

Sample Exam – Answers

Sample Exam set D
Version 1.0

ISTQB® Certified Tester Syllabus Foundation Level

Compatible with Syllabus version 4.0

International Software Testing Qualifications Board



Copyright Notice

Copyright Notice © International Software Testing Qualifications Board (hereinafter called ISTQB®).

ISTQB® is a registered trademark of the International Software Testing Qualifications Board.

All rights reserved.

The authors hereby transfer the copyright to the ISTQB®. The authors (as current copyright holders) and ISTQB® (as the future copyright holder) have agreed to the following conditions of use:

Extracts, for non-commercial use, from this document may be copied if the source is acknowledged.

Any Accredited Training Provider may use this sample exam in their training course if the authors and the ISTQB® are acknowledged as the source and copyright owners of the sample exam and provided that any advertisement of such a training course is done only after official Accreditation of the training materials has been received from an ISTQB®-recognized Member Board.

Any individual or group of individuals may use this sample exam in articles and books, if the authors and the ISTQB® are acknowledged as the source and copyright owners of the sample exam.

Any other use of this sample exam is prohibited without first obtaining the approval in writing of the ISTQB®.

Any ISTQB®-recognized Member Board may translate this sample exam provided they reproduce the abovementioned Copyright Notice in the translated version of the sample exam.

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

This document was produced by a core team from the ISTQB®: Stuart Reid and Adam Roman

The core team thanks the Exam Working Group review team, the Syllabus Working Group and Member Boards for their suggestions and input.

Revision History

Sample Exam – Answers Layout Template used: Version 2.11 Date: October 16, 2023

| Version | Date | Remarks |
|---------|------------------|---------------|
| 1.0 | October 16, 2023 | First version |
| | | |
| | | |

Table of Contents

| | |
|--------------------------------|----|
| Copyright Notice | 2 |
| Document Responsibility | 2 |
| Acknowledgements | 2 |
| Revision History | 3 |
| Table of Contents | 4 |
| Introduction | 5 |
| Purpose of this document | 5 |
| Instructions | 5 |
| Answer Key | 6 |
| Answers | 7 |
| 1 | 7 |
| 2 | 7 |
| 3 | 8 |
| 4 | 9 |
| 5 | 10 |
| 6 | 11 |
| 7 | 11 |
| 8 | 12 |
| 9 | 13 |
| 10 | 13 |
| 11 | 14 |
| 12 | 14 |
| 13 | 15 |
| 14 | 15 |
| 15 | 16 |
| 16 | 16 |
| 17 | 17 |
| 18 | 18 |
| 19 | 18 |
| 20 | 19 |
| 21 | 19 |
| 22 | 20 |
| 23 | 21 |
| 24 | 22 |
| 25 | 23 |
| 26 | 23 |
| 27 | 24 |
| 28 | 24 |
| 29 | 25 |
| 30 | 25 |
| 31 | 26 |
| 32 | 26 |
| 33 | 27 |
| 34 | 27 |
| 35 | 27 |
| 36 | 28 |
| 37 | 29 |
| 38 | 30 |
| 39 | 31 |
| 40 | 32 |

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 |
|---------------------|----------------|----------|---------|--------|
| 1 | d | FL-1.1.1 | K1 | 1 |
| 2 | c | FL-1.2.3 | K2 | 1 |
| 3 | a | FL-1.3.1 | K2 | 1 |
| 4 | b | FL-1.4.1 | K2 | 1 |
| 5 | a | FL-1.4.3 | K2 | 1 |
| 6 | d | FL-1.4.5 | K2 | 1 |
| 7 | a | FL-1.5.2 | K1 | 1 |
| 8 | b | FL-1.5.3 | K2 | 1 |
| 9 | a | FL-2.1.2 | K1 | 1 |
| 10 | a | FL-2.1.3 | K1 | 1 |
| 11 | d | FL-2.1.4 | K2 | 1 |
| 12 | b | FL-2.1.6 | K2 | 1 |
| 13 | a | FL-2.2.2 | K2 | 1 |
| 14 | b | FL-2.3.1 | K2 | 1 |
| 15 | c | FL-3.1.1 | K1 | 1 |
| 16 | c | FL-3.1.2 | K2 | 1 |
| 17 | b | FL-3.2.2 | K2 | 1 |
| 18 | b | FL-3.2.3 | K1 | 1 |
| 19 | b | FL-4.1.1 | K2 | 1 |
| 20 | b, e | FL-4.2.1 | K3 | 1 |

