

【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

采用微内核结构的操作系统提高了系统的灵活性和可扩展性，(1)

- (1) A. 并增强了系统的可靠性和可移植性，可运行于分布式系统中
- B. 并增强了系统的可靠性和可移植性，但不适用于分布式系统
- C. 但降低了系统的可靠性和可移植性，可运行于分布式系统中
- D. 但降低了系统的可靠性和可移植性，不适用于分布式系统

【答案】A

【解析】本题考查操作系统的基本概念。

在设计微内核 OS 时，采用了面向对象的技术，其中的“封装”，“继承”，“对象类”和“多态性”，以及在对象之间采用消息传递机制等，都十分有利于提高系统的“正确性”、“可靠性”、“易修改性”、“易扩展性”等，而且还能显著地减少开发系统所付出的开销。采用微内核结构的操作系统与传统的操作系统相比，其优点是提高了系统的灵活性、可扩充性，增强了系统的可靠性，提供了对分布式系统的支持。其原因如下。

①灵活性和可扩展性：由于微内核 OS 的许多功能是由相对独立的服务器软件来实现的，当开发了新的硬件和软件时，微内核 OS 只需在相应的服务器中增加新的功能，或再增加一个专门的服务器。与此同时，也必然改善系统的灵活性，不仅可在操作系统中增加新的功能，还可修改原有功能，以及删除已过时的功能，以形成一个更为精干有效的操作系统。

②增强了系统的可靠性和可移植性：由于微内核是出于精心设计和严格测试的，容易保证其正确性；另一方面是它提供了规范而精简的应用程序接口（API），为微内核外部的程序编制高质量的代码创造了条件。此外，由于所有服务器都是运行在用户态，服务器与服务器之间采用的是消息传递通信机制，因此，当某个服务器出现错误时，不会影响内核，也不会影响其他服务器。另外，由于在微内核结构的操作系统中，所有与特定 CPU 和 I/O 设备硬件有关的代码，均放在内核和内核下面的硬件隐藏层中，而操作系统其他绝大部分（即各种服务器）均与硬件平台无关，因而，把操作系统移植到另一个计算机硬件平台上所需作的修改是比较小的。

③提供了对分布式系统的支持：由于在微内核 OS 中，客户和服务器之间以及服务器和服务器之间的通信，是采用消息传递通信机制进行的，致使微内核 OS 能很好地支持分布式系统和网络系统。事实上，只要在分布式系统中赋予所有进程和服务器唯一的标识符，在微内核中再配置一张系统映射表（即进程和服务器的标识符与它们所驻留的机器之间的对应表），在进行客户与服务器通信时，只需在所发送的消息中标上发送进程和接收进程的标识符，微内核便可利用系统映射表将消息发往目标，而无论目标是驻留在哪台机器上。

若操作系统文件管理程序正在将修改后的_(2)_文件写回磁盘时系统发生崩溃，对系统的影响相对较大。

- (2) A. 用户数据 B. 用户程序 C. 系统目录 D. 空闲块管理

【答案】C

【解析】本题考查操作系统的基本概念。

操作系统为了实现“按名存取”，必须为每个文件设置用于描述和控制文件的数据结构，专门用于文件的检索，因此至少应包括文件名和存放文件的物理地址，该数据结构称为文件控制块（File Control Block, FCB），文件控制块的有序集合称为文件目录，或称系统目录文件。若操作系统正在将修改后的系统目录文件写回磁盘时系统发生崩溃，则对系统的影响相对较大。

某虚拟存储系统采用最近最少使用（LRU）页面淘汰算法，假定系统为每个作业分配 4 个页面的主存空间，其中一个页面用来存放程序。现有某作业的程序如下：

```
Var A: Array[1..100, 1..100] Of integer;
    i, j: integer;
FOR i:=1 to 100 DO
FOR j:=1 to 100 DO
    A[i, j]:=0;
```

设每个页面可存放 200 个整数变量，变量 i、j 存放在程序页中。初始时，程序及 i、j 均已在内存，其余 3 页为空。若矩阵 A 按行序存放，那么当程序执行完后共产生_(3)_次缺页中断；若矩阵 A 按列序存放，那么当程序执行完后共产生_(4)_次缺页中断。

- (3) A. 50 B. 100 C. 5000 D. 10000
(4) A. 50 B. 100 C. 5000 D. 10000

【答案】A C

【解析】本题考查计算机存储管理方面的基本知识。

矩阵 A[100][100] 总共有 100 行、100 列，若矩阵 A 按行序存放，那么每一个页面可以存放 2 行，也就是说矩阵的 2 行刚好放在 1 页内，访问它们需要中断 1 次，这样 100 行总共需要中断 50 次。

若矩阵 A 按列序存放，那么每一个页面可以存放 2 列，也就是说矩阵的 2 列刚好放在 1

页内，由于内循环“FORj:=1 to 100 DO”是按列序变化，访问它们需要中断 50 次，这样 100 行总共需要中断 50X100 次。

在数据库设计的(5)阶段进行关系规范化。

- (5) A. 需求分析 B. 概念设计 C. 逻辑设计 D. 物理设计

【答案】C

【解析】本题考查数据库设计的基础知识。

数据库设计分为用户需求分析、概念设计、逻辑设计和物理设计 4 个主要阶段。将抽象的概念模型转化为与选用的 DBMS 产品所支持的数据模型相符合的逻辑模型，它是物理设计的基础。包括模式初始设计、子模式设计、应用程序设计、模式评价以及模式求精。

逻辑设计阶段的任务是将概念模型设计阶段得到的基本 E-R 图转换为与选用的 DBMS 产品所支持的数据模型相符合的逻辑结构。如采用基于 E-R 模型的数据库设计方法，该阶段就是将所设计的 E-R 模型转换为某个 DBMS 所支持的数据模型；如采用用户视图法，则应进行模式的规范化，列出所有的关键字以及用数据结构图描述表集合中的约束与联系，汇总各用户视图的设计结果，将所有的用户视图合成一个复杂的数据库系统。

某数据库中有员工关系 E（员工号，姓名，部门，职称，月薪）；产品关系 P（产品号，产品名称，型号，尺寸，颜色）；仓库关系 W（仓库号，仓库名称，地址，负责人）库存关系 I（仓库号，产品号，产品数量）。

a. 若数据库设计中要求：

- ①仓库关系 W 中的“负责人”引用员工关系的员工号
- ②库存关系 I 中的“仓库号，产品号”唯一标识 I 中的每一个记录
- ③员工关系 E 中的职称为“工程师”的月薪不能低于 3500 元

则①②③依次要满足的完整性约束是(6)。

b. 若需得到每种产品的名称和该产品的总库存量，则对应的查询语句为：

```
SELECT 产品名称, SUM (产品数量)
FROM P, I
WHERE P. 产品号=I. 产品号 (7);
```

- (6) A. 实体完整性、参照完整性、用户定义完整性
B. 参照完整性、实体完整性、用户定义完整性

C. 用户定义完整性、实体完整性、参照完整性

D. 实体完整性、用户定义完整性、参照完整性

(7) A. ORDER BY 产品名称

B. ORDER BY 产品数量

C. GROUP BY 产品名称

D. GROUP BY 产品数量

【答案】B C

【解析】本题考查关系的完整性约束和 SQL 的基本知识及应用。

(6) 关系模型的完整性规则是对关系的某种约束条件。关系模型中可以有三类完整性约束：实体完整性、参照完整性和用户定义的完整性。实体完整性规定基本关系的主属性不能取空值。由于①仓库关系 W 中的“负责人”引用员工关系的员工号，所以应满足参照完整性约束；②库存关系 I 中的“仓库号，产品号”唯一标识 I 中的每一个记录，所以应满足实体完整性约束；③职称为“工程师”的月薪不能低于 3500 元，是针对某一具体关系数据库的约束条件，它反映某一具体应用所涉及的数据必须满足的语义要求，所以应满足用户定义完整性约束。因此，试题 (6) 的正确答案为 B。

(7) SQL 查询是数据库中非常重要的内容。该 SQL 查询要求对查询结果进行分组，即具有相同名称的产品的元组为一组，然后计算每组的库存数量。由此可排除 A、B 和 D，所以试题 (7) 的正确答案为 C。

若对关系 R (A, B, C, D) 和 S (C, D, E) 进行关系代数运算，则表达式

$$\pi_{3,4,7}(\sigma_{4<5}(R \times S))$$

与(8)等价。

(8) A. $\pi_{C,D,E}(\sigma_{D<C}(R \times S))$

B. $\pi_{R.C,R.D,E}(\sigma_{R.D<S.C}(R \times S))$

C. $\pi_{C,D,E}(\sigma_{R.D<S.C}(R \times S))$

D. $\pi_{R.C,R.D,E}(\sigma_{D<C}(R \times S))$

【答案】B

【解析】本题考查关系代数运算方面的基础知识。

题目要求计算关系代数表达式 $\pi_{3,4,7}(\sigma_{4<5}(R \times S))$ 的结果集，其中， $R \times S$ 的属性列名分别为：R.A, R.B, R.C, R.D, S.C, S.D, S.E。从 $\sigma_{4<5}(R \times S)$ 的结果集中选取第 4 个分量(R.D)小于第 5 个分量(S.C)的元组，故 $\sigma_{4<5}(R \times S)$ 与 $\sigma_{R.D<S.C}(R \times S)$ 等价。 $\pi_{3,4,7}(\sigma_{4<5}(R \times S))$ 的含义是从 $\sigma_{4<5}(R \times S)$ 的结果集中选取第 3 列 R.C、第 4 列 R.D，和第 7 列 S.E（或 E），故 $\pi_{3,4,7}(\sigma_{4<5}(R \times S))$ 与 $\pi_{R.C,R.D,E}(\sigma_{R.D<S.C}(R \times S))$ 等价。需要说明的是第 3 列 R.C 不能简写为 C，因为关系 S 的第一列属性名也为 C，故必须标上关系名加以区别；同样，第 4 列也不能简写为 D，因为关系 S 的第 2 列属性名也为 D，故必须标上关系名加以区别。

计算机执行程序时，在一个指令周期的过程中，为了能够从内存中读指令操作码，首先是将 (9) 的内容送到地址总线上。

(9) A. 程序计数器 PC B. 指令寄存器 IR C. 状态寄存器 SR D. 通用寄存器 GR

【答案】A

【解析】

计算机执行程序时，在一个指令周期的过程中，为了能够从内存中读指令操作码，首先是将程序计数器（PC）的内容送到地址总线上。

内存按字节编址，利用 8K×4b 的存储器芯片构成 84000H 到 8FFFFH 的内存，共需 (10) 片。

(10) A. 6 B. 8 C. 12 D. 24

【答案】C

【解析】

根据题目描述，采用该存储器芯片需要构成 $8\text{FFFFH}-84000\text{H}=\text{BFFFH}$ 的空间，且内存按照字节（8b）编码，需要的容量是 $\text{BFFFH}\times 8\text{b}$ ，一片存储芯片的容量是 $8\text{K}\times 4\text{b}$ ，统一单位后相除并取最近的偶数（因为地址空间是字节），可以得到答案为 12。

某磁盘磁头从一个磁道移至另一个磁道需要 10ms。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 10 个磁道，每块的旋转延迟时间及传输时间分别为 100ms 和 2ms，则读取一个 100 块的文件需要 (11) ms 的时间。

- (11) A. 10200 B. 11000 C. 11200 D. 20200

【答案】D

【解析】

根据题目描述，读取一个连续数据需要的时间包括移动时间、旋转延迟时间和传输时间三个部分，总时间花为 $(10\times 10)+100+2=202\text{ms}$ 。一次读取一个 100 块的文件需要的时间为 $202\times 100=20200\text{ms}$ 。

计算机系统中，在 (12) 的情况下一般应采用异步传输方式。

- (12) A. CPU 访问内存 B. CPU 与 I/O 接口交换信息
C. CPU 与 PCI 总线交换信息 D. I/O 接口与打印机交换信息

【答案】D

【解析】本题考查计算机系统中数据传输的方式。

CPU 访问内存通常是同步方式，CPU 与 I/O 接口交换信息通常是同步方式，CPU 与 PCI 总线交换信息通常是同步方式，I/O 接口与打印机交换信息则通常采用基于缓存池的异步方式，因此答案为 D。

大型局域网通常划分为核心层、汇聚层和接入层，以下关于各个网络层次的描述中，不正确的是 (13)。

- (13) A. 核心层进行访问控制列表检查 B. 汇聚层定义了网络的访问策略
C. 接入层提供局域网络接入功能 D. 接入层可以使用集线器代替交换机

【答案】A

【解析】本题主要考查大型局域网的层次和各个层次的功能。

大型局域网通常划分为核心层、汇聚层和接入层，其中核心层在逻辑上只有一个，它连接多个分布层交换机，通常是一个园区中连接多个建筑物的总交换机的核心网络设备；汇聚层定义网络的访问策略；接入层提供局域网络接入功能，可以使用集线器代替交换机。

网络系统设计过程中，逻辑网络设计阶段的任务是 (14)。

- (14) A. 依据逻辑网络设计的要求，确定设备的物理分布和运行环境
B. 分析现有网络和新网络的资源分布，掌握网络的运行状态
C. 根据需求规范和通信规范，实施资源分配和安全规划
D. 理解网络应该具有的功能和性能，设计出符合用户需求的网络

【答案】C

【解析】本题主要考查网络设计方面的基础知识。

根据网络系统设计的一般规则，在逻辑网络设计阶段的任务通常是根据需求规范和通信规范，实施资源分配和安全规划。其他几个选项都不是逻辑网络设计阶段的任务。

网络系统生命周期可以划分为 5 个阶段，实施这 5 个阶段的合理顺序是 (15)。

- (15) A. 需求规范、通信规范、逻辑网络设计、物理网络设计、实施阶段
B. 需求规范、逻辑网络设计、通信规范、物理网络设计、实施阶段
C. 通信规范、物理网络设计、需求规范、逻辑网络设计、实施阶段
D. 通信规范、需求规范、逻辑网络设计、物理网络设计、实施阶段

【答案】A

【解析】本题主要考查网络系统生命周期的基础知识。

网络系统生命周期可以划分为 5 个阶段，实施这 5 个阶段的合理顺序是需求规范、通信规范、逻辑网络设计、物理网络设计、实施阶段。

假设单个 CPU 的性能为 1，则由个这种 CPU 组成的多处理机系统的性能 P 为：

$$p = \frac{n}{1 + (n - 1)a}$$

其中，a 是一个表示开销的常数。例如，a=0.1，n=4 时，P 约为 3。也就是说，由 4 个这种 CPU 组成的多机系统的性能约为 3。该公式表明，多机系统的性能有一个上限，不管 n 如何增加，p 都不会超过某个值。当 a=0.1 时，这个上限是 (16)。

- (16) A. 5 B. 10 C. 15 D. 20

【答案】B

【解析】本题主要考查多处理机系统的性能上限计算问题

$$p = \frac{n}{1 + (n-1)a}$$

根据计算公式

可以看出当 a=0.1 时，公式变为

$$p = \frac{10n}{10 + (n-1)}$$

对该表达式进行求导，得到该式的极限为 10。

以下关于系统性能的叙述中，不正确的是 (17)。

- (17) A. 常见的 Web 服务器性能评估方法有基准测试、压力测试和可靠性测试
B. 评价 Web 服务器的主要性能指标有最大并发连接数、响应延迟和吞吐量
C. 对运行系统进行性能评估的主要目的是以更好的性能/价格比更新系统
D. 当系统性能降到基本水平时，需要查找影响性能的瓶颈并消除该瓶颈

【答案】C

【解析】本题主要考查系统性能评估的主要方法和需要注意的问题。

对运行系统进行评估的主要目的是评价信息系统在性能方面的表现，找出系统可能存在的性能瓶颈。其中，常见的 Web 服务器性能评估方法有基准测试、压力测试和可靠性测试等，评价 Web 服务器的主要性能指标有最大并发连接数、响应延迟和吞吐量等。当系统性能降到基本水平时，需要查找影响性能的瓶颈并消除该瓶颈。

某大型公司欲开发一个门户系统，该系统以商业流程和企业应用为核心，将商业流程中不同的功能模块通过门户集成在一起，以提高公司的集中贸易能力、协同能力和信息管理能力。根据这种需求，采用企业 (18) 门户解决方案最为合适。

- (18) A. 信息 B. 知识 C. 应用 D. 垂直

【答案】C

【解析】本题主要考查企业应用集成

企业门户是一个信息技术平台，这个平台可以提供个性化的信息服务，为企业提供一个

单一的访问企业各种信息资源和应用程序的入口。现有的企业门户大致可以分为企业信息门户、企业知识门户和企业应用门户三种。其中企业信息门户重点强调为访问结构数据和无结构数据提供统一入口，实现收集、访问、管理和无缝集成。企业知识门户提供了一个创造、搜集和传播企业知识的平台，通过企业知识门户，员工可以与工作团队中的其他成员取得联系，寻找能够提供帮助的专家。企业应用门户是一个用来提高企业的集中贸易能力、协同能力和信息管理能力的平台。它以商业流程和企业应用为核心，将商业流程中功能不同的应用模块通过门户集成在一起，提高公司的集中贸易能力、协同能力和信息管理能力。

