Sri Indu College of Engineering & Technology
UGC Autonomous Institution
Recognized under 2(f) & 12(B) of UGC Act 1956,
NAAC, Approved by AICTE &
Permanently Affiliated to JNTUH
Estd.2001

NAAC
NATIONAL ASSESSMENT AND ACCREDITATION COUNCIL

HANDOUT

III Year-II Semester

AUTOMATA THEORY AND COMPILER DESIGN

DEPARTMENT OF AI&DS

ACADEMIC YEAR 2025-26

SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY

(An Autonomous Institution under UGC, New Delhi)

(Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi and Accredited by NBA, NAAC)
Sheriguda Village, Ibrahimpatnam Mandal, Ranga Reddy Dist. – 501 510

DEPARTMENT OF AI&DS

HANDOUT- INDEX

S. No	Contents
1	Vision, Mission, PEOs, POs, PSOs & Cos
2	Institution Academic Calendar
3	Department Academic Calendar
4	Subject wise
i)	Syllabus Copy
ii)	Lesson Plan
iii)	Question Bank
iv)	End Examination Questions (Model Paper)
v)	Mid-1 & Mid-2 Questions (Model Paper)

SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY

INSTITUTION VISION

To be a premier Institution in Engineering & Technology and Management with competency, valuesand social consciousness.

INSTITUTION MISSION

IM1 Provide high quality academic programs, training activities and research facilities.

IM2 Promote Continuous Industry-Institute Interaction for Employability, Entrepreneurship,Leadership and Research aptitude among stakeholders.

IM3 Contribute to the Economical and technological development of the region, state and nation.

DEPARTMENT VISION

To be a recognized knowledge center in the field of AI&DS with self - motivated, employable engineers to society.

DEPARTMENT MISSION

The Department has following Missions:

DM1 To offer high quality student centric education in AI&DS

DM2 To provide a conducive environment towards innovation and skills.

DM3 To involve in activities that provide social and professional solutions.

DM4 To impart training on emerging technologies namely cloud computing and IOT with involvementof stake holders.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: **Higher Studies:** Graduates with an ability to apply knowledge of Basic sciences and programming skills in their career and higher education.

PEO2: **Lifelong Learning:** Graduates with an ability to adopt new technologies for ever changing IT industry needs through Self-Study, Critical thinking and Problem solving skills.

PEO3: **Professional skills:** Graduates will be ready to work in projects related to complex problems involving multi-disciplinary projects with effective analytical skills.

PEO4: **Engineering Citizenship:** Graduates with an ability to communicate well and exhibitsocial, technical and ethical responsibility in process or product.

PROGRAM OUTCOMES (POs) & PROGRAM SPECIFIC OUTCOMES (PSOs)

PO	Description
PO 1	Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.
PO 2	Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)
PO 3	Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)
PO 4	Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).
PO 5	Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)
PO 6	The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).
PO 7	Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)
PO 8	Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
PO 9	Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences
PO 10	Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and

	leader in a team, and to manage projects and in multidisciplinary environments.
PO 11	**Life-Long Learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)
Program Specific Outcomes	
PSO 1	**Software Development:** To apply the knowledge of Software Engineering, Data Communication, Web Technology and Operating Systems for building IOT and Cloud Computing applications.
PSO 2	**Industrial Skills Ability:** Design, develop and test software systems for world-wide network of computers toprovide solutions to real world problems.
PSO 3	**Project implementation:** Analyze and recommend the appropriate IT Infrastructure required for the implementationof a project.

COURSE OUTCOMES (CO's)

Academic Year: 2025-26

YEAR/SEM **:** III -II

Course Name : AUTOMATA THEORY AND COMPILER DESIGN COURSE

Sub Code **:** R22CSM2212

At the end of the course student will be able to:

Course Outcomes	Statements
C321.1	Able to employ finite state machines for modeling and solving computing problems
C321.2	Able to design context free grammars for formal languages.
C321.3	Able to distinguish between decidability and un decidability.
C321.4	Demonstrate the knowledge of patterns, tokens & regular expressions for lexical analysis
C321.5	Acquire skills in using lex tool and design LR parsers
C321.6	Understand intermediate code generation and run-time environment

COURSE ARTICULATION MATRIX

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS01	PSO2	PSO 3
C321.1	2	-	1	3	3	-	-	-	-	-	-	-	-	1	-
C321.2	1	3	3	2	1	-	-	-	-	-	-	-	-	-	-
C321.3	1	3	3	1	1	-	-	-	-	-	-	-	-	-	-
C321.4	1	1	1	3	3	-	-	-	-	-	-	-	-	-	-

C321.5	3	-	2	1	3	-	-	-	-	-	-	-	-	1	-
C321.6	3	1	2	1	3	-	-	-	-	-	-	-	1	1	1
C321	1.83	2	2	1.83	2.33	-	-	-	-	-	-	-	1	1	1

\

SOFTWARE TESTING METHODOLOGIES

SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY

(An Autonomous Institution under UGC, New Delhi)

B.Tech. - III Year – II Semester

L	T	P	C
3	0	0	3

Professional Elective –III
(R18CSE3231) Software Testing Methodologies

Objectives:
To understand the software testing methodologies such as flow graphs and path testing, transaction flows testing, data flow testing domain testing and logic based testing.

UNIT I :
Introduction : Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs
Flow graphs and Path testing : Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT II :
Transaction Flow Testing : Transaction flows, transaction flow testing techniques.
Dataflow testing: Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

UNIT III:
Domain Testing: Domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT IV :
Paths, Path products and Regular expressions : Path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.
Logic Based Testing : Overview, decision tables, path expressions, kv charts, specifications.

UNIT V :
State, State Graphs and Transition testing : State graphs, good & bad state graphs, state testing, Testability tips.
Graph Matrices and Application : Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools (student should be given an exposure to a tool like JMeter or Win-runner).

TEXT BOOKS :
1. Software Testing techniques - Boris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

REFERENCES :
1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing, 3rd Edition, P.C. Jorgensen, Aurbach Publication (Dist by SPO)
3. Software Testing, n. Chauhan, Oxford University Press.
4. Introduction to Software Testing, P. Ammann & J. Offutt, Cambridge Univ. Press.
5. Effective methods of Software Testing, Perry, John Wiley 3rd Edition, 1999.
6. Software Testing Concepts and Tools, P. Nageswara Rao, dreamtech Press
7. Software Testing, M.G. Limye, TMH.
8. Software Testing, S. Desikan, G. Reamesh, Pearson
9. Foundations of Software Testing, D. Graham & Others, Cengage Learning.
10. Foundations of Software Testing, A.P. Madhur, Pearson.

