

# 全国计算机技术与软件专业技术资格（水平）考试

## 2021 年上半年 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

2021 年上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. 4  | B. 5  | C. 6  | D. 7  |
| (89) A. 27 | B. 28 | C. 29 | D. 30 |

因为考试日期是“5 月 29 日”，故 (88) 选 B，(89) 选 C，应在答题卡序号 88 下对 B 填涂，在序号 89 下对 C 填涂（参看答题卡）。

在 CPU 中，用\_\_（1）\_\_给出将要执行的下一条指令在内存中的地址。

- （1） A. 程序计数器 B. 指令寄存器  
C. 主存地址寄存器 D. 状态条件寄存器

以下关于 RISC 和 CISC 计算机的叙述中，正确的是\_\_（2）\_\_。

- （2） A. RISC 不采用流水线技术，CISC 采用流水线技术  
B. RISC 使用复杂的指令，CISC 使用简单的指令  
C. RISC 采用很少的通用寄存器，CISC 采用很多的通用寄存器  
D. RISC 采用组合逻辑控制器，CISC 普遍采用微程序控制器

采用 DMA 方式传送数据时，每传送一个数据都需要占用一个\_\_（3）\_\_。

- （3） A. 指令周期 B. 总线周期  
C. 存储周期 D. 机器周期

以下关于闪存（Flash Memory）的叙述中，错误的是\_\_（4）\_\_。

- （4） A. 掉电后信息不会丢失，属于非易失性存储器  
B. 以块为单位进行删除操作  
C. 采用随机访问方式，常用来代替主存  
D. 在嵌入式系统中可以用 Flash 来代替 ROM 存储器

若磁盘的转速提高一倍，则\_\_（5）\_\_。

- （5） A. 平均存取时间减半 B. 平均寻道时间加倍  
C. 旋转等待时间减半 D. 数据传输速率加倍

异常是指令执行过程中在处理器内部发生的特殊事件，中断是来自处理器外部的请求事件。以下关于中断和异常的叙述中，正确的是\_\_（6）\_\_。

- （6） A. “DMA 传送结束”、“除运算时除数为 0”都为中断  
B. “DMA 传送结束”为中断、“除运算时除数为 0”为异常  
C. “DMA 传送结束”为异常、“除运算时除数为 0”为中断  
D. “DMA 传送结束”、“除运算时除数为 0”都为异常

下列协议中，属于安全远程登录协议的是\_\_（7）\_\_。

- （7） A. TLS B. TCP C. SSH D. TFTP

下列攻击类型中，\_\_（8）\_\_是以被攻击对象不能继续提供服务为首要目标。

- （8） A. 跨站脚本 B. 拒绝服务 C. 信息篡改 D. 口令猜测

下列算法中属于非对称加密算法的是\_\_（9）\_\_。

- （9） A. DES B. RSA C. AES D. MD5

SQL 是一种数据库结构化查询语言，SQL 注入攻击的首要目标是\_\_（10）\_\_。

- （10） A. 破坏 Web 服务  
B. 窃取用户口令等机密信息  
C. 攻击用户浏览器，以获得访问权限  
D. 获得数据库的权限

通常使用\_\_（11）\_\_为 IP 数据报文进行加密。

- （11） A. IPSec B. PP2P C. HTTPS D. TLS

根据《计算机软件保护条例》的规定，对软件著作权的保护不包括\_\_（12）\_\_。

- （12） A. 目标程序 B. 软件文档  
C. 源程序 D. 开发软件所有的操作方法

甲、乙两互联网公司于 2020 年 7 月 7 日就各自开发库存管理软件分别申请“宏达”和“鸿达”商标注册，两个库存管理软件相似，甲第一次使用时间为 2019 年 7 月，乙第一次使用时间为 2019 年 5 月，此情景下，\_\_（13）\_\_能获准注册。

- （13） A. “宏达” B. “宏达”和“鸿达”均  
C. 由甲、乙协商哪个 D. “鸿达”

A 经销售擅自复制并销售 B 公司开发的 OA 软件光盘已构成侵权,C 企业在未知情形下从 A 处购入 100 张并已安装使用, 在 C 企业知道了所使用的软件为侵权复制的情形下, 以下说法正确的是\_\_ (14) \_\_。

- (14) A. C 企业的使用行为侵权, 须承担赔偿责任  
B. C 企业的使用行为侵权, 支付合理费用后可以继续使用这 100 张软件光盘  
C. C 企业的使用行为不侵权, 可以继续使用这 100 张软件光盘  
D. C 企业的使用行为不侵权, 不需承担任何法律责任

