

CTFL4 Dumps

ISTQB Certified Tester Foundation Level CTFL 4.0 Exam

<https://www.certleader.com/CTFL4-dumps.html>



NEW QUESTION 1

Which of the statements correctly describes when a whole team approach may NOT be suitable?

- A. When a high level of test independence may be required.
- B. When acceptance tests need to be created.
- C. When a test automation approach needs to be determined.
- D. When the team dynamics need to be improved.

Answer: A

Explanation:

The whole team approach involves collaboration among all team members, including testers, developers, and business representatives, to achieve quality goals. However, this approach may not be suitable in situations where a high level of test independence is required. Test independence is essential in cases where unbiased testing is critical, such as in regulated environments or where high-risk systems are involved. This is because team members might unintentionally influence each other's work, leading to potential bias in testing outcomes.

NEW QUESTION 2

Which statement is true regarding confirmation testing and regression testing?

- A. Confirmation testing confirms the quality of the test being run while regression testing ensures that the software still works after a change has been made.
- B. Confirmation testing is an optional activity whilst regression testing is not negotiable.
- C. Confirmation testing aims to verify that a defect has been resolved and regression testing ensuring that existing functionality still works after a change.
- D. Testers' involvement is essential whilst running retesting and regression testing.
- E. TESTER Involvement is essential whilst running retesting and regression testing.

Answer: C

Explanation:

Confirmation testing, also known as retesting, is conducted to verify that specific defects have been fixed. Regression testing, on the other hand, is performed to ensure that recent changes have not adversely affected existing features of the software. Both types of testing are crucial for maintaining the integrity and quality of the software after modifications.

NEW QUESTION 3

Consider the following user story about the authentication functionality of an e-commerce website:

"As a logged-in user, I want to change my current password with a new one, so that I can make my account safer".

The following are some of the acceptance criteria defined for the user story:

- [a] After the logged-in user has successfully changed his password, an email confirming the change must be sent to him
 - [b] To successfully change the password, the logged-in user must enter the current password, enter a new valid password, and finally confirm by pressing the 'Change Password' button
 - [c] To be valid, the new password entered by the logged-in user is not only required to meet the criteria related to the length and type of characters, but must also be different from the last 5 passwords of that user
 - [d] A dedicated error message must be presented to the logged-in user when he enters a wrong current password
 - [e] A dedicated error message must be presented to the logged-in user when he enters the correct current password, but enters an invalid password
- Based only on the given information, which of the following ATDD tests is most likely to be written first?

- A. The logged-in user enters a wrong current password and views the dedicated error message
- B. The logged-in user enters the correct current password, enters a valid new password(different from the last 5 passwords), presses the Change Password' button, and finally receives the e-mail confirming that the password has been successfully changed
- C. The logged-in user enters the correct current password, enters an invalid password, and finally views the dedicated error
- D. The logged-in user submits a purchase order containing ten items, selects to pay with a Visa credit card, enters credit card information of a valid card, presses the 'Confirm' button, and finally views the dedicated message confirming that the purchase has been successful

Answer: B

Explanation:

ATDD stands for Acceptance Test-Driven Development, which is a collaborative approach to software development and testing, in which the acceptance criteria of a user story are defined and automated as executable tests before the implementation of the software system. ATDD tests are usually written in a Given-When-Then format, which describes the preconditions, the actions, and the expected outcomes of a test scenario. ATDD tests are intended to verify that the software system meets the expectations and the needs of the users and the stakeholders, as well as to provide feedback and guidance for the developers and the testers. Based on the given information, the ATDD test that is most likely to be written first is the one that corresponds to option B, which is:

Given the logged-in user is on the Change Password page When the user enters the correct current password, enters a valid new password (different from the last 5 passwords), and presses the Change Password button Then the user receives an email confirming that the password has been successfully changed

This ATDD test is most likely to be written first, because it covers the main functionality and the happy path of the user story, as well as the most important acceptance criterion [a]. It also verifies that the user can change the password with a valid new password that meets the criteria related to the length, the type of characters, and the history of the passwords, as specified in the acceptance criterion [c]. The other options are not likely to be written first, because they either cover less critical or less frequent scenarios, such as entering a wrong current password [d] or an invalid new password [e], or they are not related to the user story or the acceptance criteria at all, such as submitting a purchase order [d]. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.3.1, Testing in Software Development Lifecycles¹

? ISTQB® Glossary of Testing Terms v4.0, Acceptance Test-Driven Development, User Story, Acceptance Criterion, Given-When-Then²

NEW QUESTION 4

A calculator software is used to calculate the result for 5+6. The user noticed that the result given is 6.

This is an example of;

- A. Mistake
- B. Fault

- C. Error
- D. Failure

Answer: D

Explanation:

According to the ISTQB Glossary of Testing Terms, Version 4.0, 2018, page 18, a failure is ??an event in which a component or system does not perform a required function within specified limits??. In this case, the calculator software does not perform the required function of calculating the correct result for 5+6 within the specified limits of accuracy and precision. Therefore, this is an example of a failure.

The other options are incorrect because:

? A mistake is ??a human action that produces an incorrect result?? (page 25). A mistake is not an event, but an action, and it may or may not lead to a failure. For example, a mistake could be a typo in the code, a wrong assumption in the design, or a misunderstanding of the requirement.

? A fault is ??a defect in a component or system that can cause the component or system to fail to perform its required function?? (page 16). A fault is not an event, but a defect, and it may or may not cause a failure. For example, a fault could be a logical error in the code, a missing specification in the design, or a contradiction in the requirement.

? An error is ??the difference between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition?? (page 15). An error is not an event, but a difference, and it may or may not result in a failure. For example, an error could be a rounding error in the calculation, a measurement error in the observation, or a deviation error in the condition.

References = ISTQB Glossary of Testing Terms, Version 4.0, 2018, pages 15-18, 25;

ISTQB CTFL 4.0 - Sample Exam - Answers, Version 1.1, 2023, Question 96, page 34.

NEW QUESTION 5

Which of the following statements about traceability is FALSE?

- A. Traceability between test basis items and the test cases designed to cover them, makes it possible to determine which test basis items have been covered by the executed test cases.
- B. Traceability between test basis items and the test cases designed to cover them, enables experience-based test techniques to be applied
- C. Traceability between test basis items and the test cases designed to cover them, enables identification of which test cases will be affected by changes to the test basis items.
- D. Traceability can be established and maintained through all the test documentation for a given test level, such as from test conditions through test cases to test scripts.

Answer: B

Explanation:

Traceability primarily refers to the ability to link test cases back to their sources in the test basis, such as requirements or design documents. This linkage allows for the determination of coverage, impact analysis, and maintaining consistency across test documentation. However, traceability does not directly enable the application of experience-based test techniques, which are more about using the tester's intuition and experience. The ISTQB CTFL Syllabus v4.0 does not state that traceability enables experience-based techniques, making option B the false statement.

NEW QUESTION 6

Which of the following statements is an example of testing contributing to higher quality?

- A. A test leader writes a test summary report
- B. A project manager asks to a test leader to estimate the test effort
- C. A tester installs a test ten in the lest environment
- D. A tester finds a bug which is resolved prior to release

Answer: D

Explanation:

? The question is about identifying an example of testing contributing to higher quality. Quality is the degree to which a component, system or process meets specified requirements and/or user/customer needs and expectations¹. Testing is the process consisting of all lifecycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects².

? Therefore, testing contributes to higher quality by verifying and validating that the software products and related work products meet the specified requirements, are fit for purpose and have no defects, or at least have a reduced number of defects. Testing also provides information about the quality of the software products and related work products to the stakeholders, who can make informed decisions based on the test results³.

? Out of the four given statements, only option D is an example of testing contributing to higher quality, as it shows that testing has detected a defect (a flaw in a component or system that can cause the component or system to fail to perform its required function⁴) and that the defect has been resolved (fixed and confirmed) prior to release (delivery of the software product to the customer or end user). This means that testing has prevented a potential failure (an event in which a component or system does not perform a required function within specified limits) from occurring in the operational environment, and thus has improved the quality of the software product.

? Option A is not an example of testing contributing to higher quality, as it is a reporting activity that summarizes the test results and evaluates the test objectives, but does not directly affect the quality of the software product or related work products. A test summary report is a document that records and communicates the outcomes of testing activities, including test completion criteria, test results, incident reports, test summary and evaluation, and lessons learned.

? Option B is not an example of testing contributing to higher quality, as it is a planning activity that estimates the resources and time needed for testing activities, but does not directly affect the quality of the software product or related work products. A test effort estimate is an approximation of the amount of work and/or the duration of time required to perform testing activities.

? Option C is not an example of testing contributing to higher quality, as it is a preparation activity that sets up the test environment (an environment containing hardware, instrumentation, simulators, software tools, and other support elements needed to conduct a test), but does not directly affect the quality of the software product or related work products. A test environment installation is a process of installing and configuring the test environment according to the test environment specification.

References:

? 1: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 10

? 2: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 11

? 3: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 12

? 4: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 13

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 13

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 77

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 78

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 79
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 80
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 81
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 82
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 83
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 84
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 85
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 86
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 87
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 88
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 89
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 90
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 91
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 92
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 93
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 94
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 95
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 96
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 97
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 98
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 99
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 100
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 101
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 102
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 103
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 104
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 105
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 106
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 107

NEW QUESTION 7

Which of the following statements is TRUE?

- A. User acceptance tests are usually automated and aim to verify the acceptance criteria for user stories
- B. Acceptance criteria for user stories can include details on data definitions, for example by describing the format, allowed values, and default values for a data item
- C. Acceptance criteria for user stories should focus on positive scenarios, while negative scenarios should be excluded
- D. Tests derived from acceptance criteria for user stories are not included in any of the four testing quadrants

Answer: B

Explanation:

Acceptance criteria for user stories often include detailed specifications about data definitions, such as the format, allowed values, and default values for a data item. This helps ensure that the developed feature meets the expected requirements and provides a clear understanding for both developers and testers on what needs to be validated. Therefore, statement B is true as per the ISTQB CTFL syllabus.

NEW QUESTION 8

Testing Quadrants, as a model, is effective in aligning stakeholders within Agile teams. Which of the following examples demonstrates this?

- A. Using Testing Quadrants, the test manager is able to measure and communicate test coverage to all stakeholders.
- B. Using Testing Quadrants, the test manager is able to communicate potential product risk to all stakeholders.
- C. Using Testing Quadrant, the test manager is able to prioritize defects by linking these to a specific type of test.
- D. Using Testing Quadrants, the test manager is able to differentiate and describe the types of tests to all stakeholders.

Answer: D

Explanation:

The Testing Quadrants model helps Agile teams by categorizing different types of tests and their purposes. This differentiation helps test managers explain the testing strategy to all stakeholders, ensuring everyone understands the scope and objectives of each test type. This model aids in planning, executing, and tracking testing activities across different quadrants, making it easier to align with stakeholders' expectations and project goals. Reference: ISTQB CTFL Syllabus V4.0, Section 5.1.7

NEW QUESTION 9

Which of the following best describes the way in which statement coverage is measured?

