

# 模拟考试

## **iSAQB® Certified Professional for Software Architecture Foundation Level (CPSA-F®)**

考试答案卷 2024.1-rev0-CN-20240202



## 关于软件系统架构专业人士模拟考试的说明 - - 基础级 (CPSA-F®)

软件系统架构专业人士认证-基础级 (CPSA-F®) 模拟考试说明。本次考试是一次模拟考试，在形式和范围上都源于正式的软件系统架构专业人士认证 - 基础级 (CPSA-F®) 的认证考试。其旨在说明真实的iSAQB® CPSA®考试，并为正式考试做准备。本次模拟考试包含39道多项选择题，根据难度级别，每道题分值为1或2分。至少须达到总分数的60%才能通过考试。

在本次模拟考试中，最高可以获得51.0分，通过需要达到30.6分。

以下是考试规则：

- 根据问题的难度和长度，每道题您可以获得1或2分。
- 选择正确答案会得到相应分数，错误答案会倒扣分，但分数计算仅限于当前题。如果某道题因错误答案导致得分为负，那么该题总分按0分计。

模拟考试的多项选择题分为三种类型：

**A类问题（单选，单一正确答案）：**从可能的答案列表中选择问题的唯一正确答案。只且只有一个正确答案。选择正确答案将获得相应的指定分数。

**P类问题（多选，选择多个答案）：**从可能的答案列表里选择问题文本中给出的正确（或最合适的）答案数量。根据导语文本中所需的答案数量选择相同数量的答案。每个正确答案将获得总分的1/n。每个不正确的选择会扣除总分的1/n。

**K类问题（分配型问题，选择类别）：**对于每个答案作出选择，选择两个选项中的正确选项（“正确”或“错误”或“适用”或“不适用”）。每个正确的选择将获得总分的1/n。每个错误的选择将导致扣除总分的1/n。如果在某一行中未选择任何答案，则既没有得分也没有扣分。

有关问题类型和评分系统更详细的说明，[请参阅CPSA-F®考试规则](#)。

母语考生的考试时间为75分钟，非母语考生为90分钟。为了确保备考尽可能真实，在模拟考试中应遵守正式考试时间，并且不得使用任何辅助工具（如研讨会资料、书籍、互联网等）。随后可以使用本模拟考试的答案卷对考试进行评估。鉴于iSAQB® e.V. 是资料来源和版权所有者，本模拟考试可以在培训课程、备考或免费传递的情况下使用。

但是，明确禁止在真正的考试中使用这些考试题目。

## 问卷

### 问题1

ID: Q-20-04-01

---

A类问题： 请选择一个答案。 1 分

---

“软件架构”存在多少种定义？

- (a) 对于所有类型的系统只存在一种定义。
- (b) 每种类型的(例如, “嵌入式”, “实时”, “决策支持”, “web网络”, “批处理”, …) 软件系统都存在一种定义。
- (c) 十几个或者更多不同的定义。
- 

### 问题 2

ID: Q-20-04-02

---

P类问题： 请从以下五个答案中选择最适合的三个。 1 分

---

以下哪三个方面包含在术语“软件架构”内？

- (a) 组件
- (b) 横切概念
- (c) (内部和外部) 接口
- (d) 编码规则
- (e) 硬件尺寸

## 问题 3

ID: Q-17-13-01

---

P类问题： 请从以下七个答案中选择最恰当的四个。 2 分

---

以下关于（横切）方案的陈述中最为恰当的是哪四个？

- (a) 该方案的统一使用减少了构建块之间的耦合。
- (b) 适当方案的定义确保了架构模式的合规性。
- (c) 当架构师和开发人员在实现前就合适的方案达成一致时，可以实现统一的异常处理。
- (d) 每个质量目标都应有一个明确编档记录的方案。
- (e) 方案是增加一致性的手段。
- (f) 方案可以为多个构建块的实现定义约束。
- (g) 方案可以由单个构建块实现。

