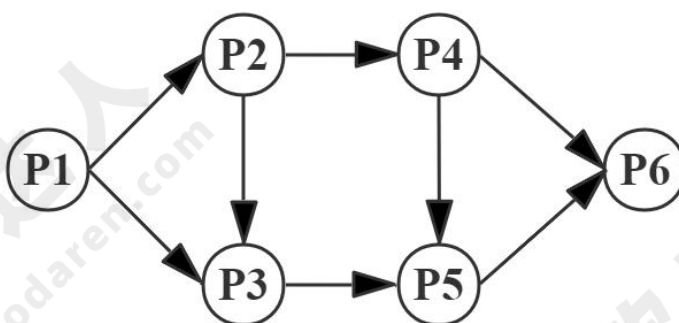
对高级语言源程序进行编译或解释的过程中需要进行语法分析，递归子程序分析属于\_\_（22）\_\_的分析法。

- （22） A. 自上而下 B. 自下而上 C. 从左至右 D. 从右至左

在计算机系统中，若 P1 进程正在运行，操作系统强行撤下 P1 进程所占用的 CPU，让具有更高优先级的进程 P2 运行，这种调度方式称为\_\_（23）\_\_。

- （23） A. 中断方式 B. 先进先出方式  
C. 可剥夺方式 D. 不可剥夺方式

进程 P1、P2、P3、P4、P5 和 P6 的前趋图如下所示。



假设用 PV 操作控制这 6 个进程的同步与互斥的程序如下，程序中的空①和空②处应分别为 (24)，空③和空④应分别为 (25)，空⑤和空⑥应分别为 (26)。

```
begin
  S1, S2, S3, S4, S5, S6, S7, S8: semaphore; // 定义信号量
  S1:=0; S2:=0; S3:=0; S4:=0; S5:=0; S6:=0; S7:=0; S8:=0;
  Cobegin
    process P1
      Begin
        P1 执行;
        ①;
      End;
    process P2
      Begin
        P(S1);
        P2 执行;
        ②;
      End;
    process P3
      Begin
        P(S2);
        ③;
        P3 执行;
        V(S5);
      End;
    process P4
      Begin
        P(S4);
        P4 执行;
        ④;
      End;
    process P5
      Begin
        P(S5);
        P(S6);
        P5 执行;
        ⑤;
      End;
    process P6
      Begin
        ⑥;
        P6 执行;
      End;
  Coend;
End.
```

- (24) A. V (S1) V (S2) 和 P (S2) P (S3)      B. V (S1) P (S2) 和 V (S3) P (S4)  
 C. V (S1) V (S2) 和 V (S3) V (S4)      D. P (S1) P (S2) 和 V (S2) V (S3)
- (25) A. V (S3) 和 V (S6) V (S7)      B. V (S3) 和 V (S6) P (S7)  
 C. P (S3) 和 V (S6) V (S7)      D. P (S3) 和 P (S6) V (S7)
- (26) A. V (S6) 和 P (S7) P (S8)      B. P (S8) 和 P (S7) P (S8)  
 C. P (S8) 和 P (S7) V (S8)      D. V (S8) 和 P (S7) P (S8)

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 (27)。

31	22	21	12	11	0
段 号	页 号			页内地址	

- (27) A. 最多可有 512 个段，每个段的大小均为 2048 个页，页的大小为 8K  
 B. 最多可有 512 个段，每个段最大允许有 2048 个页，页的大小为 8K  
 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

假设磁盘磁头从一个磁道移至相邻磁道需要 2ms。文件在磁盘上非连续存放，逻辑上相邻数据块的平均移动距离为 5 个磁道，每块的旋转延迟时间及传输时间分别为 10ms 和 1ms，则读取一个 100 块的文件需要 (28) ms。

- (28) A. 1100 B. 1200 C. 2100 D. 2200

以下关于快速原型模型优点的叙述中，不正确的是 (29)。

- (29) A. 有助于满足用户的真实需求  
B. 适用于大型软件系统的开发  
C. 开发人员快速开发出原型系统，因此可以加速软件开发过程，节约开发成本  
D. 原型系统已经通过与用户的交互得到验证，因此对应的规格说明文档能正确描述用户需求