	SRI INDU COLLEGE OF ENGG &TECH **(Regulation :R22)** **Department of IT,AI&DS**	
	Sub. Code & Title	**R22CSM2212 Software Testing Methodologies**
	Academic Year: 2025-26	**Year/Sem./Section** **III-II**
	Faculty Name & Designation	**A.Venu Madhavi ,Assistant Professor**

LESSONPLAN

(L1-Remembering L2-Underst, ding,L3-Applying,L4-Analyzing,L5-Evaluation,L6-Creating)

Unit / Item No.	**Topic (s)**	Book Reference	Page (s)		Teaching Methodology	Proposed No. of Periods	Proposed Date of Handling	C
			From	To				
I	**INTRODUCTION, FLOW GRAPHS AND PATH TESTING**					**15**		
1.1	Importance of the subject, Course objectives and Outcomes, Purpose of testing	T-1	1	3	PPT	01	22-3-21	C
1.2	Goals of testing, Phases in Tester's Mental life	T-1	3	6	PPT	01	22-3-21	C
1.3	Test Design,Testing is Not Everything, Pesticide Paradox and Complexity Barrier	T-1	7	9	PPT	01	22-3-21	C
1.4	Dichotomies	T-1	9	15	PPT	02	23-3-21	C
1.5	A Model for testing	T-1	15	20	PPT	01	23-3-21	C
1.6	Tests, Role of models	T-1	20	27	PPT	01	23-3-21	C
1.7	Consequences of bugs	T-1	27	28	PPT	01	06-4-21	C
1.8	How Bugs Affects us	T-1	28	33	PPT	01	06-4-21	C
1.9	Taxonomy of bugs	T-1	33	57	PPT	02	07-4-21	C
1.10	Basic concepts of path testing	T-1	59	90	PPT	02	07-4-21	C
1.11	Predicates, Path predicates and achievable paths	T-1	92	99	PPT	01	07-4-21	C
1.12	Path sensitizing, Path instrumentation	T-1	101	112	PPT	01		C
1.13	Application of path testing	T-1	115	117	PPT	01		C

II	**TRANSACTION FLOW TESTING, DATAFLOW TESTING**					**12**		
2.1	Transaction flow testing	T-1,R-1	122	131	PPT	03	08-04-21	C
2.2	Transaction flow testing Techniques	T-1,R-1	133	139	PPT	03	08-04-21	C
2.3	Dataflow testing: Basics of dataflow testing	T-1,R-1	145	157	PPT	03	08-04-21	C
2.4	Strategies in dataflow Testing	T-1,R-1	161	167	PPT	02	12-04-21	C
2.5	Applications of dataflow Testing	T-1	168	171	PPT	01		C
	Review	**Signature of the HOD/Coordinator**						
	UNIT- III							
III	**DOMAIN TESTING**					**12**		
3.1	Domains and paths	T-1,R-1	173	179	PPT	03	20-04-21	C
3.2	Nice & ugly domains	T-1	182	188	PPT	02	20-04-21	C
3.3	Domain testing	T-1	192	201	PPT	02	26-04-21	C
3.4	Domains and interfaces testing	T-1,R-1	202	206	PPT	02	26-04-21	C
3.5	Domains and testability	T-1,R-1	207	210	PPT	02	27-04-21	C
3.6	Revision ofdomain Testing	T-1			PPT	01	27-04-21	C
	Review	**Signature of the HOD/Coordinator**						
	UNIT-IV							
IV	**PATHS, PATH PRODUCTS AND REGULAR EXPRESSIONS,GRAPH MATRICES AND APPLICATION**					**14**		
4.1	Path products & path Expressions	T-1,R-1	244	250	PPT	02	10-05-21	C
4.2	Reduction procedure	T-1,R-1	251	256	PPT	02	10-05-21	C
4.3	Applications	T-1,R-2	257	277	PPT	01	11-05-21	C
4.4	Regular expressions	T-1,R-2	278	280	PPT	01	11-05-21	C
4.5	Flow anomaly detection	T-1	281	282	PPT	01	12-05-21	C

4.6	Logic based testingOverview	T-1	320	322	PPT	01	12-05-21	C
4.7	Decision tables	T-1	322	329	PPT	02	13-05-21	C
4.8	Path expressions	T-1	332	341	PPT	02	13-05-21	C
4.9	KV charts	T-1	343	352	PPT	01	17-05-21	C
4.10	Specifications	T-1	352	357	PPT	01	17-05-21	C
	UNIT-V							
V	**STATES,STATE GRAPHS,AND TRANSITION TESTING,GRAPH MATRICES AND APPLICATIONS**					12		
5.1	State graphs	T-1	363	369	PPT	01	18-05-21	C
5.2	Good & Bad state graphs	T-1	373	386	PPT	01	18-05-21	C
5.3	State testing	T-1	387	390	PPT	2		C
5.4	Testability tips	T-1	391	394	PPT	1		C
5.5	Graph Matrices overview	T-1	397	398	PPT	1		C
5.6	Matrix of graph, relations, power of a matrix	T-1	399	414	PPT	2		C
5.7	Node reduction algorithm	T-1	415	420	PPT	1		C
5.8	Building tools	T-1	421	426	PPT	1		C
	Review	**Signature of the HOD/Coordinator**						

MID-I ASSIGNMENT

Batch. No	Assignment Questions	Register Number	Course outcome	Level
1.	1. What is the primary purpose of testing in software development? 2. Explain the concept of dichotomies in software testing. 3. Explain what is a predicate in the context of path testing 4. Explain the differences between Nice domains and Ugly domains. 5. Explain the concept of domains and paths in domain testing	23D41A7202 To 23D41A7207	CO 1 CO 1 CO 1	1 2 1 2 2

