Harris/L3: Foundations of Software Engineering

Exam 1: Study Guide

Spring 2024

Solution:

- This is the study guide answer key for Exam 1.

Chapter 1

1. Software is a product and can be manufactured using the same technologies used for other engineering artifacts.

   A. True

   B. False

Solution: Section 1.1

2. Software deteriorates rather than wears out because

   A. Software suffers from exposure to hostile environments

   B. Defects are more likely to arise after software has been used often

   C. Multiple change requests introduce errors in component interactions

   D. Software spare parts become harder to order

Solution: Section 1.1

3. Which of the items listed below is not one of the software engineering layers?

   A. Process

   B. Manufacturing

   C. Methods

   D. Tools

Solution: Section 1.3

4. Software engineering umbrella activities are only applied during the initial phases of software development projects.

   A. True

   B. False

Solution: Section 1.4

5. Which of these are the 5 generic software engineering framework activities?

   A. communication, planning, modeling, construction, deployment

   B. communication, risk management, measurement, production, reviewing

   C. analysis, designing, programming, debugging, maintenance

   D. analysis, planning, designing, programming, testing

Solution: Section 1.4

6. Planning ahead for software reuse reduces the cost and increases the value of the systems into which they are incorporated.

   A. True

   B. False

Solution: Section 1.5

9: The essence of software engineering practice might be described as understand the problem, plan a solution, carry out the plan, and examine the result for accuracy.

