

Exam Questions ISTQB-CTFL

ISTQB-Foundation Level Exam

<https://www.2passeasy.com/dumps/ISTQB-CTFL/>



NEW QUESTION 1

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 2

A software system checks age in order to determine which welcome screen to display. Age groups are:

Group I: 0-12

Group II: 13-18 Group III: over 18

Which of the below represent boundary values?

- A. (-1,0,12,13,18,19)
- B. (-1,0,11,12,13,14,18,19)
- C. (0,12,13,18,19)
- D. (4,5,15,20)

Answer: A

Explanation:

A correct list of boundary values for the age input should include the minimum and maximum values of each age group (0, 12, 13, 18), as well as the values just below and above each boundary (-1, 19). Boundary value analysis is a test design technique that involves testing the values at or near the boundaries of an input domain or output range, as these values are more likely to cause errors than values in the middle. Option A satisfies this condition, as it has all six boundary values (-1, 0, 12, 13, 18, 19). Option B has two values from the same equivalence class (12 and 13), option C has only four boundary values (0, 12, 18, 19), and option D has no boundary values at all. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 34.

NEW QUESTION 3

Which ONE of the following statements does NOT describe how testing contributes to higher quality?

- A. Properly designed tests that pass reduce the level of risk in a system.
- B. The testing of software demonstrates the absence of defects.
- C. Software testing identifies defects, which can be used to improve development activities.
- D. Performing a review of the requirement specifications before implementing the system can enhance quality.

Answer: B

Explanation:

? The testing of software does not demonstrate the absence of defects, but rather the presence of defects or the conformance of the software to the specified requirements¹. Testing can never prove that the software is defect-free, as it is impossible to test all possible scenarios, inputs, outputs, and behaviors of the software². Testing can only provide a level of confidence in the quality of the software, based on the coverage, effectiveness, and efficiency of the testing activities³.

? The other options are correct because: References =

? 1 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 10

? 2 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 11

? 3 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 12

? 4 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 13

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

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

? 7 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 14

? [8] ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 15

? [9] ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 16

? [10] ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 17

? [11] ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 18

? [12] ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 19

NEW QUESTION 4

Which of the following is a function of a dynamic analysis tool?

- A. Provide support for traceability of tests, test results and incidents to source documents
- B. Monitor the allocation, use and de-allocation of memory during run-time of a program
- C. Execute programs step-by-step in order to reproduce failures and find corresponding defects
- D. Provide support for release of baselines consisting of configuration items

Answer: B

Explanation:

A dynamic analysis tool is a tool that performs analysis of a software product based on its behavior during execution. A dynamic analysis tool can monitor various aspects of a program's run-time performance, such as memory usage, CPU load, response time, or resource leaks. A dynamic analysis tool can monitor the allocation, use and de- allocation of memory during run-time of a program, which can help detect defects such as memory leaks, buffer overflows, or memory corruption. A dynamic analysis tool cannot provide support for traceability of tests, test results and incidents to source documents, as this is a function of a test management tool. A dynamic analysis tool cannot execute programs step-by-step in order to reproduce failures and find corresponding defects, as this is a function of a debugging tool. A dynamic analysis tool cannot provide support for release of baselines consisting of configuration items, as this is a function of a configuration management tool. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 6, page 56-57.

NEW QUESTION 5

Given the following review types and review characteristics:

- * a. Pair review
- * b. Walkthrough
- * c. Technical review
- * d. Inspection
- * 1. Formal
- * 2. Informal
- * 3. Purposes include evaluating the quality of the work product under review and generating new ideas (e.g., brainstorming solutions)
- * 4. Purposes include Improving the software product and training the review participants Which of the following BEST matches the review type with the review characteristic?

- A. a-1. b-4, c-3. d-2
- B. a-4, b-3. c-2. d-1
- C. a-2, b-3, c-4, d-1
- D. a-2, b-4, c-3. d-1

Answer: C

Explanation:

Pair reviews are informal and typically involve two people reviewing the work product together, often in an informal setting. Walkthroughs are more formal and aim to educate stakeholders and evaluate the product, serving the dual purpose of improving the product and training participants. Technical reviews have a strong focus on improving the product's quality, often involving technical stakeholders. Inspections are the most formal review type and are aimed primarily at detecting defects. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 3.2.3 "Review Types".

NEW QUESTION 6

Which of the following statements about Experience Based Techniques (EBT) is correct?

- A. EBT use tests derived from the test engineers' previous experience with similar technologies.
- B. EBT is based on the ability of the test engineer to implement various testing techniques.
- C. EBT is done as a second stage of testing, after non-experienced-based testing took place.
- D. EBT require broad and deep knowledge in testing but not necessarily in the application or technological domain.

Answer: A

Explanation:

Experience based techniques (EBT) are techniques that use the knowledge, intuition and skills of the test engineers to design and execute tests. EBT use tests derived from the test engineers' previous experience with similar technologies, domains, applications or systems. EBT are not based on the ability of the test engineer to implement various testing techniques, but rather on their personal judgment and creativity. EBT are not done as a second stage of testing, after non-experience-based testing took place, but rather as a complementary or alternative approach to other techniques. EBT require broad and deep knowledge in both testing and the application or technological domain, as this can help the test engineer identify potential risks, scenarios or defects. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 5, page 48-49.

NEW QUESTION 7

A test engineer finds a defect while testing. After the developer has fixed the defect, the test engineer decides to re-run a complete section of the tests. Which of the following is correct?

- A. The test engineer should not re-run the tests, as they have already been run, and results recorded.
- B. The test engineer should not re-run the tests, they should be part of the developer tests.
- C. The test engineer should re-run the tests, in order to ensure that new defects have not been introduced by the fix.
- D. The test engineer should re-run the tests, because the defect shows that the test cases need to be updated.

Answer: C

Explanation:

The test engineer should re-run the tests, in order to ensure that new defects have not been introduced by the fix. This is also known as regression testing, which is a type of testing that verifies that previously tested software still performs correctly after a change. Regression testing helps to detect any side effects or unintended consequences of a fix or a modification. The other options are incorrect reasons for re-running the tests. The test engineer should not re-run the tests, as they have already been run, and results recorded, because this ignores the possibility of new defects caused by the fix. The test engineer should not re-run the tests, they should be part of the developer tests, because this assumes that developer tests are sufficient and reliable, which may not be true. The test engineer should not re-run the tests, because the defect shows that the test cases need to be updated, because this does not address the impact of the fix on other test cases or functionalities. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 41.

NEW QUESTION 8

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 9

The following requirement is given "Set X to be the sum of Y and Z". All the following four implementations have bugs. Which one of the following bugs can be caught by Static Analysis?

