

## 第 6 章开发方法

1. 【2011 题 24】下列关于各种软件开发方法的叙述中, 错误的是( )。
  - A. 结构化开发方法的缺点是开发周期较长, 难以适应需求变化
  - B. 可以把结构化方法和面向对象方法结合起来进行系统开发, 使用面向对象方法进行自顶向下的划分, 自底向上地使用结构化方法开发系统
  - C. 与传统方法相比, 敏捷开发方法比较适合需求变化较大或者开发前期需求不是很清晰的项目, 以它的灵活性来适应需求的变化
  - D. 面向服务的方法以粗粒度、松散耦合和基于标准的服务为基础, 增强了系统的灵活性、可复用性和可演化性
2. 【2013 年题 21】逆向工程导出的信息可以分为实现级、结构级、功能级和领域级四个抽象层次。程序的抽象语法树属于( ); 反映程序分量之间相互依赖关系的信息属于( )。
  - A. 实现级    B. 结构级    C. 功能级    D. 领域级
  - A. 实现级    B. 结构级    C. 功能级    D. 领域级
3. 【2017 年题 29】应用系统构建中可以采用多种不同的技术, ( )可以将软件某种形式的描述转换为更高级的抽象表现形式, 而利用这些获取的信息, ( )能够对现有系统进行修改或重构, 从而产生系统的一个新版本。
  - (1) A. 逆向工程 (Reverse Engineering)
    - B. 系统改进 (System Improvement)
    - C. 设计恢复 (Design Recovery)
    - D. 再工程 (Re-engineering)
  - (2) A. 逆向工程 (Reverse Engineering)
    - B. 系统改进 (System Improvement)
    - C. 设计恢复 (Design Recovery)
    - D. 再工程 (Re-engineering)

### 6.1 软件生命周期

1. 【2011 年题 29】系统设计是软件开发的重要阶段, ( )主要是按系统需求说明来确定此系统的软件结构, 并设计出各个部分的功能和接口。
  - A. 外部设计
  - B. 内部设计
  - C. 程序设计
  - D. 输入/输出设计
2. 【2019 年题 16】按照传统的软件生命周期方法学, 可以把软件生命周期划分为软件定义、软件开发和( )三个阶段。其中, 可行性研究属于( )阶段的主要任务。
  - (1) A. 软件运行与维护
    - B. 软件对象管理
    - C. 软件详细设计
    - D. 问题描述
  - (2) A. 软件定义
    - B. 软件开发

- C. 软件评估
- D. 软件运行与维护

## 6.2 软件开发模型

1. 【2010 年题 25】( )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。
  - A. 原型模型
  - B. 瀑布模型
  - C. 螺旋模型
  - D. V 模型
2. 【2011 年题 30】快速迭代式的原型开发能够有效控制成本, ( )是指在开发过程中逐步改进和细化原型直至产生出目标系统。
  - A. 可视化原型开发
  - B. 抛弃式原型开发
  - C. 演化式原型开发
  - D. 增量式原型开发
4. 【2012 年题 23】快速应用开发(Rapid Application Development, RAD)通过使用基于( )的开发方法获得快速开发。当( )时, 最适合于采用 RAD 方法。
  - (1)A.用例          B.数据结构          C.剧情          D.构件
  - (2)A. 一个新系统要采用很多新技术
  - B. 新系统与现有系统有较高的互操作性
  - C. 系统模块化程度较高
  - D. 用户不能很好地参与到需求分析中
5. 【2012 年题 25】螺旋模型将整个软件开发过程分为多个阶段, 每个阶段都由目标设定、( )、开发和有效性验证以及评审 4 个部分组成。
  - A. 需求分析
  - B. 风险分析
  - C. 系统设计
  - D. 架构设计
6. 【2012 年题 26】以下关于软件生存周期模型的叙述, 正确的是( )。
  - A. 在瀑布模型中, 前一个阶段的错误和疏漏会隐蔽地带到后一个阶段
  - B. 在任何情况下使用演化模型, 都能在一定周期内由原型演化到最终产品
  - C. 软件生存周期模型的主要目标是为了加快软件开发的速度
  - D. 当一个软件系统的生存周期结束之后, 它就进入到一个新的生存周期模型
7. 【2016 年题 18】螺旋模型在( )的基础上扩展而成。
  - A.瀑布模型          B.原型模型          C.快速模型          D.面向对象模型
8. 【2017 年题 20】软件过程是制作软件产品的一组活动以及结果, 这些活动主要由软件人员来完成, 主要包括( )。软件过程模型是软件开发实际过程的抽象与概括, 它应该包括构成软件过程的各种活动。软件过程有各种各样的模型, 其中, ( )的活动之间存在因果关系, 前一阶段工作的结果是后一段阶段工作的输入描述。
  - (1)A.软件描述、软件开发和软件测试

