

- 2.4.i Structural Graph Coverage for Design Elements
- 2.4.ii Data Flow Graph Coverage for Design Elements
- 2.5 Graph Coverage for Specifications
 - 2.5.i Testing Sequencing Constraints
 - 2.5.ii Testing State Behavior of Software
- 2.6 Graph Coverage for Use Cases
 - 2.6.i Use Case Scenarios
- 2.7 Representing Graphs Algebraically
 - 2.7.i Reducing Graphs to Path Expressions
 - 2.7.ii Applications of Path Expressions
 - 2.7.iii Deriving Test Inputs
 - 2.7.iv Counting Paths in a Flow Graph & Max Path Length
 - 2.7.v Minimum Number of Paths to Reach All Edges
 - 2.7.vi Complementary Operations Analysis
- 2.8 Bibliographic Notes
- 3. **Logic Testing**
 - 3.1 Overview-Logic Predicates and Clauses
 - 3.2 Logic Expression Coverage Criteria
 - 3.2.i Active Clause Coverage
 - 3.2.ii Inactive Clause Coverage
 - 3.2.iii Infeasibility and Subsumption
 - 3.2.iv Making a Clause Determine a Predicate
 - 3.2.v Finding Satisfying Values
 - 3.3 Structural Logic Coverage of Programs
 - 3.3.i Predicate Transformation Issues
 - 3.4 Specification-Based Logic Coverage
 - 3.5 Logic Coverage of Finite State Machines
 - 3.6 Disjunctive Normal Form Criteria
 - 3.7 Bibliographic Notes
- 4. **Input Space Partitioning**
 - 4.1 Input Domain Modeling
 - 4.1.i Interface-Based Input Domain Modeling
 - 4.1.ii Functionality-Based Input Domain Modeling