

202511 文老师软件设计师真题回忆

综合知识

部分真题还原如下：

软件公司的软件设计师王某根据工作安排参与了某软件项目的开发，项目进行过程中王某辞职并将相关开发任务交接给同事李某，因此项目负责人将该软件开发人员名单的王某变更为李某，该软件开发完成后王某也不再署名。那么，该项目负责人的行为（ ）。

- A. 不构成侵权，因为王某不是软件著作权人
- B. 侵犯了王某的软件著作权
- C. 侵犯了王某作为开发者的署名权
- D. 不构成侵权，因为王某已不是项目组成员

答案：C

解析：根据著作权法，软件的开发者享有署名权。王某参与了开发，即使他后来辞职，他仍然对他在开发期间所做的贡献享有署名权。项目负责人将他的名字从开发人员名单中移除，侵犯了他的署名权。软件著作权通常属于公司，但署名权是开发者个人的权利。所以，答案应该是 C。

下列关于报文摘要的叙述中，错误的是（ ）。

- A. 在物联网环境中，可以为资源受限的传感器生成轻量级摘要
- B. 对电子文档生成摘要，接收方用私钥解密
- C. 对发送方报文生成的摘要值，接收方应重新计算并与原始摘要比对
- D. 报文摘要仅用于验证完整性，传输数据仍需 TLS 加密

答案：B

解析：报文摘要用于验证数据完整性，但本身不需要加密或解密。选项 B 错误，因为接收方不需要用私钥解密摘要；而是发送方可能用私钥对摘要进行数字签名，接收方用公钥验证签名。

（ ）是评估软件可维护性的关键度量。

- A. 用户数量
- B. 功能点数量
- C. 数据库容量
- D. 平均修复时间 (MTTR)

答案：D

解析：软件可维护性是指软件被修改、修复或改进的容易程度。平均修复时间 (Mean Time To Repair, MTTR) 是衡量修复软件缺陷所需时间的关键度量，直接反映软件的可维护性。其他选项如用

户数量、功能点数量和数据库容量与可维护性无关，因此选项 D 正确。

使用 SPooling 技术的主要目的是（ ）。

- A. 将独占设备虚拟为共享设备
- B. 提高进程并发度
- C. 加快内存访问速度
- D. 提高 CPU 利用率

答案：A

解析：SPooling 技术的主要目的是将独占设备（如打印机）虚拟为共享设备，通过缓冲和排队机制允许多个进程同时使用该设备，从而提高设备利用率，因此选项 A 正确。

以下关于软件著作权产生时间的叙述中，正确的是（ ）。

- A. 自开发者获得软件著作权登记证书起
- B. 自开发计划文档完成时
- C. 自软件首次公开发表时
- D. 自软件开发完成之日起

答案：D

解析：根据《计算机软件保护条例》，软件著作权自软件开发完成之日起自动产生，不需要登记或发表。登记证书仅起证明作用，首次发表也不是著作权产生的条件，因此选项 D 正确。

以下（ ）不是面向对象设计的基本原则。

- A. 过程优先
- B. 单一职责
- C. 依赖倒置
- D. 开放-封闭

答案：A

解析：面向对象设计的基本原则包括单一职责原则、依赖倒置原则、开放-封闭原则等。但“过程优先”不是面向对象的原则，而是面向过程编程的特点，因此选项 A 不是面向对象设计的基本原则。

黑盒测试的主要依据是（ ）。

- A. 代码复杂度
- B. 软件需求规格说明书
- C. 设计文档
- D. 程序内部逻辑

答案：B

解析：黑盒测试基于软件的外部功能和需求，而不考虑内部代码结构。软件需求规格说明书定义了软件的功能和性能要求，是黑盒测试的主要依据，因此选项 B 正确。

以下序列中，满足大顶堆条件的是（ ）。

- A. 50, 38, 40, 28, 43, 16, 25, 19
- B. 50, 37, 45, 25, 16, 40, 35, 19
- C. 45, 25, 15, 16, 24, 20, 7, 10
- D. 55, 43, 45, 48, 52, 29, 75, 15

答案：B

解析：大顶堆要求每个节点的值都大于或等于其子节点的值。对选项 B 的序列 (50, 37, 45, 25, 16, 40, 35, 19) 进行层次遍历检查：根节点 50 大于子节点 37 和 45；37 大于子节点 25 和 16；45 大于子节点 40 和 35；25 大于子节点 19。所有节点满足大顶堆条件。因此选项 B 正确。