- B.软件开发、软件有效性验证和软件测试
  - C.软件描述、软件设计、软件实现和软件测试
  - D.软件描述、软件开发、软件有效性验证和软件进化
- (2)A.瀑布模型
- B.原型模式
  - C.螺旋模型
  - D.基于构建的模型
9. 【2018 年题 20】软件开发过程模型中, ( ) 主要由原型开发阶段和目标软件开发阶段构成。
- A. 原型模型
  - B. 瀑布模型
  - C. 螺旋模型
  - D. 基于构件的模型
10. 【2018 年题 21】系统模块化程度较高时, 更适合于采用 (27) 方法, 该方法通过使用基于构件的开发方法获得快速开发。(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。
- (27) A. 快速应用开发
- B. 瀑布模型
  - C. 螺旋模型
  - D. 原型模型
- (28) A. 原型模型
- B. 瀑布模型
  - C. 螺旋模型
  - D. V 模型

### 6.2.5 构件组装模型

1. 【2009 年题 27】基于构件的开发模型包括软件的需求分析定义(1)、(2)、(3) 以及测试和发布 5 个顺序执行的阶段。
- (1)A. 构件接口设计
- B. 体系结构设计
  - C. 元数据设计
  - D. 集成环境设计
- (2)A. 数据库建模
- B. 业务过程建模
  - C. 对象建模
  - D. 构件库建立
- (3)A. 应用软件构建
- B. 构件配置管理
  - C. 构件单元测试
  - D. 构件编码实现

2. 【2014 年题 30】在构件组装过程中需要检测并解决架构失配问题。其中( )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。( )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。
- A.构件      B.模型      C.协议      D.连接子  
A.构件      B.模型      C.协议      D.连接子

### 6.3 统一过程

1. 【2011 年题 21】RUP 是一个二维的软件开发模型,其核心特点之一是( )。RUP 将软件开发生存周期划分为多个循环(cycle),每个循环由 4 个连续的阶段组成,每个阶段完成确定的任务。设计及确定系统的体系结构,制订工作计划及资源要求是在( )阶段完成的。
- A.数据驱动      B.模型驱动      C.用例驱动      D.状态驱动  
A.初始(inception) B.细化(elaboration)      C.构造(construction)      D.移交(transition)
2. 【2013 年题 23】基于 RUP 的软件过程是一个迭代过程。一个开发周期包括初始、细化、构建和移交四个阶段,每次通过这四个阶段就会产生一代软件,其中建立完善的架构是( )阶段的任务。采用迭代式开发,( )。
- (1)A. 初始      B. 细化      C. 构建      D. 移交  
(2)A. 在每一轮迭代中都要进行测试与集成  
B. 每一轮迭代的重点是对特定的用例进行部分实现  
C. 在后续迭代中强调用户的主动参与  
D. 通常以功能分解为基础
3. 【2015 年题 25】RUP 强调采用(1)的方式来开发软件,这样做的好处是(2)。
- (1)A.原型和螺旋      B.螺旋和增量      C.迭代和增量      D.快速和迭代  
(2)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