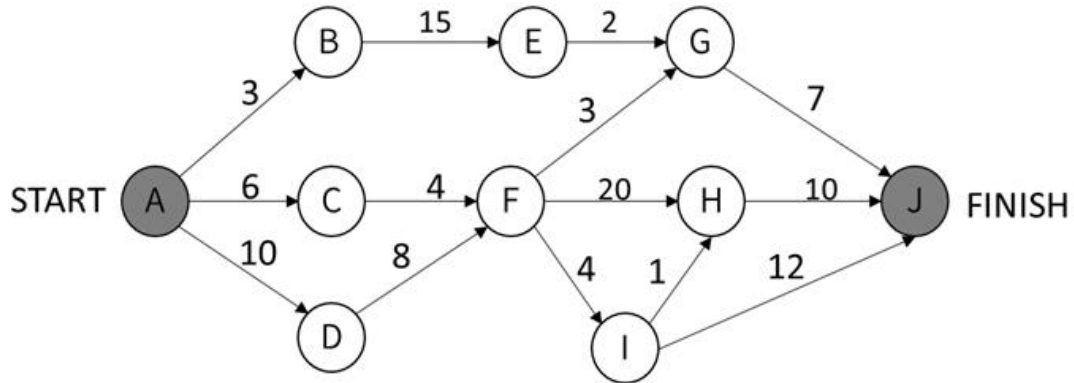
下列关于结构化分析方法的数据字典加工逻辑的叙述中, 不正确的是\_\_ (15) \_\_。

- (15) A. 对每一个基本加工, 应该有一个加工逻辑  
B. 加工逻辑描述输入数据流变换位输出数据的加工规则  
C. 加工逻辑必须实现加工的数据结构和算法  
D. 结构化语言, 判定树和判定表可以用来表示加工逻辑

在软件设计阶段进行模块划分时, 一个模块的\_\_ (16) \_\_。

- (16) A. 控制范围应该在其作用范围之内  
B. 作用范围应该在其控制范围之内  
C. 作用范围与控制范围互不包含  
D. 作用范围与控制范围不受任何限制

下面是一个软件项目活动图，其中顶点表示项目里程碑，连接顶点的边表示包含的活动，边上的权重表示活动的持续时间（天），则关键路径长度为（17），在该活动图中，活动（18）晚 16 天开始不会影响上班。



- (17) A. 20                      B. 25                      C. 27                      **D. 48**
- (18) A. AC                      **B. BE**                      C. FI                      D. HJ

下列关于风险的叙述中，不正确的是（19）。

- (19) A. 风险是可能发生的事件  
**B. 如果能预测到风险，则可以避免其发生**  
 C. 风险是可能会带来损失的事件  
 D. 对于风险进行干预，以期减少损失

算术表达式  $a * (b + c/d) - e$  的后缀式为（20）。

- (20) **A. abcd/+ \* e-**                      B. abcde \*+/-  
 C.  $a * b + c/d - e$                       D.  $ab * cd / + e -$

以编译方式翻译 C/C++ 源程序的过程中，类型检查在（21）阶段处理。

- (21) A、词法分析                      **B、语义分析**  
 C、语法分析                      D、目标代码生成

Java 语言符合的特征有 (22) 和自动的垃圾回收处理。

- ①采用即时编译
- ②采用静态优化编译
- ③对象在堆空间分配
- ④对象在栈空间分配

(22) A. ①③ B. ①④ C. ②③ D. ②④

云计算有多种部署模型 (Deployment Models)。若云的基础设施是为某个客户单独使用而构建的, 那么该部署模型属于 (23)。

(23) A. 公有云 B. 私有云 C. 社区云 D. 混合云

若计算机系统的字长为 128 位, 磁盘的容量为 2048GB, 物理块的大小为 8MB, 假设文件管理系统采用位示图 (bitmap) 法记录该计算机系统磁盘的使用情况, 那么位示图的大小需要 (24) 个字。

(24) A. 1024 B. 2048 C. 4096 D. 8192

进程 P 有 5 个页面, 页号为 0-4, 页面变换表及状态位、访问位和修改位的含义如下图所示, 若系统给进程 P 分配了 3 个存储块, 当访问的页面 3 不在内存时, 应该淘汰表中页号为 (25) 的页面。