- A. Measured as the number of decision outcomes executed by the tests, divided by the total number of decision outcomes in the test object.
- B. It is not possible to accurately measure statement coverage.
- C. Measured as the number of statements executed by the tests, divided by the total number of executable statements in the code.
- D. Measured as the number of lines of code executed by the test, divided by the total number of lines of code in the test object.

Answer: C

Explanation:

Statement coverage is a metric used in white-box testing that measures the percentage of executable statements in the code that have been executed by the test cases. It is calculated as the number of statements executed by the tests divided by the total number of executable statements in the code, providing an indication of how much of the code has been tested.

NEW QUESTION 10

Which of the following is a test-first approach, where tests that express a shared understanding from stakeholders of how the application is expected to work, are

first written in business-readable language (following the Given/When/Then format), and then made executable to drive development?

- A. Test-Driven Development (TDD)
- B. Acceptance Test-Driven Development (ATDD)
- C. Behavior-Driven Development (BDD)
- D. Domain-Driven Design (DDD)

Answer: C

Explanation:

This answer is correct because Behavior-Driven Development (BDD) is a test-first approach, where tests that express a shared understanding from stakeholders of how the application is expected to work, are first written in business-readable language (following the Given/When/Then format), and then made executable to drive development. BDD is a collaborative approach that involves testers, developers, business analysts, product owners, and other stakeholders in defining the expected behavior of the application using scenarios that describe the preconditions, actions, and outcomes of the application. BDD scenarios are written using a domain-specific language (DSL) that can be translated into executable test cases using tools such as Cucumber or SpecFlow. BDD aims to improve communication, collaboration, and feedback among the team members, and to deliver software that meets the customer's needs and expectations. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 3.1.1.4

NEW QUESTION 10

Which of the following statements is true?

- A. Experience-based test techniques rely on the experience of testers to identify the root causes of defects found by black-box test techniques
- B. Some of the most common test basis used by white-box test techniques include user stories, use cases and business processes
- C. Experience-based test techniques are often useful to detect hidden defects that have not been targeted by black-box test techniques
- D. The primary goal of experience-based test techniques is to design test cases that can be easily automated using a GUI-based test automation tool

Answer: C

Explanation:

Experience-based test techniques are test design techniques that rely on the experience, knowledge, intuition, and creativity of the testers to identify and execute test cases that are likely to find defects in the software system. Experience-based test techniques are often useful to detect hidden defects that have not been targeted by black-box test techniques, which are test design techniques that use the external behavior and specifications of the software system as the test basis, without considering its internal structure or implementation. Experience-based test techniques can complement black-box test techniques by covering aspects that are not explicitly specified, such as usability, security, reliability, performance, etc. The other statements are false, because:
? Experience-based test techniques do not rely on the experience of testers to identify the root causes of defects found by black-box test techniques, but rather to identify the potential sources of defects based on their own insights, heuristics, or exploratory testing. The root causes of defects are usually identified by debugging or root cause analysis, which are activities that involve examining the code or the development process to find and fix the errors that led to the defects.
? Some of the most common test basis used by white-box test techniques include the source code, the design documents, the architecture diagrams, and the control flow graphs of the software system. White-box test techniques are test design techniques that use the internal structure and implementation of the software system as the test basis, and aim to achieve a certain level of test coverage based on the code elements, such as statements, branches, paths, etc. User stories, use cases, and business processes are examples of test basis used by black-box test techniques, as they describe the functional and non-functional requirements of the software system from the perspective of the users or the stakeholders.
? The primary goal of experience-based test techniques is not to design test cases that can be easily automated using a GUI-based test automation tool, but rather to design test cases that can reveal defects that are not easily detected by other test techniques, such as boundary value analysis, equivalence partitioning, state transition testing, etc. Test automation is the use of software tools to execute test cases and compare actual results with expected results, without human intervention. Test automation can be applied to different types of test techniques, depending on the test objectives, the test levels, the test tools, and the test resources. However, test automation is not always feasible or beneficial, especially for test cases that require human judgment, creativity, or exploration, such as those designed by experience-based test techniques. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:
? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.1, Black-box Test Design Techniques
? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.2, White-box Test Design Techniques
? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.3, Experience-based Test Design Techniques
? ISTQB® Glossary of Testing Terms v4.0, Experience-based Test Technique, Black-box Test Technique, White-box Test Technique, Test Basis, Test Coverage, Test Automation

NEW QUESTION 13

Which of the following statements is not correct?

- A. Looking for defects in a system may require Ignoring system details
- B. Identifying defects may be perceived as criticism against product
- C. Looking for defects in system requires professional pessimism and curiosity
- D. Testing is often seen as a destructive activity instead of constructive activity

Answer: A

Explanation:

? Looking for defects in a system does not require ignoring system details, but rather paying attention to them and understanding how they affect the system's quality, functionality, and usability. Ignoring system details could lead to missing important defects or testing irrelevant aspects of the system.
? Identifying defects may be perceived as criticism against product, especially by the developers or stakeholders who are invested in the product's success. However, identifying defects is not meant to be a personal attack, but rather a constructive feedback that helps to improve the product and ensure its alignment with the requirements and expectations of the users and clients.
? Looking for defects in system requires professional pessimism and curiosity, as testers need to anticipate and explore the possible ways that the system could fail, malfunction, or behave unexpectedly. Professional pessimism means being skeptical and critical of the system's quality and reliability, while curiosity means being eager and interested in finding out the root causes and consequences of the defects.
? Testing is often seen as a destructive activity instead of constructive activity, as it involves finding and reporting the flaws and weaknesses of the system, rather than creating or enhancing it. However, testing is actually a constructive activity, as it contributes to the system's improvement, verification, validation, and optimization, and ultimately to the delivery of a high-quality product that meets the needs and expectations of the users and clients.

NEW QUESTION 15

Which of the following is a typical potential risk of using test automation tools?

- A. Reduced feedback times regarding software quality compared to manual testing.
- B. Reduced test execution times compared to manual testing.
- C. Reduced repeatability and consistency of tests compared to manual testing
- D. Underestimation of effort required to maintain test scripts.

Answer: D

Explanation:

One of the common risks associated with test automation tools is the underestimation of the effort required to maintain test scripts. Test scripts can become outdated or broken due to changes in the application, requiring significant effort to update and maintain them. This risk is highlighted in the ISTQB CTFL syllabus under the discussion of the benefits and risks of test automation.

References:ISTQB CTFL Syllabus, Section on test tools and automation.

NEW QUESTION 19

Which review type, also known as a ??buddy check??. is commonly used in Agile development?

- A. Inspection.
- B. Walkthrough.
- C. Technical review.
- D. Informal review.

Answer: D

Explanation:

In Agile development, an informal review, often referred to as a "buddy check," is a common review type. Informal reviews are unstructured and involve a pair of colleagues reviewing each other's work to identify defects early and provide immediate feedback. This type of review is less formal than inspections or walkthroughs and is particularly suitable for Agile environments where rapid feedback and flexibility are essential.

References:ISTQB CTFL Syllabus, Section 3.2.4, "Types of Reviews" and Section 2.1.4, "Agile Testing Practices."

NEW QUESTION 24

A document describes the test procedures that have been derived for the identified test sets Among other things, the order in which the test cases in the corresponding test set are to be executed according to the dependencies described by preconditions and postconditions is specified This document is a typical work product produced as part of:

- A. Test design.
- B. Test analysis
- C. Test Implementation.
- D. Test monitoring and control

Answer: C

Explanation:

Test implementation involves finalizing the test procedures, including the order of execution of test cases based on their dependencies, preconditions, and postconditions. This phase ensures that all necessary test scripts, test data, and test environments are ready for execution. According to the ISTQB CTFL Syllabus v4.0, test implementation is the phase where detailed test procedures are derived and documented, making it a critical step before actual test execution.

NEW QUESTION 27

Which of the following statements about white-box test techniques is true?

- A. Achieving full statement coverage and full branch coverage for a software product means that such software product has been fully tested and there are no remaining bugs within the code
- B. Code-related white-box test techniques are not required to measure the actual code coverage achieved by black-box testing, as code coverage can be measured using the coverage criteria associated with black-box test techniques
- C. Branch coverage is the most thorough code-related white-box test technique, and therefore applicable standards prescribe achieving full branch coverage at the highest safety levels for safety-critical systems
- D. Code-related white-box test techniques provide an objective measure of coverage and can be used to complement black-box test techniques to increase confidence in the code

Answer: D

Explanation:

This answer is correct because code-related white-box test techniques are test design techniques that use the structure of the code to derive test cases. They provide an objective measure of coverage, such as statement coverage, branch coverage, or path coverage, which indicate how much of the code has been exercised by the test cases. Code-related white-box test techniques can be used to complement black-box test techniques, which are test design techniques that use the functional or non-functional requirements of the system or component to derive test cases. By combining both types of techniques, testers can increase their confidence in the code and find more

defects. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.3.2.2

NEW QUESTION 31

Given the following User Story: "As an online customer, I would like to be able to cancel the purchase of an individual item from a shopping list so that it only displays the relevant items, in less than 1 second", which of the following can be considered as applicable acceptance test cases?

- A. Click on my online shopping list, select the unwanted Item, delete the unwanted item, the unwanted Item is deleted from the shopping list in less than 1 second.i
- B. Click on my online shopping list, select all the items, delete all the items, the unwanted items are deleted from the shopping list in less than 1 second.ii
- C. Tab to the online shopping list and press enter, select the unwanted item, delete the unwanted item, the unwanted item is deleted from the shopping list In less

than 1 second.I

D. Click on the checkout button, select the payment method, make payment, confirmation received of payment and shipping date.

E. Click on my shopping list, select the unwanted item, delete the unwanted item, the unwanted item is deleted from the shopping list. Select the correct Answer

F. I, ii and v

G. iv

H. i and iii

I. v

Answer: C

Explanation:

Reference: ISTQB CTFL Syllabus V4.0, Section 5.2.2

NEW QUESTION 33

Which of the following coverage criteria results in the highest coverage for state transition based test cases?

A. Can't be determined

B. Covering all transitions at least once

C. Covering only start and end states

D. Covering all states at least once

Answer: B

Explanation:

Covering all transitions at least once is the highest coverage criterion for state transition based test cases, because it ensures that every possible change of state is tested at least once. This means that all the events that trigger the transitions, as well as the actions and outputs that result from the transitions, are verified.

Covering all transitions at least once also implies covering all states at least once, but not vice versa. Therefore, option D is not the highest coverage criterion.

Option C is the lowest coverage criterion, because it only tests the initial and final states of the system or component, without checking the intermediate states or transitions. Option A is incorrect, because the coverage criteria for state transition based test cases can be determined and compared based on the number of transitions and states covered. References = CTFL 4.0 Syllabus, Section 4.2.3, page 49-50.

NEW QUESTION 37

Which of the following is a test task that usually occurs during test implementation?

