

【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)

在实时操作系统中，两个任务并发执行，一个任务要等待另一个任务发来消息，或建立某个条件后再向前执行，这种制约性合作关系被称为任务的(9)。

- (9) A. 同步                      B. 互斥                      C. 调度                      D. 执行

**【答案】A**

**【解析】**《本题考查实时操作系统基础知识》

由于资源共享与进程合作，并发执行的任务（进程）之间可能产生相互制约关系，这些制约关系可分为两类：竞争与协作。并发进程之间的竞争关系为互斥，并发进程之间的协作关系体现为同步。

同步是因合作进程之间协调彼此的工作而控制 S 己的执行速度，即因相互合作，相互等待而产生的制约关系。而互斥是进程之间竞争临界资源而禁止两个 y 上的进程同时进入临界区所发生的制约关系。

题目中一个任务要等待另一个任务发来消息，或建立某个条件后再向前执行，显然体现的制约关系是任务的同步。

在嵌入式系统设计中，用来讲行 CPU 调试的常用接口是(10)。

- (10) A. PCI 接口                      B. USB 接口                      C. 网络接口                      D. JTAG 接口

**【答案】D**

**【解析】**《本题考查嵌入式系统应用基础知识。》

PCI 是一种局部总线标准，它是在 CPU 和原来的系统总线之间插入的一级总线，具体由一个桥接电路实现对这一层的管理，并实现上下之间的接口以协调数据的传送。

JTAG 是一个调试接口，用来供开发人员调试 CPU 的工作状态。JTAG 软件通过该接口控制 CPU 来调试 CPU 以及读写 Flash。

把应用程序中应用最频繁的那部分核心程序作为评价计算机性能的标准程序，称为(16)程序。(17)不是对 Web 服务器进行性能评估的主要指标。

- (16) A. 仿真测试                      B. 核心测试                      C. 基准测试                      D. 标准测试

- (17) A. 丢包率                      B. 最大并发连接数                      C. 响应延迟                      D. 吞吐量

**【答案】C A**

**【解析】**《本题考查性能评估的基础知识。》

把应用程序中应用最频繁的那部分核心程序作为评价计算机性能的标准程序，称为基准测试程序。作为承载 Web 应用的 Web 服务器，对其进行性能评估时，主要关注最大并发连接数、响应延迟、吞吐量等指标。相对来说，对个别数据的丢包率并不是很关心。

企业信息化涉及对企业管理理念的创新，按照市场发展的要求，对企业现有的管理流程重新整合，管理核心从对(19)的管理，转向对(20)的管理，并延伸到对企业技术创新、工艺设计、产品设计、生产制造过程的管理，进而还要扩展到对(21)的管理乃至发展到电子商务。

(19) A. 人力资源和物资      B. 信息技术和知识      C. 财务和物料      D. 业务流程和数据

(20) A. 业务流程和数据      B. 企业信息系统和技术

C. 业务流程、数据和接口      D. 技术、物资和人力资源

(21) A. 客户关系和供应链      B. 信息技术和知识

C. 生产技术和信息技术      D. 信息采集、存储和共享

**【答案】** C   D   A

**【解析】** 本题考查企业信息化与电子商务的基础知识。

企业信息化涉及对企业管理理念的创新，管理流程的优化，管理团队的重组和管理手段的革新。管理创新是按照市场发展的要求，对企业现有的管理流程重新整合，从作为管理核心的财务、物料管理，转向技术、物资、人力资源的管理，并延伸到企业技术创新、工艺设计、产品设计、生产制造过程的管理，进而还要扩展到客户关系管理、供应链管理乃至发展到电子商务。

企业信息集成按照组织范围分为企业内部的信息集成和外部的信息集成。在企业内部的信息集成中，(22)实现了不同系统之间的互操作，使得不同系统之间能够实现数据和方法的共享；(23)实现了不同应用系统之间的连接、协调运作和信息共享。

(22) A. 技术平台集成      B. 数据集成      C. 应用系统集成      D. 业务过程集成

(23) A. 技术平台集成      B. 数据集成      C. 应用系统集成      D. 业务过程集成

**【答案】** C   D

**【解析】** 本题考查企业信息集成的基础知识。

企业信息集成是指企业在不同应用系统之间实现数据共享，即实现数据在不同数据格式和存储方式之间的转换、来源不同、形态不一、内容不等的信息资源进行系统分析、辨清正误、消除冗余、合并同类，进而产生具有统一数据形式的有价值信息的过程。企业信息集成是一个十分复杂的问题，按照组织范围来分，分为企业内部的信息集成和外部的信息集成两个方面。按集成内容，企业内部的信息集成一般可分为以下四个方面：技术平台集成，数据集成，应用系统集成和业务过程集成。其中，应用系统集成是实现不同系统之间的互操作，

使得不同应用系统之间能够实现数据和方法的共享；业务过程集成使得在不同应用系统中的流程能够无缝连接，实现流程的协调运作和流程信息的充分共享。

以下关于自顶向下开发方法的叙述中，正确的是\_(37)

- (37)A. 自顶向下过程因为单元测试而比较耗费时间  
B. 自顶向下过程可以更快地发现系统性能方面的问题  
C. 相对于自底向上方法，自顶向下方法可以更快地得到系统的演示原型  
D. 在自顶向下的设计中，如发现了一个错误，通常是因为底层模块没有满足其规格说明（因为高层模块已经被测试过了）

**【答案】C**

**【解析】**

自顶向下方法是一种决策策略。软件开发涉及作什么决策、如何决策和决策顺序等决策问题。

自顶向下方法在任何时刻所作的决定都是当时对整个设计影响最大的那些决定。如果把所有决定分组或者分级，那么决策顺序是首先作最高级的决定，然后依次地作较低级的决定。同级的决定则按照随机的顺序或者按别的方法。一个决策的级别是看它距离要达到的最终目的（因此是软件的实际实现）的远近程度。从问题本身来看，或是由外（用户所见的）向内（系统的实现）看，以距离实现近的决定为低级决定，远的为高级决定。

在这个自顶向下的过程中，一个复杂的问题（任务）被分解成若干个较小较简单的问题（子任务），并且一直继续下去，直到每个小问题（子任务）都简单到能够直接解决（实现）为止。

自顶向下方法的优点是：

- 可为企业或机构的重要决策和任务实现提供信息。
- 支持企业信息系统的整体性规划，并对系统的各子系统的协调和通信提供保证。
- 方法的实践有利于提高企业人员整体观察问题的能力，从而有利于寻找到改进企业组织的途径。

自顶向下方法的缺点是：

- 对系统分析和设计人员的要求较高。
- 开发周期长，系统复杂，一般属于一种高成本、大投资的工程。
- 对于大系统而言，自上而下的规划对于下层系统的实施往往缺乏约束力，

•从经济角度来看，很难说自顶向下的做法在经济上市合算的。

特定领域软件架构 (Domain Specific Software Architecture, DSSA) 是在一个特定应用领域中，为一组应用提供组织结构参考的标准软件体系结构。DSSA 通常是一个具有三个层次的系统模型，包括 (45) 环境、领域特定应用开发环境和应用执行环境，其中 (46) 主要在领域特定应用开发环境中工作。

(45) A. 领域需求                      B. 领域开发                      C. 领域执行                      D. 领域应用

(46) A. 操作员                          B. 领域架构师                      C. 应用工程师                      D. 程序员

**【答案】 B    C**

**【解析】 本题主要考查特定领域软件架构的基础知识。**

特定领域软件架构 (Domain Specific Software Architecture, DSSA) 是在一个特定应用领域中，为一组应用提供组织结构参考的标准软件体系结构。DSSA 通常是一个具有三个层次的系统模型，包括领域开发环境、领域特定应用开发环境和应用执行环境，其中应用工程师主要在领域特定应用开发环境中工作。

某公司采用基于架构的软件设计 (Architecture-Based Software Design, ABSD) 方法进行软件设计与开发。ABSD 方法有三个基础，分别是对系统进行功能分解、采用 (52) 实现质量属性与商业需求、采用软件模板设计软件结构。ABSD 方法主要包括架构需求等 6 个主要活动，其中 (53) 活动的目标是标识潜在的风险，及早发现架构设计中的缺陷和错误；(54) 活动针对用户的需求变化，修改应用架构，满足新的需求。

小王是该公司的一位新任架构师，在某项目中主要负责架构文档化方面的工作。小王 (55) 的做法不符合架构文档化的原则。架构文档化的主要输出结果是架构规格说明书和 (56)。

(52) A. 架构风格                      B. 设计模式                      C. 架构策略                      D. 架构描述

(53) A. 架构设计                      B. 架构实现                      C. 架构复审                      D. 架构演化

(54) A. 架构设计                      B. 架构实现                      C. 架构复审                      D. 架构演化

(55) A. 从使用者的角度书写文档                      B. 随时保证文档都是最新的  
C. 将文档分发给相关人员                      D. 针对不同背景的人员书写文档的方式不同

(56) A. 架构需求说明书                      B. 架构实现说明书  
C. 架构质量说明书                      D. 架构评审说明书