- A. int x = 1. int y = 2. int y = 3.X = y=z;
- B. int x = 1. int y = 2. int z = 3.X = z-y
- C. int x = 1. Int y = 2. Int z = 3.Z = x +y
- D. int y = 2 Int z = 3. Y = z+y

Answer: A

Explanation:

Static analysis is a technique that analyzes the source code or other software artifacts without executing them. Static analysis can detect defects such as syntax errors, coding standards violations, potential security vulnerabilities, or logical flaws. Static analysis can catch the bug in the first implementation, as it contains two syntax errors: the variable y is declared twice, and the assignment statement X = y=z is invalid. Static analysis cannot catch the bugs in the other three implementations, as they are logical errors that do not violate any syntax rules, but produce incorrect results. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 3, page 25-26.

NEW QUESTION 10

Which of the following software development models BEST exemplifies a model that does NOT support the principle of early testing?

- A. The iterative development model
- B. The V-model
- C. The Waterfall model
- D. The incremental development model

Answer: C

Explanation:

The Waterfall model exemplifies a software development model that does not support the principle of early testing. In the Waterfall model, each phase must be completed before the next begins, which delays testing until after the completion of the earlier phases like requirements gathering and design. This can often result in finding defects later in the development cycle, making them more expensive and time-consuming to fix (ISTQB not-for-profit association) (ISTQB not-for-profit association).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf

? ISTQB News Release on CTFL v4.0: <https://www.istqb.org/news/posts/istqb-releases-certified-tester-foundation-level-v40-ctfl/>

NEW QUESTION 10

Which of the following definitions is NOT true?

- A. Test data preparation tools fill databases, create files or data transmissions to set up test data to be used during the execution of tests.
- B. Test execution tools execute test objects using automated test scripts.
- C. Test Management tools monitor and report on how a system behaves during the testing activities.
- D. Test comparators determine differences between files, databases or test results.

Answer: C

Explanation:

Test Management tools are designed to support the planning, execution, and monitoring of the testing process. They provide features for managing test cases, test runs, tracking defects, and reporting on testing activities. However, the statement in option C describes Test Management tools as monitoring and reporting on the system's behavior during testing activities, which is not accurate. Test Management tools focus on the testing process itself rather than on the behavior of the system under test.

? Test data preparation tools (A) indeed create and manage test data for use during test execution.

? Test execution tools (B) automate the execution of test cases and the comparison of actual outcomes against expected results.

? Test comparators (D) are tools that compare actual outcomes with expected outcomes, highlighting discrepancies.

Therefore, option C is the correct answer as it inaccurately describes the function of Test Management tools.

NEW QUESTION 14

Which of the following project scenario gives the BEST example where maintenance testing should be triggered?

- A. Completion of architecture of the bank system

- B. Release of the early draft of the low level project design of an IoT application
- C. Defect was found in a pre-released version of the customer service application
- D. Delivery of the hot fix to mobile operating system and ensuring that it still works

Answer: D

Explanation:

Maintenance testing is triggered by changes such as bug fixes, enhancements, or environmental changes.

Option A: "Completion of architecture of the bank system" is not a typical scenario for maintenance testing, as it describes a design phase rather than an operational change. Option B: "Release of the early draft of the low level project design of an IoT application" is again not suitable for maintenance testing, as it refers to the design phase.

Option C: "Defect was found in a pre-released version of the customer service application" is closer but not quite accurate, as maintenance testing focuses on changes mad (ISTQB not-for-profit association)system is released.

Option D: "Delivery of the hot fix to mobile operating system and ensuring that it still works" is the best example as it directly involves testing after a fix has been implemented. Therefore, the correct answer is D6†source.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 17

Which of the following options cover the test types performed during typical system testing phase:

- A. UsabilityII Requirements based scenariosIII Testing parts of the code in isolationIV Correct order of parameters in API calls
- B. I, III
- C. II
- D. I
- E. IV
- F. II
- G. IV

Answer: B

Explanation:

System testing is a level of testing performed to evaluate the behavior and quality of a whole software product or system. System testing can include various types of testing, such as:

? I) Usability testing: A type of testing that evaluates how easy, efficient and satisfying it is to use the software product or system from the user's perspective.

? II) Requirements based scenarios testing: A type of testing that verifies that the software product or system meets its specified requirements or user stories by executing realistic scenarios or workflows. System testing does not include the following types of testing, as they are more suitable for lower levels of testing, such as unit testing or integration testing:

? III) Testing parts of the code in isolation: A type of testing that verifies the functionality and quality of individual software components or units by isolating them from other components or units.

? IV) Correct order of parameters in API calls: A type of testing that verifies the functionality and quality of software components or units that communicate with each other through application programming interfaces (APIs) by checking the correct order and format of parameters in API calls. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 20-21; Chapter 4, page 34-35.

NEW QUESTION 20

Which of the following can be considered a VALID exit criterion? I Estimates of defect density or reliability measures.

II. The completion and publication of an exhaustive Test Report.

III. Accuracy measures, such as code, functionality or risk coverage. IV Residual risks such as lack of code coverage in certain areas.

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

Answer: A

Explanation:

An exit criterion is a condition that defines when a test activity has been completed or when a test phase can be concluded. An exit criterion can be based on various factors, such as:

? I) Estimates of defect density or reliability measures. These are quantitative measures that indicate how many defects are present in the software product or how likely it is to fail under certain conditions. These can be used as exit criteria to ensure that the software product meets a certain level of quality or performance before moving to the next phase or releasing it to the customer.

? III) Accuracy measures, such as code coverage, functionality coverage or risk coverage. These are quantitative measures that indicate how much of the software product has been tested in terms of its code, functionality or risk. These can be used as exit criteria to ensure that the test suite is adequate or complete before moving to the next phase or releasing it to the customer.

? IV) Residual risks, such as lack of code coverage in certain areas, unresolved defects or unknown factors. These are qualitative measures that indicate the remaining risks or uncertainties associated with the software product after testing. These can be used as exit criteria to ensure that the residual risks are acceptable or manageable before moving to the next phase or releasing it to the customer. The following factor is not a valid exit criterion:

? II) The completion and publication of an exhaustive Test Report. This is not a valid exit criterion, as it does not reflect the quality or completeness of the testing process or product. A test report is a document that summarizes the results and outcomes of a test activity or phase. A test report can be used as an input for deciding whether to exit a test activity or phase, but it is not a condition that defines when to exit. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 2, page 13; Chapter 6, page 58-59.

NEW QUESTION 21

In maintenance testing, what is the relationship between impact analysis and regression testing?

- A. Impact analysis requires a regression testing for only the tests that have detected faults in previous SW release
- B. There is no relationship between impact analysis and regression testing.
- C. Impact analysis requires a regression testing for all program elements which were newly integrated (new functionalities).

