4/4 B.Tech. SECOND SEMESTER
SOFTWARE TESTING METHODOLOGIES

CS8T2B   **Credits: 4**

Elective – III

Lecture: 4 periods/week   **Internal assessment:30 M**
Tutorial: 1 period /week   **Semester end examination: 70 marks**

Course Context and Overview: This course introduces the fundamental concepts of Software Testing Methodologies With this foundation students can test various types of Applications built and programs in SDLC.

Prerequisites: C LANGUAGE, I/O ANALOG AND DIGITAL INTERFACING, AND PERIPHERALS

Learning Outcomes:

1) Understand the fundamentals of software testing
2) Apply the flow graphs and path testing concepts to identify the path and data flows.
3) Analyze the appropriate domain type while applying to domains and interfaces.
4) Compute path products, regular expressions, decision tables, path expressions and KV Charts for simple software code.

Use state graph techniques for state testing and graph metrics concepts to reduce the nodes.
UNIT - I
Introduction:
Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs.

UNIT - II
Flow graphs and Path testing:
Basics concepts, predicates, path predicates, achievable paths, path sensitizing, path instrumentation, application of path testing.
UNIT - III
Transaction Flow Testing:
Transaction flows, transaction flow testing techniques. Dataflow testing basics, strategies in dataflow testing, application of dataflow testing.

UNIT - IV
Domain Testing:-domains and paths, Nice & ugly domains, domain testing, domains andinterfaces testing, domain and interface testing, domains and testability.

UNIT - V
Paths, Path products and Regular expressions:
Path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

UNIT - VI
Logic Based Testing:
Overview, decision tables, path expressions, kv charts, specifications.

UNIT - VII
State, State Graphs and Transition testing: State graphs, good & bad state graphs, state testing, Testability tips.

UNIT - VIII
Graph Matrices and Application:
Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools

Learning Resources

Text Books:

1. Software Testing Techniques - Boris Beizer, Dreamtech, second edition.

Reference Books:

1. Software Testing Techniques – SPD(Oreille)
2. Software Testing in the Real World – Edward Kit, Pearson.
3. Effective methods of Software Testing, Perry, John Wiley.