DUMPSARENA

ISTQB Certified Tester Foundation Level

iSQI CTFL-001

Version Demo

Total Demo Questions: 36

Total Premium Questions: 364

Buy Premium PDF

https://dumpsarena.co

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

Topic Break Down

Topic	No. of Questions
Topic 1, Fundamentals of Testing	48
Topic 2, Testing Throughout the Software Development Lifecycle	48
Topic 3, Static Testing	30
Topic 4, Test Analysis and Design	77
Topic 5, Managing the Test Activities	131
Topic 6, Test Tools	30
Total	364

QUESTION NO: 1

What is the concept of introducing a small change to the program and having theeffects of that change show up in some test..?

A. Mutation testing

B. Performance testing

C. A mutation error

D. Debugging a program

ANSWER: A

Explanation:

Mutation testing is the technique in which a tester deliberately introduces small changes into a program, creating altered versions known as mutants, and then executes the test suite against them. The objective is to determine whether the tests detect the behavioral effect of each introduced change. If a test fails when run against a mutant, the mutant has been “killed,” providing evidence that the test suite can detect that type of defect. If the mutant produces the same observed behavior for all tests, it may reveal a weakness in the tests or represent an equivalent mutant whose changed code does not alter externally observable behavior. Mutation testing is therefore a test-quality and test-suite-effectiveness technique rather than a normal functional test level. It helps assess whether tests are sufficiently sensitive to small faults that could realistically occur in software. The ISTQB glossary defines mutation testing as the process of deliberately introducing faults into software and verifying whether tests detect them. See the ISTQB Glossary entry for mutation testing and the ISTQB Certified Tester Foundation Level certification page.

QUESTION NO: 2

Identify the THREE types of formal peer reviews that can be recognized. 1 credit

A. Inspection

B. Management review

C. Walkthrough

D. Audit

E. Technical review

F. Informal review

G. Assessment

ANSWER: A C E

Explanation:

The recognized formal peer-review types are **Inspection**, **Walkthrough**, and **Technical review**. These are structured approaches used to evaluate a work product, such as requirements, test cases, architecture, code, or user documentation, without executing the software. Their common purpose is early defect detection, improving the quality of the work product, and sharing knowledge among participants.

An inspection is the most rigorous of these review types and normally follows a defined process with clear roles, entry and exit criteria, individual preparation, and defect logging. A walkthrough is commonly led by the author, who presents the work product to reviewers to obtain feedback and build shared understanding. A technical review is performed by technically qualified reviewers who assess the work product’s technical content and its suitability for the intended purpose. The ISTQB Foundation Level syllabus identifies these as the principal formal review types and explains that the selected approach should fit the review objective, work product, organizational context, and relevant risks. See the ISTQB Certified Tester Foundation Level certification information and the ISTQB Glossary entry for review.

QUESTION NO: 3

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

ANSWER: E

Explanation:

To plan when to stop testing is correct because test completion criteria, also called exit criteria, define the conditions that must be satisfied before a test activity can be considered complete. They enable the test team and stakeholders to make a planned, evidence-based decision about whether sufficient testing has been performed, rather than stopping simply because a scheduled date has arrived or because no further tests are immediately available. Typical completion criteria may include achieving a specified level of requirements, code, or risk coverage; executing a defined proportion of planned tests; reaching an agreed pass rate; resolving defects above a stated severity threshold; or obtaining stakeholder acceptance of residual risk. These criteria should be established during test planning, be measurable where practical, and align with the test objectives, risks, scope, and available resources. If the criteria are not fully met, the team can assess the remaining risks and decide whether to continue testing, adjust the plan, or formally accept the residual risk. This use of completion criteria supports transparency and informed release decisions. See the ISTQB Certified Tester Foundation Level syllabus and the ISTQB Glossary entry for test completion criteria.

QUESTION NO: 4

The introduction of reviews and inspections has often failed as a process improvement action. Identify the THREE most important measures that should be taken to reduce the risk that this test process improvement will fail. 2 Credits (for 2 out of 3 correct 1 credit)

A. Process ownership and experienced moderators who drive the inspection process.

B. Management support

C. Training of those involved

D. The availability of stands and processes

E. Usage of a more traditional software development lifecycle

F. Alignment with software process improvement

G. Using a reference model, e.g. TMMi

ANSWER: A B C

Explanation:

Successful introduction of reviews and inspections depends on making the activity a supported, repeatable organizational practice rather than an isolated quality initiative. **Process ownership and experienced moderators who drive the inspection process.** is essential because ownership establishes accountability for maintaining and improving the process, while competent moderation ensures that reviews are planned, conducted, and followed up consistently. **Management support** provides the authority, time, resources, and organizational commitment needed for participants to treat reviews as valuable process activities. This support is particularly important where review findings require changes to development practices or priorities. **Training of those involved** ensures that authors, reviewers, and moderators understand their respective responsibilities, review techniques, entry and exit criteria, defect recording, and constructive review behaviour.

Together, these measures address the process, organizational, and people dimensions required for sustained review effectiveness. The ISTQB Foundation Level syllabus identifies management support, adequate training, clear objectives, and capable participants as key review success factors. See the ISTQB Certified Tester Foundation Level certification information and the ISTQB glossary entry for review.