**【答案】 A    C    D    B    C**

**【解析】 本题主要考查采用基于架构的软件设计的基础知识与应用。**

基于架构的软件设计 (Architecture-Based Software Design, ABSD) 方法有三个基础，



分别是对系统进行功能分解、采用架构风格实现质量属性与商业需求、采用软件模板设计软件结构。ABSD 方法主要包括架构需求等 6 个主要活动，其中架构复审活动的目标是标识潜在的风险，及早发现架构设计中的缺陷和错误；架构演化活动针对用户的需求变化，修改应用架构，满足新的需求。

软件架构文档应该从使用者的角度进行书写，针对不同背景的人员采用不同的书写方式，并将文档分发给相关人员。架构文档要保持较新，但不要随时保证文档最新，要保持文档的稳定性。架构文档化的主要输出结果是架构规格说明书和架构质量说明书。

IETF 定义的集成服务 (IntServ) 把 Internet 服务分成了三种服务质量不同的类型，这三种服务不包括 (13)。

- (13) A. 保证质量的服务：对带宽、时延、抖动和丢包率提供定量的保证
- B. 尽力而为的服务：这是一般的 Internet 服务，不保证服务质量
- C. 负载受控的服务：提供类似于网络欠载时的服务，定性地提供
- D. 突发式服务：如果有富余的带宽，网络保证满足服务质量的需求

**【答案】D**

**【解析】**

IETF 集成服务 (IntServ) 工作组根据服务质量的不同，把 Internet 服务分成了三种类型：

- ① 保证质量的服务 (Guaranteed Services)：对带宽、时延、抖动和丢包率提供定量的保证；
- ② 负载受控的服务 (Controlled-load Services)：提供一种类似于网络欠载情况下的服务，这是一种定性的指标；
- ③ 尽力而为的服务 (Best-Effort)：这是 Internet 提供的一般服务，基本上无任何质量保证。

在实际应用中，用户通常依靠评价程序来测试系统的性能。以下评价程序中，(16) 的评测准确程度最低。事务处理性能委员会 (Transaction Processing Performance Council, TPC) 是制定商务应用基准程序 (benchmark) 标准规范、性能和价格度量，并管理测试结果发布的非营利组织，其发布的 TPC-C 是 (17) 的基准程序。

- (16) A. 核心程序      B. 真实程序      C. 合成基准程序      D. 小型基准程序
- (17) A. 决策支持      B. 在线事务处理      C. 企业信息服务      D. 联机分析处理

**【答案】C B**

**【解析】** 本题主要考查性能评估知识。

在大多数情况下，为测试新系统的性能，用户必须依靠评价程序来评价机器的性能。对于真实程序、核心程序、小型基准程序和合成基准程序来说，其评测程度依次递减。把应用程序中用的最多、最频繁的那部分核心程序作为评价计算机性能的标准程序，称为基准测试程序（Benchmark）。事务处理性能委员会（Transaction Processing Performance Council, TPC）是制定商务应用基准程序（Benchmark）标准规范、性能和价格度量，并管理测试结果发布的非营利组织，其发布的 TPC-C 是在线事务处理的基准程序，TPC-D 是决策支持的基准程序。

企业信息化一定要建立在企业战略规划基础之上，以企业战略规划为基础建立的企业管理模式是建立（18）的依据。

- (18) A. 企业战略数据模型                                      B. 企业业务运作模型  
C. 企业信息系统架构   D. 企业决策支持

**【答案】A**

**【解析】** 本题考查企业信息化基础知识。

企业信息化是指企业以业务流程的优化和重构为基础，在一定的深度和广度上利用计算机技术、网络技术和数据库技术，控制和集成化管理企业生产经营活动中的各种信息，实现企业内外部信息的共享和有效利用，以提高企业的经济效益和市场竞争力，这将涉及到企业的管理理念的创新，管理流程的优化，管理团队的重组和管理手段的革新。企业信息化一定要建立在企业战略规划的基础之上，以企业战略规划为基础建立的企业管理模式是建立企业战略数据模型的依据。

ERP 是对企业物流资源，资金流资源和信息流资源进行全面集成管理的管理信息系统。在 ERP 五个层次的计划中，（19）根据经营计划的生产目标制定，是对企业经营计划的细化；（20）说明了在一定时期内生产什么，生产多少和什么时候交货，它的编制是 ERP 的主要工作内容；（21）能够帮助企业尽早发现企业生产能力的瓶颈，为实现企业的生产任务提供能力方面的保障。

- (19) A. 销售管理计划    B. 生产预测计划    C. 生产计划大纲    D. 主生产计划  
(20) A. 经营计划            B. 车间作业计划    C. 物料需求计划    D. 主生产计划  
(21) A. 采购与库存计划                                      B. 能力需求计划  
C. 物料需求计划    D. 质量管理计划

**【答案】C D B**

**【解析】** 本题考查企业资源规划的基础知识。

ERP 是对企业物流、资金流和信息流资源进行全面集成管理的管理信息系统。在 ERP 五个层次的计划中，生产预测计划是对市场需求进行比较准确的预测，是经营计划、生产计划大纲和主生产计划编制的基础；销售管理计划是针对企业的销售部门的相关业务进行管理，属于最高层计划的范畴，是企业最重要的决策层计划之一；生产计划大纲根据经营计划的生产目标制定，是对企业经营计划的细化；主生产计划说明了在一定时期内生产什么，生产多少和什么时候交货，它的编制是 ERP 的主要工作内容；物料需求计划是对主生产计划的各个项目 0 所需的全部制造件和全部采购件的网路支持计划和时间进度计划；能力需求计划是对物料需求计划所需能力进行核算的一种计划管理方法，能够帮助企业尽早发现企业生产能力的瓶颈，为实现企业的生产任务提供能力帮面的保障。

集成平台是支持企业信息集成的支撑环境，包括硬件、软件、软件工具和系统。集成平台的基本功能中，(22) 实现不同数据库系统之间的数据交换、互操作、分布数据管理和共享信息模型定义；(23) 能够为应用提供数据交换和访问操作，使各种不同的系统能够相互协作。

(22) A. 数据通信服务      B. 信息集成服务      C. 应用集成服务      D. 操作集成服务

(23) A. 数据通信服务      B. 信息集成服务      C. 应用集成服务      D. 操作集成服务

**【答案】** B    C

**【解析】** 本题考查企业信息集成的基础知识。

企业集成平台是一个支持复杂信息环境下信息系统开发、集成、协同运行的软件支撑环境，包括硬件、软件、软件工具和系统。基本功能包括：①通信服务：提供分布环境下透明的同步/异步通信服务功能；②信息集成服务：为应用提供透明的信息访问服务，实现异种数据库系统之间数据的交换、互操作、分布数据管理和共享信息模型定义；③应用集成服务：通过高层应用编程接口来实现对相应应用程序的访问，能够为应用提供数据交换和访问操作，使各种不同的系统能够相互协作；④二次开发工具：是集成平台提供的一组帮助用户开发特定应用程序的支持工具；⑤平台运行管理工具：是企业集成平台的运行管理和控制模块。

商业智能是企业对商业数据的搜集、管理和分析的系统过程，主要技术包括 (24)。

(24) A. 数据仓库、联机分析和数据挖掘      B. 数据采集、数据清洗和数据挖掘

C. 联机分析、多维度分析和跨维度分析      D. 数据仓库、数据挖掘和业务优化重组



**【答案】A**

**【解析】**本题考查商业智能的基础知识。

商业智能通常被理解为将组织中现有的数据转化为知识，帮助组织做出明智的业务经营决策。商业智能的实现涉及到软件、硬件、咨询服务及应用，是对商业信息的搜集、管理和分析过程，目的是使企业的各级决策者获得知识或洞察力，促使他们做出对企业更有利的决策。商业智能一般由数据仓库、联机分析处理、数据挖掘、数据备份和恢复等部分组成。

在静态测试中，主要是对程序代码进行静态分析。“数据初始化、赋值或引用过程中的异常”属于静态分析中的(37)。

- (37) A. 控制流分析      B. 数据流分析      C. 接口分析      D. 表达式分析

**【答案】B**

**【解析】**本题考查静态测试方法的知识点。

静态测试是指被测试程序不在机器上运行，而采用人工检测和计算机辅助静态分析的手段对程序进行检测。静态测试包括对文档的静态测试和对代码的静态测试。对代码的静态测试包括

控制流分析、数据流分析、接口分析和表达式分析。

①控制流分析。控制流分析是指使用控制流程图检查被测程序控制结构的过程。例如，可检查被测程序是否存在没有使用的语句或子程序、是否调用并不存在的子程序，以及是否存在无法达到的语句等。

②数据流分析。数据流分析是指使用控制流程图分析数据各种异常情况的过程，包括数据初始化、赋值或引用过程中的异常。例如，引用未定义的变量、对以前未使用的变量再次赋值等程序差错或异常情况。

③接口分析。接口分析主要包括模块之间接口的一致性分析、模块与外部数据库及其他软件配置项之间的一致性分析、子程序和函数之间的接口一致性分析等。例如可以检查函数形参与实现的数量、顺序、类型和使用的一致性。

