

# SNS COLLEGE OF TECHNOLOGY

## DEPARTMENT OF MCA

### CA717 – SOFTWARE TESTING AND QUALITY ASSURANCE – IV SEMESTER

#### TWO MARKS QUESTIONS & ANSWERS

---

#### UNIT - I: TESTING ENVIRONMENT AND TEST PROCESSES

**(1) Define software Testing.**

Testing can be described as a process used for revealing defects in software, and for establishing that the software has attained a specified degree of quality with respect to selected attributes.

**(2) Define Debugging.**

Debugging, or fault localization is the process of

- Locating the fault or defect.
- Repairing the codes.
- Retesting the code.

**(3) What is an Error.**

An error is mistake or misconception or misunderstanding on the part of a software developer.

**(4) What is meant by Faults (Defects).**

A fault is introduced into the software as the result of an error. It is an anomaly in the software that may cause nit to behave incorrectly, and not according to its specification.

**(5) Define failures.**

A failure is the inability of a software or component to perform its required functions within specified performance requirements.

**(6) Define Test Cases.**

A test case in a practical sense is a test related item which contains the following information.

- A set of test inputs. These are data items received from an external source by the code under test. The external source can be hardware, software, or human.
- Execution conditions. These are conditions required for running the test, for example, a certain state of a database, or a configuration of a hardware device.
- Expected outputs. These are the specified results to be produced by the code under test.

**(7) Write short notes on Test, Test Set, and Test Suite.**

A Test is a group of related test cases, or a group of related test cases and test Procedure.

A group of related test is sometimes referred to as a test set. A group of related tests that are associated with a database, and are usually run together, is sometimes referred to as a Test Suite.

**(8) Define Validation.**

Validation is the process of evaluating a software system or component during, or at the end of, the development cycle in order to determine whether it satisfies specified requirements.

**(9) Define Verification.**

Verification is the process of evaluating a software system or component to determine whether the product of a given development phase satisfy the conditions imposed at the start of that phase.

**(10) Differentiate between verification and validation?**

<b>Verification</b>	<b>Validation</b>
Verification is the process of evaluating software system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.	Validation is the process of evaluating software system or component during or at the end of the , the development phase satisfy the conditions imposed at the start of that phase.
Verification is usually associated with activities such as inspections and reviews of the s/w deliverables.	Validation is usually associated with Traditional execution _based testing, i.e., Exercising the code with test case.

**(11) Differentiate between testing and debugging**

<b>Testing</b>	<b>Debugging</b>
Testing as a dual purpose process <ul style="list-style-type: none"><li>• Reveal defects</li><li>• And to evaluate quality attributes</li></ul>	Debugging or fault localization is the process of <ul style="list-style-type: none"><li>• Locating the fault or defect</li><li>• Repairing the code, and</li><li>• Retesting the code</li></ul>

**(12) Define Test Oracle.**

Test Oracle is a document, or a piece of software that allows tester to determine whether a test as been passed or failed.

**(13) Define Test Bed.**

A test bed is an environment that contains all the hardware and software needed to test a software component or a software system.

#### **(14) Define Software Quality.**

Quality relates to the degree to which a system, system component, or process meets specified requirements. Quality relates to the degree to which a system, system component, or process meets customer or user needs, or expectations.

#### **(15) List the Quality Attributes.**

- Correctness
- Reliability
- Usability
- Integrity
- Portability
- Maintainability
- Interoperability

#### **(16) Define reviews.**

A review is a group meeting whose purpose is to evaluate a software artifact or a set of Software artifacts. Review and audit is usually conducted by a SQA group.

#### **(17) Define Errors.**

An error is a mistake, misconception, or misunderstanding on the part of a software developer.

#### **(18) Define Faults.**

A fault (defect) is introduced into the software as the result of an error. It is an anomaly in the software that may cause it to behave incorrectly, and not according to its specification.

#### **(19) Define Failures.**

A failure is the inability of a software system or component to perform its required functions within specified performance requirements.

#### **(20) Define Test case.**

A test case in the practical sense is a test- related item which contain the following information:

- A set of test inputs.
- Execution conditions.
- Expected outputs.

# SNS COLLEGE OF TECHNOLOGY

## DEPARTMENT OF MCA

### CA717 – SOFTWARE TESTING AND QUALITY ASSURANCE – IV SEMESTER

#### TWO MARKS QUESTIONS & ANSWERS

---

#### UNIT - II: TESTING TECHNIQUES AND LEVELS OF TESTING

**(1) List the two basic Testing strategies.**

- Black box testing.
- White box testing.

