# Sample Exam – Answers
**Sample Exam set B**
**Version 1.7**

# ISTQB® Certified Tester Syllabus Foundation Level
**Compatible with Syllabus version 4.0.1**

## International Software Testing Qualifications Board

# Copyright Notice

# Document Responsibility

The ISTQB® Examination Working Group is responsible for this document.

This document is maintained by a core team from ISTQB® consisting of the Syllabus Working Group and Exam Working Group.

# Acknowledgements

# Revision History

| Version | Date | Remarks |
|---|---|---|
| 1.7 | April 1, 2025 | Correction of usage of Glossary terms |
| 1.6 | July 8, 2024 | Correction of Question #32 |
| 1.5 | May 29, 2024 | Minor correction of Answer: #28 |
| 1.4 | April 12, 2024 | Cleanup of Answer: #2 |
| 1.3 | January 8, 2024 | Bump to match Question document |
| 1.2 | December 5, 2023 | Correction of LO for Question: #15 |
| 1.1 | November 15, 2023 | Bump to match Question document |
| 1.0 | October 16 2023 | First version |

# Table of Contents

# Introduction

## Purpose of this document

The example questions and answers and associated justifications in this sample exam have been created by a team of subject matter experts and experienced question writers with the aim of:

- Assisting ISTQB® Member Boards and Exam Boards in their question writing activities
- Providing training providers and exam candidates with examples of exam questions

These questions cannot be used as-is in any official examination.

**Note**, that real exams may include a wide variety of questions, and this sample exam ***is not*** intended to include examples of all possible question types, styles or lengths, also this sample exam may both be more difficult or less difficult than any official exam.

## Instructions

In this document you may find:

- Answer Key table, including for each correct answer:
    - K-level, Learning Objective, and Point value
- Answer sets, including for all questions:
    - Correct answer
    - Justification for each response (answer) option
    - K-level, Learning Objective, and Point value
- Additional answer sets, including for all questions [does not apply to all sample exams]:
    - Correct answer
    - Justification for each response (answer) option
    - K-level, Learning Objective, and Point value


- *Questions are contained in a separate document*

# Answer Key

| Question Number (#) | Correct Answer | LO | K-Level | Points | Question Number (#) | Correct Answer | LO | K-Level | Points |
|---|---|---|---|---|---|---|---|---|---|
| 1 | d | FL-1.2.1 | K2 | 1 | 21 | d | FL-4.2.2 | K3 | 1 |
| 2 | b | FL-1.2.2 | K1 | 1 | 22 | b | FL-4.2.3 | K3 | 1 |
| 3 | d | FL-1.3.1 | K2 | 1 | 23 | c | FL-4.2.4 | K3 | 1 |
| 4 | a | FL-1.4.1 | K2 | 1 | 24 | b | FL-4.3.1 | K2 | 1 |
| 5 | c | FL-1.4.2 | K2 | 1 | 25 | c | FL-4.3.2 | K2 | 1 |
| 6 | b | FL-1.4.4 | K2 | 1 | 26 | a, e | FL-4.4.2 | K2 | 1 |
| 7 | b | FL-1.5.1 | K2 | 1 | 27 | d | FL-4.4.3 | K2 | 1 |
| 8 | d | FL-1.5.2 | K1 | 1 | 28 | b | FL-4.5.2 | K2 | 1 |
| 9 | b | FL-2.1.1 | K2 | 1 | 29 | d | FL-4.5.3 | K3 | 1 |
| 10 | b | FL-2.1.2 | K1 | 1 | 30 | a | FL-5.1.3 | K2 | 1 |
| 11 | a | FL-2.1.3 | K1 | 1 | 31 | b | FL-5.1.4 | K3 | 1 |
| 12 | b | FL-2.1.4 | K2 | 1 | 32 | b | FL-5.1.5 | K3 | 1 |
| 13 | a | FL-2.2.1 | K2 | 1 | 33 | d | FL-5.1.7 | K2 | 1 |
| 14 | d | FL-2.2.3 | K2 | 1 | 34 | c | FL-5.2.4 | K2 | 1 |
| 15 | b | FL-3.1.3 | K2 | 1 | 35 | a | FL-5.3.1 | K1 | 1 |
| 16 | c | FL-3.2.1 | K1 | 1 | 36 | a | FL-5.3.3 | K2 | 1 |
| 17 | d | FL-3.2.2 | K2 | 1 | 37 | a | FL-5.4.1 | K2 | 1 |
| 18 | c | FL-3.2.3 | K1 | 1 | 38 | b | FL-5.5.1 | K3 | 1 |
| 19 | d | FL-4.1.1 | K2 | 1 | 39 | c | FL-6.1.1 | K2 | 1 |
| 20 | a | FL-4.2.1 | K3 | 1 | 40 | a | FL-6.2.1 | K1 | 1 |

