hillel
компьютерная школа

ISTQB

Preparation to ISTQB Foundation Level Certification Exam

Test Management

By Vladimir Arutin

Keywords

Configuration management- A discipline applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements.

Defect management The process of recognizing and recording defects, classifying them, investigating them, taking action to resolve them, and disposing of them when resolved.

Entry criteria (definition of ready) The set of conditions for officially starting a defined task.

Exit criteria (completion criteria , test completion criteria , definition of done) - The set of conditions for officially completing a defined task.

Product risk A risk impacting the quality of a product.

Project risk A risk that impacts project success

.

Risk A factor that could result in future negative consequences.

Risk level (risk exposure) - The qualitative or quantitative measure of a risk defined by impact and likelihood.

Risk-based testing- Testing in which the management, selection, prioritization, and use of testing activities and resources are based on corresponding risk types and risk levels.

Test approach The implementation of the test strategy for a specific project.

Test control A test management task that deals with developing and applying a set of corrective actions to get a test project on track when monitoring shows a deviation from what was planned.

Test estimation The calculated approximation of a result related to various aspects of testing (e.g., effort spent, completion date, costs involved, number of test cases, etc.) which is usable even if input data may be incomplete, uncertain, or noisy.

Test manager The person responsible for project management of testing activities and resources, and evaluation of a test object. The individual who directs, controls, administers, plans and regulates the evaluation of a test object.

Test monitoring A test management activity that involves checking the status of testing activities, identifying any variances from the planned or expected status, and reporting status to stakeholders.

Test plan - Documentation describing the test objectives to be achieved and the means and the schedule for achieving them, organized to coordinate testing activities.

Test planning The activity of establishing or updating a test plan.

Test progress report (test status report) A test report produced at regular intervals about the progress of test activities against a baseline, risks, and alternatives requiring a decision.

Test strategy (organizational test strategy) Documentation that expresses the generic requirements for testing one or more projects run within an organization, providing detail on how testing is to be performed, and is aligned with the test policy.

Test summary report (test report) A test report that provides an evaluation of the corresponding test items against exit criteria.

Tester A skilled professional who is involved in the testing of a component or system.

Test Organization

Independent Testing

A certain degree of independence often makes the tester more effective at finding defects due to differences between the author's and the tester's cognitive biases.

Degrees of independence in testing include the following (from low level of independence to high level):

- No independent testers; the only form of testing available is developers testing their own code
- Independent developers or testers within the development teams or the project team; this could be developers testing their colleagues' products
- Independent test team or group within the organization, reporting to project management or executive management
- Independent testers from the business organization or user community, or with specializations in specific test types such as usability, security, performance, regulatory/compliance, or portability
- Independent testers external to the organization, either working on-site (insourcing) or off-site (outsourcing)

Potential benefits:

- Independent testers are likely to recognize different kinds of failures compared to developers **because of their different backgrounds, technical perspectives, and biases**
- An independent tester **can verify, challenge, or disprove assumptions** made by stakeholders during specification and implementation of the system

Potential drawbacks:

- **Isolation** from the development team, **leading to a lack of collaboration**, delays in providing feedback to the development team, or an adversarial relationship with the development team
- **Developers** may **lose a sense of responsibility** for quality
- Independent testers may be seen as **a bottleneck or blamed for delays in release**
- Independent testers may **lack some important information** (e.g., about the test object)

Tasks of a Test Manager and Tester

Test Manager is responsible for the test process and successful leadership of the test activities.

The test management role might be performed by a **professional test manager**, or by a **project manager**, a **development manager**, or a **quality assurance manager**.

In larger projects or organizations, **several test teams may report to** a test manager, test coach, or test coordinator, each team being headed by a test leader or lead tester.

Typical test manager tasks may include:

- ❑ Develop or review a test policy and test strategy for the organization
- ❑ Plan the test activities by considering the context, and understanding the test objectives and Risks (selecting test approaches, estimating test time, effort and cost, acquiring resources, defining test levels and test cycles, and planning defect management)
- ❑ Write and update the test plan(s)
- ❑ Coordinate the test plan(s) with project managers, product owners, and others
- ❑ Share testing perspectives with other project activities, such as integration planning
- ❑ Initiate the analysis, design, implementation, and execution of tests, monitor test progress and results, and check the status of exit criteria (or definition of done)
- ❑ Prepare and deliver test progress reports and test summary reports based on the information gathered.
- ❑ Adapt planning based on test results and progress and take any actions necessary for test control
- ❑ Support setting up the defect management system and adequate configuration management of testware
- ❑ Introduce suitable metrics for measuring test progress and evaluating the quality of the testing and the product
- ❑ Support the selection and implementation of tools to support the test process, allocating time and effort for pilot projects, and providing continuing support in the use of the tool(s)
- ❑ Decide about the implementation of test environment(s)
- ❑ Promote and advocate the testers, the test team, and the test profession within the organization
- ❑ Develop the skills and careers of testers

Typical tester tasks may include:

- Review and contribute to test plans
- Analyze, review, and assess requirements, user stories and acceptance criteria, specifications, and models for testability (i.e., the test basis)
- Identify and document test conditions, and capture traceability between test cases, test conditions, and the test basis
- Design, set up, and verify test environment(s), often coordinating with system administration and network management
- Design and implement test cases and test procedures
- Prepare and acquire test data
- Create the detailed test execution schedule
- Execute tests, evaluate the results, and document deviations from expected results
- Use appropriate tools to facilitate the test process
- Automate tests as needed (may be supported by a developer or a test automation expert)
- Evaluate non-functional characteristics such as performance efficiency, reliability, usability, security, compatibility, and portability
- Review tests developed by others

Depending on the **risks** related to the product and the project, and the **software development lifecycle model** selected, **different people** may take over the role of tester at different test levels.

At the **component testing level** and the component integration testing level, the role of a tester is
often done **by developers**.

At the **acceptance test level**, the role of a tester is often done **by business** analysts, subject matter experts, and users.

At the **system test level** and **the system integration test level**, the role of a tester is often done **by an independent test team**.

At the **operational acceptance test leve**l, the role of a tester is often done **by operations and/or systems administration staff.**

Purpose and Content of a Test Plan

A **test plan outlines test activities** for development and maintenance projects.

Planning is influenced by:

- the **test policy** and **test strategy** of the organization
- the **development lifecycles and methods being used**
- the **scope of testing**,
- **objectives,**
- **risks,**
- **constraints,**
- **criticality,**
- **testability,**
- **the availability of resources.**

Test planning **is a continuous activity and is performed throughout the product's lifecycle** (including maintenance phase.)

