



# ISTQB

## Exam Questions ISTQB-CTFL

ISTQB-Foundation Level Exam

## About ExamBible

### *Your Partner of IT Exam*

## Found in 1998

ExamBible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got. There are so many alike companies in this industry, however, ExamBible has its unique advantages that other companies could not achieve.

## Our Advances

### \* 99.9% Uptime

All examinations will be up to date.

### \* 24/7 Quality Support

We will provide service round the clock.

### \* 100% Pass Rate

Our guarantee that you will pass the exam.

### \* Unique Gurantee

If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

### NEW QUESTION 1

Which of the following is a correct reason to apply test automation?

- A. When a new test automation tool is launched
- B. When there are a lot of repetitive testing tasks
- C. When it is easy to automate
- D. When it is cheap to buy test automation tools

**Answer: B**

#### Explanation:

A correct reason to apply test automation is when there are a lot of repetitive testing tasks. Test automation is the use of software tools or scripts to perform or support testing activities, such as test case execution, test result comparison, test data generation, etc. Test automation can be beneficial when there are a lot of repetitive testing tasks that need to be performed frequently or consistently, such as regression testing, performance testing, load testing, etc. Test automation can help to save time and effort, increase reliability and accuracy, and improve coverage and efficiency of testing. The other options are not correct reasons to apply test automation. When a new test automation tool is launched is not a reason to apply test automation, but rather a factor for choosing a test automation tool. When it is easy to automate is not a reason to apply test automation, but rather a factor for evaluating the feasibility of test automation. When it is cheap to buy test automation tools is not a reason to apply test automation, but rather a factor for estimating the cost and benefit of test automation. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 10.

### NEW QUESTION 2

Manager responsibilities in formal review includes ad except one of the following:

- A. Planning the review
- B. Determines if the review objectives have been met
- C. Decide on the execution of reviews
- D. Allocate time for review

**Answer: B**

#### Explanation:

A formal review is a type of review that follows a defined process with formal entry and exit criteria and roles and responsibilities for participants. A formal review can have various roles involved, such as manager, moderator, author, reviewer and scribe. The manager responsibilities in formal review include all except one of the following:

- ? Planning the review (correct responsibility)
- ? Determines if the review objectives have been met (incorrect responsibility)
- ? Decide on the execution of reviews (correct responsibility)
- ? Allocate time for review (correct responsibility) The responsibility of determining if the review objectives have been met belongs to the moderator role, not to the manager role. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 3, page 28-29.

### 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 requirements<sup>1</sup>. 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 software<sup>2</sup>. Testing can only provide a level of confidence in the quality of the software, based on the coverage, effectiveness, and efficiency of the testing activities<sup>3</sup>.

? 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**

A test score indicator for students produces a performance score based on a combination of the number of consecutive hours studied (below 4 hours, 4 to 8 hours. 9 to 12 hours or above 12 hours) and the average intensity of focus on the material during the study time (low, medium or high).

Given the following test cases: hours intensity score

T1 3 low55

T2 14 high 95

T3 9 low75

What is the minimum number of additional test cases that are needed to ensure full coverage of all valid INPUT equivalence partitions?

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

**Answer:** C

**Explanation:**

Considering the various valid input equivalence partitions of hours studied and intensity, three additional test cases are needed to fully cover all valid partitions. This would typically include testing combinations that vary both the number of hours and the intensity levels not covered by the initial test cases (ISTQB Main Web). References:

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

**NEW QUESTION 7**

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 8**

A class grade application for instructors assigns letter grades based on students' numerical grades.

The letter grades for different numerical grades should be:

Above 89, up to 100 - A  
Above 79, up to 89 • B  
Above 69, up to 79 • C  
Above 59, up to 69 - D  
Below 60- F

Which of the following sets of test inputs would achieve the relatively highest equivalence partition coverage?

- A. 0, 58.59,70, 80
- B. 74, 79.84,85, 89
- C. 79, 89.90,99, 100
- D. 69, 79. 80, 89, 90

**Answer: D**

**Explanation:**

The set of test inputs that achieve the relatively highest equivalence partition coverage for grading students is option D: 69, 79, 80, 89, 90. This set effectively tests the boundaries between each grade category, ensuring that the grading system accurately transitions from one grade to another at the correct thresholds (ISTQB Main Web)

.References:

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

**NEW QUESTION 9**

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 10**

Which of the following is a typical product risk?

- A. Poor usability of the software
- B. A problem in the code developed by a 3rd party
- C. Low quality of the configuration data, test data and tests
- D. Problem in defining the right requirements

**Answer: A**

**Explanation:**

A typical product risk involves issues directly related to the software product's functionality, performance, usability, reliability, etc. Option A, "Poor usability of the software," directly impacts the end-user's interaction with the software and is a quality attribute of the product itself, making it a product risk. Options B, "A problem in the code developed by a 3rd party," C, "Low quality of the configuration data, test data and tests," and D, "Problem in defining the right requirements," can be considered either product or project risks depending on the context, but option A is the most directly associated with a typical product risk concerning the quality and usability of the software.

**NEW QUESTION 10**

Which of the following is the BEST reason for selecting a particular type of software development lifecycle model?

- A. The project manager's preference
- B. Tester skill level with the software development lifecycle model
- C. The project team's overall familiarity with the model
- D. The type of product being developed

**Answer: D**

**Explanation:**

The choice of a software development lifecycle (SDLC) model is primarily influenced by the type of product being developed. Different products and project requirements may demand different SDLC models to address specific challenges and needs efficiently. For instance, a complex, safety-critical product might best be served by a Waterfall model due to its structured nature and phase dependencies, while a more iterative and incremental model might be suited for projects requiring frequent feedback and changes. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 2.1 "Software Development Lifecycles".

**NEW QUESTION 12**

In the newest version of payroll system number of changes were made. As a tester you got a task to perform regression and confirmation tests. Which of the following project activities are related to confirmation testing?

- A. Testing due to the application of a new version of the interface



- B. Testing that fixes resolved the defects in the search function
- C. Testing if a system still works after update of an operating system
- D. Testing to ensure the adding of a new functionalities haven't broken existing functions

**Answer:** B

**Explanation:**

Confirmation testing, also known as re-testing, is performed to verify that specific defects have been successfully fixed.

Option A: "Testing due to the application of a new version of the interface" would typically involve regression testing, not confirmation testing.

Option B: "Testing that fixes resolved the defects in the search function" fits the description of confirmation testing as it focuses on ensuring that specific issues have been addressed. Option C: "Testing if a system still works after update of an operating system" is an example of regression testing, as it checks the overall system behavior after an update. Option D: "Testing to ensure the adding of new functionalities haven't broken existing functions" is another example of re (ISTQB not-for-profit association) (Udemy) it checks for unintended consequences of new changes.

Therefore, the correct answer is B.

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

**NEW QUESTION 14**

ST is a Software Testing organization which utilizes a testing knowledge base. Access to ST knowledge base can be either full or limited. Access level is determined based on ST certification and testing experience as follows:

- \* 1. If ST certified, with less than 5 years testing experience - allow limited access
- \* 2. If ST certified, 5-10 years of testing experience - allow full access
- \* 3. If not ST certified with 5-10 years of testing experience - allow limited access.