A. Make sure the planned test environment is ready to be delivered

B. Find, analyze, and remove the causes of the failures highlighted by the tests

C. Archive the testware for use in future test projects

D. Gather the metrics that are used to guide the test project

Answer: A

Explanation:

A test task that usually occurs during test implementation is to make sure the planned test environment is ready to be delivered. The test environment is the hardware and software configuration on which the tests are executed, and it should be as close as possible to the production environment where the software system will operate. The test environment should be planned, prepared, and verified before the test execution, to ensure that the test conditions, the test data, the test tools, and the test interfaces are available and functional. The other options are not test tasks that usually occur during test implementation, but rather test tasks that occur during other test activities, such as:

? Find, analyze, and remove the causes of the failures highlighted by the tests: This is a test task that usually occurs during test analysis and design, which is the activity of analyzing the test basis, designing the test cases, and identifying the test data. During this activity, the testers can use techniques such as root cause analysis, defect prevention, or defect analysis, to find, analyze, and remove the causes of the failures highlighted by the previous tests, and to prevent or reduce the occurrence of similar failures in the future tests.

? Archive the testware for use in future test projects: This is a test task that usually occurs during test closure, which is the activity of finalizing and reporting the test results, evaluating the test process, and identifying the test improvement actions. During this activity, the testers can archive the testware, which are the test artifacts produced during the testing process, such as the test plan, the test cases, the test data, the test results, the defect reports, etc., for use in future test projects, such as regression testing, maintenance testing, or reuse testing.

? Gather the metrics that are used to guide the test project: This is a test task that usually occurs during test monitoring and control, which is the activity of tracking and reviewing the test progress, status, and quality, and taking corrective actions when necessary. During this activity, the testers can gather the metrics, which are the measurements of the testing process, such as the test coverage, the defect density, the test effort, the test duration, etc., that are used to guide the test project, such as planning, estimating, scheduling, reporting, or improving the testing process. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.1, Test Planning¹

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.2, Test Monitoring and Control¹

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.3, Test Analysis and Design¹

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.4, Test Implementation¹

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.5, Test Execution¹

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.6, Test Closure¹

? ISTQB® Glossary of Testing Terms v4.0, Test Environment, Test Condition, Test Data, Test Tool, Test Interface, Failure, Root Cause Analysis, Defect Prevention, Defect Analysis, Testware, Regression Testing, Maintenance Testing, Reuse Testing, Test Coverage, Defect Density, Test Effort, Test Duration²

NEW QUESTION 42

In addition to thorough testing of the requirements specification, a development team aims to involve users as early as possible in the development process, using practices such as prototyping, to ensure that the software systems being developed will meet the users' expectations. This approach is especially useful at mitigating the risks associated with one of the seven testing principles, which one?

A. Tests wear out

B. Absence-of-errors fallacy

C. Working software over comprehensive documentation.

D. Defects cluster together

Answer: B

Explanation:

The absence-of-errors fallacy is the mistaken belief that just because a software system is free of defects, it will meet the user's needs and expectations. Involving users early through practices like prototyping helps ensure that the development team is building the right system that meets user expectations, not just a system that is defect-free. This approach aligns with the testing principle that emphasizes understanding the users' needs and ensuring the system fulfills them. This principle is explained in the ISTQB CTFL Syllabus v4.0.

NEW QUESTION 44

What is test oracle?

- A. The source of test objectives
- B. The source for the actual results
- C. The source of expected results
- D. The source of input conditions

Answer: C

Explanation:

A test oracle is a mechanism or principle that can be used to determine whether the observed behavior or output of a system under test is correct or not¹. A test oracle can be based on various sources of expected results, such as specifications, user expectations, previous versions, comparable systems, etc². References: ISTQB Certified Tester Foundation Level(CTFL) v4.0 Syllabus, Section 1.2.1, Page 91; ISTQB Glossary of Testing Terms, Version 4.0, Page 332.

NEW QUESTION 47

Consider a review for a high-level architectural document written by a software architect. The architect does most of the review preparation work, including distributing the document to reviewers before the review meeting. However, reviewers are not required to analyze the document in advance, and during the review meeting the software architect explains the document step by step. The only goal of this review is to establish a common understanding of the software architecture that will be used in a software development project.

Which of the following review types does this review refer to?

- A. Inspection
- B. Audit
- C. Walkthrough
- D. Informal review

Answer: C

Explanation:

This answer is correct because a walkthrough is a type of review where the author of the work product leads the review process and explains the work product to the reviewers. The reviewers are not required to prepare for the review in advance, and the main objective of the walkthrough is to establish a common understanding of the work product and to identify any major defects or issues. A walkthrough is usually informal and does not follow a defined process or roles. In this case, the review for a high-level architectural document written by a software architect matches the characteristics of a walkthrough. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.4.2.2

NEW QUESTION 50

Which of the following are the phases of the ISTQB fundamental test process?

- A. Test planning and control, Test analysis and design, Test implementation and execution, Evaluating exit criteria and reporting
- B. Test closure activities
- C. Test planning, Test analysis and design
- D. Test implementation and control
- E. Checking test coverage and reporting, Test closure activities
- F. Test planning and control, Test specification and design
- G. Test implementation and execution, Evaluating test coverage and reporting, Retesting and regression testing, Test closure activities
- H. Test planning
- I. Test specification and design
- J. Test implementation and execution
- K. Evaluating exit criteria and reporting
- L. Retesting and test closure activities

Answer: A

Explanation:

The ISTQB fundamental test process consists of five main phases, as described in the ISTQB Foundation Level Syllabus, Version 4.0, 2018, Section 2.2, page 15:

? Test planning and control: This phase involves defining the test objectives, scope, strategy, resources, schedule, risks, and metrics, as well as monitoring and controlling the test activities and results throughout the test process.

? Test analysis and design: This phase involves analyzing the test basis (such as requirements, specifications, or user stories) to identify test conditions (such as features, functions, or scenarios) that need to be tested, and designing test cases and test procedures (such as inputs, expected outcomes, and execution steps) to cover the test conditions. This phase also involves evaluating the testability of the test basis and the test items (such as software or system components), and selecting and implementing test techniques (such as equivalence partitioning, boundary value analysis, or state transition testing) to achieve the test objectives and optimize the test coverage and efficiency.

? Test implementation and execution: This phase involves preparing the test environment (such as hardware, software, data, or tools) and testware (such as test cases, test procedures, test data, or test scripts) for test execution, and executing the test procedures or scripts according to the test plan and schedule. This phase also involves logging the outcome of test execution, comparing the actual results with the expected results, and reporting any discrepancies as incidents (such as defects, errors, or failures).

? Evaluating exit criteria and reporting: This phase involves checking if the planned test activities have been completed and the exit criteria (such as quality, coverage, or risk levels) have been met, and reporting the test results and outcomes to the stakeholders. This phase also involves making recommendations for the release or acceptance decision based on the test results and outcomes, and identifying any residual risks (such as known defects or untested areas) that need to be addressed or mitigated.

? Test closure activities: This phase involves finalizing and archiving the testware and test environment for future reuse, and evaluating the test process and the test project against the test objectives and the test plan. This phase also involves identifying any lessons learned and best practices, and communicating the findings and suggestions for improvement to the relevant parties.
References = ISTQB Certified Tester Foundation Level Syllabus, Version 4.0, 2018, Section 2.2, page 15; ISTQB Glossary of Testing Terms, Version 4.0, 2018, pages 37-38;
ISTQB CTFL 4.0 - Sample Exam - Answers, Version 1.1, 2023, Question 88, page 32.

NEW QUESTION 51

Which of the following is a task the Author is responsible for, as part of a typical formal review?

- A. Determining the people who will be involved in the review
- B. Recording the anomalies found during the review meeting
- C. Identifying potential anomalies in the work product under review
- D. Fixing the anomalies found in the work product under review

Answer: C

Explanation:

This answer is correct because identifying potential anomalies in the work product under review is one of the tasks the Author is responsible for, as part of a typical formal review. The Author is the person who creates the work product to be reviewed, such as a requirement specification, a design document, or a test case. The Author's tasks include preparing the work product for the review, identifying potential anomalies in the work product, and fixing the anomalies found in the work product after the review. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.4.2.1

NEW QUESTION 52

Consider a given test plan which, among others, contains the following three sections: "Test Scope", "Testing Communication", and "Stakeholders". The features of the test object to be tested and those excluded from the testing represent information that is:

- A. not usually included in a test plan, and therefore in the given test plan it should not be specified neither within the three sections mentioned, nor within the others
- B. usually included in a test plan and, in the given test plan, it is more likely to be specified within "Test Scope" rather than in the other two sections mentioned
- C. usually included in a test plan and, in the given test plan, it is more likely to be specified within "Testing Communication" rather than in the other two sections mentioned
- D. usually included in a test plan and, in the given test plan, it is more likely to be specified within "Stakeholders" rather than in the other two sections mentioned

Answer: B

Explanation:

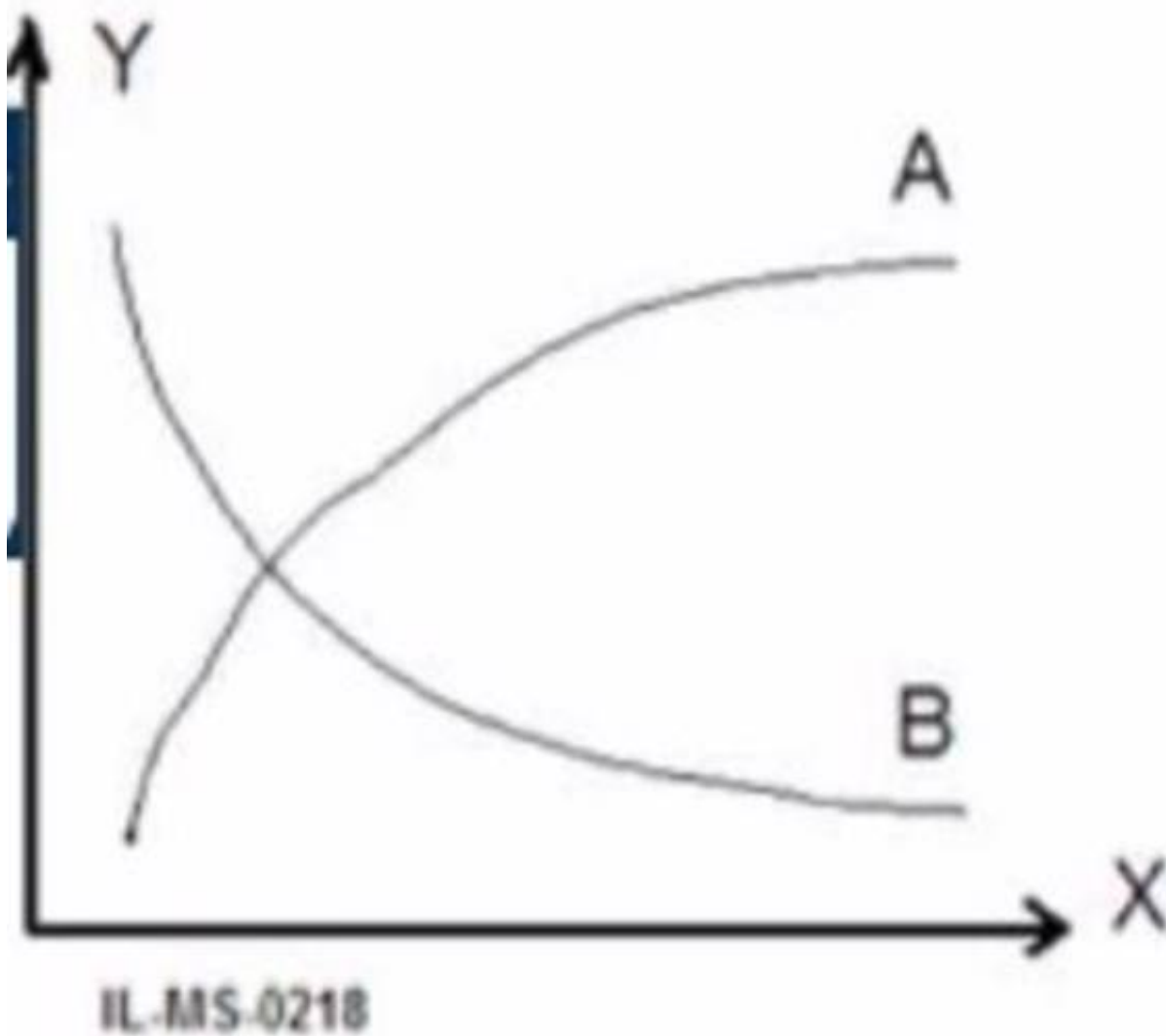
The features of the test object to be tested and those excluded from the testing represent information that is usually included in a test plan and, in the given test plan, it is more likely to be specified within "Test Scope" rather than in the other two sections mentioned. The test scope defines the boundaries and limitations of the testing activities, such as the test items, the features to be tested, the features not to be tested, the test objectives, the test environment, the test resources, the test assumptions, the test risks, etc. The test scope helps to establish a common understanding of what is included and excluded from the testing, and to avoid ambiguity, confusion, or misunderstanding among the stakeholders. The other two sections, "Testing Communication" and "Stakeholders", are also important parts of a test plan, but they do not directly address the features of the test object. The testing communication describes the methods, frequency, and responsibilities for the communication and reporting of the testing progress, status, issues, and results. The stakeholders identify the roles and responsibilities of the people involved in or affected by the testing activities, such as the test manager, the test team, the project manager, the developers, the customers, the users, etc. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.1, Test Planning1