D. The impact analysis is used to evaluate the amount of regression testing to be performed.

Answer: D

Explanation:

In maintenance testing, the relationship between impact analysis and regression testing is that the impact analysis is used to evaluate the amount of regression testing to be performed. Maintenance testing is a type of testing that is performed on an existing software product after it has been delivered or deployed, in order to ensure that it still meets its requirements and functions correctly after a change or a modification. Maintenance testing can be triggered by various reasons, such as corrective maintenance (fixing defects), adaptive maintenance (adapting to new environments), perfective maintenance (improving performance), preventive maintenance (avoiding future problems), etc. Impact analysis is a technique that is used to assess the extent and nature of changes introduced by maintenance activities on the software product or project. Impact analysis helps to identify which parts of the software product are affected by the changes, which parts need to be modified or updated accordingly, which parts need to be retested or verified for correctness or compatibility, etc. Regression testing is a type of testing that verifies that previously tested software still performs correctly after a change or a modification. Regression testing helps to detect any side effects or unintended consequences of maintenance activities on the software product's functionality or quality. Regression testing can be performed at various levels and scopes depending on the impact analysis results. Therefore, in maintenance testing, impact analysis is used to evaluate the amount of regression testing to be performed. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 20.

NEW QUESTION 22

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 testing. 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 deliverables. 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 criteria:

? 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 type. The project plan describes the overall objectives, scope, assumptions, risks, and deliverables of the software project. The requirements specification describes the functional and non-functional requirements of the software product. 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 25

Which of the following statements about test reports are TRUE?

- II. Test reports shall give stakeholders information as basis for decisions.
- III. Test reports shall summarize what happened through a period of testing.
- IV. Test reports shall be approved by the development team, the test team and the customer

- A. Test reports shall include information about remaining risks.
- B. II, III, V
- C. I, II, IV
- D. I, III, v
- E. II, III, IV

Answer: A

Explanation:

Statements II, III and V are true about test reports. Test reports are documents that provide information on the results and status of testing activities for a given period or phase. Test reports should give stakeholders information as basis for decisions, such as whether to release the software product, whether to continue testing, whether to change the scope or priorities of testing, etc. Test reports should summarize what happened through a period of testing, such as what test cases were executed, what defects were found, what risks were identified, what issues were encountered, what achievements were made, etc. Test reports should include information about remaining risks, such as what defects are still open, what test cases are still pending, what functionalities are still untested, what uncertainties are still unresolved, etc. Statements I and IV are not true about test reports. Test reports do not need to be approved by the test team, the development team, or the customer, unless it is specified by the test policy or the test plan. Test reports only need to be reviewed and verified by the test leader or the test manager before being distributed to the intended recipients. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 141.

NEW QUESTION 27

- I. When closing the test activities, all the testware resources can be uninstalled and released
- II. All the testware should be subject to Configuration Management
- III. The testware, at the end of the project, should be transferred to the organization responsible for maintenance
- IV. The developers are responsible for the correct installation of the testware

- A. II, III

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

Answer: A

Explanation:

Testware is a term that refers to all artifacts produced during the testing process, such as test plans, test cases, test scripts, test data, test results, defect reports, etc. The following statements about testware are correct:

? II) All the testware should be subject to Configuration Management. Configuration management is a process that establishes and maintains consistency among work products throughout their life cycle. Configuration management applies to all testware, as it helps ensure their quality and consistency, track their changes and defects, control their versions and access rights, and link them to other artifacts.

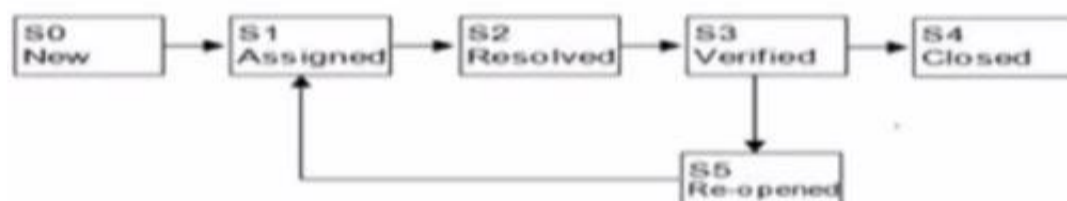
? III) The testware at the end of the project should be transferred to the organization responsible for maintenance. Maintenance testing is testing performed on a software product after delivery to correct defects or improve performance or other attributes. Maintenance testing requires testware from previous testing activities or phases, such as test cases, test data, test results, etc. Therefore, the testware at the end of the project should be transferred to the organization responsible for maintenance testing, such as support team or maintenance team. The following statements about testware are incorrect:

? I) When closing the test activities, all the testware resources can be uninstalled and released. This statement is incorrect, as some testware resources may still be needed for future testing activities or phases, such as maintenance testing or regression testing. Therefore, when closing the test activities, some testware resources should be archived and stored for future use, while others can be uninstalled and released.

? IV) The developers are responsible for the correct installation of the testware. This statement is incorrect, as the testers are responsible for the correct installation of the testware. The testers should ensure that they have access to all necessary testware resources and that they are installed and configured properly before starting the test execution. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 6, page 58-61.

NEW QUESTION 32

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 33

The following open incident report provided: Date: 01.01.01

Description: When pressing the stop button the application status remain in "Attention" instead of "Ready".

Severity: High

Life Cycle: Integration

Which of the following details are missing in the giving incident report?

- A. Identification or configuration of the applicationI
- B. The name of the developerII
- C. Recommendation of the developerIV The actions and/or conditions that came before the pressing of the button
- D. IV
- E. IV
- F. II
- G. II, III

Answer: B

Explanation:

In an incident report, essential details provide context and facilitate the investigation and resolution of the incident. The missing elements in the given incident report are:

I. Identification or configuration of the application: This detail is crucial as it specifies which version or configuration of the application is affected, helping in reproducing the issue. IV. The actions and/or conditions that came before pressing the button: Understanding the sequence of actions leading to the issue is vital for replicating and diagnosing the problem. The name of the developer (II) and the recommendation of the developer (III) are not typically included in an incident report as they do not contribute to identifying or resolving the incident. The focus is on the incident's details, reproduction steps, and the system's state rather than on personnel or proposed solutions at this stage. Therefore, option B, which includes both I and IV, is the correct answer.

NEW QUESTION 36

The following 4 equivalence classes are given:

$x \leq -100$
 $-100 < x < 100$
 $100 \leq x < 1000$
 $x \geq 1000$

Which of the following alternatives includes correct test values for x. based on equivalence partitioning?

- A. -100; 100;1000; 1001
- B. -500; 0; 100; 1000
- C. -99; 99;101; 1001
- D. -1000; -100; 100; 1000

Answer: D

Explanation:

? The question is about selecting the correct test values for x based on equivalence partitioning. Equivalence partitioning is a software test design technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived. In this case, the given equivalence classes are:

Option D provides a value from each of these partitions:

? For $(x \leq -100)$, it gives -1000.