QUESTION NO: 5

Instead of having an independent test team within the company, the company is considering to outsource testing. What are THREE key challenges that are typical for outsourcing? 1 credit

A. Test environment more complex

B. Define expectation for tasks and deliverables

C. Clear channels of communications

D. Possibly different cultures

E. Testing of non-functional requirements

F. Audit trail from requirements to test cases

G. Applying test automation

ANSWER: B C D

Explanation:

Outsourced testing requires deliberate management of the relationship between the customer organization and the external supplier. “Define expectation for tasks and deliverables” is a key challenge because the parties need a shared, sufficiently detailed understanding of the testing scope, responsibilities, work products, quality criteria, reporting, acceptance conditions, and timing. This prevents gaps in accountability and makes the supplier’s service measurable. “Clear channels of communications” is also essential: outsourcing introduces organizational boundaries, so escalation paths, meeting arrangements, reporting routes, contacts, and decision-making authority must be established and maintained. Effective communication supports timely resolution of defects, risks, clarifications, and changes. “Possibly different cultures” is a typical challenge because outsourced work may involve differences in national, organizational, professional, and language cultures. These differences can affect how requirements are interpreted, how issues are raised, and how feedback is understood. The test manager should account for these factors when planning collaboration, defining working agreements, and monitoring the service. These topics align with the Foundation Level syllabus discussion of managing independent testing and distributed or outsourced test work. See the ISTQB Certified Tester Foundation Level certification page and the CTFL syllabus resources.

QUESTION NO: 6

Which of the following is a role of a formal review? [K1]

A. Adjudicator

B. Moderator

C. Governor

D. Corrector

ANSWER: B

Explanation:

Moderator is a defined role in a formal review. The moderator is responsible for facilitating the review process so that it is conducted effectively and in accordance with the agreed review approach. Typical moderator activities include planning the review, arranging meetings where applicable, ensuring that participants are prepared, managing the review discussion,

maintaining an appropriate focus on identifying defects, and ensuring review outcomes are recorded and followed up. The moderator also supports an environment in which reviewers can raise issues constructively and objectively. Formal reviews use clearly allocated roles because this makes the review repeatable, controlled, and measurable; it also helps separate facilitation of the process from authorship of the work product being reviewed. The ISTQB Certified Tester Foundation Level syllabus identifies moderator as one of the possible review roles, alongside roles such as author, reviewer, review leader, and scribe. Therefore, “Moderator” is the correct formal-review role. See the ISTQB Certified Tester Foundation Level certification page and the CTFL syllabus and supporting resources.

QUESTION NO: 7

Within the projects, a master test plan and phase test plan will be used. Following is a list of characteristics applicable for test plans:

a. Any deviation from the procedures described in the test strategy document

b. The overall estimated costs, timescales and resource requirements

c. A detailed schedule of testing activities

d. The development deliverables to be tested

e. Which test staff members (names) will be involved and when

f. Level of requirements coverage achieved

Which THREE of the above mentioned characteristics relate to the master test plan? 1 credit

A. a

B. b

C. c

D. d

E. e

F. f

ANSWER: A B D

Explanation:

A master test plan provides the high-level, project-wide basis for organizing and controlling testing across phases or test levels. It therefore records any deviation from the procedures described in the test strategy document, ensuring that the project’s intended testing approach and any approved tailoring of that approach are visible and governed. It also defines the overall estimated costs, timescales and resource requirements, because these are planning assumptions and commitments that must be coordinated at the overall test-management level rather than only within an individual phase. Finally, it identifies the development deliverables to be tested, establishing the broad scope of the testing work and the items to which the strategy applies. Together, these characteristics support the master plan’s purpose of defining overall scope, approach, resources, schedule expectations, and management controls. Detailed operational scheduling and assignment are subsequently elaborated in lower-level or phase test plans, while achieved coverage is test-monitoring and reporting information. ISTQB describes a test plan as documentation that specifies the scope, approach, resources, and schedule of intended test activities; see the ISTQB Glossary entry for test plan. The applicable Foundation Level certification context is available from ISTQB Certified Tester Foundation Level.

QUESTION NO: 8

Regarding the process described above, what is the most important recommendation you would make using IEEE 1044 as a guide? 2 credits

A. No priority or severity assigned

B. Incident numbering is manual rather than automated

C. No mentioning of reproduceability

D. No classification on type of incident

ANSWER: A

Explanation:

No priority or severity assigned is the most important recommendation because IEEE 1044 guides organizations to record classification information that supports consistent evaluation and management of reported anomalies. Severity expresses the extent of the actual or potential impact of an anomaly, while priority supports deciding the order and urgency with which the anomaly should be addressed. Capturing both attributes enables the team to make risk-based decisions, allocate limited investigation and correction capacity appropriately, and communicate clearly with stakeholders about the consequences of leaving an issue unresolved. It also supports meaningful reporting and trend analysis: teams can identify whether high-impact issues are recurring, assess quality risks, and verify that urgent items receive timely attention. Without a defined severity and priority approach, resolution order can become subjective and inconsistent, potentially delaying action on anomalies with the greatest product, business, safety, or user impact. IEEE 1044 defines a standard classification framework for anomaly information, including information used to characterize impact and support disposition and resolution decisions. See the IEEE 1044 standard overview and the ISTQB Certified Tester Foundation Level overview.

QUESTION NO: 9

Identify THREE items that would be part of the work-breakdown structure showing the key testing activities for the acceptance test project. 2 credits (for 2 out of 3 correct 1 credit)

A. Test planning, test case preparation and test execution for each of the four iterations

B. Work should be explicitly allocated to test completion, test management, installation and to training on using the system

C. Activities to deploy the system in the user environment

D. Regression testing in the second, third and fourth iterations

E. Development activities for unit and integration testing

F. Reviews on requirements documentation

G. Defining test environment requirements for system testing

ANSWER: A B D

Explanation:

A work-breakdown structure for an acceptance test project should identify the testing work to be planned, managed, performed, and completed, with sufficient detail to support estimating, assigning responsibilities, and tracking progress across the project schedule. “Test planning, test case preparation and test execution for each of the four iterations” is appropriate because it breaks core test activities down by iteration, making the work visible and manageable. “Regression testing in the second, third and fourth iterations” is also a valid work item: where iterative delivery changes an already tested product, regression testing is planned to provide confidence that previously working functionality continues to work after changes.