页号	页帧号	状态位	访问位	修改位
0	8	1	1	0
1	-	0	0	0
2	3	1	1	1
3	-	0	0	0
4	13	1	1	1

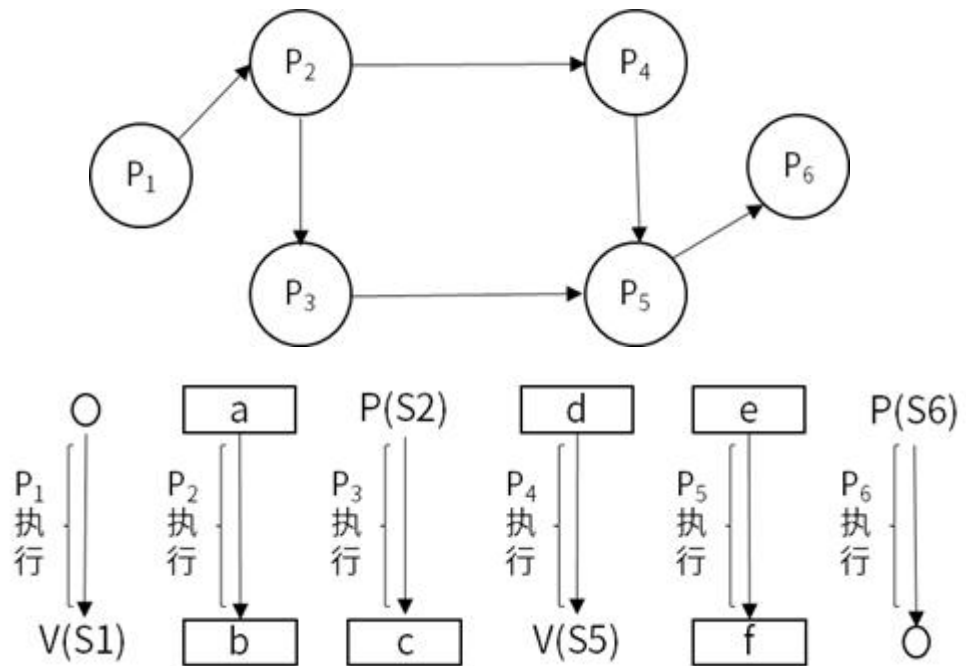
状态位含义 =  $\begin{cases} 0 & \text{不在内存} \\ 1 & \text{在内存} \end{cases}$

访问位含义 =  $\begin{cases} 0 & \text{未访问过} \\ 1 & \text{访问过} \end{cases}$

修改位含义 =  $\begin{cases} 0 & \text{未修改过} \\ 1 & \text{修改过} \end{cases}$

(25) A. 0 B. 1 C. 2 D. 4

假进程 P1、P2、P3、P4、P5 和 P6 的前驱图如下所示：



若用 PV 操作控制进程 P1、P2、P3、P4、P5 和 P6 开发执行的过程，需要设置 6 个信号量 S1、S2、S3、S4、S5 和 S6，且信号量 S1~S6 的初值都等于零。下面的进程执行图中 a 和 b 处分别填写 (26); c 和 d 处应分别填写 (27); e 和 f 处应分别填写 (28);

- |                                  |                                  |
|----------------------------------|----------------------------------|
| (26) A. V (S1) 和 P (S2) V (S3)   | B. P (S1) 和 P (S2) V (S3)        |
| C. V (S1) 和 V (S2) V (S3)        | <b>D. P (S1) 和 V (S2) V (S3)</b> |
| (27) A. P (S2) 和 P (S4)          | <b>B. V (S4) 和 P (S3)</b>        |
| C. P (S2) 和 V (S4)               | D. V (S2) 和 V (S4)               |
| (28) A. P (S4) V (S5) 和 V (S6)   | B. P (S4) V (S5) 和 P (S6)        |
| <b>C. P (S4) P (S5) 和 V (S6)</b> | D. P (S4) P (S5) 和 P (S6)        |

关于螺旋模型，下列陈述中不正确的是\_\_（29）\_\_，\_\_（30）\_\_。

- （29） A. 将风险分析加入到瀑布模型中  
B. 将开发过程划分为几个螺旋周期，每个螺旋周期大致和瀑布模型相符  
C. 适合于大规模、复杂且具有高风险的项目  
D. 可以快速的提供一个初始版本让用户测试
- （30） A. 支持用户需求的动态变化  
B. 要求开发人员具有风险分析能力  
C. 基于该模型进行软件开发，开发成本低  
D. 过多的迭代次数可能会增加开发成本，进而延迟提交时间

模块 A 通过非正常入口转入模块 B 内部，则这两个模块之间是\_\_（31）\_\_耦合。

- （31） A. 数据                      B. 公共                      C. 外部                      D. 内容