? ISTQB® Glossary of Testing Terms v4.0, Test Plan, Test Scope2

NEW QUESTION 56

The following chart represents metrics related to testing of a project that was completed. Indicate what is represented by tie lines A, B and the axes X.Y



- A)
X - Time
Y - Cost
A - Cost of test (per week)
B - Cost of finding a single bug (per week)
- B)
X - Time
Y - Number of defects
A - Number of open defects
B - Number of closed defects
- C)
X - Time
Y - Percent
A - % of functional tests in the test suite
B - % of non-functional tests in the test suite
- D)
X - Time
Y - Count
A - Total number of executed tests
B - Number of open bugs

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Explanation:

Option D correctly explains what is represented by the lines A, B and the axes X, Y in a testing metrics chart. According to option D:

? X-axis represents Time

? Y-axis represents Count

? Line A represents Number of open bugs

? Line B represents Total number of executed tests

This information is essential in understanding and analyzing the testing metrics of a completed project.

References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Section 2.5.1, Page 35.

NEW QUESTION 60

Which of the following types of tools is BEST suited for determining source code compliance with the guidelines provided by a coding standard?

- A. Containerisation tool
- B. Fault seeding tool.
- C. Static analysis tool.
- D. Test data preparation tool

Answer: C

Explanation:

A static analysis tool is best suited for determining source code compliance with coding standards. These tools analyze the code without executing it and can check for adherence to coding standards, syntax errors, and other static properties of the code. The ISTQB CTFL syllabus emphasizes the role of static analysis tools in verifying that code meets predefined standards and guidelines.

References:ISTQB CTFL Syllabus, Section on static testing and tools.

NEW QUESTION 65

Which of the following statements about the testing quadrants is TRUE?

- A. The higher the number of the testing quadrant, the more important the tests associated with this quadrant are
- B. Automated acceptance tests produced during BDD and ATDD are classified in quadrant 02.
- C. Exploratory tests are classified in quadrant Q3, and they are usually included in a continuous integration process.
- D. Automated unit tests produced during TDD are classified in quadrant Q4 as they are technology facing.

Answer: B

Explanation:

The correct statement is B. According to the ISTQB CTFL syllabus, the testing quadrants help to categorize tests based on their purpose and whether they are technology-facing or business-facing, and whether they support the team or critique the product. Quadrant Q2 includes tests that are business-facing and support the team, such as automated acceptance tests produced during Behavior-Driven Development (BDD) and Acceptance Test-Driven Development (ATDD) .

Quadrant Q3 includes business-facing tests that critique the product, such as exploratory testing, usability testing, and user acceptance testing. These tests are typically manual and focus on evaluating the product from a user perspective, rather than being part of a continuous integration process.

Quadrant Q4 includes technology-facing tests that critique the product, such as performance tests, security tests, and other non-functional tests, which can be automated but are not related to unit tests produced during TDD .

NEW QUESTION 66

A Test Manager conducts risk assessment for a project. One of the identified risks is: The sub-contractor may fail to meet his commitment". If this risk materializes. it will lead to delay in completion of testing required for the current cycle.

Which of the following sentences correctly describes the risk?

- A. It is a product risk since any risk associated with development timeline is a product risk.
- B. It is no longer a risk for the Test Manager since an independent party (the sub- contractor) is now managing it
- C. It is a object risk since successful completion of the object depends on successful and timely completion of the tests
- D. It is a product risk since default on part of the sub-contractor may lead to delay in release of the product

Answer: D

Explanation:

? A product risk is a risk that affects the quality or timeliness of the software product being developed or tested1. Product risks are related to the requirements, design, implementation, verification, and maintenance of the software product2.

? The risk of the sub-contractor failing to meet his commitment is a product risk, as it could cause a delay in the completion of the testing required for the current cycle, which in turn could affect the release date of the product. The release date is an important aspect of the product quality, as it reflects the customer satisfaction and the market competitiveness of the product3.

? The other options are not correct because: References =

? 1 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 97

? 2 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 98

? 3 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 99

? 4 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 100

? 5 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 101

? 6 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 102

NEW QUESTION 70

A test manager decided to skip static testing since he believes bugs can be found easily by doing dynamic testing. Was this decision right or wrong?

- A. The decision was wron
- B. Ensuring quality mandates that static testing is performed after performing the dynamic testing.
- C. The decision was righ
- D. Static testing is usually redundant if a product is planned to go through a full-cycle of dynamic testing.
- E. The decision was righ
- F. Most of the bugs are easier to identify during the dynamic testing.
- G. The decision was wron
- H. Static testing can find defects early in the development process, reducing the overall cost of testing and development

Answer: D

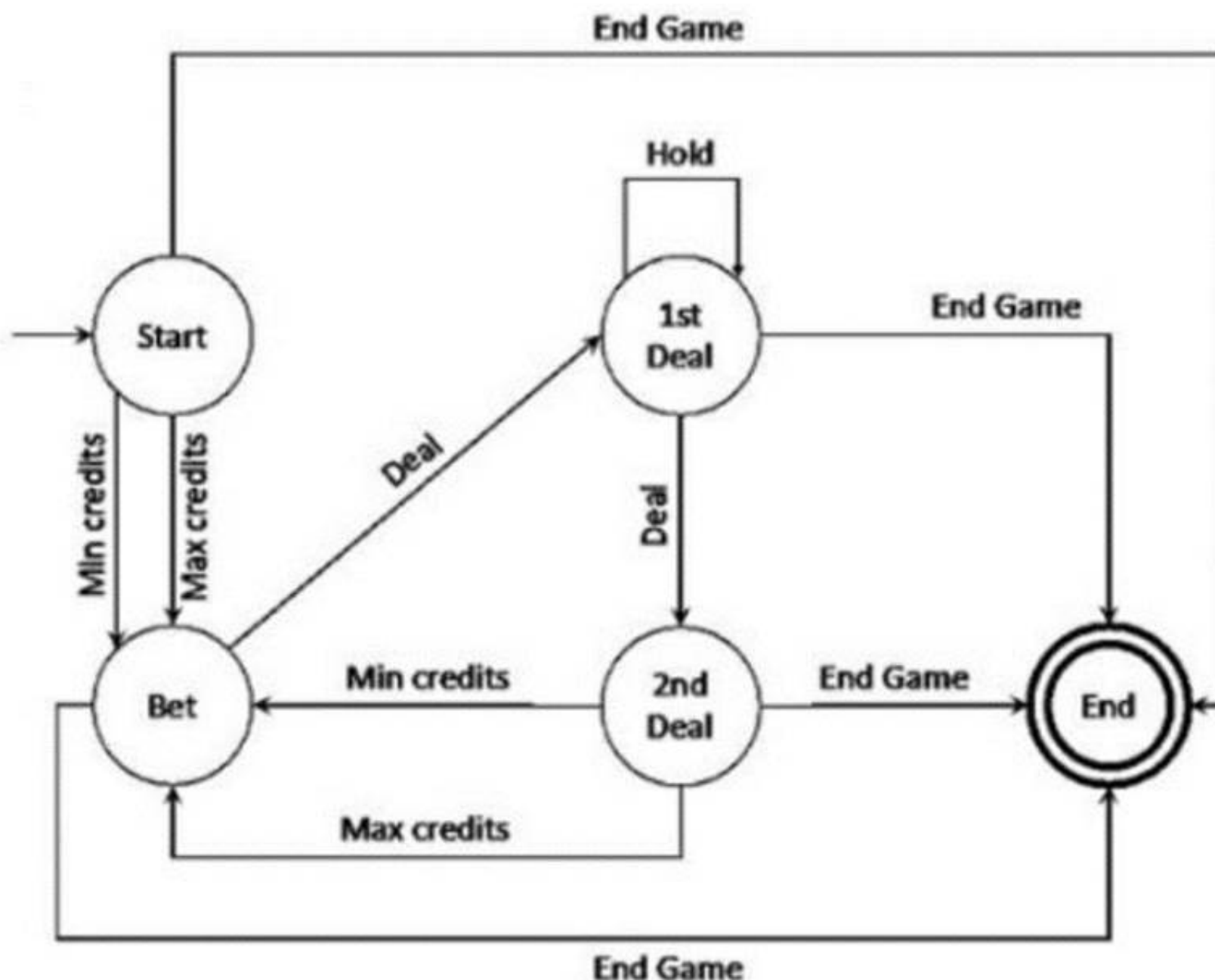
Explanation:

Static testing is a form of testing that does not involve executing the software or system under test. It includes activities such as reviews, inspections, walkthroughs, and analysis of documents, code, and models. Static testing can find defects early in the development process, before they become more expensive and difficult to fix in later stages. Static testing can also improve the quality of the software or system by preventing defects from being introduced in the first place. Static testing can complement dynamic testing, which involves executing the software or system under test and checking the results against expected outcomes. Dynamic testing can find defects that static testing may miss, such as performance, usability, or integration issues. However, dynamic testing alone is not sufficient to ensure quality, as it may not cover all possible scenarios, inputs, or paths. Therefore, a test manager who decides to skip static testing is making a wrong decision, as he or she is ignoring the benefits of static testing and relying solely on dynamic testing, which may not be effective or efficient enough to find and

prevent defects. References = ISTQB Certified Tester Foundation Level Syllabus, Version 4.0, 2018, Section 2.1.1, page 14; ISTQB Glossary of Testing Terms, Version 4.0, 2018, page 36; ISTQB CTFL 4.0 - Sample Exam - Answers, Version 1.1, 2023, Question 3, page 9.