Feedback from test activities should be used to recognize changing risks so that planning can be adjusted.

Planning may be documented in a **master test plan** and in **separate test plans for test levels**, such as system testing and acceptance testing, or for separate test types, such as usability testing and performance testing.

Test planning activities may include the following(some may be documented in a test plan):

- Determining the scope, objectives, and risks of testing
- Defining the overall approach of testing
- Integrating and coordinating the test activities into the software lifecycle activities
- Making decisions about what to test, the people and other resources required to perform the various test activities, and how test activities will be carried out
- Scheduling of test analysis, design, implementation, execution, and evaluation activities, either on particular dates (e.g., in sequential development) or in the context of each iteration (e.g., in iterative development)
- Selecting metrics for test monitoring and control
- Budgeting for the test activities
- Determining the level of detail and structure for test documentation (e.g., by providing templates or example documents)

Test Strategy and Test Approach

Analytical: This type of test strategy is based on an analysis of some factor (e.g., requirement or risk). **Risk-based testing** is an example of an analytical approach, where tests are designed and prioritized based on the level of risk.

Model-Based: In this type of test strategy, tests are designed based on some model of some required aspect of the product, such as a function, a business process, an internal structure, or a non-functional characteristic (e.g., reliability). Examples of such models include **business process models, state models, and reliability growth models.**

Methodical: This type of test strategy relies on **making systematic use of some predefined set of tests or test conditions**, such as a taxonomy of common or likely types of failures, a list of important quality characteristics, or company-wide look-and-feel standards for mobile apps or web pages.

Process-compliant (or standard-compliant): This type of test strategy involves analyzing, designing, and implementing tests based on **external rules and standards**, such as those specified by industry-specific standards, by process documentation, by the rigorous identification and use of the test basis, or by any process or standard imposed on or by the organization.

Directed (or consultative): This type of test strategy **is driven primarily by the advice, guidance, or instructions** of stakeholders, business domain experts, or technology experts, who may be outside the test team or outside the organization itself.

Test Strategy and Test Approach

Regression-averse: This type of test strategy is motivated by a **desire to avoid regression of existing capabilities**. This test strategy includes reuse of existing testware (especially test cases and test data), extensive automation of regression tests, and standard test suites.

Reactive: In this type of test strategy, **testing is reactive to the component or system being tested, and the events occurring during test execution, rather than being pre-planned** (as the preceding strategies are). Tests are designed and implemented, and may immediately be executed in response to knowledge gained from prior test results. Exploratory testing is a common technique employed in reactive strategies.

An appropriate test strategy is often created **by combining several of these types of test strategies.** For example, risk-based testing (an analytical strategy) can be combined with exploratory testing (a reactive strategy).

The selected approach depends on the context and may consider factors such as **risks, safety, available resources and skills, technology, the nature of the system (e.g., custom-built versus COTS), test objectives, and regulations.**

Entry Criteria and Exit Criteria (Definition of Ready and Definition of Done)

In order to exercise effective control over the quality of the software, and of the testing, **it is advisable to have criteria** which define **when a given test activity should start** and **when the activity is complete**.

Entry criteria (more typically called definition of ready in Agile development) define the preconditions for undertaking a given test activity. If entry criteria are not met, it is likely that the activity will prove more difficult, more time-consuming, more costly, and more risky.

Exit criteria (more typically called definition of done in Agile development) define what conditions must be achieved in order to declare a test level or a set of tests completed.

Entry and exit criteria should be defined for each test level and test type, and will differ based on the test objectives.

Typical entry criteria include:

- Availability of testable requirements, user stories, and/or models (e.g., when following a model based testing strategy)
- Availability of test items that have met the exit criteria for any prior test levels
- Availability of test environment
- Availability of necessary test tools
- Availability of test data and other necessary resources

Typical exit criteria include:

- Planned tests have been executed
- A defined level of coverage
  (e.g., of requirements, user stories, acceptance criteria, risks, code) has been achieved
- The number of unresolved defects is within an agreed limit
- The number of estimated remaining defects is sufficiently low
- The evaluated levels of reliability, performance efficiency, usability, security, and other relevant quality characteristics are sufficient

Even without exit criteria being satisfied, it is also common for test activities to be curtailed due to the budget being expended, the scheduled time being completed, and/or pressure to bring the product to market.

It can be acceptable to end testing under such circumstances, if the project stakeholders and business owners have reviewed and accepted the risk to go live without further testing.

Test Execution Schedule

Once the various test cases and test procedures are produced (with some test procedures potentially automated) and assembled into test suites, the test suites can be arranged in a **test execution schedule** that **defines the order in which they are to be run.**

The **test execution schedule** should take into account such factors as:

1. **prioritization**
2. **dependencies**
3. **confirmation tests**
4. **regression tests**
5. **the most efficient sequence for executing the tests.**

If a test case with a higher priority is dependent on a test case with a lower priority, the lower priority test case must be executed first.
Similarly, if there are dependencies across test cases, they must be ordered appropriately regardless of their relative priorities.
Confirmation and regression tests must be prioritized as well, based on the importance of rapid feedback on changes, but here again dependencies may apply.

Factors Influencing the Test Effort

Product characteristics

- ❑ The risks associated with the product
- ❑ The quality of the test basis
- ❑ The size of the product
- ❑ The complexity of the product domain
- ❑ The requirements for quality characteristics (e.g., security, reliability)
- ❑ The required level of detail for test documentation
- ❑ Requirements for legal and regulatory compliance

Development process characteristics

- ❑ The stability and maturity of the organization
- ❑ The development model in use
- ❑ The test approach
- ❑ The tools used
- ❑ The test process
- ❑ Time pressure

Factors Influencing the Test Effort

People characteristics

- The skills and experience of the people involved, especially with similar projects and products(e.g., domain knowledge)
- Team cohesion and leadership

Test results

- The number and severity of defects found
- The amount of rework required

Test Estimation Techniques

- The metrics-based technique: estimating the test effort based on metrics of former similar projects, or based on typical values (burndown charts, defect removal models)

- The expert-based technique: estimating the test effort based on the experience of the owners of the testing tasks or by experts (planning poker, Wideband Delphi)

Test Monitoring and Control

The purpose of **test monitoring** is **to gather information and provide feedback and visibility** about test
activities.

Test control describes **any guiding or corrective actions** taken as a result of information and metrics
gathered and (possibly) reported.