			CO 2 CO2	
2.	1. Explain the concept of taxonomy of bugs in software testing. 2. What is the consequence of a bug in a software system? 3. Explain the basics of dataflow testing. 4. Describe the strategies used in dataflow testing. 5. Define Data-Flow testing and explain in detail	23D41A7209 To 23D41A7213	CO 1 CO 1 CO2 CO 2 CO 2	2 1 2 3 1
3.	1. Describe the transaction flow testing Techniques? 2. Explain the purpose of transaction flow testing techniques? 3.Describe the application of Data-flow Testing 4. What is the consequence of a bug in a software system? 5. Explain the difference between a path predicate and an achievable path?	23D41A7214 To 23D41A7218	CO 2 CO 2 CO 2 CO 1 CO1	3 2 3 1 2
4.	1. Describe the process of path sensitizing in path testing? 2. Describe the application of path testing in software development? 3. Evaluate the effectiveness of path testing in identifying bugs in a software system? 4. Define Domain Testing and Explain? 5. What is transaction flow testing? Explain	23D41A7219 To 23D41A7223	CO 1 CO 1 CO 1 CO 2 CO 2	3 3 6 1 1

5.	1. Design a simple path testing strategy for a given program? 2. Compare and contrast different types of bugs that can occur in a software system? 3. Explain in detail Domain and Interface Testing? 4. Evaluate the effectiveness of domain testing in identifying defects? 5. Discuss Path Sums and Path Product?	23D41A7224 To 23D41A7228	CO1 CO1 CO2 CO2 CO3	5 4 1 6 6
6.	1. Write Short Notes on (a)Distributive Laws (b) Absorption Rules (c) Loop Identity elements? 2. Explain reduction procedure with example? 3. Explicate Regular Expressions and Flow Anomaly detection? 4. Describe Nice domains and Ugly domains in domain testing? 5. Design a simple path testing strategy for a given program?	 23D41A7229 To 23D41A7233	CO3 CO3 CO3 CO2 CO1	3 1 2 3 5
7.	1. Discuss in detail about applications of path expressions? 2. Discuss mean processing time and push/pop node reduction methods? 3. Explain reduction procedure? 4. Explain the concept of taxonomy of bugs in software testing. 5.Define Data-Flow testing and explain in detail	 23D41A7234 To 23D41A7238	CO3 CO3 CO3 CO1 CO2	6 6 3,5 2 1
8.	1. What is the primary purpose of testing in software development? 2. Explain the concept of dichotomies in software testing? 3. Explain what is a predicate in the context of path testing? 4. Explain the differences between Nice domains and Ugly domains? 5. Explain the concept of domains and paths in domain testing?	 23D41A7239 To 23D41A7243	CO1 CO1 CO1 CO2 CO2	1 2 1 2 2
9.	1. Describe the transaction flow testing Techniques? 2. Explain the purpose of transaction flow testing techniques? 3. Describe the application of Data-flow Testing? 4. What is the consequence of a bug in a software system? 5. Explain the difference between a path predicate and an achievable path?	 23D41A7244 To 23D41A7248	CO2 CO2 CO2 CO1 CO1	3 2 3 1 2

10.	1. Design a simple path testing strategy for a given program? 2. Compare and contrast different types of bugs that can occur in a software system? 3. Explain in detail Domain and Interface Testing? 4. Evaluate the effectiveness of domain testing in identifying defects?		CO1	5
	5. Discuss Path Sums and Path Product?	23D41A7249 To 23D41A7353	CO1 CO2 CO2 CO3	4 1 6 6
11.	1. What is the primary purpose of testing in software development? 2. Explain the concept of dichotomies in software testing? 3. Discuss mean processing time and push/pop node reduction methods? 4. Explain reduction procedure with example? 5. Explain the difference between a path predicate and an achievable path?	23D41A7254 To 23D41A7258	CO1 CO1 CO3 CO3 CO1	1 2 6 3,5 2
12.	1. Discuss in detail about applications of path expressions? 2. Discuss mean processing time and push/pop node reduction methods? 3. Explain reduction procedure? 4. Explain the concept of taxonomy of bugs in software testing? 5. Define Data-Flow testing and explain in detail?	23D41A7259 To 23D41A7263	CO3 CO3 CO3 CO1 CO2	6 6 3,5 1 2
13.	1. Write Short Notes on (a)Distributive Laws (b) Absorption Rules (c) Loop Identity elements? 2. Explain reduction procedure with example? 3. Explicate Regular Expressions and Flow Anomaly detection? 4. Describe Nice domains and Ugly domains in domain testing? 5. Design a simple path testing strategy for a given program?	24D45A7201 TO 24D45A7204	CO3 CO3 CO3 CO2 CO1	3 1 2 3 5

MID-II ASSIGNMENT

Batch. No	Assignment Questions	Register Number	Course outcome	Level
1.	1. Explain about building tools with examples? 2. Set down the concept of KV chart? Draw the k-map for 3 variable functions? 3. What are decision tables? Illustrate the applications of decision tables. How is a decision table Useful in testing? 4. Write in detail about logical based testing? 5. Explain condition stub and action stub?	23D41A7202 To 23D41A7207	CO5 CO3 CO3 CO4 CO4	3 3 1 2 1
2.	1. Explain in detail Path Expressions in Logic based testing? 2. Compare and contrast decision tables and path expressions in logic-based testing. 3. Compare and contrast path products and path expressions. 4. Describe the reduction procedure for path expressions? 5. Explain in detail KV-Charts?	23D41A7209 To 23D41A7213	CO4 CO4 CO4 CO4 CO4	1 4 4 3 1
3.	1.Write short notes on i. Transition Bugs ii. Dead States iii. State Bugs iv. Encoding Bugs 2. What are the principles of state testing? Discuss advantages and disadvantages? 3. Write the design guidelines for building finite state machine into code? 4. Discuss about good and bad state graphs? 5. Explain about building tools with examples?	23D41A7214 To 23D41A7218	CO5 CO5 CO2 CO5 CO5	2 2 5 2 3
4.	1. What are decision tables? Illustrate the applications of decision tables. How is a decision table Useful in testing? 2. Write in detail about logical based testing? 3. Explain condition stub and action stub? 4. Explain in detail Path Expressions in Logic based testing? 5. Compare and contrast decision tables and path expressions in logic-based testing.	23D41A7219 To 23D41A7223	CO3 CO4 CO4 CO4 CO4	1 2 1 1 4
5.	1. What are the principles of state testing? Discuss advantages and disadvantages? 2. Discuss about good and bad state graphs? 3. What are the software implementation issues in state testing? 4. Define State Graph, Dead states and Equivalence states and Explain?	23D41A7224 To 23D41A7228	CO5 CO5 CO5 CO5	2 2 2 1 3