软件详细设计阶段的主要任务不包括 （32） 。

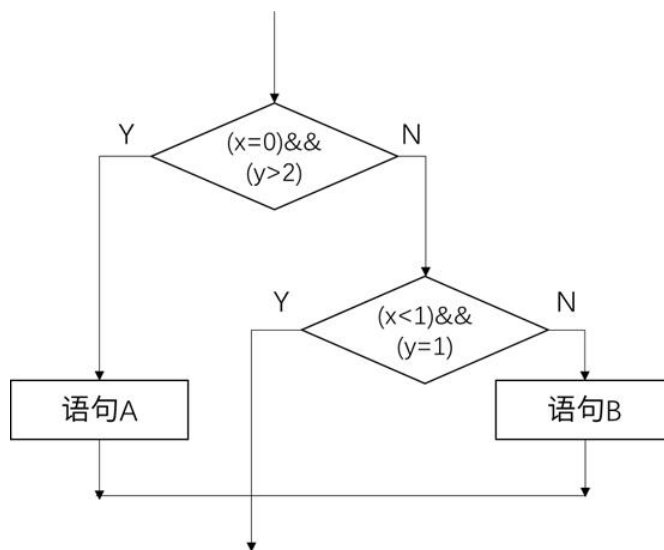
- （32） A. 数据结构设计                      B. 算法设计  
C. 模块之间的接口设计                      D. 数据库的物理设计

以下关于文档的叙述中，不正确的是\_\_（33）\_\_。

- （33） A. 文档也是软件产品的一部分，没有文档的软件就不能称之为软件  
B. 文档只对软件维护活动有用，对开发活动意义不大  
C. 软件文档的编制在软件开发工作中占有突出的地位和相当大的工作量  
D. 高质量文档对于软件产品的效益有着重要的意义



用白盒测试技术对下面流程图进行测试，至少采用（34）个测试用例才可以实现路径覆盖。



- (34) **A. 3**                      B. 4                      C. 6                      D. 8

软件可维护性是一个系统在特定的时间间隔内可以正常进行维护活动的概率。用 MTTF 和 MTTR 分别表示平均无故障时间和平均故障修复时间，则软件可维护性计算公式为（35）。

- (35) A.  $MTTF / (1+MTTF)$                       B.  $1 / (1+MTTF)$   
C.  $MTTR / (1+MTTR)$                       **D.  $1 / (1+MTTR)$**

某搜索引擎在交付后，开发人员修改了其中的索引方法，使得用户可以更快地得到搜索结果。这种修改属于（36）维护。

- (36) A. 正确性                      B. 适应性                      **C. 完善性**                      D. 预防性

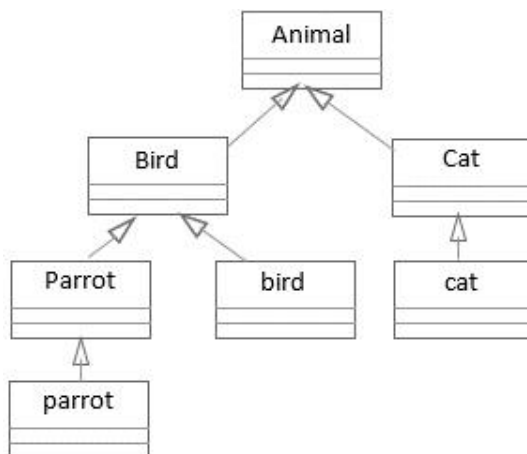
面向对象分析时，执行的活动顺序通常是（37）。

- (37) **A. 认定对象、组织对象、描述对象的相互作用、确定对象的操作**  
B. 认定对象、定义属性、组织对象、确定对象的操作  
C. 认定对象、描述对象间的相互作用、确定对象的操作、识别包  
D. 识别类及对象、识别关系、定义属性、确定对象的操作

采用面向对象方法进行系统设计时，不应该强迫客户依赖于他们不用的方法，接口属于客户，不属于它所在的类的层次结构。即：依赖于抽象，不要依赖于具体，同时在抽象级别不应该有对于细节的依赖。这属于 (38)。

- (38) A. 单一责任 B. 开放—封闭  
C. 接口分离 D. 里氏替换

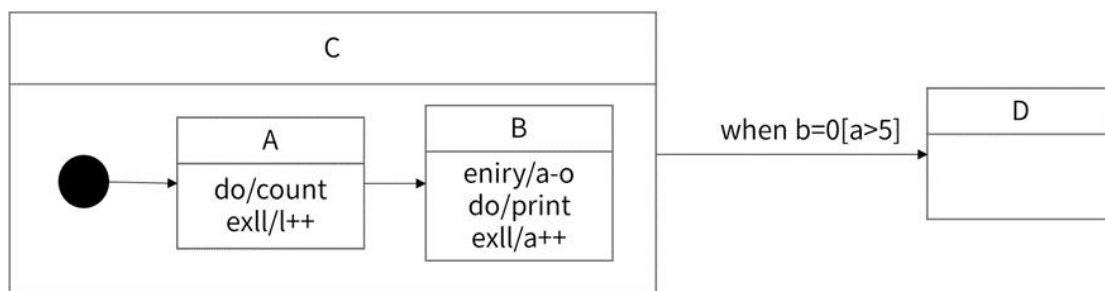
假设 Bird 和 Cat 是 Animal 的子类，Parrot 是 Bird 的子类，bird 是 Bird 的一个对象，cat 是 Cat 的一个对象，parrot 是 Parrot 的一个对象。以下叙述中不正确的是 (39)。假设 Animal 类中定义接口 move()，Bird、Cat 和 Parrot 分别实现自己的 move()，调用 move() 时，不同对象收到同一消息可以产生各自不同的结果，这一现象称为 (40)。