Examples of test control actions include:

- Re-prioritizing tests when an identified risk occurs (e.g., software delivered late)
- Changing the test schedule due to availability or unavailability of a test environment or other resources
- Re-evaluating whether a test item meets an entry or exit criterion due to rework

Metrics Used in Testing

Metrics can be collected during and at the end of test activities in order to assess:

- Progress against the planned schedule and budget
- Current quality of the test object
- Adequacy of the test approach
- Effectiveness of the test activities with respect to the objectives

Metrics Used in Testing

Common test metrics include:

- Percentage of planned work done in test case preparation (or percentage of planned test cases implemented)
- Percentage of planned work done in test environment preparation
- Test case execution (e.g., number of test cases run/not run, test cases passed/failed, and/or test conditions passed/failed)
- Defect information (e.g., defect density, defects found and fixed, failure rate, and confirmation test results)
- Test coverage of requirements, user stories, acceptance criteria, risks, or code
- Task completion, resource allocation and usage, and effort
- Cost of testing, including the cost compared to the benefit of finding the next defect or the cost compared to the benefit of running the next test

Purposes, Contents, and Audiences for Test Reports

The purpose of test reporting is **to summarize and communicate test activity information**, both during and at the end of a test activity (e.g., a test level).

The test report prepared during a test activity may be referred to as a **test progress report**, while a test report prepared at the end of a test activity may be referred to as a **test summary report.**

During test monitoring and control, the test manager regularly issues test progress reports for stakeholders. Typical test progress reports may also include:

- The status of the test activities and progress against the test plan
- Factors impeding progress
- Testing planned for the next reporting period
- The quality of the test object

Purposes, Contents, and Audiences for Test Reports

When exit criteria are reached, the test manager issues the **test summary report**.

Typical test progress reports and test summary reports may include:

- Summary of testing performed
- Information on what occurred during a test period
- Deviations from plan, including deviations in schedule, duration, or effort of test activities
- Status of testing and product quality with respect to the exit criteria or definition of done
- Factors that have blocked or continue to block progress
- Metrics of defects, test cases, test coverage, activity progress, and resource consumption.
- Residual risks
- Reusable test work products produced

Purposes, Contents, and Audiences for Test Reports

The contents of a test report will vary depending on the project, the organizational requirements, and the software development lifecycle.

A **complex project** with many stakeholders or a regulated project may require **more detailed and rigorous reporting** than a quick software update.

In **Agile** development, test progress reporting may be incorporated into task boards, defect summaries, and burndown charts, may be discussed during a **daily stand-up meeting.**

Configuration Management

The purpose of configuration management is to **establish and maintain the integrity** of the component or system, the testware, and their relationships to one another through the project and product lifecycle.

To properly support testing, configuration management may involve ensuring the following:

- All test items are uniquely identified, version controlled, tracked for changes, and related to each other
- All items of testware are uniquely identified, version controlled, tracked for changes, related to each other and related to versions of the test item(s) so that traceability can be maintained throughout the test process
- All identified documents and software items are referenced unambiguously in test documentation
- During test planning, configuration management procedures and infrastructure (tools) should be identified and implemented.

Risks and Testing

Risk involves the possibility of an event in the future which has negative consequences.
The level of risk is determined by the **likelihood** and the **impact (the harm)** from event.

Product Risks

Product risk involves the possibility that a work product (e.g., a specification, component, system, or test) may fail to satisfy the legitimate needs of its users and/or stakeholders.

When the **product risks are associated with specific quality characteristics** of a product (e.g., *functional suitability, reliability, performance efficiency, usability, security, compatibility, maintainability, and portability*), product risks are also called **quality risks**.

Risks and Testing

Product risks:

- Software might not perform its intended functions according to the specification
- Software might not perform its intended functions according to user, customer, and/or stakeholder needs
- A system architecture may not adequately support some non-functional requirement(s)
- A particular computation may be performed incorrectly in some circumstances
- A loop control structure may be coded incorrectly
- Response-times may be inadequate for a high-performance transaction processing system
- User experience (UX) feedback might not meet product expectations

Project Risks

Project risk involves situations that, should they occur, may have a negative effect on a project's ability to achieve its objectives.

Project risks include:

- **Project issues:**
  - ✓ Delays may occur in delivery, task completion, or satisfaction of exit criteria or definition of done
  - ✓ Inaccurate estimates, reallocation of funds to higher priority projects, or general cost cutting across the organization may result in inadequate funding
  - ✓ Late changes may result in substantial re-work

- **Organizational issues:**
  - ✓ Skills, training, and staff may not be sufficient
  - ✓ Personnel issues may cause conflict and problems
  - ✓ Users, business staff, or subject matter experts may not be available due to conflicting business priorities

Project Risks

- **Political issues:**
  - ✓ Testers may not communicate their needs and/or the test results adequately
  - ✓ Developers and/or testers may fail to follow up on information found in testing and reviews (e.g., not improving development and testing practices)
  - ✓ There may be an improper attitude toward, or expectations of, testing (e.g., not appreciating the value of finding defects during testing)

- **Technical issues**:
  - ✓ Requirements may not be defined well enough
  - ✓ The requirements may not be met, given existing constraints
  - ✓ The test environment may not be ready on time
  - ✓ Data conversion, migration planning, and their tool support may be late
  - ✓ Weaknesses in the development process may impact the consistency or quality of project work products such as design, code, configuration, test data, and test cases
  - ✓ Poor defect management and similar problems may result in accumulated defects and other technical debt

- **Supplier issues:**
  - ✓ A third party may fail to deliver a necessary product or service, or go bankrupt
  - ✓ Contractual issues may cause problems to the project

Risk-based Testing and Product Quality

A **risk-based approach** to testing provides proactive opportunities **to reduce the levels of product risk.** It involves product risk analysis, which includes the **identification** of product risks and the **assessment of each risk's likelihood and impact**

In a risk-based approach, the results of product risk analysis are used to:

- Determine the test techniques to be employed
- Determine the particular levels and types of testing to be performed (e.g., security testing, accessibility testing)
- Determine the extent of testing to be carried out
- Prioritize testing in an attempt to find the critical defects as early as possible
- Determine whether any activities in addition to testing could be employed to reduce risk (e.g., providing training to inexperienced designers)

Risk-based Testing and Product Quality

To ensure that the likelihood of a product failure is minimized, risk management activities provide a disciplined approach to:

- Analyze (and re-evaluate on a regular basis) what can go wrong (risks)
- Determine which risks are important to deal with
- Implement actions to mitigate those risks
- Make contingency plans to deal with the risks should they become actual events
- In addition, testing may identify new risks, help to determine what risks should be mitigated, and lower uncertainty about risks.

Defect Management

Defects found during testing **should be logged**.