	5. Explain State transition table in detail?		CO5	
6.	1. Explain impact of bugs in state testing? 2. Explain switches, tags and Unachievable paths? 3. Explain in detail Limitations and Extensions in State Testing? 4. Describe Nice domains and Ugly domains in domain testing 5. Explain Inputs and Transitions in State Graphs?	23D41A7229 To 23D41A7233	CO5 CO5 CO5 CO2 CO5	3 1 2 3 2
7.	1. Explain in detail Limitations and Extensions in State Testing? 2. Discuss mean processing time and push/pop node reduction methods? 3. Set down the concept of KV chart? 4. Discuss about good and bad state graphs? 5. Explain State, state graphs and Transition Testing?	23D41A7234 To 23D41A7238	CO3 CO5 CO3 CO5 CO5	6 2 3,5 2 1
8.	1. Define Graph matrix? Explain? 2. Define Reflexive Relations? Define directed & undirected Graph? 3. Explain in detail Node-Reduction Algorithm? 4. What are the applications of Node-Reduction Algorithm? 5. Discuss about Matrix representation in software in building tools?	23D41A7239 To 23D41A7243	CO5 CO6 CO6 CO6 CO6	2 2 3 2 6
9.	1. Discuss Node Reduction Optimization of building tools? 2. Explain about the Partitioning Algorithm. 3. What is a graph matrix, and how is it used to represent a graph? 4. Analyze the impact of node reduction in simplifying the representation of a graph 5. Explain in details Relations in Graph Matrices?	23D41A7244 To 23D41A7248	CO6 CO6 CO6 CO6 CO6	6 3 1 5 3
10.	1. What is Matrix Powers and Products? 2. How an Algorithm is partitioned? Explain? 3. What are the principles of Powers of Matrix and Explain Loops? 4. Explain Properties of Relations? 5. Discuss the problem with pictorial graphs?	23D41A7249 To 23D41A7253	CO6 CO6 CO6 CO6 CO6	2 4 3 4 5
11.	1. Explain in detail Limitations and Extensions in State Testing? 2. Discuss mean processing time and push/pop node reduction methods 3. Set down the concept of KV chart? 4. Discuss about good and bad state graphs? 5. Explain State, state graphs and Transition Testing?	23D41A7254 To 23D41A7258	CO3 CO5 CO3 CO5 CO5	6 2 3,5 2 1

12.	1. Explain about building tools with examples? 2. Explain switches, tags and Unachievable paths? 3. Explain in detail Limitations and Extensions in State Testing? 4. Describe Nice domains and Ugly domains in domain testing 5. Explain Inputs and Transitions in State Graphs?	23D41A7259 To 23D41A7263	CO5 CO5 CO5 CO2 CO5	3 1 2 3 2
13.	1. Explain Properties of Relations? 2. Discuss the problem with pictorial graphs? 3. What are the principles of state testing? Discuss advantages and disadvantages? 4. Discuss about good and bad state graphs? 5. What are the software implementation issues in state testing?	24D45A7201 TO 24D45A7204	CO5 CO5 CO5 CO5 CO5	4 5 2 2 2

Batch. No	Assignment Questions	Register Number	Course outcome	Level
1.	1. Explain about building tools with examples? 2. Set down the concept of KV chart? Draw the k-map for 3 variable functions? 3. What are decision tables? Illustrate the applications of decision tables. How is a decision table Useful in testing? 4. Write in detail about logical based testing? 5. Explain condition stub and action stub?	23D41A7202 To 23D41A7207	CO5 CO3 CO3 CO4 CO4	3 3 1 2 1
2.	1. Explain in detail Path Expressions in Logic based testing? 2. Compare and contrast decision tables and path expressions in logic-based testing. 3. Compare and contrast path products and path expressions. 4. Describe the reduction procedure for path expressions? 5. Explain in detail KV-Charts?	23D41A7209 To 23D41A7213	CO4 CO4 CO4 CO4 CO4	1 4 4 3 1
3.	1.Write short notes on v. Transition Bugs vi. Dead States vii. State Bugs viii. Encoding Bugs 2. What are the principles of state testing? Discuss advantages and disadvantages? 3. Write the design guidelines for building finite state machine into code? 4. Discuss about good and bad state graphs? 5. Explain about building tools with examples?	23D41A7214 To 23D41A7218	CO5 CO5 CO2 CO5 CO5	2 2 5 2 3
4.	1. What are decision tables? Illustrate the applications of decision tables. How is a decision table Useful in testing? 2. Write in detail about logical based testing? 3. Explain condition stub and action stub? 4. Explain in detail Path Expressions in Logic based testing? 5. Compare and contrast decision tables and path expressions in logic-based testing.	23D41A7219 To 23D41A7223	CO3 CO4 CO4 CO4 CO4	1 2 1 1 4
5.	6. What are the principles of state testing? Discuss advantages and disadvantages? 7. Discuss about good and bad state graphs? 8. What are the software implementation issues in state testing? 9. Define State Graph, Dead states and Equivalence states and Explain? 10. Explain State transition table in detail?	23D41A7224 To 23D41A7228	CO5 CO5 CO5 CO5 CO5	2 2 2 1 3