What would be the results for:

- A - ST certified. 12 years of testing experience
- B - Not ST certified. 7 years of testing experience
- C - Not ST certified. 3 years of testing experience

- A. A - unknown B - limited access C - unknown
- B. A - full access B - limited access C - unknown
- C. A - full access B - limited access C - limited access
- D. A - unknown B - full access C - unknown

**Answer:** B

**Explanation:**

The correct answer can be derived by applying the given rules to each case:

? A is ST certified and has 12 years of testing experience, which is more than 10 years. Therefore, A does not match any of the rules and the result is unknown.

? B is not ST certified and has 7 years of testing experience, which is between 5 and 10 years. Therefore, B matches rule 3 and the result is limited access.

? C is not ST certified and has 3 years of testing experience, which is less than 5 years. Therefore, C does not match any of the rules and the result is unknown.

Verified References: This question does not require any external references, as it is based on logical reasoning.

**NEW QUESTION 15**

A test manager defined the following test levels in her test plan; Component, System and Acceptance.

Which Software Development lifecycle is the Test Manager most likely following?

- A. V-Model
- B. Agile
- C. Waterfall
- D. Prototyping

**Answer:** A

**Explanation:**

The test manager is most likely following the V-model for software development. The V-model is a software development model that defines four testing levels that correspond to four development phases: component testing corresponds to component design, integration testing corresponds to architectural design, system testing corresponds to system requirements specification, and acceptance testing corresponds to user requirements specification. The V-model also defines the test planning and test execution activities for each testing level. Agile is a software development model that follows an iterative and incremental approach, where testing is integrated into each iteration and adapts to changing requirements and feedback. Waterfall is a software development model that follows a sequential and linear approach, where testing is performed after the development phase is completed. Prototyping is a software development model that involves creating a simplified version of the software to elicit user feedback and validate requirements before developing the final product. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 18.

**NEW QUESTION 17**

Which of the following statements contradicts the general principles of testing?

- A. Most defects are found in a small subset of a system's modules.
- B. If new defects are to be found we should run the same test set more often.
- C. Testing is better if it starts at the beginning of a project.
- D. How testing is done, is based on the situation in a particular project.

**Answer:** B

**Explanation:**

Statement B contradicts the general principles of testing, because running the same test set more often will not increase the chances of finding new defects, unless there are some changes in the system or environment that affect the test results. Running different test sets with different inputs, outputs or conditions would be more effective in finding new defects. Statements A, C and D are consistent with the general principles of testing. Statement A states that most defects are found in a small subset of a system's modules, which is true according to the defect clustering principle. Statement C states that testing is better if it starts at the beginning of a project, which is true according to the early testing principle. Statement D states that how testing is done, is based on the situation in a particular project, which is true according to the context-dependent testing principle. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus -

Springer, pages 4-6.

### NEW QUESTION 22

Why is it important to select a test technique?

- A. There are usually too many test cases that may be run on a system
- B. Test techniques help reduce the number of tests.
- C. The only way to test a software application is by using well-proven test techniques.
- D. Selecting the right test technique in a given situation increases the effectiveness of the test process by creating tests with a higher chance of finding bugs.
- E. Test techniques define the number of regression cycles, which in turn impact the project schedule.

**Answer:** C

#### Explanation:

Selecting the right test technique is crucial because different techniques are suited to different types of testing and can significantly increase the effectiveness of the testing process by creating tests that are more likely to find defects. While reducing the number of tests (A) and defining the number of regression cycles (D) are considerations in the testing process, they are not the primary reasons for selecting a test technique. The assertion that the only way to test a software application is by using well-proven test techniques (B) is too restrictive and does not acknowledge the adaptability required in testing to suit different contexts and objectives. Therefore, option C is the most comprehensive reason, as it focuses on the effectiveness and efficiency of testing, leading to the creation of high-quality tests that have a higher chance of finding bugs.

### NEW QUESTION 25

Why should you choose a test technique?

- A. Because you need to match the way you test to the content of the product under test
- B. Because of the time constraints that usually accompany a test project
- C. Because this way you cover the full scope of the product's functionality
- D. Because choosing a test technique is a common practice in software testing

**Answer:** A

#### Explanation:

You should choose a test technique because you need to match the way you test to the content of the product under test. A test technique is a method or process for deriving and selecting test cases based on some criteria or rules. Different test techniques are suitable for different types of software products, depending on their characteristics, functionalities, requirements, specifications, risks, etc. Choosing a test technique helps to ensure that the test cases are relevant, effective, and efficient for the product under test. The other options are not correct reasons to choose a test technique. Time constraints are not a factor for choosing a test technique, but rather for prioritizing or optimizing testing activities. Covering the full scope of the product's functionality is not a guarantee of choosing a test technique, but rather a goal of testing. Choosing a test technique is not a common practice in software testing, but rather a professional skill and responsibility. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 31.

### NEW QUESTION 28

Consider the following code  
int premium=2500; if (age<30)  
{  
premium = premium +1500;  
}

Which option suits for a correct combination of Boundary value and expected result. Assume first number as boundary followed by expected result.

- A. \* 29, 1500 \* 30. 2500
- B. \* 29. 4000 \* 30. 2500
- C. \* 29, 2500 \* 30. 1500
- D. \* 30, 1500 \* 31, 2500

**Answer:** B

#### Explanation:

In the given code snippet, the premium is increased by 1500 if the age is less than 30. Therefore, at the boundary value of age 29, the premium should be 2500 + 1500 = 4000, and at age 30, the premium should remain at its initial value of 2500, as the condition is no longer met. Option B correctly reflects this with 29, 4000 and 30, 2500 as the boundary value and the expected results, respectively.

### NEW QUESTION 33

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

- A. Usability II Requirements based scenarios III Testing parts of the code in isolation IV 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 34

While repotting a defect, which attribute indicates the degree of impact that the defect has on the system?

- A. Priority
- B. Severity
- C. Status
- D. Description

**Answer: B**

#### Explanation:

In defect reporting, the attribute that indicates the degree of impact that the defect has on the system is the severity. Severity reflects the seriousness of the defect in terms of its impact on the operation of the system, ranging from minor issues that do not significantly affect the system's functionality to critical defects that can cause system failure. Therefore, option B is the correct answer.

#### NEW QUESTION 38

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 42

Which of the following statements about reviews are TRUE?

- A. In walkthroughs the review meeting is typically led by the autho
- B. II Inspection is characterized by an open-ended review meetingIII Preparation before the review meeting is part of informal reviews IV Management rarely participates in technical review meetings
- C. II, III
- D. I, II
- E. I, IV
- F. III, IV

**Answer: C**

#### Explanation:

The following statements about reviews are true:

? I) In walkthroughs the review meeting is typically led by the author. A walkthrough is a type of review that has a predefined objective and agenda but no formal process or roles. A walkthrough is typically led by the author of the work product under review, who guides the participants through a scenario and solicits feedback.