| Question Number (#) | Correct Answer | LO | K-Level | Points |
|---------------------|----------------|----------|---------|--------|
| 21 | c | FL-4.2.2 | K3 | 1 |
| 22 | a | FL-4.2.3 | K3 | 1 |
| 23 | b | FL-4.2.4 | K3 | 1 |
| 24 | c | FL-4.3.1 | K2 | 1 |
| 25 | a | FL-4.3.3 | K2 | 1 |
| 26 | c | FL-4.4.1 | K2 | 1 |
| 27 | d | FL-4.4.2 | K2 | 1 |
| 28 | d | FL-4.5.1 | K2 | 1 |
| 29 | a | FL-4.5.3 | K3 | 1 |
| 30 | b, d | FL-5.1.3 | K2 | 1 |
| 31 | a | FL-5.1.4 | K3 | 1 |
| 32 | b | FL-5.1.5 | K3 | 1 |
| 33 | c | FL-5.1.7 | K2 | 1 |
| 34 | b | FL-5.2.1 | K1 | 1 |
| 35 | b, e | FL-5.2.2 | K2 | 1 |
| 36 | c | FL-5.3.2 | K2 | 1 |
| 37 | d | FL-5.4.1 | K2 | 1 |
| 38 | a | FL-5.5.1 | K3 | 1 |
| 39 | b | FL-6.1.1 | K2 | 1 |
| 40 | c | FL-6.2.1 | K1 | 1 |

Answers

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---------------------|----------------|---|-------------------------|---------|------------------|
| 1 | d | <p>a) Is not correct. Finding and fixing defects in the test object is not a typical test objective as although identifying defects is an objective of testing, fixing defects is not a testing activity</p> <p>b) Is not correct. Maintaining effective communications with developers is not a typical test objective as although it is useful in achieving other objectives of testing, such as providing stakeholders with information that enables them to make informed decisions, it is not a primary reason for performing testing</p> <p>c) Is not correct. Validating that legal requirements have been met is not a typical test objective because validation is concerned with checking whether the system meets users' and other stakeholders' needs in its operational environment. Checking that legal requirements have been met is a form of verification</p> <p>d) Is correct. Building confidence in the quality of the test object is achieved by executing tests that pass</p> | FL-1.1.1 | K1 | 1 |
| 2 | c | <p>a) Is not correct. The miscalculation of bonuses is a failure by the system, not a defect</p> <p>b) Is not correct. The system not suitably supporting disabled users is a failure which eventually results in a fine, but the fine itself is not a failure (it appears to be the correct functioning of the regulatory system)</p> <p>c) Is correct. The error is made by the programmer and this mistake is caused by them working under severe time pressure, which is the root cause of the subsequent defect</p> <p>d) Is not correct. The poor design of the user interface, which does not suitably address disabled users, is a design defect caused by the designer error. Thus the design of the user interface includes a design defect not a designer error</p> | FL-1.2.3 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---------------------|----------------|---|-------------------------|---------|------------------|
| 3 | a | <p>a) Is correct. The ‘tests wear out’ principle is concerned with the idea that repeating identical tests on unaltered code is unlikely to uncover novel defects and therefore, modifying tests may be essential. By using high-level test conditions to generate new tests each time, the tests will not be identical and should not ‘wear out’</p> <p>b) Is not correct. The ‘absence-of-defects fallacy’ principle is concerned with ensuring that users’ needs are fulfilled even if lots of testing is done and no defects are found (i.e., validation is also necessary). The use of high-level test conditions to generate test cases and execute tests does not directly address this concern</p> <p>c) Is not correct. 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. This is typically addressed by starting testing (both static and dynamic) as early as possible, but this is not addressed by using high-level test conditions to generate test cases and execute tests</p> <p>d) Is not correct. The ‘Defects cluster together’ principle is concerned with the distribution of defects in a system, which typically follows a Pareto distribution. The use of high-level test conditions to generate test cases and execute tests does not address this concern, which is typically addressed by risk-based testing</p> | FL-1.3.1 | K2 | 1 |

| Question Number (#) | Correct Answer | Explanation / Rationale | Learning Objective (LO) | K-Level | Number of Points |
|---------------------|----------------|---|-------------------------|---------|------------------|
| 4 | b | <p>Considering each of the listed test activities and their tasks:</p> <ul style="list-style-type: none"> A. Test analysis - 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. During this test analysis, defects in the test basis are typically uncovered, and the test object's testability may also be assessed. (Task 4) B. Test design - Involves using test conditions to create test cases and other necessary testware, such as test data requirements and test charters for exploratory testing. (Task 1) C. Test implementation - Test procedures, such as manual and automated test scripts, are created from test cases and may be assembled into test suites. Test procedures are prioritized and arranged in a test execution schedule. (Task 3) D. Test completion - Occurs at project milestones, such as release, end of iteration or end of test level. Testware is identified and archived or handed to the appropriate teams for reuse, the test environment is shut down, and the test activities are analyzed for lessons learned and future improvements. (Task 2) <p>Thus:</p> <ul style="list-style-type: none"> a) Is not correct b) Is correct. The CORRECT match is: 1B, 2D, 3C, 4A c) Is not correct d) Is not correct | FL-1.4.1 | K2 | 1 |

