3. Post-implementation goals
    - Reduced maintenance cost
    - Improved testing process

- Testing should be performed with a mindset of finding bugs. This suspicious strategy (destructive approach) helps in finding more and more bugs.
- Software testing is a process that detects important bugs with the objective of having better quality software.
- Exhaustive testing is not possible due to the following reasons:
    - It is not possible to test every possible input, as the input domain is too large.
    - There are too many possible paths through the program to test.
    - It is difficult to locate every design error.
- Effective software testing, instead of complete or exhaustive testing, is adopted such that critical test cases are covered first.
- There are different views on how to perform testing which have been categorized as schools of software testing, namely (i) analytical school, (ii) standard school, (iii) quality school, (iv) context school, and (v) agile school.
- Software testing is a complete process like software development.

EXERCISES

MULTIPLE CHOICE QUESTIONS

1. Bug discovery is a _____ goal of software testing.
    (a) Long-term
    (b) Short-term
    (c) Post-implementation
    (d) All
2. Customer satisfaction and risk management are _____ goals of software testing.
    (a) Long-term
    (b) Short-term
    (c) Post-implementation
    (d) All
3. Reduced maintenance is a _____ goal of software testing.
    (a) Long-term
    (b) Short-term
    (c) Post-implementation
    (d) All

4. Software testing produces _____.
   (a) Reliability
   (b) Quality
   (c) Customer Satisfaction
   (d) All
5. Testing is the process of _____ errors.
   (a) Hiding
   (b) Finding
   (c) Removing
   (d) None
6. Complete testing is _____.
   (a) Possible
   (b) Impossible
   (c) None
7. The domain of possible inputs to the software is too _____ to test.
   (a) Large
   (b) Short
   (c) none
8. The set of invalid inputs is too _____ to test.
   (a) Large
   (b) Short
   (c) none
9. Race conditions are among the _____ tested.
   (a) Most
   (b) Least
   (c) None
10. Every design error _____ be found.
    (a) Can
    (b) Can definitely
    (c) Cannot
    (d) None

Review Questions

1. How does testing help in producing quality software?
2. 'Testing is the process of executing a program with the intent of finding errors.' Comment on this statement.
3. Differentiate between effective and exhaustive software testing.

- Software testing methodology is the organization of software testing by means of which the test strategy and test tactics are achieved.
- Testing strategy is planning the whole testing process into a well-planned series of steps.
- A test strategy matrix identifies the concerns that will become the focus of test planning and execution. The matrix is prepared using the test factors and test phase.
- The test strategy has two parts: Verification and Validation
- Verification is to check the software with its specification at every development phase such that any defect can be detected at an early stage of testing.
- Validation is to test the software as a whole in conformance with customer expectations.
- V-testing model emphasizes the concept of early testing. In this model, the scope of verification is more in the early phases and decreases slowly as one module is ready. After this, the scope of validation increases.
- There are three validation activities, generally known as levels of testing: Unit testing, Integration testing, and System testing.
- There are two types of testing techniques: Static and Dynamic testing.
- Static testing is to test the software and its documents without executing the software.
- Dynamic testing is to test the software by executing it.

EXERCISES

MULTIPLE CHOICE QUESTIONS

1. Fault is synonymous with the word _____.
   (a) Failure
   (b) Defect
   (c) Error
   (d) All of the above
2. The inability of a system or component to perform a required function according to its specification is called as _____.
   (a) Failure
   (b) Bug
   (c) Error
   (d) None of the above
3. Testware includes _____.
   (a) test planning document
   (b) test data
   (c) test specifications
   (d) All of the above

4. Symptom(s) associated with a failure that alerts the user to the occurrence of a failure is called _____.
   (a) Bug
   (b) Error
   (c) Defect
   (d) Incident
5. Testing process starts as soon as the _____ for the system are prepared.
   (a) Design
   (b) Coding
   (c) Specifications
   (d) None of the above
6. Testing strategy should start at the _____ module level and expand towards the whole program.
   (a) Smallest
   (b) Largest
   (c) None of the above
7. Testing is a _____ process.
   (a) Intuitive
   (b) Random
   (c) Planned
   (d) None of the above
8. Planning the whole testing process into a well-planned series of steps is called _____.
   (a) Test strategy matrix
   (b) Test factor
   (c) Test phase
   (d) Test strategy
9. The test strategy matrix is prepared using the _____.
   (a) test planning and execution
   (b) test factor and test phase
   (c) test factor
   (d) test phase
10. The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase is called _____.
   (a) Verification
   (b) Validation
   (c) SDLC
   (d) None of the above