? For $(-100 < x < 100)$, it gives -100 and 100.

? For $(100 \leq x < 1000)$, it gives 500.

? For $(x \geq 1000)$, it gives 1500.

So, option D covers all four given equivalence classes with appropriate values. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 documents available at ISTQB and ASTQB.

? 1: ISTQB Foundation Level Syllabus 2018, Version 4.0, p. 38

? 2: ISTQB Foundation Level Syllabus 2018, Version 4.0, p. 39

? : ISTQB Foundation Level Syllabus 2018, Version 4.0, p. 40

NEW QUESTION 41

The testers in company A were part of the development team. Due to an organizational change they moved to be part of the support team.

What are the advantages and the disadvantages of this change?

- A. Advantage: More independence in deciding what and how to test, Disadvantage: Isolation from me development team knowledge
- B. Advantage: being closer to customer perspective, Disadvantage less independence in perspectives
- C. Advantage: pulled to support tasks and having less time for testing, Disadvantage less chances to move a tester to development
- D. Advantage: increased chances to move a tester to development; Disadvantage: pulled to support tasks and having less time for testing

Answer: B

Explanation:

Being part of the support team means that the testers are closer to the customer perspective, which is an advantage for testing, as they can better understand the user needs and expectations, and identify more realistic scenarios and risks. However, being part of the support team also means that they have less independence in deciding what and how to test, as they may be influenced by the customer's preferences or requests, which could compromise the objectivity and effectiveness of testing. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 6.

NEW QUESTION 45

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 48

Which statement about use case testing is true?

- A. The test cases are designed to find defects in the data flow.
- B. The test cases are designed to be used by real users, not by professional testers
- C. The test cases are always designed by customers or end users.
- D. The test cases are designed to find defects in the process flow.

Answer: D

Explanation:

Use case testing is a technique that helps identify test cases that exercise the whole system on a transaction by transaction basis from start to finish. Use cases are descriptions of how users interact with the system to achieve a specific goal. Use case testing is not focused on data flow, but rather on process flow. Use case testing can be performed by professional testers, customers or end users, depending on the context. Use case testing does not require the test cases to be designed by customers or end users, but rather by anyone who has access to the use case specifications. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 4, page 36.

NEW QUESTION 53

Which of the types of test tools noted below BEST describes tools that support reviews?

- A. Tools to assess data quality
- B. Tools to support usability testing
- C. Tools to support specialized testing needs
- D. Tools to support static testing

Answer: D

Explanation:

Static testing refers to testing that doesn't involve executing code. It includes activities like reviews, inspections, and static analysis. Tools that support static testing help with activities such as analyzing source code, checking coding standards, and aiding in document reviews. These tools can automate or facilitate various aspects of static testing processes, such as highlighting potential issues in code or documents without executing the software.

References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 6.1.1.

NEW QUESTION 58

Which of the following is a key characteristic of informal reviews?

- A. Kick-off meeting
- B. Low cost
- C. Individual preparation
- D. Metrics analysis

Answer: B

Explanation:

A key characteristic of informal reviews is low cost. Informal reviews are a type of review that does not follow a formal process or have any formal documentation. Informal reviews are usually performed by individuals or small groups of peers or colleagues who have some knowledge or interest in the product under review. Informal reviews can be done at any time and for any purpose, such as checking for errors, clarifying doubts, sharing ideas, etc. Informal reviews have low cost, as they do not require much time, effort, or resources to conduct. The other options are not key characteristics of informal reviews. Kick-off meeting is a characteristic of formal reviews, such as inspections or walkthroughs. Kick-off meeting is a meeting that is held before the review process starts, where the roles and responsibilities of the participants are defined, the objectives and scope of the review are agreed, and the logistics and schedule of the review are planned. Individual preparation is a characteristic of formal reviews, such as inspections or walkthroughs. Individual preparation is an activity that is performed by the reviewers before the review meeting, where they examine the product under review and identify any issues or questions that need to be discussed or resolved during the review meeting. Metrics analysis is a characteristic of formal reviews, such as inspections or walkthroughs. Metrics analysis is an activity that is performed after the review process is completed, where the data and results of the review are collected and analyzed to measure the effectiveness and efficiency of the review, as well as to identify any improvement actions or lessons learned for future reviews. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 9.

NEW QUESTION 59

A system computes prices for bus tickets. The price depends on

- the passenger type (baby, child, adult, senior citizen, student, military)
- the travelling type (as single or in a group)
- the distance (zone 1. 2. 3)
- the kind of transport (ordinary, express)

Which of the following test techniques is the most appropriate one for testing the price computation?

- A. Statement coverage
- B. State transition testing
- C. Equivalence partitioning
- D. Use case testing

Answer: C

Explanation:

Equivalence partitioning is a technique that divides the input data and output results of a software component into partitions of equivalent data. Each partition should contain data that is treated in the same way by the component. Equivalence partitioning can be used to reduce the number of test cases by selecting one representative value from each partition. Equivalence partitioning is suitable for testing the price computation, as it can identify different partitions based on the passenger type, the travelling type, the distance and the kind of transport. Equivalence partitioning is not statement coverage, which is a technique that measures how many executable statements in a source code are executed by a test suite. Statement coverage is not appropriate for testing the price computation, as it does not consider the input data or output results. Equivalence partitioning is not state transition testing, which is a technique that models how a system transitions from one state to another depending on events or conditions. State transition testing is not relevant for testing the price computation, as it does not involve any states or transitions. Equivalence partitioning is not use case testing, which is a technique that tests how users interact with a system to achieve a specific goal. Use case testing is not applicable for testing the price computation, as it does not focus on a single function or component. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 37-38.

NEW QUESTION 62

Which of the following is NOT a deciding factor in determining the extent of testing required?

- A. Level of risk of the product or features
- B. Budget to do testing
- C. A particular tester involved in testing
- D. Time available to do testing

Answer: C

Explanation:

The extent of testing required for a software product depends on various factors, such as the level of risk, the budget, and the time available. The level of risk reflects the potential impact of failures on the stakeholders and the environment. The budget determines how much resources can be allocated for testing. The time available defines the schedule and deadlines for testing activities. The particular tester involved in testing is not a deciding factor for the extent of testing required, as testing should be based on objective criteria and not on personal preferences or abilities. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 14-15.

NEW QUESTION 66

Which of the following BEST describes a test summary report for executive-level employees

- A. The report is detailed and includes a status summary of defects by priority or budget
- B. The report is detailed and includes specific information on defects and trends
- C. The report is high-level and includes a status summary of defects by priority or budget
- D. The report is high-level and includes specific information on defects and trends

Answer: C

Explanation:

For executive-level employees, a test summary report should be concise and focus on high-level information. It typically includes a summary of defects categorized by priority or budget. Executives are generally interested in the overall status and the impact on critical business objectives rather than detailed technical information. The report should provide an overview of the most important aspects of testing, such as key issues, test progress, and any risks or concerns that could affect project outcomes. References:
? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 5.3.1.