Is not correct

| | | | | | |
|---|---|---|----------|----|---|
| 5 | a | <p>Considering each of the listed testware, and the test activity that produces it:</p> <ul style="list-style-type: none"> i. The test completion report is an output of the test completion activity ii. Data held in a database used for test inputs and expected results is the test data - output of test implementation activity iii. The list of elements needed to build the test environment is the test environment requirements - output of test design activity. iv. Documented sequences of test cases in execution order are the test procedures - output of test implementation activity v. Test cases - output of test design activity <p>Test implementation produces the following outputs: test procedures (iv), automated test scripts, test suites, test data (ii), test execution schedule, and test environment elements such as stubs, drivers, simulators, and service virtualizations.</p> <p>Thus:</p> <ul style="list-style-type: none"> a) Is correct. Items ii and iv in the list are produced as a result of test implementation b) Is not correct c) Is not correct d) Is not correct | FL-1.4.3 | K2 | 1 |
|---|---|---|----------|----|---|

| | | | | | |
|---|---|---|----------|----|---|
| 6 | d | <p>a) Is not correct. The testing role is primarily responsible for the technical and engineering aspects of testing, such as test analysis, test design, test implementation, and test execution. Evaluating the test basis for defects and the test object for testability are tasks performed as part of test analysis, so it is likely they are tasks performed by the testing role</p> <p>b) Is not correct. The testing role is primarily responsible for the technical and engineering aspects of testing, such as test analysis, test design, test implementation, and test execution. Defining the test environment requirements is a task performed as part of test design, so it is likely to be a task performed by the testing role</p> <p>c) Is not correct. The testing role is primarily responsible for the technical and engineering aspects of testing, such as test analysis, test design, test implementation, and test execution. Assessing the testability of a test object is a task performed as part of test analysis, so it is likely to be a task performed by the testing role</p> <p>d) Is correct. The test management role primarily involves activities related to test planning, test monitoring and control, and test completion. Thus, creating the test completion report, which is the prime output from the test completion activity, is likely to be a task performed by the test management role</p> | FL-1.4.5 | K2 | 1 |
| 7 | a | <p>a) Is correct. The whole team approach promotes robust communication and collaboration between the team members</p> <p>b) Is not correct. While the whole team approach prioritizes collective accountability for quality, each individual team member is still equally accountable for quality</p> <p>c) Is not correct. The whole team approach is about how the team works together, with the aim of higher quality deliverables, but it does not necessarily result in faster deployment to end users</p> <p>d) Is not correct. When using the whole team approach, testers work with business representatives to create acceptance tests. There is no suggestion that the approach will reduce collaboration with external business users</p> | FL-1.5.2 | K1 | 1 |

| | | | | | |
|---|---|--|----------|----|---|
| 8 | b | <p>Considering each of the listed benefits and drawbacks of the independence of testing:</p> <ul style="list-style-type: none"> i. The testers work in a different location from the developers – Ideally, we want close collaboration between testers and developers, which is not increased by isolation. Thus, this is a disadvantage ii. Testers question the assumptions programmers make while writing code - Testers and developers have varied backgrounds, technical viewpoints, and potential biases, allowing testers to usefully challenge assumptions made by stakeholders during system specification and implementation. Thus, this is an advantage iii. A confrontational dynamic has been established between testers and developers - The main disadvantage of independence in testing is that testers may become isolated from the development team, leading to communication problems, a lack of collaboration, and potentially an adversarial relationship, with testers being blamed for delays and bottlenecks in the release process. Thus, this is a disadvantage iv. Developers have convinced themselves that testers are mostly accountable for quality – One of the disadvantages of independence in testing is that testers may become isolated from the development team, leading to developers feeling less accountable for quality. Thus, this is a disadvantage v. Testers have different biases than those held by the developers - The primary benefit of independence in testing is that testers are more likely to identify different types of failures and defects compared to developers, due to their varied backgrounds, technical viewpoints, and potential biases, including cognitive bias <p>Thus:</p> <ul style="list-style-type: none"> a) Is not correct b) Is correct. The list entries showing benefits are ii and v | FL-1.5.3 | K2 | 1 |
|---|---|--|----------|----|---|