“Work should be explicitly allocated to test completion, test management, installation and to training on using the system” belongs in the acceptance-test project structure because test management and test completion are essential test activities, while installation and user training are practical prerequisites for executing acceptance testing in a representative operational context. A test plan and its associated work breakdown should cover the activities, resources, schedule, responsibilities, and infrastructure needed to achieve the agreed acceptance-test objectives. This is consistent with the ISTQB syllabus discussion of test planning, monitoring and control, test implementation and execution, and test completion. See the ISTQB Certified Tester Foundation Level syllabus and the ISTQB glossary definition of regression testing.

QUESTION NO: 10

Which TWO of the following are valid reasons for performing reviews of requirements? 1 credit

A. To identify ambiguities and inconsistencies before implementation

B. To establish a common understanding among stakeholders

C. To measure response times under peak load

D. To prove that all implemented code paths have been executed

E. To replace the need for acceptance testing

ANSWER: A B

Explanation:

To identify ambiguities and inconsistencies before implementation is correct because requirements reviews can reveal unclear, contradictory, incomplete, or untestable statements while changes are comparatively inexpensive. Correcting these defects early prevents them from being propagated into design, code, and testware.

To establish a common understanding among stakeholders is also correct. Reviews bring together people with different perspectives, such as business representatives, developers, testers, and operations personnel. Discussing requirements helps them agree on intended behavior, constraints, and acceptance criteria.

Reviews are static testing activities and do not measure execution-time characteristics or replace dynamic testing. See the ISTQB Certified Tester Foundation Level resources and the ISTQB Glossary entry for review.

QUESTION NO: 11

Model characteristics:

Which THREE of the below mentioned characteristics relate to TMMi? 1 credit

A. 5 maturity levels

B. Focussed on higher level testing

C. 20 key areas

D. Highly related to CMMI

E. Continuous model

F. Staged model

G. Focussed on component and integration testing

ANSWER: A D F

Explanation:

TMMi (the Test Maturity Model integration) is a test-process improvement framework organized as a staged maturity model with five maturity levels. Organizations progress through defined levels, beginning with an initial level and moving through managed, defined, measured, and optimization-oriented testing practices. This staged structure provides an evolutionary improvement path for assessing and improving the maturity of an organization's testing process.

TMMi is also highly related to CMMI. It was developed as a complementary model that applies process-improvement concepts to testing, and its structure deliberately aligns with CMMI maturity concepts. This relationship helps organizations integrate testing-process improvement with broader software and systems engineering process-improvement initiatives. The TMMi Foundation describes the model as a reference framework for test-process improvement and identifies its five maturity

levels and staged representation. Further detail is available from the TMMi Foundation's TMMi model overview and the TMMi Framework.

QUESTION NO: 12

Which TWO of the review types below are the BEST fitted (most adequate) options to choose for reviewing safety critical components in a software project? Select 2 options.

A. Informal review.

B. Management review.

C. Inspection.

D. Walkthrough

E. Technical Review

ANSWER: C E

Explanation:

Inspection. and **Technical Review** are the most appropriate choices for safety-critical software components because they provide a structured, rigorous examination of work products by suitably qualified participants. Safety-critical development needs high confidence that requirements, design decisions, code, and test artefacts are correct, complete, consistent, and compliant with applicable standards. An inspection is the most formal review type: it uses defined roles, individual preparation, documented entry and exit criteria, defect logging, and follow-up. These controls create objective evidence that identified issues have been addressed, which is particularly valuable where failure could cause harm or unacceptable risk.

A technical review is also well suited because it brings together people with relevant technical expertise to evaluate the work product and reach an informed assessment of its technical quality and fitness for purpose. It can examine complex safety properties, interfaces, architectural decisions, and compliance-related technical concerns that require specialist judgment. The ISTQB Foundation Level syllabus describes inspections and technical reviews as formal review types, with formal reviews offering greater control and documentation than less formal approaches. Their disciplined process and technical depth make them the best fit for reviewing safety-critical components. See the ISTQB Certified Tester Foundation Level certification resources and the ISTQB syllabi library.

QUESTION NO: 13

In addition to risk, identify TWO other components of a testing strategy. 1 credit

A. The entry and exit criteria for each test phase

B. Test training needs for the project resources

C. The test design techniques to be used

D. Test performance indicators

E. The test schedule

ANSWER: A C

Explanation:

A testing strategy provides a high-level, risk-aware framework for how testing will be performed. Alongside the identified product and project risks, it establishes the conditions used to control progression through testing. "The entry and exit criteria for each test phase" are therefore valid strategy components: entry criteria define the prerequisites for starting a test activity, while exit criteria provide measurable conditions for deciding whether the relevant test activity or level can be considered complete. This supports consistent, evidence-based decisions throughout the test process.

“The test design techniques to be used” are also part of a testing strategy because the strategy specifies the general approach for deriving test conditions and test cases. Selecting techniques such as equivalence partitioning, boundary value analysis, decision table testing, or state transition testing aligns the test design work with the nature of the product, applicable risks, and required coverage. The techniques selected at strategy level guide the systematic design of tests, while detailed application is refined during test analysis and design. ISTQB describes test approaches and their relationship to risk and test techniques in its Foundation Level syllabus and terminology: ISTQB Certified Tester Foundation Level and ISTQB Glossary: test strategy.

QUESTION NO: 14

Test Implementation and execution has which of the following major tasks?

i. Developing and prioritizing test cases, creating test data, writing test procedures and optionally preparing the test harnesses and writing automated test scripts.

ii. Creating the test suite from the test cases for efficient test execution.

iii. Verifying that the test environment has been set up correctly.

iv. Determining the exit criteria.