④表达式分析。表达式分析用于检查程序代码中的表达式错误。例如，括号不配对、数组引用越界、除数为零，以及浮点数变量比较时的误差等错误。

- (41) A. 如何通过多视图模型描述软件系统的架构  
B. 如何确定架构模型中有哪些元素构成  
C. 如何采用表格或用例映射保证转换的可追踪性

D. 如何通过模型转换技术，将高层架构模型逐步细化为细粒度架构模型

**【答案】C**

**【解析】**本题主要考查软件架构设计与生命周期的关系。

从本质上看，需求和软件架构设计面临的是不同的对象：一个是问题空间；另一个是解空间。保持两者的可追踪性和转换，一直是软件工程领域追求的目标。从软件需求模型向SA模型的转换主要关注两个问题：①如何根据需求模型构建软件架构模型；②如何保证模型转换的可追踪性。本题答案中A、B是软件架构设计阶段需要考虑的问题，D是软件架构实现阶段中需要考虑的问题。

在构件组装过程中需要检测并解决架构失配问题。其中(42)失配主要包括由于系统对构件基础设施、控制模型和数据模型的假设存在冲突引起的失配。(43)失配包括由于系统对构件交互协议、构件连接时数据格式的假设存在冲突引起的失配。

(42)A. 构件                      B. 模型                      C. 协议                      D. 连接器

(43)A. 构件                      B. 模型                      C. 协议                      D. 连接器

**【答案】A D**

**【解析】**本题主要考查构件组装过程知识。

在架构模型的指导下，可复用构件可以通过组装的方式在较高层次上实现系统，并能够提高系统实现的效率。在构件组装过程中需要检测并解决架构失配问题。其中构件失配主要包括由于系统对构件基础设施、控制模型和数据模型的假设存在冲突引起的失配。连接器失配包括由于系统对构件交互协议、构件连接时数据格式的假设存在冲突引起的失配。

“4+1”视图主要用于描述系统逻辑架构，最早由Philippe Kruchten于1995年提出。其中(44)视图用于描述对象模型，并说明系统应该为用户提供哪些服务。当采用面向对象的设计方法描述对象模型时，通常使用(45)表达类的内部属性和行为，以及类集合之间的交互关系；采用(46)定义对象的内部行为。

(44)A. 逻辑                      B. 过程                      C. 开发                      D. 物理

(45)A. 对象图                      B. 活动图                      C. 状态图                      D. 类图

(46)A. 对象图                      B. 活动图                      C. 状态图                      D. 类图

**【答案】A D C**

**【解析】**本题主要考查考生对“4+1”视图的即.解与掌握。

“4+1”视图是对逻辑架构进行描述，最早由Philippe Kruchten提出，他在1995年的

IEEE Software 上发表了题为 The 4+1 View Model of Architecture 的论文，引起了业界的极大关注，并最终被 RUP 采纳，现在已经成为架构设计的结构标准。“4+1”视图主要包括：

- ①逻辑视图(Logical View),设计的对象模型（使用面向对象的设计方法时）。
- ②过程视图（Process View），捕捉设计的并发和同步特征。
- ③物理视图（Physical View),描述了软件到硬件的映射，反映了分布式特性。
- ④开发视图(Development View),描述了在开发环境中软件的静态组织结构。
- ⑤架构的描述，即所做的各种决定，可以围绕着这四个视图来组织，然后由一些用例（Use Cases)或场景（Scenarios)来说明，从而形成了第五个视图。

当采用面向对象的设计方法描述对象模型时，通常使用类图表达类的内部属性和行为，以及类集合之间的交互关系；**采用状态图定义对象的内部行为。**

特定领域软件架构(Domain Specific Software Architecture, DSSA)是在一个特定应用领域中，为一组应用提供组织结构参考的标准软件体系结构。参加 DSSA 的人员可以划分为多种角色，其中 (47) 的任务是控制整个领域分析过程，进行知识获取，将获取的知识组织到领域模型中；(48) 的任务是根据领域模型和现有系统开发出 DSSA，并对 DSSA 的准确性和一致性进行验证。

(47) A. 领域专家      B. 领域分析者      C. 领域设计者      D. 领域实现者

(48) 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】B C**

**【解析】**

特定领域软件架构 (Domain Specific Software Architecture, DSSA) 以一个特定问题领域为对象，形成由领域参考模型、参考需求、参考架构等组成的开发基础架构，其目标是支持一个特定领域中多个应用的生成。DSSA 的基本活动包括领域分析、领域设计和领域实现。其中领域分析的主要目的是获得领域模型，领域模型描述领域中系统之间共同的需求，即领域需求；领域设计的主要目标是获得 DSSA，DSSA 描述领域模型中表示需求的解决方案；领域实现的主要目标是依据领域模型和 DSSA 开发和组织可重用信息，并对基础软件架构进行实现。参加 DSSA 的人员可以划分为多种角色，其中领域分析者的任务是控制整个领域分析过程，进行知识获取，将获取的知识组织到领域模型中；领域设计者的任务是根据领域模型和现有系统开发出 DSSA，并对 DSSA 的准确性和一致性进行验证。

软件架构风格描述某一特定领域中的系统组织方式和惯用模式，反映了领域中众多系统

所共有的(51)特征。对于语音识别、知识推理等问题复杂、解空间很大、求解过程不确定的这一类软件系统。通常会采用(52)架构风格。

(51)A. 语法和语义      B. 结构和语义      C. 静态和动态      D. 行为和约束

(52)A. 管道-过滤器      B. 解释器      C. 黑板      D. 过程控制

**【答案】B C**

**【解析】**

软件架构风格描述某一特定领域中的系统组织方式和惯用模式，反映了领域中众多系统所共有的结构和语义两个方面的特征。对于语音识别、知识推理等问题复杂、解空间很大、求解过程不确定的这一类软件系统，通常会采用黑板架构风格，以知识为中心进行分析与推理。

在对一个软件系统的架构进行设计与确认之后，需要进行架构复审。架构复审的目的是为了标识潜在的风险，及早发现架构设计中的缺陷和错误。在架构复审过程中，主要由(53)决定架构是否满足需求、质量需求是否在设计中得到体现。

(53)A. 系统分析师与架构师      B. 用户代表与领域专家

C. 系统所有者与项目经理      D. 系统开发与测试人员

**【答案】B**

**【解析】**

在对一个软件系统的架构进行设计与确认之后，需要进行架构复审。架构复审的目的是为了标识潜在的风险，及早发现架构设计中的缺陷和错误。在架构复审过程中，主要由用户代表与领域专家决定架构是否满足需求、质量需求是否在设计中得到体现。

识别风险、非风险、敏感点和权衡点是进行软件架构评估的重要过程。“改变业务数据编码方式会对系统的性能和安全性产生影响”是对(60)的描述，“假设用户请求的频率为每秒1个，业务处理时间小于30毫秒，则将请求响应时间设定为1秒钟是可以接受的”是对(61)的描述。

(60)A. 风险点      B. 非风险      C. 敏感点      D. 权衡点

(61)A. 风险点      B. 非风险      C. 敏感点      D. 权衡点

**【答案】D B**

**【解析】**本题主要考查考生对风险、非风险、敏感点和权衡点等重要评估概念的掌握和理解。

风险是某个存在问题的架构设计决策，可能会导致问题；非风险与风险相对，是良好的架构设计决策；

敏感点是一个或多个构件的特性；权衡点是影响多个质量属性的特性，是多个质量属性的敏感点。根据上述定义，可以看出“改变业务数据编码方式会对系统的性能和安全性产生影响”是对权衡点的描述，“假设用户请求的频率为每秒 1 个，业务处理时间小于 30 毫秒，则将请求响应时间设定为 1 秒钟是可以接受的”是对非风险的描述。

体系结构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)是一种常见的系统架构评估框架，该框架主要关注系统的\_(62)，针对性能\_(63)安全性和可修改性，在系统开发之前进行分析、评价与折中。

(62)A. 架构视图                      B. 架构描述                      C. 需求说明                      D. 需求建模

(63)A. 架构视图                      B. 架构描述                      C. 架构设计                      D. 架构实现

**【答案】C B**

**【解析】**

架构权衡分析方法是一种系统架构评估方法，主要在系统开发之前，针对性能、可用性、安全性和可修改性等质量属性进行评价和折中。ATAM可以分为4个主要的活动阶段，包括需求收集、架构视图描述、属性模型构造和分析、架构决策与折中，整个评估过程强调以属性作为架构评估的核心概念。题目中提到“某软件公司采用 ATAM 进行软件架构评估，在评估过程中识别出了多个关于质量属性的描述。其中，系统在进行文件保存操作时，应该与 Windows 系统的操作方式保持一致。”与用户所熟悉的操作方式，操作界面保持一致，这是一种减轻用户记忆负担，降低学习成本的做法，这有利于提高系统的易用性。“系统应该提供一个开放的 API 接口，支持远程对系统的行为进行控制与调试”，在此处，我们注意到描述的核心落在“支持远程对系统的行为进行控制与调试”上了，而调试是在测试之后精确定位系统错误的一种机制，所以这种做法有利于提高系统的可测试性。最后的两空也是考概念：在识别出上述描述后，通常采用效用树对质量属性的描述进行刻画与排序。在评估过程中，权衡点是一个会影响多个质量属性的架构设计决策。