| | | | | | |
|-----------|---|---|----------|----|---|
| | | <p>c) Is not correct</p> <p>d) Is not correct</p> | | | |
| 9 | a | <p>a) Is correct. Each test level has specific and distinct test objectives as a different form of test object (e.g., single component, complete system) is tested at each test level and overlapping test objectives would lead to unnecessary duplication</p> <p>b) Is not correct. Test analysis and design for a given test level should start during the corresponding development phase to facilitate early testing (e.g., acceptance test analysis and design should begin during requirements analysis). Test implementation will generally start later, and test execution will start during the test level</p> <p>c) Is not correct. Test design for a given test level should start during the corresponding development phase to facilitate early testing, however test design (e.g., test case generation) needs to be based on an agreed test basis, not an early draft, otherwise significant test effort may be wasted on creating test cases for a design that later changes</p> <p>d) Is not correct. Quality control applies to all development activities, meaning that every software development activity has a corresponding test activity. However, the same symmetry does not apply to dynamic and static testing. There are some static testing activities (e.g., static analysis) for which there is no obvious corresponding dynamic testing activity</p> | FL-2.1.2 | K1 | 1 |
| 10 | a | <p>a) Is correct. Behavior-Driven Development (BDD) is a well-known example of a test-first approach to development</p> <p>b) Is not correct. Test Level Driven Development is not a correct example of a test-first approach to development</p> <p>c) Is not correct. Function-Driven Development is not a correct example of a test-first approach to development</p> <p>d) Is not correct. Performance-Driven Development is not a correct example of a test-first approach to development</p> | FL-2.1.3 | K1 | 1 |

| | | | | | |
|----|---|---|----------|----|---|
| 11 | d | <p>a) Is not correct. DevOps generally increases the visibility of non-functional quality characteristics, such as performance and reliability</p> <p>b) Is not correct. Automated processes like continuous integration/continuous delivery (CI/CD) used in DevOps facilitate stable test environments</p> <p>c) Is not correct. Automated processes like CI/CD used in DevOps generally reduce the need for manual testing</p> <p>d) Is correct. DevOps implementation can pose several risks and challenges, including the need to define and set up the delivery pipeline, introduce and maintain CI/CD tools, and establish and maintain test automation</p> | FL-2.1.4 | K2 | 1 |
| 12 | b | <p>a) Is not correct. The benefits of retrospectives include team bonding and learning from sharing issues, and better collaboration between developers and testers through reviewing and improving working practices. Calling out individuals who a team member may feel did not fully contribute to achieving quality as required by the whole-team approach will not contribute to this team bonding and collaboration</p> <p>b) Is correct. During the retrospective, the group discusses what aspects of the project were successful and should be retained, as well as areas that could be improved, and how to do so</p> <p>c) Is not correct. The benefits of retrospectives are based on increased effectiveness and efficiency through process improvements; they are not an opportunity to let off steam and criticize management and customers. Also, the results are recorded, usually in the test completion report, so anything said in the meeting could be read by other stakeholders</p> <p>d) Is not correct. Retrospectives are meetings that are typically held at the end of an iteration where team members will focus on discussing quality-related issues that have occurred in the current iteration. They are not used for making plans or technical decisions for the next iteration; this would be done in the iteration planning meeting at the start of the next iteration</p> | FL-2.1.6 | K2 | 1 |

| | | | | | |
|----|---|--|----------|----|---|
| 13 | a | <p>a) Is correct. Checking that the sort function puts the elements of the list or array in ascending order is evaluating the functional correctness of the sort function, which is part of functional testing</p> <p>b) Is not correct. Assessing whether the sort function meets its non-functional requirement to complete within one second is part of testing its performance efficiency, which is part of non-functional testing</p> <p>c) Is not correct. Evaluating the ease with which the sort function can be modified from sorting ascending to sorting descending is testing its modifiability, a form of non-functional maintainability testing, which is part of non-functional testing</p> <p>d) Is not correct. Assessing that the sort function still functions correctly when moved from a 32-bit to a 64-bit architecture is testing its adaptability, a form of portability testing, which is part of non-functional testing</p> | FL-2.2.2 | K2 | 1 |
| 14 | b | <p>a) Is not correct. Assuming that testers could check the ease of changing the currency exchange system then it would be done by maintainability testing rather than maintenance testing, so this is not a trigger for maintenance testing</p> <p>b) Is correct. A system modification (such as a fix or enhancement) is an example of a trigger for maintenance testing. The removal of the refund option of the currency exchange system was a fix that would lead to maintenance testing</p> <p>c) Is not correct. If the agile team has started developing a user story that adds a new customer loyalty feature to the currency exchange system, then this will result in them testing the new feature, and then they would perform regression testing. No maintenance testing is required in this situation</p> <p>d) Is not correct. Reconfiguration of the currency exchange system to support both the local language and English currency transactions is not a system modification, a change to the operational environment, or a system retirement, which are the three triggers for maintenance testing</p> | FL-2.3.1 | K2 | 1 |

