**Đề 9\_ Đáp án**

1. Which of the problems below BEST characterize a result of software failure?
2. **Damaged reputation**
3. Lack of methodology
4. Inadequate training
5. Regulatory compliance
6. What should be taken into account to determine when to stop testing?
7. Technical risk
8. Business risk
9. Project constraints
10. Product documentation
11. I and II are true; III and IV are false
12. III is true, I,II and IV are false
13. I,II and IV are true, III is false
14. **I,II and III are true; IV is false**
15. What is the process of analyzing and removing cause of failures in software?
16. Validation
17. Testing
18. **Debugging**
19. Verification
20. Which general testing principles are characterized by the descriptions below?

W. Early testing

X. Defect clustering

Y. Pesticide paradox

Z. Absence-of-errors fallacy

1. Testing should start at the beginning of the project

2. Conformance to requirements and fitness for use

3. Small number of modules contain the most defects

4. Test cases must be regularly reviewed and revised

1. W1,X2,Y3 and Z4
2. **W1, X3, Y4 and Z2**
3. W2, X3, Y1 and Z4
4. W1, X4, Y2 and Z3
5. Which of the following MAIN activity is part of the fundamental test process?
6. Initiating and planning
7. Documenting root-causes
8. Capturing lesson learned
9. **Planning and control**
10. Which of the following are MAJOR test implementation and execution tasks?
11. Repeating test activities
12. Creating test suites
13. Reporting discrepancies
14. Logging the outcome
15. Analyzing lessons learned
16. II, III and IV
17. I, III, IV and V
18. **I, II, III and IV**
19. III, IV and V
20. What principle is BEST described when test designs are written by a third-party?
21. Exploratory testing
22. **Independent testing**
23. Integration testing
24. Interoperability testing
25. Which test levels are USUALLY included in the common type of V-model?
26. Integration testing, system testing, acceptance testing and regression testing
27. **Component testing, integration testing, system testing, and acceptance testing**
28. Incremental testing, exhaustive testing, exploratory testing and data driven testing
29. Alpha testing, beta testing, black-box testing and white-box testing
30. What test can be conducted for off-the-shelf software to get market feedback?
31. **Beta testing**
32. Usability testing
33. Alpha testing
34. COTS testing
35. Who OFTEN performs system testing and acceptance testing respectively?
36. Senior programmers and professional testers
37. Technical system testers and potential customers
38. **Independent test team and users of the system**
39. Development team and customers of the system
40. What is the key difference between (a) contract and regulation acceptance testing, and (b) alpha and beta testing?
41. (a) are performed outside the company and (b) are conducted by the test team
42. (a) are conducted by regulators and (b) are performed by system administrators
43. (a) are mandatory test for government applications and (b) are usually optional
44. **(a) are for custom-developed software and (b) are for off-the-shelf software**
45. Which test measures the system at or beyond the limits of its specified requirements?
46. Structural testing
47. **Stress testing**
48. Error guessing
49. Black-box testing
50. Which test ensures that modifications did not introduce new problems?
51. Stress testing
52. Black-box testing
53. Structural testing
54. **Regression testing**
55. Which typical defects are easier to find using static instead of dynamic testing?

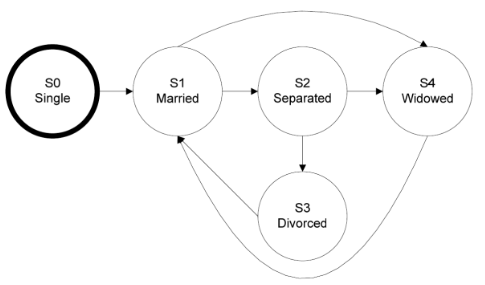
L. Deviation from standards

M. Requirements defects

N. Insufficient maintainability

O. Incorrect interface specifications

1. **L, M, N and O**
2. L and N
3. L, N and O
4. L, M and N
5. In a formal review, who is primarily responsible for the documents to be reviewed?
6. **Author**
7. Manager
8. Moderator
9. Reviewers
10. Who typically use static analysis tools?
11. Customers and users
12. **Developers and designers**
13. Business and systems analysts
14. System and acceptance testers
15. Which aspects of testing will establishing traceability helps?
16. Configuration management and test data generation
17. Test case specification and change control
18. Test condition and test procedure specification
19. **Impact analysis and requirements coverage**
20. Features to be tested, approach, item pass/fail criteria and test deliverables should be specified in which document?
21. Test case specification
22. Test procedure specification
23. **Test plan**
24. Test design specification
25. Which test technique is based on requirements specifications?
26. White-box technique
27. Component testing
28. **Black-box technique**
29. Data driven testing
30. Which test design techniques should a tester use to respectively achieve the following:
31. Check the document features of the system
32. Ensure 100% decision coverage
33. Defect likely defects and distribution?
34. Specification-based, data driven testing, and defect density techniques
35. Specification-based, branch coverage, and exploratory techniques
36. Structure-based, equivalence partitioning, and exploratory techniques
37. **Specification-based, structure-based, and experience-based techniques**
38. What technique captures system requirements that contain logical conditions?
39. Boundary value
40. Equivalence partition
41. **Decision table**
42. State transition
43. Input and output combinations that will be treated the same way by the system can be tested using which technique?
44. Boundary value
45. **Equivalence partition**
46. Decision table
47. State transition
48. Which test suite will check for an invalid transition using the diagram below?