在嵌入式操作系统中，板级支持包 BSP 作为对硬件的抽象，实现了\_(9)。

(9)A. 硬件无关性，操作系统无关性                      B. 硬件有关性，操作系统有关性  
C. 硬件无关性，操作系统有关性                      D. 硬件有关性，操作系统无关性

**【答案】D**



**【解析】** 本题考查嵌入式系统的基础知识。

在嵌入式系统中，板级支持包 Board Support Package (简称 BSP) 是对硬件抽象层的实现，是介于主板的硬件和操作系统驱动程序之间的一层，为整个软件系统提供底层硬件支持，是介于底层硬件和上层软件之间的底层软件开发包，它主要的功能是给上层提供统一接口，同时屏蔽各种硬件底层的差异，以及提供操作系统的驱动及硬件驱动。简单地说，就是 BSP 包含了所有与硬件有关的代码，为操作系统提供了硬件平台无关性。

假如有 3 块容量是 80G 的硬盘做 RAID 5 阵列，则这个 RAID 5 的容量是 (13)；而如果有 2 块 80G 的盘和 1 块 40G 的盘，此时 RAID 5 的容量是 (14)。

(13) A. 240G

B. 160G

C. 80G

D. 40G

(14) A. 40G

B. 80G

C. 160G

D. 200G

**【答案】** B B

**【解析】** 本题考查 RAID 的基础概念。

RAID 是英文 Redundant Array of Independent Disks 的缩写，中文简称为独立冗余磁盘阵列。简单地说，RAID 是一种把多块独立的硬盘 (物理硬盘) 按不同的方式组合起来形成一个硬盘组 (逻辑硬盘)，从而提供比单个硬盘更高的存储性能和提供数据备份技术。组成磁盘阵列的不同方式称为 RAID 级别 (RAID Levels)。在用户看起来，组成的磁盘组就像是一个硬盘，用户可以对它进行分区，格式化等。总之，对磁盘阵列的操作与单个硬盘一模一样。不同的是，磁盘阵列的存储速度要比单个硬盘高很多，而且可以提供自动数据备份。数据备份的功能是在用户数据一旦发生损坏后，利用备份信息可以使损坏数据得以恢复，从而保障了用户数据的安全性。RAID 技术分为几种不同的等级，分别可以提供不同的速度，安全性和性价比。根据实际情况选择适当的 RAID 级别可以满足用户对存储系统可用性、性能和容量的要求。常用的 RAID 级别有以下几种：NRAID，JBOD，RAID0，RAID1，RAID1+0，RAID3，RAID5 等。目前经常使用的是 RAID5 和 RAID (1+0)。如果使用物理硬盘容量不相等的硬盘做 RAID，那么创建的 RAID 阵列的总容量为较小的硬盘的计算方式。

RAID5 的存储机制是两块存数据，一块存另外两块硬盘的交易校验结果。RAID5 的建立后，坏掉一块硬盘，可以通过另外两块硬盘的数据算出第三块的，所以至少要 3 块。RAID5 是一种旋转奇偶校验独立存取的阵列方式，它与 RAID3，RAID4 不同的是没有固定的校验盘，而是按某种规则把奇偶校验信息均匀地分布在阵列所属的硬盘上，所以在每块硬盘上，既有数据信息也有校验信息。这一改变解决了争用校验盘的问题，使得在同一组内并发进行多个写操作。所以 RAID5 既适用于大数据量的操作，也适用于各种事务处理，它是一种快速、

大容量和容错分布合理的磁盘阵列。当有 N 块阵列盘时，用户空间为 N-1 块盘容量。

根据以上原理，共有 3 块 80G 的硬盘做 RAID5，则总容量为  $(3-1) \times 80=160G$ ；如果有 2 块 80G 的盘和 1 块 40G 的盘，则以较小的盘的容量为计算方式，总容量为  $(3-1) \times 40=80G$ 。

以下关于 IPv6 的论述中，正确的是(15)。

- (15) A. IPv6 数据包的首部比 IPv4 复杂    B. IPv6 的地址分为单播、广播和任意播 3 种  
C. IPv6 的地址长度为 128 比特        D. 每个主机拥有唯一的 IPv6 地址

**【答案】C**

**【解析】**

IPv6 地址增加到 128 位，并且能够支持多级地址层次；地址自动配置功能简化了网络地址的管理；在组播地址中增加了范围字段，改进了组播路由的可伸缩性；增加的任意播地址比 IPv4 中的广播地址更加实用。

IPv6 地址是一个或一组接口的标识符。IPv6 地址被分配到接口，而不是分配给结点。

IPv6 地址有三种类型：

- (1) 单播(Unicast)地址
- (2) 任意播(AnyCast)地址
- (3) 组播(Multicast)地址

在 IPv6 地址中，任何全“0”和全“1”字段都是合法的，除非特别排除的之外。特别是前缀可以包含“0”值字段，也可以用“0”作为终结字段。一个接口可以被赋予任何类型的多个地址(单播、任意播、组播)或地址范围。

与 IPv4 相比，IPv6 首部有下列改进：

- 分组头格式得到简化：IPv4 头中的很多字段被丢弃，IPv6 头中字段的数量从 12 个降到了 8 个，中间路由器必须处理的字段从 6 个降到了 4 个，这样就简化了路由器的处理过程，提高了路由选择的效率。

- 改进了对分组头部选项的支持：与 IPv4 不同，路由选项不再集成在分组头中，而是把扩展头作为任选项处理，仅在需要时才插入到 IPv6 头与负载之间。这种方式使得分组头的处理更灵活，也更流畅。以后如果需要，还可以很方便地定义新的扩展功能。

- 提供了流标记能力：IPv6 增加了流标记，可以按照发送端的要求对某些分组进行特别的处理，从而提供了特别的服务质量支持，简化了对多媒体信息的处理，可以更好地传送具有实时需求的应用数据。

供应链中的信息流覆盖了从供应商、制造商到分销商，再到零售商等供应链中的所有环节，其信息流分为需求信息流和供应信息流，(18)属于需求信息流，(19)属于供应信息流。

(18)A. 库存记录      B. 生产计划      C. 商品入库单      D. 提货发运单

(19)A. 客户订单      B. 采购合同      C. 完工报告单      D. 销售报告

**【答案】B C**

**【解析】**本题考查信息化方面的基础知识。

供应链中的信息流覆盖了从供应商、制造商到分销商，再到零售商等供应链中的所有环节，其信息流分为需求信息流和供应信息流，这是两个不同流向的信息流。当需求信息(如客户订单、生产计划和采购合同等)从需方向供方流动时，便引发物流。同时，供应信息(如入库单、完工报告单、库存记录、可供销售量和提货发运单等)又同物料一起沿着供应链从供方向需方流动。

电子商务系统中参与电子商务活动的实体包括(21)。

(21)A. 客户、商户、银行和认证中心      B. 客户、银行、商户和政府机构

C. 客户、商户、银行和物流企业      D. 客户、商户、政府和物流企业

**【答案】A**

**【解析】**本题考查电子商务方面的基础知识。

电子商务分五个方面，即电子商情广告、电子选购与交易、电子交易凭证. 的交换、电子支付与结算，以及网上售后服务等。参与电子商务的实体有4类：客户(个人消费者或集团购买)、商户(包括销售商、制造商和储运商)、银行(包括发行和收单行)及认证中心。

商业智能系统的处理过程包括四个主要阶段：数据预处理通过(22)实现企业原始数据的初步整合；建立数据仓库是后续数据处理的基础；数据分析是体现系统智能的关键，主要采用(23)和(24)技术，前者能够实现数据的上卷、下钻和旋转分析，后者利用隐藏的知识，通过建立分析模型预测企业未来发展趋势；数据展现主要完成数据处理结果的可视化。

(22)A. 数据映射和关联      B. 数据集市和数据立方体

C. 数据抽取、转换和装载      D. 数据清洗和数据集成

(23)A. 知识库      B. 数据挖掘      C. 联机事务处理      D. 联机分析处理

(24)A. 知识库      B. 数据挖掘      C. 联机事务处理      D. 联机分析处理

**【答案】C D B**

**【解析】**本题考查商业智能方面的基础知识。

商业智能系统的处理过程包括数据预处理、建立数据仓库、数据分析及数据展现 4 个主要阶段。数据预处理是整合企业原始数据的第一步，包括数据的抽取、转换和装载三个过程。建立数据仓库则是处理海量数据的基础。数据分析是体现系统智能的关键，一般采用 OLAP 和数据挖掘技术。联机分析处理不仅进行数据汇总/聚集，同时还提供切片、切块、下钻、上卷和旋转等数据分析功能，用户可以方便地对海量数据进行多维分析。数据挖掘的目标则是挖掘数据背后隐藏的知识，通过关联分析、聚类和分类等方法建立分析模型，预测企业未来发展趋势和将要面临的问题。在海量数据和分析手段增多的情况下，数据展现则主要保障系统分析结果的可视化。