6.	1. Explain impact of bugs in state testing? 2. Explain switches, tags and Unachievable paths? 3. Explain in detail Limitations and Extensions in State Testing? 4. Describe Nice domains and Ugly domains in domain testing 5. Explain Inputs and Transitions in State Graphs?	23D41A7229 To 23D41A7233	CO5 CO5 CO5 CO2 CO5	3 1 2 3 2
7.	1. Explain in detail Limitations and Extensions in State Testing? 2. Discuss mean processing time and push/pop node reduction methods? 6. Set down the concept of KV chart? 7. Discuss about good and bad state graphs? 8. Explain State, state graphs and Transition Testing?	23D41A7234 To 23D41A7238	CO3 CO5 CO3 CO5 CO5	6 2 3,5 2 1
8.	1. Define Graph matrix? Explain? 2. Define Reflexive Relations? Define directed & undirected Graph? 3. Explain in detail Node-Reduction Algorithm? 4. What are the applications of Node-Reduction Algorithm? 5. Discuss about Matrix representation in software in building tools?	23D41A7239 To 23D41A7243	CO5 CO6 CO6 CO6 CO6	2 2 3 2 6
9.	6. Discuss Node Reduction Optimization of building tools? 7. Explain about the Partitioning Algorithm. 8. What is a graph matrix, and how is it used to represent a graph? 9. Analyze the impact of node reduction in simplifying the representation of a graph 10. Explain in details Relations in Graph Matrices?	23D41A7244 To 23D41A7248	CO6 CO6 CO6 CO6 CO6	6 3 1 5 3
10.	6. What is Matrix Powers and Products? 7. How an Algorithm is partitioned? Explain? 8. What are the principles of Powers of Matrix and Explain Loops? 9. Explain Properties of Relations? 10. Discuss the problem with pictorial graphs?	23D41A7249 To 23D41A7253	CO6 CO6 CO6 CO6 CO6	2 4 3 4 5
11.	1. Explain in detail Limitations and Extensions in State Testing? 2. Discuss mean processing time and push/pop node reduction methods 3. Set down the concept of KV chart? 4. Discuss about good and bad state graphs? 5. Explain State, state graphs and Transition Testing?	23D41A7254 To 23D41A7258	CO3 CO5 CO3 CO5 CO5	6 2 3,5 2 1
12.	1. Explain about building tools with examples? 2. Explain switches, tags and Unachievable paths? 3. Explain in detail Limitations and Extensions in State Testing? 4. Describe Nice domains and Ugly domains in domain testing 5. Explain Inputs and Transitions in State Graphs?	23D41A7259 To 23D41A7263	CO5 CO5 CO5 CO2 CO5	3 1 2 3 2

13.	1. Explain Properties of Relations? 2. Discuss the problem with pictorial graphs? 3. What are the principles of state testing? Discuss advantages and disadvantages? 4. Discuss about good and bad state graphs? 5. What are the software implementation issues in state testing?	24D45A7201 TO 24D45A7204	CO5 CO5 CO5 CO5 CO5	 4 5 2 2 2

QUESTION BANK

NIT-I	SRI INDU COLLEGE OF ENGG & TECHOLOGY		
	QUESTION BANK (Regulation: QR22) Multiple Choice Questions Department of AI&DS	BT Level	Course Outcome
	Sub. Code & Title: R22CSE3244 & SOFTWARE TESTING METHODOLOGIES		
	Academic Year: 2025-26 / Year/Sem./Section: III B.TECH II-SEM		
	Faculty Names& Designation: A.Venu Madhavi, Asst.Prof.(AI&DS)		
C-1	ary purpose of software testing? e D) To ensure reliability C) To improve performance D) To reduce costs	1	CO1
C-2	Which of the following is a dichotomy in software testing? A) Manual vs. automated testing B) Black box vs. white box testing C) Unit testing vs. integration testing D) All of the above	1	CO1
C-3	What is a model for testing? A) A framework for testing B) A testing methodology C) A testing tool D) A testing technique	1	CO1
C-4	What are the consequences of bugs in software? A) Financial loss B) Reputation damage C) User frustration D) All of the above	2	CO1
C-5	Evaluate the consequences of not using path testing in a software development project. Which of the following statements is most accurate? A) The software will likely have fewer defects. B) The software will likely have more defects. C) The software will have the same number of defects. D) The software will be more efficient.	6	CO1
C-6	What are flow graphs used for in software testing? A) To model program flow	2	CO1

	B) To identify defects C) To prioritize testing D) To optimize performance		
C-7	Design a testing strategy that incorporates path testing and risk-based testing. Which of the following strategies would be most effective? A) Use path testing to identify critical paths and then use risk-based testing to prioritize testing of those paths. B) Use path testing to identify all possible paths and then use risk-based testing to identify high-risk paths. C) Use path testing to identify frequently executed paths and then use risk-based testing to identify low-risk paths. D) Use path testing to identify complex paths and then use risk-based testing to identify high-risk paths.	5	CO1
C-8	Analyze the concept of path testing and identify the primary advantage of using this approach. Which of the following statements is most accurate? A) Path testing ensures that all possible paths through the code are executed. B) Path testing identifies the most critical paths through the code. C) Path testing reduces the number of test cases required. D) Path testing improves code coverage.	4	CO1
C-9	What is the purpose of path sensitizing? A) To identify achievable paths B) To prioritize paths for testing C) To optimize path instrumentation D) To eliminate unnecessary paths	3	CO1
C-10	How does path instrumentation aid in path testing? A) By monitoring and controlling path execution B) By optimizing path execution C) By eliminating unnecessary paths D) By prioritizing paths for testing	3	CO1
C-11	What is the application of path testing? A) Identifying defects in code B) Improving performance of code C) Optimizing code for execution D) All of the above	2	CO1
C-12	Analyze the relationship between predicates and path predicates in a software system. Which of the following statements is most accurate? A) Predicates and path predicates are mutually exclusive. B) Predicates are a type of path predicate. C) Path predicates are a type of predicate. D) Predicates and path predicates are unrelated.	4	CO1
C-13	What is the relationship between predicates and path predicates? A) Predicates are used to determine path predicates B) Path predicates are used to determine predicates C) Predicates and path predicates are unrelated D) Predicates are a type of path predicate	3	CO1

C-14	Analyze the concept of achievable paths in path testing. Which of the following statements is most accurate? A) Achievable paths are those that can be executed with valid input. B) Achievable paths are those that cannot be executed with valid input. C) Achievable paths are those that are frequently executed. D) Achievable paths are those that are critical to the software's functionality.	4	CO1
C-15	Evaluate the effectiveness of path testing in improving code coverage. Which of the following statements is most accurate? A) Path testing ensures 100% code coverage. B) Path testing improves code coverage but does not guarantee 100% coverage. C) Path testing does not improve code coverage. D) Path testing reduces code coverage.	6	CO1

Filling The Blank Questions

F-1	The primary purpose of software testing is to _____ defects in the software.	1	CO1
F-2	A model for testing is a _____ that guides the testing process.	2	CO1
F-3	The consequences of bugs in software can include _____ damage to a company's reputation.	1	CO1
F-4	Path testing is a type of testing that involves analyzing the _____ of a program.	3	CO1
F -5	The goal of path sensitizing is to identify the _____ that affect the execution of a particular path.	4	CO1
F -6	A predicate is a _____ that is used to control the flow of a program.	1	CO1
F -7	Path predicates are used to determine which _____ to take in a program.	2	CO1
F -8	Achievable paths are those that can be _____ with valid input.	3	CO1
F -9	Path instrumentation involves adding _____ to a program to monitor and control its execution.	5	CO1
F -10	The application of path testing can help to _____ the reliability and quality of a software system.	6	CO1
F -11	The taxonomy of bugs refers to the _____ of defects in software.	1	CO1
F -12	Flow graphs are a type of _____ that is used to represent the flow of a program.	2	CO1
F -13	Path testing can be used to identify _____ in a program.	3	CO1
F -14	The use of path testing can help to _____ the number of defects in a software system.	4	CO1
F -15	The integration of path testing with other testing techniques can help to _____ the overall quality of a software system.	5	CO1