A. i, ii, iii are true and iv is false

B. i, iv are true and ii is false

C. i, ii are true and iii, iv are false

D. ii, iii, iv are true and i is false

ANSWER: A

Explanation:

The correct answer is *i, ii, iii are true and iv is false*. Test implementation turns the previously designed test conditions and test cases into executable testware. This includes developing and prioritizing test cases, preparing the required test data, documenting test procedures, and—where automation is used—preparing harnesses and automated test scripts. Prioritization is important because it supports efficient execution and enables the most valuable or risk-relevant tests to be run first when time is limited.

Implementation also includes grouping individual test cases into test suites or execution schedules. A suite organizes tests so that they can be run effectively, including arranging any needed setup, dependencies, and execution order. Before execution begins, the test team must confirm that the test environment is correctly established and ready: the required hardware, software, interfaces, data, access permissions, and tools must support the planned tests.

These activities align with the implementation and execution work described in the Foundation Level test process. ISTQB identifies test implementation as preparation of test procedures, suites, and environment readiness, while exit criteria are established as part of test planning rather than implementation. See the ISTQB Certified Tester Foundation Level certification page and the CTFL syllabus resources.

QUESTION NO: 15

Within the projects, a master test plan and phase test plan will be used. Following is a list of characteristics applicable for test plans:

a. Any deviation from the procedures described in the test

b. strategy document

c. The overall estimated costs, timescales and resource

d. requirements

e. A detailed schedule of testing activities

f. The development deliverables to be tested

g. Which test staff members (names) will be involved and when

h. Level of requirements coverage achieved

Which TWO of the above mentioned characteristics relate to the phase test plan? 1 credit

A. a

B. b

C. c

D. d

E. e

F. f

ANSWER: E F

Explanation:

A phase test plan provides the operational planning detail for a specific test phase or level. It therefore identifies *"A detailed schedule of testing activities"*, enabling the team to coordinate analysis, design, implementation, execution, reporting, and completion work within that phase. A phase-level schedule is sufficiently granular to support day-to-day control of test work and coordination with the project schedule.

The phase plan also identifies *"The development deliverables to be tested"*. Defining the relevant deliverables establishes the scope of the phase, including the particular components, builds, interfaces, documents, or system capabilities that will be available for evaluation. This gives testers and stakeholders a clear basis for preparing testware, arranging environments, and confirming that the phase is ready to start.

ISTQB describes test planning as determining the scope, objectives, approach, resources, schedule, and test deliverables for the applicable test activities. At phase level, this planning is applied concretely to the specific items and activities of that phase. See the ISTQB Certified Tester Foundation Level overview and the CTFL certification page.

QUESTION NO: 16

Which of the following are most characteristic of structure-based testing?

(i) Information about how the software is constructed is used to derive test cases.

(ii) Statement coverage and/or decision coverage can be measured for existing test cases.

(iii) The knowledge and experience of people are used to derive test cases.

(iv) Test cases are derived from a model or specification of the system.

A. (i) and (ii)

B. (ii) and (iii)

C. (ii) and (iv)

D. (i) and (iii)

ANSWER: A

Explanation:

(i) and (ii) is correct because structure-based testing derives and evaluates tests using the internal structure of the test item. This structure may be represented by source code, control flow, branches, decisions, conditions, or other implementation-level elements. Information about how the software is constructed is therefore directly used to identify test conditions and design test cases that exercise selected structural elements.

A central feature of this approach is coverage measurement. After executing a test suite, the tester can determine the proportion of structural elements that were exercised. Statement coverage measures which executable statements have been executed, while decision coverage measures whether decision outcomes have been exercised. These measures provide objective feedback about the structural extent of testing and can help identify code that has not yet been exercised by the existing tests.

The ISTQB CTFL syllabus identifies statement testing and branch testing as common structure-based test techniques and explains their associated coverage measures. Structure-based techniques can be applied at appropriate test levels, particularly where internal implementation information is available. See the ISTQB Certified Tester Foundation Level certification page and the CTFL syllabus resources.

QUESTION NO: 17

Complete statement and branch coverage means..?

A. That you have tested every statement in the program

B. That tests have exercised every executable statement and every branch in the program

C. That you have tested every IF statement in the program

D. That you have tested every combination of values of IF statements in the program

ANSWER: B

Explanation:

Complete statement and branch coverage means that the test suite has exercised every executable statement at least once and has exercised every branch outcome of each decision at least once. Statement coverage measures the proportion of executable statements that have been executed by tests. Branch coverage, also called decision coverage, measures whether each possible branch from a decision point has been taken; for example, both the true and false outcomes of a two-way decision must be exercised. Therefore, achieving complete coverage for both criteria requires tests that execute all statements while also traversing all decision outcomes. This is a structural, or white-box, coverage objective: it describes what the tests have executed in the program's control flow, rather than merely confirming that decision statements exist. In ISTQB terminology, branch coverage is calculated from the branches exercised divided by the total number of branches, expressed as a percentage. At 100% statement and branch coverage, all executable statements and all branches have been exercised. See the ISTQB Glossary definition of statement coverage and the ISTQB Glossary definition of branch coverage.

QUESTION NO: 18

During test process improvement it is recommended to use standards where possible. Standards originate from various sources and they cover different subjects in relation to testing Pick TWO sources of software standards, useful to software testing from the ones mentioned below. 1 credit

A. ISO 9126-1 'Software engineering- Product quality Part 1:
Quality model' is an international standard, that provides a basis on which to define quality assurance solutions.

B. ISA 4126-1 'Software engineering- Product quality Part 1:
Quality model' is an international standard, that provides a basis on which to define quality assurance solutions.

C. BS-7925-2 'Software testing. Software component testing is a national standard used internationally. It covers a number of testing techniques that may be useful both on component testing level and on system testing level.