# Answers

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 1 | d | a) Is not correct. It is often possible to use dynamic testing to cause a test object to fail in ways that could never be achieved by the users, such as by using fault injection. However, if the failure can never occur with real end users, then identifying it is not especially valuable as testing is ultimately aimed at improving the work product for the end users. Spending time testing for failures that cannot occur with real users is not an efficient use of a tester's time<br>b) Is not correct. Static testing in the form of static analysis is used by developers to identify defects in their code earlier than can be achieved through dynamic testing. Note, however, that static testing (and static analysis) is used to detect defects, not failures, which are found by dynamic testing. Thus it is the use of the term 'failures' that makes this an incorrect option<br>c) Is not correct. Static analysis directly detects defects in code, and this is normally information for the developer, not the customer.<br>d) Is correct. Reviews are a form of static testing that can be applied from the very start of the software development lifecycle and are used to find defects that can be removed before subsequent development activities waste effort on faulty requirements. If the defects are not detected and removed early on, then when the defect is found in derived work products, such as the design and code, the requirements will need to be changed. | FL-1.2.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 2 | b | a) Is not correct. QA concentrates on process improvement and implementation, using a preventive approach to avoid errors and defects, while testing is a form of QC that is used to detect defects<br>b) Is correct. QC aims to achieve appropriate levels of quality by focusing on identifying and correcting product defects. Testing is a significant part of QC and helps to uncover these defects<br>c) Is not correct. Although testing is a significant part of QC and helps to uncover defects, other (non-testing) techniques utilized in QC include formal methods like model checking and proof of correctness, as well as simulation and prototyping<br>d) Is not correct. QA concentrates on process improvement and implementation, using a preventive approach to avoid errors and defects, while testing is a form of QC that is used to detect defects | FL-1.2.2 | K1 | 1 |

| 3 | d | The 'exhaustive testing is impossible' principle is concerned with the fact that it is not feasible to test every possible variation of inputs in all different circumstances, except in trivial cases. Instead, testing utilizes test techniques, test case prioritization, and risk-based testing to sample from the set of possibilities and focus test efforts.<br><br>a) Is not correct. The principle states that it is not feasible to test everything except in trivial cases. Testing everything would require testing every possible variation of <u>inputs</u> in all different circumstances, which is generally infeasible as there will be a practically infinite number of possibilities. Testing every possible <u>expected result</u> will not address this problem as the relationship between inputs and expected results can be different for each test object. Sometimes there may be a practically infinite number of possible expected results (e.g., when there are several variables representing real numbers), whereas at other times there may be just two expected results, such as with a single variable that can be either true or false<br>b) Is not correct. The principle states that it is not feasible to test every possible variation of inputs in all different circumstances. This is because for non-trivial systems there is a practically infinite number. Therefore, in practice, documenting all possible input variations would be impractical as it would take an infinite length of time<br>c) Is not correct. Starting testing as early as possible with reviews and other static testing approaches will not address the problem of there being too many possible test cases. The 'early testing saves time and money' principle is concerned with fixing defects early on to prevent the occurrence of subsequent defects in derived work products, thereby reducing costs and the likelihood of failures<br>d) Is correct. The use of equivalence partitioning and boundary value analysis to generate test cases is one way to address the principle as these test techniques provide a systematic way to derive a finite subset of all possible test cases | FL-1.3.1 | K2 | 1 |

