**Software Tester Test**

**Time: 120 minutes**

***Notes*:**

* *Ask any questions you may have now. No questions will be permitted during the test.*
* *Please do not write on question papers*
* *Write all your answers on the answer sheet*
* *Do not use mobile phone*
* *1-45: 1 mark/each question*
* *45: 10 marks*

1. Faults found by users are due to:
   1. Poor quality software
   2. Poor soft ware and poor testing
   3. Bad luck
   4. Insufficient time for testing

Answer: b

1. Which of the following could be a reason for a failure?

1) Testing fault  
2) Software fault  
3) Design fault  
4) Environment Fault  
5) Documentation Fault

* 1. 2 is a valid reason; 1,3,4 & 5 are not
  2. 1,2,3,4 are valid reasons; 5 is not
  3. 1,2,3 are valid reasons; 4 & 5 are not
  4. All of them are valid reasons for failure

Answer: d

1. What is failure?
   1. Deviation from expected result to actual result
   2. Defect in the software.
   3. Error in the program code.
   4. Fault in the system

Answer: a

1. Which of the following could be a root cause of a defect in financial software in which an incorrect interest rate is calculated?
   1. Insufficient funds were available to pay the interest rate calculated.
   2. Insufficient calculations of compound interest were included.
   3. Insufficient training was given to the developers concerning compound interest calculation rules.
   4. Incorrect calculators were used to calculate the expected results

Answer: c

1. What is the main reason for testing software before releasing it ?
   1. To show that system will work after release
   2. To decide when the software is of sufficient he to release
   3. To find as many bugs as possible before release
   4. To give information for a risk based decision about release

Answer: d

1. Which of the following is true?
   1. Component testing should be black box, system testing should be white box.
   2. If you find a lot of bugs in testing, you should not be very confident about the quality of software
   3. The fewer bugs you find, the better your testing was
   4. The more tests you run, the more bugs you will find

Answer: b

1. Select a reason that does not agree with the fact that complete testing is impossible:
   1. The domain of possible inputs is too large to test.
   2. Limited financial resources.
   3. There are too many possible paths through the program to test.
   4. The user interface issues (and thus the design issues) are too complex to completely test

Answer: b

1. In foundation level syllabus you will find the main basic principles of testing. Which of the following sentences describes one of these basic principles?
   1. Complete testing of software is attainable if you have enough resources and test tools
   2. With automated testing you can make statements with more confidence about the quality of a product than with manual testing
   3. For a software system, it is not possible, under normal conditions, to test all input and output combinations.
   4. A goal of testing is to show that the software is defect free

Answer: c

1. In prioritizing what to test, the most important objective is to:
   1. Find as many faults as possible.
   2. Test high risk areas.
   3. Obtain good test coverage.
   4. Test whatever is easiest to test

Answer: b

1. Designing the test environment set-up and identifying any required infrastructure and tools are a part of which phase
   1. Test Implementation and execution
   2. Test Analysis and Design
   3. Evaluating the Exit Criteria and reporting
   4. Test Closure Activities

Answer: d

1. Which of the following is a characteristic of good testing in any life cycle model?
   1. All document reviews involve the development team.
   2. Some, but not all, development activities have corresponding test activities.
   3. Each test level has test objectives specific to that level.
   4. Analysis and design of test s begins as soon as development is complete

Answer: c

1. Which of the following is true of the V-model?
   1. It states that modules are tested against user requirements.
   2. It only models the testing phase.
   3. It specifies the test techniques to be used.
   4. It includes the verification of designs

Answer: d

1. Which is not a Component testing
   1. Check the memory leaks
   2. Check the robustness
   3. Check t he branch coverage
   4. Check the decision tables

Answer: d

1. System Integration testing should be done after
   1. Integration testing
   2. System testing
   3. Unit testing
   4. Component integration testing

Answer: c

1. Integration testing where no incremental testing takes place prior to all the system’s component s being combined to form the system.
   1. System testing
   2. Component Testing
   3. Incremental Testing
   4. Big bang testing

Answer: d

1. During this event the entire system is tested to verify that all functional information structural and quality requirements have been met. A predetermined combination of tests is designed that when executed successfully satisfy management that the system meets specifications
   1. Validation Testing
   2. Integration Testing
   3. User Acceptance Testing
   4. System Testing

Answer: c

1. The main focus of acceptance testing is:
   1. finding faults in the system
   2. ensuring that the system is acceptable to all users
   3. testing the system with other systems
   4. testing for a business perspective

Answer: d

1. What type of testing will you perform on internet banking solution?
   1. System integration
   2. Functional testing
   3. Non-functional testing.
   4. Requirement s testing

Answer: c

1. Black-box testing technique is also called as structure based technique
   1. True
   2. False

Answer: b

1. Regression testing always involves
   1. Testing whet her a known software fault been fixed.
   2. Executing a large number of different tests.
   3. Testing whet her modifications have introduced adverse side effects.
   4. Using a test automation tool

Answer: c

1. Which expression best matches the following characteristics or review processes?

1. Led by author

2. Undocumented

3. No management participation

4. Led by a trained moderator or leader

5. Uses entry exit criteria

s) Inspection

t) Peer review

u) Informal review

v) Walkthrough

* 1. s = 4, t = 3, u = 2 and 5, v = 1
  2. s = 4 and 5, t = 3, u = 2, v = 1
  3. s = 1 and 5, t = 3, u = 2, v = 4
  4. s = 5, t = 4, u = 3, v = 1 and 2
  5. s = 4 and 5, t = 1, u = 2, v = 3