客户关系管理(CRM)系统将市场营销的科学管理理念通过信息技术的手段集成在软件上，能够帮助企业构建良好的客户关系。以下关于 CRM 系统的叙述中，错误的是(19)。

(19)A. 销售自动化是 CRM 系统中最基本的模块

B. 营销自动化作为销售自动化的补充，包括营销计划的编制和执行、计划结果分析等

C. CRM 系统能够与 ERP 系统在财务、制造、库存等环节进行连接，但两者关系相对松散，一般不会形成闭环结构

D. 客户服务与支持是 CRM 系统的重要功能。目前，客户服务与支持的主要手段是通过呼叫中心和互联网来实现

【答案】C

【解析】

客户关系管理(CRM)系统将市场营销的科学管理理念通过信息技术的手段集成在软件上，能够帮助企业构建良好的客户关系。在客户管理系统中，销售自动化是其中最为基本的模块，营销自动化作为销售自动化的补充，包括营销计划的编制和执行、计划结果分析等功能。客户服务与支持是 CRM 系统的重要功能。目前，客户服务与支持的主要手段有两种，分别是呼叫中心和互联网。CRM 系统能够与 ERP 系统在财务、制造、库存等环节进行连接，两者之间虽然关系比较独立，但由于两者之间具有一定的关系，因此会形成一定的闭环反馈结构。

共享数据库是一种重要的企业应用集成方式。以下关于共享数据库集成方式的叙述中，错误的是(20)。

(20)A. 共享数据库集成方式通常将应用程序的数据存储在一个共享数据库中，通过制定统一的数据库模式来处理不同应用的集成需求

B. 共享数据库为不同的应用程序提供了统一的数据存储与格式定义，能够解决不同应用程序中数据语义不一致的问题

C. 多个应用程序可能通过共享数据库频繁地读取和修改相同的数据，这会使共享数据库成为一个性能瓶颈

D. 共享数据库集成方式的一个重要限制来自外部的已封装应用，这些封装好的应用程序只能采用自 a 定义的数据库模式，调整和集成余地较小

【答案】B

【解析】

共享数据库是一种重要的企业应用集成方式，它通常将应用程序的数据存储在一个共享数据库中，通过制定统一的数据库模式来处理不同应用的集成需求。共享数据库为不同的应用程序提供了统一的数据存储与格式定义，能够在一定程度上缓解数据语义不一致的问题，但无法完全解决该问题。在共享数据库集成中，多个应用程序可能通过共享数据库频繁地读取和修改相同的数据，这会使数据库成为一个性能瓶颈。共享数据库集成方式的一个重要限制来自外部的已封装应用，这些封装好的应用程序只能采用自己定义的数据库模式，调整和集成余地较小。

某公司欲对其内部的信息系统进行集成，需要实现在系统之间快速传递可定制格式的数据包，并且当有新的数据包到达时，接收系统会自动得到通知。另外还要求支持数据重传，以确保传输的成功。针对这些集成需求，应该采用 (21) 的集成方式。

(21) A. 远程过程调用 B. 共享数据库 C. 文件传输 D. 消息传递

【答案】D

【解析】

根据题干描述，该公司需要在应用集成后实现采用可定制的格式频繁地、立即地、可靠地、异步地传输数据包。远程过程调用一般是基于同步的方式，效率较低，而且容易失败；共享数据库和文件传输的集成方式在性能方面较差，系统不能保持即时数据同步，而且容易造成应用与数据紧耦合；消息传递的集成方式能够保证数据的异步、立即、可靠传输，恰好能够满足该公司的集成需求。

详细的项目范围说明书是项目成功的关键。(22) 不应该属于范围定义的输入。

(22) A. 项目章程 B. 项目范围管理计划

C. 批准的变更申请

D. 项目文档管理方案

【答案】D

【解析】本题主要考查范围管理

在初步项目范围说明书中已文档化的主要的可交付物、假设和约束条件的基础上准备详细的项目范围说明书，是项目成功的关键。范围定义的输入包括以下内容：

①项目章程。如果项目章程或初始的范围说明书没有在项目执行组织中使用，同样的信息需要进一步收集和开发，以产生详细的项目范围说明书。

②项目范围管理计划。

③组织过程资产。

④批准的变更申请。

所以项目文档管理方案不属于范围定义的输入。

项目时间管理包括使项目按时完成所必需的管理过程，活动定义是其中的一个重要过程。通常可以使用 (23) 来进行活动定义。

(23)A. 鱼骨图

B. 工作分解结构 (WBS)

C. 层次分解结构

D. 功能分解

【答案】B

【解析】本题主要考查时间管理

项目时间管理包括使项目按时完成所必需的管理过程。项目时间管理中的过程包括：活动定义、活动排序、活动的资源估算、活动历时估算、制定进度计划以及进度控制。

为了得到工作分解结构 (Work Breakdown Structure, WBS) 中最底层的交付物，必须执行一系列的活动。对这些活动的识别以及归档的过程就是活动定义。

鱼骨图 (也称为 Ishikawa 图) 是一种发现问题“根本原因”的方法，通常用来进行因果分析。

在实际的项目开发中，人们总是希望使用自动工具来执行需求变更控制过程。下列描述中， (24) 不是这类工具所具有的功能。

(24)A. 可以定义变更请求的数据项以及变更请求生存期的状态转换图

B. 记录每一种状态变更的数据，确认做出变更的人员

C. 可以加强状态转换图使经授权的用户仅能做出所允许的状态变更

D. 定义变更控制计划，并指导设计人员按照所制定的计划实施变更

【答案】D

【解析】本题主要考查范围管理

对许多项目来说，系统软件总需要不断完善，一些需求的改进是合理的而且不可避免，要使得软件需求完全不变更，也许是不可能的，但毫无控制的变更是项目陷入混乱、不能按进度完成或者软件质量无法保证的主要原因之一。

一个好的变更控制过程，给项目风险承担者提供了正式的建议需求变更机制。可以通过需求变更控制过程来跟踪已建议变更的状态，使已建议的变更确保不会丢失或疏忽。在实际中，人们总是希望使用自动工具来执行变更控制过程。有许多人使用商业问题跟踪工具来收集、存储、管理需求变更；可以使用工具对一系列最近提交的变更建议产生一个列表给变更控制委员会开会时做议程用。问题跟踪工具也可以随时按变更状态分类包裹变更请求的数目。

挑选工具时可以考虑以下几个方面：

- ①可以定义变更请求的数据项。
- ②可以定义变更请求生存期的状态转换图。
- ③可以加强状态转换图，使经授权的用户仅能做出所允许的状态变更。
- ④记录每一种状态变更的数据，确认做出变更的人员。
- ⑤可以定义在提交新请求或请求状态被更新后应该自动通知的设计人员。
- ⑥可以根据需要生成标准的或定制的报告和图表。

需求管理是 CMM 可重复级中的 6 个关键过程域之一，其主要目标是 (25)。

(25)A. 对于软件需求，必须建立基线以进行控制，软件计划、产品和活动必须与软件需求保持一致

- B. 客观地验证需求管理活动符合规定的标准、程序和要求
- C. 策划软件需求管理的活动，识别和控制已获取的软件需求
- D. 跟踪软件需求管理的过程、实际结果和执行情况

【答案】A

【解析】

过程能力成熟度模型 (Capability Maturity Model, CMM) 在软件开发机构中被广泛用来指导软件过程改进。该模型描述了软件过程能力的 5 个成熟度级别，每一级都包含若干关键过程域 (Key Process Areas, KPA)»

CMM 的第二级为可重复级，它包括了 6 个关键过程域，分别是：需求管理、软件项目计划、软件项目跟踪和监督、软件分包合同管理、软件质量保证和软件配置管理。

需求管理的目标是为软件需求建立一个基线，提供给软件工程和管理使用：软件计划、产品和活动与软件需求保持一致。

在 RUP 中采用“4+1”视图模型来描述软件系统的体系结构。在该模型中，最终用户侧重于 (26)，系统工程师侧重于 (27)。

- | | | | |
|--------------|---------|---------|---------|
| (26) A. 实现视图 | B. 进程视图 | C. 逻辑视图 | D. 部署视图 |
| (27) A. 实现视图 | B. 进程视图 | C. 逻辑视图 | D. 部署视图 |

【答案】C D

【解析】

在 RUP 中采用“4+1”视图模型来描述软件系统的体系结构。“4+1”视图包括逻辑视图、实现视图、进程视图、部署视图和用例视图。

分析人员和测试人员关心的是系统的行为，因此会侧重于用例视图；最终用户关心的是系统的功能，因此会侧重于逻辑视图；程序员关心的是系统的配置、装配等问题，因此会侧重于实现视图；系统集成人员关心的是系统的性能、可伸缩性、吞吐率等问题，因此会侧重于进程视图；系统工程师关心的是系统的发布、安装、拓扑结构等问题，因此会侧重于部署视图。

(28)把整个软件开发流程分成多个阶段，每一个阶段都由目标设定、风险分析、开发和有效性验证以及评审构成。

- | | | | |
|--------------|---------|---------|---------|
| (28) A. 原型模型 | B. 瀑布模型 | C. 螺旋模型 | D. V 模型 |
|--------------|---------|---------|---------|

【答案】C

【解析】

原型模型又称快速原型。原型模型主要有两个阶段：①原型开发阶段。软件开发人员根据用户提出的软件系统的定义，快速地开发一个原型。该原型应该包含目标系统的关键问题和反映目标系统的大致面貌，展示目标系统的全部或部分功能、性能等。②目标软件开发阶段。在征求用户对原型的意见后对原型进行修改完善，确认软件系统的需求并达到一致的理解，进一步开发实际系统。

瀑布模型可以说是最早使用的软件生存周期模型之一。由于这个模型描述了软件生存的

一些基本过程活动，所以它被称为软件生存周期模型。这些活动从一个阶段到另一个阶段逐次下降，形式上很像瀑布。瀑布模型的特点是因果关系紧密相连，前一个阶段工作的结果是后一个阶段工作的输入。

螺旋模型是在快速原型的基础上扩展而成的。这个模型把整个软件开发流程分成多个阶段，每个阶段都由4部分组成，它们是：

①目标设定。为该项目进行需求分析，定义和确定这一个阶段的专门目标，指定对过程和产品的约束，并且制定详细的管理计划。

②风险分析。对可选方案进行风险识别和详细分析，制定解决办法，采取有效的措施避免这些风险。

③开发和有效性验证。风险评估后，可以为系统选择开发模型，并且进行原型开发，即开发软件产品。

④评审。对项目进行评审，以确定是否需要进入螺旋线的下一次回路，如果决定继续，就要制定下一阶段计划。

V模型是一种典型的测试模型。在V模型中测试过程被加在开发过程的后半部分，分别包括单元测试、集成测试、系统测试和验收测试。

软件开发环境是支持软件产品开发的软件系统，它由软件工具集和环境集成机制构成。环境集成机制包括：提供统一的数据模式和数据接口规范的数据集成机制；支持各开发活动之间通信、切换、调度和协同的(29)；为统一操作方式提供支持的(30)。

(29)A. 操作集成机制

B. 控制集成机制

C. 平台集成机制

D. 界面集成机制

(30)A. 操作集成机制

B. 控制集成机制

C. 平台集成机制

D. 界面集成机制

【答案】B D

【解析】

软件开发环境（Software Development Environment）是支持软件产品开发的软件系统。它由软件工具集和环境集成机制构成，前者用来支持软件开发的相关过程、活动和任务年；后者为工具集成和软件开发、维护和管理提供统一的支持，它通常包括数据集成、控制集成和界面集成。数据集成机制提供了存储或访问环境信息库的统一的数据接口规范；界面集成机制采用统一的界面形式，提供统一的操作方式；控制集成机制支持各开发活动之间的通信、

切换、调度和协同工作。

软件的横向重用是指重用不同应用领域中的软件元素。(31)是一种典型的、原始的横向重用机制。

- (31) A. 对象 B. 构件 C. 标准函数库 D. 设计模式

【答案】C

【解析】

软件重用是指在两次或多次不同的软件开发过程中重复使用相同或相似软件元素的过程。按照重用活动是否跨越相似性较少的多个应用领域，软件重用可以区分为横向重用和纵向重用。横向重用是指重用不同应用领域中的软件元素，例如数据结构、分类算法和人机界面构建等。标准函数是一种典型的、原始的横向重用机制。纵向重用是指在一类具有较多公共性的应用领域之间进行软部件重用。纵向重用活动的主要关键点是域分析：根据应用领域的特征及相似性预测软部件的可重用性。

下列关于不同软件开发方法所使用的模型的描述中，正确的是(32)。

- (32) A. 在进行结构化分析时，必须使用数据流图和软件结构图这两种模型
B. 采用面向对象开发方法时，可以使用状态图和活动图对系统的动态行为进行建模
C. 实体联系图（E-R 图）是在数据库逻辑结构设计时才开始创建的模型
D. UML 的活动图与程序流程图的表达能力等价

【答案】B

【解析】

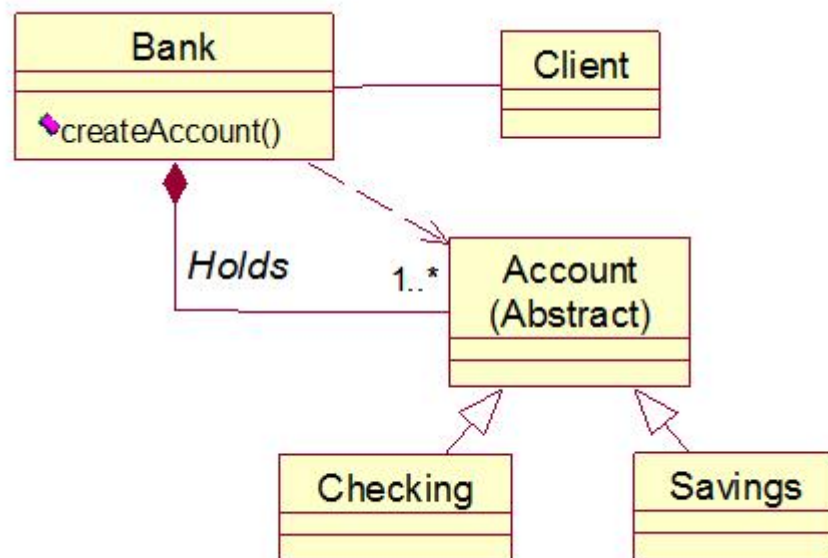
结构化分析方法是一种面向数据流的需求分析方法，其基本思想是自顶向下逐层分解。数据流图是进行结构化分析时所使用的模型，其基本成分包括数据流、加工、数据 存储和外部实体。在进行结构化设计时，通过对数据流图进行变换分析和事务分析可以导出程序结构图。

数据库设计可以分为 4 个主要阶段：①用户需求分析。数据库设计人员采用一定的辅助工具对应用对象的功能、性能、限制等要求所进行的科学分析。②概念设计。概念结构设计是对信息分析和定义，如视图模型化、视图分析和汇总。对应用对象精确地抽象、概括而形成的独立于计算机系统的企业信息模型。描述概念模型的较理想的工具是 E-R 图。③逻辑设计。将抽象的概念模型转化为与选用的 DBMS 产品所支持的数据模型相符合的逻辑模型，它

是物理设计的基础。包括模式初始设计、子模式设计、应用程序设计、模式评价以及模式求精。④物理设计。逻辑模型在计算机中的具体实现方案。

UML 是面向对象软件的标准化建模语言，其中状态图、活动图、顺序图和通信图可以用来对系统的动态行为进行建模。活动图展现了在系统内从一个活动到另一个活动的流程。活动图强调对象之间的控制流程。在活动图上可以表示分支和汇合。活动图与传统的程序流程图是不等价的。

某银行系统采用 Factory Method 方法描述其不同账户之间的关系，设计出的类图如下所示。其中与 Factory Method 中的“Creator”角色相对应的类是_(33)；与“Product”角色相对应的类是_(34)。



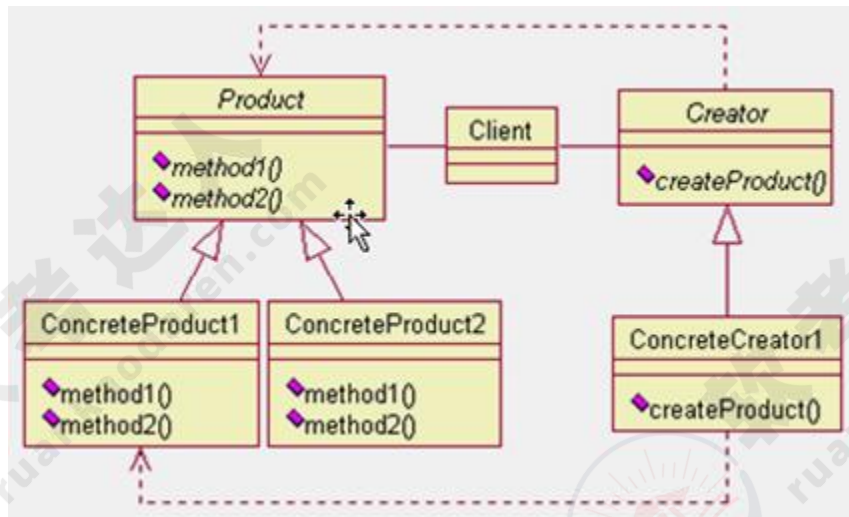
(33) A. Bank B. Account C. Checking D. Savings