   A. True

   B. False

Solution: Section 1.5

7. In agile process models the only deliverable work product is the working program.

   A. True

   B. False

Solution: Section 1.6

8. Most software development projects are initiated to try to meet some business need.

   A. True

   B. False

Solution: Section 1.7

Chapter 2

9. Which of the following are recognized process flow types?

   A. Concurrent process flow

   B. Iterative process flow

   C. Linear process flow

   D. Spiral process flow

E. both a and C

Solution: Section 2.1

10. The waterfall model of software development is

   A. A reasonable approach when requirements are well defined.

   B. A good approach when a working program is required quickly.

   C. The best approach to use for projects with large development teams.

   D. An old fashioned model that is rarely used any more.

Solution: Section 2.3.1

11. The incremental model of software development is

   A. A reasonable approach when requirements are well defined.

   B. A good approach when a working core product is required quickly.

   C. The best approach to use for projects with large development teams.

   D. A revolutionary model that is not used for commercial products.

Solution: Section 2.3.2

12. Evolutionary software process models

   A. Are iterative in nature

   B. Can easily accommodate product requirements changes

   C. Do not generally produce throwaway systems

   D. All of the above

Solution: Section 2.3.3

13. The prototyping model of software development is

   A. A reasonable approach when requirements are well defined.

   B. A useful approach when a customer cannot define requirements clearly.

   C. The best approach to use for projects with large development teams.

   D. A risky model that rarely produces a meaningful product.

Solution: Section 2.3.3

14. The spiral model of software development

A. Ends with the delivery of the software product

B. Is more chaotic than the incremental model

C. Includes project risks evaluation during each iteration

D. All of the above

Solution: Section 2.3.3

15. The component-based development model is

A. Only appropriate for computer hardware design.

B. Not able to support the development of reusable components.

C. Dependent on object technologies for support.

D. Not cost effective by known quantifiable software metrics.

Solution: Section 2.4.1

16. The formal methods model of software development makes use of mathematical methods to:

A. Define the specification for computer-based systems

B. Develop defect free computer-based systems

C. Verify the correctness of computer-based systems

D. All of the above

Solution: Section 2.4.2

17. Which of these is not one of the phase names defined by the Unified Process model for software development?

A. Inception phase

B. Elaboration phase

C. Construction phase

D. Validation phase

Solution: Section 2.5.2

18. Which of these is not a characteristic of Personal Software Process?

    A. Emphasizes personal measurement of work product

    B. Practitioner requires careful supervision by the project manager

    C. Individual practitioner is responsible for estimating and scheduling

    D. Practitioner is empowered to control quality of software work products

Solution: Section 2.6.1

19. Which of these are the objectives of Team Software Process?

    A. Accelerate software process improvement

    B. Allow better time management by highly trained professionals

    C. Build self-directed software teams

    D. Show managers how to reduce costs and sustain quality

    E. both b and c

Solution: Section 2.6.2

20. Process technology tools allow software organizations to compress schedules by skipping unimportant activities.

    A. True

    B. False

Solution: Section 2.7

21. It is generally accepted that one cannot have weak software processes and create high quality end products.

    A. True

    B. False

Solution: Section 2.8

Chapter 4

22. Software engineering principles have about a three year half-life.

    A. True

    B. False

Solution: Section 4.1

23. Which of the following is not one of the core principles of software engineering practice?

    A. All design should be as simple as possible, but no simpler

    B. A software system exists only to provide value to its users.

    C. Pareto principle (20

    D. Remember that you produce others will consume

Solution: Section 4.2

24. Every communication activity should have a facilitator to make sure that the customer is not allowed to dominate the proceedings.

    A. True

    B. False

Solution: Section 4.3.1

25. The agile view of iterative customer communication and collaboration is applicable to all software engineering Practice.

    A. True

    B. False

Solution: Section 4.3.1

26. Project plans should not be changed once they are adopted by a team.

    A. True

    B. False

Solution: Section 4.3.2

27. The design model should be traceable to the requirements model?

    A. True

    B. False

Solution: Section 4.3.3

28. Teams using agile software practices do not generally create models.

    A. True

    B. False

Solution: Section 4.3.3

29. Which of the following is not one of the principles of good coding?

    A. Create unit tests before you begin coding

    B. Create a visual layout that aids understanding

    C. Refractor the code after you complete the first coding pass

    D. Write self-documenting code, not program documentation

Solution: Section 4.3.4

30. A successful test are ones that discovers at least one as-yet undiscovered error.

    A. True

    B. False

Solution: Section 4.3.4

31. Which of the following are valid reasons for collecting customer feedback concerning delivered software?

    A. Allows developers to make changes to the delivered increment

    B. Delivery schedule can be revised to reflect changes

    C. Developers can identify changes to incorporate into next increment

    D. All of the above

Solution: Section 4.3.5

Chapter 5

32. Requirements engineering is a generic process that does not vary from one software project to another.

    A. True

    B. False

Solution: Section 5.1

33. During project inception the intent of the of the tasks are to determine

    A. basic problem understanding

    B. nature of the solution needed

    C. people who want a solution

    D. none of the above

    E. A, b and c

Solution: Section 5.1

34. Three things that make requirements elicitation difficult are problems of

    A. budgeting

    B. scope

    C. understanding

    D. Volatility

    E. B, c and D

Solution: Section 5.1

35. It is relatively common for different customers to propose conflicting requirements, each arguing that his or her version is the right one.

    A. True

    B. False

Solution: Section 5.2.2

36. Which of the following is not one of the context-free questions that would be used during project inception?

    A. What will be the economic benefit from a good solution?

B. Who is behind the request for work?

C. Who will pay for the work?

D. Who will use the solution?

Solution: Section 5.2.4

37. In collaborative requirements gathering the facilitator

A. arranges the meeting place

B. can not be a customer

C. controls the meeting

D. must be an outsider

Solution: Section 5.3.1

38. The work products produced during requirement elicitation will vary depending on the

A. size of the budget

B. size of the product being built

C. software process being used

D. stakeholders needs

Solution: Section 5.3.4

39. Developers and customers create use-cases to help the software team understand how different classes of end-users will use functions.

A. True

B. False

Solution: Section 5.4

40. Use-case actors are always people, never system devices.

A. True

B. False

Solution: Section 5.4

41. The result of the requirements engineering task is an analysis model that defines which of the following problem domain(s)?

A. information

B. functional

C. behavioral

D. all of the above

Solution: Section 5.5

42. Analysis patterns facilitate the transformation of the analysis model into a design model by suggesting reliable solutions to common problems.