以下关于三层 C/S 结构的叙述中，不正确的是 (30)。

- (30) A. 允许合理划分三层结构的功能，使之在逻辑上保持相对独立性，提高系统的可维护性和可扩展性  
B. 允许更灵活有效地选用相应的软硬件平台和系统  
C. 应用的各层可以并发开发，但需要相同的开发语言  
D. 利用功能层有效地隔离表示层和数据层，便于严格的安全管理

若模块 A 和模块 B 通过外部变量来交换输入、输出信息，则这两个模块的耦合类型是 (31) 耦合。

- (31) A. 数据 B. 标记 C. 控制 D. 公共

软件开发的目的是开发出高质量的软件系统，这里的高质量不包括 (32)。

- (32) A. 软件必须满足用户规定的需求  
B. 软件应遵循规定标准所定义的一系列开发准则  
C. 软件开发应采用最新的开发技术  
D. 软件应满足某些隐含的需求，如可理解性、可维护性等



白盒测试技术的各种覆盖方法中，（33）具有最弱的错误发现能力。

- (33) A. 判定覆盖 B. 语句覆盖 C. 条件覆盖 D. 路径覆盖

文档是软件的重要因素，关于高质量文档，以下说法不正确的是（34）。

- (34) A. 不论项目规模和复杂程度如何，都要用统一的标准指定相同类型和相同要素的文档  
B. 应该分清读者对象  
C. 应当是完整的、独立的、自成体系的  
D. 行文应十分确切，不出现多义性描述

某财务系统的一个组件中，某个变量没有正确初始化，（35）最可能发现该错误。

- (35) A. 单元测试 B. 集成测试 C. 接受测试 D. 安装测试

软件交付给用户之后进入维护阶段，根据维护具体内容的不同将维护分为不同的类型，其中“采用专用的程序模块对文件或数据中的记录进行增加、修改和删除等操作”的维护属于（36）。

- (36) A. 程序维护 B. 数据维护 C. 代码维护 D. 设备维护

采用面向对象方法进行某游戏设计，游戏中有野鸭、红头鸭等各种鸭子边游泳戏水边呱呱叫，不同种类的鸭子具有不同颜色，设计鸭子类负责呱呱叫和游泳方法的实现，显示颜色设计为抽象方法，由野鸭和红头鸭各自具体实现，这一机制称为（37）。当给这些类型的一组不同对象发送同一显示颜色消息时，能实现各自显示自己不同颜色的结果，这种现象称为（38）。

- (37) A. 继承 B. 聚合 C. 组合 D. 多态  
(38) A. 覆盖 B. 重载 C. 动态绑定 D. 多态

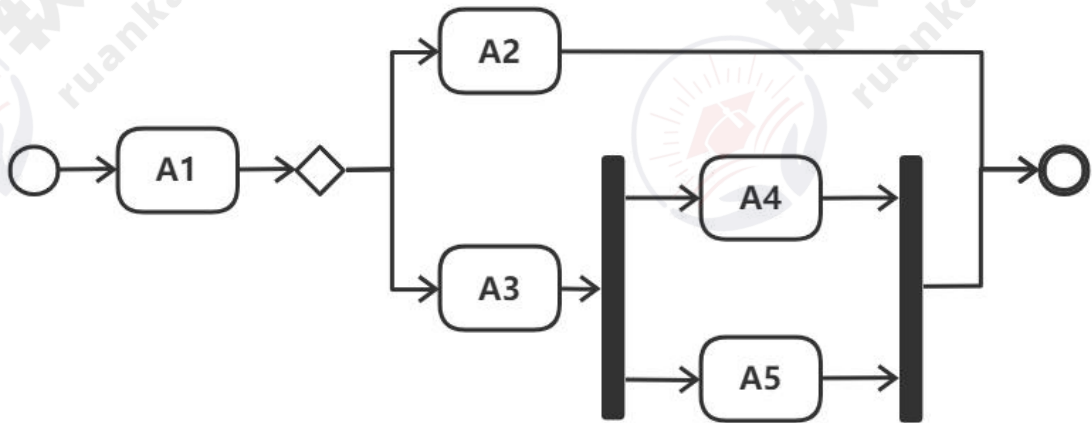
采用面向对象方法分析时，首先要在应用领域中按自然存在的实体认定对象，即将自然存在的（39）作为一个对象。