Page 9 of 37

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 4 | a | a) Is correct. Test design involves using <u>test conditions</u> to create <u>test cases</u> and other necessary testware, such as <u>test data requirements</u> and test charters for exploratory testing. <u>Test environment requirements</u> are also specified, including the necessary infrastructure and tools<br><br>b) Is not correct. Test execution involves executing test cases (as part of test procedures), however it does not directly cover the other testware mentioned in the question, such as test data requirements, test environment requirements and test conditions<br><br>c) Is not correct. Test analysis is used to identify the features that require testing. The test basis is analyzed and defined as test conditions, which are then prioritized along with related risks. While this activity involves working with test conditions, it does not cover the other testware mentioned in the question, such as test data requirements, test environment requirements and test cases<br><br>d) Is not correct. Test implementation includes the generation of test procedures, such as manual and automated test scripts, which are created from test cases and may be assembled into test suites. Test procedures are prioritized and arranged in a test execution schedule. Test data is created, and the test environment built, and its set up verified. While this activity involves explicitly working with test cases, and may use test data requirements and test environment requirements to create test data and the test environment, it does not cover test conditions | FL-1.4.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 5 | c | a) Is not correct. The organization's marketing team is unlikely to perform much testing (although in some organizations they may be involved with acceptance testing), so their average level of experience (most of which would be in marketing) is not likely to impact how testing is performed for a given test object<br><br>b) Is not correct. The level of knowledge of users that a new system is being built for them is unlikely to affect how testing is performed. Any user involvement that could affect how testing is performed is more likely to be as a result of decisions made by the testers, customer and project manager<br><br>c) Is correct. The number of years' experience of the members of the performance efficiency testing team will help to determine the capabilities and knowledge (e.g., of different tools and defect types) that the team members will apply when they are testing<br><br>d) Is not correct. The organizational structure of the different end users (who may be varied) will change between users. So, it may not even be known when the application is being tested, and the end user's organizational structure can thus have little effect on how the testing is performed | FL-1.4.2 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 6 | b | a) Is not correct. Traceability between the mitigated risks and test cases that passed provides little information, because to be mitigated (by testing) the risks would need to have a corresponding test case that passed. To be able to assess residual risk, traceability between all risks and test results needs to be available, so that the risks that do not have a corresponding passing test can be identified as the residual risks<br>b) Is correct. Traceability between user requirements and test results provides an indication of which user requirements have been tested and so provides a means of measuring project progress (in the context of testing) against business goals<br>c) Is not correct. It is not clear that test cases that failed provide an indication of tester's skills any more than test cases that passed. It would partly depend on the test objective (e.g., building confidence or causing failures). Also, such measurement of testers based on test cases that passed and failed can be counter-productive as it could cause the testers to optimize their testing based on that metric rather than the test objective<br>d) Is not correct. Traceability between the identified risks and written test conditions provides a means of determining which further test conditions need to be written. Determining which risks are worth testing is part of risk management, and risk mitigation in particular | FL-1.4.4 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 7 | b | a) Is not correct. Strong communication skills, active listening, and teamwork abilities enable a tester to interact effectively with all stakeholders, however a deep knowledge of a variety of computer games that allowed them to get on well with one developer is not an example of a generic skill useful to testers<br><br>b) Is correct. Domain knowledge that can be used to understand and communicate with end-users and business representatives is one of the generic skills required by testers. A tester with experience as a pilot would make them better able to appreciate the acceptance criteria for the helicopter control system<br><br>c) Is not correct. Although programming skills could be considered as technical knowledge which can increase efficiency when utilizing some test tools, it is unlikely that these skills would improve their communication with business analysts<br><br>d) Is not correct. Although thoroughness, attention to detail, curiosity, and a methodical approach to identifying hard-to-find defects are all useful generic skills for testers, it is doubtful they would be generating test cases prior to starting exploratory testing. This is because one of the main tenets of exploratory testing is that the test cases are generated during the testing, not scripted in advance | FL-1.5.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 8 | d | a) Is not correct. The whole-team approach allows any team member with the requisite skills and knowledge to undertake any task, however that does not mean that team members can take on any role at any time. Typically, they only take on roles in which they are competent, and there is no suggestion that every team member can do every role<br>b) Is not correct. The whole-team approach applies to how a single team (typically in Agile software development) works; it does not cover how multiple teams are supposed to work on larger projects, and it does not suggest that only one 'whole' team is needed for a complete project<br>c) Is not correct. The whole-team approach does not expect every team member to be involved in every important decision. For instance, there is no need for the business representative (i.e., the Product Owner) to be involved in every technical decision that does not affect the business outcome and implementing such an approach would unnecessarily slow down the team's progress<br>d) Is correct. By leveraging the diverse skill sets of each team member most effectively, the whole-team approach fosters superior team dynamics, promotes robust communication and collaboration, and generates a team synergy that benefits the entire project | FL-1.5.2 | K1 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 9 | b | a) Is not correct. In agile software development, deliverables are produced in each iteration, and the frequent delivery of increments necessitates extensive regression testing. Although some (or all) of this regression testing may be automated, the regression testing (automated or not) cannot be replaced by automation of system tests<br>b) Is correct. If a sequential development model is used, then early in the software development lifecycle no code is available for execution, and so during this time static testing (e.g., reviews) is performed. Later in the lifecycle, when code is available for execution, dynamic testing is possible. Note, however, that preparation for dynamic testing will often occur early in any software development lifecycle<br>c) Is not correct. If an iterative development model, like agile software development, is used, then component tests may well be used for regression testing for each iteration. In which case, there is a strong argument for automating these component tests, which will have to be run frequently, and there is unlikely to be a strong argument for developers performing these component tests manually<br>d) Is not correct. In most incremental development models, deliverables are produced in each increment, requiring both static testing and dynamic testing at all test levels for each increment delivered | FL-2.1.