NEW QUESTION 71

Consider the following simplified version of a state transition diagram that specifies the behavior of a video poker game:



What is the minimum number of test cases needed to cover every unique sequence of up to 3 states/2 transitions starting in the "Start" state and ending in the "End" state?

- A. 1
- B. 2
- C. 3
- D. 4

Answer: D

Explanation:

The minimum number of test cases needed to cover every unique sequence of up to 3 states/2 transitions starting in the ??Start?? state and ending in the ??End?? state is 4. This is because there are 4 unique sequences of up to 3 states/2 transitions starting in the ??Start?? state and ending in the ??End?? state:

? Start -> Bet -> End

? Start -> Deal -> End

? Start -> 1st Deal -> End

? Start -> 2nd Deal -> End References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents.

NEW QUESTION 72

A new web app aims at offering a rich user experience. As a functional tester, you have run some functional tests to verify that, before releasing the app, such app works correctly on several mobile devices, all of which are listed as supported devices within the requirements specification. These tests were performed on stable and isolated test environments where you were the only user interacting with the application. All tests passed, but in some of those tests you observed the following issue: on some mobile devices only, the response time for two web pages containing images was extremely slow. Based only on the given information, which of the following recommendation would you follow?

- A. You should open a defect report providing detailed information on which devices and by running which tests you observed the issue
- B. The issue is related to performance efficiency, not functionalit
- C. Thus, as a functional tester, you should not open any defect report as all the functional tests passed
- D. You should not open any defect report as the problem is most likely due to poor hardware equipment on the devices where you observed the issue
- E. You should not open any defect report and inform the test manager that the devices on which you observed the issue should no longer be supported so that they will be removed from the requirements specification

Answer: A

Explanation:

As a functional tester, you should open a defect report providing detailed information on which devices and by running which tests you observed the issue. A defect report is a document that records the occurrence, nature, and status of a defect detected during testing, and provides information for further investigation

and resolution. A defect report should include relevant information such as the defect summary, the defect description, the defect severity, the defect priority, the defect status, the defect origin, the defect category, the defect reproduction steps, the defect screenshots, the defect attachments, etc. Opening a defect report is a good practice for any tester who finds a defect in the software system, regardless of the type or level of testing performed. The other options are not recommended, because:

? The issue is related to performance efficiency, not functionality, but that does not mean that as a functional tester, you should not open any defect report as all the functional tests passed. Performance efficiency is a quality characteristic that measures how well the software system performs its functions under stated conditions, such as the response time, the resource utilization, the throughput, etc. Performance efficiency is an important aspect of the user experience, especially for web applications that run on different devices and networks. Even if the functional tests passed, meaning that the software system met the functional requirements, the performance issue observed on some devices could still affect the user satisfaction, the usability, the reliability, and the security of the software system. Therefore, as a functional tester, you have the responsibility to report the performance issue as a defect, and provide as much information as possible to help the developers or the performance testers to investigate and resolve it.

NEW QUESTION 77

The fact that defects are usually not evenly distributed among the various modules that make up a software application, but rather their distribution tend to reflect the Pareto principle:

- A. is a false myth
- B. is expressed by the testing principle referred to as 'Tests wear out'
- C. is expressed by the testing principle referred to as 'Defects cluster together'
- D. is expressed by the testing principle referred to as 'Bug prediction'

Answer: C

Explanation:

The fact that defects are usually not evenly distributed among the various modules that make up a software application, but rather their distribution tend to reflect the Pareto principle, is expressed by the testing principle referred to as ??Defects cluster together??. This principle states that a small number of modules contain most of the defects detected, or that a small number of causes are responsible for most of the defects. This principle can be used to guide the test analysis and design activities, by prioritizing the testing of the most critical or risky modules, or by applying more rigorous test techniques to them. Therefore, option C is the correct answer.

References: ISTQB® Certified Tester Foundation Level Syllabus v4.01, Section 1.2.1, page 11; ISTQB® Glossary v4.02, page 16.

NEW QUESTION 82

In which one of the following test techniques are test cases derived from the analysis of the software architecture?

- A. Black-box test techniques.
- B. Experience-based test techniques.
- C. Checklist-based test techniques.
- D. White-box test techniques.

Answer: D

Explanation:

White-box test techniques are test design techniques where the test cases are derived from the internal structure of the software, including its architecture, code, and logical flow. These techniques involve the tester having knowledge of the internal workings of the software to create test cases that ensure all possible paths and conditions are tested. This is in contrast to black-box test techniques, which focus on input-output behavior without considering the internal structure. Reference: ISTQB CTFL Syllabus V4.0, Section 4.3

NEW QUESTION 86

Which two of the following statements describe the advantages provided by good traceability between the test basis and test work products?

- A. Analyzing the impact of changes.i
- B. A measure of code quality.ii
- C. Accurate test estimation.i
- D. Making testing auditable
- E. Select the correct Answer:
- F. i and ii
- G. i and iv
- H. i and iii
- I. ii and iii

Answer: B

Explanation:

Good traceability between the test basis and test work products provides several advantages: i.Analyzing the impact of changes:Traceability allows for easy identification of which parts of the test work products will be affected by changes in the requirements or design, facilitating impact analysis. iv.Making testing auditable:Traceability ensures that there is a clear connection between the requirements and the test cases, which makes the testing process auditable and provides evidence that all requirements have been tested.

NEW QUESTION 87

Which of the following lists factors That contribute to PROJECT risks?

- A. skill and staff shortages; problems in defining the right requirements, contractual issues.
- B. skill and staff shortages; software does not perform its intended functions; problems in defining the right requirements.
- C. problems in defining the right requirements; contractual issues; poor software quality characteristics.
- D. poor software quality characteristics; software does not perform its intended functions.

Answer: A

Explanation:

Project risks are the uncertainties or threats that may affect the project objectives, such as scope, schedule, cost, and quality. According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, some of the factors that contribute to project risks are:

? Skill and staff shortages: This factor refers to the lack of adequate or qualified human resources to perform the project tasks. This may result in delays, errors, rework, or low productivity.

? Problems in defining the right requirements: This factor refers to the difficulties or ambiguities in eliciting, analyzing, specifying, validating, or managing the requirements of the project. This may result in misalignment, inconsistencies, gaps, or changes in the requirements, affecting the project scope and quality.

? Contractual issues: This factor refers to the challenges or disputes that may arise from the contractual agreements between the project parties, such as clients, suppliers, vendors, or subcontractors. This may result in legal, financial, or ethical risks, affecting the project delivery and satisfaction.

The other options are not correct because they list factors that contribute to PRODUCT risks, not project risks. Product risks are the uncertainties or threats that may affect the quality or functionality of the software product or system. Some of the factors that contribute to product risks are:

? Poor software quality characteristics: This factor refers to the lack of adherence or compliance to the quality attributes or criteria of the software product or system, such as reliability, usability, security, performance, or maintainability. This may result in defects, failures, or dissatisfaction of the users or stakeholders.

? Software does not perform its intended functions: This factor refers to the deviation or discrepancy between the expected and actual behavior or output of the software product or system. This may result in errors, faults, or malfunctions of the software product or system.

References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, Chapter 1: Fundamentals of Testing, Section 1.5: Risks and Testing, Pages 14-16.

NEW QUESTION 90

Which of the following statements about the value of maintaining traceability between the test basis and test work products is not true?

- A. Traceability can be useful for assessing the impact of a change to a test basis item on the corresponding tests
- B. Traceability can be useful for determining how many test basis items are covered by the corresponding tests
- C. Traceability can be useful for determining the most suitable test techniques to be used in a testing project
- D. Traceability can be useful to support the needs required by the auditing of testing

Answer: C

Explanation:

Traceability is the ability to trace the relationships between the items of the test basis, such as the requirements, the design, the risks, etc., and the test artifacts, such as the test cases, the test results, the defects, etc. Traceability can provide various benefits for the testing process, such as improving the test coverage, the test quality, the test efficiency, and the test communication. However, not all the statements given are true about the value of maintaining traceability between the test basis and test work products. The statement that is not true is option C, which says that test objectives should be the same for all test levels, although the number of tests designed at various levels can vary significantly. This statement is false, because test objectives are the goals or the purposes of testing, which can vary depending on the test level, the test type, the test technique, the test environment, the test stakeholder, etc. Test objectives can be defined in terms of the test basis, the test coverage, the test quality, the test risk, the test cost, the test time, etc. Test objectives should be specific, measurable, achievable, relevant, and time-bound, and they should be aligned with the project objectives and the quality characteristics. Test objectives should not be the same for all test levels, as different test levels have different focuses, scopes, and perspectives of testing, such as component testing, integration testing, system testing, and acceptance testing. The other statements are true about the value of maintaining traceability between the test basis and test work products, such as:

? Traceability can be useful for assessing the impact of a change to a test basis item on the corresponding tests: This statement is true, because traceability can help to identify which tests are affected by a change in the test basis, such as a new requirement, a modified design, a revised risk, etc., and to determine the necessary actions to update, re-execute, or re-evaluate the tests. Traceability can also help to estimate the effort, the cost, and the time needed to implement the change and to verify its impact on the software system.

? Traceability can be useful for determining how many test basis items are covered by the corresponding tests: This statement is true, because traceability can help to measure the test coverage, which is the degree to which the test basis is exercised by the test cases. Traceability can help to identify which test basis items are covered, partially covered, or not covered by the tests, and to evaluate the adequacy, the completeness, and the effectiveness of the testing process. Traceability can also help to identify the gaps, the overlaps, or the redundancies in the test coverage, and to prioritize, optimize, or improve the test cases.

? Traceability can be useful to support the needs required by the auditing of testing:

This statement is true, because traceability can help to provide evidence, documentation, and justification for the testing activities, results, and outcomes.

Traceability can help to demonstrate that the testing process follows the standards, the regulations, the policies, and the best practices that are applicable to the software system, the project, or the organization. Traceability can also help to verify that the testing process meets the expectations, the needs, and the satisfaction of the users and the stakeholders. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.2.2, Testing Policies, Strategies, and Test Approaches1

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.1, Test Planning1

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.2, Test Monitoring and Control1

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.1.3, Test Analysis and Design1

? ISTQB® Glossary of Testing Terms v4.0, Traceability, Test Basis, Test Artifact, Test Objective, Test Level, Test Coverage, Test Quality, Test Risk, Test Cost, Test Time2

NEW QUESTION 92

Mark the correct sentences:

- * Defects are a result of environmental conditions and are also referred to as "Failures"
- * A human mistake may produce a defect
- * A system may totally fail to operate correctly when a failure exists in it
- * When a defect exists in a system it may result in a failure
- * Defects occur only as a result of technology changes

- A. II, IV
- B. I, II
- C. IV, V
- D. II, III, IV

Answer: A

Explanation:

? The question is about marking the correct sentences among the given statements related to defects, failures, and mistakes. According to the ISTQB glossary, the definitions of these terms are1:

? Therefore, out of the five given statements, only two are correct, namely:

? The other three statements are incorrect, namely: References:
? 1: ISTQB Glossary of Testing Terms 4.0, 2023, available at ISTQB) and ASTQB).