## 问题 4

ID: Q-17-13-02

---

K类问题： 请对所有答案进行选择。 2 分

---

在您的项目中，有三位架构师和七位开发人员正在进行软件架构的文档编写。哪些方法适用于实现一致和充分的文档，哪些方法不适用？

- | 适用                                  | 不适用                                 |                        |
|-------------------------------------|-------------------------------------|------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (a) 首席架构师协调文档的创建。      |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (b) 文档使用相同的模板。         |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (c) 文档的所有部分都从源代码中自动提取。 |

解释：诸如论证或备选方案等内容不会包含在代码中，但需要包含在文档中，因此并非文档的所有内容都能从源代码中提取。

**问题 5****ID: Q-17-13-03**

---

P类问题： 请从以下八个答案中选择最恰当的四个。 1 分

---

以下哪四种技术用来说明系统在运行时的工作流程或行为最恰当？

- (a) 流程图
- (b) 活动图
- (c) 屏幕流程的描述（用户交互顺序）
- (d) 顺序图
- (e) 线性文氏图（韦恩图）
- (f) 顺序步骤的编号列表
- (g) 接口的表格描述
- (h) 类图

**问题 6****ID: Q-17-13-04**

---

P类问题： 请从以下五个答案中选择最适合的三个。 1 分

---

以下哪三个原则适用于测试？

- (a) 一般来说，不可能发现系统中的所有错误。
- (b) 在之前具有许多已知错误的组件中，出现其他错误的可能性很高。
- (c) 充分的测试可以表明程序没有错误。
- (d) 测试表明错误的存在而不是错误的不存在。
- (e) 函数式编程不允许自动化测试。

## 问题 7

ID: Q-17-03-05

K类问题: 请对所有答案进行选择。

1 分

以下是关于信息隐藏原则的陈述，哪些是正确的，哪些是错误的？

- | 正确                                  | 错误                                  |                                |
|-------------------------------------|-------------------------------------|--------------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (a) 遵循信息隐藏原则增加了变更的灵活性。         |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (c) 信息隐藏涉及刻意向构件块的调用者或使用者的隐藏信息。 |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (c) 信息隐藏使得自下而上工作更加困难。          |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (d) 信息隐藏是沿控制流逐步求精方法的一个派生概念。    |

## 问题 8

ID: Q-20-04-03

P类问题: 请从以下四个答案中选择最恰当的两个。

1 分

软件架构的最重要的两个目标是什么？

- |                                     |     |                             |
|-------------------------------------|-----|-----------------------------|
| <input type="checkbox"/>            | (a) | 提高结构和实现中模式的准确性。             |
| <input checked="" type="checkbox"/> | (b) | 以一种易于理解的方式实现质量需求。           |
| <input type="checkbox"/>            | (c) | 实现有性价比的系统集成和验收测试。           |
| <input checked="" type="checkbox"/> | (d) | 为开发团队和其他利益相关者提供对结构和方案的基本理解。 |

## 问题 9

ID: Q-20-04-12

K类问题： 请对所有答案进行选择。

1 分

请您将自己置于银行或保险领域的大型分布式业务应用的软件架构师的位置上。以下哪些陈述是正确的，哪些是错误的？

- | 正确                                  | 错误                                  |     |   |
|-------------------------------------|-------------------------------------|-----|---|
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (a) | 架构师与利益相关者合作，确定哪些需求和约束会频繁变更（例如，业务流程、技术），并设计架构，以便发生变更时无需对架构进行大规模重构。 |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (b) | 必需的产品质量应该驱动您的架构决策。  |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (c) | 软件架构可以完全独立于硬件和基础设施进行设计。   |

## 问题 10

ID: Q-20-04-03

P类问题： 请从以下五个答案中选择恰当的三个。

2 分

您作为软件架构师，在需求方面的三个最重要的责任是什么？

- |                                     |     |                            |
|-------------------------------------|-----|----------------------------|
| <input checked="" type="checkbox"/> | (a) | 支持业务人员指定明确具体的质量需求。         |
| <input checked="" type="checkbox"/> | (b) | 根据您的技术知识帮助识别新的业务可能性。       |
| <input type="checkbox"/>            | (c) | 拒绝包含技术风险的业务需求。             |
| <input type="checkbox"/>            | (d) | 将所有的业务需求以您的开发团队能理解的术语捕获下来。 |
| <input checked="" type="checkbox"/> | (e) | 检查需求的技术可行性。                |