**(2) What is Black-box testing?**

- Knowing the specified function that a product has been designed to perform, test to see if that function is fully operational and error free
- Includes tests that are conducted at the software interface
- Not concerned with internal logical structure of the software

**(3) What is White-box testing?**

- Knowing the internal workings of a product, test that all internal operations are performed according to specifications and all internal components have been exercised
- Involves tests that concentrate on close examination of procedural detail
- Logical paths through the software are tested
- Test cases exercise specific sets of conditions and loops

**(4) What are the knowledge sources for Black box testing?**

- Requirements
- Document specification
- Domain knowledge
- Defect analysis data

**(5) What are the knowledge sources for White box testing?**

- High level design
- Detailed design
- Control flow graphs
- Cyclomatic complexity

**(6) List the methods of Black box testing.**

- Equivalence class partitioning
- Boundary value analysis
- State transition testing
- Cause and effect graphing
- Error guessing

**(7) List the methods of White box testing.**

- Statement testing
- Branch testing
- Path testing
- Data flow testing
- Mutation testing
- Loop testing

**(8) What are the basic primes for all structured program.**

- Sequential ( e.g., Assignment statements)
- Condition (e.g., if/then/else statements)
- Iteration (e.g., while, for loops)

**(9) Define Random testing.**

Each software system or module has an input domain from which test input data is selected. If a tester randomly selects input from the domain, this is called Random testing.

**(10) What is Equivalence class partitioning.**

If a tester is viewing the software-under-test as a black box with well defined inputs and outputs, a good approach to selecting test inputs is to use a method called Equivalence class partitioning.

**(11) List the advantages of Equivalence class partitioning.**

- It eliminates the need for exhaustive testing, which is not feasible.
- It guides a tester in selecting a subset of test inputs with a high probability of detecting a defect.
- It allows a tester to cover a larger domain of inputs/outputs with a smaller subset selected from an Equivalence class.

**(12) What is Cause effect graphing?**

It is a technique that can be used to combine conditions and derive an effective set of test cases that may disclose inconsistencies in a specification.

**(13) Define State.**

A state is an internal configuration of a system or component. It is defined in terms of values assumed at a particular time for the variables that characterize the system or component.

**(14) What is meant by Finite-state machine?**

It is an abstract machine that can be represented by a state graph having a finite number of states and a finite number of transitions between states.

**(15) Define Usage profiles.**

Usage profiles are characterizations of the population of intended uses of the software in its intended environment.

**(16) What is Certification?**

Certification refers to third-party assurance that a product, process, or service meets a specific set of requirements.

**(17) What is Test data set?**

A test data set is statement, or branch, adequate if a test set T for program P causes all the statements, or branches, to be executed respectively.

**(18) Define Path.**

A path is a sequence of control flow nodes usually beginning from the entry node of a graph through to the exit node.

**(19) List the two major assumptions in Mutation testing.**

- The component programmer hypothesis
- The coupling effect

**(20) Define Error guessing.**

Error guessing approach is based on the testers/developers past experience with code similar to code-under-test, and their intuition as to where defects may lurk in the code.

**(21) What is the goal of smart tester?**

The goal of the smart tester is to understand the functionality, input/output domain, and the environment of use for the code being tested.

**(22) List the different levels of testing.**

- Unit test
- Integration test
- System test
- Acceptance test.

**(23) Define Unit Testing**

A unit is the smallest possible testable software component that can be characterized in several ways.

**(24) List the components suitable for unit test.**

- Procedures and functions
- Classes/objects and methods
- Procedure-sized reusable components.

**(25) List the phases in the unit test planning.**

- Phase 1: Describe unit test approach and risks.
- Phase 2: Identify unit features to be tested.
- Phase 3: Add levels of detailed to the plan.

**(26) What is Test harness?**

The auxiliary code developed to support to testing of units and components is called a test harness. The harness consists of drivers that call the target code and stubs that represent modules it calls.

**(27) List the major goals of Integration test.**

- To detect defects that occurs on the interfaces of units.
- To assemble the individual units into working subsystems and the finally a complete system that is ready for system test

**(28) What are the Integration strategies?**

- Top Down: In this strategy integration of the module begins with testing the upper level modules.
- Bottom Up: In this strategy integration of the module begins with testing the lowest level modules.

**(29) What is the advantage of Bottom up integration?**

Bottom-up integration has the advantage that the lower-level modules are usually well tested early in the integration process. This is important if these modules are candidates for reuse.

**(30) What is meant by a stub and driver?**

**Driver**

A simple main program that accepts test case data, passes such data to the component being tested, and prints the returned results.