Match The Following Questions

M-1		i. Predicate	A) A condition that affects program flow	I	CO1
		ii. Path predicate	B) A path that can be executed with valid input		
		iii. Achievable path	C) A condition that determines which path to take		
		iv. Path sensitizing	D) A technique used to identify conditions that affect program		

			flow		
M-2		i. Path testing	A) A technique that involves analyzing the flow of a program	II	CO1
		ii. Flow graph testing	B) A technique that involves analyzing the paths of a program		
		iii. Path instrumentation	C) A technique used to monitor and control program execution		
		iv. Path sensitizing	D) A technique used to identify conditions that affect program flow		
M-3		i. Path instrumentation	A) Used to monitor and control program execution	III	CO1
		ii. Path sensitizing	B) Used to identify conditions th affect program flow		
		iii. Achievable paths	C) Used to identify paths that ca be executed with valid input		
		iv. Path predicates	D) Used to determine which patl to take		
M-4		i. Improved code coverage	A) Path testing	IV	CO1
		ii. Increased defect detection	B) Flow graph testing		
		iii. Reduced testing time	C) Path instrumentation		
		iv. Improved reliability	D) Path sensitizing		
M-5		i. Integration of path testing with other testing techniques	A) To improve the overall qualit of a software system	V	CO1
		ii. Use of path testing to identify critical paths	B) To reduce the number of defects in a software system		
		iii. Use of path testing to improve code coverage	C) To improve the reliability of software system		
		iv. Use of path testing to reduce testing time	D) To reduce the time and cost c testing		

5 MARKS QUESTIONS

D-1	What is the primary purpose of testing in software development?	1	CO1
D-2	Explain the concept of dichotomies in software testing.	2	CO1
D-3	What is the consequence of a bug in a software system?	1	CO1
D-4	Explain the concept of taxonomy of bugs in software testing.	2	CO1
D-5	What is a predicate in the context of path testing?	1	CO1
D-6	Explain the difference between a path predicate and an achievable path.	2	CO1
D-7	Describe the process of path sensitizing in path testing.	3	CO1
D-8	Describe the application of path testing in software development.	3	CO1

D-9	Evaluate the effectiveness of path testing in identifying bugs in a software system.	6	CO1
D-10	Design a simple path testing strategy for a given program.	5	CO1
D-11	Compare and contrast different types of bugs that can occur in a software system.	4	CO1
D-12	What is a model in the context of software testing?	1	CO1
D-13	What is White-Box Testing? What are the advantages and disadvantages of white-box testing?	1	CO1
D-14	What is Black-Box Testing? What are the advantages and disadvantages of Black-box testing?	1	CO1
D-15	Define Software Testing and write down the different levels in software testing	1	CO1

UNIT-II

	Multiple Choice Questions	BT Level	Course Outcome
C-1	What is transaction flow testing? A) A testing technique that analyzes the flow of transactions through a system B) A testing technique that analyzes the data flow through a system C) A testing technique that analyzes the control flow through a system D) A testing technique that analyzes the interface of a system	1	CO2
C-2.	What is the primary goal of transaction flow testing? A) To ensure that all transactions are processed correctly B)To ensure that all data is stored correctly C)To ensure that all interfaces are working correctly D) To ensure that all systems are integrated correctly	2	CO2
2C-3.	How is dataflow testing used in software testing? A) To analyze the flow of data through a program B) To analyze the control flow through a program C) To analyze the transaction flow through a system D) To analyze the interface of a system	3	CO2
C-4.	What is domain testing? A) A testing technique that analyzes the domains of a system B) A testing technique that analyzes the interfaces of a system C) A testing technique that analyzes the control flow of a system D) A testing technique that analyzes the data flow of a system	1	CO2
C-5.	What is the primary goal of domain testing? A) To ensure that all domains are correctly defined B) To ensure that all interfaces are correctly defined C) To ensure that all control flows are correctly defined D) To ensure that all data flows are correctly defined	2	CO2
C-6.	How are nice and ugly domains used in domain testing? A) Nice domains are used to test valid inputs, while ugly domains are used to test invalid inputs B) Nice domains are used to test invalid inputs, while ugly domains are used to test valid inputs C) Nice domains are used to test control flow, while ugly domains are used to test data flow D) Nice domains are used to test data flow, while ugly domains are used to test control flow	3	CO2
C-7.	What is the relationship between domains and paths? A) Domains define the possible inputs to a system, while paths define the	4	CO2

	possible flows through a system B) Domains define the possible flows through a system, while paths define the possible inputs to a system C) Domains and paths are unrelated D) Domains and paths are the same thing		
C-8.	How can domain testing be used to improve the testability of a system? A) By identifying and testing all possible domains B) By identifying and testing all possible paths C) By identifying and testing all possible interfaces D) By identifying and testing all possible control flows	5	CO2
C-9.	Evaluate the effectiveness of domain testing in identifying defects in a system. What is the primary benefit of domain testing? A) It ensures that all possible inputs are tested B) It ensures that all possible flows through a system are tested C) It ensures that all possible interfaces are tested D) It ensures that all possible control flows are tested	6	CO2
C-10.	What is the primary goal of dataflow testing? A) To analyze the flow of data through a program B) To analyze the control flow through a program C) To analyze the transaction flow through a system D) To analyze the interface of a system	1	CO2
C-11.	What is the relationship between dataflow testing and transaction flow testing? A) Dataflow testing analyzes the flow of data, while transaction flow testing analyzes the flow of transactions B) Dataflow testing analyzes the flow of transactions, while transaction flow testing analyzes the flow of data C) Dataflow testing and transaction flow testing are the same thing D) Dataflow testing and transaction flow testing are unrelated	2	CO2
C-12.	How is dataflow testing used to identify defects in a program? A) By analyzing the flow of data through a program B) By analyzing the control flow through a program C) By analyzing the transaction flow through a system D) By analyzing the interface of a system	3	CO2
C-13.	What is the primary benefit of using dataflow testing in software testing? A) It ensures that all possible data flows are tested B) It ensures that all possible control flows are tested C) It ensures that all possible transactions are tested D) It ensures that all possible interfaces are tested	4	CO2
C-14.	How can dataflow testing be used to improve the reliability of a system? A) By identifying and testing all possible data flows B) By identifying and testing all possible control flows C) By identifying and testing all possible transactions D) By identifying and testing all possible interfaces	5	CO2
C-15	Evaluate the effectiveness of dataflow testing in identifying defects in a system. What is the primary benefit of dataflow testing? A) It ensures that all possible data flows are tested B) It ensures that all possible control flows are tested C) It ensures that all possible transactions are tested D) It ensures that all possible interfaces are tested	6	CO2
	Filling The Blank Questions		
F-1	Transaction flow testing is a software testing technique that analyzes the _____ of transactions through a system.	1	CO2