| | | | | | |
|----|---|---|----------|----|---|
| 15 | c | <p>a) Is not correct. Most work products can be examined using some form of static testing, and a contract must be interpretable by humans and so could be reviewed, which is a form of static testing</p> <p>b) Is not correct. Most work products can be examined using some form of static testing, and a test plan must be interpretable by humans and so could be reviewed, which is a form of static testing</p> <p>c) Is correct. Most work products can be examined using some form of static testing; however it is not suitable for work products that are too complex for human interpretation and should not be analyzed by tools, and encrypted code is too complex for humans and if it is properly encrypted it will not be analyzable by most tools</p> <p>d) Is not correct. Most work products can be examined using some form of static testing, and a test charter must be interpretable by humans and so could be reviewed, which is a form of static testing</p> | FL-3.1.1 | K1 | 1 |
| 16 | c | <p>a) Is not correct. There are some defect types that can be found by both static testing and dynamic testing, such as a programming defect that can be observed by a reviewer in a code review and which causes an observable failure during dynamic testing</p> <p>b) Is not correct. There are some defect types that can only be detected by static testing, such as unreachable code, design patterns not implemented as desired and defects in non-executable work products</p> <p>c) Is correct. There are some defect types that can be found by both static testing and dynamic testing, such as a programming defect that can be observed by a reviewer in a code review and which causes an observable failure during dynamic testing. There are also some defect types that can only be detected by static testing, such as unreachable code, design patterns not implemented as desired and defects in non-executable work products</p> <p>d) Is not correct. There are some defect types that can only be detected by dynamic testing, such as performance issues or memory issues that can only be observed when executing the code or system</p> | FL-3.1.2 | K2 | 1 |

| | | | | | |
|----|---|--|----------|----|---|
| 17 | b | <p>The five listed descriptions and the corresponding review process activities are:</p> <ol style="list-style-type: none">1. This describes part of the 'communication and analysis' activity2. This describes part of the 'fixing and reporting' activity3. This describes part of the 'individual review' activity4. This describes part of the 'planning' activity5. This describes part of the 'review initiation' activity. <p>The generic review process from ISO/IEC 20246, which is outlined in the syllabus, comprises the following activities in this logical order:</p> <ul style="list-style-type: none">• Planning (4)• Review initiation (5)• Individual review (3)• Communication and analysis (1)• Fixing and reporting (2) <p>Thus:</p> <ol style="list-style-type: none">a) Is not correctb) Is correct. The correct sequence of activities is: 4 – 5 – 3 – 1 – 2c) Is not correctd) Is not correct | FL-3.2.2 | K2 | 1 |
|----|---|--|----------|----|---|

| | | | | | |
|----|---|--|----------|----|---|
| 18 | b | <p>a) Is not correct. The manager is responsible for deciding what needs to be reviewed and allocating resources, such as staff and time, for the review</p> <p>b) Is correct. The moderator (or facilitator) is 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</p> <p>c) Is not correct. The chairperson is not a recognized role in reviews</p> <p>d) Is not correct. The review leader is responsible for overseeing the review process, such as selecting the review team members, scheduling review meetings, and ensuring that the review is completed successfully</p> | FL-3.2.3 | K1 | 1 |
| 19 | b | <p>a) Is not correct. The document does not refer to the test object's internal structure but specifies the desired behavior of the test object. Therefore, white-box test techniques will not be helpful in designing test cases</p> <p>b) Is correct. The document is a requirement that specifies the desired behavior of the test object. Therefore, the most suitable test techniques in this case are the black-box test techniques (e.g., Boundary Value Analysis or Decision Table Testing)</p> <p>c) Is not correct. Although experience-based test techniques can be used to design test cases based on this document, black-box test techniques will be more suitable. The document describes a precise business rule and, in addition, wording like "exceeds \$100" suggests the existence of important equivalence partition boundaries that should be tested using black-box test techniques like boundary value analysis</p> <p>d) Is not correct. Risk-based test techniques are not a recognized type of test technique</p> | FL-4.1.1 | K2 | 1 |

| | | | | | |
|----|------|---|----------|----|---|
| 20 | b, e | <p>There are two equivalence partitions that are not yet covered, which correspond to “student discount” and “pensioner discount”.</p> <p>a) Is not correct. $CY - BY = 64$, so these inputs correspond to the already covered “no discount” partition</p> <p>b) Is correct. $CY - BY = 65$, so these inputs correspond to a partition that is not yet covered (“pensioner discount”)</p> <p>c) Is not correct. $CY - BY = -65$, so these inputs correspond to the already covered “error message” partition</p> <p>d) Is not correct. $CY - BY = 18$, so these inputs correspond to the already covered “no discount” partition</p> <p>e) Is correct. $CY - BY = 0$, so these inputs correspond to a partition that is not yet covered (“student discount”)</p> | FL-4.2.1 | K3 | 1 |
| 21 | c | <p>There are three equivalence partitions: $\{\dots, -2, -1\}$, $\{0, 1, 2\}$, $\{3, 4, \dots\}$. For 2-value BVA all the boundary values for all the equivalence partitions must be covered.</p> <p>The boundary values are -1 (for the “temperature too low” partition), 0, 2 (for the “temperature OK” partition) and 3 (for the “temperature too high” partition).</p> <p>Thus:</p> <p>a) Is not correct</p> <p>b) Is not correct</p> <p>c) Is correct. The correct option is: $-1, 0, 2, 3$</p> <p>d) Is not correct</p> | FL-4.2.2 | K3 | 1 |