- (39) A. cat 和 bird 可看作是 Animal 的对象  
B. parrot 和 bird 可看作是 Animal 的对象  
C. bird 可看作是 Parrot 的对象  
D. parrot 可以看作是 Bird 的对象

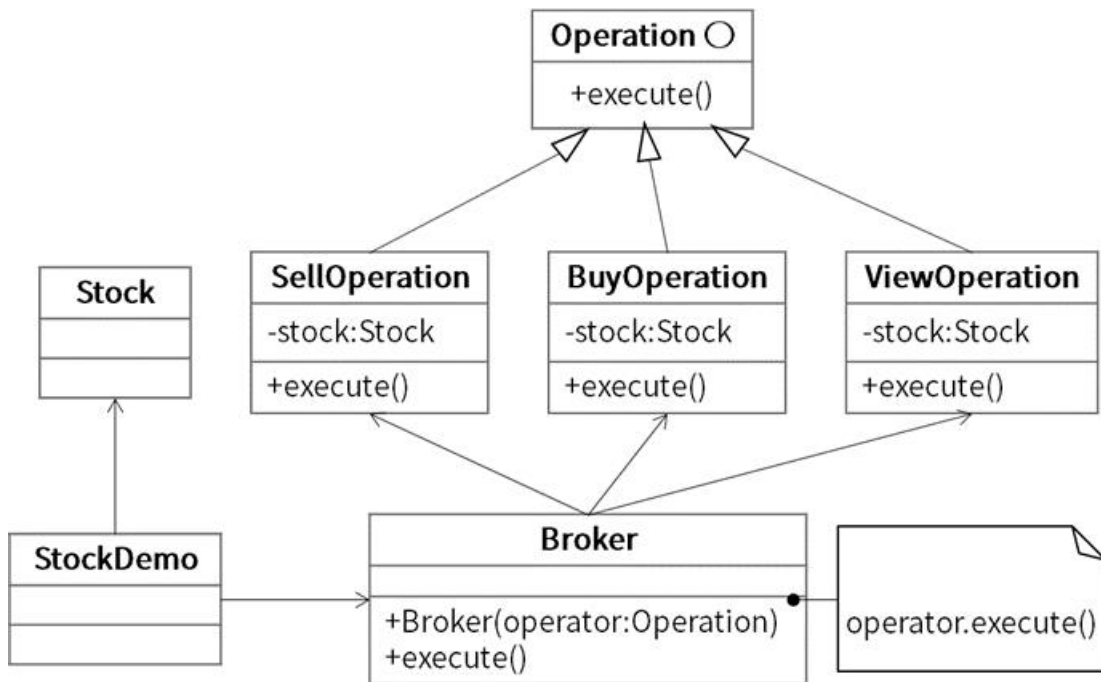
- (40) A. 封装 B. 继承 C. 消息传递 D. 多态

当 UML 状态用于对系统、类或用例的动态方面建模时，通常是对 (41) 建模。以下 UML 状态图中，假设活动的状态是 A，事件  $b=0$  发生并且  $a>5$ ，发生条件是 C 状态到 D 状态的转换条件的是 (42)，D 变为活动的状态，有关状态图的叙述中，不正确的是 (43)。



- (41) A. 系统的词汇  
B. 反应型对象  
C. 活动流程  
D. 对象快照
- (42) A. 一旦状态 A 的 **exit** 动作完成或如果当前执行 **do** 动作，则终止执行  
B. 一旦状态 A 和 B 的所有动作完成  
C. 一旦正在进行的状态 A 完成  
D. 一旦状态 B 的 **exit** 动作完成
- (43) A. 动作可以在状态内执行，也可以在状态转换时执行  
B. 当处罚转换的事件发生并且转换没有指定的监护条件时，对象将离开当前状态，并且其 **do** 动作终止  
C. **when (b=5)** 称为时间事件  
D. 状态由事件触发

股票交易中，股票代理（Broker）根据客户发出的股票操作指示进行股票的买卖，设计如下所示类图。该设计模式采用（44）模式将一个请求封装为一个对象，从而使得以不同的请求对客户进行参数化；对请求排队或记录请求日志，以及支持可撤销的操作，其中，（45）声明执行操作的接口。该模式属于（46）模式，该模式适用于：（47）。



