Vendor: ISTQB

Exam Code: ISTQB-CTFL

Exam Name: ISTQB-Foundation Level Exam

Version: DEMO

QUESTION 1
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.

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.

QUESTION 3
Which ONE of the following statements about acceptance testing is NOT correct?

A. Testing of disaster recovery and backup/restore is usually NOT part of acceptance testing.
B. The customers or system users are often responsible for the acceptance testing.
C. The main goal of acceptance testing is to build confidence in the system, not find defects.
D. Acceptance testing is the last level of testing performed prior to system release.

Answer: A
Explanation:
Acceptance testing is a level of testing performed to verify that a software product meets the agreed acceptance criteria and is acceptable for delivery. Acceptance testing is often performed

by the customers or system users, who are the main stakeholders of the software product. The main goal of acceptance testing is to build confidence in the system, not find defects, as defects should have been detected and fixed in earlier levels of testing. Acceptance testing is the last level of testing performed prior to system release, unless there are any changes or fixes that require re-testing. Testing of disaster recovery and backup/restore is usually part of acceptance testing, as these are important aspects of system reliability and security that affect the customer satisfaction and trust. Therefore, statement A is not correct, while statements B, C and D are correct.

QUESTION 4
Which are the MAIN goals of risk management in a software project?

A. To increase the success probability for the project regardless of costs.
B. To increase focus on preventative processes and to increase satisfaction for the testers.
C. To control contractual problems and minimize the impacts of company policies.
D. To reduce the probability of undesired situations and to reduce the effect of potential impact.

Answer: D
Explanation:
Risk management is a process that identifies, analyzes, evaluates and mitigates risks in a software project. Risks are factors that may negatively affect the quality, schedule, budget or scope of a software project. The main goals of risk management in a software project are to reduce the probability of undesired situations and to reduce the effect of potential impact. This can be achieved by applying various strategies, such as avoidance, transfer, reduction or acceptance. Risk management does not aim to increase the success probability for the project regardless of costs, as this may not be feasible or realistic. Risk management does not aim to increase focus on preventative processes and to increase satisfaction for the testers, as these are secondary or indirect outcomes. Risk management does not aim to control contractual problems and minimize the impacts of company policies, as these are specific types of risks that may not apply to all projects.

QUESTION 5
A team's test strategy was to invest equal effort in testing each of a system's modules. After running one test cycle, it turned out that most of the critical bugs were detected in one of the system's modules.

Which testing principal suggests a change to the current test strategy for the next test cycle?

A. Pesticide Paradox
B. Early testing
C. Absence-of-errors fallacy
D. Defect clustering

Answer: D
Explanation:
Defect clustering is a testing principle that states that a small number of modules contain most of the defects detected during pre-release testing, or are responsible for most of the operational failures. Defect clustering can be explained by Pareto's principle (also known as the 80-20 rule), which states that approximately 80% of the problems are found in 20% of the modules. Defect clustering suggests a change to the current test strategy for the next test cycle, as it implies that more effort should be allocated to test the modules that have shown high defect density or criticality. Pesticide paradox is another testing principle that states that if the same tests are repeated over and over again, eventually they will no longer find any new defects. Pesticide

paradox suggests a change to the current test strategy for the next test cycle, but not based on defect clustering, but rather on test diversity and coverage. Early testing is a testing principle that states that testing activities should start as early as possible in the software development life cycle and should be focused on defined objectives. Early testing does not suggest a change to the current test strategy for the next test cycle, but rather a proactive approach to prevent defects from occurring or propagating. Absence-of-errors fallacy is a testing principle that states that finding and fixing defects does not help if the system built is unusable and does not fulfill the users' needs and expectations. Absence-of-errors fallacy does not suggest a change to the current test strategy for the next test cycle, but rather a focus on quality attributes and user requirements.

QUESTION 6
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 identity 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.

QUESTION 7
During system testing phase of a word processor, a tester finds that on opening a file from a particular set of files, which are part of a critical workflow, the word processor crashes. Which of the following is the next step the tester should take poor to recording the deviation?