关于项目范围管理描述，正确的是 (25)。

- (25) A. 项目范围是指信息系统产品或者服务所包含的功能
- B. 项目范围描述是产品范围说明书的重要组成部分
- C. 项目范围定义是信息系统要求的度量
- D. 项目范围定义是生产项目计划的基础

**【答案】D**

**【解析】** 本题考查软件项目范围管理方面的基础知识。

项目范围是为了达到项目目标，为了交付具有某种特制的产品和服务，项目所规定要做的。在信息系统项目中，产品范围是指信息系统产品或者服务所应该包含的功能，项目范围是指为了能够交付信息系统项目所必须做的工作。产品范围是项目范围的基础，产品的范围定义是信息系统要求的度量，而项目范围的定义是生产项目计划的基础。产品范围描述是项目范围说明书的重要组成部分。

处理流程设计是系统设计的重要内容。以下关于处理流程设计工具的叙述中，不正确的是 (30)。

- (30) A. 程序流程图 (PFD) 用于描述系统中每个模块的输入，输出和数据加工
- B. N-S 图容易表示嵌套关系和层次关系，并具有强烈的结构化特征
- C. IPO 图的主体是处理过程说明，可以采用流程图、判定树/表等进行描述
- D. 问题分析图 (PAD) 包含 5 种基本控制结构，并允许递归使用

**【答案】A**

**【解析】**

在处理流程设计过程中，为了更清晰地表达过程规则说明，陆续出现了一些用于表示处理流程的工具，这些工具包括三类：图形工具、表格工具和语言工具。其中常见的图形工具包括程序流程图、IPO图、盒图、问题分析图、判定树，表格工具包括判定表，语言工具包括过程设计语言等。

程序流程图(Program Flow Diagram, PFD)用一些图框表示各种操作，它独立于任何一种程序设计语言，比较直观、清晰，易于学习掌握。流程图中只能包括5种基本控制结构：顺序型、选择型、WHILE循环型(当型循环)、UNTIL循环型(直到型循环)和多分支选择型。

IPO图是由IBM公司发起并逐步完善的一种流程描述工具，其主体是处理过程说明，可以采用流程图、判定树、判定表、盒图、问题分析图或过程设计语言来进行描述。IPO图中的输入、输出与功能模块、文件及系统外部项都需要通过数据字典来描述，同时需要为其中的某些元素添加注释。

N-S图与PFD类似，也包括5种控制结构，分别是顺序型、选择型、WHILE循环型(当型循环)、UNTIL循环型(直到型循环)和多分支选择型，任何一个N-S图都是这5种基本控制结构相互组合与嵌套的结果。在N-S图中，过程的作用域明确；它没有箭头，不能随意转移控制；而且容易表示嵌套关系和层次关系；并具有强烈的结构化特征。但是当问题很复杂时，N-S图可能很大。

问题分析图(Problem Analysis Diagram, PAD)是继PFD和N-S图之后，又一种描述详细设计的工具。PAD也包含5种基本控制结构，并允许递归使用。

过程设计语言(Process Design Language, PDL)也称为结构化语言或伪代码(pseudocode)，它是一种混合语言，采用自然语言的词汇和结构化程序设计语言的语法，用于描述处理过程怎么做，类似于编程语言。过程设计语言用于描述模块中算法和加工逻辑的具体细节，以便在开发人员之间比较精确地进行交流。

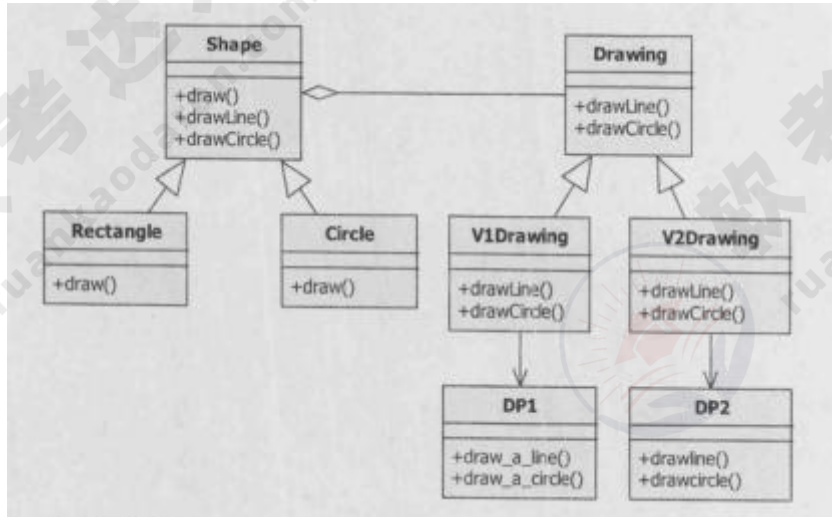
对于具有多个互相联系的条件和可能产生多种结果的问题，用结构化语言描述则显得不够直观和紧凑，这时可以用以清楚、简明为特征的判定表(Decision Table)来描述。判定表采用表格形式来表达逻辑判断问题，表格分成4个部分，左上部分为条件说明，左下部分为行动说明，右上部分为各种条件的组合说明，右下部分为各条件组合下相应的行动。

判定树(Decision Tree)也是用来表示逻辑判断问题的一种常用的图形工具，它用树来表达不同条件下的不同处理流程，比语言、表格的方式更为直观。判定树的左侧(称为树根)为加工名，中间是各种条件，所有的行动都列于最右侧。

某软件公司欲开发一个绘图软件，要求使用不同的绘图程序绘制不同的图形。在明确用



户需求后，该公司的架构师决定采用 Bridge 模式实现该软件，并设计 UML 类图如下图所示。图中与 Bridge 模式中的“Abstraction”角色相对应的类是(32)，与“Implementor”角色相对应的类是(33)。



(32) A. Shape                      B. Drawing                      C. Rectangle                      D. V2Drawing

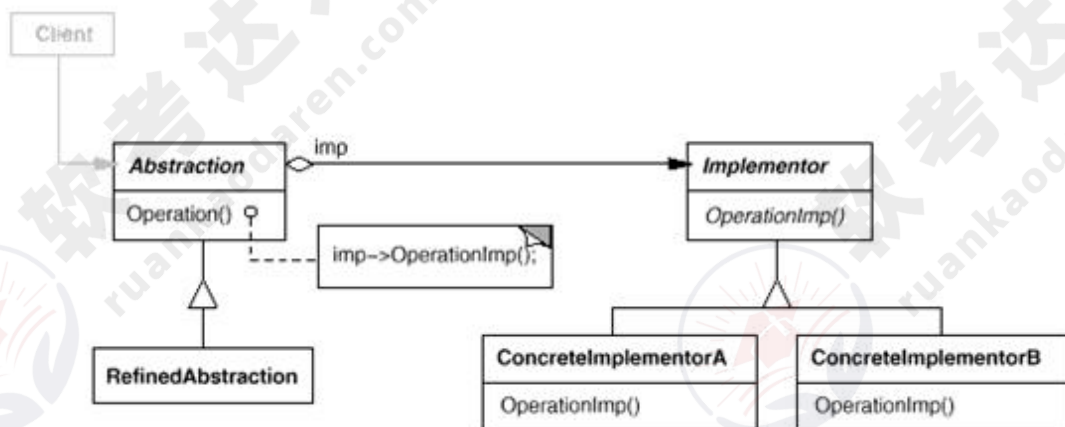
(33) A. Shape                      B. Drawing                      C. Rectangle                      D. V2Drawing

**【答案】** A    B

**【解析】**

桥接模式将抽象部分与它的实现部分分离，使它们都可以独立地变化。它是一种对象结构型模式，又称为柄体(Handle and Body)模式或接口(Interface)模式。桥接模式类似于多重继承方案，但是多重继承方案往往违背了类的单一职责原则，其复用性比较差，桥接模式是比多重继承方案更好的解决方法。

桥接模式的结构如下图所示，其中：



• **Abstraction** 定义抽象类的接口；维护一个指向 **Implementor** 类型对象的指针。

- RefinedAbstraction 扩充由 Abstraction 定义的接口。
  - Implementor 定义实现类的接口，该接口不一定要与 Abstraction 的接口完全一致；事实上这两个接口可以完全不同。一般来说，Implementor 接口仅提供基本操作，而 Abstraction 则定义了基于这些基本操作的较高层次的操作。
  - ConcretelImplementor 实现 Implementor 接口并定义它的具体实现。
- 图中与 Bridge 模式中的“Abstraction”角色相对应的类是 Shape，与“Implementor”角色相对应的类是 Drawing。

RUP 强调采用 (34) 的方式来开发软件，这样做的好处是 (35)。

- (34) A. 原型和螺旋                      B. 螺旋和增量                      C. 迭代和增量                      D. 快速和迭代

- (35) A. 在软件开发的早期就可以对关键的，影响大的风险进行处理  
B. 可以避免需求的变更  
C. 能够非常快速地实现系统的所有需求  
D. 能够更好地控制软件的质量