(34) A. Bank B. Account C. Checking D. Savings

【答案】A B

【解析】

Factory Method 模式的意图是，定义一个用于创建对象的接口，让子类决定实例化哪一个类。Factory Method 是一个类的实例化延迟到其子类。Factory Method 模式的类图如下图所示。



其中：

- 类 Product 定义了 Factory Method 所创建的对象接口。
- 类 ConcreteProduct 用于实现 Product 接口。
- 类 Creator 声明了工厂方法，该方法返回一个 Product 类型的对象。Creator 也可以定义一个工厂方法的缺省实现，它返回一个缺省的 ConcreteProduct 对象。
- 类 ConcreteCreator 重定义了工厂方法，以返回一个 ConcreteProduct 实例。

对照两张类图可以看出，与“Creator”角色相对应的类是 Bank;与“Product”角色相对应的类是 Account。

(35)是一个独立可交付的功能单元，外界通过接口访问其提供的服务。

- (35)A. 面向对象系统中的对象 (Object)
- B. 模块化程序设计中的子程序 (Subroutine)
- C. 基于构件开发中的构件 (Component)
- D. 系统模型中的包 (Package)

【答案】C

【解析】

在基于构件的开发中，构件包含并扩展了模块化程序设计中子程序、面向对象系统中对象或类和系统模型中包的思想，它是系统设计、实现和维护的基础。构件定义为通过接口访问服务的一个独立可交付的功能单元。

在基于构件的软件开发中，(36)用来了解系统的性能，(37)描述系统设计蓝图以保证系

统提供适当的功能：吞吐率等非功能性属性。

(36) A. 逻辑构件模型 B. 物理构件模型 C. 组件接口模型 D. 系统交互模型

(37) A. 逻辑构件模型 B. 物理构件模型 C. 组件接口模型 D. 系统交互模型

【答案】A B

【解析】

在基于构件的软件开发中，逻辑构件模型用功能包描述系统的抽象设计，用接口描述每个服务集合，以及功能之间如何交互以满足用户需求，它作为系统的设计蓝图以保证系统提供适当的功能。物理构件模型用技术设施产品、硬件分布和拓扑结构，以及用于绑定的网络和通信协议描述系统的物理设计，这种架构用于了解系统的性能、吞吐率等许多非功能性属性。

对象管理组织（OMG）基于 CORBA 基础设施定义了 4 种构件标准。其中 (38) 的状态信息是由构件自身而不是由容器维护。

(38) A. 实体构件 B. 加工构件 C. 服务构件 D. 会话构件

【答案】D

【解析】

对象管理组织（OMG）基于 CORBA 基础设施定义了 4 种构件标准。实体（Entity）构件需要长期持久化并主要用于事务性行为，由容器管理其持久化。加工（Process）构件同样需要容器管理其持久化，但没有客户端可访问的主键。会话（Session）构件不需要容器管理其持久化，其状态信息必须由构件自己管理。服务（Service）构件是无状态的。

分布式系统开发中，通常需要将任务分配到不同的逻辑计算层。业务数据的综合计算分析任务属于 (39)。

(39) A. 表示逻辑层 B. 应用逻辑层 C. 数据处理层 D. 数据层

【答案】B

【解析】

分布式系统开发分为 5 个逻辑计算层：表示层实现用户界面；表示逻辑层包括为了生成数据表示而必须进行的处理任务，如输入数据编辑等；应用逻辑层包括为支持实际业务应用和规则所需的应用逻辑和处理过程，如信用检查、数据计算和分析等；数据处理层包括存储和访问数据库中的数据所需的应用逻辑和命令，如查询语句和存储过程等；数据层是数据库

中实际存储的业务数据。

在客户机/服务器系统开发中，采用(40)时，应将数据层和数据处理层放置于服务器，应用逻辑层、表示逻辑层和表示层放置于客户机。

- (40) A. 分布式表示结构 B. 分布式应用结构
C. 分布式数据和应用结构 D. 分布式数据结构

【答案】D

【解析】

客户机/服务器系统开发时可以采用不同的分布式计算架构：分布式表示架构是将表示层和表示逻辑层迁移到客户机，应用逻辑层、数据处理层和数据层仍保留在服务器上；分布式数据架构是将数据层和数据处理层放置于服务器，应用逻辑层、表示逻辑层和表示层放置于客户机；分布式数据和应用架构是将数据层和数据处理层放置在数据服务器上，应用逻辑层放置于应用服务器上，表示逻辑层和表示层放置于客户机上。

系统输入设计中,采用内部控制方式以确保输入系统数据的有效性,(41)用于验证数据是否位于合法的取值范围。

- (41) A. 数据类型检查 B. 自检位 C. 域检查 D. 格式检查

【答案】 C

【解析】

系统输入设计中，通常通过内部控制的方式验证输入数据的有效性。数据类型检查确保输入了正确的数据类型；自检位用于对主关键字进行基于校验位的检查；域检查用于验证数据是否位于合法的取值范围；格式检查按照已知的数据格式对照检查输入数据的格式。

系统测试由若干个不同的测试类型组成，其中（42）检查系统能力的最高实际限度，即软件在一些超负荷情况下的运行情况；（43）主要是检查系统的容错能力。

- (42) A. 强度测试 B. 性能测试 C. 恢复测试 D. 可靠性测试
- (43) A. 强度测试 B. 性能测试 C. 恢复测试 D. 可靠性测试

【答案】A C

【解析】

系统测试是将已经确认的软件、计算机硬件、外设和网络等其他因素结合在一起,进行

信息系统的各种组装测试和确认测试，其目的是通过与系统的需求相比较，发现开发的系统与用户需求不符或矛盾的地方。系统测试是根据系统方案说明书来设计测试例子的，常见的系统测试主要有以下内容：

(1) 恢复测试：恢复测试监测系统的容错能力。检测方法是采用各种方法让系统出现故障，检验系统是否按照要求能从故障中恢复过来，并在约定的时间内开始事务处理，而且不对系统造成任何伤害。如果系统的恢复是自动的（由系统自动完成），需要验证重新初始化、检查点、数据恢复等是否正确。如果恢复需要人工干预，就要对恢复的平均时间进行评估并判断它是否在允许的范围内。

(2) 安全性测试：系统的安全性测试是检测系统的安全机制、保密措施是否完善，主要是为了检验系统的防范能力。测试的方法是测试人员模拟非法入侵者，采用各种方法冲破防线。系统安全性设计准则是使非法入侵者所花费的代价比进入系统后所得到的好处要大，此时非法入侵已无利可图。

(3) 强度测试：是对系统在异常情况下的承受能力的测试，是检查系统在极限状态下运行时，性能下降的幅度是否在允许的范围内。因此，强度测试要求系统在非正常数量、频率或容量的情况下运行。强度测试主要是为了发现在有效的输入数据中可能引起不稳定或不正确的数据组合。例如，运行使系统处理超过设计能力的最大允许值的测试例子；使系统传输超过设计最大能力的数据，包括内存的写入和读出等。

(4) 性能测试：检查系统是否满足系统设计方案说明书对性能的要求。性能测试覆盖了软件测试的各阶段，而不是等到系统的各部分都组装之后，才确定系统的真正性能。通常与强度测试结合起来进行，并同时软件、硬件进行测试。软件方面主要从响应时间、处理速度、吞吐量、处理精度等方面来检测。

(5) 可靠性测试：通常使用以下两个指标来衡量系统的可靠性：平均失效间隔时间 MTBF (mean time between failures) 是否超过了规定的时限，因故障而停机时间 MTTR (mean time to repairs) 在一年中不应超过多少时间。

(6) 安装测试：在安装软件系统时，会有多种选择。安装测试就是为了检测在安装过程中是否有误、是否容易操作等。主要监测系统的每一个部分是否齐全，硬件的配置是否合理，安装中需要产生的文件和数据库是否已产生，其内容是否正确等。

软件架构是降低成本、改进质量、按时和按需交付产品的关键因素。以下关于软件架构的描述，错误的是 (44)。

- (44)A. 根据用户需求，能够确定一个最佳的软件架构，指导整个软件的开发过程
- B. 软件架构设计需要满足系统的质量属性，如性能、安全性和可修改性等
- C. 软件架构设计需要确定组件之间的依赖关系，支持项目计划和管理活动
- D. 软件架构能够指导设计人员和实现人员的工作

【答案】A

【解析】

软件架构是降低成本、改进质量、按时和按需交付产品的关键因素，软件架构设计需要满足系统的质量属性，如性能、安全性和可修改性等，软件架构设计需要确定组件之间的依赖关系，支持项目计划和管理活动，软件架构能够指导设计人员和实现人员的工作。一般在设计软件架构之初，会根据用户需求，确定多个候选架构，从中选择一个较优的架构，并随着软件的开发，对这个架构进行微调，以达到最佳效果。

软件架构设计包括提出架构模型、产生架构设计和进行设计评审等活动，是一个迭代的过程。以下关于软件架构设计活动的描述，错误的是(45)。

- (45)A. 在建立软件架构的初期，一般需要选择一个合适的架构风格
- B. 将架构分析阶段已标识的构件映射到架构中，并分析这些构件之间的关系
- C. 软件架构设计活动将已标识构件集成到软件架构中，设计并实现这些构件
- D. 一旦得到了详细的软件架构设计，需要邀请独立于系统开发的外部人员对系统进行评审

【答案】C

【解析】

软件架构设计包括提出架构模型、产生架构设计和进行设计评审等活动，是一个迭代的过程，在建立软件架构的初期，一般需要选择一个合适的架构风格，将架构分析阶段已标识的构件映射到架构中，并分析这些构件之间的关系，一旦得到了详细的软件架构设计，需要邀请独立于系统开发的外部人员对系统进行评审。一般来说，软件架构设计活动将已标识构件集成到软件架构中，设计这些构件，但不予以实现。

基于软件架构的设计（Architecture Based Software Development，ABSD）强调由商业、质量和功能需求的组合驱动软件架构设计。它强调采用(46)来描述软件架构，采用(47)来描述需求。

- (46) A. 类图和序列图 B. 视角与视图 C. 构件和类图 D. 构件与功能
- (47) A. 用例与类图 B. 用例与视角
- C. 用例与质量场景 D. 视角与质量场景

【答案】B C

【解析】

根据基于软件架构的设计的定义，基于软件架构的设计（Architecture Based Software Development, ABSD）强调由商业、质量和功能需求的组合驱动软件架构设计。它强调采用视角和视图来描述软件架构，采用用例和质量属性场景来描述需求。

某游戏公司欲开发一个大型多人即时战略游戏，游戏设计的目标之一是能够支持玩家自行创建战役地图，定义游戏对象的行为和对象之间的关系。针对该目标，公司应该采用（48）架构风格最为合适。

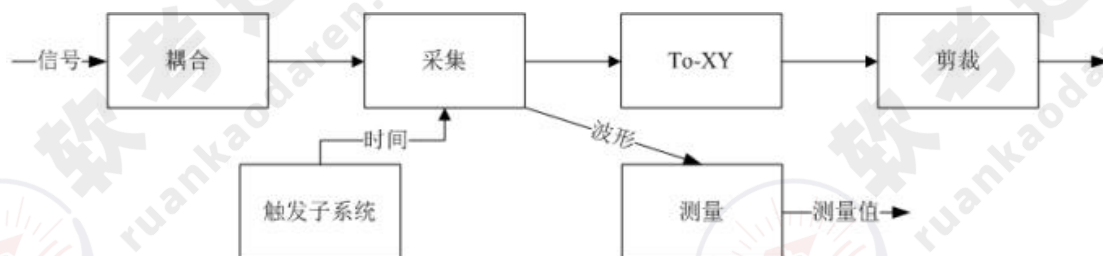
- (48) A. 管道-过滤器 B. 隐式调用 C. 主程序-子程序 D. 解释器

【答案】D

【解析】本题主要考查软件架构设计策略与架构风格问题。

根据题干描述，该软件系统特别强调用户定义系统中对象的关系和行为这一特性，这需要在软件架构层面提供一种运行时的系统行为定义与改变的能力，根据常见架构风格的特点和适用环境，可以知道最合适的架构设计风格应该是解释器风格。

某公司欲为某种型号的示波器开发内置软件。该公司的架构师设计了如下图所示的软件架构。在软件架构评审时，专家认为该架构存在的问题是（49）。



- (49) A. 在功能划分上将各个模块独立起来
- B. 在硬件构件的混合和替换方面不是很灵活
- C. 没有清晰地说明用户怎样与其交互

D. 没有明确的层次关系，没有强调功能之间的交互试

【答案】C

【解析】本题主要考查架构评审和软件架构设计的应用。

根据图中示波器的功能描述，结合示波器常见的功能和使用方式，可以看出图中的系统设计最大的缺陷在于没有建模系统与外界，特别是用户之间的交互方式。而与用户的交互无疑是示波器的一个十分重要的功能。

某公司承接了一个开发家用空调自动调温器的任务，调温器测量外部空气温度，根据设定的期望温度控制空调的开关。根据该需求，公司应采用_(50)_架构风格最为合适。

(50)A. 解释器 B. 过程控制 C. 分层 D. 管道-过滤器

【答案】B

【解析】本题主要考查架构风格与架构设计策略。

根据题目描述，调温器需要实时获取外界的温度信息，与用户定义的温度进行比较并做出动作。根据该系统的应用领域和实际需求，可以看出这是一个典型的过程控制架构风格的应用场景。

某公司欲开发一个漫步者机器人，用来完成火星探测任务。机器人的控制者首先定义探测任务和任务之间的时序依赖性，机器人接受任务后，需要根据自身状态和外界环境进行动态调整，最终自动完成任务。针对这些需求，该机器人应该采用_(51)_架构风格最为合适。

(51)A. 解释器 B. 主程序·子程序 C. 隐式调用 D. 管道-过滤器

【答案】C

【解析】本题主要考查架构风格与架构设计策略。

根据题目描述，漫步者机器人需要根据自身状态和外界环境进行自动调整，这是一个典型的根据外部事件进行响应的场景。比较 4 个候选项，隐式调用比较适合根据外部事件进行处理和动作的情景。但本题出题本就不严谨，从描述来看多种架构风格均合适：过程控制，虚拟机，隐式调用。当次考试参考答案为 C，但从此后的同类问题来看，答案修改为“虚拟机（解释器，规则系统）”，所以再次出现该类问题，建议首选虚拟机类风格，选 A。

某公司欲开发一个语音识别系统，语音识别的主要过程包括分割原始语音信号、识别音素、产生候选词、判定语法片断、提供语义解释等。每个过程都需要进行基于先验知识的

条件判断并进行相应的识别动作。针对该系统的特点，采用 (52) 架构风格最为合适。

- (52) A. 解释器 B. 面向对象 C. 黑板 D. 隐式调用

【答案】C

【解析】本题主要考查架构风格与架构设计策略。

根据题目描述，语音识别系统是一个十分典型的专家系统，其特点是求解的正确结果不止一个，求解过程比较复杂，需要通过专家知识和反馈逐步得到正确结果。因此对比 4 个选项，黑板结构特别适合求解这类问题。

特定领域软件架构 (DSSA) 是在一个特定应用领域为一组应用提供组织结构参考的标准软件架构。实施 DSSA 的过程中包括一系列基本的活动，其中 (53) 活动的主要目的是为了获得 DSSA。该活动参加人员中，(54) 的主要任务是提供关于领域中系统的需求规约和实现的知识。

- (53) A. 领域需 B. 领域分析 C. 领域设计 D. 领域实现
(54) A. 领域专家 B. 领域分析者 C. 领域设计者 D. 领域实现者

【答案】C A

【解析】本题主要考查特定领域软件架构的基本定义和基本活动。

特定领域软件架构 (DSSA) 是在一个特定应用领域为一组应用提供组织结构参考的标准软件架构。实施 DSSA 的过程中包括一系列基本的活动，其中领域设计活动的主要目的是为了获得 DSSA。该活动参加人员中，领域专家的主要任务是提供关于领域中系统的需求规约和实现的知识。

某服务器软件系统对可用性 (Availability)、性能 (Performance) 和可修改性 (Modification) 的要求较高，(55) 设计策略能提高该系统的可用性，(56) 设计策略能够提高该系统的性能，(57) 设计策略能够提高该系统的可修改性。

- (55) A. Ping/Echo B. 限制访问 C. 运行时注册 D. 接口-实现分离
(56) A. 分层结构 B. 事务机制 C. 主动冗余 D. 队列调度
(57) A. 信息隐藏 B. 记录/回放 C. 任务模型 D. 回滚

