Object-Oriented Methodology Quiz 14

2024 Fall Semester

21 CST H3Art

Final Score: 97.5/100

1. If a package has a very low RC (Relational Cohesion) value, it may suggest that:

> A. The package is well-designed
>
> B. The package contains highly related types
>
> **C. The package contains unrelated things and is not factored well**
>
> D. The package is very stable

2. The art of architecture involves:

> A. Only following established design patterns
>
> **B. Making skillful choices to resolve architectural factors considering trade-offs**
>
> C. Ignoring business goals and focusing on technical aspects
>
> D. Avoiding any changes to the architecture

3. According to the guideline, where should a family of functionally related interfaces be placed?

> A. In the same package as implementation classes
>
> **B. In a separate package from implementation classes**
>
> C. In a random package
>
> D. In a package with unrelated interfaces

4. If a group of packages have cyclic dependency, what can be done?

> A. Ignore the cyclic dependency
>
> B. Increase the size of the packages
>
> **C. Factor out the types participating in the cycle or break the cycle with an interface**
>
> D. Make the cycle more complex

5. How can the dependency on concrete classes in other packages be reduced?

> A. By using more concrete classes
>
> **B. By using factory objects**
>
> C. By increasing the number of packages
>
> D. By ignoring the dependencies

6. The Layers pattern is an example of:

> A. A way to increase coupling
>
> **B. A mechanism to achieve low coupling and separation of concerns**
>
> C. A design pattern only for small-scale objects
>
> D. A pattern that is not commonly used in architecture

7. If a large package has a subset of frequently modified classes, what should be done?

> A. Ignore the subset and keep the package as it is
>
> **B. Refactor the package into more stable and less stable subsets**

> C. Delete the subset
> D. Move all classes in the subset to other packages

8. A key principle in architectural design that is also applicable at the small-scale object design level is:

> A. High coupling
> B. Low cohesion
> **C. Protected variation**
> D. Complexity

9. When considering architectural decisions, the highest priority is given to:

> A. Business goals
> **B. Inflexible constraints like safety and legal compliance**
> C. All other goals
> D. Technical complexity

10. Architectural analysis in the UP should start:

> A. After the first development iteration
> B. During the first development iteration
> **C. Before the first development iteration**
> D. At the end of the project

11. The refactoring of packages to reduce the impact of unstable subsets is usually:

> A. Done in the very early iterations
> **B. Done incrementally over the elaboration iterations**
> C. Not necessary
> D. Done at the end of the project

12. The main difference between evolution points and variation points is that evolution points:

> A. Are present in the current requirements
> **B. Are speculative and may arise in the future**
> C. Affect the current design more significantly
> D. Are related to functional requirements only

13. Functional cohesion in package design means:

> A. Grouping types randomly
> **B. Grouping types that are related in terms of a common purpose**
> C. Grouping types based on their size
> D. Grouping types based on their location in the source code

14. The RC value is less applicable to packages of:

> A. Mostly implementation classes
> **B. Mostly interfaces**
> C. Small size
> D. Large size

15. In the UP, during which phase is most of the architectural analysis completed?

> A. Inception
> **B. Elaboration**

> C. Transition
> D. Construction

16. Quality scenarios are recommended in architectural factor analysis because they:

> A. Are easy to write
> **B. Define measurable responses and can be verified**
> C. Are required by all architectural methods
> D. Are the only way to describe quality requirements

17. Variation points refer to:

> A. Speculative points of variation in the future
> **B. Variations in the existing current system or requirements**
> C. Changes in the architecture due to new technologies
> D. None of the above

18. Architectural analysis is useful for:

> A. Increasing the complexity of the design
> B. Avoiding alignment with business goals
> **C. Reducing the risk of missing important design aspects**
> D. Focusing only on low priority issues

19. If a subsystem for persistence services has general utility/helper classes that can be used independently, what should be done with these classes?

> A. Keep them in the persistence service package
> **B. Migrate them to a separate package**
> C. Delete them
> D. Ignore them

20. To increase the stability of a package, it can:

> **A. Contain only or mostly interfaces and abstract classes**
> B. Have many dependencies on other unstable packages
> C. Contain highly unstable code
> D. Change frequently

21. The Supplementary Specification in the UP is used to record architectural factors.

> ✅

22. The factor table in architectural analysis is used to document only functional requirements.

> ❌

23. In C++, packages are typically realized as namespaces.

> ✅

24. It is common to work on and release just one class as a unit of development.

> ❌

25. The main focus of architectural analysis is on functional requirements.

> ❌

26. One way to increase package stability is to reduce its dependency on concrete classes in other packages.

> ✅

27. Cyclic dependencies between packages should be avoided.

> ✅

28. The basic principle of package organization is modularization based on functional cohesion.

> ✅

29. In package design, it is desirable for widely depended-upon packages to be very unstable.

> ❌

30. In the UP, the SAD document is fully created before programming begins.

> ❌

31. Variation points refer to speculative points of variation that may arise in the future.

> ❌

32. Packages that contain only or mostly interfaces and abstract classes are more likely to be stable.

> ✅

33. The priority of architectural factors should not reflect the priority of business goals.

> ❌

34. Quality scenarios are recommended for defining measurable responses to quality requirements.

> ✅

35. Architectural analysis in the UP focuses on identifying and resolving non-functional requirements.

> ✅

❌ **36. Internal package coupling, or relational cohesion, can be easily quantified.**

> ✅
>
> Correct Answer: ❌

37. The UP recommends fully identifying all architectural requirements before development begins.

> ❌

38. Architectural analysis is a specialization of requirements analysis.

> ✅

39. The more depended-on packages should be the least stable.

> ❌

40. Architectural decisions are recorded in the SAD document.

> ✅