**【答案】C A**

**【解析】**

RUP 将项目管理、业务建模、分析与设计等统一起来，贯穿整个开发过程。RUP 中的软件过程在时间上被分解为 4 个顺序的阶段，分别是初始阶段、细化阶段、构建阶段和移交阶段。每个阶段结束时都要安排一次技术评审，以确定这个阶段的目标是否已经满足。如果评审结果令人满意，就可以允许项目进入下一个阶段。可以看出，基于 RUP 的软件过程是一个迭代和增量的过程。通过初始、细化、构建和移交 4 个阶段就是一个开发周期，每次经过这 4 个阶段就会产生一代软件。除非产品退役，否则通过重复同样的 4 个阶段，产品将演化为下一代产品，但每一次的侧重点都将放在不同的阶段上。这样做的好处是在软件开发的早期就可以对关键的、影响大的风险进行处理。

架构描述语言 (Architecture Description Language, ADL) 是一种为明确说明软件系统的概念架构和对这些概念架构建模提供功能的语言。ADL 主要包括以下组成部分：组件、组件接口、(43) 和架构配置。

- (43) A. 架构风格                      B. 架构实现                      C. 连接件                      D. 组件约束

**【答案】C**

**【解析】** 本题考查架构描述语言的理解与掌握。

架构描述语言 (Architecture Description Language, ADL) 是一种为明确说明软件系统的概念架构和对这些概念架构建模提供功能的语言。ADL 主要包括以下组成部分：组件、组件接口、连接件和架构配置。ADL 对连接件的重视成为区分 ADL 和其他建模语言的重要特征之一。

基于架构的软件开发 (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 (44) 描述软件架构，用 (45) 来描述需求。

- (44) A. 类图和序列图      B. 视角与视图      C. 构建和类图      D. 构建与功能
- (45) 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】** 本题考查基于架构的软件开发方法的基础知识。

根据定义，基于软件架构的开发 (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。

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。架构设计主要关注软件组件的结构、属性和 (51)，并通过多种 (52) 全面描述特定系统的架构。

- (51) A. 实现方式      B. 交互作用      C. 设计方案      D. 测试方式
- (52) A. 对象      B. 代码      C. 文档      D. 视图

**【答案】** B    D

**【解析】** 本题主要考查软件架构设计过程的基础知识。

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。架构设计主要关注软件组件的结构、属性和交互作用，并通过多种视图全面描述特定系统的架构。

特定领域软件架构 (Domain Specific Software Architecture, DSSA) 以一个特定问题领域为对象，形成由领域参考模型，参考需求，(53) 等组成的开发基础架构，支持一个特定领域中多个应用的生成。DSSA 的基本活动包括领域分析、领域设计和领域实现。其中

领域分析的主要目的是获得(54)，从而描述领域中系统之间共同的需求，即领域需求；领域设计的主要目标是获得(55)，从而描述领域模型中表示需求的解决方案；领域实现的主要目标是开发和组织可重用信息，并实现基础软件架构。

- (53) A. 参考设计                      B. 参考规约                      C. 参考架构                      D. 参考实现
- (54) A. 领域边界                      B. 领域信息                      C. 领域对象                      D. 领域模型
- (55) A. 特点领域软件需求                      B. 特定领域软件架构
- C. 特定领域软件设计模型                      D. 特定领域软件重用模型

**【答案】** C D B

**【解析】**

特定领域软件架构(Domain Specific Software Architecture, DSSA)以一个特定问题领域为对象，形成由领域参考模型、参考需求、参考架构等组成的开发基础架构，其目标是支持一个特定领域中多个应用的生成。DSSA的基本活动包括领域分析、领域设计和领域实现。其中领域分析的主要目的是获得领域模型，领域模型描述领域中系统之间共同的需求，即领域需求；领域设计的主要目标是获得 DSSA，DSSA 描述领域模型中表示需求的解决方案；领域实现的主要目标是依据领域模型和 DSSA 开发和组织可重用信息，并对基础软件架构进行实现。

架构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)是在基于场景的架构分析方法(Scenarios-based Architecture Analysis Method, SAAM)基础之上发展起来的，主要包括场景和需求收集、(62)，属性模型构造和分析，属性模型折中等四个阶段。ATAM方法要求在系统开发之前，首先对这些质量属性进行(63)和折中。

- (62) A. 架构视图和场景实现                      B. 架构风格和场景分析
- C. 架构设计和目标分析                      D. 架构描述和需求评估
- (63) A. 设计                      B. 实现                      C. 测试                      D. 评价

**【答案】** A D

**【解析】**本题主要考查考生对架构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)的掌握和理解。

ATAM是在基于场景的架构分析方法(Scenarios-based Architecture Analysis Method, SAAM)基础之上发展起来的，主要包括场景和需求收集、架构视图和场景实现、属性模型构造和分析、属性模型折中等4个阶段。ATAM方法要求在系统开发之前，首先对这些质量属性进行评价和折中。

在嵌入式系统的存储部件中，存取速度最快的是(1)。

- (1) A. 内存                      B. 寄存器组                      C. Flash                      D. Cache

**【答案】B**

**【解析】**本题考查嵌入式系统存储结构的基础知识

嵌入式系统的存储结构采用分级的方法来设计，从而使得整个存储系统分为四级，即寄存器组、高速缓冲(Cache)、内存(包括 flash)和外存，它们在存取速度上依次递减，而在存储容量上逐级递增。

IETF 定义的区分服务(DiffServ)模型要求每个 IP 分组都要根据 IPv4 协议头中的(13)字段加上一个 DS 码点，然后内部路由器根据 DS 码点的值对分组进行调度和转发。

- (13) A. 数据报生存期              B. 服务类型              C. 段偏置值              D. 源地址

**【答案】B**

**【解析】**

区分服务是为解决服务质量问题在网络上将用户发送的数据流按照它对服务质量的要求划分等级的一种协议。

区分服务将具有相同特性的若干业务流汇聚起来，为整个汇聚流提供服务，而不是面向单个业务流来提供服务。每个 IP 分组都要根据其 QoS 需求打上一个标记，这种标记称为 DS 码点，可以利用 IPv4 协议头中的服务类型字段，或者 IPv6 协议头中的通信类别字段来实现，这样就维持了现有的 IP 分组格式不变。

把应用程序中应用最频繁的那部分核心程序作为评价计算机性能的标准程序，称为(16)程序。(17)不是对 Web 服务器进行性能评估的主要指标。

- (16) A. 仿真测试              B. 核心测试              C. 基准测试              D. 标准测试

- (17) A. 丢包率              B. 最大并发连接数              C. 响应延迟              D. 吞吐量

**【答案】C A**

**【解析】**

(16) 本题考查基本概念，应用最频繁的那部分核心程序作为评价计算机性能的标准程序，称为基准测试程序。

(17) 丢包率不应作为 Web 服务器性能评估的主要指标，而应作为网络的相关指标。

电子政务是对现有的政府形态的一种改造，利用信息技术和其他相关技术，将其管理和



服务职能进行集成，在网络上实现政府组织结构和 workflows 优化重组。与电子政务相关的行为主体有三个，即政府、(18)及居民。国家和地方人口信息的采集、处理和利用，属于(19)的电子政务活动。

(18)A. 部门      B. 企（事）业单位      C. 管理机构      D. 行政机关

(19)A. 政府对政府      B. 政府对居民      C. 居民对居民      D. 居民对政府

**【答案】B** B项目的成本管理中，(22)将总的成本估算分配到各项活动和工作包上，来建立一个成本的基线。

(22)A. 成本估算      B. 成本预算      C. 成本跟踪      D. 成本控制

**【答案】B**

**【解析】** 本题考查成本预算的定义。

在软件系统工具中，版本控制工具属于(29)，软件评价工具属于(30)。

(29)A. 软件开发工具      B. 软件维护工具  
C. 编码与排错工具      D. 软件管理和软件支持工具

(30)A. 逆向工程工具      B. 开发信息库工具  
C. 编码与排错工具      D. 软件管理和软件支持工具

**【答案】B D**

**【解析】**

软件系统工具的种类繁多，很难有统一的分类方法。通常可以按软件过程活动将软件工具分为软件开发工具、软件维护工具、软件管理和软件支持工具。

软件开发工具：需求分析工具、设计工具、编码与排错工具。

软件维护工具：版本控制工具、文档分析工具、开发信息库工具、逆向工程工具、再工程工具。

软件管理和软件支持工具：项目管理工具、配置管理工具、软件评价工具、软件开发工具的评价和选择。

面向对象的分析模型主要由(31)、用例与用例图、领域概念模型构成；设计模型则包含以包图表示的软件体系结构图、以交互图表示的(32)、完整精确的类图、针对复杂对象的状态图和描述流程化处理过程的(33)等。

(31)A. 业务活动图      B. 顶层架构图      C. 数据流模型      D. 实体联系图

(32)A. 功能分解图      B. 时序关系图      C. 用例实现图      D. 软件部署图

- (33)A. 序列图      B. 协作图      C. 流程图      D. 活动图

**【答案】B C D**

**【解析】**本题考查的是教程“4.4.2 面向对象的分析设计”的内容。

面向对象的分析模型主要由顶层架构图、用例与用例图、领域概念模型构成。

设计模型则包含以包图表示的软件体系结构图、以交互图表示的用例实现图、完整精确的类图、针对复杂对象的状态图和用以描述流程化处理过程的活动图等。

CORBA 构件模型中，(36)的作用是在底层传输平台与接收调用并返回结果的对象实现之间进行协调，(37)是最终完成客户请求的服务对象实现。

- (36)A. 伺服对象激活器      B. 适配器激活器  
C. 伺服对象定位器      D. 可移植对象适配器 POA  
(37)A. CORBA 对象      B. 分布式对象标识  
C. 伺服对象 Servant      D. 活动对象映射表