NEW QUESTION 69

Which of the following exemplifies how a software bug can cause harm to a company?

- A. "Print" prints the last page twice for a file with 1000 pages
- B. The timeout on the login page of a web site is 9 minutes, while the requirement was for 10 minutes
- C. When uninstalling the application, the uninstall dialog has a spelling mistake
- D. When calculating the final price in a shopping list, the price of the last item is not added

Answer: D

Explanation:

A software bug can cause harm to a company by directly affecting its operations, reputation, user satisfaction, and financials. Option D, "When calculating the final price in a shopping list, the price of the last item is not added," describes a defect that directly impacts the core functionality of a financial transaction, potentially leading to financial loss and customer dissatisfaction. This can have severe implications for the company's credibility and revenue. Options A, B, and C describe bugs that, while potentially annoying, do not have the same direct impact on the company's core operations and financial integrity as option D.

NEW QUESTION 72

Which of the following BEST matches the descriptions with the different categories of test techniques?

- * 1. Test cases are based on the test basis which may include the requirements, use cases and user stories
 - * 2. Test cases are based on the test basis which may include the software architecture or code
 - * 3. Test cases can show deviations from the requirements
 - * 4. These test techniques are applicable to both functional and non-functional testing
 - * 5. Tests are based on knowledge of developers, users and other stakeholders
- Black - Black-box test techniques
White * White-box test techniques
Experience - Experience-based test techniques

- A. Black - 1,3,4 White - 2 Experience - 5
- B. Black - 2, 3 White - 1 Experience - 4, 5
- C. Black - 1,2 White - 3, 4 Experience - 5
- D. Black - 2, 3 White - 1,5 Experience - 4

Answer: A

Explanation:

Matching the descriptions with the test techniques:

- ? Black-box test techniques use the external description of the software, including requirements, use cases, and user stories.
- ? White-box test techniques use the internal structure of the software system, including software architecture and code.
- ? Black-box test techniques can reveal deviations from the requirements as they validate the external behavior of the software.
- ? Both black-box and white-box test techniques are applicable to functional and non-functional testing.
- ? Experience-based test techniques rely on the knowledge and intuition of developers, users, and other stakeholders (ISTQB Main Web). References:
? ISTQB® Certified Tester Foundation Level Syllabus v4.0: ISTQB CTFL Syllabus v4.0 PDF

NEW QUESTION 75

Decision table testing is being performed on transactions in a bank's ATM (Automated Teller Machine) system. Two test cases have already been generated for rules 1 and 4. which are shown below:

SEE ATTACHMENT 1

Given the following additional test cases: SEE ATTACHMENT 2

Which two of the additional test cases would achieve full coverage of the full decision table (when combined with the test cases that have already been generated for rules 1 and 4)?

- A. DT1, DT4
- B. DT3, DT4
- C. DT2, DT3
- D. DT1.DT2

Answer: C

Explanation:

Decision table testing is used to analyze combinations of inputs to determine the appropriate outputs, often based on specific rules or conditions.

For the problem statement:

? Rule 1: (Withdrawal = Allowed, Balance = Sufficient, Fast Cash = True, Correct PIN = True)

? Rule 4: (Withdrawal = Allowed, Balance = Sufficient, Fast Cash = True, Correct PIN = False)

The additional test cases are:

? DT1: (Withdrawal = Allowed, Balance = Insufficient, Fast Cash = True, Correct PIN = True)

? DT2: (Withdrawal = Allowed, Balance = Sufficient, Fast Cash = False, Correct PIN = True)

? DT3: (Withdrawal = Allowed, Balance = Insufficient, Fast Cash = True, Correct PIN = False)

? DT4: (Withdrawal = Allowed, Balance = Sufficient, Fast Cash = False, Correct PIN = False)

From the given test cases, DT2 covers the scenario where Fast Cash is False, which is not covered in the initial cases. DT3 covers the case where Balance is Insufficient and PIN is incorrect.

Combining Rules 1 and 4 with DT2 and DT3 covers all the scenarios. References:

? Certified Tester Foundation Level v4.0

? 10 Sample Exams ISTQB Foundation Level (CTFL) v4.0

NEW QUESTION 76

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 79

In which of the following cases you would NOT execute maintenance testing?

- A. Retirement of the software or system
- B. Modifications to a released software or system
- C. Migration of the system data to a replacement system
- D. Update to the Maintainability requirements during the development phase

Answer: D

Explanation:

Maintenance testing is testing performed on a software product after delivery to correct defects or improve performance or other attributes. Maintenance testing can be triggered by various situations, such as modifications to a released software or system, migration of the system data to a replacement system, or retirement of the software or system. Maintenance testing is not executed when there is an update to the maintainability requirements during the development phase, as this is not a maintenance situation but rather a change request that should be handled by the development process. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 18-19.

NEW QUESTION 81

The following incident report that was generated during test of a web application. What would you suggest as the most important report improvement?

Defect detected date: 15 8.2010 Defect detected by: Joe Smith Test level System test

Test case: Area 5/TC 98 Build version: 2011-16.2

Defect description After having filled out all required fields in screen 1, t click ENTER to continue to screen 2 Nothing happens, no system response at all.

- A. Add information about which web browser was used
- B. Add information about which developer should fix the bug
- C. Add the time stamp when the incident happened
- D. Add an impact analysis

Answer: A

Explanation:

The most important report improvement for the given incident report would be to add information about which web browser was used when the defect was detected. This information is relevant for reproducing and debugging the defect, as different web browsers may have different behaviors or compatibility issues with the web application. The other options are less important or irrelevant for the incident report. The developer who should fix the bug can be assigned by the project manager or the defect tracking system, not by the tester who reports the defect. The time stamp when the incident happened is not very useful, as it does not indicate the cause or the frequency of the defect. The impact analysis is not part of the incident report, but rather of the risk assessment or prioritization process. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 140.

NEW QUESTION 83

Which of the following provides the BEST description of statement coverage?

- A. A white-box test technique which covers the decision results which determine the next statement to be executed
- B. A black-box test technique which uses a state table to derive test cases
- C. A white-box test technique which focuses on the percentage of executable statements that has been executed by a test suite
- D. An experience-based test technique in which test cases are based on the tester's knowledge of past failures

Answer: C

Explanation:

Statement coverage is a white-box test technique which focuses on executing all possible statements in the code at least once during testing. This helps in determining the percentage of executable statements that have been executed by the test suite, aiming to ensure that all parts of the program have been tested at least once (ISTQB Main Web).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: ISTQB CTFL Syllabus v4.0 PDF

NEW QUESTION 85

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 90

Which of the following CORRECTLY matches the roles and responsibilities in a formal review?

