

(14) 交互概览图(interaction overview diagram): 交互概览图是活动图和顺序图的混合物。

### 试题 17-【2017 年下半年-第 27 题】- UML

( ) 又称为设计视图, 它表示了设计模型中在架构方面具有重要意义的一部分, 即类、子系统、包和用例实现的子集。

- A. 逻辑视图                  B. 进程视图                  C. 实现视图                  D. 用例视图

【答案】A

【解析】高级教材第 4 版 P150, UML 对系统架构的定义是系统的组织结构, 包括系统分解的组成部分, 以及它们的关联性、交互机制和指导原则等提供系统设计的信息。具体来说, 就是指以下 5 个系统视图:

(1) 逻辑视图: 逻辑视图也称为设计视图, 它表示了设计模型中在架构方面具有重要意义的一部分, 即类、子系统、包和用例实现的子集。

(2) 进程视图: 进程视图是可执行线程和进程作为活动类的建模, 它是逻辑视图的一次执行实例, 描述了并发与同步结构。

(3) 实现视图: 实现视图对组成基于系统的物理代码的文件和构件进行建模。

(4) 部署视图: 部署视图把构件部署到一组物理节点上, 表示软件到硬件的映射和分布结构。

(5) 用例视图: 用例视图是最基本的需求分析模型。

### 试题 18-【2018 年上半年-第 7 题】- 软件架构风格

老于是某银行的系统架构师, 他为银行投资管理系统设计的软件架构包括进程通信和事件驱动的系统, 该软件架构风格属于 ( )。

- A. 数据流风格                  B. 独立构件风格                  C. 仓库风格                  D. 虚拟机风格

【答案】B

【解析】第 4 版教材 P130: Garlan 和 Shaw 对通用软件架构风格进行了分类, 他们将软件架构分为数据流风格、调用/返回风格、独立构件风格、虚拟机风格和仓库风格。

(1) 数据流风格: 数据流风格包括批处理序列和管道/过滤器两种风格。

(2) 调用/返回风格: 调用/返回风格包括主程序/子程序、数据抽象和面向对象, 以及层次结构。

(3) 独立构件风格: 独立构件风格包括进程通信和事件驱动的系统。

(4) 虚拟机风格: 虚拟机风格包括解释器和基于规则的系统。

(5) 仓库风格: 仓库风格包括数据库系统、黑板系统和超文本系统。

### 试题 19-【2018 年上半年-第 9 题】- 软件测试

软件测试是发现软件错误(缺陷)的主要手段, 软件测试方法可分为静态测试和动态测试, 其中 ( ) 属于静态测试。

- A. 代码走查                  B. 功能测试                  C. 黑盒测试                  D. 白盒测试

【答案】A

【解析】软件测试方法可分为静态测试和动态测试。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。对文档的静态测试主要以检查单的形式进行, 而对代码的静态测试一般采用桌前检查(Desk Checking)、代码走查和代码审查。