**【答案】D C**

**【解析】**

POA 是对象实现与 ORB 其它组件之间的中介，它将客户请求传送到伺服对象，按需创建子 POA，提供管理伺服对象的策略。

CORBA 对象可看作是一个具有对象标识、对象接口及对象实现的抽象实体。之所以称为抽象的，是因为并没有硬性规定 CORBA 对象的实现机制。由于独立于程序设计语言和特定 ORB 产品，一个 CORBA 对象的引用又称可互操作的对象引用（Interoperable Object Reference）。从客户程序的角度看，IOR 中包含了对对象的标识、接口类型及其他信息以查找对象实现。

伺服对象（servant）是指具体程序设计语言的对象或实体，通常存在于一个服务程序进程之中。

客户程序通过对象引用发出的请求经过 ORB 担当中介角色，转换为对特定的伺服对象的调用。在一个 CORBA 对象的生命期中，它可能与多个伺服对象相关联，因而对该对象的请求可能被发送到不同的伺服对象。

对象标识（Object ID）是一个用于在 POA 中标识一个 CORBA 对象的字符串。

它既可由程序员指派，也可由对象适配器自动分配，这两种方式都要求对象标识在创建它的对象适配器中必须具有唯一性。

关于构件的描述，正确的是(38)。

- (38)A. 构件包含了一组需要同时部署的原子构件  
B. 构件可以单独部署，原子构件不能被单独部署  
C. 一个原子构件可以同时多个构件家族中共享  
D. 一个模块可以看作带有单独资源的原子构件

**【答案】A**

**【解析】**

构件是一组通常需要同时部署的原子构件。构件和原子构件之间的区别在于，大多数原子构件永远都不会被单独部署，尽管它们可以被单独部署。相反，大多数原子构件都属于一个构件家族，一次部署往往涉及整个家族。

一个原子构件是一个模块和一组资源。

原子构件是部署、版本控制和替换的基本单位。原子构件通常成组地部署，但是它也能够被单独部署。一个模块是不带单独资源的原子构件（在这个严格定义下，Java 包不是模块——在 Java 中部署的原子单元是类文件。一个单独的包被编译成多个单独的类文件——每个公共类都有一个）。

模块是一组类和可能的非面向对象的结构体，比如过程或者函数。

基于 JavaEE 平台的基础功能服务构建应用系统时，(41) 可用来集成遗产系统。

- (41)A. JDBC、JCA 和 Java IDL                      B. JDBC、JCA 和 JMS  
C. JDBC、JMS 和 Java IDL                      D. JCA、JMS 和 Java IDL

**【答案】D**

**【解析】**

JDBC (Java Data Base Connectivity, java 数据库连接) 是一种用于执行 SQL 语句的 Java API, 可以为多种关系数据库提供统一访问, 它由一组用 Java 语言编写的类和接口组成。JDBC 提供了一种基准, 据此可以构建更高级的工具和接口, 使数据库开发人员能够编写数据库应用程序。

JCA 标准化连接器是由 J2EE 1.3 首先提出的, 它位于 J2EE 应用服务器和企业信息系统 (EIS) 之间, 比如数据库管理、企业资源规划 (ERP)、企业资产管理 (EAM) 和客户关系管理 (CRM) 系统。不是用 Java 开发的企业应用或者在 J2EE 框架内的应用都可以通过 JCA 连接。JCA 是在 javax.resource 包和它的子包 (cci, spi 和 spi.security) 中定义的。

(JCA 的缩写形式也用于 Java cryptography API 的缩写。)

JMS 是 Java 对消息系统的访问机制，但它本身并不实现消息。JMS 支持点对点分发的消息队列，也支持多个目标订阅的消息主题。当消息发布给一个主题的适合，消息就会发送给所有那个主题的订阅者。JMS 支持各种消息类型（二进制、流、名-值表、序列化的对象和文本）。通过声明与 SQL 的 WHERE 相近的句段，可以建立消息的过滤器。

Java IDL 即 idltojava 编译器就是一个 ORB，可用来在 Java 语言中定义、实现和访问 CORBA 对象。Java IDL 支持的是一个瞬间的 CORBA 对象，即在对象服务器处理过程中有效。实际上，Java IDL 的 ORB 是一个类库而已，并不是一个完整的平台软件，但它对 Java IDL 应用系统和其他 CORBA 应用系统之间提供了很好的底层通信支持，实现了 OMG 定义的 ORB 基本功能。

软件集成测试将已通过单元测试的模块集成在一起，主要测试模块之间的协作性。从组装策略而言，可以分为(42)。集成测试计划通常是在(43)阶段完成，集成测试一般采用黑盒测试方法。

- (42) A. 批量式组装和增量式组装      B. 自顶向下和自底向上组装  
C. 一次性组装和增量式组装      D. 整体性组装和混合式组装

- (43) A. 软件方案建议    B. 软件概要设计    C. 软件详细设计    D. 软件模块集成

**【答案】C B**

**【解析】**

(42)集成测试可以分为一次性组装和增量式组装，增量式组装测试效果更好。

(43)集成测试计划一般在概要设计阶段完成。

(44)架构风格可以概括为通过连接件绑定在一起按照一组规则运作的并行构件。

- (44) A. C2      B. 黑板系统      C. 规则系统      D. 虚拟机

**【答案】A**

**【解析】**

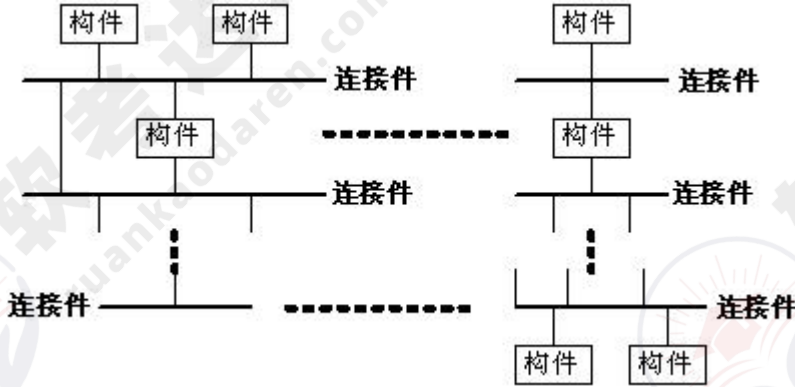
C2 体系结构风格可以概括为：通过连接件绑定在一起的按照一组规则运作的并行构件网络。C2 风格中的系统组织规则如下：

- (1) 系统中的构件和连接件都有一个顶部和一个底部；
- (2) 构件的顶部应连接到某连接件的底部，构件的底部则应连接到某连接件的顶部，而

构件与构件之间的直接连接是不允许的；

(3) 一个连接件可以和任意数目的其它构件和连接件连接；

(4) 当两个连接件进行直接连接时，必须由其中一个的底部到另一个的顶部。



DSSA 是在一个特定应用领域中为一组应用提供组织结构参考的软件体系结构，参与 DSSA 的人员可以划分为 4 种角色，包括领域专家、领域设计人员、领域实现人员和 (45)，其基本活动包括领域分析、领域设计和 (46)。

(45) A. 领域测试人员 B. 领域顾问 C. 领域分析师 D. 领域经理

(46) A. 领域建模 B. 架构设计 C. 领域实现 D. 领域评估

**【答案】** C C

**【解析】**

(45) 参与 DSSA 的人员可以划分为四种角色：领域专家、领域分析师、领域设计人员和领域实现人员。

#### 1、领域专家

领域专家可能包括该领域中系统的有经验的用户、从事该领域中系统的需求分析、设计、实现以及项目管理的有经验的软件工程师等。领域专家的主要任务包括提供关于领域中系统的需求规约和实现的知识，帮助组织规范的、一致的领域字典，帮助选择样本系统作为领域工程的依据，复审领域模型、DSSA 等领域工程产品等。

领域专家应该熟悉该领域中系统的软件设计和实现、硬件限制、未来的用户需求及技术走向等。

#### 2、领域分析人员

领域分析人员应由具有知识工程背景的有经验的系统分析员来担任。领域分析人员的主要任务包括控制整个领域分析过程，进行知识获取，将获取的知识组织到领域模型中，根据



现有系统、标准规范等验证领域模型的准确性和一致性，维护领域模型。

领域分析人员应熟悉软件重用和领域分析方法；熟悉进行知识获取和知识表示所需的技术、语言和工具；应具有一定的该领域的经验，以便于分析领域中的问题及与领域专家进行交互；应具有较高的进行抽象、关联和类比的能力；应具有较高的与他人交互和合作的能力。