**Stubs**

Serve to replace modules that are subordinate to (called by) the component to be tested.

**(31) List the several types of system tests.**

- Functional testing
- Performance testing

- Stress testing
- Configuration testing
- Security testing
- Recovery testing

**(32) What are the two major requirements in the Performance testing?**

- Functional Requirement: User describes what functions the software should perform. We test for compliance of the requirement at the system level with the functional based system test.
- Quality Requirement: They are non functional in nature but describe quality levels expected for the software.

**(33) Define stress Testing.**

When a system is tested with a load that causes it to allocate its resources in maximum amounts. It is important because it can reveal defects in real-time and other types of systems.

**(34) Define Alpha and Beta Test.**

Alpha test developer's to use the software and note the problems. Beta test who use it under real world conditions and report the defect to the Developing organization.



# SNS COLLEGE OF TECHNOLOGY

## DEPARTMENT OF MCA

### CA717 – SOFTWARE TESTING AND QUALITY ASSURANCE – IV SEMESTER

#### TWO MARKS QUESTIONS & ANSWERS

---

#### UNIT - III: INCORPORATING SPECIALIZED TESTING RESPONSIBILITIES

**(1) What is a Client/Server system?**

In Client / Server system, application software resides on the client workstations. The application server handles processing requests. The back-end processing (typically a mainframe or super-minicomputer) handles processing such as batch transactions that are accumulated and processed together at one time on a regular basis.

**(2) List out the Software Development Process Maturity Levels.**

- Ad hoc
- Repeatable
- Consistent
- Measured
- Optimized

**(3) Define Rapid Application Development.**

Rapid application development (RAD) is an effective software development paradigm that provides a systematic and automatable means of developing a software system under circumstances where initial requirements are not well known or where requirements change frequently during development.

**(4) What is meant by spiral testing?**

The RAD testing strategy, termed spiral testing, remains iterative and parallels the RAD process. Spiral testing characterizes the varying types of RAD iterations by tailoring the testing process to account for these differences.

**(5) How testing is performed on multiplatform software systems?**

Software designed to run on more than one platform must undergo two tests. The first test is to validate that the software performs its intended functions. The second test is that the software will perform in the same manner regardless of the platform on which it is executed.

**(6) What are the distinct characteristics of a multiplatform software?**

Each platform on which software is designed to execute operationally may have slightly different characteristics. These distinct characteristics include various operating systems, hardware configurations, operating instructions, and supporting software, such as database management systems.

**(7) List out the major concerns for testing in a multiplatform environment.**

- The platforms in the test lab will not be representative of the platforms in the real world.
- The software will be expected to work on platforms not included in the test labs.
- The supporting software on various platforms is not comprehensive

**(8) What is a vulnerability?**

A vulnerability is a weakness in an information system. It is the point at which software systems are easiest to penetrate. Understanding the vulnerabilities helps in designing security for information systems.

**(9) Name some of the vulnerable locations in which the software security is compromised.**

The following list ranks the nine functional locations according to vulnerability:

- Data- and report-preparation facilities
- Computer operations
- Non-IT areas
- Online terminal systems
- Programming offices
- Handling areas for online data preparation and output reports
- Digital media storage facilities
- Online operations
- Central processors

**(10) Define a data warehouse.**

A data warehouse is a central repository of data made available to users. The centralized storage of data provides significant processing advantages but at the same time raises concerns of the data's security, accessibility, and integrity.

**(11) Write a note on various data warehouse activity processes to test.**

- Organizational Process
- Data Documentation Process
- System Development Process
- Access Control Process
- Data Integrity Process
- Operations Process
- Backup/Recovery Process

**(12) What is a web-based system?**

In web-based systems, the browsers reside on client workstations. These client workstations are networked to a web server, either through a remote connection or through a network such as a local area network (LAN) or wide area network (WAN). As the web server receives and processes requests from the client workstation, requests may be sent to the application server to perform actions such as data queries, electronic commerce transactions, and so forth.

**(13) List out some of the concerns when conducting web-based testing.**

- Browser compatibility
- Functional correctness
- Integration
- Usability
- Security
- Performance
- Verification of code

**(14) Write the categories of the more common web-based test tools.**

- HTML tools
- Site validation tools
- Load/stress testing tools
- Test case generators.

**(15) What are the various levels of object oriented testing?**

In object-oriented systems, testing encompasses three levels, namely, unit testing, subsystem testing, and system testing.

**(16) List out some of the Object-Oriented testing techniques.**

- State model based testing : This encompasses state coverage, state transition coverage, and state transition path coverage.
- Use case based testing : Each scenario in each use case is tested.
- Class diagram based testing : Each class, derived class, associations, and aggregations are tested.
- Sequence diagram based testing : The methods in the messages in the sequence diagrams are tested.