- (39) A. 问题 B. 关系 C. 名词 D. 动词

进行面向对象系统设计时，修改某个类的原因有且只有一个，即一个类只做一种类型的功能，这属于（40）原则

- (40) A. 单一责任 B. 开放-封闭 C. 接口分离 D. 依赖倒置

UML 活动图用于建模（41）。以下活动图中，活动 A1 之后，可能的活动执行序列顺序是（42）。



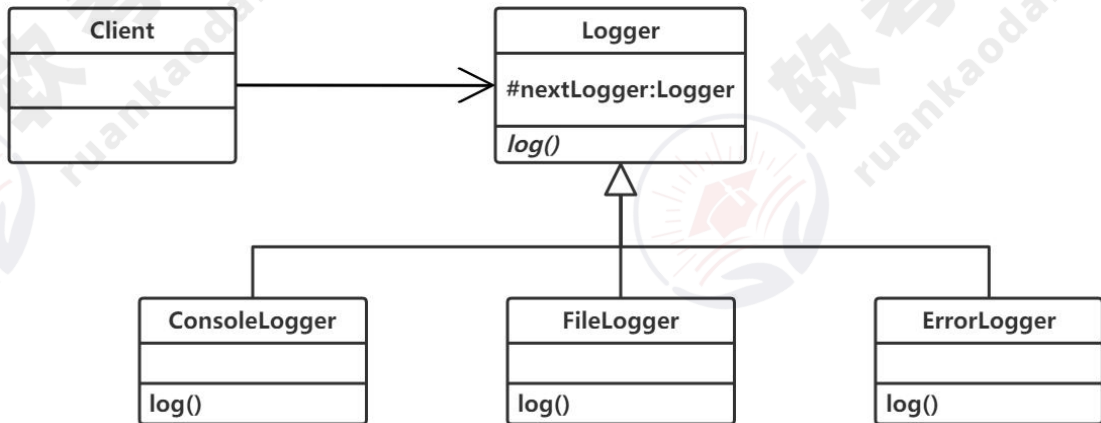
- (41) A. 系统在它的周边环境的语境中所提供的外部可见服务  
 B. 某一时刻一组对象以及它们之间的关系  
 C. 系统内从一个活动到另一个活动的流程  
 D. 对象的生命周期中某个条件或者状态

- (42) A. A2、A3、A4 和 A5 B. A3、A4 和 A5，或 A2、A4 或 A5  
 C. A2、A4 和 A5 D. A2 或 A3、A4 和 A5

UML 构件图（component diagram）展现了一组构件之间的组织和依赖，专注于系统的静态（43）图，图中通常包括构件、接口以及各种关系。

- (43) A. 关联 B. 实现 C. 结构 D. 行为

在某系统中，不同级别的日志信息记录方式不同，每个级别的日志处理对象根据信息级别高低，采用不同方式进行记录。每个日志处理对象检查消息的级别，如果达到它的级别则进行记录，否则不记录；然后将消息传递给它的下一个日志处理对象。针对此需求，设计如下所示类图。该设计模式采用\_\_(44)\_\_模式使多个前后连接的对象都有机会处理请求，从而避免请求的发送者和接收者之间的耦合关系。该模式属于\_\_(45)\_\_模式，该模式适用于\_\_(46)\_\_。



- (44) A. 责任链 (Chain of Responsibility) B. 策略 (Strategy)  
C. 过滤器 (Filter) D. 备忘录 (Memento)
- (45) A. 行为型类 B. 行为型对象 C. 结构型类 D. 结构型对象
- (46) A. 不同的标准过滤一组对象，并通过逻辑操作以解耦的方式将它们链接起来  
B. 可处理一个请求的对象集合应被动态指定  
C. 必须保存一个对象在某一个时刻的状态，需要时它才能恢复到先前的状态  
D. 一个类定义了多种行为，并且以多个条件语句的形式出现