- A. Facilitator - Fixes defects in the work product under review
- B. Scribe - Collates potential defects found during the individual review activity
- C. Review Leader - Creates the work product under review
- D. Author - Identify potential defects in the work product under review

Answer: B

Explanation:

In formal reviews, the scribe's role is to collate potential defects and other findings during the review process. This position is crucial as it ensures all observations and defects are recorded accurately, facilitating efficient analysis and resolution of issues identified during the review. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 3.2.4 "Roles and Responsibilities in a Formal Review".

NEW QUESTION 93

Which of the following is a possible reason for introducing a defect in software code?

- A. Rushing to meet a tight deadline to turn code over for testing
- B. Improper unit testing
- C. Improper system testing
- D. Focus on static testing over dynamic testing

Answer: A

Explanation:

The ISTQB CTFL syllabus identifies several causes for defects in software. One prominent reason, as highlighted in the curriculum, is the pressure and rush to meet tight deadlines, which can lead to insufficiently reviewed or tested code being moved into further stages of testing or production. This scenario describes rushing to meet a deadline as a potential cause for defects because it may compromise the thoroughness of code development and testing. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.4.1 "Why is Testing Necessary?".

NEW QUESTION 97

For the following pseudo-code determine number of tests required for 100% statement coverage

```
IF Gender = Boy
If Age > 3 AND Age < 5 Shoe Size = 1
ELSE IF Age >= 5 AND Age < 7
Shoe Size = 2 ENDIF
ELSE
IF Age > 3 AND Age < 5
Shoe Size = 0
ELSE IF Age >= 5 AND Age < 7
Shoe Size = 1 ENDIF ENDIF
```

- A. 6
- B. 4
- C. 2
- D. 6

Answer: B

Explanation:

To achieve 100% statement coverage, we need to design test cases that ensure every statement in the given pseudo-code is executed at least once. Analyzing the pseudo-code, we notice that there are conditions based on two variables: Gender and Age. To cover all statements, we need to consider the paths that lead to each assignment of the Shoe Size variable.

- ? Gender = Boy, Age <= 3 (Shoe Size assignment is not reached, but the condition is evaluated)
- ? Gender = Boy, Age > 3 AND Age < 5 (Shoe Size = 1)
- ? Gender = Boy, Age >= 5 AND Age < 7 (Shoe Size = 2)
- ? Gender != Boy, Age <= 3 (Again, Shoe Size assignment is not reached, but the condition is evaluated)
- ? Gender != Boy, Age > 3 AND Age < 5 (Shoe Size = 0)
- ? Gender != Boy, Age >= 5 AND Age < 7 (Shoe Size = 1)

However, upon closer inspection, we see that tests 1 and 4 do not contribute to statement coverage as they do not lead to a Shoe Size assignment. Therefore, we only need 4 test cases to achieve 100% statement coverage, making option B the correct answer.

NEW QUESTION 98

Which of the following statements about decision tables are TRUE?

- A. Generally, decision tables are generated for low risk test items. I
- B. Test cases derived from decision tables can be used for component tests. II
- C. Several test cases can be selected for each column of the decision table. I
- D. The conditions in the decision table represent negative tests generally.
- E. III
- F. I, IV
- G. I
- H. IV
- I. I
- J. III

Answer: D

Explanation:

A decision table is a technique that shows combinations of inputs and/or stimuli (causes) with their associated outputs and/or actions (effects). A decision table consists of four quadrants: conditions (inputs), actions (outputs), condition entries (values) and action entries (results). The following statements about decision tables are true:

- ? II. Test cases derived from decision tables can be used for component tests.

Decision tables can be used to test components that have multiple inputs and outputs that depend on logical combinations of conditions. Decision tables can help cover all possible combinations or scenarios in a systematic way.

- ? III. Several test cases can be selected for each column of the decision table. A column of a decision table represents a unique combination of condition entries and action entries. Several test cases can be selected for each column by varying other input values or expected results that are not part of the decision table. The following statements about decision tables are false:

- ? I. Generally, decision tables are generated for low risk test items. Decision tables are not related to risk level, but rather to complexity level. Decision tables are

generated for test items that have complex logic or multiple conditions and actions that need to be tested.

? IV. The conditions in the decision table represent negative tests generally. The conditions in the decision table represent both positive and negative tests, depending on whether they are valid or invalid inputs for the test item. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 4, page 42-43.

NEW QUESTION 103

Which of the following BEST describes exploratory testing?

- A. Exploratory testing is a suitable test technique which may replace both black-box and white-box test techniques
- B. Exploratory testing is a valid and useful black-box test technique since it focuses on test cases related to the architecture and design of a system
- C. Exploratory testing requires both solid specifications and much project time available for test execution
- D. Exploratory testing may be used within defined time periods, during which the tester may follow a test charter as a guide

Answer: D

Explanation:

Exploratory testing involves simultaneous test design and execution and is guided by a test charter, which outlines what needs to be tested, how it should be tested, and what to look for. This technique is typically conducted within predefined time periods, known as time-boxing, which allows testers to explore a system, understand its functionalities, and identify potential issues without detailed documentation or prior test case planning. The key aspects of exploratory testing include flexibility, adaptability, and the ability to respond to system behavior during testing. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.4.2.

NEW QUESTION 104

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 106

Which of the following is NOT a product risk?

- A. Poor software usability
- B. Failure-prone software is delivered
- C. Problems in defining the right requirements
- D. Software does not perform the intended functions

Answer: C

Explanation:

Problems in defining the right requirements is not a product risk, but rather a project risk. A product risk is a risk that affects the quality or performance of the software product itself, such as poor usability, failure-prone functionality, security vulnerabilities, compatibility issues, etc. A project risk is a risk that affects the management or delivery of the software project itself, such as unrealistic schedule, insufficient resources, unclear scope, changing requirements, etc. The other options are examples of product risks, as they relate to the software product's characteristics or features. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 12.

NEW QUESTION 108

Which of the following statements is LEAST likely to be describing component testing?

- A. It identifies defects in modules and classes.
- B. Simulators and stubs may be required.
- C. It mainly tests interfaces and interaction between components.
- D. It may be applied using a test-first approach.

Answer: C

Explanation:

Component testing (also known as unit testing or module testing) is a level of testing that focuses on verifying the functionality and quality of individual software components (such as modules, classes, functions, methods, etc.). Component testing mainly tests interfaces and interaction between components, as well as internal logic and data structures of the components. Component testing may be applied using a test-first approach (such as test-driven development or behavior-driven development), where tests are written before the code is implemented. Component testing does not identify defects in modules and classes, as this is a result of component testing, not an objective. Simulators and stubs may be required for component testing, as they can simulate or replace missing or incomplete components or external systems that are needed for testing. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 19.

NEW QUESTION 112

Can "cost" be regarded as Exit criteria?