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 10 | b | a) Is not correct. Testers should review work products as soon as drafts are available to enable early testing as part of a shift-left approach. If they waited until the next development phase, then unnecessary development (and test) work could be started on unreviewed, flawed work products<br>b) Is correct. Testers should review work products as soon as drafts are available to enable early testing as part of a shift-left approach<br>c) Is not correct. Testers typically review work products that form the test basis as part of test analysis, not before test analysis and test design<br>d) Is not correct. Testers should review work products as soon as drafts are available to enable early testing as part of the shift-left approach. Waiting until they are published means that any defects that could be found by tester's review will be in the published document | FL-2.1.2 | K1 | 1 |
| 11 | a | a) Is correct. Test-Driven Development (TDD) is a well-known example of a test-first approach to development<br>b) Is not correct. Coverage-Driven Development is not a correct example of a test-first approach to development<br>c) Is not correct. Quality-Driven Development is not a correct example of a test-first approach to development<br>d) Is not correct. Feature-Driven Development is not an example of a test-first approach to development, but is, instead, an agile software development methodology based around delivering features (as opposed to user stories in Scrum) | FL-2.1.3 | K1 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 12 | b | a) Is not correct. DevOps enhances testing in several ways, such as by providing fast feedback on code quality, automated regression testing that minimizes regression risk, and promoting a shift-left approach with high-quality code submission and component tests. This is largely provided through continuous integration, where the developers submit component (unit) tests with their new code, which must pass for the code to be admitted to the build. Therefore, developers do need to complete component testing <br> b) Is correct. DevOps enhances testing in several ways, such as by providing fast feedback on code quality, automated regression testing that minimizes regression risk, and promoting a shift-left approach with high-quality code submission and component tests <br> c) Is not correct. DevOps enhances testing in several ways, such as by providing fast feedback on code quality, automated regression testing that minimizes regression risk, and promoting a shift-left approach with high-quality code submission and component tests. Testers do not attempt to treat developers and operations equally by spending more time on release testing, although a shift-right approach to testing (testing in production) may well be used <br> d) Is not correct. Automated processes like continuous integration/continuous delivery (CI/CD) in DevOps facilitate stable test environments and reduce the need for manual testing, however, there is a risk of overlooking the importance of manual testing, especially from a user's perspective | FL-2.1.4 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 13 | a | a) Is correct. System testing examines the behavior and capabilities of the complete system and covers non-functional testing of quality characteristics, which includes security testing. This type of testing is often performed by an independent test team based on system specifications<br>b) Is not correct. System integration testing examines the interfaces with other systems and external services<br>c) Is not correct. Beta testing is a type of acceptance testing performed at an external site by roles outside the development organization<br>d) Is not correct. Component integration testing involves testing the (interfaces and) interactions between components of a system, such as the user interface and database | FL-2.2.1 | K2 | 1 |
| 14 | d | a) Is not correct. Regression tests increase in number as the project progresses, as new regression tests are typically required as changes are made to the system. Similarly, the number of confirmation tests also typically increases as the project progresses as new confirmation tests are needed for each fix made to a system<br>b) Is not correct. It is the other way round. Confirmation tests are created and run when the test object is fixed, and regression tests are (ideally) run whenever the test object is enhanced (changed)<br>c) Is not correct. Confirmation testing verifies that a defect has been fixed correctly and so is concerned with testing changes to the test object. However, regression testing ensures that changes (including changes to the operational environment) do not have negative effects on unchanged software and so does not check that the operational environment remains unchanged<br>d) Is correct. Regression testing ensures that changes do not have negative effects on unchanged software. Confirmation testing verifies that a defect has been fixed – and so is concerned with changed code | FL-2.2.3 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 15 | b | a) Is not correct. A lack of usability provided through the user interface can be detected through a review using a suitable checklist, but the lack of usability can also be identified by getting several typical users to dynamically test the user interface and provide feedback on its usability<br>b) Is correct. A code review can detect code that cannot be reached by any path, however dynamic tests can only exercise reachable code and cannot determine that code cannot be reached without running every possible combination of inputs and input states, which is impractical for real code<br>c) Is not correct. Poor response times for most of the expected users are difficult to determine without executing the code (i.e., by static testing), so in this situation dynamic testing could find a defect, but static testing is unlikely to find it<br>d) Is not correct. A review of the code by someone who is aware of the required features could detect that the required features had not been implemented in the code, and dynamic testing could also be used to determine that these required features had not been implemented | FL-3.1.3 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 16 | c | a) Is not correct. The feedback is from stakeholders (e.g., business representatives and end users), not from developers, so this feedback is not likely to inform managers which developers are more or less