- (44) A. 命令 (Command) B. 观察者 (Observer)  
C. 状态 (State) D. 中介者 (Mediator)
- (45) A. Operation B. SellOperation/BuyOperation/ViewOperation  
C. Broker D. Stock
- (46) A. 结构类型 B. 结构型对象  
C. 创建类型 D. 行为型对象
- (47) A. 一个对象必须通知其他对象，而它又不能假定其他对象是谁  
B. 抽象出执行的动作以参数化某对象  
C. 一个对象的行为决定于其状态且必须在运行时刻根据状态改变行为  
D. 一个对象引用其他很多对象并且直接与这些对象通信，导致难以复用该对象

设有描述简单算术表达式的上下文无关文法如下，其中 **id** 表示单字母。

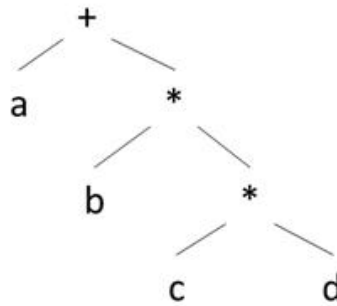
$E \rightarrow E+T \mid T$

$T \rightarrow F*T \mid F$

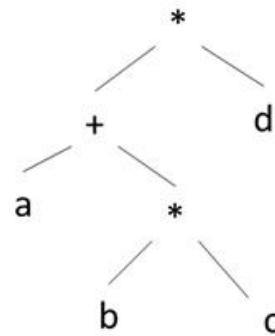
$F \rightarrow \text{id}$

与使用该文法描述的表达式 **a+b\*c\*d** 相符的语法树为 (48)。

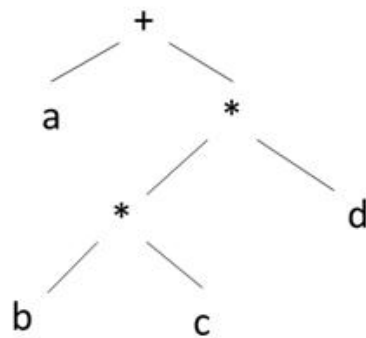
(48) A.



B.

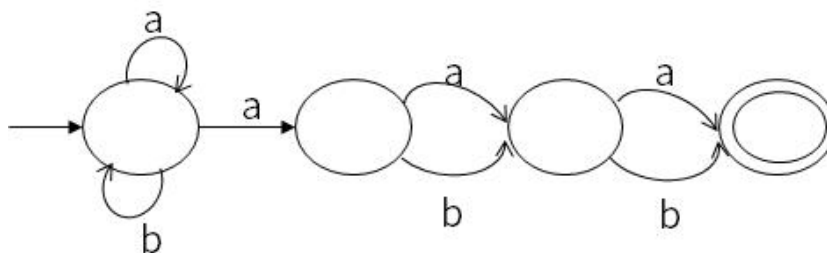


C.



D. 暂无

下图所示有限自动机（DFA）是 (49)。



- (49) A. 确定的有限自动机，它能识别以 **bab** 结尾的  
 B. 确定的有限自动机，他不能识别以 **bab** 结尾的  
 C. 非确定的有限自动机，他能识别以 **bab** 结尾的  
 D. 非确定的有限自动机，他不能识别以 **bab** 结尾的

函数 `foo()`、`hoo()` 定义如下，调用函数 `hoo(a, x)` 的两个参数分别采用引用调用（call by reference）和值调用（call by value）方式传递，则函数调用 `foo(5)` 的输出结果为 (50)。

`foo (int x)`

```
int a = 2;
hoo(a, x)
print(a, x);
```

`hoo (int &x, int y)`

```
y = y + 10;
x = y - x;
return;
```

(50) A. 2, 5

B. 2, 15

C. 13, 5

D. 13, 15

如下 E-R 图中，两个实体 `R1`、`R2` 之间有一个联系 `E`，当 `E` 的类型为 (51) 时必须将 `E` 转换成一个独立的关系模式？



(51) A. 1:1

B. 1:\*

C. \*:1

D. \*:\*

给定关系  $R(U, F)$ ，其中

$U = \{A, B, C, D, E, H\}$ ,  $F = \{A \rightarrow B, B \rightarrow DH, A \rightarrow H, C \rightarrow E\}$ ,

