TED (15) -6136
(Revision- 2015)

A21-04448

Reg.No…….....
Signature. .....

DIPLOMA EXAMINATION IN ENGINEERING/TECHNOLOGY/MANAGEMENT/
COMMERCIAL PRACTICE – APRIL -2021.

SOFTWARE TESTING

(Maximum Marks: 75) [Time: 2.15 hours]

PART–A

Marks

I. Answer **any three** questions in one or two sentences. Each question carries 2 marks.

1. List the long term goals of software testing.
2. Differentiate static testing and dynamic testing.
3. List any 4 Black-Box testing techniques.
4. List any two open source testing tools.
5. List any two debugging tools. (3x2=6)

PART - B

II Answer **any four** of the following questions. Each question carries 6 marks.

1. Explain the model for software testing with the help of a diagram.
2. Explain test strategy matrix.
3. Explain the needs of White-Box testing.
4. Define cyclomatic complexity and explain the methods of finding the cyclomatic complexity.
5. Explain the advantages of test automation.
6. Explain debugging process with the help of a diagram.
7. Explain how to correct bugs.

[4x6 =24]

PART - C

(Answer **any of the three units** from the following. Each full question carries 15 marks)

UNIT I

III (a) Explain software testing life cycle with the help of a diagram. (9)

(b) Explain software testing process. (6)

OR

IV (a) Explain testing methodology with the help of a diagram. (9)

(b) Explain V testing model. (6)

UNIT- II

V (a) Explain mutation testing with the help of examples. (9)

(b) Explain stubs and drivers in validation testing. (6)

OR

VI (a) Explain integration testing and its methods. (9)

(b) Differentiate progressive and regression testing. (6)

UNIT- III

VII (a) Explain the guidelines to be followed while selecting a testing tool. (9)

(b) Explain the issues on object oriented testing. (6)

OR

VIII (a) Explain the categories of testing tools. (9)

(b) Explain the challenges in web based testing. (6)

UNIT – IV

IX (a) Explain the three debugging techniques. (9)

(b) Explain kernel debugger and give an example. (6)

OR

X (a) Explain the different types of debuggers. (9)

(b) Explain bug tracking. (6)

****