NPTEL Online Certification Courses

Indian Institute of Technology Kharagpur

Software Testing

Assignment- 4

TYPE OF QUESTION: MCQ/MSQ

Number of questions: **10** **Total mark: 10 X 1 = 10**

For each of the following questions one or more of the given options are correct. Choose the correct option(s).

QUESTION 1:

Which of the following statements concerning mutation testing is correct?

a. Mutation testing is used to test if a program has bugs
b. Mutation testing is used to test if it is possible to mutate the program
c. Mutation testing is used to test if the test suite is adequate
d. Mutation testing is used to count the number of mutants of the program
e. Mutation testing is used to test whether the performance of the software is satisfactory

Correct Answer: c. Mutation testing is used to test if the test suite is adequate

Detailed Solution:

Main idea of mutation testing is Insert faults into a program: Check whether the test suite is able to detect these. If a mutant remains alive: Even after all test cases have been exhausted, the test suite is enhanced to kill the mutant.

QUESTION 2:

Which of the following mutants are considered as equivalent mutants?

a. Mutants which arise from identical changes made to the program
b. Mutants which arise from similar changes made to the program
c. Mutants which fail with the same set test cases
d. Mutants pass the same set of test cases
e. Mutants which are detected by all the test cases
f. Mutants which cannot be killed by any test case

Correct Answer: f. Mutants which cannot be killed by any test case

Detailed Solution:

There may be surviving mutants that cannot be killed, these are called Equivalent Mutants.

QUESTION 3:

Which of the following are **not** true of mutation testing technique?

a. Hard to automate mutant generation
b. Mutant generation is computationally expensive
c. Presence of equivalent mutants make it difficult to automate the entire mutation testing process
d. A very large number of mutants can be generated.
e. Mutation testing is very effective for eliminating simple programming errors

Correct Answer: a. Hard to automate mutant generation
b. Mutant generation is computationally expensive

Detailed Solution:

Disadvantages of Mutation Testing

- Equivalent mutants
- Computationally very expensive.
    - A large number of possible mutants can be generated.
- Certain types of faults are very difficult to inject.

QUESTION 4:

In the context of mutation testing, suppose in your program you replace the instruction **y=2*x** by **y=x+x**, to create a mutant. Which one of the following kinds of mutant would you create?

a. Trivial mutant
b. Transparent mutant
c. Stillborn mutant
d. Equivalent mutant
e. Erroneous mutant

Correct Answer: d. Equivalent mutant

Detailed Solution:

Equivalent mutants have the same meaning as the original source code even though they have different syntax.

QUESTION 5:

Which of the following statements about alpha testing are **NOT** true?

a. Alpha testing is often performed by an independent test team.
b. Alpha test cases usually comprise performance tests only.
c. Alpha test cases consist primarily of white-box test cases
d. Faults found during alpha testing are much more expensive to fix than those found during unit and integration testing.
e. End-users are often asked to perform alpha testing.

Correct Answer: b. Alpha test cases usually comprise performance tests only.
e. End-users are often asked to perform alpha testing.

Detailed Solution:

Alpha testing is a functional carried out by the test team within the developing organization. So, options b and e are not true about alpha testing.

QUESTION 6:

Which one of the following can be said about regression testing?

a. Ensures that bug fixes are proper and justified
b. Ensures that bugs are not induced in unchanged areas due to bug fixes
c. Ensures that changed areas of the software have no bugs
d. Ensures that unchanged areas of the software have no bugs
e. Regression test cases remain valid throughout the maintenance phase

Correct Answer: b. Ensures that bugs are not induced in unchanged areas due to bug fixes

Detailed Solution:

Regression testing needed after every change. It Ensures unchanged features continue to work fine.

QUESTION 7:

Which one of the following is a suitable unit for testing object-oriented programs?

a. Statement
b. Method
c. Class
d. Package
e. Class hierarchy

Correct Answer: c. Class

Detailed Solution:
Testing object-oriented programs done in class level.
Class level:
–Testing interactions between attributes and methods must be addressed.

QUESTION 8:

In an object-oriented program, even if a base class has been thoroughly tested, why is it necessary to retest the inherited methods?

a. An inherited attribute may be modified in an unanticipated way in the derived class
b. Code of methods can get corrupted during the process of inheritance
c. Encapsulation prevents testing base class code
d. Inherited methods can be polymorphic
e. Inherited class may redefine attributes and methods of the base class

Correct Answer: a. An inherited attribute may be modified in an unanticipated way in the derived class

Detailed Solution:

Should Inherited Methods be Retested?

• Retesting required:
–Because a new context of usage results when a subclass is derived. (Anticomposition axiom)

QUESTION 9:

Which of the following are implied by the polymorphism feature while testing object-oriented programs?

a. Polymorphism makes testing problematic as difficult to find all bindings that may occur.
b. Polymorphism complicates integration testing as many server classes may need to be integrated before a client class can be tested.
c. Simplifies testing as testing any method binding should take care of all possible method bindings
d. Polymorphism provides flexibility and therefore simplifies testing
e. Polymorphism simplifies integration testing as it can be carried out by any of the polymorphic server classes

Correct Answer: a. Polymorphism makes testing problematic as difficult to find all bindings that may occur.

b. Polymorphism complicates integration testing as many server classes may need to be integrated before a client class can be tested.

Detailed Solution:
For the details understanding of this question, please refer slide no: 75-76 of lecture material of week-4.

QUESTION 10:

Why are top-down and bottom-up testing are not appropriate integration strategies for object-oriented programs?

a. Object-oriented programs are large
b. Object interactions are usually complex
c. Encapsulation complicates integration testing
d. Inheritance complicates integration testing
e. Polymorphism hinders top-down and bottom-up testing

Correct Answer: b. Object interactions are usually complex

Detailed Solution:
OO programs do not have a hierarchical control structure as object interactions are usually complex:
–So conventional top-down and bottom-up integration tests have little meaning

---

**************END*************