productive<br>b) Is not correct. Early and frequent feedback from stakeholders is not used by project managers to prioritize how they interact with the different stakeholders<br>c) Is correct. Obtaining feedback from stakeholders early and often in the software development process can be highly beneficial as it facilitates early communication of potential quality issues, can prevent misunderstandings about requirements, and ensures that any changes in stakeholder requirements are understood and implemented sooner<br>d) Is not correct. Early and frequent feedback can prevent the development of a product that does not meet stakeholder needs, and results in costly rework and missed deadlines, so, ideally there should be no delay. Also, the feedback is from stakeholders (not to them), which includes the end users, so the end users providing feedback will not aid the end users' understanding | FL-3.2.1 | K1 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 17 | d | Considering each of the listed task descriptions:<br>1. The quality characteristics to be evaluated and the exit criteria are selected - (Planning (C): Defining the review scope, purpose, work product to be reviewed, quality characteristics to be evaluated, areas of focus, exit criteria, supporting information such as standards, effort, and timeframes.)<br>2. Everyone has access to the work product - (Review initiation (B): Ensuring all participants have access to the work product and necessary resources, and clarifying their roles and responsibilities.)<br>3. Anomalies are identified in the work product - (Individual review (A): Evaluating the work product's quality, identifying and logging anomalies, recommendations, and questions using review techniques like checklist-based reviewing and scenario-based reviewing.)<br>4. Anomalies are analyzed and discussed - (Communication and analysis (D): Analyzing and discussing each anomaly, determining its status, ownership, and required actions, and making review decisions, normally in a meeting. This could include determining the need for a follow-up review.)<br><br>Thus:<br>a) Is not correct<br>b) Is not correct<br>c) Is not correct<br>d) Is correct. The correct match is: 1C, 2B, 3A, 4D | FL-3.2.2 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 18 | c | Considering each of the listed roles:<br>1. Scribe (or Recorder) - responsible for gathering feedback from reviewers and documenting review information, such as decisions made, and any new anomalies identified during the review meeting. (Records review information, such as decisions and new anomalies found during the review meeting - B)<br>2. Review Leader - responsible for overseeing the review process, such as selecting the review team members, scheduling review meetings, and ensuring that the review is completed successfully. (Takes overall responsibility for the review such as organizing when and where the review will take place - D)<br>3. Facilitator (or Moderator) - responsible for ensuring that the review meetings run effectively, including managing time, mediating discussions, and creating a safe environment where everyone can voice their opinions freely. (Ensures the effective running of review meetings and the setting up of a safe review environment - A)<br>4. Manager - responsible for deciding what needs to be reviewed and allocating resources, such as staff and time, for the review. (Decides what is to be reviewed and provides resources, such as staff and time for the review - C)<br><br>Thus:<br>a) Is not correct<br>b) Is not correct<br>c) Is correct. The correct match is: 1B, 2D, 3A, 4C<br>d) Is not correct | FL-3.2.3 | K1 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 19 | d | a) Is not correct. Decision table testing is a black-box test technique, not a white-box test technique – the test cases are not based on the decisions in the source code. In branch testing, the test cases are derived from knowledge of the control flow of the test object <br><br> b) Is not correct. Anticipation of potential defects is used in error guessing (an experience-based test technique), not in branch testing (a white-box test technique). In decision table testing, the test cases are derived from the specification that describes the business logic <br><br> c) Is not correct. If a test case is based on the knowledge of the control flow of the test object, it is a white-box test technique. Decision table testing is typically based on an analysis of business logic, so it is a black-box test technique. In branch testing, test cases are not derived from the specification – this would make it a black-box test technique. Branch testing is a white-box test technique, where test cases are derived based on the source code structure <br><br> d) Is correct. Decision table testing is a black-box test technique, so it is based on an analysis of the specified behavior of the test object without reference to its internal structure. Therefore, the test cases are independent of how the software is implemented. Branch testing is a white-box test technique, so test cases are based on an analysis of the test object's internal structure and processing. As the test cases are dependent on how the software is designed and coded, they can only be created after the design or implementation of the test object | FL-4.1.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| **20** | a | a) Is correct. 19 covers the "no discount" partition, 20 covers the "50% discount" partition, and 30 covers the "10% discount" partition. These three values cover all three of the valid equivalence partitions<br><br>b) Is not correct. 11 and 12 cover the "no discount" partition, while 20 covers the "50% discount" partition, so covering two of the three valid equivalence partitions<br><br>c) Is not correct. 1 covers the "no discount" partition, while 10 and 50 cover the "10% discount" partition. The "50% discount" partition is not covered, so overall two of the three valid equivalence partitions are covered<br><br>d) Is not correct. 29 and 31 cover the "no discount" partition, while 10 and 30 cover the "10% discount" partition. The "50% discount" partition is not covered, so overall two of the three valid equivalence partitions are covered | FL-4.2.1 | K3 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 21 | d | The domain for the password length has three equivalence partitions:<br>• passwords too short {0, 1, …, 4, 5}<br>• passwords OK {6, 7, …, 11, 12}<br>• passwords too long {13, 14, …}<br><br>To achieve full coverage for 3-value BVA we need to test the following values:<br>    0, 1, 4, 5, 6, 7, 11, 12, 13, 14.<br><br>Since 2-value BVA is already covered, this means that we have already tested the passwords of length:<br>    0, 5, 6, 12 and 13.<br><br>This means that the additional lengths that need to be covered to move from 2-value to 3-value are:<br>    1, 4, 7, 11 and 14.<br><br>Thus:<br>a)  Is not correct<br>b)  Is not correct<br>c)  Is not correct<br>d)  Is correct | FL-4.2.2 | K3 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 22 | b | There are five columns in the decision table. Each test case covers one of them.<br>      TC1 and TC2 both cover Rule 4<br>      TC3 and TC4 both cover Rule 2<br>      TC5 covers Rule 5<br>So, these five test cases cover three out of five columns, achieving a coverage of (3/5)*100% = 60%. Therefore, option b) is the CORRECT option.<br><br>Thus:<br>a)  Is not correct<br>b)  Is correct<br>c)  Is not correct<br>d)  Is not correct | FL-4.2.3 | K3 | 1 |

