



Free Questions for ISTQB-CTFL

Shared by Craig on 02-08-2023

For More Free Questions and Preparation Resources

[Check the Links on Last Page](#)



Question 1

Question Type: MultipleChoice

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

Options:

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

P2P
exams

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.

Question 2

Question Type: MultipleChoice

When should component integration tests be carried out?

Options:

- 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

P2P
exams

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.

P2P
exams

Question 3

Question Type: MultipleChoice

Which of the following sentences describe a product risk?

Options:

- 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

P2P
exams

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.

Question 4

Question Type: MultipleChoice

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

Options:

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

Answer:

C

Explanation:

Decision coverage, also known as branch coverage, is measured as the number of decision outcomes executed by the tests divided by the total number of decision outcomes in the test object. It ensures that every possible branch (true/false) decision in the code has been executed at least once. Reference: ISTQB CTFL Syllabus V4.0, Section 4.3.2

Question 5

Question Type: MultipleChoice

A program is used to control a manufacturing line (turn machines on and off. start and stop conveyer belts, add raw materials to the flow. etc.). Not all actions are possible at all times. For example, there are certain manufacturing stages that cannot be stopped - unless there is an

emergency. A tester attempts to evaluate if all such cases (where a specific action is not allowed) are covered by the tests.

Which coverage metric will provide the needed information for this analysis?

Options:

- A- Code coverage
- B- Data flow coverage
- C- Statement coverage
- D- Branch Coverage

Answer:

D

Explanation:

Branch coverage is a type of structural coverage metric that measures the percentage of branches or decision outcomes that are executed by the test cases. A branch is a point in the code where the control flow can take two or more alternative paths based on a condition. For example, an if-else statement is a branch that can execute either the if-block or the else-block depending on the evaluation of the condition. Branch coverage ensures that each branch is taken at least once by the test cases, and thus reveals the behavior of the software under different scenarios. Branch coverage is also known as decision coverage or all-edges coverage.

Branch coverage is suitable for testing the cases where a specific action is not allowed, because it can verify that the test cases cover all the possible outcomes of the conditions that determine the action. For example, if the program has a condition that checks if the manufacturing stage can be stopped, then branch coverage can ensure that the test cases cover both the cases where the stage can be stopped and where it cannot be stopped. This way, branch coverage can help identify any missing or incorrect branches that may lead to undesired or unsafe actions.

The other options are not correct because they are not suitable for testing the cases where a specific action is not allowed. Code coverage is a general term that encompasses various types of coverage metrics, such as statement coverage, branch coverage, data flow coverage, etc. Code coverage does not specify which type of coverage metric is used for the analysis. Data flow coverage is a type of structural coverage metric that measures the percentage of data flow paths that are executed by the test cases. A data flow path is a sequence of statements that define, use, or kill a variable. Data flow coverage is useful for testing the correctness and completeness of the data manipulation in the software, but not for testing the conditions that determine the actions. Statement coverage is a type of structural coverage metric that measures the percentage of statements or lines of code that are executed by the test cases. Statement coverage ensures that each statement is executed at least once by the test cases, but it does not

reveal the behavior of the software under different scenarios. Statement coverage is a weaker criterion than branch coverage, because it does not account for the branches or decision outcomes in the code. Reference= ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, Chapter 4: Test Techniques, Section 4.3: Structural Testing Techniques, Pages 51-54.

Question 6

Question Type: MultipleChoice

Which of the following definitions is NOT true?



Options:

- 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.

Question 7

Question Type: MultipleChoice

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

Options:

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

Answer:

D

Explanation:

While configuration management is crucial for maintaining consistent versions of software artifacts and supporting the build process, it does not directly increase the chances of finding defects in the software under test. Version-controlled test ware ensures that the correct versions of test cases and other test artifacts are used, but it is the quality and thoroughness of the tests that determine the effectiveness of defect detection. Reference: ISTQB CTFL Syllabus V4.0, Section 5.4

To Get Premium Files for ISTQB-CTFL Visit

<https://www.p2pexams.com/products/istqb-ctfl>

For More Free Questions Visit

<https://www.p2pexams.com/istqb/pdf/istqb-ctfl>

20%
DISCOUNT

P2P
exams