NEW QUESTION 97

Which one of the following is a typical entry criteria for testing?

- A. Planned tests have been executed.
- B. Availability of testable requirements.
- C. The number of unresolved defects is within an agreed limit.
- D. The number of estimated remaining defects is sufficiently low.

Answer: B

Explanation:

A typical entry criterion for testing is the availability of testable requirements. Testable requirements provide a basis for designing and executing test cases. Without clear and testable requirements, it is challenging to determine what needs to be tested and to create effective test cases. Entry criteria ensure that the necessary preconditions are met before testing begins, which helps in conducting efficient and effective testing. References:ISTQB CTFL Syllabus, Section 5.1.3, "Entry and Exit Criteria."

NEW QUESTION 98

The statement: "Test activities should start in the early stages of the lifecycle, adhering to the testing principle of early testing?? is relevant to which of the recognized software development models?

- A. Sequential development model.
- B. Iterative development model.
- C. Incremental development model.
- D. All the above

Answer: D

Explanation:

The principle of early testing is applicable to all recognized software development models, including sequential, iterative, and incremental models. Starting test activities early in the lifecycle helps in identifying and addressing defects as soon as possible, which can save time and costs by preventing defects from propagating to later stages of development. This proactive approach enhances the overall quality and efficiency of the software development process.Reference:ISTQB CTFL Syllabus V4.0, Section 1.3

NEW QUESTION 102

Following a risk-based testing approach you have designed 10 tests to cover a product risk with a high-risk level. You want to estimate, adopting the three-point test estimation technique, the test effort required to reduce the risk level to zero by executing those 10 tests. You made the following three initial estimates:

- most optimistic = 6 person hours
- most likely = 30 person hours
- most pessimistic = 54 person hours

Based only on the given information, which of the following answers about the three-point test estimation technique applied to this problem is true?

- A. The final estimate is between 22 person hours and 38 person hours
- B. The final estimate is exactly 30 person hours because the technique uses the initial most likely estimate as the final estimate
- C. The final estimate is between 6 person hours and 54 person hours
- D. The final estimate is exactly 30 person hours because the technique uses the arithmetic mean of the three initial estimates as the final estimate

Answer: A

Explanation:

The three-point test estimation technique is a method of estimating the test effort based on three initial estimates: the most optimistic, the most likely, and the most pessimistic. The technique uses a weighted average of these three estimates to calculate the final estimate, which is also known as the expected value. The formula for the expected value is:

Expected value = (most optimistic + 4 * most likely + most pessimistic) / 6 Using the given values, the expected value is:

Expected value = (6 + 4 * 30 + 54) / 6 Expected value = 30 person hours

However, the expected value is not the only factor to consider when estimating the test effort. The technique also calculates the standard deviation, which is a measure of the variability or uncertainty of the estimates. The formula for the standard deviation is: Standard deviation = (most pessimistic - most optimistic) / 6

Using the given values, the standard deviation is:

Standard deviation = (54 - 6) / 6 Standard deviation = 8 person hours

The standard deviation can be used to determine a range of possible values for the test effort, based on a certain level of confidence. For example, using a 68% confidence level, the range is:

Expected value ?? standard deviation Using the calculated values, the range is: 30 ?? 8 person hours

Therefore, the final estimate is between 22 person hours and 38 person hours, which is option A.

References: ISTQB® Certified Tester Foundation Level Syllabus v4.01, Section 2.3.2, page 24-25; ISTQB® Glossary v4.02, page 33.

NEW QUESTION 106

Which of the following statements is NOT true about Configuration management and software testing?

- A. Configuration management helps maintain consistent versions of software artifacts.
- B. Configuration management supports the build process, which is essential for delivering a test release into the test environment.
- C. When testers report defects, they need to reference version-controlled items.
- D. Version controlled test ware increases the chances of finding defects in the software under test.

Answer: D

Explanation:

Reference:ISTQB CTFL Syllabus V4.0, Section 5.4

NEW QUESTION 107

Which of the following issues cannot be identified by static analysis tools?

- A. Very low MTBF (Mean Time Between failure)
- B. Potentially endless loops
- C. Referencing a variable with an undefined value
- D. Security vulnerabilities

Answer: A

Explanation:

Static analysis tools are software tools that examine the source code of a program without executing it. They can detect various types of issues, such as syntax errors, coding standards violations, security vulnerabilities, and potential bugs¹². However, static analysis tools cannot identify issues that depend on the runtime behavior or performance of the program, such as very low MTBF (Mean Time Between failure)³. MTBF is a measure of the reliability of a system or component. It is calculated by dividing the total operating time by the number of failures. MTBF reflects how often a system or component fails during its expected lifetime. Static analysis tools cannot measure MTBF because they do not run the program or observe its failures. MTBF can only be estimated by dynamic testing, which involves executing the program under various conditions and collecting data on its failures⁴. Therefore, very low MTBF is an issue that cannot be identified by static analysis tools. The other options, such as potentially endless loops, referencing a variable with an undefined value, and security vulnerabilities, are issues that can be identified by static analysis tools. Static analysis tools can detect potentially endless loops by analyzing the control flow and data flow of the program and checking for conditions that may never become false⁵. Static analysis tools can detect referencing a variable with an undefined value by checking the scope and initialization of variables and reporting any use of uninitialized variables⁶. Static analysis tools can detect security vulnerabilities by checking for common patterns of insecure code, such as buffer overflows, SQL injections, cross-site scripting, and weak encryption. References = What Is Static Analysis? Static Code Analysis Tools - Perforce Software, How Static Code Analysis Works | Perforce, Static Code Analysis: Techniques, Top 5 Benefits & 3 Challenges, What is MTBF? Mean Time Between Failures Explained | Perforce, Static analysis tools - Software Testing MCQs - CareerRide, ISTQB_Chapter3 | Quizizz, [Static Code Analysis for Security Vulnerabilities | Perforce].

NEW QUESTION 111

Which of the following statements are true?

- * 1. Early and frequent feedback helps to avoid requirements misunderstanding.
- * 2. Early feedback allows teams to do more with less.
- * 3. Early feedback allows the team to focus on the most Important features.
- * 4. Early and frequent feedback clarifies customer feedback by applying static testing techniques

Select the correct Answer:

- A. 3
- B. 2
- C. 1
- D. 4

Answer: C

Explanation:

The statement "Early and frequent feedback helps to avoid requirements misunderstanding" is true. Early feedback from stakeholders, through reviews and other static testing techniques, helps clarify requirements and ensures that any misunderstandings are addressed promptly. This practice aligns with Agile principles and contributes to developing software that meets user needs more accurately. References: ISTQB CTFL Syllabus, Section 2.1.1, "The Influence of Development Models on Testing" and Section 3.2.1, "The Advantages of Early Feedback."

NEW QUESTION 114

Who of the following has the best knowledge to decide what tests in a test project should be automated?

- A. The developer
- B. The customer
- C. The development manager
- D. The test leader

Answer: D

Explanation:

The test leader is the person who is responsible for planning, monitoring, and controlling the test activities and resources in a test project. The test leader should have the best knowledge of the test objectives, scope, risks, resources, schedule, and quality criteria. The test leader should also be aware of the test automation criteria, such as the execution frequency, the test support, the team education, the roles and responsibilities, and the devs and testers collaboration¹. Based on these factors, the test leader can decide which tests are suitable for automation and which are not, and prioritize them accordingly. The test leader can also coordinate with the test automation engineers, the developers, and the stakeholders to ensure the alignment of the test automation strategy with the test project goals and expectations. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 2, Section 2.3.1, Page 152; ISTQB Glossary of Testing Terms v4.0, Page 403; ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 6, Section 6.1.1, Page 514; Top 8 Test Automation Criteria You Need To Fulfill - QAMIND¹

NEW QUESTION 119

Which of the following about typical information found within a test plan is FALSE?

- A. The need to temporarily have additional test personnel available for specific test phases and/or test activities
- B. The conditions that must be met in order for the test execution activities to be considered completed.
- C. The list of the product risks which have not been fully mitigated at the end of test execution.
- D. The conditions that must be met for part of all the planned activities to be suspended and resumed.

Answer: C

Explanation:

A typical test plan includes various elements, such as resource requirements, test completion criteria, and suspension/resumption criteria. However, the list of

product risks that have not been fully mitigated is generally not included in the test plan but rather in the risk management documentation.

? The test plan focuses on planning and executing tests, including resource allocation and defining criteria for test suspension and resumption.

? While risk management is crucial, unmitigated risks are typically documented in risk logs or separate risk management plans.

Reference: ISTQB CTFL Syllabus V4.0, Chapter 5.1.1, Test Planning.

NEW QUESTION 124

The whole-team approach:

- A. promotes the idea that all team members should have a thorough understanding of test techniques
- B. is a consensus-based approach that engages the whole team in estimating the user stories
- C. promotes the idea that all team members should be responsible for the quality of the product
- D. is mostly adopted in projects aimed at developing safety-critical systems, as it ensures the highest level of testing independence

Answer: C

Explanation:

This answer is correct because the whole-team approach is a way of working in agile projects where all team members share the responsibility for the quality of the product, and collaborate on delivering value to the customer. The whole-team approach involves testers, developers, business analysts, product owners, and other stakeholders in planning, designing, developing, testing, and delivering the product. The whole-team approach fosters communication, feedback, learning, and continuous improvement within the

team. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 3.1.1.1

NEW QUESTION 129

During component testing of a program if 100% decision coverage is achieved, which of the following coverage criteria is also guaranteed to be 100%?

- A. 100% State transition coverage
- B. 100% Equivalence class coverage
- C. 100% Boundary value coverage
- D. 100% Statement coverage

Answer: D

Explanation:

Statement coverage is a structural coverage metric that measures the percentage of executable statements in the source code that are executed by a test suite¹. Decision coverage is another structural coverage metric that measures the percentage of decision outcomes (such as branches or conditions) in the source code that are executed by a test suite¹. Decision coverage is a stronger metric than statement coverage, because it requires that every possible outcome of each decision is tested, while statement coverage only requires that every statement is executed at least once². Therefore, if a test suite achieves 100% decision coverage, it also implies that it achieves 100% statement coverage, because every statement in every branch or condition must have been executed. However, the converse is not true: 100% statement coverage does not guarantee 100% decision coverage, because some branches or conditions may have multiple outcomes that are not tested by the test suite². For example, consider the following pseudocode:

if (x > 0) then print(??Positive??) else print(??Non-positive??) end if

A test suite that executes this code with x = 1 and x = -1 will achieve 100% statement coverage, because both print statements are executed. However, it will not achieve 100% decision coverage, because the condition x > 0 has only been tested with two outcomes: true and false. The third possible outcome, x = 0, has not been tested by the test suite. Therefore, the test suite may miss a potential bug or error in the condition or the branch. The other options, such as state transition coverage, equivalence class coverage, and boundary value coverage, are not guaranteed to be 100% by achieving 100% decision coverage. State transition coverage is a structural coverage metric that measures the percentage of transitions between states in a state machine that are executed by a test suite³.