? IV) Management rarely participates in technical review meetings. A technical review is a type of review that has a predefined objective and agenda but no formal process or roles. A technical review is typically performed by peers with technical expertise in order to evaluate technical aspects of a work product. Management rarely participates in technical review meetings, as they may not have sufficient technical knowledge or skills to contribute effectively. The following statements about reviews are false:

? II) Inspection is characterized by an open-ended review meeting. An inspection is a type of review that follows a defined process with formal entry and exit criteria and roles and responsibilities for participants. An inspection is characterized by a structured review meeting with a fixed duration and agenda.

? III) Preparation before the review meeting is part of informal reviews. Preparation before the review meeting is part of formal reviews, such as inspections or technical reviews. Preparation involves checking

#### NEW QUESTION 47

Which of the following activities are part of test planning?

- I) Setting the entry and exit criteria
- II) Determining the validity of bug reports
- III) Determining the number of resources required
- IV) Determining the expected result for test cases



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

**Answer:** B

**Explanation:**

Test planning is a key activity in the testing process that involves defining the objectives, approach, resources, and schedule of intended test activities. Setting the entry and exit criteria (I) and determining the number of resources required (III) are integral parts of test planning. Determining the validity of bug reports (II) is more aligned with test analysis or test management activities post-execution, and determining the expected result for test cases (IV) is part of test design. Therefore, options I and III (B) are the activities that belong to test planning.

**NEW QUESTION 51**

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. 1. 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. 2. 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: 3.

? 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. 3. The project plan describes the overall objectives, scope, assumptions, risks, and deliverables of the software project. 4. The requirements specification describes the functional and non-functional requirements of the software product. 5. 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 53**

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 incorrect
- B. The whole company should adopt same standard for review of all artifacts.
- C. The thought process is correct
- D. The whole company should decide on the review method based on their CMM level.
- E. The thought process is incorrect
- F. Same artifact can be reviewed using different review methods
- G. The thought process is correct
- H. It wastes time to review same artifact using different 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 57**

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 61**

Which of the following BEST distinguishes the terms "validation" and "verification"?

- A. Verification is confirmation through the provision of objective evidence that the specified requirements have been met while validation is confirmation through the provision of objective evidence that the requirements for a specific intended use have been met
- B. Verification is confirmation through the provision of subjective evidence that the specified requirements have been met while validation is confirmation through the provision of subjective evidence that the designs for a specific intended use have been met
- C. Validation is confirmation through the provision of objective evidence that the specified requirements have been met while verification is confirmation through the provision of objective evidence that the requirements for a specific intended use have been met
- D. Validation is confirmation through the provision of subjective evidence that the specified requirements have been met while verification is confirmation through the provision of subjective evidence that the designs for a specific intended use have been met

**Answer:** A

**Explanation:**

In the context of software testing, the ISTQB Certified Tester Foundation Level (CTFL) v4.0 differentiates between "validation" and "verification" based on their respective focuses in the software development lifecycle. Verification is the process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. In simpler terms, verification is about checking the product against the specified requirements to ensure it was built correctly. Validation, on the other hand, involves evaluating a system or component during or at the end of the development process to determine whether it meets specified requirements for its intended use. This means validation is about ensuring the product fulfills its intended use and meets the needs of the user.

References:

? ISTQB CTFL Syllabus v4.0: ISTQB Official Website

? ISTQB Foundation Level Resources v4.0: ASTQB Resources

**NEW QUESTION 62**

"Statement Testing" is part of;

- A. Specification Based testing
- B. Decision Testing
- C. Experience based testing
- D. Structured based testing

**Answer:** D

**Explanation:**

Statement Testing is a type of white-box testing technique where the test cases are designed based on the implementation of the software, specifically aiming to execute every statement in the code at least once. This falls under the category of structure-based testing (also known as white-box testing), where the internal structure of the system is used to design test cases. Therefore, option D is correct.

**NEW QUESTION 64**

Which of the following is NOT an objective of testing?

- A. Finding defects
- B. Providing information for decision-making
- C. Gaining confidence about the level of quality of the software
- D. Analyzing and removing the cause of failures

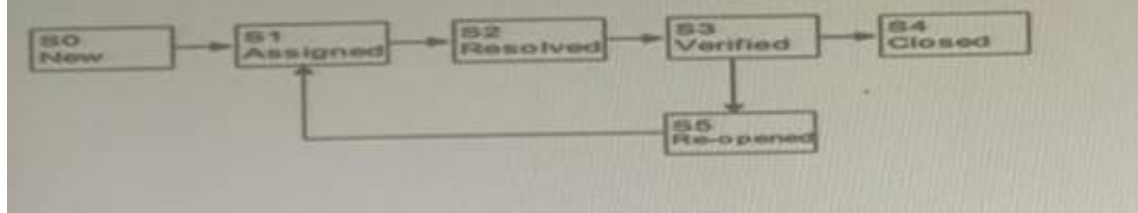
**Answer:** D

**Explanation:**

Analyzing and removing the cause of failures is not an objective of testing, but rather a task of development or maintenance. A failure is an event or behavior that deviates from the expected or specified result of a system under test. A failure is caused by an error (also known as a mistake or a fault) in the software code, design, or specification. Analyzing and removing the cause of failures is a process of locating and fixing errors in the software code, design, or specification, which is also known as debugging or defect resolution. Analyzing and removing the cause of failures does not aim to find or report defects, but rather to correct or prevent them. The other options are objectives of testing. Finding defects is one of the main objectives of testing, as it helps to improve the quality and reliability of the software product. Providing information for decision-making is another objective of testing, as it helps to support decision making and risk management. Gaining confidence about the level of quality of the software is another objective of testing, as it helps to assure that the software product meets its requirements and customer or user needs and expectations. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 3.

**NEW QUESTION 67**

Which sequence of state transition 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->S4
- B. S0->S1->S2->S3->S5^>S1
- C. S0->S1->S2->S3->S5->S1->S2->S3
- D. S0->S1->S2->S3->S5->S3->S4

**Answer: C**

#### Explanation:

The figure depicts the life-cycle of a defect using state transition testing. State transition testing is a technique that models how a system transitions from one state to another depending on events or conditions. The figure shows six states (S0 to S5) and seven transitions (T0 to T6). The correct sequence of state transitions that follows the figure is S0->S1->S2->S3->S5->S1->S2->S3. This sequence represents the following scenario:

- ? S0: The defect is not yet detected (initial state).
- ? T0: The defect is detected by testing (event).
- ? S1: The defect is reported and registered (state).
- ? T1: The defect is assigned to a developer for fixing (event).
- ? S2: The defect is being fixed by the developer (state).
- ? T2: The developer fixes the defect and delivers a new version (event).
- ? S3: The defect is verified by testing (state).
- ? T5: The testing fails to confirm that the defect is fixed (event).
- ? S5: The defect is rejected by testing (state).
- ? T6: The defect is reassigned to a developer for fixing (event).
- ? S1: The defect is reported and registered (state).
- ? T1: The defect is assigned to a developer for fixing (event).
- ? S2: The defect is being fixed by the developer (state).
- ? T2: The developer fixes the defect and delivers a new version (event).
- ? S3: The defect is verified by testing (state). The other sequences are incorrect, as they do not follow the transitions shown in the figure. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 40-41.