The **way** in which **defects are logged** may vary, depending on:

- **context of the component or system being tested,**
- **the test level**
- **the software development lifecycle model.**

Any defects identified **should be investigated and should be tracked** from discovery and classification to their resolution

Defect Management

Typical defect reports have the following objectives:

- **Provide developers and other parties with information about any adverse event** that occurred, to enable them to identify specific effects, to isolate the problem with a minimal reproducing test, and to correct the potential defect(s), as needed or to otherwise resolve the problem
- **Provide test managers a means of tracking the quality of the work product and the impact on the testing** (e.g., if a lot of defects are reported, the testers will have spent a lot of time reporting them instead of running tests, and there will be more confirmation testing needed)
- **Provide ideas** for development and test process improvement

Defect Management

A **defect report** filed during dynamic testing typically **includes**:

- An identifier
- A title and a short summary of the defect being reported
- Date of the defect report, issuing organization, and author
- Identification of the test item (configuration item being tested) and environment
- The development lifecycle phase(s) in which the defect was observed
- A description of the defect to enable reproduction and resolution, including logs, database dumps screenshots, or recordings (if found during test execution)
- Expected and actual results
- Scope or degree of impact (severity) of the defect on the interests of stakeholder(s)
- Urgency/priority to fix of the defect report (e.g., open, deferred, duplicate, waiting to be fixed, awaiting confirmation testing, re-opened, closed)
- Conclusions, recommendations and approvals
- Global issues, such as other areas that may be affected by a change resulting from the defect
- Change history, such as the sequence of actions taken by project team members with respect to the defect to isolate, repair, and confirm it as fixed
- References, including the test case that revealed the problem

Sample Question 1.

Which of the following is a MAJOR task of evaluating exit criteria and reporting?

A Writing a test summary report for stakeholders
B Logging the outcome of test execution
C Repeating test activities as a result of action taken for each discrepancy.
D Evaluating testability of the requirements and system

Sample Question 1.

Which of the following is a MAJOR task of evaluating exit criteria and reporting?

A Writing a test summary report for stakeholders
B Logging the outcome of test execution
C Repeating test activities as a result of action taken for each discrepancy.
D Evaluating testability of the requirements and system

Sample Question 2.

Which of the following activities would NORMALLY be undertaken during test planning?

a. Scheduling test analysis and design.
b. Designing Test Conditions.
c. Monitoring test progress.
d. Identifying the objectives of testing.
e. Evaluating test tools.
f. Selecting test metrics for monitoring and control.

A b, c and d
B a, d and f
C a, d and e
D b, c and f

Sample Question 2.

Which of the following activities would NORMALLY be undertaken during test planning?

a. Scheduling test analysis and design.
b. Designing Test Conditions.
c. Monitoring test progress.
d. Identifying the objectives of testing.
e. Evaluating test tools.
f. Selecting test metrics for monitoring and control.

A b, c and d
B a, d and f
C a, d and e
D b, c and f

Sample Question 3.

Why can be tester dependent on configuration management?

A Because configuration management assures that we know the exact version of the testware and the test object
B Because test execution is not allowed to proceed without the consent of the change control board
C Because changes in the test object are always subject to configuration management
D Because configuration management assures the right configuration of the test too

Sample Question 3.

Why can be tester dependent on configuration management?

A Because configuration management assures that we know the exact version of the testware and the test object
B Because test execution is not allowed to proceed without the consent of the change control board
C Because changes in the test object are always subject to configuration management
D Because configuration management assures the right configuration of the test too

Sample Question 4.

Why is it necessary to define a Test Strategy?

A As there are many different ways to test software, thought must be given to decide what will be the most effective way to test the project on hand.

B Starting testing without prior planning leads to chaotic and inefficient test project

C A strategy is needed to inform the project management how the test team will schedule the test-cycles

D Software failure may cause loss of money, time, business reputation, and in extreme cases injury and death. It is therefore critical to have a proper test strategy in place.

Sample Question 4.

Why is it necessary to define a Test Strategy?

A As there are many different ways to test software, thought must be given to decide what will be the most effective way to test the project on hand.

B Starting testing without prior planning leads to chaotic and inefficient test project

C A strategy is needed to inform the project management how the test team will schedule the test-cycles

D Software failure may cause loss of money, time, business reputation, and in extreme cases injury and death. It is therefore critical to have a proper test strategy in place.

Sample Question 5.

Which of the following is not a part of Configuration Management?

A Controlled library accesses
B Record of changes to
C documentation over time
D Auditing conformance to ISO9001
E Identification of test versions

Sample Question 5.

Which of the following is not a part of Configuration Management?

A Controlled library accesses
B Record of changes to
C documentation over time
D Auditing conformance to ISO9001
E Identification of test versions

Sample Question 6.

Which of the following is the most important difference between the metrics-based approach and the expert-based approach to test estimation?

A The metrics-based approach is more accurate than the expert-based approach.
B The metrics-based approach uses calculations from historical data while the expert-based approach relies on team wisdom.
C The metrics-based approach can be used to verify an estimate created using the expert-based approach, but not vice versa.
D The expert-based approach takes longer than the metrics-based approach.

Sample Question 6.

Which of the following is the most important difference between the metrics-based approach and the expert-based approach to test estimation?

A The metrics-based approach is more accurate than the expert-based approach.
B The metrics-based approach uses calculations from historical data while the expert-based approach relies on team wisdom.
C The metrics-based approach can be used to verify an estimate created using the expert-based approach, but not vice versa.
D The expert-based approach takes longer than the metrics-based approach.

Sample Question 7.

A configuration management system would NOT normally provide:

A Linkage of customer requirements to version numbers.
B Facilities to compare test results with expected results.
C The precise differences in versions of software component source code.
D Restricted access to the source code library

Sample Question 7.

A configuration management system would NOT normally provide:

A Linkage of customer requirements to version numbers.
B Facilities to compare test results with expected results.
C The precise differences in versions of software component source code.
D Restricted access to the source code library

Sample Question 8.

The following list contains risks that have been identified for a software product to be developed. Which of these risks is an example of a product risk?

A Not enough qualified testers to complete the planned tests
B Software delivery is behind schedule
C Threat to a patient's life
D 3rd party supplier does not supply as stipulated

Sample Question 8.

The following list contains risks that have been identified for a software product to be developed. Which of these risks is an example of a product risk?

A Not enough qualified testers to complete the planned tests
B Software delivery is behind schedule
C Threat to a patient's life
D 3rd party supplier does not supply as stipulated

Sample Question 9.

