



## Software Testing

### Assignment- 2

#### 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 are not black box testing techniques?

- a. Boundary value testing
- b. Cause-effect graph testing
- c. Path testing
- d. Combinatorial testing
- e. Basic Condition testing

**Correct Answer: c. Path testing**

**e. Basic condition testing**

#### **Detailed Solution:**

Path testing and Basic condition testing are white-box testing techniques.

#### QUESTION 2:

Assume you have developed a simple web-based calculator. The user interface has 3 input fields: *Operand1*, *Operator*, and *Operand2*. The user can give 9 valid inputs: {1, 2, ..., 9} for *Operand1* and also for *Operand2*. The user can enter four valid inputs for *Operator*: {+, -, \*, /}. How many test cases would exhaustive functional testing require?

- a. 3
- b. 12
- c. 22
- d. 256
- e. 324

**Correct Answer: e. 324**

#### **Detailed Solution:**

$$9 (\text{Operand1}) \times 4 (\text{Operator}) \times 9 (\text{Operand 2}) = 324$$



---

**QUESTION 3:**

Which of the following are **false** about white-box testing?

- a. Test cases are designed using only functional specification of the software:
- b. Test cases are designed without any knowledge of the internal structure of the software.
- c. White-box testing is also known as structural testing.
- d. Test case design requires a knowledge of the internal structure of the unit under test
- e. White-box tests can detect trojans in the code
- f. White-box testing can be used to detect if a software meets all its functional requirements

**Correct Answer:** **a.** Test cases are designed using only functional specification of the software  
**b.** Test cases are designed without any knowledge of the internal structure of the software  
**f.** White-box testing can be used to detect if a software meets all its functional requirements

**Detailed Solution:**

In White box testing design test cases so that certain program elements are executed (or covered), design test cases that focus on discovering certain types of faults. So, option **a.**, option **b.** and option **f.** are false about white box testing.

---



**QUESTION 4:**

Assume that a function **test** takes three parameters  $x$ ,  $y$ , and  $z$  as arguments. Each of these three parameters can assume values in the range  $-100$  to  $+100$ . How many boundary value test cases with valid values should be designed?

- a. 7
- b. 9
- c. 11
- d. 13
- e. 17

**Correct Answer: d. 13**

**Detailed Solution:**

In general, need  $(4z + 1)$  test cases for  $z$  independent inputs. Here  $z=3$ , so,  $4*3+1=13$  test cases required. Please refer week-2 lecture material (slide no. 39-40).

**QUESTION 5:**

Consider the function **find-intersection(float m1, float c1, float m2, float c2)** that computes the point of intersection of two straight lines of the form  **$y=mx+c$** . For equivalence class testing, at the first level of the equivalent class hierarchy the valid and invalid equivalence classes can be formed. The valid set of input values can be further divided into how many equivalence classes?

- a. 1
- b. 2
- c. 3
- d. 4
- e. 6

**Correct Answer: c. 3**

**Detailed Solution:**

The equivalence classes are the following:

- Parallel lines ( $m_1=m_2$ ,  $c_1 \neq c_2$ )
- Intersecting lines ( $m_1 \neq m_2$ )
- Coincident lines ( $m_1=m_2$ ,  $c_1=c_2$ )



---

**QUESTION 6:**

Which of the following are not combinatorial testing techniques?

- a. Decision table-based testing
- b. Condition-coverage testing
- c. Cause-effect graphing
- d. Pair-wise testing
- e. Path testing

**Correct Answer:** b. Condition-coverage testing  
e. Path testing

**Detailed Solution:**

Several types of combinatorial testing strategies:

Decision table-based testing, Cause-effect graphing, Pair-wise testing. Conditional and path testing part of white box testing.

---

**QUESTION 7:**

Which one of the following is an implicit assumption made in equivalence class testing?

- a. A program behaves in similar ways to every input value belonging to an equivalence class.
- b. Different equivalence classes of a program contain similar bugs
- c. Different equivalence classes of a program behave similarly to an input
- d. Equivalence classes define the behaviorally similar components of a program
- e. Equivalence classes help identify program functions that can be considered to be behaviorally similar

**Correct Answer:** a. A program behaves in similar ways to every input value belonging to an equivalence class.

**Detailed Solution:**

An implicit assumption made in equivalence class testing is that: A program behaves in similar ways to every input value belonging to an equivalence class.

---



**QUESTION 8:**

Consider a function that takes an integer  $a$  (which can assume integral values between 0 to 100) as argument and carries out actions involving setting the value of a variable  $b$  as specified in the following table.

Condition	Action
$(a < 10) \vee (a > 80)$	$b = b + 10$
$(a == 30)$	$b = b + 20;$
$(a == 40)$	$b = b + 30;$
For all other cases	$b = b + 40;$

During black box testing of the function, how many valid equivalence class test cases with valid input values are required?

- a. 4
- b. 5
- c. 8
- d. 9
- e. 10

**Correct Answer: b. 5**

**Detailed Solution:**

valid equivalence test cases:  $a < 10$  any value,  $a > 80$  any value,  $a = 30$ ,  $a = 40$  and for other cases a value of  $a$  variable. So, five valid test cases.

**QUESTION 9:**

Which one of the following techniques is **not** used to identify the equivalence classes of a unit?

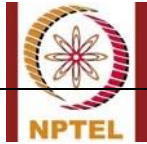
- a. Identify scenarios
- b. Examine the input data.
- c. Analyze program structure
- d. Examine output
- e. Develop cause-effect model of the program

**Correct Answer: c. Analyze program structure**

**Detailed Solution:**

The techniques used to identify the equivalence classes:

Identify scenarios, Examine the input data, Examine output. Please refer week-2 lecture material (slide no.4) for the same



**QUESTION 10:**

Suppose a function named **try** takes two Boolean variables as input. At least how many valid equivalence classes needs to be designed for the black-box testing of the function **try**?

- a. 2
- b. 4
- c. 6
- d. 8
- e. 10

**Correct Answer: b. 4**

**Detailed Solution:**

Two Boolean variables so each of them has two values. So, valid equivalence classes =  $2 \times 2 = 4$ .

\*\*\*\*\*END\*\*\*\*\*