#### NEW QUESTION 69

Which of the following is the main benefit of a configuration management of testware?

- A. All testware is backed up with restore option, including incident reports and change request
- B. The testware can be traced to information in requirements tools and to the bug tracking system.
- C. All testware items are identified, version controlled, tracked for changes with relation to each other
- D. There is an easy way to assess the level to test coverage provided by the existing tests

**Answer: C**

#### Explanation:

Configuration management of testware is a critical aspect of maintaining the integrity and traceability of test assets throughout the testing lifecycle. The main benefit of configuration management is to ensure that all testware items, such as test cases, test scripts, test data, and test results, are systematically identified, version controlled, and tracked for changes in relation to each other.

Option C accurately describes this benefit. By applying configuration management principles to testware, teams can manage changes to test assets efficiently, ensuring that the testware remains consistent, up-to-date, and aligned with the version of the software under test. This control mechanism facilitates the reproducibility of tests, enhances the reliability of testing activities, and supports traceability from requirements through to defects.

Options A, B, and D describe other aspects of test management and testing processes but do not capture the core benefit of configuration management of testware, which is centered on the systematic control and tracking of testware items.

#### NEW QUESTION 70

Which of the following tasks is MOST LIKELY to be performed by the tester?

- A. Develop a test strategy and test policy for the organization
- B. Promote and advocate the test team within the organization
- C. Create the detailed test execution schedule
- D. Introduce suitable metrics for measuring test progress

**Answer: C**

#### Explanation:

Testers are typically involved in creating detailed test execution schedules, among other tasks such as designing tests, executing tests, and logging defects. Creating a test strategy and test policy, promoting and advocating the test team, and introducing metrics are typically responsibilities of test managers or senior roles.

In the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, the responsibilities of testers include creating test cases, setting up test (ISTQB not-for-profit association)nts, executing tests, and reporting defects, which align with creating detailed test execution schedules6†source.

References:

- ? Certified Tester Foundation Level v4.0
- ? ISTQB Foundation Level Syllabus 4.0 (2023)

#### NEW QUESTION 73

Which of the following statements BEST describes how test cases are derived from a use case?

- A. Test cases are derived based on non-functional requirements such as usability
- B. Test cases are created using white-box test techniques to execute scenarios of use cases
- C. Test cases are derived based on pair testing between a user and a tester to find defects
- D. Test cases are designed to cover various user behaviors, including basic, exceptional or alternative and error behaviors associated with human users or systems

**Answer: D**

**Explanation:**

Use cases describe a system's behavior as it responds to a request from a user. They typically consist of various scenarios, such as basic flow, alternative flow, and exceptional flow, which represent possible behaviors when a user interacts with the system. When deriving test cases from use cases, it is important to cover these different types of user behaviors. Test cases should be designed to verify how the system behaves during each of these scenarios. This ensures that the system operates correctly for normal and error conditions encountered by human users or systems interacting with the application. Thus, test cases derived from use cases aim to cover basic, exceptional, and alternative flows, ensuring comprehensive coverage. References: ? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.2.4.

**NEW QUESTION 77**

4 equivalence classes are given for integer values:

$0 < x < 100$

$100 \leq x \leq 200$

$200 < x < 500$

$x \geq 500$

Which of the following options represent correct set of data for valid equivalence class partitions?

- A. 50; 100; 200. 1000
- B. 0. 1.99, 100.200,201.499, 500;
- C. 0.50; 100; 150.200.350.500;
- D. 50; 100; 250; 1000

**Answer: C**

**Explanation:**

The correct set of data for valid equivalence class partitions should include one value from each equivalence class, and no value from outside the range. Option C satisfies this condition, as it has one value from each of the four equivalence classes (50, 100, 250, 500). Option A has two values from the same equivalence class (100 and 200), option B has values outside the range (0 and 0.99), and option D has two values from the same equivalence class (1000 and 500). Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 35.

**NEW QUESTION 82**

A bank software has a feature of locking down accounts that have an overdraft higher than \$10,000.

Due to a bug in the code, the system locks down accounts once they reached \$1000 overdraft.

Based on this context, which of the following is correct statement

- A. It is an example of a defect that it goes undetected, will cause harm to the end customer and to the company (the bank)
- B. It is an example of a defect, that if goes undetected, will cause harm to the end customer but not to the company (the bank)
- C. It is an example of a defect that if goes undetected, will not cause harm to the end customer, but will cause harm to the company (the bank)
- D. It is an example of a defect that if goes undetected, will not cause any real harm to either the end customer or to the company (the bank)

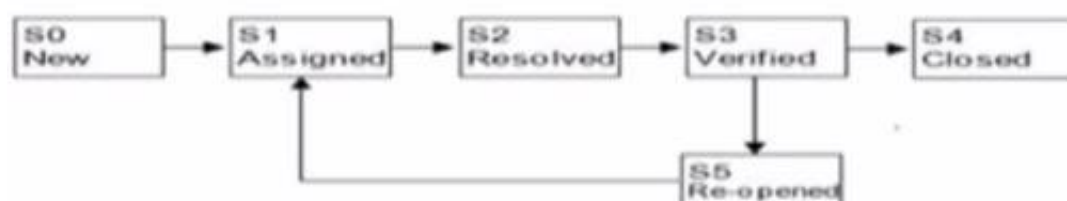
**Answer: A**

**Explanation:**

A defect is a flaw in a component or system that can cause the component or system to fail to perform its required function. A defect can cause harm to the end customer, the company, or both, depending on the impact and severity of the failure. In this case, the defect in the code causes the system to lock down accounts that have an overdraft higher than \$1000, instead of \$10,000. This can cause harm to the end customer, as they may lose access to their funds and face inconvenience or financial difficulties. This can also cause harm to the company, as they may lose customer trust and satisfaction, face legal issues or complaints, or incur additional costs for fixing the defect and restoring the accounts. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 1, page 3-4.

**NEW QUESTION 83**

Which sequence of states 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 85

Which of the following is an example of black-box dynamic testing?

- A. Functional Testing
- B. Code inspection
- C. Checking memory leaks for a program by executing it
- D. Coverage analysis

**Answer:** A

#### Explanation:

Functional testing is an example of black-box dynamic testing. Black-box testing (also known as specification-based testing) is a type of testing that does not consider the internal structure or implementation of the system under test, but rather its external behavior or functionality. Dynamic testing is a type of testing that involves executing the system under test with various inputs and observing its outputs. Functional testing is a type of black-box dynamic testing that verifies that the system under test performs its intended functions according to its requirements or specifications. Functional testing can be performed at various levels and scopes depending on the objectives and criteria of testing. The other options are not examples of black-box dynamic testing. Code inspection is an example of white-box static testing. White-box testing (also known as structure-based testing) is a type of testing that considers the internal structure or implementation of the system under test. Static testing is a type of testing that does not involve executing the system under test, but rather analyzing it for defects, errors, or violations of standards. Code inspection is a type of white-box static testing that involves examining the source code of the system under test for quality, readability, maintainability, etc. Checking memory leaks for a program by executing it is an example of white-box dynamic testing. Memory leaks are defects that occur when a program fails to release memory that it has allocated but no longer needs. Checking memory leaks for a program by executing it requires knowledge and access to the internal structure or implementation of the program, such as memory allocation and deallocation mechanisms, pointers, references, etc. Coverage analysis is an example of white-box static testing. Coverage analysis is a technique that measures how much of the code or structure of the system under test has been exercised by a test suite. Coverage analysis requires knowledge and access to the internal structure or implementation of the system under test, such as statements, branches, paths, conditions, etc. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 7.