11. The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies the specified requirements is called _____.
    (a) Verification
    (b) Validation
    (c) SDLC
    (d) None of the above
12. In the early stages of SDLC, testing comprises more _____ activities and towards the later stages, the emphasis is on the _____ activities.
    (a) verification, validation
    (b) validation, verification
    (c) integration, coding
    (d) None
13. Technique for assessing the structural characteristics of source code, design specifications, or any notational representation that conforms to well-defined syntactic rules is called _____.
    (a) Dynamic testing
    (b) Static Testing
    (c) Black-Box Testing
    (d) None of the above
14. Every design feature and its corresponding code is checked logically with every possible path execution in _____.
    (a) Black-box testing
    (b) White-box testing
    (c) Testing tool
    (d) None of the above

Review Questions

1. Differentiate error, bug, defect, fault, failure, testware, and incident, giving examples of each.
2. How many states does a bug have?
3. Take a live project and demonstrate the fact that 'Bugs in earlier stages go undetected and propagate'.
4. Collect examples of some commercial products that prove the statement 'Bug affects the economics of Software Testing'
5. Give examples of each category of bug classified based on criticality.
6. Give examples of each category of bug classified based on SDLC.
7. How do you support destructive approach for software testing?

EXERCISES

MULTIPLE CHOICE QUESTIONS

1. Which of the following is true?
   (a) Testing is performed after coding.
   (b) Testing is performed after the integration of modules.
   (c) Testing starts as soon as the SRS is prepared.
   (d) None of the above
2. V&V diagram includes
   (a) Verification only
   (b) Validation only
   (c) Both verification and validation
   (d) None of the above
3. Which test plan is made corresponding to requirement gathering?
   (a) Acceptance test plan
   (b) System test plan
   (c) Function & Integration test plan
   (d) Unit test plan
4. Which test plan is made corresponding to requirement specifications?
   (a) Acceptance test plan
   (b) System test plan
   (c) Function & Integration test plan
   (d) Unit test plan
5. Which test plan is made corresponding to HLD?
   (a) Acceptance test plan
   (b) System test plan
   (c) Function & Integration test plan
   (d) Unit test plan
6. Which test plan is made corresponding to LLD?
   (a) Acceptance test plan
   (b) System test plan
   (c) Function & Integration test plan
   (d) Unit test plan
7. In the V model of testing, the scope of verification from top to bottom
   (a) Increases
   (b) Decreases
   (c) Remains the same
   (d) None of the above

8. For the verification of requirements, you must check for
   (a) Correctness
   (b) Ambiguity
   (c) Completeness
   (d) All of the above
9. For the verification of high-level design, you must check
   (a) Data design
   (b) Architectural design
   (c) User interface design
   (d) All of the above
10. For verifying architectural design, which is true?
   (a) Check that every functional requirement in the SRS has been take care of in this design.
   (b) Check whether all exceptions handling conditions have been taken care of.
   (c) Check the inter-dependence and interface between the modules.
   (d) All of the above
11. For verifying data design, you must check
   (a) Sizes of data structure
   (b) Size of module
   (c) Overflow in a data structure
   (d) All of the above
12. What is the relation between static and dynamic testing technique?
   (a) Mutually exclusive
   (b) Complementary
   (c) Independent
   (d) None
13. What is the proper sequence of various testing?
   (a) Function testing, integration testing, system testing, acceptance testing, unit testing
   (b) Unit testing, integration testing, function testing, system testing, acceptance testing
   (c) Unit testing, integration testing, system testing, function testing, acceptance testing
   (d) None

Review Questions

1. What are the activities performed by a tester at the time of development of a project?
2. What is a V-diagram? What are its benefits?
3. What is the need for verification?
4. What is the need for validation?
5. What is the role of test plans in a V&V diagram?