VLAN 技术基于 MAC 地址实现逻辑分组，工作在 OSI 参考模型的（ ）。

- A. 传输层
- B. 数据链路层
- C. 网络层
- D. 物理层

答案：B

解析：VLAN（虚拟局域网）技术基于 MAC 地址（数据链路层地址）实现逻辑分组，因此工作在 OSI 参考模型的数据链路层（第二层），因此选项 B 正确。

在 SQL 查询中，用于过滤分组后数据的子句是（ ）。

- A. GROUP BY
- B. HAVING
- C. ORDER BY
- D. WHEREB

答案：B

解析：在 SQL 查询中，HAVING 子句用于对分组后的数据进行过滤（通常与 GROUP BY 一起使用），而 WHERE 子句用于在分组前过滤行。ORDER BY 用于排序，GROUP BY 用于分组。因此选项 B 正确。

部分知识点回忆如下：

邮箱用的加密协议是什么

选项有：S/MIME，pop3，SMTP，还有一个选项忘了。

哈希表的加密协议？（提到哈希表了，关键词是哈希和加密）

选项有 ECC, AES, RAS 还有俩忘了

考了折半查找，活动图最短时间，有向边（矩阵），大顶堆，考设计模式的选项包含策略，桥接，工厂，观察者（一道题包含四道选择），问的大概就是什么样的情况下用哪种设计模式

哪个算法会产生饥饿？

选项：电梯，最少使用，最近使用（就常见的那几个）

选项应该是 电梯算法 最近寻道 先来先进 循环算法

给了哈夫曼树的加权值（0.18, 0.32...），问排序

避免脏读（还是预防来着 记不清了）

选项有个重复读，未提交数据，已提交数据，还有个忘了

还考了一个计算流水线时间的题（比较简单）答案应该是 203

↑补充一下 流水线 $(2n_s - 2n_s + 1n_s)$ 100 次 的原题

问哪个是 202.31.17.8 的同一子网优先选择哪个子网。

选项 202.0.0.0 202.31.0.0, 202.31.17.0 202.31.16.0

问以下哪组数据符合大顶堆的条件，选项忘记了

COCOMO 模型是用来评估（）

每秒百万次浮点运算的英文是什么（CPI, API, MFLOPS, ?）

A 受公司委托开发一个软件，期间 A 离职并且把工作交给了 B 同事，软件开发完成后公司负责人没有在软件上署名 A 的名字，问是否侵权。

vpn 技术中没使用的是，有个选项是入侵检测技术

案例分析

部分知识点回忆如下：

算法题考的是，皇后问题，皇后问题用的什么（答案是回溯）并求解；（15 年原题，但是没给常量和变量说明 靠自己悟，只说明了 i 代表皇后所在的行 $X[i]$ 代表所在的列。

可参考 2015 年皇后问题原题，看懂该算法原理：

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

n-皇后问题是在 n 行 n 列的棋盘上放置 n 个皇后，使得皇后彼此之间不受攻击，其规则是任意两个皇后不在同一行、同一列和相同的对角线上。

拟采用以下思路解决 n-皇后问题：第 i 个皇后放在第 i 行。从第一个皇后开始，对每个皇后，从其对应行（第 i 个皇后对应第 i 行）的第一列开始尝试放置，若可以放置，确定该位置，考虑下一个皇后；若与之前的皇后冲突，则考虑下一列；若超出最后一列，则重新确定上一个皇后的位置。重复该过程，直到找到所有的放置方案。

【C 代码】

下面是算法的 C 语言实现。

(1) 常量和变量说明

pos: 一维数组, pos[i]表示第 i 个皇后放置在第 i 行的具体位置

count: 统计放置方案数

i, j, k: 变量

N: 皇后数

(2) C 程序

```
#include <stdio.h>
#include <math.h>
#define N4
/*判断第 k 个皇后目前放置位置是否与前面的皇后冲突*/
int isplace(int pos[], int k) {
    int i;
    for(i=1; i<k; i++) {
        if( (1) || fabs(i-k) == fabs(pos[i] - pos[k])) {
            return();
        }
    }
    return 1;
}

int main() {
    int i, j, count=1;
    int pos[N+1];
    //初始化位置
    for(i=1; i<=N; i++) {
        pos[i]=0;
    }
    (2) ;
    while(j>=1) {
        pos[j]= pos[j]+1;
        /*尝试摆放第 i 个皇后*/
        while(pos[j]<=N&& (3) ) {
            pos[j]= pos[j]+1;
        }
        /*得到一个摆放方案*/
        if(pos[j]<=N&&j== N) {
            printf("方案%d: ", count++);
            for(i=1; i<=N; i++) {
                printf("%d ", pos[i]);
            }
        }
    }
}
```

```
}  
printf("\n");  
}  
/*考虑下一个皇后*/  
if(pos[j]<=N&& (4) ) {  
    j=j+1;  
} else{ //返回考虑上一个皇后  
    pos[j]=0;  
    (5) ;  
}  
}  
return 1;  
}
```