A test engineer is testing a Video Player (VCR), and logs the following report:
Title: Fast Forward stops after 2 minutes. It happens every time
Expected result: Fast forward continues until the end of the tape
Severity: High Priority: Urgent What important information did the engineer leave out?

A Identification (Software and hardware) of the VCR
B Actual result
C History of the report
D Ideas for the test case improvement

Sample Question 9.

A test engineer is testing a Video Player (VCR), and logs the following report:
Title: Fast Forward stops after 2 minutes. It happens every time
Expected result: Fast forward continues until the end of the tape
Severity: High Priority: Urgent What important information did the engineer leave out?

A Identification (Software and hardware) of the VCR
B Actual result
C History of the report
D Ideas for the test case improvement

Sample Question 10.

Which set of metrics can be used for monitoring of the test execution?

A Number of detected defects, testing cost
B Number of residual defects in the test object.
C Percentage of completed tasks in the preparation of test environment; test cases prepared
D Number of test cases run / not run; test cases passed / failed Correct

Sample Question 10.

Which set of metrics can be used for monitoring of the test execution?

A Number of detected defects, testing cost
B Number of residual defects in the test object.
C Percentage of completed tasks in the preparation of test environment; test cases prepared
D Number of test cases run / not run; test cases passed / failed Correct

Sample Question 11.

From a Testing perspective, what are the MAIN purposes of Configuration Management?:
i) Identifying the version of software under test.
ii) Controlling the version of testware items.
iii) Developing new testware items.
iv) Tracking changes to testw are items.
v) Analysing the need for new testware items.

A. ii, iv and v
B. ii, iii and iv
C. i, ii and iv
D. i, iii and v

Sample Question 11.

From a Testing perspective, what are the MAIN purposes of Configuration Management?:
i) Identifying the version of software under test.
ii) Controlling the version of testware items.
iii) Developing new testware items.
iv) Tracking changes to testware items.
v) Analyzing the need for new testware items.

A. ii, iv and v
B. ii, iii and iv
C. i, ii and iv
D. i, iii and v

Sample Question 12.

Which is a potential product risk factor?

A. Failure of third party vendor
B. Training issues
C. Problems requirements definition
D. Poor software functionality

Sample Question 12.

Which is a potential product risk factor?

A. Failure of third party vendor
B. Training issues
C. Problems requirements definition
D. Poor software functionality

Sample Question 13.

Which of the following would you NOT usually find on a software incident report?

A. The name and / or organizational position of the person raising the problem
B. Version of the Software Under Test
C. Suggestions as to how to fix the problem
D. Actual and expected results.

Sample Question 13.

Which of the following would you NOT usually find on a software incident report?

A. The name and / or organizational position of the person raising the problem
B. Version of the Software Under Test
C. Suggestions as to how to fix the problem
D. Actual and expected results.

Sample Question 14.

When should configuration management procedures be implemented?

A. During test planning
B. During test analysis
C. During test execution
D. When evaluating exit criteria

Sample Question 14.

When should configuration management procedures be implemented?

A. During test planning
B. During test analysis
C. During test execution
D. When evaluating exit criteria

Sample Question 15.

Measurement dysfunction is a problem because

A. Even though the numbers you look at appear better, to achieve these numbers, people are doing other aspects of their work much less well
B. We don't know how to measure a variable (our measurement is dysfunctional) and so we don't know how to interpret the result
C. You are measuring the wrong thing and thus reaching the wrong conclusions
D. All of the above

Sample Question 15.

Measurement dysfunction is a problem because

A. Even though the numbers you look at appear better, to achieve these numbers, people are doing other aspects of their work much less well

B. We don't know how to measure a variable (our measurement is dysfunctional) and so we don't know how to interpret the result

C. You are measuring the wrong thing and thus reaching the wrong conclusions

D. All of the above

Sample Question 16.

Poor software characteristics are

A. Only Project risks
B. Only Product risks
C. Project risks and Product risks
D. Project risks or Product risks

Sample Question 16.

Poor software characteristics are

A. Only Project risks
B. Only Product risks
C. Project risks and Product risks
D. Project risks or Product risks

Sample Question 17.

Which is not the project risks

A. Supplier issues
B. Organization factors
C. Technical issues
D. Error-prone software delivered

Sample Question 17.

Which is not the project risks

A. Supplier issues
B. Organization factors
C. Technical issues
D. Error-prone software delivered

Sample Question 18.

FPA is used to

A. To measure the functional requirements of the project
B. To measure the size of the functionality of an Information system
C. To measure the functional testing effort
D. To measure the functional flow

Sample Question 18.

FPA is used to

A. To measure the functional requirements of the project
B. To measure the size of the functionality of an Information system
C. To measure the functional testing effort
D. To measure the functional flow

Sample Question 19.

Based on the IEEE Standard for Software Test Documentation (IEEE Std 829-1998), which sections of the test incident report should the following items be recorded?

Sections

a) Test incident report identifier
b) Summary
c) Incident description
d) Impact

Items

1. Impact on test plans
2. Unique identifier
3. Anomalies
4. Procedure step
5. Environment
6. References to other relevant documents

A. a: 2; b: 4; c: 1, 3 and 5; d: 6
B. a: 2; b: 3; c: 4, 5 and 6; d: 1
C. a: 2; b: 6; c: 3, 4 and 5; d: 1
D. a: 2; b: 1; c: 3, 4 and 5; d: 6

Sample Question 19.

Based on the IEEE Standard for Software Test Documentation (IEEE Std 829-1998), which sections of the test incident report should the following items be recorded?

Sections

a) Test incident report identifier
b) Summary
c) Incident description
d) Impact

Items

1. Impact on test plans
2. Unique identifier
3. Anomalies
4. Procedure step
5. Environment
6. References to other relevant documents

A. a: 2; b: 4; c: 1, 3 and 5; d: 6
B. a: 2; b: 3; c: 4, 5 and 6; d: 1
C. a: 2; b: 6; c: 3, 4 and 5; d: 1
D. a: 2; b: 1; c: 3, 4 and 5; d: 6

Sample Question 20.

A Project risk includes which of the following :

A. Organizational Factors
B. Poor Software characteristics
C. Error Prone software delivered.
D. Software that does not perform its intended functions

Sample Question 20.

A Project risk includes which of the following :

A. Organizational Factors
B. Poor Software characteristics
C. Error Prone software delivered.
D. Software that does not perform its intended functions

Sample Question 21.

Reporting Discrepancies as incidents is a part of which phase :

A. Test Analysis and Design
B. Test Implementation and execution
C. Test Closure Activities
D. Evaluating exit criteria and reporting

Sample Question 21.