【答案】A D A

【解析】本题主要考查质量属性以及实现质量属性的一般策略。

不同策略主要针对一个或多个软件质量属性，其中 Ping/Echo 主要提高系统的可用性；

限制访问主要提高系统的安全性；运行时注册主要提高系统的可修改性；接口-实现分离主要提高系统的可修改性；主动冗余提高系统的可靠性；队列调度主要提高系统的性能；信息隐藏主要提高系统的可修改性；记录-回放主要提高系统的可测试性，等等。

某服务器软件系统能够正确运行并得出计算结果，但存在“系统出错后不能在要求的时间内恢复到正常状态”和“对系统进行二次开发时总要超过半年的时间”两个问题，上述问题依次与质量属性中的(58)相关。

- (58) A. 可用性和性能 B. 性能和可修改性
C. 性能和可测试性 D. 可用性和可修改性

【答案】D

【解析】本题主要考查软件质量属性的判断与应用。

“系统出错后不能在要求的时间内恢复到正常状态”，这是对系统错误恢复能力的描述，属于系统可用性的范畴。“对系统进行二次开发时总要超过半年的时间”，这是对系统进行调整和维护方面能力的描述，属于系统可修改性的范畴。

某公司欲开发一个软件系统的在线文档帮助系统，用户可以在任何一个查询上下文中输入查询关键字，如果当前查询环境下没有相关内容，则系统会将查询按照一定的顺序转发给其他查询环境。基于上述需求，采用(59)最为合适。

- (59) A. 责任链模式 B. 桥接模式 C. 装饰模式 D. 适配器模式

【答案】A

【解析】本题主要考查设计模式的理解与应用。

根据题干描述，在线文档系统需要根据用户的查询需求逐步将查询请求依次传递，对比4个候选项，其中在责任链模式中，很多对象由每一个对象对其下家的引用而连接起来形成一条链。请求在这个链上传递，直到链上的某一个对象决定处理此请求。因此责任链模式是能够满足该要求的最好模式。

某公司欲开发一套窗体图形界面类库。该类库需要包含若干预定义的窗格(Pane)对象，例如TextPane、ListPane等，窗格之间不允许直接引用。基于该类库的应用由一个包含一组窗格的窗口组成，并需要协调窗格之间的行为。基于该类库，在不引用窗格的前提下实现窗格之间的协作，应用开发者应采用(60)最为合适。

(60) A. 备忘录模式 B. 中介者模式 C. 访问者模式 D. 迭代器模式

【答案】B

【解析】本题主要考查设计模式的理解与应用。

根据题干描述，应用系统需要使用某公司开发的类库，该应用系统由一组窗格组成，应用需要协调窗格之间的行为，并且不能引用窗格自身，在这种要求下，对比4个候选项，其中中介者模式用一个中介对象封装一系列的对象交互。中介者使用的各对象不需要显式的相互调用，从而使其耦合松散。可以看出该模式最符合需求。

某公司开发一个文档编辑器，该编辑器允许在文档中直接嵌入图形对象，但开销很大。用户在系统设计之初提出编辑器在打开文档时必须十分迅速，可以暂时不显示当前页面以外的图形。针对这种需求，公司可以采用(61)避免同时创建这些图形对象。

(61) A. 代理模式 B. 外观模式 C. 桥接模式 D. 组合模式

【答案】A

【解析】本题主要考查设计模式的理解与应用。

根据题干描述，该编辑器需要在文档中高系统效率，需要避免同时创建这些图像。针对这些要求，对比候选项，可以发现代理模式可以解决直接访问对象时带来的问题，例如，要访问的对象在远程的机器上；对象创建开销很大，或者某些操作需要安全控制，或者需要进程外的访问等。因此代理模式是最为合适的设计模式。

正确识别风险点、非风险点、敏感点和权衡点是进行软件架构评价的关键步骤。其中(62)是实现一个特定质量属性的关键特征，该特征为一个或多个软件构件所共有。“改变加密的级别可能会对安全性和性能都产生显著的影响”，这是一个对系统(63)的描述。

(62) A. 风险点 B. 非风险点 C. 敏感点 D. 权衡点

(63) A. 风险点 B. 非风险点 C. 敏感点 D. 权衡点

【答案】C D

【解析】本题主要考查软件架构评价的理解和应用。

正确识别风险点、非风险点、敏感点和权衡点是进行软件架构评价的关键步骤。其中敏感点是实现一个特定质量属性的关键特征，该特征为一个或多个软件构件所共有。系统权衡点会影响一个或多个属性，并对于多个属性来说都是敏感点。基于该定义，可以看出“改变加密的级别可能会对安全性和性能都产生显著的影响”正是一个对系统权衡点的描述。

所谓网络安全漏洞是指(64)。

- (64) A. 用户的误操作引起的系统故障
- B. 网络节点的系统软件或应用软件在逻辑设计上的缺陷
- C. 网络硬件性能下降产生的缺陷
- D. 网络协议运行中出现的错误

【答案】B

【解析】本题主要考查网络安全漏洞的基本概念。

网络安全漏洞通常是指网络节点的系统软件或应用软件在逻辑上的缺陷，因此本题应该选择 B。

ARP 攻击造成网络无法跨网段通信的原因是(65)。

- (65) A. 发送大量 ARP 报文造成网络拥塞
- B. 伪造网关 ARP 报文使得数据包无法发送到网关
- C. ARP 攻击破坏了网络的物理连通性
- D. ARP 攻击破坏了网关设备

【答案】B

【解析】本题主要考查 ARP 攻击的定义和特点。

ARP 攻击是针对以太网地址解析协议（ARP）的一种攻击技术，此种攻击可让攻击者取得局域网上的数据封包甚至可篡改封包，且可让网络上特定计算机或所有计算机无法正常连接。ARP 攻击造成网络无法跨网段通信的原因是伪造网关 ARP 报文使得数据包无法发送到网关。

《计算机软件产品开发文件编制指南 GB 8567—88》是(66)标准，违反该标准而造成不良后果时，将依法根据情节轻重受到行政处罚或追究刑事责任。

- (66) A. 强制性国家 B. 推荐性国家 C. 强制性软件行业 D. 推荐性软件行业

【答案】A

【解析】

我国国家标准的代号由大写汉字拼音字母构成，强制性国家标准代号为 GB，推荐性国家标准的代号为 GB/T。

强制性标准是国家技术法规，具有法律约束性。其范围限制在国家安全、防止欺诈行为、

保护人身健康与安全等方面。根据《标准化法》的规定，企业和有关部门对涉及其经营、生产、服务、管理有关的强制性标准都必须严格执行，任何单位和个人不得擅自更改或降低标准。对违反强制性标准而造成不良后果以至重大事故者，由法律、行政法规规定的行政主管部门依法根据情节轻重给予行政处罚，直至由司法机关追究刑事责任。

推荐性标准是自愿采用的标准。这类标准是指导性标准，不具有强制性，一般是为了通用或反复使用的目的，为产品或相关生产方法提供规则、指南或特性的文件。任何单位均有权决定是否采用，违反这类标准，不构成经济或法律方面的责任。由于推荐性标准是协调一致的文件，不受政府和社会团体的利益干预，能更科学地规定特性或指导生产，我国《标准化法》鼓励企业积极采用推荐性标准。应当指出的是，推荐性标准一经接受并采用，或由各方商定后同意纳入经济合同中，就成为各方必须共同遵守的技术依据，具有法律上的约束性。行业标准是由行业机构、学术团体或国防机构制定，并适用于某个业务领域的标准。行业标准代号由国务院各有关行政主管部门提出其所管理的行业标准范围的申请报告，国务院标准化行政主管部门审查确定并正式公布该行业标准代号。已正式公布的行业代号：QJ（航天）、SJ（电子）、JB（机械）、JR（金融）等，暂无软件行业。行业标准代号由汉字拼音大写字母组成，再加上斜线 T 组成推荐性行业标准（如 SJ/T）。

某软件企业开发了一套能够同硬件结合以提高设备性能的软件产品，向国家专利局申请方法发明专利，获得了专利权，并为该软件产品冠以“昆仑”商品专用标识，但未进行商标注册就上市销售。此情况下，该软件产品不可能得到我国(67)的保护。

- (67) A. 著作权法 B. 专利法 C. 商标法 D. 刑法

【答案】C

【解析】

该企业在软件开发完成后就取得了著作权，该企业向国家专利局申请方法发明专利并获得了专利权，此时该软件就可以同时受到著作权法、专利法的保护。

在《刑法》中明确规定了严重侵犯知识产权的行为应当承担刑事责任。在《计算机软件保护条例》第二十四条中，规定对侵权行为触犯刑律的，依照刑法关于侵犯著作权罪、销售侵权复制品罪的规定，依法追究刑事责任。例如《刑法》中第二百一十三条规定，未经注册商标所有人许可，在同一种商品上使用与其注册商标相同的商标，情节严重的，处三年以下有期徒刑或者拘役，并处或者单处罚金；情节特别严重的，处三年以上七年以下有期徒刑，并处罚金；第二百一十六条规定对假冒他人专利，情节严重的，处三年以下有期徒刑或者拘役，

并处或者单处罚金：二百一十七条规定对以营利为目的，未经著作权人许可，复制发行其计算机软件的，违法所得数额较大或者有其他严重情节的，处三年以下有期徒刑或者拘役，并处或者单处罚金；违法所得数额巨大或者有其他特别严重情节的，处三年以上七年以下有期徒刑，并处罚金：二百一十八条规定对以营利为目的，销售明知是本法第二百一十七条规定的侵权复制品，违法所得数额巨大的，处三年以下有期徒刑或者拘役，并处或者单处罚金。所以，该软件产品能够得到刑法的保护。

商标权是商标所有人依法对其商标所享有的专有使用权。在我国，商标权的取得实行的是注册原则，即商标所有人只有依法将自己的商标注册后，商标注册人才能取得商标权，其商标才能得到法律的保护。该企业虽然对其软件产品已经冠以商品专用标识，但未进行商标注册，没有取得商标专用权，此时该软件产品就不能得到商标法的保护。

商标法虽然不能提供针对计算机软件的实质内容与表达的直接保护，却可以为软件提供商业化的保护。计算机软件的权利人可以通过商标法来实现对其所属软件的保护。首先，软件的权利人可以针对软件产品申请注册商标，并通过标注在其产品包装上的商标来表明其身份以及商誉；其次，软件的权利人还可以通过技术措施在软件中设置其特有的商业标记，例如，通过技术手段将与注册商标相同的文字商标、图形商标或者其二者的结合出现在软件的界面上。这些措施不仅可以防止他人对商品化了的计算机软件实施侵权，也可以在遭受侵权后，利用商标法有效地打击侵权行为。尤其是当软件被盗版的时候，这些盗版软件经常是不标明商标、产地等，甚至是假冒商标，软件权利人可以依据商标法请求工商行政管理部门进行查处，在实践中，行政手段通常要比诉讼程序更简便、快捷，达到保护计算机软件的目的。当然，商标法还对侵犯软件商标权的行为应承担的民事以及刑事责任进行了规定。

张某是M国际运输有限公司计算机系统管理员。任职期间，根据公司的业务要求开发了“空运出口业务系统”，并由公司使用。随后，张某向国家版权局申请了计算机软件著作权登记，并取得了《计算机软件著作权登记证书》，证书明确软件名称是“空运出口业务系统V1.0”，著作权人为张某。以下说法中，正确的是(68)。

- (68) A. 空运出口业务系统 V1.0 的著作权属于张某
B. 空运出口业务系统 V1.0 的著作权属于 M 公司
C. 空运出口业务系统 V1.0 的著作权属于张某和 M 公司
D. 张某获取的软件著作权登记证是不可以撤销的

【答案】B

【解析】

张某开发的软件是在 M 国际运输有限公司担任计算机系统管理员期间根据国际运输有限公司业务要求开发的“空运出口业务系统 V1.0”，即该软件是针对本职工作中明确指定的开发目标所开发的。根据《著作权法》第十六条规定，公民为完成法人或者非法人单位工作任务所创作的作品是职务作品。认定作品为职务作品还是个人作品，应考虑两个前提条件：一是作者和所在单位存在劳动关系，二是作品的创作属于作者应当履行的职责。职务作品分为一般职务作品和特殊的职务作品：一般职务作品的著作权由作者享有，单位或其他组织享有在其业务范围内优先使用的权利，期限为两年；特殊的职务作品，除署名权以外，著作权的其他权利由单位享有。所谓特殊职务作品是指《著作权法》第十六条第 2 款规定的两种情况：一是主要利用法人或者其他组织的物质技术条件创作，并由法人或者其他组织承担责任的工程设计、产品设计图、计算机软件、地图等科学技术作品；二是法律、法规规定或合同约定著作权由单位享有的职务作品。《计算机软件保护条例》也有类似的规定，在第十三条中规定了三种情况，一是针对本职工作中明确指定的开发目标所开发的软件；二是开发的软件是从事本职工作活动所预见的结果或者自然的结果；三是主要使用了法人或者其他组织的资金、专用设备、未公开的专门信息等物质技术条件所开发并由法人或者其他组织承担责任的软件。张某在公司任职期间利用公司的资金、设备和各种资料开发了软件，且该软件是从事本职工作活动所预见的结果。所以，其进行的软件开发行为是职务行为，其工作成果应由公司享有。因此，该软件的著作权应属于 M 国际运输有限公司，但根据法律规定，张某享有署名权。

根据《计算机软件保护条例》第七条规定，软件登记机构发放的登记证明文件是登记事项的初步证明，只是证明登记主体享有软件著作权以及订立许可合同、转让合同的重要的书面证据，并不是软件著作权产生的依据。该软件是张某针对本职工作中明确指定的开发目标所开发的，该软件的著作权应属于公司。明确真正的著作权人之后，软件著作权登记证书的证明力自然就消失了（只有审判机关才能确定登记证书的有效性）。

为促进我国软件产业发展，增强我国软件产业的创新能力和竞争能力，1992 年 4 月 6 日机械电子部发布了《计算机软件著作权登记办法》，鼓励软件登记并对登记的软件予以重点保护，而不是强制软件登记。软件登记可以分为软件著作权登记、软件著作权专有许可合同和转让合同的登记。软件著作权登记的申请人应当是该软件的著作权人，而软件著作权合同登记的申请人，应当是软件著作权专有许可合同和转让合同的当事人。如果未经软件著作权人许可登记其软件，或是将他人软件作为自己的软件登记的，或未经合作者许可、将与

他人合作开发的软件作为自己单独完成的软件登记，这些行为都属于侵权行为，侵权人要承担法律责任。

对实际应用问题建立数学模型并求得结果后，还需要根据建模的目的和要求，利用相关知识，结合研究对象的特点，进行模型分析。模型分析工作一般不包括 (69)。

- (69) A. 模型的合理性分析 B. 模型的误差分析
C. 模型的先进性分析 D. 参数的灵敏性分析

【答案】C

【解析】本题主要考查数学建模的基本过程。

在对实际应用问题建立数学模型并求得结果后，还需要根据建模的目的和要求，利用相关知识，结合研究对象的特点，进行模型分析。模型分析工作主要包括模型的合理性分析、模型的误差分析和参数的灵敏性分析等，一般不包括模型的先进性分析。

某软件公司开发某种软件产品时花费的固定成本为 16 万元，每套产品的可变成本为 2 元，设销售单价为 10 元，则需要销售 (70) 套才能达到盈亏平衡点。

- (70) A. 13000 B. 16000 C. 18000 D. 20000

【答案】D

【解析】

设共销售了 n 套，则总成本=固定成本+每套产品的可变成本 $X_n=160000+2n$ 。销售总收入=单价 $X_n=10n$ 。盈亏平衡时，总成本=销售总收入，所以， $160000+2n=10n$ ，从而， $n=20000$ (套)。当销售量低于 2 万套时会有亏损；当销售量超过 2 万套时就会有盈利。

The software architecture is a set of software components, subsystems, relationships, interactions, the properties of each of these elements, and the set of guiding principles that together constitute the fundamental properties and constraints of a software system or set of systems. (71) defines a general set of element types and their interactions. The examples include Pipes and Filters, Model-View-Controller, and Reflection, A (72) in software architecture is a representation used to understand or document one or more aspects of a problem or solution. Architecture is usually used in conjunction with many adjunct terms. The

(73) defines the key strategies, organization, goals and related processes of the enterprise. At the enterprise level, the (74) may be more of a set of guidelines on how the various software architectures should be constructed consistently across the enterprise. The (75), which describes the high-level set of elements involved in application from a particular domain along with their interactions, is often used to focus on subsystem definition rather than application process level definition.