解释：关于选项（c）：架构师的任务不是仅因为其中包含风险就拒绝需求。我们应该识别和沟通这些风险，而不是拒绝这些需求。

**问题 11****ID: Q-20-04-07**

---

P类问题： 请从以下五个答案中选择最适合的三个。

---

2 分

作为架构师，您负责根据业务的持续需求维护一个遗留系统的运行。您议程上的三个最重要的行动项目是什么？

- (a) 为您的团队就维护的预算进行谈判
  - (b) 确保已部署系统的文档更新
  - (c) 分析新需求对于当前系统的影响
  - (d) 鼓励团队成员学习新的编程语言
  - (e) 除业务需求外向管理层建议技术更新
- 

**问题 12****ID: Q-21-05-01**

---

K类问题： 请对所有答案进行选择。

---

1 分

以下关于架构决策的陈述那些是正确的，哪些是错误的？

- | 正确                                  | 错误                                  |     |                                     |
|-------------------------------------|-------------------------------------|-----|-------------------------------------|
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (a) | 架构决策永远不需要被书面记录，因为它们已经为开发团队所知晓。      |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (b) | 架构决策记录有助于理解决策的背景。                   |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (c) | 一旦就中心或基本框架（例如，持久性框架）做出决策，那么该决策不得变更。 |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (d) | 质量需求对架构决策有很大帮助。                     |
-

## 问题 13

ID: Q-20-04-09

K类问题: 请对所有答案进行选择。

1 分

请判断以下陈述哪些正确，哪些错误。

正确	错误		
<input checked="" type="checkbox"/>	<input type="checkbox"/>	(a)	敏捷开发方法的每次迭代都可能对基本架构决策产生影响。
<input type="checkbox"/>	<input checked="" type="checkbox"/>	(b)	与瀑布式开发项目相比，迭代式项目在架构工作上的总投入要高得多。
<input type="checkbox"/>	<input checked="" type="checkbox"/>	(c)	敏捷项目不需要架构文档，因为开发团队使用每日站立会议来沟通决策。
<input type="checkbox"/>	<input checked="" type="checkbox"/>	(d)	如果您的系统由一组微服务组成，则无需核心架构文档，因为每个微服务都可以自由选择其技术。

## 问题 14

ID: Q-20-04-10

K类问题: 请对所有答案进行选择。

2 分

关于项目目标和架构的以下陈述，哪些正确，哪些错误。

正确	错误		
<input checked="" type="checkbox"/>	<input type="checkbox"/>	(a)	项目目标可以包括功能需求和质量需求。
<input checked="" type="checkbox"/>	<input type="checkbox"/>	(b)	架构目标来源于系统或产品的质量需求。
<input type="checkbox"/>	<input checked="" type="checkbox"/>	(c)	业务利益相关者应该关注于业务目标，不干涉架构目标。
<input type="checkbox"/>	<input checked="" type="checkbox"/>	(d)	为了避免冲突，业务目标和架构目标应该是非重叠的集合。

解释：业务利益相关者可能非常关心性能、灵活性或安全性这些目标，这些目标被认为是“架构目标”。

**问题 15****ID: Q-20-04-11**

---

P类问题: 请从以下五个答案中选择最适合的**两个**。 1 分

---

架构工作中的规则“明确、非隐含”是什么意思？请选择最合适的两个答案。

- (a) 架构师应该避免使用递归结构，并用明确的循环来替代它们。
  - (b) 架构师应该明确导致决策的假设。
  - (c) 架构师应该明确要求每个构建块都有自然语言解释（即注释）。
  - (d) 架构师应该明确要求团队提供估计开发工作量的书面或至少口头理由。
  - (e) 架构师应该明确其决策的先决条件。
- 