| | | | | | |
|----|---|---|----------|----|---|
| 22 | a | <p>Test cases TC1, TC2, TC3 and TC4 cover, respectively, rules R2, R3, R7 and R6 in the decision table.</p> <p>a) Is correct. The conditions “66-year-old”, “unregistered” and “no experience” match rule R4, which is not covered by the existing test cases, so after adding this test case, the decision table coverage will increase</p> <p>b) Is not correct. The conditions “55-year-old”, “unregistered” and “2 years of experience” match rule R2, already covered by TC1. So adding this test case will not increase the coverage</p> <p>c) Is not correct. The conditions “19-year-old”, “registered” and “5 years of experience” match rule R6, already covered by TC4. So adding this test case will not increase the coverage</p> <p>d) Is not correct. The existing test cases cover only 4 out of 7 columns of the decision table. The coverage can be increased by adding test cases that cover yet uncovered columns, that is, R1, R4 and R5</p> | FL-4.2.3 | K3 | 1 |
|----|---|---|----------|----|---|

| | | | | | |
|----|---|--|----------|----|---|
| 23 | b | <p>a) Is not correct. This sequence of five events covers 4 different valid transitions (both “NotAvailable” events correspond to the same transition between S1 and S3). This test case covers 4 out of 7 valid transitions</p> <p>b) Is correct. This sequence of five events covers 5 different transitions (the first “Available” event corresponds to a transition between S1 and S2, and the second “Available” event corresponds to a transition between S3 and S2, so two different transitions are covered). This test case covers 5 out of 7 valid transitions and achieves the highest valid transitions coverage</p> <p>c) Is not correct. This sequence of five events covers 3 different transitions (both “Available” events correspond to the same transition from S1 to S2; both “ChangeRoom” events correspond to the same transition from S2 to S1). This test case covers 3 out of 7 valid transitions</p> <p>d) Is not correct. This sequence of five events does not represent a feasible test case, because after “Cancel” the system ends up in the End state and no further valid transitions can be executed</p> | FL-4.2.4 | K3 | 1 |
|----|---|--|----------|----|---|

| | | | | | |
|----|---|--|----------|----|---|
| 24 | c | <p>a) Is not correct. A line with a defect, when executed, does not have to cause a failure. For example, a line $x := y / z$ will cause a failure <i>only</i> when z equals 0</p> <p>b) Is not correct. 100% statement coverage does not guarantee 100% branch coverage. For example, a test case with $x=0$ for the code</p> <ol style="list-style-type: none"> 1. IF ($x=0$) THEN 2. A; 3. ENDIF <p>achieves 100% statement coverage but does not cover the branch from 1 to 3</p> <p>c) Is correct. 100% statement coverage means that each executable statement was executed at least once</p> <p>d) Is not correct. The removed test case may provide coverage of some statements that are not covered by either of the other two test cases, in which case the remaining two test cases together will not achieve 100% statement coverage</p> | FL-4.3.1 | K2 | 1 |
|----|---|--|----------|----|---|

| | | | | | |
|----|---|--|----------|----|---|
| 25 | a | <p>a) Is correct. A fundamental strength that all white-box test techniques share is that the entire software implementation is taken into account during testing, which facilitates defect detection even when the software specification is vague, outdated or incomplete. This means white-box testing can find defects such as an extra feature added to the code (either accidentally or deliberately) that is not supposed to be there, which black-box testing cannot detect</p> <p>b) Is not correct. The fact that the coverage can be precisely defined is not the right reason. The achieved level of coverage would have much more impact than the possibility to measure the coverage</p> <p>c) Is not correct. If the software does not implement one or more requirements, white-box testing is unlikely to detect the resulting defects of omission</p> <p>d) Is not correct. While this is true, this is not the right answer, because there is no connection between the capability to be used in both static testing and dynamic testing and the claim that white-box testing facilitates defect detection with poor specifications</p> | FL-4.3.3 | K2 | 1 |
| 26 | c | <p>Error guessing is about anticipating the errors, defects and failures based on the tester's knowledge.</p> <p>a) Is not correct. This is an example of anticipating the developer's error</p> <p>b) Is not correct. This is an example of anticipating the defect</p> <p>c) Is correct. This is an example of a potential root cause of a defect, which is neither an error, defect nor failure, and difficult for the tester to anticipate</p> <p>d) Is not correct. This is an example of anticipating a failure, perhaps based on experience of previous systems in this application domain</p> | FL-4.4.1 | K2 | 1 |