- | | |
|-----------------------------------|--------------------------------|
| (71) A. Architectural pattern | B. Architectural description |
| C. Architectural view | D. Architectural viewpoint |
| (72) A. model | B. domain |
| C. component | D. subsystem |
| (73) A. enterprise architecture | B. technical architecture |
| C. infrastructure architecture | D. business architecture |
| (74) A. enterprise architecture | B. data architecture |
| C. application architecture | D. information architect |
| (75) A. product-line architecture | B. reference architecture |
| C. technology architecture | D. infrastructure architecture |

【答案】A A D C B

【解析】

软件架构是一组软件构件、子系统、关联关系、交互关系以及其中每个元素的特性和一组指导原则，这些共同构成一个软件系统或一组系统的基本特性和约束。架构模式定义了一组通用的元素类型及其交互关系。其中的例子包括管道-过滤器、MVC 和 Reflection。软件架构中的模型是一种表现形式，用于从一个或多个方面理解或记录一个问题或解决方案。架构通常会与一些附属词结合起来使用。业务架构定义了企业的关键策略、组织、目标和相关过程。在企业层面，应用架构可能更多的是一组关于应该如何在整个企业一致地构建各种软件架构的指导原则。参考架构描述了源自特定领域的应用涉及的高层元素集合及其交互关系，通常用于关注子系统的定义而不是应用过程级别定义。

试题一

阅读以下关于软件架构设计的叙述，在答题纸上回答问题 1 至问题 3。

【说明】

某公司欲针对 Linux 操作系统开发一个 KWIC (KeyWord in Context)检索系统。该系统接收用户输入的查询关键字，依据字母顺序给出相关帮助文档并根据帮助内容进行循环滚动阅读。在对 KWIC 系统进行需求分析时，公司的业务专家发现用户后续还有可能采用其他方式展示帮助内容。根据目前需求，公司的技术人员决定通过重复剪切帮助文档中的第一个单词并将其插入到行尾的方式实现帮助文档内容的循环滚动，后续还将采用其他的方法实现这一功能。

在对 KWIC 系统的架构进行设计时，公司的架构师王工提出采用共享数据的主程序-子程序的架构风格，而李工则主张采用管道-过滤器的架构风格。在架构评估会议上，大家从系统的算法变更、功能变更、数据表示变更和性能等方面对这两种方案进行评价，最终采用了李工的方案。

【问题 1】(7 分)

在实际的软件项目开发中，采用恰当的架构风格是项目成功的保证。请用 200 字以内的文字说明什么是软件架构风格，并对主程序-子程序和管道-过滤器这两种架构风格的特点进行描述。

【答案】

软件架构风格是描述特定软件系统组织方式的惯用模式。组织方式描述了系统的组成构件和这些构件的组织方式，惯用模式则反映众多系统共有的结构和语义。

主程序-子程序架构风格中，所有的计算构件作为子程序协作工作，并由一个主程序顺序地调用这些子程序，构件通过共享存储区交换数据。

管道-过滤器架构风格中，每个构件都有一组输入和输出，构件接受数据输入，经过内部处理，然后产生数据输出。这里的构件称为过滤器，构件之间的连接件称为数据流传输的管道。

评价要素 \ 架构风格	共享数据的主程序-子程序	管道-过滤器
算法变更	-	(1) +
功能变更	(2) -	+
数据表示变更	(3) -	(4) -
性能	(5) +	(6) -

【解析】

本题主要考查软件架构风格的定义，并考查对两种与本题相关的架构风格的定义。软件架构风格是描述特定软件系统组织方式的惯用模式。组织方式描述了系统的组成构件和这些构件的组织方式，惯用模式则反映众多系统共有的结构和语义。

主程序-子程序架构风格中，所有的计算构件作为子程序协作工作，并由一个主程序顺序地调用这些子程序，构件通过共享存储区交换数据。

管道-过滤器架构风格中，每个构件都有一组输入和输出，构件接受数据输入，经过内部处理，然后产生数据输出。这里的构件称为过滤器，构件之间的连接件称为数据流传输的管道。

【问题2】(12分)

请完成表 1-1 中的空白部分（用+表示优、-表示差），对王工和李工提出的架构风格进行评价，并指出采用李工方案的原因。

表 1-1 王工与李工的架构风格评价

评价要素 \ 架构风格	共享数据的主程序-子程序	管道-过滤器
算法变更	-	(1)
功能变更	(2)	+
数据表示变更	(3)	(4)
性能	(5)	(6)

【答案】

根据题干描述：“用户后续还有可能采用其他方式展示帮助内容”，因此 KWIC 系统对功

能变更要求较高；

根据题干描述：“后续还将采用其他的方法实现这一功能”，因此 KWIC 系统对实现某一个功能的算法变更要求较高。

KWIC 是一个支持用户交互的窗口界面程序，因此对性能要求并不高。

KWIC 系统的显示帮助内容为文本，数据的表示基本不变，因此对数据表示变更要求不高。

综合上述分析，可以看出应该采用李工提出的管道-过滤器架构风格。

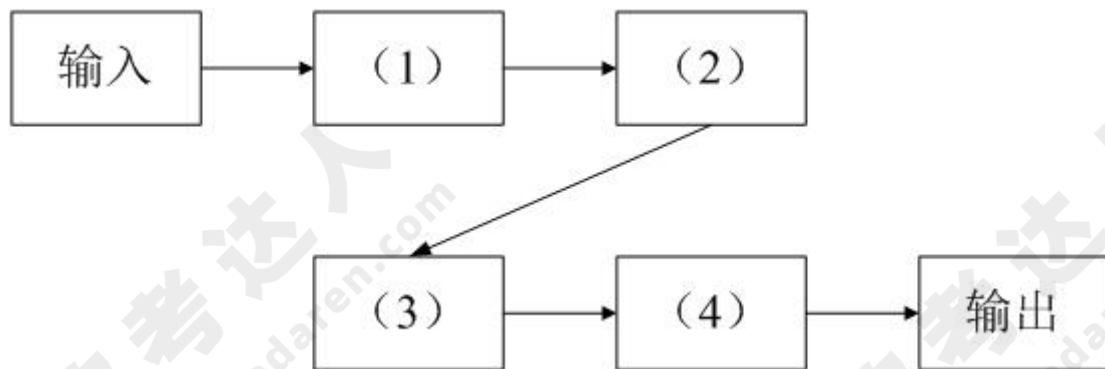
【解析】

本问题是一道填表题，其核心是对两种架构风格在算法变更、功能变更、数据表示变更和性能等方面的特点进行比较。其中共享数据的主程序-子程序在算法变更方面灵活性较差，算法变更时一般需要重新编译整个系统；在功能变更方面也比较差；在数据表示方面，当需要变更时，也意味着程序传递参数的变化以及整个程序的调整，表现比较差；在性能方面，由于整个程序处在一个紧耦合的状态，因此性能较高。管道-过滤器架构风格在算法变更方面实现比较简单，只需要修改过滤器的实现即可；在功能变更方面也比较简单；在数据表示方面，需要同时改变数据格式和过滤器的结构，相对比较复杂；在性能方面，由于整个系统是松耦合连接在一起的，因此性能不高。

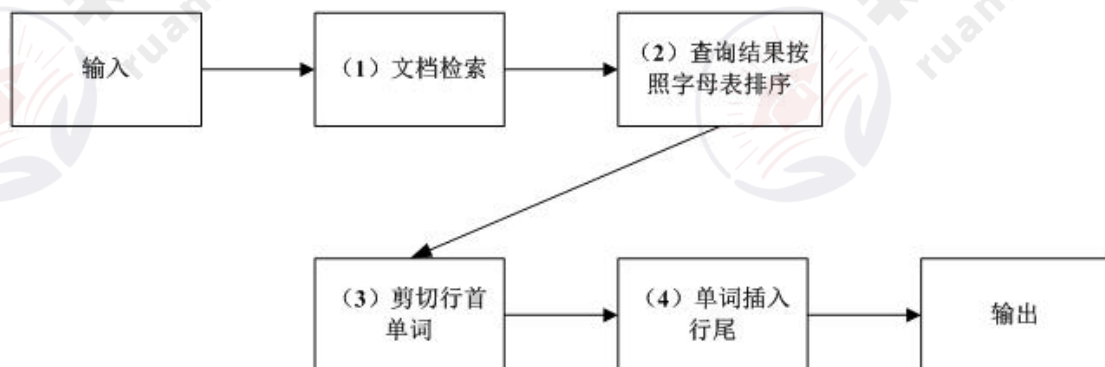
根据题干描述：“用户后续还有可能采用其它方式展示帮助内容”，因此 KWIC 系统对功能变更要求较高；根据题干描述：“…，后续还将采用其它的方法实现这一功能”，因此 KWIC 系统对实现某一个功能的算法变更要求较高；KWIC 是一个支持用户交互的窗口界面程序，因此对性能要求并不高；KWIC 系统的显示帮助内容为文本，数据的表示基本不变，因此对数据表示变更要求不高。针对这些考虑，可以看出应该采用管道-过滤器的架构风格。

【问题3】（6分）

图 1-1 是李工给出的架构设计示意图，请将恰当的功能描述填入图中的（1）～（4）。



【答案】



【解析】

本问题是一道读图题，考查用户对系统架构的理解程度。根据题干中的关键描述“…接收用户输入的查询关键字，依据字母顺序给出相关帮助文档并根据帮助内容进行循环滚动阅读…”和“…公司的技术人员决定通过重复剪切帮助文档中的第一个单词并将其插入到行尾的方式实现帮助文档内容的循环滚动…”，可以看出整个系统的流程是：输入→文档检索→查询结果按字母排序→剪切行首单词→单词插入行尾→输出。

试题二

阅读以下关于软件架构设计的叙述，在答题纸上回答问题 1 至问题 3。

【说明】

RMO 是一家运动服装制造销售公司，计划在 5 年时间内将销售区域从华南地区扩展至全国范围。为了扩大信息技术对于未来业务发展的价值，公司邀请咨询顾问帮助他们制订战略信息系统规划。经过评审，咨询顾问给出的战略规划要点之一是建立客户关系支持系统 CRSS。RMO 公司决定由其技术部成立专门的项目组负责 CRSS 的开发和维护工作。

项目组在仔细调研和分析了系统需求的基础上，确定了基于互联网的 CRSS 系统架构。但在确定系统数据架构时，张工认为应该采用集中式的数据架构，给出的理由是结构简单、易维护且开发及运行成本低；而刘工建议采用分布式的数据架构，并提出在开发中通过“局部数据库+缓存”的读写分离结构实现，具有较好的运行性能和可扩展性。

项目组经过集体讨论，考虑到公司的未来发展规划，最终采用了刘工的建议。

【解析】 本题考查考生对于软件系统数据架构建模的掌握情况。

数据架构定义了信息系统中文件和数据库的分布结构。数据架构建模是以数据为中心，建模业务数据类型和结构，以及设计满足应用需求的数据库系统。传统以主机为中心的信息系统开发中，利用单个的数据库系统实现数据的集中式存储，物理上所有的数据位于同一个位置，构成的是一种集中式的数据架构；现代基于网络的分布式系统开发中，很少有组织会将其全部的数据存储在单个的数据库中，通常需要多个数据库系统组成，数据在这些数据库系统之间可以传送，由多个不同的数据库管理系统控制，构成的是一种分布式的数据架构。

【问题 1】（8 分）

请用 300 字以内的文字，说明张工和刘工提出的数据架构的基本思想。

【答案】

(1) 张工提出的集中式数据架构是由一个处理器、与它相关联的数据存储设备以及其他外围设备组成，它被物理地定义到单个位置。系统提供数据处理能力，用户可以在同样的站点上操作，也可以在地理位置隔开的其他站点上通过远程终端来操作。系统及其数据管理被某个或中心站点集中控制。

(2) 刘工提出的分布式数据架构使用多个计算机系统上的多个局部数据库系统构成，数据可以在多个不同的局部数据库中进行传送，由不同的数据库管理系统软件并行管理，运行在多种不同的计算机上，支持多种不同的操作系统。这些机器位于（或分布在）不同的地理

位置并通过多种通信网络连接在一起。企业数据可以分布在不同的计算机上，一个应用程序可以操作位于不同地理位置的机器上的数据。

【解析】

集中式数据架构中，一个或多个局域网中的客户共享一个单独计算机系统中的一个数据库。系统提供数据处理能力，用户可以在同样的站点上操作，也可以在地理位置隔开的其他站点上通过远程终端来操作。系统及其数据管理被某个或中心站点集中控制。单个数据库服务器结构的主要优点就是简单、易维护开发及运行成本低；但由于所有的客户直接请求服务器，容易发生性能瓶颈，如果服务失败，单个服务器不能提供备份和恢复，所有依赖的应用程序都将不能工作。

分布式数据架构中，使用多个计算机系统以及用户能够访问远程系统的数据，数据可以在多个不同的数据库中进行传送，由不同的数据库管理系统软件进行管理，运行在多种不同的计算机上，支持多种不同的操作系统。这些机器位于（或分布在）不同的地理位置并通过多种通信网络连接在一起。企业数据可以分布在不同的计算机上，一个应用程序可以操作位于不同地理位置的机器上的数据。多个数据库服务器结构的主要优点就是系统的容错能力和对广域网容量的需求有所降低，可以采用多种策略提升整个系统的服务质量；由于多个数据库系统分布在不同的网络节点上，位于不同位置的数据之间需要同步和协作，系统结构复杂、运行成本高并且维护困难。

在实际应用系统的数据架构建模中，应根据不同的应用需求选择集中式或分布式的数据架构。本题中由于 RMO 要扩展其销售区域，其潜在客户数量也会因此大幅度增加，所以良好的可扩展性是 CRSS 系统所必需的质量属性；并且由于其销售区域扩大后，系统中的数据会存储于不同的地理位置，所以采用分布式数据架构最为合理。

【问题 2】（13 分）

在刘工建议的基础上，为了避免 CRSS 系统的单点故障，请用 200 字以内的文字简要说明如何建立 CRSS 的数据库系统；对于数据的读取、添加、更改和删除操作分别如何实现。

【答案】

读写分离架构利用了数据库的复制技术，将数据的读和写分布在不同的处理节点上，从而达到提高可用性和扩展性的目的。

CRSS 的分布式数据库系统需要由多个局部数据库系统、多个热备份数据库系统和多个数据缓存组成。局部数据库负责数据的写入，多个热备份数据库系统用以解决单点故障的问

题，数据缓存负责为应用提供所读取的数据。

(1) 读取数据：应用访问缓存，如果命中则返回，否则从局部数据库系统中读取数据并将数据加载到缓存后返回。

(2) 添加数据：采用延迟加载策略，应用将数据直接写入局部数据库。

(3) 更改数据：应用更改局部数据库中的数据，将缓存中的数据标记为失效。

(4) 删除数据：应用删除局部数据库中的数据，将缓存中的数据标记为失效。

【解析】

读写分离架构应用非常广泛，很多网站采用数据库+缓存的方式来实现。通过缓存层来承载大量的读访问，如广泛采用的 Memcached，其自身往往不具备持久层存储的功能，通常和数据库一起组成分布式的数据架构，由数据库负责数据持久化存储和写入功能，缓存负责承载大量的并发访问，从而提高了系统的数据处理效率。要避免数据访问的单点故障，通常采用主数据库热备份的方式来实现。所以，要实现题目要求的分布式数据架构，需要多个局部数据库系统、多个热备份数据库系统和多个数据缓存组成。

读写分离结构中，应用读取数据时访问缓存，如果没有命中所需数据，则从主数据库中读取数据并写入缓存；对于新增、修改和删除操作，需要采用延迟加载的策略，新增时只修改主数据库，修改和删除时除了修改主数据库中的内容，还需要将缓存中的数据标记为失效。

【问题 3】（4 分）

RMO 公司销售区域将在未来 5 年大面积扩展，其潜在客户数量也会因此大幅度增加，所以良好的可扩展性是 CRSS 系统所必需的质量属性。请分别说明在集中式和分布式数据架构下，可以采用哪些方法提升系统的可扩展性。

【答案】

张工提出的集中式数据架构通过向上扩展（Scale Up）提升系统的可扩展性。具体的实现方式包括硬件扩容（增加 CPU 数量、内存容量、磁盘数量）和硬件升级（更换为高端主机或高速磁盘等）。

刘工提出的分布式数据架构通过向外扩展（Scale Out）提升系统的可扩展性。具体的实现方式包括数据复制、数据垂直切分或/和水平切分、缓存和全文搜索。

【解析】

传统的集中式数据架构中由于只有单个的数据库系统，所以要满足可扩展性的要求，更多的只能通过硬件的方式来实现。具体的实现方式包括硬件扩容（增加 CPU/内存容量/磁盘

数量）和硬件升级（更换高端主机或高速磁盘等）。

基于网络的分布式数据架构中由多个数据库系统共同组成，可以通过更改和优化数据分布来满足系统可扩展性的要求。具体的实现方式包括数据复制、数据垂直切分（或/和）水平切分、缓存和全文搜索。

试题三

阅读以下关于软件架构设计的叙述，在答题纸上回答问题 1 至问题 3。

【说明】

某软件公司承担了某项国家重点项目的研制工作，任务重点是参考国外汽车电子发展趋势，开发某汽车公司的企业汽车电子基础软件的架构，逐步实现汽车企业未来技术发展规划。

该软件公司接受此项任务后，调动全体技术人员深入收集国外相关技术资料，经过多方调研和分析，公司提出遵照国际组织最新推出的 AUTOSAR 规范，按统筹规划、分步骤实施的原则，实现汽车公司的基础软件架构设计。图 3-1 给出了 AUTOSAR 规范所定义的工作包，图中灰色部分代表本项目工作所包含的内容，即软件架构和基础软件。