- Decision table consists of: condition stub, action stub, condition entry, and action entry.
- In a decision table, when we enter TRUE or FALSE for all input conditions for a particular combination, then it is called a Rule.
- In a decision table, when the condition entry takes only two values – TRUE or FALSE, then it is called a Limited Entry Decision Table**.** When the condition entry takes several values, then it is called an Extended Entry Decision Table.
- Cause-effect graphing is the technique to represent combinations of input conditions.

EXERCISES

MULTIPLE CHOICE QUESTIONS

1. Black-box testing is a _____.
   (a) Static testing
   (b) Dynamic testing
   (c) None of the above
2. It has been observed that test cases, which are designed with boundary input values, have a _____ chance of finding errors.
   (a) High
   (b) Low
   (c) Medium
   (d) Zero
3. How many test cases are there in BVC if there are 5 variables in a module?
   (a) 23
   (b) 13
   (c) 10
   (d) 21
4. How many test cases are there in robustness testing if there are 5 variables in a module?
   (a) 23
   (b) 31
   (c) 10
   (d) 21
5. How many test cases are there in worst-case testing if there are 4 variables in a module?
   (a) 623
   (b) 513
   (c) 625
   (d) 521

6. Each row of state table corresponds to _____.
   (a) Input
   (b) State
   (c) Transition
   (d) None of the above
7. Each column of state table corresponds to _____.
   (a) Input
   (b) State
   (c) Transition
   (d) None of the above
8. Intersection of a row and a column specifies _____.
   (a) Input
   (b) State
   (c) Transition and output
   (d) None of the above
9. What are the components of a decision table?
   (a) Condition stub
   (b) Condition entry
   (c) Action stub
   (d) All
10. If there are *k* rules over *n* binary conditions, there are at least _____ test cases and at the most _____ test cases.
    (a) k+2, 2^{n+2}
    (b) k+3, 2^{n+3}
    (c) k, 2^n
    (d) None of the above
11. Boundary value analysis and equivalence class partitioning methods do not consider _____.
    (a) Combinations of input conditions
    (b) Inputs
    (c) Outputs
    (d) None

Review Questions

1. What are the types of errors detected by black-box testing?
2. Which type of testing is possible with BVA?
3. Which type of testing is possible with equivalence class partitioning?

- A graph matrix is a square matrix whose rows and columns are equal to the number of nodes in the flow graph.
- A matrix defined with link weights is called a connection matrix.
- The connection matrix is used to find the cyclomatic complexity number of the flow graph.
- Graph matrices are also used to produce a set of all paths between all nodes. An *m*th power of the matrix represents all path segments *m* links long.
- Data flow testing is a white-box testing technique that can be used to detect improper use of data values due to coding errors.
- Data flow anomalies represent the patterns of data usage which may lead to an incorrect execution of the code.
- Definition node in a data flow graph is the node wherein a variable is assigned a value for the very first time in the program.
- Usage node in a data flow graph is the node wherein the variable has already been used in some statement of the program.
- A du-path with respect to a variable *v* is a path between the definition node and the usage node of that variable.
- A dc-path with respect to a variable *v* is a path between the definition node and the usage node such that no other node in the path is a defining node of variable *v*.
- du-paths which are not definition-clear paths are important from testing viewpoint, as these are potential problematic spots for testing persons. du-paths which are definition-clear are easy to test in comparison to those which are not dc-paths.
- The application of data flow testing can be extended to debugging where a testing person finds the problematic areas in the code to trace the bug. So the du-paths which are not definition-clear need more attention of the tester.
- During mutation testing, faults are introduced into a program by creating many versions of the program, each of which contains a fault. Test data are used to execute these faulty programs with the goal of causing each faulty program to fail. Faulty programs are called mutants of the original program and a mutant is said to be killed when a test case causes it to fail.
- When the mutants are single modifications of the initial program using some operators, they are called primary mutants.
- When multiple levels of mutation are applied on the initial program, they are called secondary mutants.