#### NEW QUESTION 89

Which of the following sentences describe a product risk?

- A. The application might not be able to provide the expected responsiveness under a load of up-to 300 concurrent users
- B. Failure in acquiring an adequate and test automation tool
- C. A wrong configuration of the test environment that causes incidents related to the environment and not to the software under test
- D. The development team lacks knowledge of the technology on which the product is based

**Answer:** A

#### Explanation:

This question relates to identifying product risks, which are potential problems associated with the product itself, such as software functionality, reliability, usability, and performance. Option A describes a scenario where the application might not meet performance requirements under specific conditions (up to 300 concurrent users), which directly impacts the product's ability to perform its intended function. This is a classic example of a product risk, as it concerns the product's quality and its ability to meet user needs. Options B, C, and D, on the other hand, relate to project risks, which are concerns related to the management and execution of the project, such as tool acquisition, environment configuration, and team expertise, rather than the quality of the product itself.

#### NEW QUESTION 92

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 the 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 97

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 101

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 104

Which of the following statements is the BEST example of non-functional testing?

- A. Tests which capture the time it takes to save a file
- B. Tests which calculate overtime pay for those employees entitled to such
- C. Tests related to "what" the system should do
- D. Tests based on the internal structure of a component or system

**Answer: A**

#### Explanation:

Non-functional testing refers to testing aspects that do not relate to specific behaviors or functions of the software but to attributes such as performance, usability, reliability, etc. Tests that capture the time it takes to save a file directly relate to the performance of the system, thus falling under non-functional testing. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.2.5 "Functional and Non- functional Testing".

#### NEW QUESTION 109

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 wrong
- B. Ensuring quality mandates that static testing is performed after performing the dynamic testing.
- C. The decision was right
- D. Static testing is usually redundant if a product is planned to go through a full-cycle of dynamic testing.
- E. The decision was right
- F. Most of the bugs are easier to identify during the dynamic testing.
- G. The decision was wrong
- 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 110

The following sentences refer to the 'Standard for Software Test Documentation' specification (IEEE 829).

Which sentence is correct?

- A. Any deviation from this standard should be approved by management, marketing & development
- B. Most test documentation regimes follow this spec to some degree, with changes done to fit a specific situation or organization
- C. The key to high quality test documentation regimes is strict adherence to this standard
- D. This test plan outline is relevant for military project
- E. For consumer market projects there is a different specification with fewer items.

**Answer: B**

#### Explanation:

The IEEE 829 standard is a widely used specification for test documentation, but it is not mandatory or universal. Most test documentation regimes follow this spec to some degree, with changes done to fit a specific situation or organization. The standard does not require any approval from management, marketing or development for any deviation, nor does it depend on the type of project (military or consumer market). The standard also does not guarantee high quality test documentation regimes, as it only provides a general outline and format, not the actual content or quality criteria. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 16.

#### NEW QUESTION 115

A system has valid input numbers ranging between 1000 and 99999 (both inclusive). Which of the following inputs are a result of designing tests for all valid equivalence classes and their boundaries?

- A. 999.1000.23232.99999.100000
- B. 999.1000.50000.100000.100001
- C. 999.100000
- D. 1000,50000,99999

**Answer:** B

#### Explanation:

A correct list of boundary values for the P input should include the minimum and maximum values of the valid range (15 and 350), as well as the values just below and above the boundaries (14 and 351). 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 B satisfies this condition, as it has all four boundary values (14, 15, 350, 351). Option A has two values from the same equivalence class (1000 and 99999), option C has two values outside the range (999 and 100000), 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 116

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 117

What is 'Component Testing'?

- A. Integration Testing
- B. Functional testing
- C. Experience-based testing
- D. A test level

**Answer:** D

#### Explanation:

Component testing is a test level. A test level is a group of test activities that are organized and managed together based on some common characteristics or objectives. A test level can be defined based on various factors, such as the scope and target of testing, the phase and model of development, the stakeholders and roles involved in testing, etc. Component testing (also known as unit testing or module testing) is a test level that focuses on verifying the functionality and quality of individual software components (such as modules, classes, functions, methods, etc.). Component testing can be performed by developers or testers using various techniques and tools depending on the type and complexity of the components. The other options are not test levels. Integration testing is another test level that focuses on verifying the functionality and quality of groups of software components that interact with each other or with external systems. Functional testing is a type of black-box dynamic testing that verifies that the system under test performs its intended functions according to its requirements or specifications. Experience-based testing is a category of test design techniques that rely on the tester's knowledge and intuition to derive and select test cases based on their experience with similar systems, technologies, domains, risks, etc. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 19.

#### NEW QUESTION 122

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 limits<sup>1</sup>. A requirement is a condition or capability needed by a user to solve a problem or achieve an objective<sup>2</sup>. 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 126

Which or the following is a valid collection of equivalence classes for the following problem: An integer field shall contain values from and including 1 to and including 15.

- A. Less than 0.1 through 14. 15 and more
- B. Less than 1.1 through 14. more than 15
- C. negative number
- D. 1 through 15. above 15
- E. Less than 1.1 through 15. more than 15

**Answer: D**

#### Explanation:

Equivalence partitioning is a black-box test design technique where inputs to the software or system are divided into groups that are expected to exhibit similar behavior. For an integer field that should accept values from 1 to 15, the valid equivalence class is 1 through 15. The invalid equivalence classes are numbers less than 1 and numbers more than 15. Therefore, option D, "Less than 1, 1 through 15, more than 15," correctly identifies the valid equivalence class along with the two invalid classes, covering all possible input scenarios for the field. Options A, B, and C either do not accurately capture the valid range or incorrectly specify the range boundaries.

#### NEW QUESTION 130

A company runs a pilot project for evaluation of a test automation tool. Which of the following is NOT a valid object of this pilot project?

- A. Get familiar with the functionality and options of the tool
- B. Check how the tool fits to the existing test processes
- C. Train all testers on using the tool
- D. Decide upon standards for tool implementation

**Answer: C**

#### Explanation:

? A pilot project is a small-scale experiment or trial that is conducted to evaluate the feasibility, effectiveness, and suitability of a test automation tool before implementing it on a larger scale<sup>1</sup>.

? The objectives of a pilot project may vary depending on the context and scope of the test automation initiative, but some common ones are<sup>2</sup>:

? Therefore, option C is not a valid objective of a pilot project, as it is not necessary to train all testers on using the tool at this stage. Training all testers on using the tool would be more appropriate after the tool has been selected and approved for full-scale implementation, and after the standards and guidelines have been established. Training all testers on using the tool during the pilot project would be inefficient, costly, and premature, as the tool may not be suitable or effective for the intended purpose, or may be replaced by another tool later.