### 6.4 敏捷方法

1. 【2009 年题 21】( )方法以原型开发思想为基础,采用迭代增量式开发,发行版本小型化,比较适合需求变化较大或者开发前期对需求不是很清晰的项目。
- A. 信息工程  
B. 结构化  
C. 面向对象  
D. 敏捷
2. 【2014 年题 19】下列关于敏捷方法的叙述中,错误的是( )。
- A. 与传统方法相比,敏捷方法比较适合需求变化大或者开发前期对需求不是很清晰的项目  
B. 敏捷方法尤其适合于开发团队比较庞大的项目

- C. 敏捷方法的思想是适应性, 而不是预设性
  - D. 敏捷方法以原型开发思想为基础, 采用迭代式增量开发
3. 【2017 年题 21】以下关于敏捷方法的叙述中, ( ) 是不正确的。
- A. 敏捷型方法的思考角度是"面向开发过程"的
  - B. 极限编程是著名的敏捷开发方法
  - C. 敏捷型方法是"适应性"而非"预设性"
  - D. 敏捷开发方法是迭代增量式的开发方法

## 6.5 软件重用

1. 【2010 年题 27】软件的横向重用是指重用不同应用领域中的软件元素。( ) 是一种典型的、原始的横向重用机制。
- A. 对象
  - B. 构件
  - C. 标准函数库
  - D. 设计模式
2. 【2018 年题 25】软件重用可以分为垂直式重用和水平式重用, ( ) 是一种典型的水平式重用。
- A. 医学词汇表
  - B. 标准函数库
  - C. 电子商务标准
  - D. 网银支付接口

### 6.5.2 构件技术

1. 【2017 年题 26】软件构件是一个独立可部署的软件单元, 与程序设计中的对象不同, 构件( )。
- A. 是一个实例单元, 具有唯一的标志
  - B. 可以利用容器管理自身对外的可见状态
  - C. 利用工厂方法(如构造函数)来创建自己的实例
  - D. 之间可以共享一个类元素
2. 【2017 年题 27】为了使一个接口的规范和实现该接口的构件得到广泛应用, 需要实现接口的标准化。接口标准他是对( ) 的标准化。
- A. 保证接口唯一性的命名方案
  - B. 接口中消息模式、格式和协议
  - C. 接口中所接收的数据格式
  - D. 接口消息适用语境