关系有 (52)， $F$  中 (53)。

(52) A. 一个候选码 A

B. 两个候选码 A、B

C. 一个候选码 AC

D. 两个候选码 A、C

(53) A. 不存在传递依赖，但存在冗余函数依赖

B. 既不存在传递依赖，也不存在冗余函数依赖

C. 存在传递依赖  $A \rightarrow D$  和  $A \rightarrow H$ ，但不存在冗余函数依赖

D. 存在传递依赖  $A \rightarrow D$  和  $A \rightarrow H$ ，并且还可能存在冗余函数依赖

某销售公司员工关系  $E$ （工号、姓名、部门名、电话、住址），商品关系  $C$ （商品号、商品名、库存数）和销售关系  $EC$ （工号、商品号、销售数、销售日期）。查询“销售部 1”在 2020 年 11 月 11 日销售“HUAWEI Mate40”商品的员工工号、姓名、部门名及其销售的商品名，销售数的关系代数表达式为  $\pi_{1, 2, 3, 7, 8} \left( (54) \bowtie ((55) \bowtie (56)) \right)$

- (54) A.  $\sigma_{3=\text{销售部 1}}(E)$  B.  $\sigma_{3=\text{销售部 1}}(C)$   
 C.  $\sigma_{3='销售部 1'}(E)$  D.  $\sigma_{3='销售部 1'}(C)$
- (55) A.  $\pi_{2, 3}(\sigma_{2='HUAWEI Mate40'}(C))$   
 B.  $\pi_{1, 2}(\sigma_{2='HUAWEI Mate40'}(C))$   
 C.  $\pi_{2, 3}(\sigma_{2='HUAWEI Mate40'}(EC))$   
 D.  $\pi_{1, 2}(\sigma_{2='HUAWEI Mate40'}(EC))$
- (56) A.  $\sigma_{4='2020 年 11 月 11 日'}(C)$   
 B.  $\sigma_{3='2020 年 11 月 11 日'}(C)$   
 C.  $\sigma_{4='2020 年 11 月 11 日'}(EC)$   
 D.  $\sigma_{3='2020 年 11 月 11 日'}(EC)$

设有栈  $S$  和队列  $Q$  初始状态为空，数据元素序列  $a, b, c, d, e, f$  依次通过栈  $S$ ，且多个元素从  $S$  出栈后立即进入队列  $Q$ ，若出队的序列是  $b, d, f, e, c, a$ ，则  $S$  中的元素最多时，栈底到栈顶的元素依次为 (57)。

- (57) A.  $a, b, c$  B.  $a, c, d$   
 C.  $a, c, e, f$  D.  $a, d, f, e$

当二叉树的结点数目确定时，(58) 的高度一定是最小的。

- (58) A. 二叉排序树 B. 完全二叉树  
 C. 线索二叉树 D. 最优二叉树

\_\_\_\_(59)\_\_\_\_是对稀疏矩阵进行压缩存储的方式。

- (59) A. 二维数组和双向链表 B. 三元组顺序表和十字链表  
C. 邻接矩阵和十字链表 D. 索引顺序表和双向链表

设用线性探查法解决冲突构造哈希表，且哈希函数为  $H(key) = key \% m$ ，若在该哈希表中查找某关键字  $e$  是成功的且与多个关键字进行了比较，则\_\_\_\_(60)\_\_\_\_。

- (60) A. 这些关键字形成一个有序序列  
B. 这些关键字都不是  $e$  的同义词  
C. 这些关键字都是  $e$  的同义词  
D. 这些关键字的第一个可以不是  $e$  的同义词

对于一个初始无序的关键字序列，在下面的排序方法中，\_\_\_\_(61)\_\_\_\_第一趟排序结束后，一定能将序列中的某个元素在最终有序序列中的位置确定下来。

①直接插入排序②冒泡排序③简单选择排序④堆排序⑤快速排序⑥归并排序

- (61) A. ①②③⑥ B. ①②③⑤⑥ C. ②③④⑤ D. ③④⑤⑥

对数组  $A = (2, 8, 7, 1, 3, 5, 6, 4)$  构建大顶堆为\_\_\_\_(62)\_\_\_\_。(用数组表示)

- (62) A. (1, 2, 3, 4, 5, 6, 7, 8) B. (1, 2, 5, 4, 3, 7, 6, 8)  
C. (8, 4, 7, 2, 3, 5, 6, 1) D. (8, 7, 6, 5, 4, 3, 2, 1)

最大尺寸和问题描述为，在  $n$  个整数（包含负数）的数组  $A$  中，求之和最大的非空连续子数组，如数组  $A = (-2, 11, -4, 13, -5, -2)$ ，其中子数组  $B = (11, -4, 13)$  具有最大子段和 20 ( $11 - 4 + 13 = 20$ )。求解该问题，可以将数组分为两个  $n/2$  个整数的子数组最大子段或或者在前半段，或者在后半段，或者跨越中间元素，通过该方法继续划分问题，直至最后求出最大子段和，该算法的时间复杂度为\_\_\_\_(63)\_\_\_\_。