A. Try to recreate the incident before reporting
B. Try to identify the code fragment causing the problem
C. Send an email to the developer and not report the bug
D. Report the incident as is without any further action

Answer: A
Explanation:
An incident is any event that occurs during testing that requires investigation. An incident report is a document that records the details of an incident. The next step the tester should take prior to recording the deviation is to try to recreate the incident before reporting. This can help confirm that the incident is reproducible and not caused by a random or external factor. This can also help gather more information about the incident, such as the steps to reproduce it, the expected and actual results, the severity and priority of the incident, or any screenshots or logs that can illustrate the incident. Trying to identify the code fragment causing the problem is not the next step the tester should take prior to recording the deviation, as this is a debugging activity that is usually performed by developers after receiving the incident report. Sending an email to the developer and not reporting the bug is not the next step the tester should take prior to recording the deviation, as this is an informal and unstructured way of communicating incidents that can lead to confusion, inconsistency or loss of information. Reporting the incident as is without any further action is not the next step the tester should take prior to recording the deviation, as this can result in incomplete or inaccurate incident reports that can hamper the investigation and resolution of incidents.

QUESTION 8
Which of the following statements about decision tables are TRUE?

I) Generally, decision tables are generated for low risk test items.
II) Test cases derived from decision tables can be used for component tests.
III) Several test cases can be selected for each column of the decision table.
IV) The conditions in the decision table represent negative tests generally.

A. I. Ill
B. I, IV
C. II. IV
D. II. Ill

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

II. Test cases derived from decision tables can be used for component tests. Decision tables can be used to test components that have multiple inputs and outputs that depend on logical combinations of conditions. Decision tables can help cover all possible combinations or scenarios in a systematic way.

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

I. Generally, decision tables are generated for low risk test items. Decision tables are not related to risk level, but rather to complexity level. Decision tables are generated for test items that have complex logic or multiple conditions and actions that need to be tested.

IV. The conditions in the decision table represent negative tests generally. The conditions in the decision table represent both positive and negative tests, depending on whether they are valid or invalid inputs for the test item.

QUESTION 9
Consider the following statements about risk-based testing.

I) Risk-based testing has the objective to reduce the level of protect risks.
II) Tests should be prioritized to find tie critical detects as early as possible.
III) Non-testing activities may also help to reduce risk
IV) Risks have to be reassessed on a regular basis.
V) The project stakeholders can give useful input to determine the risks

A. I III IV and V are true. II is false.
B. II, III IV and V are correct. I is false.
C. I, II and IV are true. III and V are false.
D. II, Ill and V are true. 1 ants Iv are false.

Answer: B
Explanation:
The following statements about risk-based testing are correct:

II) Tests should be prioritized to find tie critical detects as early as possible. Risk-based testing involves prioritizing tests based on risk level, which reflects both the likelihood and impact of defects or failures. Tests with higher risk level should be executed earlier than tests with lower risk level, in order to find and fix critical defects as soon as possible.

III) Non-testing activities may also help to reduce risk. Risk-based testing does not only involve testing activities, but also other activities that can help mitigate risks, such as reviews, inspections, audits, simulations or prototyping.

QUESTION 10
As the last stage of a test cycle of an embedded device, you are performing exploratory testing. You observed that some character. (A, X and Z) sent via a serial port to the device do not get registered on the device whereas they should be. You suspect that this could be due to a wrong configuration of the "bit parity" parameter.

Which of the following items of an incident report would you be UNABLE to write down based on this information?

A. Expected result
B. Test case identifier
C. Test setup details
D. Actual result

Answer: B
Explanation:
An incident report is a document that records the details of an incident. An incident report typically contains the following items:

Identifier: A unique identifier for the incident report
Summary: A concise summary of the incident
Description: A detailed description of the incident, including the steps to reproduce it, the expected and actual results, and any relevant screenshots or logs
Severity: The degree of impact that the incident has on the system

Priority: The level of urgency for resolving the incident
Status: The current state of the incident, such as new, open, resolved, closed, etc.
Resolution: The action taken to resolve the incident, such as fix, workaround, reject, etc. Based on the information given in the question, the tester would be able to write down all of these items except for the test case identifier. A test case identifier is a unique identifier for a test case that is used to link it to other test artifacts, such as test plans, test scripts, test results or incident reports. However, since the tester is performing exploratory testing, there is no predefined test case that can be associated with the incident. 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. Exploratory testing does not use formal test cases or scripts, but rather uses test charters or missions that guide the tester's actions and objectives.

QUESTION 11
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.
V) Re-testing should be included in the debugging activity.

A. I, III. Iv
B. II, v
C. I, Ill
D. II. 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.