Reporting Discrepancies as incidents is a part of which phase :

A. Test Analysis and Design
B. **Test Implementation and execution**
C. Test Closure Activities
D. Evaluating exit criteria and reporting

Sample Question 22.

In a risk-based approach the risks identified may be used to :
i. Determine the test technique to be employed
ii. Determine the extent of testing to be carried out
iii. Prioritize testing in an attempt to find critical defects as early as possible.
iv. Determine the cost of the project

A. ii is True; i, iii, iv & v are False
B. i,ii,iii are true and iv is false
C. ii & iii are True; i, iv are False
D. ii, iii & iv are True; i is false

Sample Question 22.

In a risk-based approach the risks identified may be used to :
i. Determine the test technique to be employed
ii. Determine the extent of testing to be carried out
iii. Prioritize testing in an attempt to find critical defects as early as possible.
iv. Determine the cost of the project

A. ii is True; i, iii, iv & v are False
B. i,ii,iii are true and iv is false
C. ii & iii are True; i, iv are False
D. ii, iii & iv are True; i is false

Sample Question 23.

Incidents would not be raised against:

A. Requirements
B. Documentation
C. Test cases
D. Improvements suggested by users

Sample Question 23.

Incidents would not be raised against:

A. Requirements
B. Documentation
C. Test cases
D. **Improvements suggested by users**

Sample Question 24.

Which of the following is a valid objective of an incident report?

A. Prove that the tester is contributing to the quality of the system.
B. Provides test management ideas for test process improvement.
C. Gives a statistical way to determine which modules to redesign.
D. Provides developers a way to critique their individual work processes.

Sample Question 24.

Which of the following is a valid objective of an incident report?

A. Prove that the tester is contributing to the quality of the system.
B. Provides test management ideas for test process improvement.
C. Gives a statistical way to determine which modules to redesign.
D. Provides developers a way to critique their individual work processes.

Sample Question 25.

Which tasks are performed by a test leader versus a tester?
S. Writing a project test strategy
T. Selecting tools to support testing
U. Preparing and acquiring data
V. Scheduling tests

A. Test leader: S and V; Tester: T and U
B. Test leader: S, T and V; Tester: U
C. Test leader: S, U and V; Tester: T
D. Test leader: S; Tester: T, U and V

Sample Question 25.

Which tasks are performed by a test leader versus a tester?
S. Writing a project test strategy
T. Selecting tools to support testing
U. Preparing and acquiring data
V. Scheduling tests

A. Test leader: S and V; Tester: T and U
B. **Test leader: S, T and V; Tester: U**
C. Test leader: S, U and V; Tester: T
D. Test leader: S; Tester: T, U and V

Sample Question 26.

What is the ratio of the number of failures relative to a category and a unit of measure?

A. Failure rate
B. Defect density
C. Failure mode
D. Fault tolerance

Sample Question 26.

What is the ratio of the number of failures relative to a category and a unit of measure?

A. Failure rate
B. Defect density
C. Failure mode
D. Fault tolerance

Sample Question 27.

EULA stands for

A. End Usability License Agreement
B. End User License Agreement
C. End User License Arrangement
D. End User License Attachment

Sample Question 27.

EULA stands for

A. End Usability License Agreement
B. **End User License Agreement**
C. End User License Arrangement
D. End User License Attachment

Sample Question 28.

How can software defects in future projects be prevented from reoccurring?

A. Creating documentation procedures and allocating resource contingencies
B. Asking programmers to perform a thorough and independent testing
C. Combining levels of testing and mandating inspections of all documents
D. Documenting lessons learned and determining the root cause of problems

Sample Question 28.

How can software defects in future projects be prevented from reoccurring?

A. Creating documentation procedures and allocating resource contingencies
B. Asking programmers to perform a thorough and independent testing
C. Combining levels of testing and mandating inspections of all documents
D. Documenting lessons learned and determining the root cause of problems

Sample Question 29.

Defect Management process does not include

A. Defect prevention
B. Deliverable base-lining
C. Management reporting
D. None of the above

Sample Question 29.

Defect Management process does not include

A. Defect prevention
B. **Deliverable base-lining**
C. Management reporting
D. None of the above

Sample Question 30.

The inputs for developing a test plan are taken from

A. Project plan
B. Business plan
C. Support plan
D. None of the above

Sample Question 30.

The inputs for developing a test plan are taken from

A. Project plan
B. Business plan
C. Support plan
D. None of the above

Sample Question 31.

Which document specifies the sequence of test executions?

A. Test procedure specification
B. Test design specification
C. Test case specification
D. Test plan

Sample Question 31.

Which document specifies the sequence of test executions?

A. Test procedure specification
B. Test design specification
C. Test case specification
D. Test plan

Sample Question 32.

Software testing accounts to what percent of software development costs?

A. 10-20
B. 40-50
C. 70-80
D. 5-10

Sample Question 32.

Software testing accounts to what percent of software development costs?

A. 10-20
B. 40-50
C. 70-80
D. 5-10

Sample Question 33.

When should we stop our testing?

A. This question is difficult to answer
B. The answer depends on the contract with the client, special requirements if any & risks your organization is willing to take
C. The answer depends on the experience & maturity of your developers
D. The answer should be standardized for the software development industry

Sample Question 33.

When should we stop our testing?

A. This question is difficult to answer
B. **The answer depends on the contract with the client, special requirements if any & risks your organization is willing to take**
C. The answer depends on the experience & maturity of your developers
D. The answer should be standardized for the software development industry

Sample Question 34.

Function/Test matrix is a type of

A. Interim Test report
B. Final test report
C. Project status report
D. Management report

Sample Question 34.

Function/Test matrix is a type of

A. Interim Test report
B. Final test report
C. Project status report
D. Management report

Sample Question 35.

Which of the following is NOT part of configuration management:

A. Status accounting of configuration items
B. Auditing conformance to ISO9001
C. Identification of test versions
D. Record of changes to documentation over time
E. Controlled library access

Sample Question 35.

Which of the following is NOT part of configuration management:

A. Status accounting of configuration items
B. Auditing conformance to ISO9001
C. Identification of test versions
D. Record of changes to documentation over time
E. Controlled library access

Sample Question 36.

What is the purpose of test completion criteria in a test plan:

A. To know when a specific test has finished its execution
B. To ensure that the test case specification is complete
C. To set the criteria used in generating test inputs
D. To know when test planning is complete
E. To plan when to stop testing

Sample Question 36.

What is the purpose of test completion criteria in a test plan:

A. To know when a specific test has finished its execution
B. To ensure that the test case specification is complete
C. To set the criteria used in generating test inputs
D. To know when test planning is complete
E. **To plan when to stop testing**

Sample Question 37.