EXERCISES

MULTIPLE CHOICE QUESTIONS

1. White-box testing is _____ to black-box testing.
   (a) mutually exclusive
   (b) complementary
   (c) not related

2. The effectiveness of path testing rapidly _____ as the size of software under test.
   (a) decreases
   (b) increases
   (c) does not change
   (d) none of the above
3. A node with more than one arrow leaving it is called a _____
   (a) decision node
   (b) junction node
   (c) region
   (d) all of the above
4. A node with more than one arrow entering it is called a _____
   (a) decision node
   (b) junction node
   (c) region
   (d) all of the above
5. Areas bounded by edges and nodes are called _____
   (a) decision node
   (b) junction node
   (c) region
   (d) all of the above
6. The length of a path is measured by the number of _____
   (a) instructions
   (b) junction nodes
   (c) decision nodes
   (d) links
7. An independent path is any path through the graph that introduces at least _____ new set of processing statements or new conditions.
   (a) 4
   (b) 3
   (c) 1
   (d) 2
8. The number of independent paths is given by _____.
   (a) $V(G) = e - n + 1$
   (b) $V(G) = 2e - n + 1$
   (c) $V(G) = e - n + 2$
   (d) none of the above
9. According to Mill's Theorem, _____
   (a) $V(G) = d + 2P$

(b) $V(G) = d + P$

(c) $V(G) = 2d + P$

(d) None of the above

10. In data flow anomalies, dd is a _____

(a) serious bug

(b) normal case

(c) harmless bug

(d) none of the above

11. In data flow anomalies, du is a _____

(a) serious bug

(b) normal case

(c) harmless bug

(d) none of the above

12. In data flow anomalies, ku is a _____

(a) serious bug

(b) normal case

(c) harmless bug

(d) none of the above

13. In single-character data anomalies, ~d is _____

(a) potential bug

(b) normal situation

(c) none of the above

15. In single-character data anomalies, ~k is _____

(a) potential bug

(b) normal situation

(c) none of the above

16. _____ is the strongest criterion for selecting test cases.

(a) AD

(b) APU

(c) AU

(d) ADPU

Review Questions

1. What is the need of white-box testing?
2. What are the different criteria for logic coverage?
3. What is basis path testing?

- The rate of inspection is calculated as the error detection efficiency of the inspection process as given below:

$$\text{Error detection efficiency} = \frac{\text{Error found by an inspection}}{\text{Total errors in the item before inspection}} \times 100$$

- Active Design Reviews inspects the design stage of SDLC. In this process, several reviews are conducted targeting a particular type of bugs.
- Formal Technical Asynchronous review method (FTArm) is a type of asynchronous inspection in which the inspectors never have to simultaneously meet. For this process, an online version of the document is made available to every member where they can add their comments and point out the bugs.
- Gilb Inspection is a type of inspection process wherein the defect detection is carried out by individual inspectors at their own level rather than in a group.
- Humphrey's Inspection emphasizes finding and logging the bugs in the preparation phase. It also includes an analysis phase between preparation and meeting. In the analysis phase, individual logs are analysed and combined into a single list.
- N-Fold Inspection process consists of many independent inspection teams to increase the effectiveness of the inspection process and results in detecting more number of bugs quickly. This process needs a coordinator who coordinates various teams, collects and collates the inspection data received by them.
- Phased Inspections are designed to verify the product in a particular domain with the help of experts in that domain. In this process, the inspection is divided into more than one phase. There is an ordered set of phases and each phase is designed such that it will check a particular feature in the product.
- A Reading Technique can be regarded as a mechanism or strategy for the individual inspector to detect defects in the inspected product.
- A Walkthrough is less formal, has fewer steps, and does not use a checklist to guide or a written report to document the team's work.
- A Technical Review is intended to evaluate the software in light of development standards, guidelines, and specifications and to provide the management with evidence that the development process is being carried out according to the stated objectives. A review is similar to an inspection or walkthrough, except that the review team also includes management.

EXERCISES

MULTIPLE CHOICE QUESTIONS

1. In static testing, a bug is found at its _____ location.
   (a) Exact
   (b) Nearby
   (c) None of the above
2. Static testing can be applied for most of the _____ .
   (a) Validation activities