F-2	The primary goal of transaction flow testing is to ensure that all transactions are processed _____.	2	CO2
F-3	Dataflow testing is a software testing technique that analyzes the _____ of data through a program.	3	CO2
F-4	Domain testing is a software testing technique that analyzes the _____ of a system.	1	CO2
F-5	The primary goal of domain testing is to ensure that all domains are correctly _____.	2	CO2
F-6	Nice domains are used in domain testing to test _____ inputs.	3	CO2
F-7	Ugly domains are used in domain testing to test _____ inputs.	4	CO2
F-8	Domain testing can be used to improve the _____ of a system.	5	CO2
F-9	The effectiveness of domain testing in identifying defects in a system can be evaluated by analyzing the _____ of the system.	6	CO2
F-10	Dataflow testing is used to analyze the _____ of data through a program.	1	CO2
F-11	The primary goal of dataflow testing is to ensure that all data flows are correctly _____.	2	CO2
F-12	Dataflow testing can be used to identify _____ in a program.	3	CO2
F-13	The application of dataflow testing can be used to improve the _____ of a system.	4	CO2
F-14	Dataflow testing can be used to improve the _____ of a system	5	CO2
F-15	The effectiveness of dataflow testing in identifying defects in a system can be evaluated by analyzing the _____ of the system.	6	CO2
	Match The Following Questions		

<table>
<tr>
<td>M-1</td>
<td>
<table>
<tr><td>i. Transaction flow testing</td><td>A) A testing technique that analyzes the flow of transactions through a system</td></tr>
<tr><td>ii. Dataflow testing</td><td>B) A testing technique that analyzes the flow of data through a program</td></tr>
<tr><td>iii. Domain testing</td><td>C) A testing technique that analyzes the domains of a system</td></tr>
<tr><td>iv. Interface testing</td><td>D) A testing technique that analyzes the interfaces of a system</td></tr>
</table>
</td>
<td>i</td>
<td>CO2</td>
</tr>
<tr>
<td>M-2</td>
<td>
<table>
<tr><td>i. Transaction flow testing</td><td>A) To ensure that all transactions are processed correctly</td></tr>
<tr><td>ii. Dataflow testing</td><td>B) To ensure that all data flows are correct</td></tr>
<tr><td>iii. Domain testing</td><td>C) To ensure that all domains are correctly defined</td></tr>
<tr><td>iv. Interface testing</td><td>D) To ensure that all interfaces are working correctly</td></tr>
</table>
</td>
<td>ii</td>
<td>CO2</td>
</tr>
</table>

M-3	<table><tr><td>i. Nice domains</td><td>A) Used to test valid inputs</td></tr><tr><td>ii. Ugly domains</td><td>B) Used to test invalid inputs</td></tr><tr><td>iii. Dataflow testing</td><td>C) Used to analyze the flow of data through a program</td></tr><tr><td>iv. Domain testing</td><td>D) Used to analyze the domains of a system</td></tr></table>	iii	CO2
M-4	<table><tr><td>i. Improved reliability</td><td>A) Transaction flow testing</td></tr><tr><td>ii. Increased testability</td><td>B) Dataflow testing</td></tr><tr><td>iii. Reduced defects</td><td>C) Domain testing</td></tr><tr><td>iv. Improved performance</td><td>D) Interface testing</td></tr></table>	iv	CO2
M-5	<table><tr><td>i. Integration of transaction flow testing with dataflow testing</td><td>A) To improve the overall quality of a system</td></tr><tr><td>ii. Use of domain testing to improve testability</td><td>B) To reduce the number of defects in a system</td></tr><tr><td>iii. Use of dataflow testing to analyze the flow of data</td><td>C) To improve the reliability of a system</td></tr><tr><td>iv. Use of interface testing to ensure correct interfaces</td><td>D) To improve the testability of a system</td></tr></table>	v	CO2

5 MARKS QUESTIONS

2D-1	Explain the differences between Nice domains and Ugly domains	2	CO2
2D-2	Explain the concept of domains and paths in domain testing.	2	CO2
2D-3	Explain the basics of dataflow testing.	2	CO2
2D-4	Describe the strategies used in dataflow testing.	3	CO2
2D-5	Define Data-Flow testing and explain in detail?	1	CO2
2D-6	Describe the transaction flow testing Techniques.	3	CO2
2D-7	Explain the purpose of transaction flow testing techniques.	2	CO2

2D-8	Describe the application of Data-flow Testing	3	CO2
2D-9	Compare and contrast transaction flow testing and dataflow testing.	4	CO2
2D-10	Illustrate data flow model? Discuss various components of it?	4	CO2
2D-11	What are data-flow anomalies? How data flow testing can explore them?	2	CO2
2D-12	Evaluate the effectiveness of domain testing in identifying defects.	6	CO2
2D-13	Describe Nice domains and Ugly domains in domain testing.	3	CO2
2D-14	Explain in detail Domain and Interface Testing?	1	CO2
2D-15	Explain in detail Domain and Testability	1	CO2