| | | | | | |
|----|---|---|----------|----|---|
| 27 | d | <p>a) Is not correct. In exploratory testing, test cases are usually created during the exploratory testing session, alongside test analysis, test implementation and test execution</p> <p>b) Is not correct. In exploratory testing, tests are simultaneously designed, executed, and evaluated while the tester learns about the test object</p> <p>c) Is not correct. Exploratory testing results depend heavily on the tester's experience, so even if the results of exploratory testing can be used as a predictor of risk and used to assess whether there will be fewer or more defects, for example, compared to the previous exploratory testing session, they are not a good example of reliable defect prediction models that can predict the number of remaining defects</p> <p>d) Is correct. During exploratory testing, the testers can use any techniques that they find useful</p> | FL-4.4.2 | K2 | 1 |
| 28 | d | <p>a) Is not correct. Planning poker can estimate effort for a user story that is already written. It does not help in understanding what should be delivered</p> <p>b) Is not correct. Reviews are not a collaborative user story writing practice</p> <p>c) Is not correct. Iteration planning is a project-related practice, used to plan the work, not to understand what needs to be delivered</p> <p>d) Is correct. Conversation explains how the software will be used and often allows the team to define meaningful acceptance criteria, thus obtaining a shared vision of what should be delivered</p> | FL-4.5.1 | K2 | 1 |

| | | | | | |
|----|------|--|----------|----|---|
| 29 | a | <p>a) Is correct. This test case is related to acceptance criteria 2 and 3, because we check if we can set price range (acceptance criterion 2) and if the results update dynamically after adjusting the price range filter (acceptance criterion 3)</p> <p>b) Is not correct. This test case is not related to any of the acceptance criteria. It checks if the filter dynamically sets the default minimum and maximum price range, and not that a customer can do it</p> <p>c) Is not correct. This test case is not related to any of the acceptance criteria. It checks the currency exchange feature, which is not discussed in this user story</p> <p>d) Is not correct. This test case is not related to any of the acceptance criteria. It checks the application's compatibility with different browsers, which is not discussed in this user story</p> | FL-4.5.3 | K3 | 1 |
| 30 | b, d | <p>a) Is not correct. The approval of the budget is an example of an entry criterion. It would make no sense to approve the budget for some activity that has already been done</p> <p>b) Is correct. Running out of budget can be viewed as a valid exit criterion</p> <p>c) Is not correct. Availability of resources is an example of an entry criterion for testing</p> <p>d) Correct. Coverage is a measure of thoroughness, so it is a typical exit criterion</p> <p>e) Is not correct. This is an example of an entry criterion, checked before the project starts</p> | FL-5.1.3 | K2 | 1 |

| | | | | | |
|----|---|---|----------|----|---|
| 31 | a | <p>Using the three-point estimation technique, the final estimate (E) is calculated as: $E = (a + 4*m + b) / 6,$ where a is the most optimistic estimate, m is the most likely estimate, and b is the most pessimistic estimate.</p> <p>Thus: a) Is correct. In this case, the estimate for executing a single test case is: $E = (1h + 4*3h + 8h) / 6 = 3.5 \text{ hours}$ So, the total time needed for the tester to execute 4 test cases is: $3.5h * 4 = 14 \text{ hours}$ b) Is not correct c) Is not correct d) Is not correct</p> | FL-5.1.4 | K3 | 1 |
| 32 | b | <p>TC1 achieves the highest coverage (4/7 – Req1, Req3, Req4 and Req7), so should be executed first. Req2, Req5 and Req6 are still not covered. The next test case that achieves the highest additional coverage of the remaining requirements is TC3, covering 2 out of these 3 requirements (Req5 and Req6). So, TC3 should be executed as the second one. Now the only requirement still not covered is Req2, which is covered by TC4. Therefore, TC4 should be executed as the third test case. So, the last test case executed will be TC2.</p> <p>Thus: a) Is not correct b) Is correct c) Is not correct d) Is not correct</p> | FL-5.1.5 | K3 | 1 |

| | | | | | |
|----|------|--|----------|----|---|
| 33 | c | <p>a) Is not correct. Testing quadrants have nothing to do with describing the relationships between test levels</p> <p>b) Is not correct. Testing quadrants cannot help in assessing any type of coverage</p> <p>c) Is correct. Testing quadrants allow managers and other stakeholders to understand the relationships between test types, the activities they support (team support or product critique), and the viewpoint they are focused on (business- or technology-facing)</p> <p>d) Is not correct. Testing quadrants is not a psychological model</p> | FL-5.1.7 | K2 | 1 |
| 34 | b | <p>Risk assessment can use a quantitative or qualitative approach, or a mix of them. In the quantitative approach the risk level is calculated as the multiplication of risk likelihood and risk impact. So, Risk level = Risk likelihood * Risk impact</p> <p>Then, Risk impact = Risk level / Risk likelihood.</p> <p>In our case, Risk impact = \$1,000 / 50% = \$1,000 / 0.5 = \$2,000.</p> <p>Thus:</p> <p>a) Is not correct</p> <p>b) Is correct</p> <p>c) Is not correct</p> <p>d) Is not correct</p> | FL-5.2.1 | K1 | 1 |
| 35 | b, e | <p>a) Is not correct. Scope creep is an example of a project risk related to technical issues</p> <p>b) Is correct. Poor architecture is an example of a product risk since it refers to a product characteristic</p> <p>c) Is not correct. Cost-cutting is an example of a project risk, related to organizational issues</p> <p>d) Is not correct. Poor tool support is an example of a project risk related to technical issues</p> <p>e) Is correct. Response time too long is an example of a product risk since it refers to a product characteristic</p> | FL-5.2.2 | K2 | 1 |