D. SY-395-01 'Standard for East Coast Hospital software' is a regional standard adapted from a national one. Besides hospital software, this standard ought to be used also by other types of software system in the region.

E. IEEE 829 'standard for software test documentation' is an international standard to be following mandatory by all testing origination regardless of lifecycle models.

ANSWER: A C

Explanation:

ISO 9126-1 'Software engineering- Product quality Part 1:
Quality model' is an international standard, that provides a basis on which to define quality assurance solutions.

is a valid standards source relevant to testing because its product-quality model provides quality characteristics that can be used to derive, organize, and assess test objectives. Testers can apply such a model when defining quality criteria for a product and when ensuring that testing addresses relevant quality attributes. ISO/IEC 9126 was the quality-model standard used in earlier Foundation Level syllabi; it has since been superseded by the ISO/IEC 25000 SQuaRE family, including ISO/IEC 25010. The historical status does not change the intended answer in this question. See ISO's ISO/IEC 25010 standard page.

BS-7925-2 'Software testing. Software component testing is a national standard used internationally. It covers a number of testing techniques that may be useful both on component testing level and on system testing level.

is also a valid source. This British Standard documented specification-based test-design techniques, including equivalence partitioning, boundary value analysis, decision tables, and state transition testing. These techniques are applicable when testing individual components and remain useful at higher test levels when behaviour is evaluated through interfaces. The testing standard was influential in practical test-design guidance and is recognized in the historical ISTQB Foundation Level context. The current ISTQB Foundation Level syllabus continues to describe test techniques and their use across test levels; see the ISTQB Certified Tester Foundation Level page.

QUESTION NO: 19

Which of the following is most likely to be performed by developers?

A. Technical review of a functional specification.

B. Walkthrough of a requirements document.

C. Informal review of a program specification.

D. Static analysis of a software model.

ANSWER: D

Explanation:

Static analysis of a software model. is most likely to be performed by developers because it examines a development work product without executing the software. Developers commonly use static-analysis techniques and tools throughout implementation and design work to detect defects, inconsistencies, violations of coding or modelling rules, overly complex structures, and maintainability concerns as early as possible. Applying these checks directly to a software model allows issues to be identified before the model is transformed into code or used as the basis for further development, reducing the cost and effort of later correction.

The ISTQB CTFL syllabus distinguishes static testing from dynamic testing: static testing evaluates work products without running the software, and static analysis is a form of static testing usually supported by tools. Models are explicitly among the work products that can be statically analyzed. This activity fits naturally into developers' day-to-day responsibility for creating and maintaining technical solution artifacts, and it supports rapid feedback during development. See the ISTQB Certified Tester Foundation Level certification page and the ISTQB glossary entry for static analysis.

QUESTION NO: 20

Which THREE statements are valid objectives of acceptance testing? 1 credit

A. Building confidence that the system is fit for use

B. Assessing whether contractual or regulatory requirements have been met

C. Supporting a decision about whether the system should be accepted

D. Measuring statement coverage of source code

E. Replacing all component testing performed by developers

F. Finding every defect before release

ANSWER: A B C

Explanation:

Building confidence that the system is fit for use, **assessing whether contractual or regulatory requirements have been met**, and **supporting a decision about whether the system should be accepted** are valid objectives of acceptance testing. Acceptance testing is performed to evaluate whether the system is ready for delivery, deployment, or operational use from the perspective of relevant stakeholders.

Acceptance testing may be performed by users, customers, operations staff, or other authorized stakeholders, depending on the context. It is not primarily intended to measure structural coverage of source code or to replace component testing. See the ISTQB Glossary entry for acceptance testing and the ISTQB Certified Tester Foundation Level resources.

QUESTION NO: 21

Which of the following statements BEST describes the difference between testing and debugging?

A. Testing pinpoints (identifies the source of) the defects. Debugging analyzes the faults and proposes prevention activities.

B. Dynamic testing shows failures caused by defects. Debugging finds, analyzes, and removes the causes of failures in the software.

C. Testing removes faults. Debugging identifies the causes of failures.

D. Dynamic testing prevents causes of failures. Debugging removes the failures.

ANSWER: B

Explanation:

Dynamic testing shows failures caused by defects. Debugging finds, analyzes, and removes the causes of failures in the software. This correctly distinguishes the two related activities. Dynamic testing involves executing the software and comparing actual behavior or results with expected behavior or results. When the observed behavior differs from what was expected, this is a failure. Testing therefore provides information that can reveal failures and supports the investigation of their underlying defects, but it does not itself perform the corrective code change.

Debugging is the development activity undertaken after a failure has been observed or otherwise identified. It includes locating the fault or defect responsible for the failure, analyzing its cause, correcting it, and typically confirming that the correction has resolved the issue. The ISTQB Foundation Level syllabus explicitly describes debugging as finding, analyzing, and removing the causes of failures, while testing can show that failures are present. This distinction is important because testers and developers may collaborate closely, but their activities have different primary purposes in the defect-resolution workflow. See the ISTQB Certified Tester Foundation Level certification page and the ISTQB Glossary definition of debugging.

QUESTION NO: 22

Which statement BEST describes when test planning should be performed? [K1]

A. Test planning is performed only once, at the beginning of the life cycle, andgenerates a Master Test Plan

B. Test planning is performed at the beginning of the life cycle and again at the beginning of test execution

C. Test planning is performed at the beginning of the life cycle and again at every test level

D. Test planning is performed continuously in all life cycle processes and activities

ANSWER: D

Explanation:

Test planning is performed continuously in all life cycle processes and activities because it is not a one-time document-production task. Initial planning establishes the test objectives, scope, approach, resources, schedule, responsibilities, risks, communications, and entry and exit criteria. However, these planning decisions must be reviewed and adapted as the project gains information: requirements may change, risks may be reassessed, estimates and availability may vary, test progress may reveal impediments, and the chosen development life cycle may require adjustments to test activities. Continuous planning keeps testing aligned with the current product and project context, while test monitoring and control provide the information needed to update the plan. This applies across the relevant test levels and throughout the software development life cycle, rather than only at a particular milestone. The ISTQB Foundation Level syllabus describes test planning as an activity that establishes and updates the test plan, and identifies planning, monitoring, and control as closely related test-management activities. See the official ISTQB Certified Tester Foundation Level certification page and the ISTQB glossary entry for test planning.