What are the main objectives of software project risk management?

A. Increase focus on preventive processes and improve tester job satisfaction
B. Reduce the probability of occurrence and decrease the potential impact
C. Control contractor problems and minimize the impact of corporate politics
D. Control contractor problems and minimize the impact of corporate politics

Sample Question 37.

What are the main objectives of software project risk management?

A. Increase focus on preventive processes and improve tester job satisfaction
B. Reduce the probability of occurrence and decrease the potential impact
C. Control contractor problems and minimize the impact of corporate politics
D. Control contractor problems and minimize the impact of corporate politics

Sample Question 38.

Control contractor problems and minimize the impact of corporate politics

A. Ishikawa diagram
B. Cause-and-effect diagram
C. Lessons learned
D. Fishbone diagram

Sample Question 38.

Control contractor problems and minimize the impact of corporate politics

A. Ishikawa diagram
B. Cause-and-effect diagram
C. **Lessons learned**
D. Fishbone diagram

Sample Question 39.

Expected results are:

A. Only important in system testing
B. Only used in component testing
C. Never specified in advance
D. Most useful when specified in advance
E. Derived from the code

Sample Question 39.

Expected results are:

A. Only important in system testing
B. Only used in component testing
C. Never specified in advance
D. Most useful when specified in advance
E. Derived from the code

Sample Question 40.

Exit Criteria may consist of:
i. Thoroughness measures, such as coverage of code, functionality or risk
ii. Estimates of Defect density or reliability measures.
iii. Residual risk such as defects not fixed or lack of test coverage in certain areas
iv. Verifying the Test Environment.

A. iv is correct and i,ii,iii are incorrect.
B. i,ii,iii is correct and iv is incorrect
C. ii is correct and i,ii,iii are incorrect
D. iii and iv are correct and i,ii are incorrect

Sample Question 40.

Exit Criteria may consist of:
i. Thoroughness measures, such as coverage of code, functionality or risk
ii. Estimates of Defect density or reliability measures.
iii. Residual risk such as defects not fixed or lack of test coverage in certain areas
iv. Verifying the Test Environment.

A. iv is correct and i,ii,iii are incorrect.
B. i,ii,iii is correct and iv is incorrect
C. ii is correct and i,ii,iii are incorrect
D. iii and iv are correct and i,ii are incorrect

Sample Question 41.

Test Conditions are derived from:

A. Specifications
B. Test Cases
C. Test Data
D. Test Design

Sample Question 41.

Test Conditions are derived from:

A. Specifications
B. Test Cases
C. Test Data
D. Test Design

Sample Question 42.

Which of the following is the task of a Test Lead / Leader.
i. Interaction with the Test Tool Vendor to identify best ways to leverage test tool on the project.
ii. Write Test Summary Reports based on the information gathered during testing
iii. Decide what should be automated , to what degree and how.
iv. Create the Test Specifications

A. i, ii, iii is true and iv is false
B. ii,iii,iv is true and i is false
C. i is true and ii,iii,iv are false
D. iii and iv is correct and i and ii are incorrect

Sample Question 42.

Which of the following is the task of a Test Lead / Leader.
i. Interaction with the Test Tool Vendor to identify best ways to leverage test tool on the project.
ii. Write Test Summary Reports based on the information gathered during testing
iii. Decide what should be automated , to what degree and how.
iv. Create the Test Specifications

A. i, ii, iii is true and iv is false
B. ii,iii,iv is true and i is false
C. i is true and ii,iii,iv are false
D. iii and iv is correct and i and ii are incorrect

Sample Question 43.

Impact Analysis helps to decide:

A. How much regression testing should be done
B. Exit Criteria
C. How many more test cases need to written.
D. Different Tools to perform Regression Testing

Sample Question 43.

Impact Analysis helps to decide:

A. How much regression testing should be done
B. Exit Criteria
C. How many more test cases need to written.
D. Different Tools to perform Regression Testing

Sample Question 44.

Which of the following are false?

A. Incidents should always be investigated and resolved.
B. Incidents occur when expected and actual results differ.
C. Incidents can be analyzed to assist in test process improvement.
D. An incident can be raised against documentation.

Sample Question 44.

Which of the following are false?

A. Incidents should always be investigated and resolved.
B. Incidents occur when expected and actual results differ.
C. Incidents can be analyzed to assist in test process improvement.
D. An incident can be raised against documentation.

Sample Question 45.

Which of the following is least important in test management?

A. Estimating test duration.
B. Incident Management.
C. Configuration Management.
D. De-bugging.

Sample Question 46.

Which of the following is least important in test management?

A. Estimating test duration.
B. Incident Management.
C. Configuration Management.
D. De-bugging.

Sample Question 47.

Amount of testing performed will not depend on

A. Risks involved
B. Contractual requirements
C. Legal requirements
D. Test data.

Sample Question 47.

Amount of testing performed will not depend on

A. Risks involved
B. Contractual requirements
C. Legal requirements
D. Test data.

Sample Question 48.

An incident logging system

A. Only records defects
B. Is of limited value
C. Is a valuable source of project information during testing if it contains all incidents
D. Should be used only by the test team.

Sample Question 48.

An incident logging system

A. Only records defects
B. Is of limited value
C. **Is a valuable source of project information during testing if it contains all incidents**
D. Should be used only by the test team.

Sample Question 49.

When should you stop testing?

A. When time for testing has run out.
B. When all planned tests have been run
C. When the test completion criteria have been met
D. When no faults have been found by the tests run

Sample Question 49.

When should you stop testing?

A. When time for testing has run out.
B. When all planned tests have been run
C. **When the test completion criteria have been met**
D. When no faults have been found by the tests run

Sample Question 50.

Which of the following are KEY tasks of a test leader?
i. Understanding the project risks
ii. Measuring performance of components
iii. Scheduling tests and other activities
iv. Using monitoring tools as needed

A. i and iii
B. i and ii
C. iii and iv
D. ii and iii

Sample Question 50.

Which of the following are KEY tasks of a test leader?
i. Understanding the project risks
ii. Measuring performance of components
iii. Scheduling tests and other activities
iv. Using monitoring tools as needed

A. i and iii
B. i and ii
C. iii and iv
D. ii and iii

Sample Question 51.

What is the purpose of test exit criteria in the test plan?

A. To specify when to stop the testing activity
B. To set the criteria used in generating test inputs
C. To ensure that the test case specification is complete
D. To know when a specific test has finished its execution

Sample Question 51.

What is the purpose of test exit criteria in the test plan?