Equivalence class coverage is a functional coverage metric that measures the percentage of equivalence classes (or partitions) of input or output values that are tested by a test suite⁴. Boundary value coverage is another functional coverage metric that measures the percentage of boundary values (or extreme values) of input or output ranges that are tested by a test suite⁴. These metrics are independent of decision coverage, because they are based on different aspects of the system under test, such as its behavior, functionality, or specification. Therefore, achieving 100% decision coverage does not imply achieving 100% of any of these metrics, and vice versa. References = ISTQB® Certified Tester Foundation Level Syllabus v4.0, Test Coverage in Software Testing - Guru99, Structural Coverage Metrics - MATLAB & Simulink - MathWorks India, Test Design Coverage in Software Testing - GeeksforGeeks.

NEW QUESTION 133

A software company decides to invest in reviews of various types. The thought process they have is that each artifact needs to be reviewed using only one of the review methods depending on the criticality of the artifact.

- A. The thought process is incorrec
- B. The whole company should adopt same standard for review of all artifacts.
- C. The thought process is correc
- D. The whole company should decide or the review method based on their CMM level.
- E. The thought process is incorrec
- F. Same artifact can be reviewed using different review methods
- G. The thought process is correc
- H. It wastes time to review same artifact using efferent review methods

Answer: C

Explanation:

The thought process of the software company is incorrect, because it assumes that each artifact can be reviewed using only one review method, and that the review method depends solely on the criticality of the artifact. This is a simplistic and rigid approach that does not consider the benefits and limitations of different review methods, the context and purpose of the review, and the feedback and improvement opportunities that can be gained from multiple reviews. According to the CTFL 4.0 Syllabus, the selection of review methods should be based on several factors, such as the type and level of detail of the artifact, the availability and competence of the reviewers, the time and budget constraints, the expected defects and risks, and the desired outcomes and quality criteria. Moreover, the same artifact can be reviewed using different review methods at different stages of the development lifecycle, to ensure that the artifact meets the changing requirements, standards, and expectations of the stakeholders. For example, a requirement specification can be reviewed using an informal review method, such as a walkthrough, to get an initial feedback from the users and developers, and then using a formal review method, such as an inspection, to verify the completeness, correctness, and consistency of the specification. Therefore, the software company should adopt a more flexible and context-sensitive approach to selecting and applying review methods for different artifacts, rather than following a fixed and arbitrary rule. References = CTFL 4.0 Syllabus, Section 3.2.1, page

31-32; Section 3.2.2, page 33-34; Section 3.2.3, page 35-36.

NEW QUESTION 135

Which of the following statements about static testing and dynamic testing is true?

- A. Unlike dynamic testing, which can be also performed manually, static testing cannot be performed without specialized tools
- B. Static testing is usually much less cost-effective than dynamic testing
- C. Unlike dynamic testing, which focuses on detecting potential defects, static testing focuses on detecting failures which may be due to actual defects
- D. Both static testing and dynamic testing can be used to highlight issues associated with non-functional characteristics

Answer: D

Explanation:

This answer is correct because static testing and dynamic testing are both types of testing that can be used to highlight issues associated with non-functional characteristics, such as usability, performance, security, reliability, etc. Static testing is a type of testing that involves the analysis of software work products, such as requirements, design, code, or test cases, without executing them. Dynamic testing is a type of testing that involves the execution of software work products, such as code or test cases, using inputs and verifying outputs. Both static testing and dynamic testing can be applied to different test levels and test types, and can use different test techniques and tools, to evaluate the non-functional characteristics of the softwareproduct. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.2.1.1, Section 2.2.1.2

NEW QUESTION 136

Which one of the following statements IS NOT a valid objective of testing?

- A. To build confidence in the level of quality of the test object.
- B. To find all defects in a product, ensuring the product is defect free.
- C. To find failures and defects
- D. To evaluate work products such as requirements, user stories, design, and code.

Answer: B

Explanation:

Reference:ISTQB CTFL Syllabus V4.0, Section 1.1.1

NEW QUESTION 138

The following decision table is used to assist a doctor in determining the drug therapy to prescribe for a patient (aged 6 to 65 years) diagnosed with acute sinusitis. The table consists of three Boolean conditions and six actions

	1	2	3	4	5	6	7	8
Conditions								
Is the patient over 18 years old?	F	F	F	F	T	T	T	T
Is the patient allergic to Penicillin?	F	F	T	T	F	F	T	T
Is the patient taking anticoagulant therapy?	F	T	F	T	F	T	F	T
Actions								
Amoxicillin is the therapy of choice					X			
Levofloxacin is the therapy of choice			X				X	
Cefuroxime is the therapy of choice	X							
Necessary consultation with the hematologist		X		X		X		X
Full dosage recommended for 10 days					X		X	
Half of the full recommended dosage for 10 days	X		X					

Based only on the given information, which of the following statements is TRUE?

- A. Column 7 represents an impossible situation and thus can be deleted
- B. Columns 1 and 3 can be merged into a single column
- C. Columns 2, 4, 6 and 8 can be merged into a single column
- D. Columns 5 and 7 can be merged into a single column

Answer: B

Explanation:

Decision tables are used to model complex decision logic by considering different combinations of conditions and actions. Based on the given decision table for prescribing drug therapy:

? Column 1 and Column 3 both result in the same actions (prescribing Amoxicillin).

? These columns can be merged because the actions taken do not depend on

whether the patient is taking anticoagulant therapy (both are 'T' for this condition). Thus, combining these columns simplifies the decision table without losing any information. Reference: ISTQB CTFL Syllabus V4.0, Chapter 4.2.3, Decision Table Testing.

NEW QUESTION 140

Which of the following statements about how different types of test tools support testers is true?

- A. The support offered by a test data preparation tool is often leveraged by testers to run automated regression test suites
- B. The support offered by a performance testing tool is often leveraged by testers to run load tests
- C. The support offered by a bug prediction tool is often used by testers to track the bugs they found
- D. The support offered by a continuous integration tool is often leveraged by testers to automatically generate test cases from a model

Answer: B

Explanation:

The support offered by a performance testing tool is often leveraged by testers to run load tests, which are tests that simulate a large number of concurrent users or transactions on the system under test, in order to measure its performance, reliability, and scalability. Performance testing tools can help testers to generate realistic workloads, monitor system behavior, collect and analyze performance metrics, and identify performance bottlenecks. The other statements are false, because:

? A test data preparation tool is a tool that helps testers to create, manage, and manipulate test data, which are the inputs and outputs of test cases. Test data preparation tools are not directly related to running automated regression test suites, which are test suites that verify that the system still works as expected after changes or modifications. Regression test suites are usually executed by test execution tools, which are tools that can automatically run test cases and compare actual results with expected results.

? A bug prediction tool is a tool that uses machine learning or statistical techniques to predict the likelihood of defects in a software system, based on various factors such as code complexity, code churn, code coverage, code smells, etc. Bug prediction tools are not used by testers to track the bugs they found, which are the actual defects that have been detected and reported during testing. Bugs are usually tracked by defect management tools, which are tools that help testers to record, monitor, analyze, and resolve defects.

? A continuous integration tool is a tool that enables the integration of code changes from multiple developers into a shared repository, and the execution of automated builds and tests, in order to ensure the quality and consistency of the software system. Continuous integration tools are not used by testers to automatically generate test cases from a model, which are test cases that are derived from a representation of the system under test, such as a state diagram, a decision table, a use case, etc. Test cases can be automatically generated by test design tools, which are tools that support the implementation and maintenance of test cases, based on test design specifications or test models. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 3.4.1, Types of Test Tools

? ISTQB® Glossary of Testing Terms v4.0, Performance Testing Tool, Test Data Preparation Tool, Bug Prediction Tool, Continuous Integration Tool, Test Execution Tool, Defect Management Tool, Test Design Tool

NEW QUESTION 144

Which of the following statements about exploratory testing is true?

- A. Exploratory testing is an experience-based test technique in which testers explore the requirements specification to detect non testable requirements
- B. When exploratory testing is conducted following a session-based approach, the issues detected by the testers can be documented in session sheets
- C. Exploratory testing is an experience-based test technique used by testers during informal code reviews to find defects by exploring the source code
- D. In exploratory testing, testers usually produce scripted tests and establish bidirectional traceability between these tests and the items of the test basis

Answer: B

Explanation:

Exploratory testing is an experience-based test technique in which testers dynamically design and execute tests based on their knowledge, intuition, and learning of the software system, without following predefined test scripts or test cases. Exploratory testing can be conducted following a session-based approach, which is a structured way of managing and measuring exploratory testing. In a session-based approach, the testers perform uninterrupted test sessions, usually lasting between 60 and 120 minutes, with a specific charter or goal, and document the issues detected, the test coverage achieved, and the time spent in session sheets. Session sheets are records of the test activities, results, and observations during a test session, which can be used for reporting, debriefing, and learning purposes. The other statements are false, because:

? Exploratory testing is not a test technique in which testers explore the requirements specification to detect non testable requirements, but rather a test technique in which testers explore the software system to detect functional and non-functional defects, as well as to learn new information, risks, or opportunities. Non testable requirements are requirements that are ambiguous, incomplete, inconsistent, or not verifiable, which can affect the quality and effectiveness of the testing process. Non testable requirements can be detected by applying static testing techniques, such as reviews or inspections, to the requirements specification, before the software system is developed or tested.

? Exploratory testing is not a test technique used by testers during informal code reviews to find defects by exploring the source code, but rather a test technique used by testers during dynamic testing to find defects by exploring the behavior and performance of the software system, without examining the source code. Informal code reviews are static testing techniques, in which the source code is analyzed by one or more reviewers, without following a formal process or using a checklist, to identify defects, violations, or improvements. Informal code reviews are usually performed by developers or peers, not by testers.

? In exploratory testing, testers usually do not produce scripted tests and establish bidirectional traceability between these tests and the items of the test basis, but rather produce unscripted tests and adapt them based on the feedback and the findings of the testing process. Scripted tests are tests that are designed and documented in advance, with predefined inputs, outputs, and expected results, and are executed according to a test plan or a test procedure.

Bidirectional traceability is the ability to trace both forward and backward the relationships between the items of the test basis, such as the requirements, the design, the risks, etc., and the test artifacts, such as the test cases, the test results, the defects, etc. Scripted tests and bidirectional traceability are usually associated with more formal and structured testing approaches, such as specification-based or structure-based test techniques, not with exploratory

testing. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.3,

Experience-based Test Design Techniques1

? ISTQB® Glossary of Testing Terms v4.0, Exploratory Testing, Session-based Testing, Session Sheet, Non Testable Requirement, Static Testing, Informal Review, Dynamic Testing, Scripted Testing, Bidirectional Traceability2

NEW QUESTION 145

Metrics can be collected during and at the end of testing activities to assess which of the following?

- A. Progress against the planned schedule and budget.i
- B. Current quality of the test objec
- C. H
- D. Adequacy of the test approach.i
- E. Effectiveness of the test activities with respect to the objectives.
- F. All the above.Select the correct Answer:
- G. Only i and ii.
- H. Only i and iii.
- I. Only I, ii and iv.
- J. Only v.

Answer: D

Explanation:

Metrics can be collected during and at the end of testing activities to assess various aspects including progress against the planned schedule and budget, the current quality of the test object, the adequacy of the test approach, and the effectiveness of the test activities with respect to the objectives. Collecting these metrics helps in understanding the overall performance and quality of the testing process.