### 3、领域设计人员

领域设计人员应由有经验的软件设计人员来担任。领域设计人员的主要任务包括控制整个软件设计过程，根据领域模型和现有的系统开发出 DSSA，对 DSSA 的准确性和一致性进行验证，建立领域模型和 DSSA 之间的联系。

领域设计人员应熟悉软件重用和领域设计方法；熟悉软件设计方法；应有一定的该领域的经验，以便于分析领域中的问题及与领域专家进行交互。

### 4、领域实现人员

领域实现人员应由有经验的程序设计人员来担任。领域实现人员的主要任务包括根据领域模型和 DSSA，或者从头开发可重用构件，或者利用再工程的技术从现有系统中提取可重用构件，对可重用构件进行验证，建立 DSSA 与可重用构件间的联系。

领域实现人员应熟悉软件重用、领域实现及软件再工程技术；熟悉程序设计；具有一定的该领域的经验。

(46) DSSA 的基本活动包括：领域分析、领域设计、领域实现。

#### 1、领域分析

这个阶段的主要目标是获得领域模型。领域模型描述领域中系统之间的共同的需求，即领域模型所描述的需求为领域需求。在这个阶段中首先要进行一些准备性的活动，包括定义领域的边界。从而明确分析的对象；识别信息源，整个领域工程过程中信息的来源，可能的信息源包括现存系统、技术文献、问题域和系统开发的专家、用户调查和市场分析、领域演化的历史记录等，在此基础上就可以分析领域中系统的需求，确定哪些需求是领域中的系统广泛共享的，从而建立领域模型。当领域中存在大量系统时，需要选择它们的一个子集作为样本系统。对样本系统需求的考察将显示领域需求的一个变化范围。一些需求对所有被考察的系统是共同的，一些需求是单个系统所独有的。很多需求位于这两个极端之间，即被部分系统共享。

#### 2、领域设计

这个阶段的目标是获得 DSSA。DSSA 描述在领域模型中表示的需求的解决方案，它不是单个系统的表示，而是能够适应领域中多个系统的需求的一个高层次的设计。建立了领域模

型之后，就可以派生出满足这些被建模的领域需求的 DSSA，由于领域模型中的领域需求具有一定的变化性，DSSA 也要相应地具有变化性。它可以通过表示多选一的(alternative)、可选的(optional)解决方案等来做到这一点。模型和 DSSA 来组织的，因此在这个阶段通过获得 DSSA，也就同时形成了重用基础设施的规约。

### 3、领域实现

这个阶段的主要目标是依据领域模型和 DSSA 开发和组织可重用信息。这些可重用信息可能是从现有系统中提取得到，也可能需要通过新的开发得到。它们依据领域模型和 DSSA 进行组织，也就是领域模型和 DSSA 定义了这些可重用信息的双重用时机，从而支持了系统化的软件重用。这个阶段也可以看作重用基础设施的实现阶段。

值得注意的是，以上过程是一个反复的、逐渐求精的过程。在实施领域工程的每个阶段中，都可能返回到以前的步骤，对以前的步骤得到的结果进行修改和完善，再回到当前步骤，在新的基础上进行本阶段的活动。

(47) 不属于可修改性考虑的内容。

- (47)A. 可维护性      B. 可扩展性      C. 结构重构      D. 可变性

**【答案】D**

**【解析】**

可修改性(modifiability)是指能够快速地对系统以较高的性能价格比进行变更的能力。通常以某些具体的变更为基准，通过考察这些变更的代价衡量可修改性。可修改性包含四个方面。

(1)可维护性(maintainability)。这主要体现在问题的修复上：在错误发生后“修复”软件系统。为可维护性做好准备的软件体系结构往往能做局部性的修改并能使对其他构件的负面影响最小化。

(2)可扩展性(extendibility)。这一点关注的是使用新特性来扩展软件系统，以及使用改进版本来替换构件并删除不需要或不必要的特性和构件。为了实现可扩展性，软件系统需要松散耦合的构件。其目标是实现一种体系结构，它能使开发人员在不影响构件客户的情况下替换构件。支持把新构件集成到现有的体系结构中也是必要的。

(3)结构重组(reassemble)。这一点处理的是重新组织软件系统的构件及构件间的关系，例如通过将构件移动到一个不同的子系统而改变它的位置。为了支持结构重组，软件系统需要精心设计构件之间的关系。理想情况下，它们允许开发人员在不影响实现的主体部分的情

况下灵活地配置构件。

(4)可移植性(portability)。可移植性使软件系统适用于多种硬件平台、用户界面、操作系统、编程语言或编译器。为了实现可移植，需要按照硬件无关的方式组织软件系统，其他软件系统和环境被提取出。可移植性是系统能够在不同计算环境下运行的能力。这些环境可能是硬件、软件，也可能是两者的结合。在关于某个特定计算环境的所有假设都集中在一个构件中时，系统是可移植的。如果移植到新的系统需要做些更改，则可移植性就是一种特殊的可修改性。

用于管理信息系统规划的方法很多，主要是关键成功因素法（**Critical Success Factors**，**CSF**）、战略目标集转化法（**Strategy Set Transformation, SST**）和企业系统规划法（**Business System Planning, BSP**）。其它还有企业信息分析与集成技术（**BIAIT**）、产出/方法分析（**E/MA**）、投资回收法（**ROI**）、征费法（**chargout**）、零线预算法、阶石法等。用得最多的是前面三种。

### 1. 关键成功因素法（**CSF**）

在现行系统中，总存在着多个变量影响系统目标的实现，其中若干个因素是关键的和主要的（即关键成功因素）。通过对关键成功因素的识别，找出实现目标所需的关键信息集合，从而确定系统开发的优先次序。

关键成功因素来自于组织的目标，通过组织的目标分解和关键成功因素识别、性能指标识别，一直到产生数据字典。

识别关键成功因素，就是要识别联系于组织目标的主要数据类型及其关系。不同的组织的关键成功因素不同，不同时期关键成功因素也不相同。当在一个时期内的关键成功因素解决后，新的识别关键成功因素又开始。

关键成功因素法能抓住主要矛盾，使目标的识别突出重点。由于经理们比较熟悉这种方法，使用这种方法所确定的目标，因而经理们乐于努力去实现。该方法最有利于确定企业的管理目标。

### 2. 战略目标集转化法（**SST**）

把整个战略目标看成是一个“信息集合”，由使命、目标、战略等组成，管理信息系统的规划过程即是把组织的战略目标转变成为管理信息系统的战略目标的过程。

战略目标集转化法从另一个角度识别管理目标，它反映了各种人的要求，而且给出了按这种要求的分层，然后转化为信息系统目标的结构化方法。它能保证目标比较全面，疏漏较少，但它在突出重点方面不如关键成功因素法。

### 3. 企业系统规划法（**BSP**）

信息支持企业运行。通过自上而下地识别系统目标、企业过程和数据，然后对数

据进行分析，自下而上地设计信息系统。该管理信息系统支持企业目标的实现，表达所有管理层次的要求，向企业提供一致性信息，对组织机构的变动具有适应性。

企业系统规划法虽然也首先强调目标，但它没有明显的目标导引过程。它通过识别企业“过程”引出了系统目标，企业目标到系统目标的转化是通过企业过程/数据类等矩阵的分析得到的。

## 20、A 21、B

一般说来，信息化需求包含 3 个层次，即战略需求、运作需求和技术需求。一是战略需求。组织信息化的目标是提升组织的竞争能力、为组织的可持续发展提供一个支持环境。从某种意义上来说，信息化对组织不仅仅是服务的手段和实现现有战略的辅助工具；信息化可以把组织战略提升到一个新的水平，为组织带来新的发展契机。特别是对于企业，信息化战略是企业竞争的基础。<

二是运作需求。组织信息化的运作需求是组织信息化需求非常重要且关键的一环，它包含三方面的内容：一是实现信息化战略目标的需要；二是运作策略的需要。三是人才培养的需要。

三是技术需求。由于系统开发时间过长等问题在信息技术层面上对系统的完善、升级、集成和整合提出了需求。也有的组织，原来基本上没有大型的信息系统项目，有的也只是一些单机应用，这样的组织的信息化需求，一般是从头开发新的系统。

移植工作大体上分为计划阶段、准备阶段、转换阶段、测试阶段、验证阶段。

1、计划阶段，在计划阶段，要进行现有系统的调查整理，从移植技术、系统内容（是否进行系统提炼等）、系统运行三个方面，探讨如何转换成新系统，决定移植方法，确立移植工作体制及移植日程。

2、准备阶段，在准备阶段要进行移植方面的研究，准备转换所需的资料。该阶段的作业质量将对以后的生产效率产生很大的影响。

3、转换阶段，这一阶段是将程序设计和数据转换成新机器能根据需要工作的阶段。提高转换工作的精度，减轻下一阶段的测试负担是提高移植工作效率的基本内容。

4、测试阶段，这一阶段是进行程序单元、工作单元测试的阶段。在本阶段要核实程序能否在新系统中准确地工作。所以，当有不能准确工作的程序时，就要回到转换阶段重新工作。

5、验证阶段，这是测试完的程序使新系统工作，最后核实系统，准备正式运行的阶段。