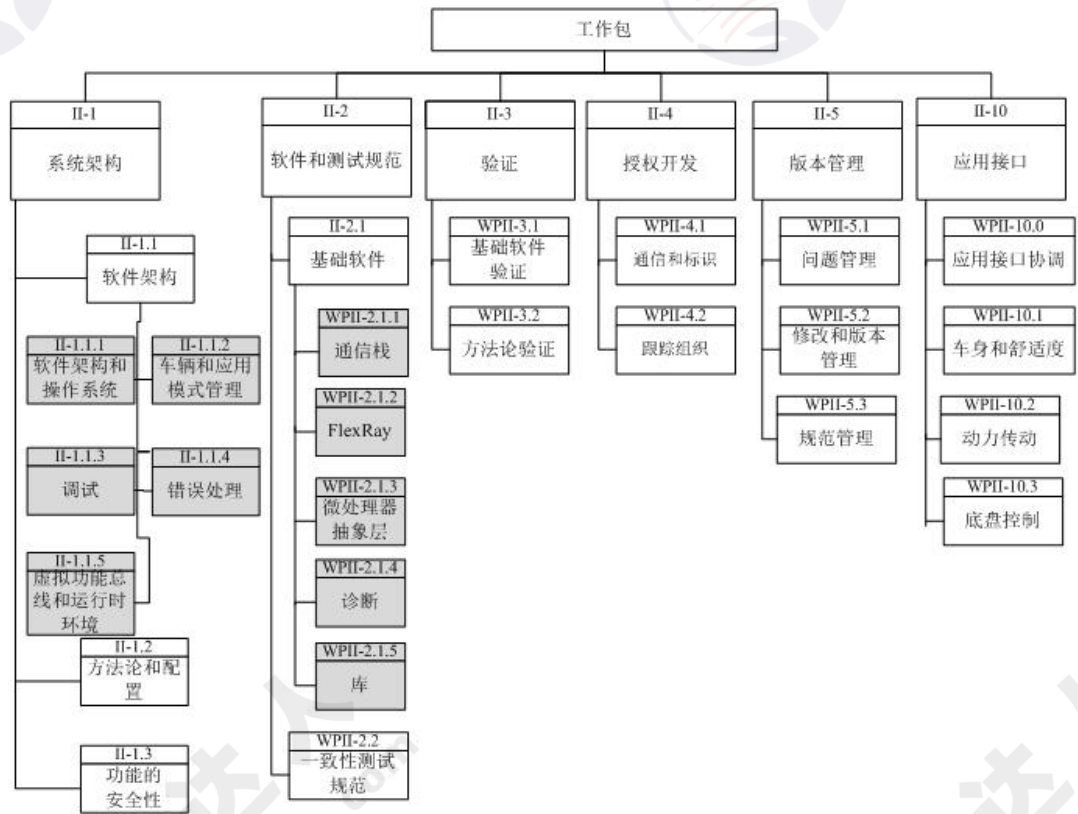


图 3-1 AUTOSAR 定义的工作包

【解析】

本题主要考查汽车电子基础软件架构的分析与设计，特别是系统的开发和架构设计方面。

【问题 1】（7 分）

AUTOSAR 规范中要求，汽车电子软件开发流程应尽量满足并发、可多次迭代的特性。为了定义汽车电子的软件开发过程，公司李工和王工分别提出了两种软件开发流程，其开发流程见图 3-2 和图 3-3（图中 ECU 是指汽车电子中的电子控制单元）。请说明李工和王工谁

定义的流程更符合 AUTOSAR 的规定，并说明理由。

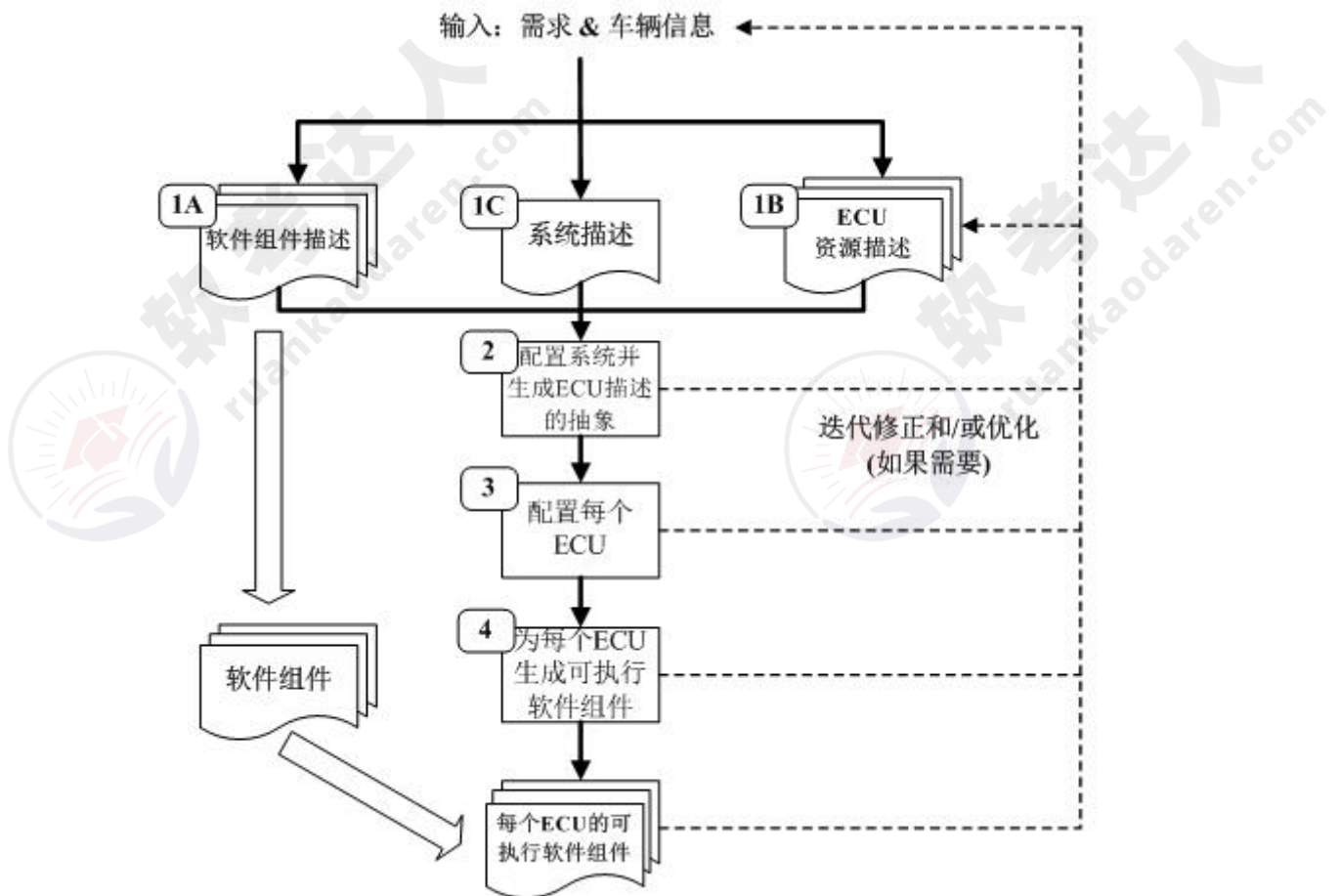


图 3-2 李工设计的流程

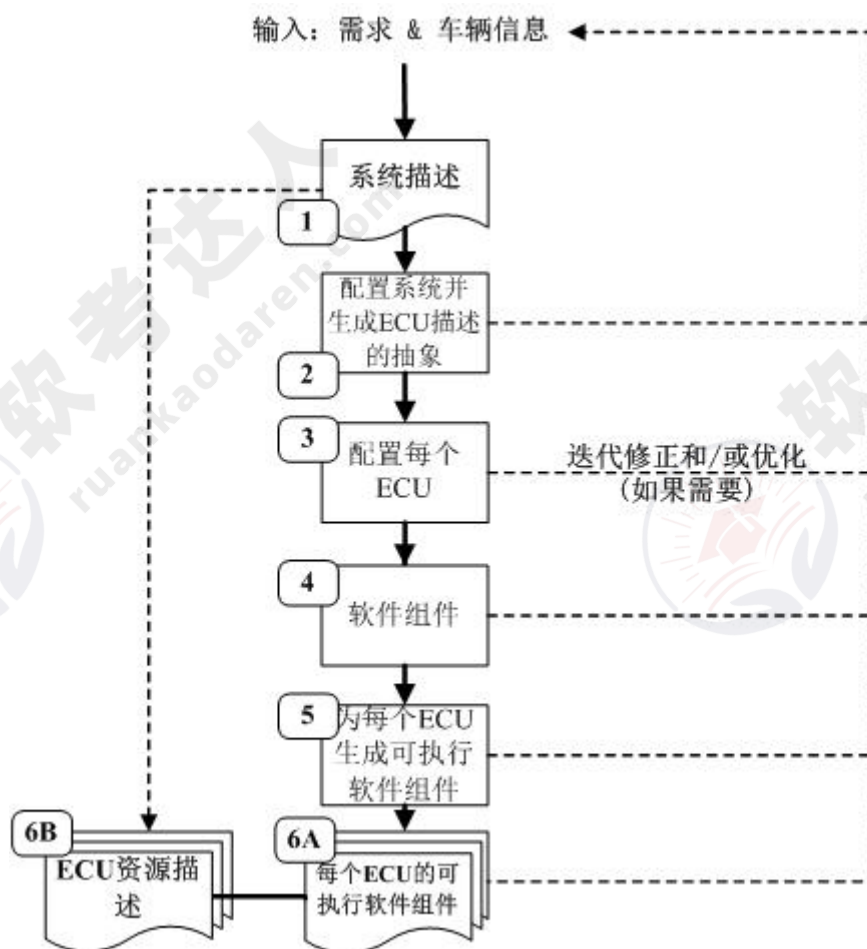


图 3-3 王工设计的流程

【答案】

李工设计的流程符合 AUTOSAR 要求，理由是：

李工定义的流程是将软件组件描述、系统描述和 ECU 资源描述同时定义，而王工定义的流程仅仅只做系统描述；

王工定义的流程没有考虑软件组件的描述，只是简单将软件组件作为第 4 步被集成；李工定义的 ECU 软件开发流程的优势是明确了顶层定义阶段，并行度高，迭代清晰。

【解析】

本问题主要考查在一定规范的约束下，系统设计流程的设计与定义。AUTOSAR 规范中要求，汽车电子软件开发流程应尽量满足并发、可多次迭代的特性。因此应该紧扣该规范对系统设计流程方面的要求，对李工和王工定义的流程进行评价。具体来说：

(1) 李工定义的流程是将软件组件描述、系统描述和 ECU 资源描述同时定义，而王工定义的流程仅仅只做系统描述；

(2) 王工定义的流程没有考虑软件组件的描述，只是简单将软件组件作为第 4 步被集

成：

(3) 李工定义的 ECU 软件开发流程的优势是明确了顶层定义阶段，并行度高，迭代清晰。

综上，应该采用李工的设计方案。

【问题 2】(10 分)

图 3-1 中的 II-U.1 项中定义了软件架构和操作系统的要求，图 3-4 是满足 AUTOSAR 定义的操作系统各功能模块的层次结构，请说明 (1)~(5) 箭头所标的具体操作含义。

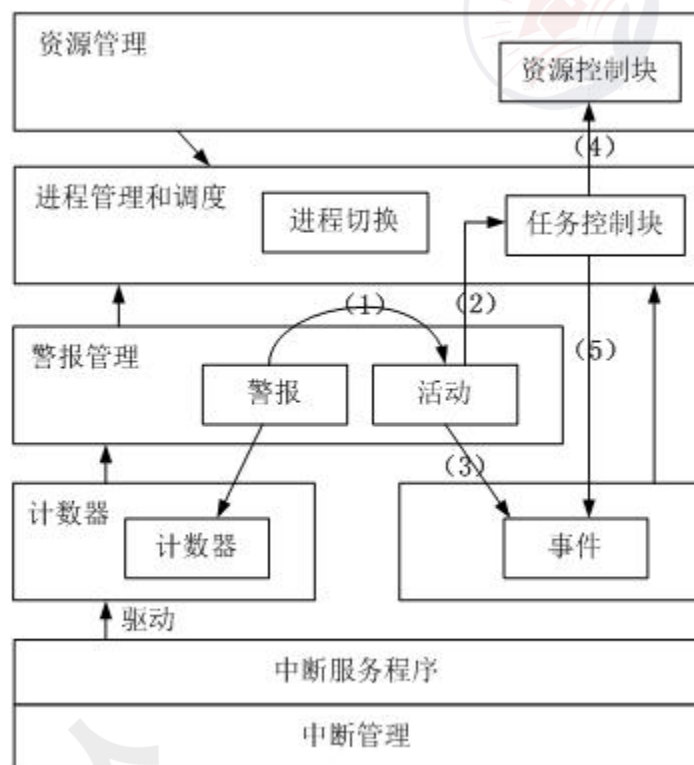


图 3-4 AUTOSAR 定义的操作系统结构

【答案】

- (1) 操作系统的警报管理发现 ECU 系统出错时，启动错误处理程序；
- (2) 错误处理程序将具体动作交由进程管理完成对发生错误的任务进行处理；
- (3) 错误处理程序产生一个错误事件；
- (4) 任务控制块处理程序调用资源管理功能，实现硬件资源重分配；
- (5) 任务控制块处理程序通知事件管理，对错误事件进行应答。

【解析】

本问题是一道读图题，要求考生在分析 AUTOSAR 定义的操作系统结构进行分析与理解的

基础上进行填写。根据图示，系统从下至上依次可以分为中断管理、事件管理、警报管理、进程管理和调度以及资源管理四个部分。根据图中模块之间的关系，可以看出，(1)处主要表示当操作系统的警报管理发现 ECU 系统出错时，启动错误处理程序；(2)表示错误处理程序将具体动作交由进程管理完成对发生错误的任务进行处理；(3)表示错误处理程序产生一个错误事件；(4)表示任务控制块处理程序调用资源管理功能，实现硬件资源重分配；(5)表示任务控制块处理程序通知事件管理，对错误事件进行应答。

【问题 3】(8 分)

AUTOSAR 是一种开放式架构，用 150 字以内的文字，说明采用 AUTOSAR 架构的主要优点，并说明汽车电子 ECU 覆盖汽车的哪三个领域。

【答案】

采用 AUTOSAR 开放式架构的优点是：

(1)可以有效支持多厂家汽车电子基础软件的研制；

(2)有利于软件的重用，可根据不同的 ECU 结构，通过数据配置，自动生成各种 ECU 软件组件；

(3)AUTOSAR 定义的软件框架支持了汽车电子软件的全生存周期，包括构架、开发、测试、验证、授权、版本和接口。

AUTOSAR 规范覆盖整个汽车电子的三大领域：动力、底盘、车身。

【解析】

本题主要考查考生对 AUTOSAR 架构的分析与总结能力。根据题干和上述两个题目的回答，可以看出，采用 AUTOSAR 开放式架构的优点主要包括：

(1)具有厂商独立性，可以有效支持多厂家汽车电子基础软件的研制；

(2)软件层次上的重用性，可根据不同的 ECU 结构，通过数据配置，自动生成各种 ECU 软件组件；

(3)支持汽车电子软件的全生存周期，包括构架、开发、测试、验证、授权、版本和接口；

另外，该规范覆盖整个汽车电子的三大领域为动力、底盘和车身。

试题四

阅读以下关于软件架构设计的叙述，在答题纸上回答问题 1 至问题 3。

【说明】

TeleDev 是一个大型的电信软件开发公司，公司内部采用多种商业/开源的工具进行软件系统设计与开发工作。为了提高系统开发效率，公司管理层决定开发一个分布式的系统设计与开发工具集成框架，将现有的系统设计与开发工具有效集成在一起。集成框架开发小组经过广泛调研，得到了如下核心需求。

(1) 目前使用的系统设计与开发工具的运行平台和开发语言差异较大，集成框架应无缝集成各个工具的功能；

(2) 目前使用的系统设计与开发工具所支持的通信协议和数据格式各不相同，集成框架应实现工具之间的灵活通信和数据格式转换；

(3) 集成框架需要根据实际的开发流程灵活、动态地定义系统工具之间的协作关系；

(4) 集成框架应能集成一些常用的第三方实用工具，如即时通信、邮件系统等。

集成框架开发小组经过分析与讨论，最终决定采用企业服务总线（ESB）作为集成框架的基础架构。

【解析】

本题主要考查系统集成的相关知识及应用，需要考生结合题干描述和自己的实际经验进行回答。

【问题 1】（8 分）

ESB 是目前企业级应用集成常用的基础架构。请列举出 ESB 的 4 个主要功能，并从集成系统的部署方式、待集成系统之间的耦合程度、集成系统的可扩展性三个方面说明为何采用 ESB 作为集成框架的基础架构。