**(17) Name some of the typical OO software characteristics that impact testing.**

- State dependent behavior
- Encapsulation
- Inheritance
- Polymorphism
- Dynamic binding
- Abstract and generic classes
- Exception handling

# SNS COLLEGE OF TECHNOLOGY

## DEPARTMENT OF MCA

### CA717 – SOFTWARE TESTING AND QUALITY ASSURANCE – IV SEMESTER

#### TWO MARKS QUESTIONS & ANSWERS

---

#### UNIT - IV: TOOLS TO AUTOMATE TESTING

**(1) What is automated software testing?**

Automated software testing is a process in which software tools execute pre-scripted tests on a software application before it is released into production.

**(2) Define test automation.**

In software testing, test automation is the use of special software (separate from the software being tested) to control the execution of tests and the comparison of actual outcomes with predicted outcomes.

**(3) List out the advantages and disadvantages of Automated testing.**

**Advantages:**

- Reliable: eliminating human error
- Reusable
- Better Quality Software
- Fast
- Cost Reduction

**Disadvantages:**

- High investment is needed in the tools and training
- High man power requirement for test preparations
- A lot of testing areas left uncovered

**(4) Differentiate Manual Testing and Automated Testing**

Manual Testing	Automated Testing
<ul style="list-style-type: none"><li>• Manual Testing is boring</li><li>• Manual Testing is not reusable</li><li>• Manual Testing has a high risk of missing out on something</li><li>• Manual Tests do not drive design</li><li>• Manual Tests do not provide a safety-net</li><li>• Manual Tests have no training value</li></ul>	<ul style="list-style-type: none"><li>• Automated Tests on the other hand are code</li><li>• Automated Tests are completely reusable</li><li>• Automated Tests have zero risk of missing out a pre-decided test</li><li>• Automated Tests and TDD / Test-First development drive design</li><li>• Automated Tests provide a safety-net for refactoring / additions</li><li>• Automated Tests act as documentation</li></ul>

**(5) What is the need for testing automation?**

- Speed - Automation scripts run very fast when compared to human users.
- Reliable - Tests perform precisely the same operations each time they are run, thereby eliminating human error.
- Repeatable - We can test how the application reacts after repeated execution of the same operations.
- Programmable - We can program sophisticated tests that bring out hidden information.
- Comprehensive - We can build a suite of tests that covers every feature in our application.
- Reusable - We can reuse tests on different versions of an application, even if the user interface changes.

**(6) List the different types of system testing.**

- Functional testing
- Performance testing
- Stress testing
- Configuration testing
- Security testing
- Recovery testing
- The other types of system Testing are,
- Reliability & Usability testing.

**(7) Define load generator and Load.**

An important tool for implementing system tests is a load generator. A load generator is essential for testing quality requirements such as performance and stress

A load is a series of inputs that simulates a group of transactions. A transaction is a unit of work seen from the system user's view. A transaction consist of a set of operation that may be perform by a person , s/w system or device that is outside the system.

**(8) What are the two major requirements in the Performance testing.**

- Functional Requirement: User describe what functions the software should perform. We test for compliance of the requirement at the system level with the functional based system test.
- Quality Requirement: They are nonfunctional in nature but describe quality.

**(9) Define stress Testing.**

When a system is tested with a load that causes it to allocate its resources in maximum amounts. It is important because it can reveal defects in real-time and other types of systems.

### **(10) Define Test incident Report**

The tester should record in a test incident report (sometimes called a problem report). It is used to find out any event that occurs during the execution of the tests that is unexpected, unexplainable, and that requires a follow-up investigation.

### **(11) Define Test Log.**

The Test log should be prepared by the person executing the tests. It is a diary of the events that take place during the test. It supports the concept of a test as a repeatable experiment.

### **(12) What are the two types of testcases?**

- Manual
- Automated

### **(13) What is LoadRunner?**

LoadRunner is an industry-leading performance and load testing product by Hewlett-Packard (since it acquired Mercury Interactive in November 2006) for examining system behavior and performance, while generating actual load.

### **(14) What is SilkTest?**

Silk Test is a tool specifically designed for doing REGRESSION AND FUNCTIONALITY testing. It is developed by Segue Software Inc. Silk Test is the industry's leading functional testing product for e-business applications, whether Window based, Web, Java, or traditional client/server-based.