QUESTION NO: 23

Which of the following are aids to good communication, and which hinder it?

i. Try to understand how the other person feels.

ii. Communicate personal feelings, concentrating upon individuals.

iii. Confirm the other person has understood what you have said and vice versa.

iv. Emphasize the common goal of better quality.

v. Each discussion is a battle to be won.

A. (i), (ii) and (iii) aid, (iv) and (v) hinder.

B. (iii), (iv) and (v) aid, (i) and (ii) hinder.

C. (i), (iii) and (iv) aid, (ii) and (v) hinder.

D. (ii), (iii) and (iv) aid, (i) and (v) hinder.

ANSWER: C

Explanation:

(i), (iii) and (iv) aid, (ii) and (v) hinder. Effective communication in testing depends on building shared understanding and maintaining a constructive working relationship. Trying to understand how another person feels promotes empathy, helps the tester choose an appropriate communication style, and reduces the likelihood that feedback will be received defensively. Confirming mutual understanding is also essential: feedback, questions, defect information, and agreed actions should be checked so that sender and receiver attach the same meaning to them. Emphasizing the common goal of better quality keeps discussions focused on the product and on collaboration between testers, developers, business representatives, and other stakeholders.

The ISTQB Foundation Level syllabus treats communication with stakeholders as a core tester skill, including communicating test information clearly and supporting collaboration across the team. A quality-focused, respectful exchange helps ensure that test results and concerns can be acted upon effectively. This aligns with the professional and collaborative approach expected of testers in the ISTQB Certified Tester Foundation Level certification. ISTQB's terminology and broader body of knowledge are maintained through its official glossary, supporting consistent communication among project participants.

QUESTION NO: 24

Which of the following would typically be identified using static analysis by tools? [K1]

A. Spelling mistake on an error message

B. A potential infinite loop

C. Memory leakage

D. A variable set to the wrong value

ANSWER: B

Explanation:

A potential infinite loop is correctly identified as something that static-analysis tools can typically detect. Static analysis examines source code, models control flow, and applies rules or data-flow analysis without executing the program. By following the possible paths through a loop and evaluating its controlling conditions and updates, a tool may flag code for which termination cannot be established, such as a loop whose condition may remain true or whose exit condition is not reachable. The finding is normally reported as a potential defect or warning because analysis is based on the code and its modeled inputs rather than an executed run. This makes static analysis especially useful early in development, when developers can detect control-flow anomalies before dynamic testing begins. In the ISTQB terminology, static analysis is the process of evaluating work products without execution, and static-analysis tools support detection of certain structural and control-flow issues in code. See the ISTQB Glossary definition of static analysis and the ISTQB Certified Tester Foundation Level information.

QUESTION NO: 25

When a defect is detected and fixed then the software should be retested to confirm that the original defect has been successfully removed. This is called:

A. Regression testing

B. Maintenance testing

C. Confirmation testing

D. None of the above

ANSWER: C

Explanation:

Confirmation testing is the correct term for testing performed after a reported defect has been fixed, specifically to establish that the original failure no longer occurs and that the fix has been successfully implemented. The tester repeats the relevant test(s), normally using the same conditions, inputs, and expected result that previously exposed the defect. Its focus is therefore direct evidence that the particular defect correction works as intended. This activity is often informally called “re-testing,” but the ISTQB glossary uses confirmation testing as the formal term. Confirmation testing is an important part of defect management because a developer’s indication that a defect is fixed does not by itself demonstrate that the corrected software behaves correctly in the test environment. The ISTQB Foundation Level syllabus distinguishes confirmation testing from the broader testing activities used to assess the effects of changes. For the official ISTQB definition of confirmation testing, see the ISTQB Glossary: confirmation testing. The concept is also covered in the ISTQB Certified Tester Foundation Level materials.

QUESTION NO: 26

During test process improvement it is recommended to use standards where possible. Standards originate from various sources and they cover different subjects in relation to testing Pick TWO sources of software standards, useful to software testing from the ones mentioned below. 1 credit

A. Quality model' is an international standard, that provides a basis on which to define quality assurance solutions.

B. Quality model' is an international standard, that provides a basis on which to define quality assurance solutions.

C. BS-7925-2 'Software testing. Software component testing is a national standard used internationally. It covers a number of testing techniques that may be useful both on component testing level and on system testing level.

D. SY-395-01 'Standard for East Coast Hospital software' is a regional standard adapted from a national one. Besides hospital software, this standard ought to be used also by other types of software system in the region.

E. IEEE 829 'standard for software test documentation' is an international standard to be following mandatory by all testing origination regardless of lifecycle models.

ANSWER: A C

Explanation:

The correct selections identify genuine standards that can provide useful foundations for a software test process. A quality model is an international standard-based resource for specifying, measuring, and evaluating software product quality characteristics. In testing, such a model helps teams derive quality-related test objectives and coverage considerations, including functional suitability, reliability, performance efficiency, usability, security, maintainability, portability, and compatibility. The ISO/IEC 25000 SQuaRE family provides the current framework for software product quality requirements and evaluation; ISO/IEC 25010 defines the quality model used for this purpose. See the ISO/IEC 25010 overview for the relevant product-quality model.