【答案】

ESB 的主要功能包括：

- (1) 应用程序的位置透明性
- (2) 传输协议转换
- (3) 消息格式转换
- (4) 消息路由
- (5) 消息增强
- (6) 安全支持

(7) 监控和管理

采用 ESB 作为集成框架，能够实现灵活的部署结构，包括 CS 结构、P2P 结构等。采用 ESB 作为集成框架，待集成系统只需要和总线进行联系，彼此之间不需要互相通信，这样就大大降低了系统的耦合程度。

采用 ESB 作为集成框架，在加入新的待集成系统时，只需要采用插件的方式实现传输协议和数据格式的适配即可，系统的可扩展性较强。

【解析】

本问题主要考查企业服务总线（ESB）的基本概念，需要考生列举出企业服务总线七个核心功能中的任意四个，根据 ESB 的特点，其核心功能包括：

- (1) 应用程序的位置透明性
- (2) 传输协议转换
- (3) 消息格式转换
- (4) 消息路由
- (5) 消息增强
- (6) 安全支持
- (7) 监控和管理

根据集成系统的部署方式，可以看出采用 ESB 作为集成框架，能够实现灵活的部署结构，包括 CS 结构、P2P 结构等。

从待集成系统之间的耦合程度，可以看出采用 ESB 作为集成框架，待集成系统只需要和总线进行联系，彼此之间不需要互相通信，这样就大大降低了系统的耦合程度。

从集成系统的可扩展性，可以看出采用 ESB 作为集成框架，在加入新的待集成系统时，只需要采用插件的方式实现传输协议和数据格式的适配即可，系统的可扩展性较强。

【问题 2】（12 分）

在 ESB 基础架构的基础上，请根据题干描述中的 4 个需求，说明每个需求应该采用何种具体的集成方式或架构风格最为合适。

【答案】

对于需求（1）来说，由于需要共享系统的功能，并且系统的运行平台与语言差异较大，应该采用面向服务的方式进行功能集成，可以将工具的功能包装为服务，实现跨语言与跨平台访问。

对于需求(2)来说，工具所支持的通信协议和数据格式各不相同，并需要实现工具之间的灵活通信协议和数据格式交换，因此应该基于消息总线，以协议及数据适配器的方式实现灵活的通信协议和数据格式转换。

对于需求(3)来说，集成框架需要根据实际的软件系统开发流程，灵活、动态地定义系统设计与开发工具之间的协作关系，因此应该引入工作流定义语言及其引擎来动态描述工具之间的协作关系。

对于需求(4)来说，应该采用界面集成的方法对第三方工具进行集成，绕过工具内部的复杂处理逻辑。

【解析】

对于需求(1)“目前使用的系统设计与开发工具的运行平台和开发语言差异较大，集成框架应无缝集成各个工具的功能”来说，由于需要共享系统的功能，并且系统的运行平台与语言差异较大，应该采用面向服务的方式进行功能集成，可以将工具的功能包装为服务，实现跨语言与跨平台访问。

对于需求(2)“目前使用的系统设计与开发工具所支持的通信协议和数据格式各不相同，集成框架应实现工具之间的灵活通信和数据格式转换”来说，工具所支持的通信协议和数据格式各不相同，并需要实现工具之间的灵活通信协议和数据格式交换，因此应该基于消息总线，以协议及数据适配器的方式实现灵活的通信协议和数据格式转换。

对于需求(3)“集成框架需要根据实际的开发流程灵活、动态地定义系统工具之间的协作关系”来说，集成框架需要根据实际的软件系统开发流程，灵活、动态地定义系统设计与开发工具之间的协作关系，因此应该采用解释器架构风格，引入工作流定义语言及其引擎来动态描述工具之间的协作关系。

对于需求(4)“集成框架应能集成一些常用的第三方实用工具，如即时通信，邮件系统等”来说，应该采用界面集成的方法对第三方工具进行集成，绕过工具内部的复杂处理逻辑，实现功能集成。

【问题3】(5分)

请指出在实现工具之间数据格式的灵活转换时，通常采用的设计模式是什么，并对实现过程进行简要描述。

【答案】

在实现工具之间数据格式的灵活转换时，通常采用适配器设计模式。即应首先定义一个

统一的数据转换接口类，然后针对不同的数据格式转换需求定义对应的实际转换类，实际转换类需要继承数据转换接口类，并实现接口转换类定义的接口。

【解析】

本题主要考查数据转换在实现层面上的常用方法。在实现工具之间数据格式的灵活转换时，通常采用适配器设计模式。即应首先定义一个统一的数据转换接口类，然后针对不同的数据格式转换需求定义对应的实际转换类，实际转换类需要继承数据转换接口类，并实现接口转换类定义的接口。）

试题五

阅读以下信息系统可靠性的问题，在答题纸上回答问题 1 至问题 3。

某软件公司开发一项基于数据流的软件，其系统的主要功能是对输入的数据进行多次分析、处理和加工，生成需要的输出数据。需求方对该系统的软件可靠性要求很高，要求系统能够长时间无故障运行。该公司将该系统设计交给王工负责。王工给出该系统的模块示意图，如图 5-1 所示。王工解释：只要各个模块的可靠度足够高，失效率足够低，则整个软件系统的可靠性是有保证的。

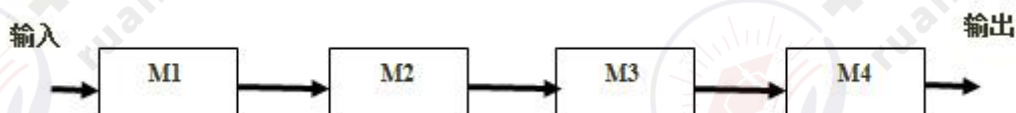


图 5-1 王工建议的软件系统模块示意图

李工对王工的方案提出了异议。李工认为王工的说法有两个问题：第一，即使每个模块的可靠度足够高，假设各个模块的可靠度均为 0.99，但是整个软件系统模块之间全部采用串联，则整个软件系统的可靠度为 $0.99^4=0.96$ ，即整个软件系统的可靠度下降明显；第二，软件系统模块全部采用串联结构，一旦某个模块失效，则意味着整个软件系统失效。

李工认为，应该在软件系统中采用冗余技术中的动态冗余或者软件容错的 N 版本程序设计技术，对容易失效或者非常重要的模块进行冗余设计，将模块之间的串联结构部分变为并联结构，来提高整个软件系统的可靠性。同时，李工给出了采用动态冗余技术后的软件系统模块示意图，如图 5-2 所示。

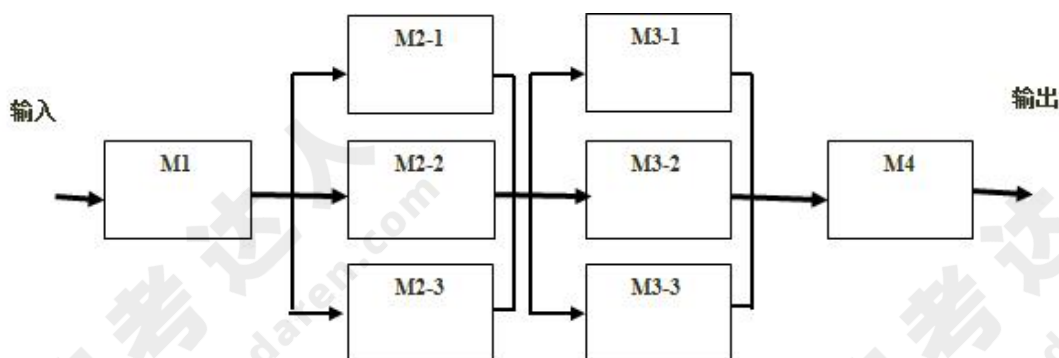


图 5-2 李工建议的系统模块示意图

刘工建议，李工方案中 M1 和 M4 模块没有采用容错设计，但 M1 和 M4 发生故障有可能导致严重后果。因此，可以在 M1 和 M4 模块设计上采用检错技术，在软件出现故障后能及时发现并报警，提醒维护人员进行处理。

注：假设各个模块的可靠度均为 0.99。

【解析】

本题考查信息系统中可靠性的设计，是比较传统的题目，要求考生细心分析题目中所描述的内容。

【问题1】（4分）

在系统可靠性中，可靠度和失效率是两个非常关键的指标，请分别解释其含义。

【答案】

可靠度就是系统在规定的条件下、规定的时间内不发生失效的概率。

失效率又称风险函数，也可以称为条件失效强度，是指运行至此时刻系统未出现失效的情况下，单位时间系统出现失效的概率。

【解析】

本问题考查信息系统可靠性的两个基本概念：可靠度和失效率。可靠性是指产品在规定的条件下和规定的时间内完成规定功能的能力。考虑到软件本身的复杂性，软件可靠性的定义是：在规定的条件下，在规定的时间内，软件不引起系统失效的概率。

在软件可靠性的定量描述中，软件可靠性可以基于使用条件、规定时间、系统输入、系统使用和软件缺陷等变量构建数学表达式，来对软件可靠性进行定量描述。相关概念有规定时间、失效概率、可靠度、失效强度、失效率、平均无失效时间等。其中可靠度是表示可靠性最直接的方式，是软件系统在规定的条件下、规定的实践内不发生失效的概率。而失效率又称风险函数，也可以称为条件失效强度，是指运行至此时刻系统未出现失效的情况下，单位时间系统出现失效的概率。

【问题2】（13分）

请解释李工提出的动态冗余和N版本程序设计技术，给出图5-1中模块M2采用图5-2动态冗余技术后的可靠度。

请给出采用李工设计方案后整个系统可靠度的计算方法，并计算结果。

【答案】

动态冗余又称为主动冗余，它是通过故障检测、故障定位及故障恢复等手段达到容错的目的。其主要方式是多重模块待机储备，当系统检测到某工作模块出现错误时，就用一个备用的模块来替代它并重新运行。各备用模块在其待机时，可与主模块一样工作，也可以不工作。前者叫热备份系统（双重系统），后者叫冷备份系统（双工系统、双份系统）。N版本程序设计是一种静态的故障屏蔽技术，其设计思想是用N个具有相同功能的程序同时执行一项

计算，结果通过多数表决来选择。其中 N 个版本的程序必须由不同的人独立设计，使用不同的方法、设计语言、开发环境和工具来实现，目的是减少 N 个版本的程序在表决点上相关错误的概率。

M2 采用动态冗余后的可靠度为： $R=1-(1-0.99)^3=0.999999$

李工给出的方案同时采用了串联和并联方式，其计算方法为首先计算出中间 M2 和 M3 两个并联系统的可靠度，再按照串联系统的计算方法计算出整个系统的可靠度。

$R=0.99 \times 0.999999 \times 0.999999 \times 0.99=0.98$

【解析】

本问题考查在常规的软件设计中，应用各种方法和技术，使程序设计在兼顾用户功能和性能需求的同时，全面满足软件的可靠性要求。常见的软件可靠性技术主要有容错设计、检错设计和降低复杂度设计等技术。

其中，容错设计技术主要有恢复快设计、N 版本程序设计和冗余设计三种方法。N 版本程序设计是一种静态的故障屏蔽技术，其设计思想是用 N 个具有相同功能的程序同时执行一项计算，结果通过多数表决来选择。其中 N 个版本的程序必须由不同的人独立设计，使用不同的方法、设计语言、开发环境和工具来实现，目的是减少 N 个版本的程序在表决点上相关错误的概率。动态冗余又称为主动冗余，它是通过故障检测、故障定位及故障恢复等手段达到容错的目的。其主要方式是多重模块待机储备，当系统检测到某工作模块出现错误时，就用一个备用的模块来替代它并重新运行。各备用模块在其待机时，可与主模块一样工作，也可以不工作。前者叫热备份系统（双重系统），后者叫冷备份系统（双工系统、双份系统）。计算机系统是一个复杂系统，影像其可靠性的因素很多，很难直接进行可靠性分析，往往需要建立对应的数学模型。组合模型是分析系统可靠性的一种常用方法。

组合模型下可靠度的计算方法为：

串联系统： $R=R_1 \times R_2 \times \dots \times R_n$ ；

并联系统： $R=1-(1-R_1) \times (1-R_2) \times \dots \times (1-R_n)$ ；

串联和并联混合系统则根据实际情况，灵活运用上述两个计算公式。

M2 采用动态冗余后，成为并联系统，则其可靠度为： $R=1-(1-0.99)^3=0.999999$ 。

李工给出的方案同时采用了串联和并联方式，其计算方法为首先计算出中间 M2 和 M3 两个并联系统的可靠度，再按照串联系统的计算方法计算出整个系统的可靠度。

$R=0.99 \times 0.999999 \times 0.999999 \times 0.99=0.98$

【问题 3】（8 分）

请给出检错技术的优缺点，并说明检测技术常见的实现方式和处理方式。

【答案】

检错技术实现的代价一般低于容错技术和冗余技术，但有一个明显的缺点，就是不能自动解决故障，出现故障后如果不进行人工干预，将最终导致软件系统不能正常运行。

检错技术常见的实现方式：最直接的一种实现方式是判断返回结果，如果返回结果超出正常范围，则进行异常处理；计算运行时间也是一种常用技术，如果某个模块或函数运行时间超过预期时间，可以判断出现故障；还有置状态标志位等多种方法，自检的实现方式需要根据实际情况来选用。

检错技术的处理方式，大多数都采用“查出故障-停止软件运行-报警”的处理方式。但根据故障的不同情况，也有采用不停止或部分停止软件系统运行的情况，这一般由故障是否需要实时处理来决定。

【解析】 本问题考查软件可靠性设计中的检错技术。

检错技术常见的实现方式有多种，最直接的一种实现方式是判断返回结果，如果返回结果超出正常范围，则进行异常处理；计算运行时间也是一种常用技术，如果某个模块或函数运行时间超过预期时间，可以判断出现故障；还有置状态标志位等多种方法，自检的实现方式需要根据实际情况来选用。

检错技术的处理方式也有多种，大多数都采用“查出故障-停止软件运行-报警”的处理方式。但根据故障的不同情况，也有采用不停止或部分停止软件系统运行的情况，这一般由故障是否需要实时处理来决定。

检错技术实现的代价一般低于容错技术和冗余技术，但有一个明显的缺点，就是不能自动解决故障，出现故障后如果不进行人工干预，将最终导致软件系统不能正常运行。

试题一 论软件的静态演化和动态演化及其应用

软件演化 (Software Evolution) 是指软件在其生命周期内的更新行为和过程。演化是一系列贯穿软件生命周期始终的活动，系统需求改变、功能实现增强、新功能加入、软件架构改变、软件缺陷修复、运行环境改变均要求软件系统能够快速适应变化，具有较强的演化能力。软件静态演化 (Static Evolution) 和动态演化 (Dynamic Evolution) 是目前软件演化的两种重要类型。

请围绕“软件的静态演化和动态演化及其应用”论题，依次从以下三个方面进行论述。

1. 概要叙述你参与管理和开发的软件项目以及你在其中所担任的主要工作。
2. 请分别对软件静态演化和动态演化的特点进行论述，说明两种软件演化类型各自的优缺点及其应用场合，并举例说明各自的常见演化技术手段。
3. 具体阐述你参与管理和开发的项目中所进行的软件演化活动的特点、演化的类型，以及所采取的对应演化技术手段，说明具体实施过程以及实际应用的效果。

1. 简要叙述所参与管理和开发的软件项目，并明确指出自己在其中承担的主要任务和开展的主要工作。

2. 软件演化可分为静态演化和动态演化两种情形。

(1) 静态演化 (Static Evolution)。静态演化是指软件在停机状态下的演化。其优点是不用考虑运行状态的迁移，同时也没有活动的进程需要处理。然而停止一个应用程序就意味着中断它提供的服务，造成软件暂时失效。

软件静态演化是指发生在应用程序停止时的软件修改和更新，即一般意义上的软件维护和升级。静态演化的优点是没有状态迁移或活动线程的问题要解决，缺陷是停止应用程序意味着停止它所提供的服务，也就是使软件系统暂时失效。在软件交付之后，静态演化（类似于一般意义上的软件维护）就成为软件变更的一个常规过程。变更可以是一种更正代码错误的简单变更，也可以是更正设计错误的较大范围的变更，还可以是对描述错误进行修正或提供新需求这样的重大改进。有三种不同的软件维护：改正性维护、适应性维护和完善性维护。维护过程一般包括变更分析、版本规划、系统实现和向客户交付系统等活动。

在面向对象技术中，使用子类型方法来扩展程序，它适合于软件静态演化和代码重用。子类型化一个类意味着保留父类中的参数和方法，并尽可能地增加新的参数和方法。

另外，使用重载和多态性作为主要的演化机制。实际上，建立类的新版本，最简单的机制是创建它的子类，然后重载需要变更的方法，最后，使用多态性调用新创建的方法。在

基于构件的软件技术中，构件采取接口和实现相分离的技术，构件之间只能通过接口进行通信，这使得具有兼容接口的不同构件实现可以相互取代，从而成为软件静态演化 的一条途径。