| 23 | c | | FL-4.2.4 | K3 | 1 |
|---|---|---|---|---|---|



Add [N < 2] / N := N + 1

E2

Add [N = 2] / N := N + 1

E4

START    Add / N := 1    NOT FULL    FULL

E1

E5

E3

Remove / N := N − 1

Remove [N > 0] / N := N − 1

Let us refer to the transitions with E1, …, E5 as in the picture. The variable N denotes the number of elements currently stored. Each "Add" event increases it by 1, and each "Remove" event decreases it by 1. Notice, that when the "Add" event occurs while being in the NOT FULL state, the state changes to FULL only if N=2. If N<2, the system stays in the NOT FULL state. If N=0, no "Remove" action is possible. Similarly, if N=3, no "Add" action is possible.

> Test a) can be written as E1, E3, E3, E2, E4 (so covers 4 out of 5 valid transitions, achieving 80% valid transitions coverage).
> Test b) is infeasible, because after the first three "Add" actions the system is in the FULL state and there is no valid transition going from FULL triggered by the "Add" event. After the first three transitions only 60% of valid transitions coverage is achieved.
> Test c) can be written as E1, E2, E4, E5, E3 (so covers 5 out of 5 valid transitions, achieving 100% valid transitions coverage).
> Test d) can be written as E1, E2, E4, E5, E4 (so covers 4 out of 5 valid transitions, achieving 80% valid transitions coverage).