BS 7925-2, *Software testing — Software component testing*, is also a recognized national standard that has been influential beyond its country of origin. It documents specification-based test-design techniques, notably equivalence partitioning, boundary value analysis, decision table testing, and state transition testing. Although directed at component testing, these techniques can be selected and applied at other test levels when their assumptions fit the test basis and risks. This makes the standard a practical source when improving consistent test-design practice. The British Standards Institution catalogue records the standard as BS 7925-2.

QUESTION NO: 27

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

ANSWER: B

Explanation:

Auditing conformance to ISO9001 is the correct answer because this describes a quality-management-system compliance audit, rather than a configuration-management activity. ISO 9001 is a standard for organizational quality management systems. Audits against it assess whether an organization's quality-management processes conform to the requirements of that standard and are being effectively implemented. Such audits may consider configuration management as one process within the wider quality system, but they are not themselves configuration management.

In software testing, configuration management is concerned with establishing and maintaining the integrity of work products throughout the test process and product lifecycle. It supports the reliable identification, versioning, control, recording, storage, and retrieval of testware and other configuration items. This ensures that a test team can determine precisely which version of a test item, test environment component, test case, or document was used at a given time, and can reproduce testing when needed. The ISTQB Foundation Level syllabus identifies configuration management as an essential supporting process for test work products and traceability. See the ISTQB Certified Tester Foundation Level information and the ISO 9001 quality-management overview.

QUESTION NO: 28

Which of the following is not true of regression testing?

A. It can be carried out at each stage of the life cycle.

B. It serves to demonstrate that the changed software works as intended.

C. It serves to demonstrate that software has not been unintentionally changed.

D. It is often automated.

ANSWER: B

Explanation:

“It serves to demonstrate that the changed software works as intended.” is the statement that is not true of regression testing. Its purpose is confirmation testing: testing performed after a defect has been fixed or a change has been made to establish that the specific item affected now behaves as required. In practical terms, confirmation testing reruns tests that previously exposed the defect, or otherwise directly checks the implemented change against its acceptance criteria.

Regression testing has a different, broader objective. It is concerned with detecting unintended adverse effects in software that was not deliberately changed, including effects caused by changes to code, interfaces, configuration, environments, or dependencies. A change can correctly satisfy its own requirement while still causing an unexpected problem elsewhere; regression testing provides evidence about that continuing behaviour. The ISTQB glossary distinguishes confirmation testing as confirming the success of a corrective action, whereas regression testing identifies defects introduced or uncovered by changes. This distinction is central to the Foundation Level terminology and test-maintenance approach. See the ISTQB glossary entry for confirmation testing and the ISTQB glossary entry for regression testing.

QUESTION NO: 29

Which THREE types of issue can a static analysis tool identify in source code without executing the software? 1 credit

A. Unreachable code

B. Violations of coding standards

C. Data-flow anomalies

D. Response time during peak usage

E. User satisfaction with the screen layout

F. Failure of an external interface during execution

ANSWER: A B C

Explanation:

Unreachable code is a valid static-analysis finding because the tool can examine control flow and identify statements that cannot be executed. **Violations of coding standards** can also be detected automatically by comparing code against defined rules for naming, formatting, constructs, or prohibited practices. **Data-flow anomalies**, such as using a variable before it is assigned a value, are another common static-analysis finding.

Static analysis does not observe the system operating under realistic workload or determine whether a user can complete a business process successfully. Such concerns require dynamic testing or reviews involving relevant stakeholders. See the ISTQB Glossary definition of static analysis and the ISTQB Certified Tester Foundation Level resources.

QUESTION NO: 30

Which TWO statements are valid when applying a risk-based testing approach? 1 credit

A. Testing effort should be prioritized according to the assessed level of product risk

B. Risk assessments should be reviewed when new information becomes available

C. All identified risks must receive the same amount of testing

D. Risk-based testing guarantees that all defects will be detected

E. Exit criteria are unnecessary when risk-based testing is used

ANSWER: A B

Explanation:

Testing effort should be prioritized according to the assessed level of product risk is correct. Areas with higher risk normally receive more test effort, more thorough coverage, or earlier testing than areas with lower risk.

Risk assessments should be reviewed when new information becomes available is also correct. Product risks can change as requirements are clarified, design decisions are made, defects are found, or operational information becomes available. The test approach and priorities should be adjusted when the risk assessment changes.

Risk-based testing cannot prove that all defects have been found, and it does not remove the need for test completion criteria or stakeholder decisions about residual risk. See the ISTQB Glossary entry for risk-based testing and the ISTQB Certified Tester Foundation Level resources.

QUESTION NO: 31

Which of the following options BEST explain the pesticide paradox principle of testing?

A. If we do not regularly review and revise our tests, we'll stop finding defects

B. Repeatedly running a set of tests will ensure that a system is defect free

C. Defects are, paradoxically, often contained in a small number of modules

D. Testing, like spraying pesticide, is an effective bug / defect removal activity

ANSWER: A

Explanation:

If we do not regularly review and revise our tests, we'll stop finding defects correctly expresses the pesticide paradox. The principle compares a test suite with pesticides: repeatedly applying the same pesticide eventually becomes less effective because the targeted pests adapt. Similarly, when the same tests are executed repeatedly without change, they continue to demonstrate the behaviors and defects they were originally designed to detect, but become progressively less likely to reveal new defects. The application may evolve, developers may become familiar with the existing checks, and untested conditions, data combinations, interfaces, and workflows can remain undiscovered.

To preserve the effectiveness of testing, testers should regularly review test conditions and test cases, revise existing tests when requirements or risks change, and add new tests based on defect analysis, exploratory testing, operational usage, and changes to the product. This does not mean that established regression tests should be discarded; they remain valuable for detecting unintended changes in previously verified behavior. Rather, the overall test set must evolve so that it continues to challenge the system in meaningful and varied ways. The pesticide paradox is one of the fundamental testing principles described in the ISTQB Certified Tester Foundation Level syllabus. See the ISTQB CTFL certification page and the ISTQB Certified Tester Foundation Level information.