| | | | | | |
|-----------|---|---|----------|----|---|
| 36 | c | a) Is not correct. Tracking test progress and identifying areas that require further attention is an example of supporting the ongoing control of testing. This is one of the purposes of test reports b) Is not correct. Providing information on the tests executed, their results, and any issues or defects found is an example of summarizing the test activities performed at a given test level. This is one of the purposes of test reports c) Is correct. Providing information about defects is the purpose of a defect report, not a test report d) Is not correct. Providing information on testing planned for the next period is one of the purposes of test reports | FL-5.3.2 | K2 | 1 |
|-----------|---|---|----------|----|---|

| | | | | | |
|----|---|--|----------|----|---|
| 37 | d | <p>a) Is not correct. Risk management consists of risk analysis and risk control. Neither of these activities supports the reassembly of the files that made up the release, because these activities deal with risks, not with configuration items</p> <p>b) Is not correct. Test monitoring is concerned with gathering information about testing. This information is used to assess test progress and to measure whether the test exit criteria or the test tasks associated with the exit criteria are satisfied, such as meeting the targets for coverage of product risks, requirements, or other acceptance criteria. Test control uses the information from test monitoring to provide, in the form of control directives, guidance and the necessary corrective actions to achieve the most effective and efficient testing. None of these activities deal with the management of configuration items</p> <p>c) Is not correct. The whole-team approach builds on the tester's skill to work effectively in a team context and to contribute positively to the team goals. So, it focuses on team-related issues, not on configuration items</p> <p>d) Is correct. Configuration management provides a discipline for identifying, controlling, and tracking work products. Configuration management keeps a record of changed configuration items when a new baseline is created. Using configuration management, it is possible to revert to a previous baseline in order to reproduce previous test results</p> | FL-5.4.1 | K2 | 1 |
|----|---|--|----------|----|---|

| | | | | | |
|-----------|---|--|----------|----|---|
| 38 | a | a) Is correct. Adding this information allows the developer to use the same input data, so it is more likely they will be able to reproduce the failure quickly and so identify the defect faster b) Is not correct. Adding the value of Priority will not help in reproducing the defect itself c) Is not correct. Although some of this information may be of value, adding the memory dumps and database snapshots after each step will be too much, because most of these artefacts will contain useless information for the developer, and make the report less readable. It will also require the developer to spend a lot of time analyzing this information, which will lengthen the repair process d) Is not correct. The question was about helping the developer to reproduce the defect observed for a specific environment configuration | FL-5.5.1 | K3 | 1 |
|-----------|---|--|----------|----|---|

| | | | | | |
|----|---|--|----------|----|---|
| 39 | b | <p>Considering each of the listed tool categories:</p> <ul style="list-style-type: none"> i. Collaboration tools – facilitate communication. Communication does not include the facilitation of test execution. ii. DevOps tools - support the DevOps delivery pipeline, workflow tracking, automated build process(es) and CI/CD. The delivery pipeline and CI/CD both include the facilitation of test execution, such as component testing for CI. iii. Management tools – increase the test process efficiency by facilitating management of the SDLC, requirements, tests, defects and configuration. The management of these items does not include the facilitation of test execution. iv. Non-functional testing tools – allow the tester to perform non-functional testing that is difficult or impossible to perform manually. Non-functional testing can include both static testing and dynamic testing, including test execution. v. Test design and implementation tools – facilitate generation of test cases, test data and test procedures. The generation of this testware does not include the facilitation of test execution. <p>Thus:</p> <ul style="list-style-type: none"> a) Is not correct b) Is correct. Both DevOps tools (ii) and Non-functional testing tools (iv) facilitate test execution c) Is not correct d) Is not correct | FL-6.1.1 | K2 | 1 |
|----|---|--|----------|----|---|

| | | | | | |
|-----------|---|--|----------|----|---|
| 40 | c | a) Is not correct. The detection of additional high-severity defects would be a benefit of test automation, rather than a risk b) Is not correct. The provision of measures that are too complicated for humans to derive themselves is normally considered to be a benefit of test automation c) Is correct. If the test automation is incompatible with the development platform, then it will not be able to integrate them, and, for instance, pass test inputs to the test object and receive test results from the test object d) Is not correct. Substantially reduced test execution times would normally be considered a benefit that is provided by test automation | FL-6.2.1 | K1 | 1 |
|-----------|---|--|----------|----|---|