Thus:
a) Is not correct
b) Is not correct
c) Is correct
d) Is not correct

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 24 | b | a) Is not correct. Coverage is always defined as the percentage of the covered elements. Therefore, it cannot exceed 100% <br><br> b) Is correct. If the statements executed by T1 and T2 were disjoint, the coverage of the test suite {T1, T2} would be 105%, which is impossible (see answer a). Therefore, at least 5% of executable statements must have been executed by both T1 and T2 <br><br> c) Is not correct. Statement coverage does not tell us anything about the number of non-executable statements in the code <br><br> d) Is not correct. Even if a test suite achieves full statement coverage, this does not imply achieving full branch coverage | FL-4.3.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 25 | c | Branch testing is a white-box test technique in which the coverage items are branches. A branch is a transfer of control between two nodes in the control flow graph, which shows the possible sequences in which source code statements are executed in the test object. Each transfer of control can be either unconditional (i.e., straight-line code) or conditional (i.e., a decision outcome). Coverage is measured as the number of branches exercised by the test cases divided by the total number of branches, and is expressed as a percentage.<br><br>Thus:<br>a) Is not correct. A decision outcome is a conditional branch. For branch testing, X counts not only conditional, but also unconditional branches<br>b) Is not correct. Branch coverage counts not only conditional, but also unconditional branches<br>c) Is correct. Branch coverage is measured as the number of branches exercised by the test cases divided by the total number of branches, and is expressed as a percentage<br>d) Is not correct. Both X and Y count only conditional branches and do not take into account the unconditional branches | FL-4.3.2 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 26 | a, e | Exploratory testing is useful when there are few or inadequate specifications or there is significant time pressure on the testing. Exploratory testing is also useful to complement other more formal test techniques. Exploratory testing will be more effective if the tester is experienced, has domain knowledge and has a high degree of essential skills, like analytical skills, curiosity and creativeness.<br><br>Thus:<br>a) Is correct. Exploratory testing is useful when there are few or inadequate specifications or there is significant time pressure on the testing<br>b) Is not correct. Exploratory testing is not a black-box test technique<br>c) Is not correct. Exploratory testing is useful when the specifications are poorly written<br>d) Is not correct. Programming skills have nothing to do with exploratory testing in principle<br>e) Is correct. Exploratory testing will be more effective if the tester is experienced, has domain knowledge and has a high degree of essential skills, like analytical skills, curiosity and creativeness | FL-4.4.2 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 27 | d | a) Is not correct. Checklists should contain test conditions to be verified. This is an example of an error, not a test condition; even if the tester was able to deduce some potential test conditions from the examples of errors, this error description is too general<br>b) Is not correct. Checklists should not contain items that are better suited as exit criteria. This is an example of an exit criterion<br>c) Is not correct. Checklists should not contain items that are too general. This is a very general item, which practically describes a test objective<br>d) Is correct. This is an example of a test condition that can be checked by a human | FL-4.4.3 | K2 | 1 |
| 28 | b | a) Is not correct. The rule-oriented format includes formats like bullet point verification lists or tabulated forms of input-output mappings, explicitly showing the rules to be followed. Given/When/Then is a scenario-oriented format because it describes a scenario to be verified<br>b) Is correct. This is a Given/When/Then format, which is scenario-oriented<br>c) Is not correct. There is no "product-oriented" format of acceptance criteria<br>d) Is not correct. There is no "process-oriented" format of acceptance criteria | FL-4.5.2 | K2 | 1 |
| 29 | d | a) Is not correct. The test case is related to viewing previous orders in the order history<br>b) Is not correct. The test case is related to viewing previous orders<br>c) Is not correct. The test case is related to viewing previous orders in the order history<br>d) Is correct. The test case is related to the registration process, which is not discussed in the user story. The user story is about viewing previous orders | FL-4.5.3 | K3 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| **30** | a | a) Is correct. This is something that can (and should) be checked *before* the code is submitted to version control<br>b) Is not correct. This is something that can be checked *after* step (2) is performed, because merge conflict reporting can be done *after* the code is submitted and merged<br>c) Is not correct. This fits better as the entry criterion for step (3)<br>d) Is not correct. This fits better as the exit criterion for step (3) | FL-5.1.3 | K2 | 1 |
| **31** | b | The average development effort is $900,000 and the average test effort is $90,000 (calculated from the four projects).<br>The average test-to-development effort ratio is 1:10 ($90,000 : $900,000), which means that historically, on average, the test effort is 10% of the development effort.<br>So if the development effort is estimated to be $800,000, the estimated test effort is estimated as:<br>$$10\% * \$800,000 = 0.1 * \$800,000 = \$80,000.$$<br>Thus:<br>a) Is not correct<br>b) Is correct<br>c) Is not correct<br>d) Is not correct | FL-5.1.4 | K3 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| **32** | b | The logical dependencies mean that for each product you have to run SEARCH → VIEW → ADD before running ORDER. You can add more products (using the same flow), before you run ORDER. Based on this, TC1 or TC2 must be executed first, otherwise no progress can be made. The first priority should be given to VIEW and ADD product B, as its test cases (TC6, TC4) are assigned with higher priority. So, the first 3 tests to execute are TC2 -> TC4 -> TC6<br><br>Now we need to consider whether to run TC7 and then the entire flow for product A or run the TCs for product A first. If TC7 has lower priority than the other tests, they should be tested first. Therefore, the entire flow should be: TC2 -> TC4 -> TC6 -> TC1 -> TC3 -> TC5 -> TC7<br><br>a)   Is not correct. TC1 must be executed before TC3<br>b)   Is correct<br>c)   Is not correct. As shown above, TC7 is the last to be executed.<br>d)   Is not correct. Product B must be executed before product A | FL-5.1.5 | K3 | 1 |