References:

? 1: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 82

? 2: 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

#### NEW QUESTION 133

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 138

Given the following state model of sales order software: SEE ATTACHMENT

Which of the following sequences of transitions provides the highest level of transition coverage for the model (assuming you can start in any state)?

- A. IN PRODUCTION -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED -> PLACED
- B. IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION
- C. PLACED -> IN PRODUCTION -> SHIPPED -> CANCELLED -> PLACED
- D. PLACED -> CANCELLED -> PLACED -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED

**Answer: B**

#### Explanation:

To achieve the highest level of transition coverage, one must consider all the possible transitions between the states in the given state model of the sales order software. The transitions in the sequence provided in Option B - "IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION" cover all the states and transitions effectively. This covers the transitions from IN PRODUCTION to SHIPPED, SHIPPED to INVOICED, INVOICED to CANCELLED, CANCELLED to PLACED, and PLACED to IN PRODUCTION, thereby maximizing the transition coverage. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.3.5.

#### NEW QUESTION 143

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 148

Which of the following activities is NOT a part of the fundamental testing process?

- A. Archiving automation code
- B. Test status reporting
- C. Test process improvement
- D. Build release and maintenance

**Answer:** D

#### Explanation:

The fundamental testing process includes activities that are directly related to the planning, preparation, execution, and evaluation of tests, as well as the closure activities of the testing phase. Option D, "Build release and maintenance," falls outside the scope of the fundamental testing process as it relates more to software development and operations rather than specific testing activities. Options A, "Archiving automation code," B, "Test status reporting," and C, "Test process improvement," are all activities that can be part of or associated with the fundamental testing process. Archiving automation code is part of test closure, test status reporting is part of test monitoring and control, and test process improvement can be an outcome of test closure activities.

#### NEW QUESTION 152

Which of the following statements about re-testing and regression testing are TRUE? I Re-testing should be performed after a defect is fixed.

II Regression testing should always be performed after a defect is fixed.

III. Re-testing and regression testing may be performed at any test level.

IV Regression testing may include functional, non-functional and structural testing.

- A. Re-testing should be included in the debugging activity.
- B. I, II
- C. IV
- D. II, V
- E. I, III
- F. I
- G. IV, V

**Answer:** A

#### Explanation:

The following statements about re-testing and regression testing are true:



? I) Re-testing should be performed after a defect is fixed. Re-testing is a type of testing that verifies that a defect has been successfully resolved by executing a test case that previously failed due to that defect. Re-testing should be performed after a defect is fixed and delivered to ensure that it does not cause any new failures or side effects.

? III) Re-testing and regression testing may be performed at any test level. Re-testing and regression testing are not limited to a specific test level, but can be applied at any level depending on the context and objectives. For example, re-testing and regression testing can be performed at unit level, integration level, system level or acceptance level.

? IV) Regression testing may include functional, non-functional and structural testing. Regression testing is a type of testing that verifies that previously tested software still performs correctly after changes. Regression testing may include various types of testing depending on the scope and purpose of the changes. For example, regression testing may include functional testing to check if the software meets its requirements, non-functional testing to check if the software meets its quality attributes, or structural testing to check if the software meets its design or code standards. The following statement about re-testing and regression testing is false:

? II) Regression testing should always be performed after a defect is fixed.

Regression testing is not always necessary after a defect is fixed, as some defects may have a low impact or low likelihood of affecting other parts of the software. Regression testing should be performed after a defect is fixed only if there is a risk of introducing new defects or causing existing defects due to the changes made to fix the defect. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 2, page 19; Chapter 4, page 45.

#### NEW QUESTION 156

A Static analysis tool analyzes a given program's CONTROL FLOW among other things. Which of the following options represents the most likely outcome of the control flow analysis:

- A. Identification of unreachable code
- B. Report on adherence to the coding standards
- C. Number of comment lines
- D. Number of source code lines

**Answer:** A

#### Explanation:

A static analysis tool is a tool that analyzes a given program's source code or executable code without executing it. A static analysis tool can perform various types of analysis on a program's code, such as syntax checking, data flow analysis, control flow analysis, complexity measurement, coding standards compliance checking, etc. Control flow analysis is a type of analysis that examines how a program's statements are executed in different paths or branches. One of the most likely outcomes of control flow analysis is identification of unreachable code, which is code that can never be executed due to logical errors or design flaws. Unreachable code can reduce readability and maintainability of the code, as well as increase complexity and size. The other options are not outcomes of control flow analysis, but rather outcomes of other types of analysis. Report on adherence to coding standards is an outcome of coding standards compliance checking. Number of comment lines and number of source code lines are outcomes of complexity measurement. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 8.

#### NEW QUESTION 160

How can testing contribute to higher quality?

- A. Testing help to measure the quality of software.
- B. Testing ensures that remaining defects are documented.
- C. Testing removes errors in the software.
- D. Testing eliminates the risk with software.

**Answer:** A

#### Explanation:

Testing can contribute to higher quality by helping to measure the quality of software. Quality is defined as the degree to which a component or system satisfies specified requirements and customer or user needs and expectations. Testing is a process of evaluating a component or system by applying inputs and observing outputs, and comparing them with expected results. Testing can help to measure the quality of software by providing information on its functionality, performance, usability, security, reliability, etc. Testing can also help to identify and report defects in software, which can lead to improvement actions and quality assurance activities. The other options are not accurate descriptions of how testing can contribute to higher quality. Testing does not ensure that remaining defects are documented, but rather that detected defects are reported. Testing does not remove errors in software, but rather finds defects in software behavior or quality. Testing does not eliminate the risk with software, but rather assesses and manages the risk with software. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 3.

#### NEW QUESTION 165

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 167

You are testing a room upgrade system for a hotel. The system accepts three differed types of room (increasing order of luxury): Platinum. Silver and Gold Luxury. ONLY a Preferred Guest Card holder s eligible for an upgrade.

Below you can find the decision table defining the upgrade eligibility:



## Conditions

Preferred Guest Card holder	YES	YES	NO	NO
Room Type	Silver	Platinum	Silver	Platinum

48	Offer upgrade to Gold Luxury	YES	NO	NO	NO
	Offer upgrade to Silver	N/A	YES	N/A	NO

What is the expected result for each of the following test cases?

Customer A: Preference Guest Card holder, holding a Silver room Customer B: Non Preferred Guest Card holder, holding a Platinum room

- A. Customer A; doesn't offer any upgrade; Customer B: offers upgrade to Gold luxury room
- B. Customer A: doesn't offer any upgrade; Customer B: doesn't offer any upgrade.
- C. Customer A: offers upgrade to Gold Luxury room; Customer B: doesn't offer any upgrade
- D. Customer A: offers upgrade to Silver room; Customer B: offers upgrade to Silver room.

**Answer: C**

### Explanation:

According to the decision table in the image, a Preferred Guest Card holder with a Silver room is eligible for an upgrade to Gold Luxury (YES), while a non-Preferred Guest Card holder, regardless of room type, is not eligible for any upgrade (NO). Therefore, Customer A (a Preferred Guest Card holder with a Silver room) would be offered an upgrade to Gold Luxury, and Customer B (a non-Preferred Guest Card holder with a Platinum room) would not be offered any upgrade. References = The answer is derived directly from the decision table provided in the image; specific ISTQB Certified Tester Foundation Level (CTFL) v4.0 documents are not referenced.