**问题 16****ID: Q-20-04-19**

---

P类问题: 请从以下五个答案中选择最适合的**三个**。 1 分

---

请识别典型软件系统类别的三个最合适的示例。

- (a) 批处理系统
- (b) 交互式在线系统
- (c) 林奈系统
- (d) 嵌入式实时系统
- (e) 集成测试系统

**问题 17****ID: Q-20-04-32**

---

P类问题： 请从以下五个答案中选择最适合的三个。

---

1 分

有很多方法可以用来设计开发软件架构。以下哪三种在实践中最常见？

- (a) 用户界面驱动设计
  - (b) 领域驱动设计
  - (c) 基于视图的架构开发
  - (d) 自下而上设计
  - (e) 多数票表决
- 

**问题 18****ID: Q-20-04-38**

---

P类问题： 请从以下六个答案中选择最适合的三个。

---

1 分

不少架构开发方法都建议采用基于视图的方法。以下哪三种视图最常被使用？

- (a) 物理数据库视图
- (b) 上下文视图
- (c) 构建块/组件视图
- (d) 测试驱动视图
- (e) 配置视图
- (f) 运行时视图

**问题 19****ID: Q-20-04-22**

---

P类问题： 请从以下四个答案中选择最适合的**两个**。

---

1 分

当记录软件架构的构建块时，黑盒描述应包含哪两种信息？

- (a) 公共接口
  - (b) 构建块的职责
  - (c) 构建块的内部结构
  - (d) 实现细节的规范
- 

**问题 20****ID: Q-20-04-17**

---

P类问题： 请根据以下五个答案选择最适合的**两个**。

---

1 分

在开发软件架构之前必须满足哪些先决条件？请选择两个最合适的答案。

- (a) 系统的需求规则是完整、详细和一致的。
- (b) 系统最重要的质量要求是已知的。
- (c) 组织上的约束限制是已知的。
- (d) 编程语言已经选择。
- (e) 开发团队的硬件是可用的。

解释：在大多数情况下，拥有完整的需求规范是不现实的。通常，拥有概述并了解某些细节（例如，质量需求）就足够了。

**问题 21****ID: Q-20-04-18**

---

P类问题: 请根据以下四个答案选择最适合的三个。 1 分

---

哪些因素可能影响软件架构的设计? 请选择三个最合适的答案。

- (a) 政治因素
  - (b) 组织因素
  - (c) 技术因素
  - (d) 虚拟因素
- 

**问题 22****ID: Q-20-04-18**

---

A类问题: 请选择一个答案。 1 分

---

以下哪些特性最有可能通过使用分层架构来改善?

- (a) 运行时效率 (性能)
- (b) 修改或变更系统的灵活性
- (c) 运行时的灵活性 (易配置性)
- (d) 不可否认性

**问题 23****ID: Q-20-04-33**

---

P类问题： 请根据以下四个答案选择最适合的**两个**。

---

1 分

哪种类型的问题最适合管道过滤器（Pipes & Filter） 模式？

- (a) 管理全局应用程序的状态
  - (b) 处理数据流的IT系统
  - (c) 解耦执行的多个步骤
  - (d) 应用程序的时间解耦
- 

**问题 24****ID: Q-20-04-20**

---

A类问题： 请选择一个答案。

---

1 分

您可以尝试通过依赖倒置原则实现哪个目标？

- (a) 大的构建块不应该依赖于小的构建块。
- (b) 组件应更容易创建依赖组件。
- (c) 构建块应该仅通过抽象相互依赖。

## 问题 25

ID: Q-20-04-21

K类问题： 请对所有答案进行选择。

1 分

紧（高）耦合和松（低）耦合的特征是什么？

紧耦合	松耦合		
<input checked="" type="checkbox"/>	<input type="checkbox"/>	(a)	构建块直接调用依赖的构建块，即不使用接口或抽象间接调用。
<input checked="" type="checkbox"/>	<input type="checkbox"/>	(b)	构建块使用共享的复杂数据结构。
<input checked="" type="checkbox"/>	<input type="checkbox"/>	(c)	构建块在关系数据库中使用共享表（用于读和写操作）。
<input type="checkbox"/>	<input checked="" type="checkbox"/>	(d)	在设计构建块时，一直遵循依赖倒置原则。

## 问题 26

ID: Q-20-04-14

P类问题： 请根据以下五个答案选择最适合的两个。

2 分

关于“不要重复自己”（DRY: Don't repeat yourself）原则，哪两个陈述最合适？换句话说：如果系统中的部分源代码或配置存在多个副本，可能会发生什么？