驱动新能源汽车的发动机时，电能和光能汽车分别采用不同驱动方法，而客户端希望使用统一的驱动方法，需定义一个统一的驱动接口屏蔽不同的驱动方法，该要求适合采用\_\_(47)\_\_模式。

- (47) A. 中介者 (Mediator) B. 访问者 (Visitor)  
C. 观察者 (Observer) D. 适配器 (Adapter)

在 Python3 中，\_\_(48)\_\_不是合法的异常处理结构。

- (48) A. try...except... B. try...except...finally  
C. try...catch... D. raise

在 Python3 中，表达式 `list(range(11))[10:0:-2]` 的值为 (49)。

- (49) A. [10, 8, 6, 4, 2, 0] B. [10, 8, 6, 4, 2]  
C. [0, 2, 4, 6, 8, 10] D. [0, 2, 4, 6, 8]

在 Python3 中，执行语句 `x = input()`，如果从键盘输入 123 并按回车键，则 x 的值为 (50)。

- (50) A. 123 B. 1,2,3 C. 1 2 3 D. '123'

E-R 模型向关系模型转换时，两个实体  $E_1$  和  $E_2$  之间的多对多联系 R 应该转换为一个独立的关系模式，且该关系模式的关键字由 (51) 组成。

- (51) A. 联系 R 的属性 B.  $E_1$  或  $E_2$  的关键字  
C.  $E_1$  和  $E_2$  的关键字 D.  $E_1$  和  $E_2$  的关键字加上 R 的属性

某高校人力资源管理系统数据库中，

教师关系模式为 T（教师号，姓名，部门号，岗位，联系地址，薪资），

函数依赖集  $F = \{\text{教师号} \rightarrow (\text{姓名}, \text{部门号}, \text{岗位}, \text{联系地址}), \text{岗位} \rightarrow \text{薪资}\}$ 。

关系模式 T 的主键为 (52)，函数依赖集 F (53)。

- (52) A. 教师号，T 存在冗余以及插入异常和删除异常的问题  
B. 教师号，T 不存在冗余以及插入异常和删除异常的问题  
C. (教师号，岗位)，T 存在冗余以及插入异常和删除异常的问题  
D. (教师号，岗位)，T 不存在冗余以及插入异常和删除异常的问题
- (53) A. 存在传递依赖，故关系模式 T 最高达到 1NF  
B. 存在传递依赖，故关系模式 T 最高达到 2NF  
C. 不存在传递依赖，故关系模式 T 最高达到 3NF  
D. 不存在传递依赖，故关系模式 T 最高达到 4NF

给定员工关系 E（员工号，员工名，部门名，电话，家庭住址）、工程关系 P（工程号，

工程名，前期工程号)、参与关系 EP (员工号，工程号，工作量)。查询“005”员工参与了“虎头山隧道”工程的员工名、部门名、工程名、工作量的关系代数表达式如下：

$$\pi_{2,3,5,6}(\pi_{1,2,3}(\text{(54)}) \bowtie (\text{(55)}))$$

(54) A.  $\sigma_{2='005'}(E)$

B.  $\sigma_{1='005'}(E)$

C.  $\sigma_{2='005'}(P)$

D.  $\sigma_{1='005'}(P)$

(55) A.  $\pi_{2,3}(\sigma_{2='虎头山隧道'}(P)) \bowtie EP$

B.  $\pi_{2,3}(\sigma_{2='虎头山隧道'}(EP)) \bowtie P$

C.  $\pi_{1,2}(\sigma_{2='虎头山隧道'}(EP)) \bowtie P$

D.  $\pi_{1,2}(\sigma_{2='虎头山隧道'}(P)) \bowtie EP$

假设事务程序 A 中的表达式  $x/y$ ，若  $y$  取值为 0，则计算该表达式时，会产生故障。该故障属于 (56)。

(56) A. 系统故障

B. 事务故障

C. 介质故障

D. 死机

设栈初始时空，对于入栈序列  $1, 2, 3, \dots, n$ ，这些元素经过栈之后得到出栈序列  $P_1, P_2, P_3, \dots, P_n$ ，若  $P_3=4$ ，则  $P_1, P_2$  不可能的取值为 (57)。