## NEW QUESTION 171

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 175

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 176

The following part of a business process flow is specified; REPEAT (book a bill) UNTIL (User presses Cancel). How many test cases are necessary in order to achieve 100% branch coverage of the process flow?

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

D. Infinite

**Answer:** C

**Explanation:**

To achieve 100% branch coverage of the process flow, we need to test both the true and false outcomes of the condition (User presses Cancel). Branch coverage is a type of structural testing that measures how many decision outcomes in a program have been executed by a test suite. Branch coverage can be used to assess the adequacy or completeness of a test suite.

To test the true outcome of the condition, we need a test case that simulates the user pressing Cancel after booking a bill. This test case will exit the loop and end the process flow.

To test the false outcome of the condition, we need a test case that simulates the user not pressing Cancel after booking a bill. This test case will repeat the loop and book another bill.

Therefore, we need at least two test cases to achieve 100% branch coverage of the process flow. One test case for each possible outcome of the condition.

Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 40-41.

**NEW QUESTION 177**

Which of the following is NOT a common objective of testing?

- A. Finding defects in the software
- B. Preventing defects
- C. Debugging the software to find the reason for defects
- D. Providing information on the status of the system

**Answer:** C

**Explanation:**

Debugging the software to find the reason for defects is not a common objective of testing, but rather a task of development or maintenance. Debugging is a process of locating and fixing errors in the software code, while testing is a process of finding and reporting defects in the software behavior or quality. Testing does not aim to fix defects, but rather to provide information on their existence and impact. The other options are common objectives of testing. Finding defects in the software is one of the main objectives of testing, as it helps to improve the quality and reliability of the software. Preventing defects is another objective of testing, as it helps to avoid rework and reduce costs and risks. Providing information on the status of the system is another objective of testing, as it helps to support decision making and risk management. Verified

References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 3.

**NEW QUESTION 182**

You need to test the login page of a web site. The page contains fields for user name and password. Which test design techniques are most appropriate for this case?

- A. Decision table testing, state transition testing.
- B. Equivalence partitioning, Boundary value analysis.
- C. Exploratory testing, statement coverage.
- D. Decision coverage, fault attack.

**Answer:** B

**Explanation:**

Equivalence partitioning and boundary value analysis are test design techniques that are most appropriate for testing the login page of a web site. The page contains fields for user name and password, which are input values that can be divided into partitions of equivalent data. 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. Boundary value analysis is a technique that tests boundary values between partitions of equivalent data. Boundary values are values at the edge of an equivalence partition or at the smallest incremental distance on either side of an edge. Boundary value analysis can be used to detect defects caused by incorrect handling of boundary conditions. For example, for testing the user name field, we can identify two equivalence partitions: valid user name (existing and correct) and invalid user name (non-existing or incorrect). The boundary values for these partitions are the minimum and maximum length of user name allowed by the system.

Decision table testing and state transition testing are not suitable for testing the login page of a web site, as they are more applicable for testing components that have multiple inputs and outputs that depend on logical combinations of conditions or events. Decision table testing is a technique that shows combinations of inputs and/or stimuli (causes) with their associated outputs and/or actions (effects). State transition testing is a technique that models how a system transitions from one state to another depending on events or conditions.

Exploratory testing and statement coverage are not suitable for testing the login page of a web site, as they are more applicable for testing components that require learning, creativity and intuition or structural analysis. Exploratory testing is an approach to testing that emphasizes learning, test design and test execution at the same time. Exploratory testing relies on the tester's skills, creativity and intuition to explore the software under test and discover defects. Statement coverage is a type of structural testing that measures how many statements in a program have been executed by a test suite. Statement coverage can be used to assess the adequacy or completeness of a test suite.

Decision coverage and fault attack are not suitable for testing the login page of a web site, as they are more applicable for testing components that have complex logic or potential errors. Decision coverage is a type of structural testing that measures how many decision outcomes in a program have been executed by a test suite. Decision coverage can be used to assess the adequacy or completeness of a test suite. Fault attack is a type of functional testing that deliberately introduces faults into a system in order to provoke failures or errors. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 34-46; Chapter 5, page 47-48.

**NEW QUESTION 183**

For a mandatory input field "ZIP code" the following rules are given:

- 1 - The valid ZIP code format is 5 numeric digits.
- 2 - The code has to exist in the post office's official ZIP code list

Using equivalence classes partitioning, how many test cases are required to test this field?

- A. 8
- B. 3
- C. 6
- D. 4

**Answer:** D

**Explanation:**

Equivalence classes 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 classes partitioning can be used to reduce the number of test cases by selecting one representative value from each partition. For the ZIP code field, there are four equivalence classes based on the given rules:

? Valid ZIP code format and valid ZIP code value (e.g., 12345)

? Valid ZIP code format and invalid ZIP code value (e.g., 99999)

? Invalid ZIP code format and valid ZIP code value (e.g., 1234)

? Invalid ZIP code format and invalid ZIP code value (e.g., ABCDE) Therefore, four test cases are required to test this field, one for each equivalence class.

Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 37-38.

**NEW QUESTION 184**

A QA manager of a start-up company needs to implement within a week a low cost incident management tool. Which of the following is the best option?

A. Document incidents on a large board in the lab

B. Purchase and deploy an incident management tool

C. Manage the incidents through E-mails and phone calls

D. Manage the incidents in a spreadsheet posted on the intranet

**Answer:** D

**Explanation:**

An incident is any event that occurs during testing that requires investigation. An incident management tool is a software tool that supports recording and tracking incidents throughout their life cycle. A QA manager of a start-up company needs to implement within a week a low cost incident management tool. The best option for this case is to manage the incidents in a spreadsheet posted on the intranet. This option has several advantages over other options:

? It is low cost, as it does not require purchasing any additional software or hardware.

? It is easy to implement within a week, as it does not require installing or configuring any complex software or hardware.

? It is accessible and transparent, as it can be viewed and updated by anyone who has access to the intranet.

? It is structured and organized, as it can store and display various information about incidents, such as identifier, summary, description, severity, priority, status, resolution, etc. The other options are not suitable for this case, as they have several disadvantages over the chosen option:

? Documenting incidents on a large board in the lab is not a good option, as it is not accessible or transparent to anyone who is not physically present in the lab. It is also not structured or organized, as it may not store or display all the necessary information about incidents.

? Purchasing and deploying an incident management tool is not a good option, as it is not low cost or easy to implement within a week. It may require spending a significant amount of money and time on acquiring, installing and configuring the software or hardware.

? Managing the incidents through emails and phone calls is not a good option, as it is not structured or organized. It may lead to confusion, inconsistency or loss of information about incidents. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 3, page 32-33.

**NEW QUESTION 188**

Which of the following is an example of tasks most associated with the test design activity?