A. **To specify when to stop the testing activity**
B. To set the criteria used in generating test inputs
C. To ensure that the test case specification is complete
D. To know when a specific test has finished its execution

Sample Question 52.

Which of the following items need not to be given in an incident report?

A. The version number of the test object
B. Test data and used environment
C. Identification of the test case that failed
D. The location and instructions on how to correct the fault

Sample Question 52.

Which of the following items need not to be given in an incident report?

A. The version number of the test object
B. Test data and used environment
C. Identification of the test case that failed
D. **The location and instructions on how to correct the fault**

Sample Question 53.

Who should be responsible for coordinating the test strategy with the project manager and others?

A. Tester
B. Developer
C. Customer
D. Test leader

Sample Question 53.

Who should be responsible for coordinating the test strategy with the project manager and others?

A. Tester
B. Developer
C. Customer
D. Test leader

Sample Question 54.

Which one is not the task of test leader?

A. Coordinate the test strategy and plan with project managers and others
B. Decide about the implementation of the test environment
C. Write test summary reports
D. Review and contribute to test plans

Sample Question 54.

Which one is not the task of test leader?

A. Coordinate the test strategy and plan with project managers and others
B. Decide about the implementation of the test environment
C. Write test summary reports
D. Review and contribute to test plans

Sample Question 55.

Size of a project is defined in terms of all the following except

A. Person days
B. Person hours
C. Calendar months
D. None of the above

Sample Question 55.

Size of a project is defined in terms of all the following except

A. Person days
B. Person hours
C. Calendar months
D. None of the above

Sample Question 56.

Risk analysis talks about

A. The data required for testing, the infrastructure requirements to manage the data as well as the methods for preparing test data, requirements, converters and sources

B. Details what types of tests must be conducted, what stages of testing are required and outlines the sequence and timing of tests

C. testing goal. It is a statement of what the tester is expected to accomplish or validate during a testing activity. These guide the development of test cases and procedures

D. None of the above

Sample Question 56.

Risk analysis talks about

A. The data required for testing, the infrastructure requirements to manage the data as well as the methods for preparing test data, requirements, converters and sources

B. Details what types of tests must be conducted, what stages of testing are required and outlines the sequence and timing of tests

C. testing goal. It is a statement of what the tester is expected to accomplish or validate during a testing activity. These guide the development of test cases and procedures

D. None of the above

Sample Question 57.

What are the 2 major components taken into consideration with risk analysis?

A. The probability the negative event will occur
B. The potential loss or impact associated with the event
C. Both A. and B
D. Neither A. nor B

Sample Question 57.

What are the 2 major components taken into consideration with risk analysis?

A. The probability the negative event will occur
B. The potential loss or impact associated with the event
C. Both A. and B
D. Neither A. nor B

Sample Question 58.

Which of the following are typical tester tasks?

A. Decide what should be automated, to what degree, and how.
B. Set up configuration management of testware; review tests developed by others.
C. Prepare and acquire test data; review tests developed by others.
D. Initiate the specification, preparation, implementation and execution of tests and monitor and control the execution.

Sample Question 58.

Which of the following are typical tester tasks?

A. Decide what should be automated, to what degree, and how.
B. Set up configuration management of testware; review tests developed by others.
C. Prepare and acquire test data; review tests developed by others.
D. Initiate the specification, preparation, implementation and execution of tests and monitor and control the execution.

Sample Question 59.

Following are some of the testing risks

A. Budget, Test environment
B. Budget, Number of qualified test resources
C. Budget, Number of qualified test resources, Test environment
D. None of the above

Sample Question 59.

Following are some of the testing risks

A. Budget, Test environment
B. Budget, Number of qualified test resources
C. Budget, Number of qualified test resources, Test environment
D. None of the above

Sample Question 60.

Which of the following demonstrates independence in testing?

J. Independent testers are external to the organization
K. Independent testers are part of the development team
L. Independent testers are from the user community
M. Programmers who wrote the code serve as independent testers
N. Customers who wrote the requirements serve as independent testers

A. J. L and N
B. J. K, L and N
C. K. M and N
D. J, L, M and N

Sample Question 60.

Which of the following demonstrates independence in testing?

J. Independent testers are external to the organization
K. Independent testers are part of the development team
L. Independent testers are from the user community
M. Programmers who wrote the code serve as independent testers
N. Customers who wrote the requirements serve as independent testers

A. J. L and N
B. J. K, L and N
C. K. M and N
D. J, L, M and N

Sample Question 61.

In prioritizing what to test, the most important objective is to:

A. Find as many faults as possible.
B. Test high risk areas.
C. Obtain good test coverage.
D. Test whatever is easiest to test.

Sample Question 61.

In prioritizing what to test, the most important objective is to:

A. Find as many faults as possible.
B. Test high risk areas.
C. Obtain good test coverage.
D. Test whatever is easiest to test.

Sample Question 62.

Which is the MOST important advantage of independence in testing?

A. An independent tester may find defects more quickly than the person who wrote the software.
B. An independent tester may be more focused on showing how the software works than the person who wrote the software.
C. An independent tester may be more effective and efficient because they are less familiar with the software than the person who wrote it.
D. An independent tester may be more effective at finding defects missed by the person who wrote the software.

Sample Question 62.

Which is the MOST important advantage of independence in testing?

A. An independent tester may find defects more quickly than the person who wrote the software.
B. An independent tester may be more focused on showing how the software works than the person who wrote the software.
C. An independent tester may be more effective and efficient because they are less familiar with the software than the person who wrote it.
D. **An independent tester may be more effective at finding defects missed by the person who wrote the software.**

Sample Question 63.

What is the difference between a project risk and a product risk?

A. Project risks are potential failure areas in the software or system; product risks are risks that surround the project's capability to deliver its objectives.

B. Project risks are the risks that surround the project's capability to deliver its objectives; product risks are potential failure areas in the software or system.

C. Project risks are typically related to supplier issues, organizational factors and technical issues; product risks are typically related to skill and staff shortages.

D. Project risks are risks that delivered software will not work; product risks are typically related to supplier issues, organizational factors and technical issues.

Sample Question 63.

What is the difference between a project risk and a product risk?

A. Project risks are potential failure areas in the software or system; product risks are risks that surround the project's capability to deliver its objectives.

B. **Project risks are the risks that surround the project's capability to deliver its objectives; product risks are potential failure areas in the software or system.**

C. Project risks are typically related to supplier issues, organizational factors and technical issues; product risks are typically related to skill and staff shortages.

D. Project risks are risks that delivered software will not work; product risks are typically related to supplier issues, organizational factors and technical issues.