## 6.6 基于架构的软件设计

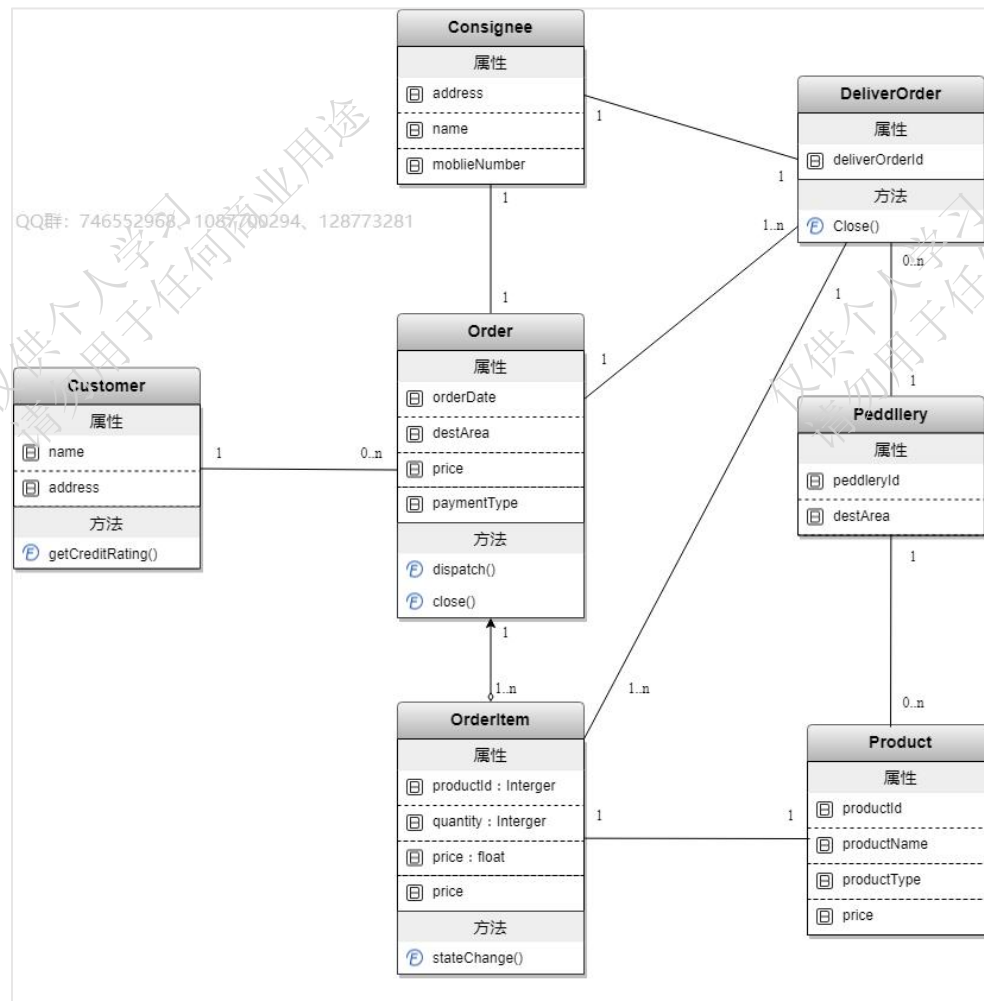
1. 【2009 年题 38】基于架构的软件设计(ABSD)强调由商业、质量和功能需求的组合驱动软件架构设计。以下关于 ABSD 的叙述中, 错误的是( )。
  - A. 使用 ABSD 方法, 设计活动可以从项目总体功能框架明确就开始
  - B. ABSD 方法是一个自顶向下, 递归细化的过程
  - C. ABSD 方法有三个基础: 功能分解、选择架构风格实现质量和商业需求以及软件模板的使用
  - D. 使用 ABSD 方法, 设计活动的开始意味着需求抽取和分析活动可以终止
3. 【2010 年题 39】基于软件架构的设计 (Architecture Based Software Development, 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( )来描述软件架构, 采用( )来描述需求。
  - (1)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
  - (2)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
4. 【2011 年题 36】基于架构的软件设计(ABSD)强调由商业、质量和功能需求的组合驱动软件架构设计。ABSD 方法有三个基础: 功能分解、( )和软件模板的使用。
  - A. 对需求进行优先级排列
  - B. 根据需求自行设计系统的总体架构
  - C. 选择架构风格实现质量及商业需求
  - D. 开发系统原型用于测试
5. 【2013 年题 33】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( )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( )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; ( )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( )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( )。
  - (1)A. 架构风格      B. 设计模式      C. 架构策略      D. 架构描述
  - (2)A. 架构设计      B. 架构实现      C. 架构复审      D. 架构演化
  - (3)A. 架构设计      B. 架构实现      C. 架构复审      D. 架构演化
  - (4)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
  - (5)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