### **(15) What is WinRunner?**

WinRunner is a test automation tool, designed to help customers save testing time and effort by automating the manual testing process

- manual process: perform operations by hand, visually check results, and log results by hand
- automated process: create a test script that will perform the same operations as a Human operator, check the same information, and create a summary report showing the test status

### **(16) What is JMeter?**

JMeter is an Apache Jakarta project that can be used as a load testing tool for analyzing and measuring the performance of a variety of services, with a focus on web applications.

# SNS COLLEGE OF TECHNOLOGY

## DEPARTMENT OF MCA

### CA717 – SOFTWARE TESTING AND QUALITY ASSURANCE – IV SEMESTER

#### TWO MARKS QUESTIONS & ANSWERS

---

#### UNIT - V: SOFTWARE TESTING AND QUALITY METRICS

**(1) Define quality.**

Quality is the degree of goodness of a product or service or perceived by the customer. Quality concept is the way business organizations perform their business activities that focuses on two things.

**(2) What are the different views of quality?**

- The transcendent view
- The product-based view
- The user-based view
- The manufacturing view
- The value-based view

**(3) What is quality cost?**

The expenditure incurred by the producer, by the user and by the community associated with the product and service quality.

**(4) What are the building blocks of total quality management?**

Management commitment, customer focus, process focus, continuous improvement, benchmarking, teams, supplier teaming, employee involvement, training of employees, inventory management, communication.

**(5) What are the 7 QC tools?**

1. Check sheet
2. Graphs
3. Histograms
4. Pareto charts
5. Cause and effect diagrams
6. Scatter charts
7. Control charts

**(6) Define Quality Function Deployment.**

QFD is a systematic and structured planning tool which is used to convert the voice of the customer into appropriate requirements.

**(7) What are the uses of arrow diagram?**

Arrow diagrams help in scheduling the activities so that any project can be completed in the best possible way. It will provide a time sequential action plan for all the tasks that are to be carried out in a project.

**(8) What are the different dimensions of quality?**

Performance, features, reliability, conformance, durability, serviceability, aesthetics, perceived quality and reputation, response.

**(9) Define TQM.**

Total quality management is customer oriented management philosophy and strategy. It is centered on quality so as to result in customer delight.

**(10) What is quality improvement?**

QM aims at attaining unprecedented levels of performance which are significantly better than the past level.

**(11) What is quality management?**

Quality management comprises all activities of the overall management function that determine the quality policy, objectives and responsibilities, and implement them by means such as other quality planning, quality control, etc.

**(12) What are the benefits of QFD?**

1. Reduces product development time
2. Reduce engineering costs
3. Reduces the time to market.
4. Improves design quality
5. Improves customer satisfaction.
6. Reduce quality costs. etc.

**(13) What is software availability?**

Software availability is the probability that a program is operating according to requirements at a given point in time.

**(14) What is Six Sigma?**

- Six Sigma - A highly disciplined process that enables organizations deliver nearly perfect products and services.
- A statistical concept that measures a process in terms of defects – at the six sigma level, there 3.4 defects per million opportunities.

**(15) List out the Six Sigma levels.**

Sigma Level ( Process Capability)	Defects per Million Opportunities
2	308,537
3	66,807
4	6,210
5	233
6	3.4



**(16) Define Taguchi loss function.**

- Taguchi defines quality as “the loss imparted by the product to society from the time the product is shipped”.
- This loss includes costs to operate, failure to function, maintenance and repair costs, customer dissatisfaction, injuries caused by poor design, and similar costs.
- Defective products / parts that are detected, repaired, reworked, or scrapped before shipment are not considered part of this loss.

**(17) Differentiate traditional and Taguchi’s approach.**

Taguchi’s	Traditional
When a product moves from its Target will cause the loss even if the product lies or not within Limits	There is Good or Bad Products only as per Limits

**(18) What is FMEA?**

- FMEA is an acronym for Failure Mode and Effect Analysis.
- A systemized group of activities designed to:
  - recognize and evaluate the potential failure of a product/process and its effects
  - identify actions which could eliminate or reduce the chance of potential failure
  - document the process

**(19) What is a metric?**

A metric is a verifiable measure stated in either quantitative or qualitative terms.

Examples :

- “95 percent inventory accuracy”
- “as evaluated by our customers, we are providing above-average service”

**(20) Define software quality metrics.**

The definition of software quality consists of two levels: intrinsic product quality and customer satisfaction. The metrics are:

- Mean time to failure
- Defect density
- Customer problems
- Customer satisfaction.

Intrinsic product quality is usually measured by the number of bugs (functional defects) in the software or by how long the software can run before encountering a crash. In operational definitions, the two metrics are defect density (rate) and mean time to failure (MTTF).