- (a) DRY原则降低信息安全。
- (b) 严格遵循DRY原则可能导致更高的耦合。
- (c) 系统中包含冗余代码的组件可以独立改进。
- (d) 遵循DRY原则会导致IT信息安全中的额外攻击向量。
- (e) 分层架构模式中允许一致性地使用DRY原则。

## 问题 27

ID: Q-20-04-15

K类问题: 请对所有答案进行选择。

2 分

您可以口头和/或书面沟通软件架构的不同方面。这些形式之间的相关性如何？对于以下每个陈述，请指出它是正确的还是错误的。

正确      错误

- |                                     |                                     |     |                                     |
|-------------------------------------|-------------------------------------|-----|-------------------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (a) | 口头沟通应该补充书面文档。                       |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (b) | 对架构决策的反馈应始终以书面形式进行，以确保可追溯性。         |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (c) | 书面文档应始终优先于口头沟通。                     |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (d) | 架构师应该选择一种形式（口头或书面），并在整个开发过程中坚持这种选择。 |

解释:

- 有时口头沟通需要先行，没有通用规则。
- 反馈不应局限于书面陈述。

## 问题 28

ID: Q-20-04-37

K类问题: 请对所有答案进行选择。

2 分

以下关于架构视图符号的陈述哪些是正确的，哪些是错误的？

正确      错误

- |                                     |                                     |     |  |
|-------------------------------------|-------------------------------------|-----|--|
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (a) | 业务流程模型和符号（BPMN）应该只由业务分析人员使用，而不是用于架构文档。 |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (b) | UML部署模型是记录软件组件到基础设施映射的唯一方法。            |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (c) | UML包图可以用于记录软件架构的构建块视图。                 |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (d) | 只要符号得到解释（例如通过图例），任何符号都足以描述构建块的结构和协作。   |

**问题 29****ID: Q-20-04-13**

---

P类问题： 请根据以下四个答案选择最适合的**两个**。 1 分

---

哪两种架构视图在开发软件架构方面具有最实际的应用？

- (a) 模式视图
- (b) 观察者视图
- (c) 构建块视图（组件视图）
- (d) 部署视图

**问题 30****ID: Q-20-04-23**

---

P类问题： 请根据以下五个答案选择最适合的**两个**。 1 分

---

上下文视图可能包含业务上下文和技术上下文，或两者兼而有之。请选择适用于技术上下文的最合适的两个答案。

- (a) 技术上下文包含系统与其环境之间的物理通道。
- (b) 技术上下文包含部署系统组件的所有基础设施。
- (c) 技术上下文应包含用作架构基础设施的硬件定价或云服务定价。
- (d) 技术上下文包含有关所选择的编程语言以及用于实现软件架构的所有框架的信息。
- (e) 技术上下文可能包含与业务上下文不同的元素。

## 问题 31

ID: Q-20-04-24

---

P类问题: 请根据以下四个答案选择最适合的**两个**。 1 分

---

软件架构文档可能包含对横切关注点的描述。请选择两个最好的理由说明为什么横切关注点的文档有用。

- (a) 横切概念应专注于业务领域，并且不应包含技术信息。
- (b) 在软件架构多个部分中使用的方面或概念应以非冗余的方式描述。
- (c) 横切概念可以在同一组织中的更多产品中重复使用。
- (d) 横切概念应由专家来实现。因此，单独的文档是有用的。
- 

## 问题 32

ID: Q-20-04-25

---

K类问题: 请对所有答案进行选择。 2 分

---

好的接口设计准则是什么？请检查以下哪些陈述是正确的，哪些是错误的。

- | 正确                                  | 错误                                  |     |                        |
|-------------------------------------|-------------------------------------|-----|------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (a) | 接口的使用应易于学习。            |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (b) | 应该能够编写相对容易理解的接口客户端代码。  |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (c) | 一个接口应该提供对全面的实现细节的访问。   |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (d) | 接口规范应包含功能和非功能方面。       |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (e) | 对接口的本地和远程调用在所有方面应表现一致。 |

## 解释:

关于选项(e)，“在所有方面具有相同的行为”：在技术上，实现具有完全相同行为是不可行的，至少是在延迟和响应时间方面。更详细的解释可以在（著名的）[《分布式计算的谬论》](#)中找到。