1. S0-S1-S2-S3-S1-S4
2. S0-S1-S4-S1-S2-S3
3. **S0-S1-S3-S1-S2-S1**
4. S0-S1-S2-S3-S1-S2
5. How are integration testing and use case testing similar and dissimilar?
6. **Both checks for interactions: Integration for components, use case for actors**
7. Both are black-box techniques: Integration is low-level, use case is high-level
8. Both static testing: developers perform integration, users execute use case tests
9. Both are V&V techniques: integration is for validation, use case is for verification
10. How many test cases are need to achieve 100% decision coverage?

If (p=q) {

s=s+1

If (s<5) {

t=10;

}

} else if (p>q) {

t=5;

}

1. 3
2. 6
3. 5
4. **4**
5. What analysis determines which parts of the software have been executed?
6. Impact analysis
7. **Code coverage**
8. Gap analysis
9. Cyclomatic complexity
10. Based on the error guessing test design technique, which of the following will an experienced tester MOST LIKELY test in calendar software?
11. First two letters of the month, e.g. MA can represent March or May
12. First letter of the day, e.g., T can mean Tuesday or Thursday
13. Leap year
14. Number of days in a month
15. Three-digit days and months
16. i, ii, iv and v
17. iii and iv
18. **i, ii, iii, and iv**
19. i, ii and v
20. Which input combinations will a knowledgeable tester MOST LIKELY use to uncover potential errors when testing a surname field?
21. Johnson, de la Cruz and Morgan
22. Go, Stephanopoulous and Venkatsewaran
23. Smit, Smyth and Smithsonian
24. **O’Brien, Zeta-Jones and Young Pów**
25. Which of the following demonstrates independence in testing?

J. Independent testers are external to the organization

K. Independent testers are part of the development team

L. Independent testers are from the user community

M. Programmers who wrote the code serve as independent testers

N. Customers who wrote the requirements serve as independent testers

1. J, L and N
2. **J, K, L and N**
3. K, M and N
4. J, L, M and N
5. Which of the following is a KEY task of a tester?
6. **Reviewing tests developed by others**
7. Writing a test strategy for the project
8. Deciding what should be automated
9. Writing test summary reports
10. In software testing, what is the MAIN purpose of exit criteria?
11. To enhance the security of the system
12. To prevent endless loops in codes
13. To serve as and alternative or “Plan B”
14. **To define when to stop testing**
15. Which test approaches or strategies are characterized by the descriptions below?

S. Analytical approaches

T. Model-based approaches

U. Methodical approaches

V. Consultative approaches

1. Relies on guidelines from domain experts

2. Includes error guessing and fault-attacks

3. Uses statistical information about failure rates

4. Focuses on areas of greatest risk

1. **S4, T3, U2, V1**
2. S1, T2, U3, V4
3. S2, T3, U1, V4
4. S3, T4, U2, V1
5. Which of the following can be used to measure progress against the exit criteria?

W. Number of test cases that passed or failed

X. Number of defects found in a unit of code

Y. Dates for milestones and deliverables

Z. Subjective confidence of testers in the product

1. **W, X, Y and Z**
2. W, X and Y
3. W and X
4. W, X and Z
5. What type of risk includes potential failure areas in the software?
6. Project risks
7. **Product risks**
8. Economic risks
9. Requirements risks
10. Based on the IEEE Standard for software test documentation (IEEE Std 829-1998), which sections of the test incident report should the following details be recorded?

Sections:

1. Test incident report identifier
2. Summary
3. Incident description
4. Impact

Details

1. Unique identifier
2. Version level of the test items
3. Inputs
4. Expected results
5. Actual results
6. Anomalies
7. Date and time
8. a:1; b:2 and 7; c:3,4 and 5; d:6
9. a:1; b:6 and 7; c:3,4 and 5; d:7
10. **a:1; b:2; c:3,4,5,6 and 7**
11. a:1; b:6 and 7; c:3,4 and 5
12. Based on the IEEE Standard for software test documentation (IEEE Std 829-1998), which of sections are the part of the test summary report?
13. Test summary and report identifier
14. Summary
15. Variances
16. Anomalies
17. Comprehensive assessment
18. Approvals
19. a,b,e and f
20. a,b,c,d and f
21. **a,b,c,e and f**
22. a,b,c and f
23. What is the name of a skeletal implementation of a software component that is used for testing?
24. Use case
25. Domain
26. Driver
27. **Stub**
28. Which of the following are potential benefits of using test support tools?
29. Ensuring greater consistency and minimizing software project risks
30. **Reducing repetitive work and gaining easy access to test information**
31. Performing objective assessment and reducing the need for training
32. Allowing for greater reliance on the tool to automate the test process
33. Which test support tool can be used to enforce coding standards?
34. **Static analysis tool**
35. Performance testing tool
36. Test comparator
37. Test management tool
38. What should be considered when introducing a tool into an organization?
39. **Assessing the organizational maturity**
40. Counting the number of systems to be tested
41. Calculating the ratio between programmers and testers
42. Reviewing the exit criteria of previous projects