问题 1 (10 分)

根据以上说明和 C 代码，填充 C 代码中的空 (1) ~ (5)。

问题 2 (2 分)

根据以上说明和 C 代码，算法采用了 (6) 设计策略。

问题 3 (3 分)

上述 C 代码的输出为：(7) 。

答案：

问题 1: (1) pos[i]==pos[k] (2) j=1
(3)! isplace(pos, j) (4) j<N (5) j=j-1

问题 2: (6) 回溯法

问题 3: (7) 方案 1: 2 4 1 3; 方案 2: 3 1 4 2。

解析：

问题 1 考察算法填空，需要根据题意理解算法原理，根据题干描述。空 (1) 所在的代码行判断皇后合法放置的约束条件，即不在同一行，这通过把第 i 个皇后放在第 i 行实现，条件“fabs(i-k)==fabs(pos[i]-pos[k])”判断的是当前摆放的皇后是否与之前摆放的皇后在同一对角线上。因此，空 (1) 判断的是当前摆放的皇后是否和之前摆放的皇后在同一列上，即应填入“pos[i]==pos[k]”。根据算法思想和主函数上下文，空 (2) 处应该考虑第 1 个皇后，即初始化 j 为 1，空 (2) 填写“j=1”。空 (3) 所在的行是判断放置第 j 个皇后的位置是否合适，“pos[j] <=N”表示在该行的合法列上，但还需要进一步判断是否与前面的皇后有冲突，根据满足条件后的语句，尝试放入下一列，因此空 (3) 处填入“!isplace(pos, j)”。根据前面的注释，空 (4) 所在的行是考虑下一个皇后，其条件是，当前皇后找到了合适的位置，而且还存在下一个皇后，因此空 (4) 处应填入“j < N”。根据下面的注释，若当前皇后没有找到合适的位置，则应回溯，即再次考虑上一个皇后的位置，因此空 (5) 处填入“j=j-1”。

问题 2 考察算法分类，从上述题干的叙述和 C 代码很容易看出，从第一个皇后开始，对每个皇后

总是从第个位置开始尝试，找到可以放置的合法位置；若某个皇后在对应的行上没有合法位置，则回溯到上一个皇后，尝试将上一个皇后放置另外的位置。这是典型的深度优先的系统搜索方式，即回溯法的思想。

问题 3 在理解了问题 1 的算法原理后，可以直接运行得出答案。

E-R 题试题有出租车公司，挂靠（联系），租客，车辆，第三题要求补充实体（车主）和联系（有个多对多的关系车主和车辆）给出主键和外键约束，补充的联系里有拥有一个车主可以拥有多辆车，一辆车可以有多个车主，车主有个什么车辆股权比例。第一题时不要添加实体，添加联系，

用例题是无人机物流配送系统。有无人机管理员，普通用户，无人机操作员（好像是这个，具体记不清了）操作无人机，普通用户可以下订单，查询订单，管理员修改订单，管理员也可以查询订单状态。系统监控配送情况。异常处理，系统检测出问题，提醒无人机操作员和管理员处理，还有什么故障管理，会通知管理员。类图是有两个聚合关系，四个类，有一个类 C1 里包含（+货物信息），还有个类 C2 叫飞行状态（设计一个聚合关系），还有个类是普通用户（不知道有没有记混，可能也没有），最后 C4 也是聚合关系，指向 C1

↑ 补充一下：普通用户类 ----- C1 类（+货物信息） ----- C2 ---聚合---飞行状态类
|（聚合）
C3

问 C1 C2 C3 是什么类

数据流图是个 ai 的智慧系统（基于大模型的视频学习系统），记得有老师，管理员，学生，大模型，知识库，加工包括互动，上传视频资源，问答，回复，汇总报告。让补充四个缺失的数据流及加工

学生可以从播放列表观看视频。学生可以点赞评价等互动。学生可以提问，大模式会根据历史问题和知识库给答案。

教师可以上传知识和视频资源，更新知识库。

管理员可以上传知识，更新知识库。

汇总报告里面包含资源标志，资源名，汇总信息，汇总信息里面有学习时长，学习人数等。

题目第三个是汇总报告的属性。

Java 题是链接传感器和设备（好像是这个）有两个接口 device 和 sensor，两个实现接口的类，有 connector 的对象（还是类），先创建，再链接