(b) Verification activities
(c) SDLC activities
(d) None of the above

3. Formal peer evaluation of a software element whose objective is to verify that the software element satisfies its specifications and conforms to standards, is called _____ .
   (a) Walkthrough
   (b) Inspections
   (c) Reviews
   (d) None of the above
4. The programmer or designer responsible for producing the program or document is known as _____ .
   (a) Author
   (b) Owner
   (c) Producer
   (d) All
5. The person who finds errors, omissions, and inconsistencies in programs and documents during an inspection is known as _____ .
   (a) Inspector
   (b) Moderator
   (c) Author
   (d) Producer
6. The key person with the responsibility of planning and successful execution of inspection is known as _____ .
   (a) Inspector
   (b) Moderator
   (c) Author
   (d) Producer
7. The inspection team points out any potential errors or problems found and records them in _____.
   (a) SDD
   (b) SRS
   (c) STD
   (d) Log Form
8. 'How much evaluation of an item has been done by the team' is called _____ .
   (a) Rate of errors
   (b) Rate of inspection
   (c) Rate of failures
   (d) None of the above
9. _____ is a more formal process.
   (a) Walkthroughs
   (b) Inspection

(c) Reviews
(d) None of the above

10. The efficiency of code coverage performed by dynamic testing _____ with the increase in size of the system.
(a) Decreases
(b) Increases
(c) Remains same
(d) None of the above

11. Through the inspection process, the modules can be analysed based on _____.
(a) Error-types
(b) Inspection reports
(c) Error-density
(d) None of the above

12. The inspection in which the inspectors never have to simultaneously meet is known as _____.
(a) Phased Inspection
(b) FTArm
(c) Gilb Inspection
(d) All

13. Checking phase has been introduced in _____.
(a) Phased Inspection
(b) FTArm
(c) Gilb Inspection
(d) None of the above

14. Analysis phase between preparation and meeting has been introduced in _____.
(a) Phased Inspection
(b) Humphrey's Inspection
(c) FTArm
(d) N-fold Inspection

15. Collation Phase has been introduced in _____.
(a) Phased Inspection
(b) Humphrey's Inspection
(c) FTArm
(d) N-fold Inspection

16. _____ process gives the chance to utilize human resources.
(a) Phased Inspection
(b) Humphrey's Inspection
(c) FTArm
(d) N-fold Inspection

17. A series of steps or procedures whose purpose is to guide an inspector in acquiring a deep understanding of the inspected software is known as _____.

(a) Checklists
(b) Inspection
(c) Reading Techniques
(d) N-fold Inspection

18. _____ is a reading technique.
(a) Checklists
(b) Inspection
(c) Usage-based method
(d) Task-based method

19. A review is similar to an inspection or walkthrough, except that the review team also includes _____.
(a) Customer
(b) Developer
(c) Tester
(d) Management

20. _____ is not an inspection variant.
(a) Active design review
(b) FTArm
(c) Walkthrough
(d) None of the above

Review Questions

1. What are the advantages of static testing as compared to dynamic testing?
2. What are the benefits of inspection process as compared to dynamic testing?
3. Who can be a member of the inspection team?
4. What are the stages of an inspection process?
5. How does the rate of inspection affect the effectiveness of the inspection process?
6. What is the difference between inspection, walkthrough, and reviews?
7. Make a table indicating the feature of every type of inspection variant.
8. What are the factors that increase the effectiveness of N-Fold inspection?
9. How is scenario-based reading different from checklists? Explain the types of scenario-based reading.
10. Take a small project and apply all the static testing techniques on its SRS, SDD, and code.
11. Develop a list based on usage-based reading method on the project taken in Problem 10.
12. What are the drawbacks of checklists?
13. Is ad-hoc method a random method of reading?

does not meet its original requirements and objectives, as stated in the requirement specification.

- Recovery is the ability of a system to restart operations after the integrity of the application has been lost.
- Security is a protection system that is needed to secure confidential information and to assure the customer that their data will be protected.
- Stress testing tries to break the system under test by overwhelming its resources and finding the circumstances under which it will crash.
- Performance testing is to test the system against all performance requirements mentioned in the SRS. It is often used as a part of performance profile tuning. The goal is to identify the 'weakest links' — the system often carries a number of parts which, with a little tweak, can significantly improve the overall performance of the system.
- Usability testing identifies discrepancies between the user interfaces of a product and the human engineering requirements of its potential users.
- Compatibility testing is to check the compatibility of a system being developed with different operating systems, hardware and software configurations available, etc.
- Configuration testing allows developers/testers to evaluate system performance and availability, when hardware exchanges and reconfigurations occur.
- Acceptance testing is the formal testing conducted to determine whether a software system satisfies its acceptance criteria and to enable buyers to determine whether to accept the system or not. It is of two types: alpha testing and beta testing.
- Alpha testing is performed with customers at the developers' site.
- Beta testing is performed by customers at their own site.