QUESTION NO: 32

Which THREE activities are part of test implementation? 1 credit

A. Creating test procedures

B. Preparing test data

C. Establishing the test environment

D. Identifying test conditions from the test basis

E. Evaluating whether exit criteria have been met

F. Archiving testware at the end of the test project

ANSWER: A B C

Explanation:

Creating test procedures is part of test implementation because test procedures define the sequence of actions needed to execute test cases. **Preparing test data** is also part of implementation, as suitable data must be made available to execute the tests and verify expected results. **Establishing the test environment** is correct because the required hardware, software, tools, interfaces, configurations, and access arrangements must be prepared before planned tests can be executed.

Identifying test conditions belongs primarily to test analysis, while evaluating completion criteria and producing a test completion report occur during test completion. See the ISTQB Glossary entry for test implementation and the ISTQB Certified Tester Foundation Level resources.

QUESTION NO: 33

Which of the following would be the TWO most appropriate examples of entry criteria documented in the system integration test plan? 2 credits

A. The percentage of decision coverage achieved during unit testing.

B. The availability of the latest version of the capture-replay tool (for testing the interface with the newly developed test management tool).

C. The sign-off of a performance test software release note (test item transmittal report) by both development and testing showing that system performance meets the specified acceptance criteria.

D. The percentage of acceptance test procedures scheduled for execution.

E. The percentage of requirements coverage achieved during system integration test.

ANSWER: A B

Explanation:

Entry criteria define the preconditions that should be satisfied before a test level or test activity starts. For system integration testing, the integrated components must be sufficiently mature to make testing meaningful, and the required test environment, including relevant tools, must be available and usable. Therefore, *The percentage of decision coverage achieved during unit testing.* is an appropriate entry criterion because it provides objective evidence that component-level testing has reached the agreed level of completion before components are combined and their interfaces are tested. This reduces the risk that basic component defects will obscure integration-related failures.

The availability of the latest version of the capture-replay tool (for testing the interface with the newly developed test management tool). is also an appropriate entry criterion. The specified tool is a necessary part of the planned system integration test environment for exercising and observing the interface. Confirming its availability and correct version before execution begins helps ensure that the planned tests can be run reliably and that results can be captured as intended. The ISTQB CTFL syllabus describes entry criteria as conditions for beginning a test activity and includes availability of the test environment and other testware among relevant considerations. See the ISTQB Certified Tester Foundation Level page and the ISTQB Glossary definition of entry criteria.

QUESTION NO: 34

Which of the following would NOT be a common metric used for monitoring test preparation and execution?

A. Number of Test cases passed and failed

B. Percentage of planned test cases designed

C. Number of test plan review comments

D. Percentage of tasks complete in test environment preparation

ANSWER: C

Explanation:

Number of test plan review comments is the correct choice because it measures findings from a review of a test-work-product, rather than progress in preparing for or executing tests. Review comments can be useful qualitative input when improving a test plan, clarifying its content, or assessing the effectiveness of a review activity. However, they do not directly indicate the operational status of test design, test-environment readiness, or test execution against the test schedule.

For test monitoring and control, the ISTQB Foundation Level syllabus identifies progress-oriented metrics that provide visibility into whether the planned testing work is advancing and whether it is on track. Such monitoring supports timely decisions about resources, priorities, schedule adjustments, and actions needed to meet test objectives. A count of comments from a test-plan review may vary with the review scope, reviewers' style, and the maturity of the document, so it is not normally a core metric for monitoring preparation and execution progress. See the ISTQB Certified Tester Foundation Level certification page and the ISTQB glossary for the official terminology and syllabus resources.

QUESTION NO: 35

System testing is:

A. Used to search for defects in software modules that are separately testable.

B. The responsibility of the users of a system.

C. Concerned with the behavior of a whole system/product as defined by the scope of a development project.

D. Triggered by modifications, migration or retirement of the software system.

ANSWER: C

Explanation:

"Concerned with the behavior of a whole system/product as defined by the scope of a development project." is correct because system testing evaluates the integrated system or product as a whole. At this test level, the test object includes the components and interfaces that together deliver the system's intended behavior within the project's defined scope. Testing focuses on whether the complete system satisfies specified functional and non-functional requirements, including end-to-end workflows and relevant quality characteristics. The precise scope can vary: it may be an entire product, a subsystem, or another defined system boundary, depending on what the development project is delivering. System testing is therefore distinguished by its whole-system perspective rather than by testing an isolated component or module. This wording aligns directly with the ISTQB CTFL syllabus definition: system testing is concerned with the behavior of a whole system or product, considered within the scope of a development project. See the ISTQB Certified Tester Foundation Level certification page and the ISTQB Glossary entry for system testing.

QUESTION NO: 36

Which TWO activities are examples of shift-left practices? 1 credit

A. Reviewing acceptance criteria before implementation begins

B. Deriving tests from requirements while they are being elaborated

C. Executing acceptance tests only after deployment to production

D. Postponing test-environment preparation until system testing starts

E. Deferring defect analysis until test completion

ANSWER: A B

Explanation:

Reviewing acceptance criteria before implementation begins and **deriving tests from requirements while they are being elaborated** are examples of shift-left practices. Shift-left means moving testing activities earlier in the software development lifecycle. Early involvement helps detect defects in work products before they are implemented and gives testers an opportunity to improve clarity, testability, and shared understanding.

Executing acceptance tests only after deployment and postponing test-environment preparation until system testing are not shift-left practices. They delay feedback and increase the likelihood that problems will be found when correction is more costly. See the ISTQB Certified Tester Foundation Level resources and the ISTQB Glossary entry for shift left.