(57) A. 6, 5

B. 2, 3

C. 3, 1

D. 3, 5

设  $m$  和  $n$  是某二叉树上的两个结点，中序遍历时， $n$  排在  $m$  之前的条件是 (58)。

(58) A.  $m$  是  $n$  的祖先结点

B.  $m$  是  $n$  的子孙结点

C.  $m$  在  $n$  的左边

D.  $m$  在  $n$  的右边

若无向图  $G$  有  $n$  个顶点  $e$  条边，则  $G$  采用邻接矩阵存储时，矩阵的大小为 (59)。

(59) A.  $n * e$

B.  $n^2$

C.  $n^2 + e^2$

D.  $(n+e)^2$

以下关于  $m$  阶 B-树的说法中，错误的是 (60)。

- (60) A. 根结点最多有  $m$  棵子树  
 B. 所有叶子结点都在同一层次上  
 C. 结点中的关键字有序排列  
 D. 叶子结点通过指针链接为有序表

下列排序算法中，占用辅助存储空间最多的是 (61)。

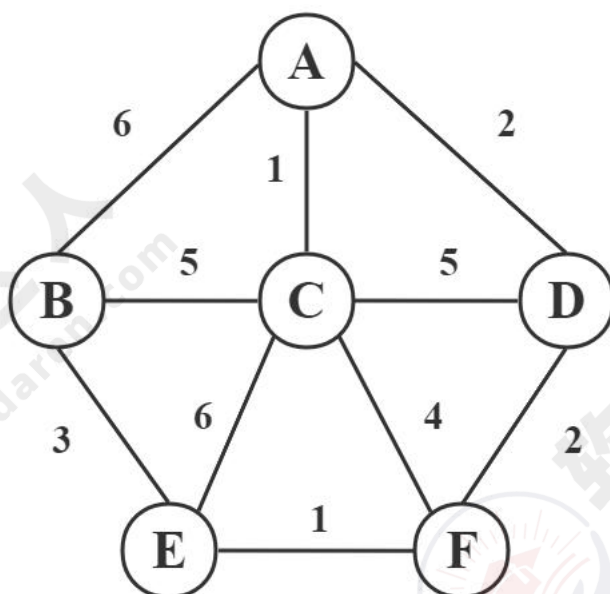
- (61) A. 归并排序 B. 快速排序 C. 堆排序 D. 冒泡排序

折半查找在有序数组  $A$  中查找特定的记录  $K$ ：通过比较  $K$  和数组中的中间元素  $A[mid]$  进行比较，如果相等，则算法结束；如果  $K$  小于  $A[mid]$ ，则对数组的前半部分进行折半查找；否则对数组的后半部分进行折半查找。根据上述描述，折半查找采用了 (62) 算法设计策略。对有序数组  $(3, 14, 27, 39, 42, 55, 70, 85, 93, 98)$ ，成功查找和失败查找所需要的平均比较次数分别是 (63)（假设查找每个元素的概率是相同的）。

- (62) A. 分治 B. 动态规划 C. 贪心 D. 回溯

- (63) A. 29/10 和 29/11  
 B. 30/10 和 30/11  
 C. 29/10 和 39/11  
 D. 30/10 和 40/11

采用 Dijkstra 算法求解下图 A 点到 E 点的最短路径，采用的算法设计策略是 (64)。  
 该最短路径的长度是 (65)。



- (64) A. 分治法 B. 动态规划 C. 贪心算法 D. 回溯法

- (65) A. 5 B. 6 C. 7 D. 9

VLAN tag 在 OSI 参考模型的 (66) 实现。



(66) A. 网络层 B. 传输层 C. 数据链路层 D. 物理层

Telnet 协议是一种 (67) 远程登录协议。

(67) A. 安全 B. B/S 模式 C. 基于 TCP D. 分布式

以下关于 HTTPS 和 HTTP 协议的叙述中，错误的是 (68)。

(68) A. HTTPS 协议使用加密传输  
B. HTTPS 协议默认服务端口号是 443  
C. HTTP 协议默认服务端口号是 80  
D. 电子支付类网站应使用 HTTP 协议