6. 【2015 年题 32】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 ( ) 描述软件架构, 用 ( ) 来描述需求。
- (1) 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
- (2) 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
7. 【2017 年题 32】在基于体系结构的软件设计方法中, 采用 ( ) 来描述软件架构, 采用 ( ) 但来描述功能需求, 采用 ( ) 来描述质量需求。
- 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
- A. 类图      B. 视角      C. 用例      D. 质量场景
- A. 连接件      B. 用例      C. 质量场景      D. 质量属性
8. 【2019 年题 22】基于构件的软件开发中, 构件分类方法可以归纳为三大类: ( ) 根据领域分析的结果将应用领域的概念按照从抽象到具体的顺序逐次分解为树形或有向无回路图结构; ( ) 利用 Facet 描述构件执行的功能、被操作的数据、构件应用的语境或任意其他特征; ( ) 使得检索者在阅读文档过程中可以按照人类的联想思维方式任意跳转到包含相关概念或构件的文档。
- A. 关键字分类法
- B. 刻面分类法
- C. 语义匹配法
- D. 超文本方法
- (2) A. 关键字分类法
- B. 刻面分类法
- C. 语义匹配法
- D. 超文本方法
- (3) A. 关键字分类法
- B. 刻面分类法
- C. 语义匹配法
- D. 超文本方法
9. 【2019 年题 23】构件组装是指将库中的构件经适当修改后相互连接构成新的目标软件。( ) 不属于构件组装技术。
- A. 基于功能的构件组装技术
- B. 基于数据的构件组装技术
- C. 基于实现的构件组装技术
- D. 面向对象的构件组装技术

10. 【2019 年题 27】一个完整的软件系统需从不同视角进行描述，下图属于软件架构设计中的( )用于( )视图来描述软件系统。



- (1) A. 对象图  
B. 时序图  
C. 构件图  
D. 类图
- (2) A. 进程  
B. 开发  
C. 物理  
D. 用户

## 6.6.2 基于架构的软件开发模型

1. 【2009 年题 37】软件架构需求是指用户对目标软件系统在功能、行为、性能、设计约束等方面的期望。以下活动中，不属于软件架构需求过程范畴的是( )。
- A. 设计构件  
B. 需求获取  
C. 标识构件  
D. 架构需求评审



3. 【2009 年题 40】架构复审是基于架构开发中一个重要的环节。以下关于架构复审的叙述中, 错误的是( )。
- A. 架构复审的目标是标识潜在的风险, 及早发现架构设计的缺陷和错误
  - B. 架构复审过程中, 通常会对一个可运行的最小化系统进行架构评估和测试
  - C. 架构复审人员由系统设计与开发人员组成
  - D. 架构设计、文档化和复审是一个迭代的过程
4. 【2010 年题 31】在基于构件的软件开发中, ( )描述系统设计蓝图以保证系统提供适当的功能; ( )用来了解系统的性能、吞吐率等非功能性属性。
- (1)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
- (2)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
5. 【2012 年题 15】采用以架构为核心的软件开发方法, 在建立软件架构的初期, 首要任务是选择一个合适的( ), 在此基础上, 开发人员通过架构模型, 可以获得关于上( )的理解, 为将来的架构实现与演化过程建立了目标。
- A.分析模式      B.设计模式      C.架构风格      D.架构标准  
A.架构需求      B.架构属性      C.架构优先级      D.架构约束
6. 【2019 年题 30】ABSDM (Architecture-Based Software Design Model) 把整个基于体系结构的软件过程划分为体系结构需求、体系结构设计、体系结构文档化、( )、( )和体系结构演化等 6 个子过程。其中, ( )过程的主要输出结果是体系结构规格说明和测试体系结构需求的质量设计说明书。
- (1) A. 体系结构复审  
B. 体系结构测试  
C. 体系结构变更  
D. 体系结构管理
- (2) A. 体系结构实现  
B. 体系结构测试  
C. 体系结构建模  
D. 体系结构管理
- (3) A. 体系结构设计  
B. 体系结构需求  
C. 体系结构文档化  
D. 体系结构测试

## 6.7 形式化方法

1. 【2012 年题 22】以下关于软件开发方法的叙述, 错误的是( )。
- A. 对于较为复杂的应用问题, 适合采用形式化方法进行需求分析
  - B. 形式化方法的优势在于能够精确地表述和研究应用问题及其软件实现