Answer: b

1. \_\_\_\_\_\_\_\_\_ reviews are often held with just the programmer who wrote the code and one or two other programmers or testers.
   1. Formal Reviews
   2. Peer Reviews
   3. Semi Formal Reviews
   4. All of the above

Answer: b

1. Which expression best matches the following characteristics of the review processes:

1. led by the author   
2. undocumented   
3. no management participation   
4. led by a moderator or leader   
5. uses entry and exit criteria

s) inspection   
t) peer review   
u) informal review   
v) walkthrough

* 1. s = 4 and 5, t = 3, u = 2, v = 1
  2. s = 4, t = 3, u = 2 and 5, v = 1
  3. s = 1 and 5, t = 3, u = 2, v = 4
  4. s = 4 and 5, t = 1, u= 2, v = 3

Answer: a

1. Which of the following is a purpose of the review planning phase?
   1. Log defects.
   2. Explain the documents to the participants.
   3. Gather metrics.
   4. Allocate the individual roles.

Answer: d

1. What is the main purpose of Informal review?
   1. Inexpensive way to get some benefit
   2. Find defects
   3. Learning, gaining understanding, effect finding
   4. Discuss, make decisions, solve technical problems

Answer: a

1. A Person who documents all the issues, problems and open points that were identified during a formal review.
   1. Moderator.
   2. Scribe
   3. Author
   4. Manager

Answer: b

1. Which rule should not be followed for reviews?
   1. Defects and issues are identified and corrected
   2. The product is reviewed not the producer
   3. All members of the reviewing team are responsible for the result of the review
   4. Each review has a clear predefined objective

Answer: c

1. Typical defects discovered by static analysis includes
   1. Programming standard violations
   2. Referring a variable wit h an undefined value
   3. Security vulnerabilities
   4. All Above

Answer: c

1. Static code analysis typically identifies all but one of the following problems. Which is it?
   1. Unreachable code
   2. Faults in requirements
   3. Undeclared variables
   4. Too few comments

Answer: c

1. Exclusive use of white box testing in a test -phase will:
   1. Ensure the test item is adequately tested.
   2. Make the need for black-box testing redundant.
   3. Run the risk that the requirements are not satisfied.
   4. Suffices for the unit testing phase.

Answer: c

1. Boundary value analysis can only be used during white-box testing.
   1. True
   2. False

Answer: b

1. A thermometer measures temperature in whole degrees only. If the temperature falls below 18 degrees, the heating is switched off. It is switched on again when the temperature reaches 21 degrees. What are the best values in degrees to cover all equivalence partitions?
   1. 15, 19 and 25.
   2. 17, 18 and19.
   3. 18, 20 and22.
   4. 16, 26 and 32.

Answer: a

1. Equivalence partitioning is:
   1. A black box testing technique used only by developers
   2. A black box testing technique than can only be used during system testing
   3. A black box testing technique appropriate to all levels of testing
   4. A white box testing technique appropriate for component testing

Answer: c

1. Which of the following would be an example of decision-table testing for a financial application applied at the system-test level?
   1. A table containing rules for combinations of inputs to two fields on a screen.
   2. A table containing rules for interfaces between components.
   3. A table containing rules for mortgage applications.
   4. A table containing rules for chess.

Answer: c

1. One technique of Black Box testing is Equivalence Partitioning.

In a program statement that accepts only one choice from among 10 possible choices,  
numbered 1 through 10, the middle partition would be from \_\_\_\_\_ to \_\_\_\_\_

* 1. 4 to 6
  2. 0 to 10
  3. 1 to 10
  4. None of the above

Answer: c

1. Which of the following is a black box design technique?
   1. Statement testing
   2. Equivalence partitioning
   3. Error- guessing
   4. Usability testing

Answer: b

1. Which of the following is an important principle for use of experienced-based test techniques?
   1. Tester skill is a critical factor in assignment of test execution tasks.
   2. Tester skills are less important than ensuring 100% tester utili-zation.
   3. Testers should always focus on defect-preventing activities.
   4. Testers should be evaluated based on the number of defects they find.

Answer: a

1. One of the following is not a part of white box testing as per BS7925-II standards.
   1. Random testing
   2. Data Flow testing
   3. Statement testing.
   4. Syntax testing

Answer: d

1. Following example is a

if (condition1 && (condition2 || function1()))  
statement1;  
else  
statement2; (Testing concepts)

* 1. Decision coverage
  2. Condition coverage
  3. Statement coverage
  4. Path Coverage

Answer: a

1. This part of a program is given:

WHILE (condition A) Do B

END WHILE

How many decisions should be tested in this code in order to achieve 100% decision

coverage?

* 1. 2
  2. Indefinite
  3. 1
  4. 4

Answer: a

1. Path coverage includes
   1. Statement coverage
   2. Condition coverage
   3. Decision coverage
   4. None of these

Answer: d

1. Which testing technique do you prefer for the following situations?

1. Severe time pressure

2. Inadequate specification

* 1. Decision testing
  2. Error guessing
  3. Statement testing
  4. Exploratory testing

Answer: d

1. Recovery testing is a system test that forces the software to fail and verifies that data recovery is properly performed. The following should be checked for correctness

1. Re-initialization

2. Restart

3. Data Recovery

4. Check Point Mechanism

* 1. 1 and 2
  2. 1, 2 and 3
  3. 1, 2, 3 and 4
  4. 2 and 4

Answer: c

1. Data flow analysis studies:
   1. Possible communications bottlenecks in a program.
   2. The rate of change of data values as a program executes.
   3. The use of data on paths through the code.
   4. The intrinsic complexity of the code.

Answer: c

1. Which of the following is NOT a white box technique?
   1. Statement testing
   2. Path testing
   3. Data flow testing
   4. State transition testing

Answer: d