- (63) A.  $O(n \lg n)$  B.  $O(n^2)$  C.  $O(n^2 \lg n)$  D.  $O(n^3)$



在求解某问题时，经过分析发现该问题具有最优子结构和重叠子问题性质。则适合（64）算法设计策略得到最优解。若了解问题的解空间，并以广度优先的方式搜索解空间，则采用的是（65）算法策略。

- （64） A. 分治                      B. 贪心                      C. 动态规划                      D. 回溯
- （65） A. 动态规划                      B. 贪心                      C. 回溯                      D. 分支限界

IPv6 地址长度为（66）bit。

- （66） A. 16                      B. 32                      C. 64                      D. 128

在地址栏输入 www. abc. com，浏览器默认的应用层协议是（67）。

- （67） A. HTTP                      B. DNS                      C. TCP                      D. FTP

使用电子邮件客户端向服务器发送邮件的协议是（68）。

- （68） A. SMTP                      B. POP3                      C. IMAP4                      D. MIME

以下 Windows 系统中用于清除本地 DNS 缓存的命令是（69）。

- （69） A. ipconfig/release                      B. ipconfig/flushdns
- C. ipconfig/displaydns                      D. ipconfig/registerdns

在网络系统设计时，不可能使所有设计目标都能达到最优，下列措施中最为合理的是（70）。

- （70） A. 尽量让最低建设成本目标达到最优
- B. 尽量让最短的故障时间达到最优
- C. 尽量让最大的安全性目标达到最优
- D. 尽量让优先级较高的目标达到最优

Designing object -oriented software is hard, and designing (71) object -oriented software is even harder. You must find pertinent (相关的) objects, factor them into class at the right granularity, define class interfaces and inheritances, and establish key relationships among them. You design should be specific to the problem at hand but also (72) enough to address future problems and requirements. You also want to avoid redesign, or at least minimize it. Experienced object -oriented designers will tell you that a reusable and flexible design is difficult if not impossible to get "right" the first time. Before a design is finished, they usually try to reuse it several times, modifying it each time.

Yet experienced object-oriented designers do make good designs. Meanwhile new designers are (73) by the options available and tend to fall back on non-object-oriented techniques they've used before. It takes a long time for novices to learn what good object-oriented design is all about. Experienced designers evidently know something inexperienced ones don't. What is it?

One thing expert designers know not to do is solve every problem from first principles. Rather, they reuse solutions that have worked for them in the past. When they find a good (74). They use it again and again. Such experience is part of what makes them experts. Consequently, you'll find (75) patterns of classes and communicating objects in many object-oriented systems.

- |      |              |              |                 |                 |
|------|--------------|--------------|-----------------|-----------------|
| (71) | A. runnable  | B. right     | C. reusable     | D. pertinent    |
| (72) | A. clear     | B. general   | C. personalized | D. customized   |
| (73) | A. excited   | B. shoken    | C. surprised    | D. Foverwhelmed |
| (74) | A. tool      | B. component | C. system       | D. solution     |
| (75) | A. recurring | B. right     | C. experienced  | D. past         |

设计面向对象的软件很难，而设计（可复用的）面向对象软件就更难了。

你必须找到合适的（相关的）对象，以适当的粒度将它们划分为类，定义类接口和继承，并在它们之间建立关键关系。

你的设计应该针对眼前的问题，但（一般）足以解决未来的问题和要求。

你也要避免重新设计，或者至少最小化它。

有经验的面向对象设计师会告诉你，一个可重用和灵活的设计是很难第一次就“正确”的。

在设计完成之前，他们通常会尝试多次重复使用，每次都是这样。

然而，有经验的面向对象设计师确实能做出好的设计。

同时，新的设计师们对可用的选项感到不安（第三题选项不知所措的意思），他们倾向于使用以前使用的非面向对象技术。

电影要花很长时间才能了解好的面向对象的设计。

经验丰富的设计师显然知道一些缺乏经验的东西。它是什么？有一件事专家设计师知道不能做的就是从第一原理解决每一个问题。相反，他们选择了过去对他们有效的解决方案。

当他们找到一个好的（解决方案）。他们一次又一次地使用它。这样的经验是他们成为专家的部分原因。

因此，您将在许多面向对象系统中找到第五题个类和通信对象的模式。

- A. runnable可运行的
- B. right对的
- C. reusable可复用的
- D. pertinent中肯的，相关的
- A. clear清除
- B. general总则
- C. personalized个性化
- D. customized定制
- A. excited兴奋
- B. shoken 震惊
- C. surprised惊讶于
- D. overwhelmed不知所措
- A. tool工具
- B. component组成部分
- C. system系统
- D. solution解决方案
- A. recurring循环
- B. right是吗
- C. experienced经验丰富
- D. past过去

【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
- 4、免费督考群



**微信扫一扫，立马获取**



**最新免费题库**



**备考资料+督考群**

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