## 问题 33

ID: Q-20-04-26

K类问题： 请对所有答案进行选择。

1 分

有一个定义是：“软件架构是在开发过程中您所做的所有决策的总和。” 请检查以下关于架构/设计决策的陈述哪些是正确的，哪些是错误的。

- | 正确                                  | 错误                                  |                          |
|-------------------------------------|-------------------------------------|--------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (a) 架构决策可以影响构建块或组件的结构。   |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (b) 软件架构师应该书面说明所有设计决策。   |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (c) 架构决策之间可能存在相互依赖关系。    |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (d) 应明确决定在冲突的质量需求之间进行权衡。 |

解释：并非所有决定都需要以书面形式进行证明 - 因为书面文档的要求取决于情境、团队、系统和其他因素。

## 问题 34

ID: Q-20-04-31

K类问题： 请对所有答案进行选择。

2 分

以下哪些陈述是引入架构文档的典型原因，哪些不是典型原因？

- | 典型                                  | 非典型                                 |                   |
|-------------------------------------|-------------------------------------|-------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (a) 支持新进开发人员的入职培训 |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (b) 支持系统的自动化测试方法  |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (c) 支持分布式团队的工作    |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (d) 协助产品的未来改进     |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (e) 遵守监管或法律约束     |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (f) 确保开发人员有足够的工作量 |

请仔细阅读：该问题是关于引入架构文档的**典型**原因。

## 问题 35

ID: Q-20-04-30

K类问题： 请对所有答案进行选择。

1 分

以下哪些质量要求通常彼此冲突，哪些不是？

- | 冲突                                  | 不冲突                                 |                   |
|-------------------------------------|-------------------------------------|-------------------|
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (a) 易理解性 - 可读性    |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (b) 易用性 - 信息安全性   |
| <input checked="" type="checkbox"/> | <input type="checkbox"/>            | (c) 运行时易配置性 - 健壮性 |
| <input type="checkbox"/>            | <input checked="" type="checkbox"/> | (d) 信息安全性 - 功能正确性 |

## 问题 36

ID: Q-20-04-27

P类问题： 请根据以下四个答案选择最适合的两个。

1 分

ISO 25010 为软件系统提供了通用的质量特性。以下哪两种方法可以使与这些质量特性相关的质量要求更具体？请选择两个最佳备选方案。

- (a) 通过开发用户界面原型
- (b) 通过定义明确的接口
- (c) 通过讨论或编写场景
- (d) 通过创建自动化测试
- (e) 通过创建质量树

## 问题 37

ID: Q-20-04-28

---

P类问题： 请根据以下六个答案选择最适合的**四个**。

---

2 分

以下哪四项答案最适合为分析软件架构质量需求的实现情况（定性分析）提供支持？请选择四个最佳备选方案。

- (a) 量化依赖分析
- (b) 架构模型
- (c) 质量场景
- (d) 团队规模
- (e) 日志文件
- (f) 组织结构

## 问题 38

ID: Q-20-04-29

---

P类问题： 请根据以下五个答案选择最适合的**两个**。

---

2 分

您尝试对架构进行定量分析。哪两个指标最能反应架构的问题区块？

- (a) 组件的高耦合
- (b) 公开方法的名称没有反映其目的
- (c) 缺少注释
- (d) 系统中某些构建块存在错误集群
- (e) 每个组件的测试用例数量

## 问题 39

ID: Q-20-04-36

---

P类问题： 请根据以下五个答案选择最适合的三个。

---

1 分

您尝试对架构进行定量分析。以下哪三种属性可以在软件架构中进行可靠的测量？请选择三个最合适的答案。

- (a) 构建块的大小（例如代码行数）
- (b) 组件源代码的变更率
- (c) 架构组件的内聚性
- (d) 组件的信息安全级别
- (e) 参与特定组件开发的开发人员数量

解释：

- 构建块大小可以通过静态分析源代码轻松可靠地测量（代码行数是可靠的大小度量标准）。
  - 当考虑版本控制历史时，可以可靠地测量变更率和每个组件的开发人员数量，这在像git、subversion或类似工具这样广泛用于开发的系统中完全可行。
-