(2) 动态演化 (Dynamic Evolution)。动态演化是指软件在执行期间的软件演化。其优点是软件不会存在暂时的失效，有持续可用性的明显优点。但由于涉及状态迁移等问题，比静态演化从技术上更难处理。

动态演化是最复杂也是最有实际意义的演化形式。动态演化使得软件在运行过程中，可以根据应用需求和环境变化，动态地进行软件的配置、维护和更新，其表现形式 包括系统元素数目的可变性、结构关系的可调节性和结构形态的动态可配置性。软件的动态演化特性对于适应未来软件发展的开放性、动态性具有重要意义。

动态演化是指软件在运行期间的演化。在许多重要的应用领域中，例如金融、电力、电信及空中交通管制等，系统的持续可用性是一个关键性的要求，运行时刻的系统演化可减少因关机和重新启动而带来的损失和风险。此外，越来越多的其他类型的应用软件也提出了运行时刻演化的要求，在不必对应用软件进行重新编译和加载的前提下，为最终用户提供系统定制和扩展的能力。

动态演化可分为两种类型：预设的和非预设的。在 Web 环境中，软件应用常常需要处理多种类型的信息，因此它们常被设计为可以动态下载并安装插件以处理当前所面临的新类型的信息；而分布式 Web 应用也常常需要增减内部处理节点的数目以适应多变的负载。这些动态改变都是软件设计者能够预先设想到的，可实现为系统的固有功能。另有一些必须对系统配置进行修改和调整的情况是直到系统投入运行以后才发现的，这就要求系统能够处理在原始设计中没有完全预料到的新需求。这种情况下一般需要关闭整个系统，重新开发、重新装入并重新启动系统。然而，为了进行局部的修改而关闭整个系统在某些情况下是不允许的（例如，关键运行系统）或者代价太高。精心设计的动态演化技术可以在不关闭整个系统的前提下修改系统的结构配置，并尽量使未受影响的部分继续工作以提高系统的可用度。

为支持软件的动态演化性，已在语言、机制和环境等方面做了大量工作。在程序语言的层次上，引进各种机制以支持软件动态演化，例如动态装载技术允许增加代码到已运行的程序中，延迟绑定是在运行时而不是编译时决定类和对象的绑定。Java hotswap 允许在运行时改变方法：当一个方法终止时，这个方法的新版本可以有效地替换旧版本，在类层次上代码的二进制兼容被支持。Gilgul 语言也允许更换运行时对象。但程序语言层次上的动态演化机制仅局限于函数、类方法和对象等小粒度的替换，只支持预设的有限变更，变更由事件触发。

通过标准化运行级构件的规约，依靠构件运行平台（中间件平台）提供的基础设施，使软件在构件层次上的动态演化成为可能。中间件中具有的如命名服务、反射技术和动态适配等机制，为运行态构件的动态替换和升级提供支撑，从而推动了软件动态演化发展。命名服务就是给构件实例提供一个名称，以便客户通过这些名称来获取构件实例。对工业标准构件 EJB 和 CORBA 构件的引用都可以通过中间件平台的命名服务进行。同一构件标识可以被映射到多个构件实例，从而根据具体情境对某一名字的构件引用导向到不同的构件实例。反射技术是系统的一种自描述（self-representation）和自推理的技术，它提供了关于自身行为的表示，这种表示可以被检查和调整，且与它所描述的系统行为是因果相联（causally connected）的。因果相联，意味着对自表示的改动将立即反映在系统的实际状态和行为中，反之亦然。将反射性引入中间件能够以可控的方式开放平台内部的实现，从而提高中间件的定制能力和运行时的适应能力。动态适配机制中比较著名的是 CORBA 提供的动态接口服务：动态调用接口 DII 和动态骨架接口 DSI。前者支持动态客户请求调用，而后者支持将请求动态指派（Dispatch）给构件。因此，软件构件化技术使得软件具有良好的构造性，软件演化的粒度更大。中间件技术则为基于构件的软件动态演化提供了坚实的基础设施和方便的操作界面。

3. 考生需结合自身参与项目的实际状况，指出其参与管理和开发的项目中所进行的软件演化活动的特点、演化的类型，以及所采取的对应该演化技术手段。要给出实施软件演化活动的具体过程、方法以及对实际应用效果的分析。

试题二

随着信息技术的高速发展，各组织机构积累的数据量急剧增长。如何从海量的数据中提取有用的知识成为当务之急。数据挖掘（DataMining）就是为顺应这种需要而发展起来的数据处理技术，是知识发现的关键步骤。数据挖掘就是从大量的、不完全的、有噪声的、模糊的、随机的实际应用数据中，提取隐含在其中的、人们事先不知道的、但又是潜在有用的信息和知识的过程。

请围绕“论数据挖掘技术的应用”论题，依次从以下三个方面进行论述。

1. 概要叙述你参与分析和开发的软件项目以及你所担任的主要工作。
2. 数据挖掘的主要任务是什么？具体论述你在项目中使用数据挖掘技术所解决的问题。
3. 数据挖掘的方法主要有哪些？分析并讨论你所选择的数据挖掘方法，简述其具体实现过程和实际应用效果。

1. 结合自己所参与的软件项目，概要介绍该项目的背景及主要内容，并明确指出在其中所承担的主要任务和开展的主要工作。

2. 数据挖掘的任务主要是关联分析、聚类分析、分类、预测、时序模式和偏差分析等。

(1) 关联分析。两个或两个以上变量的取值之间存在某种规律性，就称为关联。数据关联是数据库中存在的一类重要的、可被发现的知识。关联分析的目的是找出数据库中隐藏的关联网。一般用支持度和可信度两个阈值来度量关联规则的相关性。

(2) 聚类分析。聚类是把数据按照相似性归纳成若干类别，同一类中的数据彼此相似，不同类中的数据相异。聚类分析可以建立宏观的概念，发现数据的分布模式，以及可能的数据属性之间的相互关系。

(3) 分类。分类就是找出一个类别的概念描述，它代表了这类数据的整体信息，即该类的内涵描述，并用这种描述来构造模型，一般用规则或决策树模式表示。分类是利用训练数据集通过一定的算法而求得分类规则。分类可被用于规则描述和预测。

(4) 预测。预测是利用历史数据找出变化规律，建立模型，并由此模型对未来数据的种类及特征进行预测。预测的精度和不确定性被重点关注，通常用预测方差来度量。

(5) 时序模式。时序模式是指通过时间序列搜索出的重复发生概率较高的模式。与回归一样，它也是用已知的数据预测未来的值，但这些数据的区别是变量所处时间的不同。

(6) 偏差分析。在偏差中包括很多有用的知识，数据库中的数据存在很多异常情况，发现数据库中数据存在的异常情况是非常重要的。偏差检验的基本方法就是寻找观察结果与参

照之间的差别。

论文中须明确指出自己在该项目应用数据挖掘技术所要解决的具体问题是什么。

3. 主要的数据挖掘方法

(1) 神经网络方法

神经网络由于本身良好的鲁棒性、自组织自适应性、并行处理、分布存储和高度容错等特性非常适合解决数据挖掘的问题，因此近年来越来越受到人们的关注。典型的神经网络模型主要分三大类：以感知机、BP 反向传播模型、函数型网络为代表的用于分类、预测和模式识别的前馈式神经网络模型；以 hopfield 的离散模型和连续模型为代表的分别用于联想记忆和优化计算的反馈式神经网络模型；以 art 模型、koholon 模型为代表的，用于聚类的自组织映射方法。神经网络方法的缺点是“黑箱”性，人们难以理解网络的学习和决策过程。

(2) 遗传算法

遗传算法是一种基于生物自然选择与遗传机理的随机搜索算法，是一种仿生全局优化方法。遗传算法具有的隐含并行性、易于和其他模型结合等性质使得它在数据挖掘中被加以应用。

(3) 决策树方法

决策树是一种常用于预测模型的算法，它通过将大量数据有目的地分类，从中找到一些有价值的、潜在的信息。它的主要优点是描述简单，分类速度快，特别适备大规模的数据处理。最有影响和最早的决策树方法是由 Quinlan 提出的著名的基于信息熵的 id3 算法。它的主要问题是：id3 是非递增学习算法；id3 决策树是单变量决策树，复杂概念的表达困难；同性间的相互关系强调不够；抗噪性差。针对上述问题，出现了许多较好的改进算法，如 Schlimmer 和 Fisher 设计了 id4 递增式学习算法等。

(4) 粗集方法

粗集理论是一种研究不精确、不确定知识的数学工具。粗集方法有几个优点：不需要给出额外信息；简化输入信息的表达空间；算法简单，易于操作。粗集处理的对象是类似二维关系表的信息表。目前成熟的关系数据库管理系统和新发展起来的数据仓库管理系统为粗集的数据挖掘奠定了坚实的基础。

(5) 覆盖正例排斥反例方法

它是利用覆盖所有正例、排斥所有反例的思想来寻找规则。首先在正例集合中任选一个种子，到反例集合中逐个比较。与字段取值构成的选择子相容则舍去，相反则保留。按此思想循环所有正例种子，将得到正例的规则。比较典型的算法有 Michalski 的 aql1 方法等。

(6) 统计分析方法

在数据库字段项之间存在两种关系：函数关系（能用函数公式表示的确定性关系）和相关关系（不能用函数公式表示，但仍是相关确定性关系），对它们的分析可采用统计学方法，即利用统计学原理对数据库中的信息进行分析。可进行常用统计（求大量数据 中的最大值、最小值、总和、平均值等）、回归分析（用回归方程来表示变量间的数量关系）、相关分析（用相关系数来度量变量间的相关程度）、差异分析（从样本统计量的值得出差异来确定总体参数之间是否存在差异）等。

(7) 模糊集方法

利用模糊集合理论对实际问题进行模糊评判、模糊决策、模糊模式识别和模糊聚类分析。系统的复杂性越高，则模糊性越强，一般模糊集合理论是用隶属度来刻画模糊事物的亦此亦彼性的。

论文中必须明确指出使用了上述 7 种方法中的哪种或哪几种数据挖掘方法，并给出该方法的具体实现过程；分析所选择的数据挖掘方法的实现效果。

试题三

大规模分布式系统通常需要利用缓存技术减轻服务器负载、降低网络拥塞、增强系统可扩展性。缓存技术的基本思想是将客户最近经常访问的内容在缓存服务器中存放一个副本，当该内容下次被访问时，不必建立新的数据请求，而是直接由缓存提供。良好的缓存设计是一个大规模分布式系统能够正常、高效运行的必要前提。在进行大规模分布式系统开发时，必须从一开始就针对应用需求和场景对系统的缓存机制进行全面考虑，设计一个可伸缩的系统缓存架构。

请围绕“大规模分布式系统缓存设计策略”论题，依次从以下三个方面进行论述。

1. 概要叙述你参与实施的大规模分布式系统开发项目以及你所担任的主要工作。
2. 从不同的用途和应用场景考虑，请详细阐述至少两种常见的缓存工作模式，并说明每种工作模式的适应场景。

3. 阐述你在设计大规模分布式系统的缓存机制时遇到了哪些问题，如何解决。

1. 论文中要具体介绍项目的总体需求（特别是应用需求中对缓存机制的要求）、系统的逻辑与物理架构、采用的技术等内容和自己担任的实际工作。

2. 从不同的用途和应用场景来考虑，大体上可以将缓存分为三种工作模式，即单实例缓存模式(Single Instance)、复制模式(Replication Cache)和分区模式(Partition Cache)。每种工作模式都有其适应的场景和优缺点。

- (1) 单实例模式。单实例模式是一种较为简单的缓存模式，多个应用服务器共享一个中央的缓存服务器。通过共享缓存的数据，能够极大地提高系统的性能。该模式的主要限制在于缓存服务器的内存大小和节点增加之后服务器的处理能力和网络带宽。该模式的适用场景是：对缓存的要求比较简单；系统的吞吐量和数据量不大；性能要求不高。

- (2) 复制模式。复制模式将缓存的数据复制到多台机器上，对于单一缓存服务器性能出现问题的情况下，可以通过缓存复制的方式将压力分解到多个缓存服务器。该模式的工作原理是：缓存客户端可以访问自己的缓存服务器，多个缓存服务器之间的数据是彼此同步的，对于性能要求更高的场景，这样的部署架构能够获得更高的吞吐能力。该模式的适用场景是：数据量不是特别大；需要极高的性能；数据改动的频率不是特别大。

- (3) 分区模式。当需要缓存的数据已经超过一台服务器的内存上限时，可以考虑采用分区模式对数据进行线性缩放，也就是通过增加缓存服务器来解决数据增长和压力增加的情况。在分区模式中，其架构是无分享架构（SharedNothing Architecture，SNA），每个节点之间数

据彼此独立，一个节点出现故障后不会影响到其他节点。在出现某个节点宕机或者其他故障的情况下，致使这部分的分区缓存无法使用，并不妨碍其他数据节点数据的正常工作。该模式的适用场景是：总体数据量较大。已经超出了单个缓存服务器的内存上限；系统缓存要求具有很大的可伸缩性；客户端量庞大，单个客户端对缓存数据的数据量要求不大。

3. 进行大规模分布式系统缓存机制设计时可能遇到的问题包括：如何缓存服务器的工作模式选择；高可用性的设计考虑；缓存一致性与分布式算法；对象状态同步的考虑；缓存钝化、激活、过期和初始化，等等。

试题四 论软件可靠性评价

软件可靠性评价是指选用和建立合适的可靠性数学模型，运用统计技术和其他手段，对软件可靠性测试和系统运行期间的软件失效数据（也可能包含软件生命周期内其他可靠性数据）进行处理，并评估和预测软件可靠性的过程。

软件可靠性评价是软件可靠性活动的重要组成部分，既可在软件开发过程实施，也可针对最终软件系统实施。软件可靠性评价的难点在于软件可靠性模型的选择和软件可靠性数据的收集与处理。

请围绕“软件可靠性评价”论题，依次从以下三个方面进行论述。

1. 简要概述你参与实施的软件开发项目以及你承担的主要工作。
2. 说明你在项目实施过程中所选择的软件可靠性模型，并论述在软件可靠性模型选择时应该考虑的主要因素。
3. 收集软件可靠性数据时经常遇到的问题有哪些？简述你收集软件可靠性数据时所遇到的具体问题及解决的方法。

1. 说明软件开发项目的基本情况以及自己承担的主要工作。
2. 当前的软件可靠性模型众多，但并没有一个最好的或者可以适用所有软件系统的软件可靠性模型，因此对于不同的软件系统，出于不同的可靠性分析目的，需要选择合适的软件可靠性模型。

常见的 10 类软件可靠性模型有种子法模型、失效率类模型、曲线拟合类模型、可靠性增长模型、程序结构分析模型、输入域分类模型、执行路径分析方法模型、非齐次泊松过程模型、马尔可夫过程模型和贝叶斯分析模型。

软件可靠性模型的选择主要需要考虑以下 4 个方面：

(1) 模型假设的适用性：模型假设是可靠性模型的基础，模型假设需要符合软件系统的现有状况，在软件系统中与假设冲突的因素达到几乎不存在的程度。往往一个模型的假设有很多，需要在选择模型时对每一条假设进行分析，评估现有软件系统中不符合假设的因素对可靠性评价有多大影响，以确定模型是否符合软件系统的可靠性评价工作。

(2) 模型预测的能力与质量：预测的能力和质量是指模型根据现在和历史的可靠性数据，预测将来的可靠性和失效概率的能力，以及预测结果的准确程度。因此，应尽可能选择比较成熟的、应用较广的模型。

(3) 模型输出值能否满足可靠性评价需求：根据可靠性测试目的来确定哪些模型的输出

值满足可靠性评价需求。重要的可靠性定量指标包括：当前可靠度、平均无失效时间、故障密度、期望达到规定可靠性目标的日期、达到规定可靠性目标的成本要求等。

(4) 模型使用的简便性：模型使用的数据在软件系统中易于收集；模型应该简单易懂；模型应该便于使用，最好有工具支持。

3. 软件可靠性数据的收集是一项艰巨而又烦琐的工作，受到许多潜在因素的影响和制约。常见的问题有：

- (1) 可靠性数据规范不一致，对软件进行度量的定义混乱；
- (2) 数据收集过程存在于整个软件生命周期，但由于成本等因素，其连续性往往不能保证；
- (3) 缺乏有效的技术和工具支持，难以进行自动分析；
- (4) 数据完整性不能保证，收集到的数据大多数是不完全的；
- (5) 数据质量和准确性不能保证；
- (6) 缺乏可靠性数据的交流与共享。

考生应叙述在项目中遇到了上述中的哪些问题。

可供采用的解决方法主要有：

- (1) 尽早确定可靠性模型，明确#要搜集的可靠性数据，确定涉及的术语、记录方法等；
- (2) 制定可实施的可靠性数据搜集计划，并指定专人负责。保证数据的收集和验证与软件开发过程同步进行；
- (3) 重视软件测试特别是可靠性测试产生的测试结果的整理和分析；
- (4) 尽可能地利用工具进行收集工作，例如利用数据库进行存储和分析等。