NEW QUESTION 147

Which of the following applications will be the MOST suitable for testing by Use Cases

- A. Accuracy and usability of a new Navigation system compared with previous system
- B. A billing system used to calculate monthly charge based on large number of subscribers parameters
- C. The ability of an Anti virus package to detect and quarantine a new threat
- D. Suitability and performance of a Multi media (audio video based) system to a new operating system

Answer: A

Explanation:

A new navigation system compared with a previous system is the most suitable application for testing by use cases, because it involves a high level of interaction between the user and the system, and the expected behavior and outcomes of the system are based on the user's needs and goals. Use cases can help to specify the functional requirements of the new navigation system, such as the ability to enter a destination, select a route, follow the directions, receive alerts, etc. Use cases can also help to compare the accuracy and usability of the new system with the previous system, by defining the success and failure scenarios, the preconditions and postconditions, and the alternative flows of each use case. Use cases can also help to design and execute test cases that cover the main and exceptional paths of each use case, and to verify the satisfaction of the user's expectations.

The other options are not the most suitable applications for testing by use cases, because they do not involve a high level of interaction between the user and the system, or the expected behavior and outcomes of the system are not based on the user's needs and goals. A billing system used to calculate monthly charge based on a large number of subscriber parameters is more suitable for testing by data-driven testing, which is a technique for testing the functionality and performance of a system or component by using a large set of input and output data. The ability of an antivirus package to detect and quarantine a new threat is more suitable for testing by exploratory testing, which is a technique for testing the functionality and security of a system or component by using an informal and flexible approach, based on the tester's experience and intuition. The suitability and performance of a multimedia (audio video based) system to a new operating system is more suitable for testing by compatibility testing, which is a technique for testing the functionality and performance of a system or component by using different hardware, software, or network environments. References = CTFL 4.0 Syllabus, Section 3.1.1, page 28-29; Section 4.1.1, page 44-45; Section 4.2.1, page 47-48.

NEW QUESTION 150

Which of the following statements about the shift-left approach is true?

- A. Shift-left in testing can be implemented only in Agile/DevOps frameworks, as it relies completely on automated testing activities performed within a continuous integration process
- B. Performance testing performed during component testing, is a form of shift-left in testing that avoids planning and executing costly end-to-end testing at the system test level in a production-like environment
- C. Shift-left in testing can be implemented in several ways to find functional defects early in the lifecycle, but it cannot be relied upon to find defects associated with non-functional characteristics
- D. Continuous integration supports shift-left in testing as it can reduce the time between the introduction of a defect and its detection, thereby reducing the cost to fix it

Answer: D

Explanation:

This answer is correct because shift-left in testing is an approach that aims to perform testing activities as early as possible in the software development lifecycle, in order to find and fix defects faster and cheaper, and to improve the quality of the software product. Continuous integration is a practice that supports shift-left in testing, as it involves integrating and testing the software components frequently, usually several times a day, using automated tools and processes. Continuous integration can reduce the time between the introduction of a defect and its detection, thereby reducing the cost to fix it and the risk of accumulating defects that could affect the functionality or performance of the software product. References: ISTQB Foundation Level Syllabus v4.0, Section 3.1.1.3, Section 3.2.1.3

NEW QUESTION 153

For the same financial institution in Question 12, with the same requirements and expectations, what would be the most likely investment values used in testing if two-point boundary value analysis is used to design test cases specific to the 13% interest rate equivalence partition?

- A. R100 000, R100 001, R500 000, R500 001.
- B. R99 999, R100 000, R499 999, R500 000.
- C. R100 000. R500 000.
- D. R99 000, R500 001.

Answer: A

Explanation:

For boundary value analysis, the test cases should include the boundary values just inside and just outside the equivalence partition for the 13% interest rate range:

- ? R100,000 (just inside the previous range)
- ? R100,001 (start of the 13% range)
- ? R500,000 (end of the 13% range)
- ? R500,001 (just outside the range)

These values ensure that both the edges of the partition are tested.

NEW QUESTION 154

A system has a self-diagnostics module that starts executing after the system is reset. The diagnostics are running 12 different tests on the systems memory hardware. The following is one of the requirements set for the diagnostics module:

'The time taking the diagnostics tests to execute shall be less than 2 seconds' Which of the following is a failure related to the specified requirement?

- A. The diagnostic tests fail to start after a system reset
- B. The diagnostic tests take too much time to execute
- C. The diagnostic tests that measure the speed of the memory, fail
- D. The diagnostic tests fail due to incorrect implementation of the test code

Answer: B

Explanation:

A failure is an event in which a component or system does not perform a required function within specified limits¹. A requirement is a condition or capability needed by a user to solve a problem or achieve an objective². In this case, the requirement is that the diagnostics tests should execute in less than 2 seconds. Therefore, any event that violates this requirement is a failure. The only option that clearly violates this requirement is B. The diagnostic tests take too much time to execute. If the diagnostic tests take more than 2 seconds to complete, then they do not meet the specified limit and thus fail. The other options are not necessarily failures related to the specified requirement. Option A. The diagnostic tests fail to start after a system reset is a failure, but not related to the time limit. It is related to the functionality of the self-diagnostics module. Option C. The diagnostic tests that measure the speed of the memory, fail is also a failure, but not related to the time limit. It is related to the accuracy of the memory tests. Option D. The diagnostic tests fail due to incorrect implementation of the test code is also a failure, but not related to the time limit. It is related to the quality of the test code. References = ISTQB® Certified Tester Foundation Level Syllabus v4.0, Requirements Engineering Fundamentals.

NEW QUESTION 155

In which of the following test documents would you expect to find test exit criteria described?

- A. Test design specification
- B. Project plan
- C. Requirements specification
- D. Test plan

Answer: D

Explanation:

Test exit criteria are the conditions that must be fulfilled before concluding a particular testing phase. These criteria act as a checkpoint to assess whether we have achieved the testing objectives and are done with testing¹. Test exit criteria are typically defined in the test plan document, which is one of the outputs of the test planning phase. The test plan document describes the scope, approach, resources, and schedule of the testing activities. It also identifies the test items, the features to be tested, the testing tasks, the risks, and the test deliverables². According to the ISTQB® Certified Tester Foundation Level Syllabus v4.0, the test plan document should include the following information related to the test exit criteria³:

? The criteria for evaluating test completion, such as the percentage of test cases executed, the percentage of test coverage achieved, the number and severity of defects found and fixed, the quality and reliability of the software product, and the stakeholder satisfaction.

? The criteria for evaluating test process improvement, such as the adherence to the test strategy, the efficiency and effectiveness of the testing activities, the lessons learned and best practices identified, and the recommendations for future improvements.

Therefore, the test plan document is the most appropriate test document to find the test exit criteria described. The other options, such as test design specification, project plan, and requirements specification, are not directly related to the test exit criteria. The test design specification describes the test cases and test procedures for a specific test level or test type³. The project plan describes the overall objectives, scope, assumptions, risks, and deliverables of the software project⁴. The requirements specification describes the functional and non-functional requirements of the software product⁵. None of these documents specify the conditions for ending the testing process or evaluating the testing

outcomes. References = ISTQB® Certified Tester Foundation Level Syllabus v4.0, Entry and Exit Criteria in Software Testing | Baeldung on Computer Science, Entry And Exit

Criteria In Software Testing - Rishabh Software, Entry and Exit Criteria in Software Testing Life Cycle - STLC [2022 Updated] - Testsigma Blog, ISTQB® releases Certified Tester Foundation Level v4.0 (CTFL).

NEW QUESTION 160

Which of the following statements describes regression testing?

- A. Retesting of a fixed defectI
- B. Testing of an already tested programII
- C. Testing of new functionality in a programI
- D. Regression testing applies only to functional testingV Tests that do not have to be repeatable, because They are only used once
- E. II, IV, V
- F. I, III, IV
- G. II

H. I, IV

Answer: C

Explanation:

Regression testing is the re-running of functional and non-functional tests to ensure that previously developed and tested software still performs as expected after a change¹ It does not involve retesting of a fixed defect, testing of new functionality, or applying only to functional testing. Tests that are used for regression testing should be repeatable, because they are used to verify the stability of the software after each change² References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 4, Section 4.2.2, Page 291; ISTQB Glossary of Testing Terms v4.0, Page 292

NEW QUESTION 164

Which one of the following statements correctly describes the term 'debugging'?

- A. There is no difference between debugging and testing.
- B. Debugging is a confirmation activity that checks whether fixes resolved defects.
- C. Debugging is the development activity that finds, analyses and fixes defects.
- D. Debugging is of no relevance in Agile development.

Answer: C

Explanation:

Reference:ISTQB CTFL Syllabus V4.0, Section 1.1.2

NEW QUESTION 169

Which of the following statements about the shift-left approach is FALSE?

- A. The shift-left approach can only be implemented with test automation
- B. The shift-left approach in testing is compatible with DevOps practices.
- C. The shift-left approach can involve security vulnerabilities
- D. The shift-left approach can be supported by static analysis tools.

Answer: A

Explanation:

In a formal review process, the recorder's role is typically responsible for documenting the findings of the review team, including action items, decisions, and recommendations. This ensures that there is an accurate record of what was discussed and agreed upon, facilitating follow-up and continuous improvement. Therefore, statement C is correct as per the ISTQB CTFL syllabus.

NEW QUESTION 172

Which sequence of stated in the answer choices is correct in accordance with the following figure depicting the life-cycle of a defect?



- A. S0->S1->S2->S3->S5->S1
- B. S0->S1->S2->S3->S5->S1->S2->S3
- C. S0->S1->S2->S3->S4
- D. S0->S1 ->S2->S3->S5->S3->S4

Answer: D

Explanation:

According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0, the life cycle of a defect typically follows a sequence from its discovery to its closure. In the provided figure, it starts with S0 (New), moves to S1 (Assigned), then to S2 (Resolved), followed by S3 (Verified). If the defect is not fixed, it can be Re-opened (S5) and goes back for verification (S3). Once verified, it is Closed (S4). References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Section 1.4.3, Page 17.

NEW QUESTION 176

Which of the following statements is true?

- A. In Agile software development, work product documentation tends to be lightweight and manual tests tend to be often unscripted as they are often produced using experience- based test techniques
- B. Sequential development models impose the use of systematic test techniques and do not allow the use of experience-based test techniques
- C. In Agile software development, the first iterations are exclusively dedicated to testing activities, as testing will be used to drive development, which will be performed in the subsequent iterations
- D. Both in Agile software development and in sequential development models, such as the V-model, test levels tend to overlap since they do not usually have defined entry and exit criteria

Answer: A

Explanation:

This answer is correct because in Agile software development, work product documentation, such as user stories, acceptance criteria, or test cases, tends to be lightweight and concise, as the focus is on working software and frequent communication rather than comprehensive documentation. Manual tests tend to be often unscripted, as they are often produced using experience-based test techniques, such as error guessing or exploratory testing, which rely on the tester's skills, knowledge, and creativity to find defects and provide feedback. References: ISTQB Foundation Level Syllabus v4.0, Section 3.1.1.2, Section 3.2.1.2

NEW QUESTION 181

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your CTFL4 Exam with Our Prep Materials Via below:

<https://www.certleader.com/CTFL4-dumps.html>