- A. Ye
- B. Spending too much money on test ng will result in an unprofitable product, and having cost as an exit criterion helps avoid this
- C. N
- D. The financial value of product quality cannot be estimated, so it is incorrect to use cost as an exit criterion
- E. Ye
- F. Going by cost as an exit criterion constrains the testing project which will hello achieve the desired quality level defined for the project
- G. No The cost of testing cannot be measured effectively, so it is incorrect to use cost as an exit criterion

Answer: A

Explanation:

Cost can be regarded as an exit criterion for testing, because it is a factor that affects the profitability and feasibility of the software product. Testing is an investment that aims to improve the quality and reliability of the software product, but it also consumes resources, such as time, money, and human effort. Therefore, testing should be planned and executed in a way that balances the cost and benefit of testing activities. Having cost as an exit criterion helps to avoid spending too much money on testing, which may result in an unprofitable product or a loss of competitive advantage. Cost can also help to prioritize and focus the testing efforts on the most critical and valuable features and functions of the software product. However, cost should not be the only exit criterion for testing, as it may not reflect the true quality and risk level of the software product. Other exit criteria, such as defect rate, test coverage, user satisfaction, etc., should also be considered and defined in the test plan.

The other options are incorrect, because they either deny the importance of cost as an exit criterion, or they make false or unrealistic assumptions about the cost of testing. Option B is incorrect, because the financial value of product quality can be estimated, for example, by using cost-benefit analysis, return on investment, or cost of quality models. Option C is incorrect, because going by cost as an exit criterion does not necessarily constrain the testing project or help achieve the desired quality level. Cost is a relative and variable factor that depends on the scope, complexity, and context of the software product and the testing project. Option D is incorrect, because the cost of testing can be measured effectively, for example, by using metrics, such as test effort, test resources, test tools, test environment, etc.

NEW QUESTION 116

Which of the following statements about independent testing is WRONG?

- A. Independent testing is necessary because developers don't know any testing.
- B. Independent testing is best suited for the system test level.
- C. A certain degree of independence makes the tester more effective at finding defects.
- D. Independent test teams may find other types of detects than developers who are familiar with the system's structure.

Answer: A

Explanation:

Independent testing is testing performed by a person or group that is independent of the development team. Independent testing can have various degrees of independence, ranging from testers who are part of the same organization as developers to testers who are external contractors or consultants. Independent testing can have various benefits, such as reducing bias, increasing objectivity, improving quality, or providing different perspectives. Independent testing is not necessary because developers don't know any testing, as this is a wrong and disrespectful statement. Developers can perform various types of testing, such as unit testing, component testing, or integration testing. However, independent testing can complement developer testing by providing additional levels of verification and validation, such as system testing, acceptance testing, or non- functional testing. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 16-17.

NEW QUESTION 120

An Incident Management tool implements the following defect states; Open, Assigned, Solved,

Closed Consider the following defect report: Id T000561

Test Object "Warehouse Management" application Tester name; John Bishop

Date: 10th. April 2010 Test Case MRT558I

Status OPEN Severity Serious Priority

Problem- After inputting the Total Quantity item = 450 in the SV034 screen, the system shows an unexpected Error message=47

Correction: Developer name: Closing date:

Which of the following is a valid criticism of this report?

- A. The Priority, the Correction description and the Developer name are missing
- B. The version of the application is missing
- C. There is no link to the applicable requirement (traceability)
- D. The description is not highlighting the source of the problem

Answer: B

Explanation:

A valid criticism of this report is that the version of the application is missing. The version of the application is an important piece of information that should be included in a defect report, as it helps to identify which release or build of the software product contains the defect. The version of the application can also help to reproduce and debug the defect, as different versions may have different behaviors or features. The other options are not valid criticisms of this report. The priority, the correction description and the developer name are not missing, but rather not applicable for this report. The priority is a measure of how urgently a defect needs to be fixed, which can be assigned by the project manager or the defect tracking system, not by the tester who reports the defect. The correction description and the developer name are information that are added after the defect has been resolved, not when it has been reported. There is no link to the applicable requirement (traceability) is not a valid criticism of this report, because traceability is not a mandatory attribute of a defect report, but rather an optional one. Traceability is a relationship between two or more entities (such as requirements, test cases, defects, etc.) that shows how they are related or dependent on each other. Traceability can help to verify that the requirements are met by the test cases and defects, but it is not essential for reporting a defect. The description is not highlighting the source of the problem is not a valid criticism of this report, because highlighting the source of the problem is not a responsibility of the tester who reports the defect, but rather of the developer who fixes the defect. The description should provide enough information to describe what happened when the defect occurred, such as input values, expected results, actual results, error messages, screenshots, etc., but it does not need to explain why or how it happened. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 140.

NEW QUESTION 124

Which of the following is correct with regards to debugging?

- A. Debugging identifies the cause of a failure
- B. Debugging is often performed by test engineers
- C. Debugging is considered part of the testing activities
- D. Debugging is intended to find as many defects as possible in the code

Answer: A

Explanation:

Debugging is the process of finding, analyzing and removing the causes of failures in software. Debugging is not considered part of testing, but rather a development activity that can involve testing. Debugging is not intended to find as many defects as possible, but rather to fix the specific failure that was observed. Debugging is usually performed by developers, not by test engineers. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 1, page 6.

NEW QUESTION 126

Which of the following is a CORRECT statement about how a tester should communicate about defects, test results, and other test information?

- A. Testers should include personal opinions and judgements in defect reports and review findings
- B. Testers should emphasize the benefits of testing, such as increased quality and reduced risk
- C. Testers should reject all questions about their test findings and information
- D. Testers should take a command-and-control approach with the project team

Answer: B

Explanation:

Communication from testers about defects, test results, and other test information should emphasize the benefits of testing such as increased quality and reduced risk. This positive framing helps in reinforcing the value of testing and ensuring stakeholders understand the contribution of testing to the overall project success (ISTQB not-for-profit association).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf

NEW QUESTION 130

Which of the following is an example of the absence-of-errors fallacy?

- A. Repeating the same test cases will continue to find new defects, even after dozens of executions
- B. A small number of modules contains the most defects
- C. Since testing found very few defects, the system certainly will be successful
- D. Other than trivial cases, it is not feasible to test all combinations of inputs and preconditions

Answer: C

Explanation:

The absence-of-errors fallacy, as explained in the ISTQB syllabus, is the erroneous belief that having fewer defects found in testing equates to a system being more successful upon release. This misconception can lead stakeholders to undervalue thorough testing. Answer C illustrates this fallacy perfectly: assuming that because few defects were found, the system will be successful, neglects the many other factors that contribute to system success, including user satisfaction and fit-for-purpose.References:ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.1.5 "Absence-of-errors fallacy".