EXERCISES

MULTIPLE CHOICE QUESTIONS

1. Software validation is achieved through a series of _____ tests that demonstrate conformity with requirements.
   (a) white-box
   (b) black-box
   (c) unit tests
   (d) none of the above
2. Before we validate the entire software, _____ must be validated first.
   (a) modules
   (b) system
   (c) functionality
   (d) all of the above
3. Unit tests ensure that the software meets at least a _____ of functionality prior to integration and system testing.
   (a) high-level
   (b) low-level

(c) baseline level
(d) none of the above

4. Two types of interface modules which must be simulated, if required, to test the module are _____.
   (a) unit and integration
   (b) simulators and emulators
   (c) stubs and drivers
   (d) none of the above
5. Overhead of stubs and drivers may increase the _____ of the entire software system.
   (a) test cases
   (b) time
   (c) cost
   (d) time and cost
6. Integration of modules is according to the _____ of software.
   (a) design
   (b) coding
   (c) specifications
   (d) all of the above
7. Recovery is the ability of a system to _____ operations after the integrity of the application has been lost.
   (a) suspend
   (b) observe
   (c) restart
   (d) all of the above
8. A system that meticulously records transactions and system states periodically so that these are preserved in case of a failure is called a _____.
   (a) checkpoint
   (b) transaction system
   (c) recovery system
   (d) none of the above
9. Security requirements should be associated with each _____ requirement.
   (a) functional
   (b) design
   (c) coding
   (d) testing
10. Measures intended to allow the receiver to determine that the information which it receives has not been altered in transit is known as _____.
   (a) confidentiality
   (b) integrity

(c) authentication
(d) none of the above

11. The process of determining that a requester is allowed to receive a service or perform an operation is called _____.
    (a) confidentiality
    (b) integrity
    (c) authentication
    (d) authorization
12. A measure intended to prevent the later denial that an action happened, or a communication took place is called _____.
    (a) confidentiality
    (b) integrity
    (c) non-repudiation
    (d) authorization
13. The type of system testing related to a system's presentation rather than its functionality is called _____.
    (a) usability testing
    (b) stress testing
    (c) conversion testing
    (d) all of the above
14. A system test should not be performed by _____.
    (a) programmers
    (b) testers
    (c) designers
    (d) all of the above
15. Acceptance testing must occur at the _____ of the development process.
    (a) start
    (b) end
    (c) middle
    (d) none of the above
16. Top-down integration testing requires _____ stubs.
    (a) nodes + 2
    (b) nodes – 1
    (c) nodes + 1
    (d) none
17. Bottom-up integration testing requires _____ drivers.
    (a) nodes + 2
    (b) nodes + leaves
    (c) nodes – leaves
    (d) none

18. Total number of sessions in decomposition-based integration testing is _____.
    (a) nodes + 2
    (b) nodes + leaves
    (c) nodes – leaves + edges
    (d) none of the above
19. The total number of sessions in a pair-wise call graph-based integration testing is _____.
    (a) total edges in the graph + 2
    (b) nodes + leaves
    (c) nodes – leaves + edges
    (d) total edges in the graph
20. Total number of sessions in neighbourhood call graph-based integration testing is _____.
    (a) total edges in the graph + 2
    (b) nodes + sink nodes
    (c) nodes – leaves + edges
    (d) nodes – sink nodes
21. The nodes where the control is being transferred after calling the module, are called _____.
    (a) sink nodes
    (b) source nodes
    (c) message
    (d) none of the above
22. The nodes from which the control is transferred are called _____.
    (a) sink nodes
    (b) source nodes
    (c) message
    (d) none of the above
23. When the control from one unit is transferred to another unit, then the programming language mechanism used to do this is known as
    (a) sink nodes
    (b) source nodes
    (c) message
    (d) none of the above
24. A call graph is a _____.
    (a) undirected graph
    (b) cyclic graph
    (c) directed graph
    (d) none of the above