| **33** | d | a)   Is not correct. Usability testing is business facing testing that critiques the product (Q3)<br>b)   Is not correct. Functional testing is business facing testing (Q2)<br>c)   Is not correct. User acceptance testing is business facing testing that critiques the product (Q3)<br>d)   Is correct. Component integration testing is technology facing testing that supports the team (guides the development) (Q1) | FL-5.1.7 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| **34** | c | Considering each of the listed risks and their mitigations: <br> 1. Long system responses (1) can be tested in performance testing (B) <br> 2. Changes in consumers' preferences (2) are usually out of our control, so usually we accept this risk (A) <br> 3. Flooding of the server room (3) can cause significant loss, so we should transfer the risk, e.g., by buying an insurance policy (D) <br> 4. That patients above a certain age receive inaccurate reports (4) suggests a potential boundary problem, which can be effectively detected with test techniques like BVA (C) <br><br> Thus: <br> a) Is not correct <br> b) Is not correct <br> c) Is correct. The correct combinations of risk and mitigation are: 1B, 2A, 3D and 4C <br> d) Is not correct | FL-5.2.4 | K2 | 1 |
| **35** | a | a) Is correct. Product quality metrics measure quality characteristics. Mean time to failure measures maturity, so it is a product quality metric <br> b) Is not correct. This is an example of a defect metric, not a product quality metric <br> c) Is not correct. This is an example of a coverage metric, not a product quality metric <br> d) Is not correct. This is an example of a defect metric, not a product quality metric | FL-5.3.1 | K1 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 36 | a | a) Is correct. The client is in a different location and time zone, so it may be difficult to communicate face-to-face<br>b) Is not correct. Dashboards are usually available to any user at any time, so the difference in time zones will not be as much of a hindrance to communication as verbal, face-to-face communication<br>c) Is not correct. Although the time difference between Europe and America is several hours, and this may cause some inconvenience, it's certainly not as great as with communicating face-to-face<br>d) Is not correct. Video conferencing tools are a convenient means of communication. Although communication between Europe and America during working hours usually requires one party to connect in the very early or very late hours, this is not as much of an inconvenience as verbal, face-to-face communication | FL-5.3.3 | K2 | 1 |
| 37 | a | a) Is correct. For a complex configuration item (e.g., a test environment), configuration management (CM) records the items it consists of, their relationships, and versions<br>b) Is not correct. CM tools do not execute test cases and do not calculate coverage<br>c) Is not correct. A CM tool is not a licensed management tool<br>d) Is not correct. CM tools do not generate test data | FL-5.4.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 38 | b | a) Is not correct. While the sentence is true, it does not provide much value for the developer<br>b) Is correct. From the test results it seems that the system ignores duplicates and sorts the list disregarding the repetitions. This is probably the cause of failures in TC3, TC4, TC5. Such information may help the developer to find the defect and fix it more efficiently<br>c) Is not correct. The system does not fail in sorting negative numbers. The problem is rather in disregarding duplicates<br>d) Is not correct. The test cases TC3, TC4 and TC5 fail, but we aren't aware that the test cases have any defects | FL-5.5.1 | K3 | 1 |
| 39 | c | Considering each of the listed tool categories and their descriptions:<br>  A. Static testing tools – support the tester in performing reviews and static analysis (4)<br>  B. Tools supporting scalability and deployment standardization – For example, virtual machines, containerization tools (3)<br>  C. DevOps tools – support the DevOps delivery pipeline, workflow tracking, automated build process(es), continuous integration/continuous delivery (CI/CD) (1)<br>  D. Collaboration tools – facilitate communication (2)<br><br>Thus:<br>a) Is not correct<br>b) Is not correct<br>c) Is correct. The correct match is: 1C, 2D, 3B, 4A<br>d) Is not correct | FL-6.1.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---|---|---|---|---|---|
| 40 | a | a) Is correct. Test automation can provide measures that are too complicated for humans to derive, such as white-box testing coverage measures for all but the most trivial code<br>b) Is not correct. By using test tools, the responsibility for the testing is NOT shared with the tool vendor as the vendor is not involved in the testing, and it is the tester's responsibility. The only possible responsibility that could be assigned to the tool vendor is if the tool fails to work as expected and provides incorrect test results<br>c) Is not correct. Testers still need to apply critical thinking when analyzing anomalies in the test results to determine their likely cause<br>d) Is not correct. Neither testers nor tools can generate test cases simply from an analysis of the program code as the code is the implementation and provides no information on the expected results, which will need to come from another part of the test basis, such as the design specification | FL-6.2.1 | K1 | 1 |