NEW QUESTION 133

What type of testing measures its effectiveness by tracking which lines of code were executed by the tests?

- A. Acceptance testing
- B. Structural testing
- C. Integration testing
- D. Exploratory testing

Answer: B

Explanation:

Structural testing is a type of testing that measures its effectiveness by tracking which lines of code were executed by the tests. Structural testing, also known as white-box testing or glass-box testing, is based on the internal structure, design, or implementation of the software. Structural testing aims to verify that the software meets the specified quality attributes, such as performance, security, reliability, or maintainability, by exercising the code paths, branches, statements, conditions, or data flows. Structural testing uses various coverage metrics, such as function coverage, line coverage, branch coverage, or statement coverage, to determine how much of the code has been tested and to identify any untested or unreachable parts of the code. Structural testing can be applied at any level of testing, such as unit testing, integration testing, system testing, or acceptance testing, but it is more commonly used at lower levels, where the testers have access to the source code.

The other options are not correct because they are not types of testing that measure their effectiveness by tracking which lines of code were executed by the tests. Acceptance testing is a type of testing that verifies that the software meets the acceptance criteria and the user requirements. Acceptance testing is usually performed by the end-users or customers, who may not have access to the source code or the technical details of the software. Acceptance testing is more concerned with the functionality, usability, or suitability of the software, rather than its internal structure or implementation. Integration testing is a type of testing that verifies that the software components or subsystems work together as expected. Integration testing is usually performed by the developers or testers, who may use both structural and functional testing techniques to check the interfaces, interactions, or dependencies between the components or subsystems. Integration testing is more concerned with the integration logic, data flow, or communication of the software, rather than its individual lines of code. Exploratory testing is a type of testing that involves simultaneous learning, test design, and test execution. Exploratory testing is usually performed by the testers, who use their creativity, intuition, or experience to explore the software and discover any defects, risks, or opportunities for improvement. Exploratory testing is more concerned with the behavior, quality, or value of the software, rather than its internal structure or implementation. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, Chapter 4: Test Techniques, Section 4.3: Structural Testing Techniques, Pages 51-54; Chapter 1: Fundamentals of Testing, Section 1.4: Testing Throughout the Software Development Lifecycle, Pages 11-13; Chapter 3: Static Testing, Section 3.4: Exploratory Testing, Pages 40-41.

NEW QUESTION 136

A bank offers a savings account with various interest rates based on the current balance in the account. The balance ranges and respective interest rates are:

Up to \$100.00 = 2%

\$100.01 to \$500.00 = 4%

\$500.01 to \$1,000.00 = 5% Above \$1,000.00 = 7%

Using two-point boundary value analysis, which of the following sets of test inputs provides the relatively highest level of boundary coverage?

A. \$5.00, \$100.00, \$499.99, \$1,000.00, \$1,000.01

B. \$100.00, \$100.01, \$100.02, \$500.00, \$999.99

C. \$100.00, \$500.00, \$1,000.00, \$1,000.01

D. \$5.00, \$100.00, \$500.00, \$1,000.01

Answer: B

Explanation:

Boundary Value Analysis (BVA) is a software testing technique in which tests are designed to include values at the boundaries. The concept is to focus on the boundaries since errors tend to occur at the edges of input ranges rather than in the middle.

Given the problem statement:

? Up to \$100.00 = 2%

? \$100.01 to \$500.00 = 4%

? \$500.01 to \$1,000.00 = 5%

? Above \$1,000.00 = 7%

Two-point boundary value analysis means testing the two boundaries of each range. For each range:

? The boundaries for "Up to \$100.00" would be \$100.00 and \$100.01.

? The boundaries for "\$100.01 to \$500.00" would be \$100.00 and \$500.00.

? The boundaries for "\$500.01 to \$1,000.00" would be \$500.00 and \$1,000.00.

? The boundaries for "Above \$1,000.00" would be \$1,000.00 and \$1,000.01. Now, let's examine the options:

? A. \$5.00, \$100.00, \$499.99, \$1,000.00, \$1,000.01

? B. \$100.00, \$100.01, \$100.02, \$500.00, \$999.99

? C. \$100.00, \$500.00, \$1,000.00, \$1,000.01

? D. \$5.00, \$100.00, \$500.00, \$1,000.01

Given the options, B provides the highest boundary coverage (ISTQB not-for-profit association) (Udemy).

References:

? Certified Tester Foundation Level v4.0

? 10 Sample Exams ISTQB Foundation Level (CTFL) v4.0

NEW QUESTION 140

Which of the following statements is CORRECT?

A. Test cases are made up of input values, expected results and actual results developed to cover test objectives

B. Test cases describe items or events to test that are derived from the test basis during the test analysis activity

C. Test cases are sequences of actions for test execution specified during the test implementation activity

D. Test cases are derived during the test design activity to cover test objectives or test conditions

Answer: C

Explanation:

A test case is a set of input values, execution preconditions, expected results and execution postconditions, developed for a particular objective or test condition. A test case is a sequence of actions for test execution that can be followed by a tester or a test automation tool. A test case is specified during the test implementation activity, which is the activity that prepares the testware needed for test execution. A test case does not include actual results, as these are obtained during test execution and compared with the expected results. A test case does not describe items or events to test, as these are derived from the test basis during the test analysis activity. A test case is not derived during the test design activity, as this is the activity that specifies the test conditions or objectives that need to be tested. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 3, page 23-24; Chapter 4, page 34.

NEW QUESTION 145

Given the following examples of entry and exit criteria:

* 1. A defined level of code coverage has been achieved

* 2. The test automation tool has been installed and properly configured

* 3. The number of unresolved defects is within the predefined limit

* 4. The performance test environment has been set-up and is available

* 5. The user stories have proper acceptance criteria defined

* 6. The testing budget has been spent and the project sponsor bears the risk of not testing any further

Which of the following BEST categorizes them as entry and exit criteria:

A. Entry criteria - 2, 4, 5 Entry criteria -1, 3, 4

B. Entry criteria - 2, 4 Entry criteria - 2, 4, 5, 6

C. Exit criteria -1,3,6 Exit criteria - 2, 5, 6

D. Exit criteria -1,3,5,6 Exit criteria -1,3

Answer: A

Explanation:

Entry and exit criteria are used to determine when to start and stop testing, respectively.

? Entry Criteria:

? Exit Criteria:

According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, these criteria help in effectively managing the testing process6†source.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 150

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 151

.....

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual ISTQB-CTFL Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the ISTQB-CTFL Product From:

<https://www.2passeasy.com/dumps/ISTQB-CTFL/>

Money Back Guarantee

ISTQB-CTFL Practice Exam Features:

- * ISTQB-CTFL Questions and Answers Updated Frequently
- * ISTQB-CTFL Practice Questions Verified by Expert Senior Certified Staff
- * ISTQB-CTFL Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- * ISTQB-CTFL Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year