A. The project manager updates the project schedule as key test tasks are completed

B. Test data, derived from production data, is loaded into the test environment

C. Every day

D. the tester notes the status of his/her test cases in preparation for daily reports

E. The identification of test execution and test automation tools

**Answer:** D

**Explanation:**

Test design is one of the fundamental activities in software testing, focusing on how tests will be structured and what strategies will be employed. As per the ISTQB syllabus, identifying appropriate test execution and automation tools is a task closely associated with test design activities because it directly influences how tests will be conducted, managed, and executed efficiently. This activity involves deciding on the tools that will best support the testing goals and requirements. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.4 "Test Design Techniques".

**NEW QUESTION 193**

In what way do Configuration Management effects testing?

A. Without proper configuration management, test planning cannot proceed.

B. Proper configuration management ensures that testers can uniquely identify the tested item

C. Configuration management is important for developers, not for testers

D. There is very little influence of configuration management practices on the test project.

**Answer:** B

**Explanation:**

Configuration management is a process that establishes and maintains consistency among work products throughout their life cycle. Configuration management affects testing in various ways, such as:

? Proper configuration management ensures that testers can uniquely identify the tested item, which can help traceability, reproducibility and accountability.

? Proper configuration management ensures that testers have access to consistent versions of software components and testware, which can help reliability, compatibility and efficiency.

? Proper configuration management ensures that testers can track changes and defects in software components and testware, which can help verification, validation and reporting.

? Proper configuration management ensures that testers can control the configuration of the test environment, which can help stability, security and performance.

Configuration management is not a prerequisite for test planning, as test planning can proceed without configuration management, although it may be less effective or accurate. Configuration management is not important for developers only, but for testers as well, as it affects the quality and consistency of the testing process and products. Configuration management has a significant influence on the test project, as it affects various aspects of testing, such as traceability, reproducibility, reliability, compatibility, efficiency, verification, validation, reporting, stability, security and performance. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 6, page 60-61.



#### NEW QUESTION 197

When should component integration tests be carried out?

- A. Integration tests should always be done after system tests
- B. Integration tests should be done at the customer's site, after acceptance tests
- C. Integration tests can be done before or after system tests
- D. Integration tests should always be done before system tests

**Answer: D**

#### Explanation:

Component integration tests are designed to verify the interactions and interfaces between integrated components. These tests should be carried out after component testing (where individual components are tested in isolation) but before system testing (where the entire system is tested as a whole). This ensures that any issues arising from the integration of components are identified and resolved early in the testing process, making option D the correct answer.

#### NEW QUESTION 198

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 202

Which of the following does MOT describe a reason why testing is necessary?

- A. The customer decided that 100% branch coverage shall be achieved
- B. The acquisition of test automation tools was based on the assumption that it will be used in all projects
- C. For avionics and pharmaceutical systems software testing is mandated by standards
- D. The risks associated with delivering the system are far higher than the cost of testing

**Answer: B**

#### Explanation:

Testing is necessary for various reasons, such as:

- ? To detect defects and failures that may affect the quality, performance, reliability or security of a software product or system
  - ? To verify that a software product or system meets its specified requirements, expectations and standards
  - ? To validate that a software product or system fulfills its intended purpose and satisfies its stakeholders' needs
  - ? To provide information and feedback about the status and risks of a software product or system
  - ? To comply with regulations or contractual obligations that mandate testing for certain types of software products or systems
- The following statements describe some reasons why testing is necessary:
- ? A) The customer decided that 100% branch coverage shall be achieved. This is a reason why testing is necessary, as it reflects a contractual obligation or a quality standard that requires testing to measure and achieve a certain level of code coverage.
  - ? C) For avionics and pharmaceutical systems software testing is mandated by standards. This is a reason why testing is necessary, as it reflects a regulation or a compliance requirement that mandates testing for certain types of software products or systems that have high safety or security risks.
  - ? D) The risks associated with delivering the system are far higher than the cost of testing. This is a reason why testing is necessary, as it reflects a risk-based approach that considers testing as an investment to reduce the probability and impact of potential failures or defects. The following statement does not describe a reason why testing is necessary:
  - ? B) The acquisition of test automation tools was based on the assumption that it will be used in all projects. This is not a reason why testing is necessary, as it reflects a business decision or a resource allocation that does not justify the need or purpose of testing. Test automation tools are not always suitable or beneficial for all projects, and testing can be performed with or without test automation tools. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 1, page 5-6.

#### NEW QUESTION 204

Which of the following is the most correct statement about state testing techniques?

- A. Static techniques can be used before all code is ready for execution
- B. Static techniques find more defects than dynamic techniques.
- C. Static techniques can be used by inexperienced users.
- D. Static techniques are always cheaper than dynamic techniques.

**Answer: A**

#### Explanation:

State testing techniques are a type of dynamic testing techniques that are based on the behavior of the system under test for different input conditions and events. Dynamic testing techniques require the system to be executed with test cases, whereas static testing techniques do not. Static testing techniques can be applied before the code is ready for execution, such as reviews, inspections, walkthroughs, and static analysis. Static testing techniques can help find defects early in the development process, improve the quality of the code, and reduce the cost and effort of dynamic testing. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 4, Section 4.2.1, Page 281; ISTQB Glossary of Testing Terms v4.0, Page 292

#### NEW QUESTION 207

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 209

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 212

Which of the following are valid testing principles?

I) Exhaustive testing is in general impossible.

II) Exhaustive testing should be executed for code intended to be reused.

III) Testing may guarantee that a program is correct.

IV) Testing cannot guarantee that a program is correct.

V) Defects cluster together in certain areas of the product.

A. I, IV, V

B. II, IV

C. I, V

D. I, III

**Answer: A**

#### Explanation:

Statements I, IV and V are valid testing principles according to the ISTQB syllabus. Statement I states that exhaustive testing is in general impossible, because it would require testing all possible inputs, outputs and combinations of states, which is usually impractical or impossible. Statement IV states that testing cannot guarantee that a program is correct, because testing can only show the presence of defects, not their absence. Statement V states that defects cluster together in certain areas of the product, which means that some modules or functions are more likely to contain defects than others. Statements II and III are invalid testing principles. Statement II states that exhaustive testing should be executed for code intended to be reused, which contradicts statement I. Statement III states that testing may guarantee that a program is correct, which contradicts statement IV. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, pages 4-5.

#### NEW QUESTION 213

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 218**

Software was found to take much more time than the stated requirement of less than one second to save a file. Upon investigation it was found that there was an unnecessary check inside a loop which was slowing down the file-save operation. The software not being able to meet the desired response time is an example of

- A. It is not a defect
- B. Defect
- C. Error
- D. Failure

**Answer:** D

**Explanation:**

A failure is an event in which a component or system does not perform a required function within specified limits. A failure is observable by the software users or other stakeholders. A failure is caused by one or more defects in the software. In this case, the software not being able to meet the desired response time is an example of a failure, as it deviates from the stated requirement and affects the user experience. It is not a defect, which is a flaw in the software that causes the failure. It is not an error, which is a human action that produces an incorrect result. It is not a non-defect, as it clearly violates a specified requirement. Verified

References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 1, page 4.

**NEW QUESTION 221**

.....

## Relate Links

**100% Pass Your ISTQB-CTFL Exam with Exambible Prep Materials**

<https://www.exambible.com/ISTQB-CTFL-exam/>

## Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Viste - <https://www.exambible.com/>