A. True

B. False

Solution: Section 5.5.2

43. In requirements validation the requirements model is reviewed to ensure its technical feasibility.

A. True

B. False

Solution: Section 5.7

Chapter 8

44. Which of the following are areas of concern in the design model?

A. architecture

B. data

C. interfaces

D. project scope

E. a, b, c

Solution: Section 8.1

45. The importance of software design can be summarized in a single word

A. accuracy

B. complexity

C. efficiency

D. quality

Solution: Section 8.1

46. Which of these are characteristics of a good design?

A. . exhibits strong coupling between its modules

B. implements all requirements in the analysis model

C. includes test cases for all components

D. provides a complete picture of the software

E. both b and d

Solution: Section 8.2.1

47. Which of the following is not a characteristic common to all design methods?

A. configuration management

B. functional component representation

C. quality assessment guidelines

D. refinement heuristics

E. both b and d

Solution: Section 8.2.2

Chapter 17

48. In software quality assurance work there is no difference between software verification and software validation.

A. True

B. False

Solution: Section 17.1.1

49. The best reason for using Independent software test teams is that

A. software developers do not need to do any testing

B. strangers will test the software mercilessly

C. testers do not get involved with the project until testing begins

D. the conflicts of interest between developers and testers is reduced

Solution: Section 17.1.2

50. What is the normal order of activities in which traditional software testing is organized?

A. integration testing

B. system testing

C. unit testing

D. validation testing

E. c, a, d, and b

Solution: Section 17.1.3

51. By collecting software metrics and making use of existing software reliability models it is possible to develop meaningful guidelines for determining when software testing is done.

A. True

B. False

Solution: Section 17.1.4

52. Which of the following strategic issues needs to be addressed in a successful software testing process?

A. conduct formal technical reviews prior to testing

B. specify requirements in a quantifiable manner

C. use independent test teams

D. wait till code is written prior to writing the test plan

E. Both a and B

Solution: Section 17.2

53. Which of the following need to be assessed during unit testing?

A. algorithmic performance

B. code stability

C. error handling

D. execution paths

E. both c and d

Solution: Section 17.3.1

54. Units and stubs are not needed for unit testing because the modules are tested independently of one another.

    A. True

    B. False

Solution: Section 17.3.1

55. Top-down integration testing has as its major advantage(s) that

    A. low level modules never need testing

    B. major decision points are tested early

    C. no drivers need to be written

    D. no stubs need to be written

    E. both b and c

Solution: Section 17.3.2

56. Bottom-up integration testing has as its major advantage(s) that

    A. major decision points are tested early

    B. no drivers need to be written

    C. no stubs need to be written

    D. regression testing is not required

Solution: Section 17.3.2

57. Regression testing should be a normal part of integration testing because as a new module is added to the system new

    A. control logic is invoked

    B. data flow paths are established

    C. drivers require testing

    D. all of the above

    E. both a and b

Solution: Section 17.3.2

58. Smoke testing might best be described as

    A. bulletproofing shrink-wrapped software

    B. rolling integration testing

    C. testing that hides implementation errors

    D. unit testing for small programs

Solution: Section 17.3.2

59. The focus of validation testing is to uncover places that s user will be able to observe failure of the software to conform to its requirements.

    A. True

    B. False

Solution:

Section 17.6

60. Software validation is achieved through a series of tests performed by the user once the software is deployed in his or her work environment.

    A. True

    B. False

Solution:

Section 17.6.1

61. Configuration reviews are not needed if regression testing has been rigorously applied during software integration.

    A. True

    B. False

Solution: Section 17.6.2

62. Acceptance tests are normally conducted by the

    A. developer

    B. end users

C. test team

D. systems engineers

Solution: Section 17.6.3

63. Stress testing examines the pressures placed on the user during system use in extreme environments.

A. True

B. False

Solution: Section 17.7.3

64. Performance testing is only important for real-time or embedded systems.

A. True

B. False

Solution: Section 17.7.4

65. Debugging is not testing, but always occurs as a consequence of testing.

A. True

B. False

Solution: Section 13.7.1