NPTEL Online Certification Courses

Indian Institute of Technology Kharagpur

Software Engineering

Assignment-12

TYPE OF QUESTION: MCQ/MSQ

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

QUESTION 1:

Consider the following assertions regarding subsumption relation among various code coverage criteria. The subsumption relation between 2 code coverage criteria has been represented by the symbol →. That is, A→B means criterion A subsumes criterion B.

i) MC/DC→ Basic Condition Coverage

ii) Basic Condition Coverage → Decision Coverage

iii) Decision Coverage → Condition/Decision Coverage

iv) MC/DC → Condition/Decision Coverage

Which of the given subsumption relations are valid?

a. i), iv)

b. i), and iii)

c. iii), iv)

d. ii), iv)

e. i), ii) and iv)

Correct Answer: a. i), iv)

Detailed Solution:

For more information, you can refer to slide 4 of week 12 lecture material for the subsume relationship.

QUESTION 2:

Among the following test techniques, which one of the following is the strongest?

a. Statement coverage testing

b. Decision testing

c. Condition/Decision coverage

d. Basic condition testing

e. MC/DC testing

Correct Answer: e. MC/DC testing

Detailed Solution:
MC/DC subsumes other testing techniques. Therefore, it is the strongest among the given techniques

---

QUESTION 3:

Consider the following program written in C language?

```
int main (){
   inta,b=0;
   scanf("%d",&a);
   if( a == 20 ){
            b=b+20;}
   if( a == 30 ){
            b=b+30;}
   else{
            b=b+40; }
}
```

At least how many test cases are required for path testing?

a. 2

b. 3

c. 4

d. 5

e. 6

Correct Answer: b. 3

Detailed Solution:
Since, there are 3 possible paths in the CFG of the given code, it requires at least 3 test cases to achieve path testing.

QUESTION 4:

Which of the following statements about Cyclomatic complexity metric of a program are **FALSE**?

a. It is a measure of the computational complexity of the program

b. It is a measure of the testing difficulty of the program.

c. It is a measure of understanding difficulty of the program.

d. It is a measure of the linearly independent paths in the program

e. It is a measure of the size of the program

Correct Answer: a. It is a measure of the computational complexity of the program

e. It is a measure of the size of the program

Detailed Solution:
CC measures neither the computational complexity nor the size of the program.

QUESTION 5:

What would be the Cyclomatic complexity of the following program?

```
if(i>j) then
        if(i>k) then max=i;
        else max=k;
else    if(j>k) max=j
        else max=k;
```

a. 2

b. 3

c. 4

d. 5

e. 6

Correct Answer: c. 4

Detailed Solution:

There are 3 predicates in the given code. Therefore, the CC of the given code is 4.

QUESTION 6:

Consider the following conditional statement. At least how many test cases are necessary for modified condition/decision coverage (MC/DC)?

```
if((a>=0) or (a<=100)) a=a-1;
```

a. 1

b. 2

c. 3

d. 4

e. MC/DC is not achievable

Correct Answer: e. MC/DC is not achievable

Detailed Solution:

Due to short-circuit evaluation, if a > 5 is true, then a < 100 might not even be evaluated — so we cannot achieve all combinations.

QUESTION 7:

Why is the *all path testing* strategy rarely used in practice?

a. Too many test cases are required to be generated and executed

b. Too weak a testing technique

c. It is subsumed by basis path testing

d. It is subsumed by statement coverage

e. Weaker than even decision coverage

Correct Answer: a. Too many test cases are required to be generated and executed

Detailed Solution:

All-path testing aims to test every possible path through a program's control flow graph (CFG). This is an extremely rigorous strategy requiring too many test cases, but in practice, it is rarely used.

QUESTION 8:

Mutation testing is most effective for which one of the following types of bugs?

a. Algorithmic errors

b. Programming errors

c. Design errors

d. Requirements errors

e. Performance bugs

Correct Answer: b. Programming errors

Detailed Solution:
Mutation testing is most effective for catching programming errors such as small syntactic or logical mistakes in the code.

QUESTION 9:

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

a. Complex mutants get created by several simple mutants.

b. Mutation testing is used to evaluate if a program is bug free

c. It is implicitly assumed that programmers make only simple programming errors

d. Mutation testing is used to test the quality of test cases

e. Mutation testing is stronger than statement coverage-based testing

Correct Answer: b. Mutation testing is used to evaluate if a program is bug free

e. Mutation testing is stronger than statement coverage-based testing

Detailed Solution:

Mutation testing does not prove the absence of bugs in software — it only evaluates the strength of the existing test suite.

QUESTION 10:

Which of the following statements concerning data flow testing are **FALSE**?

a. Test cases are designed based on the definitions and uses of different variables in a program.

b. Test cases are designed based on the dataflow diagram (DFD) representation of the program

c. A simple dataflow testing strategy requires every DU chain in a program be covered at least once

d. Data flow testing helps to effectively test loops

e. Data flow testing is a black box testing technique

Correct Answer: b. Test cases are designed based on the dataflow diagram (DFD) representation of the program

e. Data flow testing is a black box testing technique

Detailed Solution:

DFD is a high-level design tool used in structured analysis, not related to data flow testing. Data flow testing works at the code level, not DFD level. Also, Data flow testing is a white box (structural) testing technique — it requires knowledge of the internal structure and control flow of the code.