- C. 净室软件工程将正确性验证作为发现和排除错误的主要机制
- D. 净室软件工程强调统计质量控制技术, 包括对客户软件使用预期的测试

## 6.8 其他

1. 【2009 年题 23】逆向工程导出的信息可以分为 4 个抽象层次, 其中( ) 可以抽象出程序的抽象语法树、符号表等信息; ( ) 可以抽象出反映程序段功能及程序段之间关系的信息。  
A. 实现级                  B. 结构级                  C. 功能级                  D. 领域级  
A. 实现级                  B. 结构级                  C. 功能级                  D. 领域级
2. 【2009 年题 39】软件架构文档是对软件架构的正式描述, 能够帮助与系统有关的开发人员更好地理解软件架构。软件架构文档的写作应该遵循一定的原则。以下关于软件架构文档写作原则的叙述中, 错误的是( )。  
A. 架构文档应该从架构设计者的角度进行编写  
B. 应该保持架构文档的即时更新, 但更新不要过于频繁  
C. 架构文档中的描述应该尽量避免不必要的重复  
D. 每次架构文档修改, 都应该记录修改的原则
3. 【2010 年题 28】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( )。  
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
4. 【2013 年题 25】以下关于自顶向下开发方法的叙述中, 正确的是( )。  
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)
5. 【2016 年题 19】( ) 适用于程序开发人员在地域上分布很广的开发团队。( ) 中, 编程开发人员分成首席程序员和“类”程序员。  
(1)、A.水晶系列 (Crystal) 开发方法  
B.开放式源码 (Open source) 开发方法  
C.SCRUM 开发方法  
D.功用驱动开发方法 (FDD)  
(2)、A.自适应软件开发 (ASD)  
B.极限编程 (XP) 开发方法  
C.开放统一过程开发方法 (OpenUP)  
D.功用驱动开发方法 (FDD)
6. 【2017 年题 28】OMG 接口定义语言 IDL 文件包含了六种不同的元素,( ) 是一个 IDL 文件核心的内容, ( ) 将映射为 Java 语言中的包 (package) 或 c++语言中的命名空间 (Namespace)。  
A.模块定义                  B.消息结构                  C.接口描述                  D.值类型

A.模块定义      B.消息结构      C.接口描述      D.值类型

7. 【2017 年题 33】体系结构文档化有助于辅助系统分析人员和程序员去实现体系结构。体系结构文档化过程的主要输出包括( )。
- A. 体系结构规格说明、测试体系结构需求的质量设计说明书
  - B. 质量属性说明书、体系结构描述
  - C. 体系结构规格说明、软件功能需求说明
  - D. 多视图体系结构模型、体系结构验证说明
8. 【2019 年题 24】软件逆向工程就是分析已有的程序, 寻求比源代码更高级的抽象表现形式。在逆向工程导出信息的四个抽象层次中, ( )包括反映程序各部分之间相互依赖关系的信息; ( )包括反映程序段功能及程序段之间关系的信息。
- (1) A. 实现级
  - B. 结构级
  - C. 功能级
  - D. 领域级
  - (2) A. 实现级
  - B. 结构级
  - C. 功能级
  - D. 领域级
9. 【2019 年题 25】( )是在逆向工程所获取信息的基础上修改或重构已有的系统, 产生系统的一个新版本。
- A. 逆向分析 (Reverse Analysis)
  - B. 重组 (Restructuring)
  - C. 设计恢复 (Design Recovery)
  - D. 重构工程 (Re-engineering)
10. 【2019 年题 41】软件方法学是以软件开发方法为研究对象的学科。其中, ( )是先对最高层次中的问题进行定义、设计、编程和测试, 而将其中未解决的问题作为一个子任务放到下一层次中去解决。( )是根据系统功能要求, 从具体的器件、逻辑部件或者相似系统开始, 通过对其进行相互连接、修改和扩大, 构成所要求的系统。( )是建立在 严格数学基础上的软件开发方法。
- (1) A. 面向对象开发方法
  - B. 形式化开发方法
  - C. 非形式化开发方法
  - D. 自顶向下开发方法
  - (2) A. 自底向上开发方法
  - B. 形式化开发方法
  - C. 非形式化开发方法
  - D. 原型开发方法
  - (3) A. 自底向上开发方法
  - B. 形式化开发方法
  - C. 非形式化开发方法
  - D. 自顶向下开发方法