将网址转换为 IP 地址要用 (69) 协议。

(69) A. 域名解析 B. IP 地址解析 C. 路由选择 D. 传输控制

以下关于 IP 地址和 MAC 地址说法错误的是 (70)。

(70) A. IP 地址长度 32 或 128 位，MAC 地址的长度 48 位  
B. IP 地址工作在网络层，MAC 地址工作在数据链路层  
C. IP 地址的分配是基于网络拓扑，MAC 地址的分配是基于制造商  
D. IP 地址具有唯一性，MAC 地址不具有唯一性

We initially described SOA without mentioning Web services, and vice versa. This is because

they are orthogonal: service-orientation is an architectural (71) , while Web services are an implementation (72) . The two can be used together, and they frequently are, but they are not mutually dependent.

For example, although it is widely considered to be a distributed-computing solution, SOA can be applied to advantage in a single system, where services might be individual processes with well-defined (73) that communicate using local channels, or in self-contained cluster, where they might communicate across a high-speed interconnect.

Similarly, while Web services are (74) as the basis for a service-oriented environment, there is nothing in their definition that requires them to embody the SOA principles. While (75) is often held up as a key characteristic of Web services, there is no technical reason that they should be stateless — that would be a design choice of the developer, which may be dictated by the architectural style of the environment in which the service is intended to participate.

- |      |                      |                       |                         |              |
|------|----------------------|-----------------------|-------------------------|--------------|
| (71) | A. design            | <b>B. style</b>       | C. technology           | D. structure |
| (72) | A. structure         | B. style              | <b>C. technology</b>    | D. method    |
| (73) | <b>A. interfaces</b> | B. functions          | C. logics               | D. formats   |
| (74) | A. regarded          | <b>B. well-suited</b> | C. worked               | D. used      |
| (75) | A. distribution      |                       | B. interconnection      |              |
|      | C. dependence        |                       | <b>D. statelessness</b> |              |

## SOA and Web Services: Style vs. Implementation

We initially described SOA without mentioning Web services, and vice versa. This is because they are orthogonal: service-orientation is an architectural **style**, while Web services are an implementation **technology**. The two can be used together, and they frequently are, but they are not mutually dependent.

For example, although it is widely considered to be a distributed-computing solution, SOA can be applied to advantage in a single system, where services might be individual processes with well-defined **interfaces** that communicate using local channels, or in a self-contained cluster, where they might communicate across a high-speed interconnect.

Similarly, while Web services are **well-suited** as the basis for a service-oriented environment, there is nothing in their definition that *requires* them to embody the SOA principles. While **statelessness** is often held up as a key characteristic of Web services, there is no technical reason that they should be stateless—that would be a design choice of the developer, which may be dictated by the architectural style of the environment in which the service is intended to participate.

我们最初描述 SOA 时没有提到 Web 服务，反之亦然。这是因为它们是正交的：面向服

务是一种架构风格，而 Web 服务是一种实现技术。两者可以一起使用，而且它们经常是，但它们并不相互依赖。

例如，尽管 SOA 被广泛认为是一种分布式计算解决方案，但它可以应用于单个系统中，其中服务可能是具有明确定义的接口的单个进程，使用本地通道进行通信，或者在自包含集群中，它们可能通过高速互连进行通信。

类似地，尽管 Web 服务非常适合作为面向服务的环境的基础，但它们的定义中没有任何东西要求它们体现 SOA 原则。虽然无状态通常被认为是 Web 服务的一个关键特征，但从技术上讲，它们没有理由应该是无状态的——这将是开发人员的设计选择，这可能由服务打算参与的环境的架构风格决定。