UNIT-III

	Multiple Choice Questions	BT Level	Course Outcome
C-1	What is a path product in software testing? A) A representation of a path through a program B) A representation of a decision table C) A representation of a regular expression D) A representation of a flowchart	1	CO3
C-2	What is the purpose of a reduction procedure in path products? A) To simplify the path product B) To complicate the path product C) To eliminate redundant paths D) To add new paths	2	CO3
C-3	How are regular expressions used in software testing? A) To detect flow anomalies B) To detect syntax errors C) To detect semantic errors D) To detect runtime errors	3	CO3
C-4	What is logic-based testing in software testing? A) A testing technique that uses logical operators B) A testing technique that uses decision tables C) A testing technique that uses path expressions D) A testing technique that uses regular expressions	1	CO3
C-5	What is the purpose of a decision table in logic-based testing? A) To represent complex logic B) To represent simple logic C) To represent conditional statements D) To represent loops	2	CO3
C-6	How are path expressions used in logic-based testing? A) To represent paths through a program B) To represent decisions in a program C) To represent loops in a program D) To represent conditional statements in a program	3	CO3
C-7	What is the advantage of using logic-based testing in software testing? A) It reduces the number of test cases B) It increases the number of test cases C) It improves the coverage of test cases D) It reduces the complexity of test cases	4	CO3
C-8	How can logic-based testing be used to improve the reliability of a software system? A) By identifying and testing all possible paths	5	CO3

	B) By identifying and testing all possible decisions C) By identifying and testing all possible loops D) By identifying and testing all possible conditional statements		
C-9	Evaluate the effectiveness of logic-based testing in identifying defects in a software system. What is the primary benefit of logic-based testing? A) It ensures that all possible paths are tested B) It ensures that all possible decisions are tested C) It ensures that all possible loops are tested D) It ensures that all possible conditional statements are tested	6	CO3
C-10	What is a Karnaugh chart (K-map) in logic-based testing? A) A graphical representation of a decision table B) A graphical representation of a path expression C) A graphical representation of a regular expression D) A graphical representation of a flowchart	1	CO4
C-11	What is the purpose of a Karnaugh chart (K-map) in logic-based testing? A) To simplify complex logic B) To complicate simple logic C) To represent conditional statements D) To represent loops	2	CO4
C-12	How are Karnaugh charts (K-maps) used in logic-based testing? A) To identify and test all possible paths B) To identify and test all possible decisions C) To identify and test all possible loops D) To identify and test all possible conditional statements	3	CO4
C-13	What is the advantage of using Karnaugh charts (K-maps) in logic-based testing? A) It reduces the number of test cases B) It increases the number of test cases C) It improves the coverage of test cases D) It reduces the complexity of test cases	4	CO4
C-14	How can Karnaugh charts (K-maps) be used to improve the reliability of a software system? A) By identifying and testing all possible paths B) By identifying and testing all possible decisions C) By identifying and testing all possible loops D) By identifying and testing all possible conditional statements	5	CO4
C-15	Evaluate the effectiveness of Karnaugh charts (K-maps) in identifying defects in a software system. What is the primary benefit of using Karnaugh charts (K-maps)? A) It ensures that all possible paths are tested B) It ensures that all possible decisions are tested C) It ensures that all possible loops are tested D) It ensures that all possible conditional statements are tested	6	CO4
	Filling The Blank Questions		
F-1	Path products are used in software testing to represent the _____ of a path through a program.	1	CO3
F-2	The reduction procedure is used in path products to _____ the path product.	2	CO3
F-3	Regular expressions are used in software testing to detect _____ in the flow of a program.	3	CO3

<table>
<tr><td>F-4</td><td>Logic-based testing is a software testing technique that uses _____ to represent the logic of a program.</td><td>1</td><td>CO3</td></tr>
<tr><td>F-5</td><td>Decision tables are used in logic-based testing to represent _____ logic.</td><td>2</td><td>CO3</td></tr>
<tr><td>F-6</td><td>Path expressions are used in logic-based testing to represent the _____ of a path through a program.</td><td>3</td><td>CO3</td></tr>
<tr><td>F-7</td><td>Karnaugh charts (K-maps) are used in logic-based testing to _____ complex logic.</td><td>4</td><td>CO3</td></tr>
<tr><td>F-8</td><td>The application of logic-based testing can be used to improve the _____ of a software system.</td><td>5</td><td>CO3</td></tr>
<tr><td>F-9</td><td>The effectiveness of logic-based testing in identifying defects in a software system can be evaluated by analyzing the _____ of the system</td><td>6</td><td>CO4</td></tr>
<tr><td>F-10</td><td>Path products are used in software testing to analyze the _____ of a path through a program.</td><td>1</td><td>CO4</td></tr>
<tr><td>F-11</td><td>The reduction procedure is used in path products to eliminate _____ paths.</td><td>2</td><td>CO4</td></tr>
<tr><td>F-12</td><td>Regular expressions are used in software testing to detect _____ in the flow of a program.</td><td>3</td><td>CO4</td></tr>
<tr><td>F-13</td><td>Logic-based testing can be used to identify _____ in a software system.</td><td>4</td><td>CO4</td></tr>
<tr><td>F-14</td><td>The application of logic-based testing can be used to improve the _____ of a software system.</td><td>5</td><td>CO4</td></tr>
<tr><td>F-15</td><td>The effectiveness of logic-based testing in identifying defects in a software system can be evaluated by analyzing the _____ of the system.</td><td>6</td><td>CO4</td></tr>
<tr><td></td><td colspan="3">Match The Following Questions</td></tr>
<tr><td>M-1</td><td colspan="3"><table>
<tr><td>i. Path products</td><td>A) A representation of a path through a program</td></tr>
<tr><td>ii. Path expressions</td><td>B) A representation of a decision in a program</td></tr>
<tr><td>iii. Decision tables</td><td>C) A graphical representation of a decision table</td></tr>
<tr><td>iv. Karnaugh charts</td><td>D) A representation of a path expression</td></tr>
</table></td><td>I</td><td>CO3</td></tr>
</table>

M-2	<table><tr><td>i. Path products</td><td>A) Used to detect flow anomalies in a program</td></tr><tr><td>ii. Regular expressions</td><td>B) Used to represent complex logic in a program</td></tr><tr><td>iii. Decision tables</td><td>C) Used to analyze the flow of a program</td></tr><tr><td>iv. Logic-based testing</td><td>D) Used to improve the reliability of a software system</td></tr></table>	II	CO3
M-3	<table><tr><td>i. Path expressions</td><td>A) Improves the coverage of test cases</td></tr><tr><td>ii. Decision tables</td><td>B) Reduces the complexity of test cases</td></tr><tr><td>iii. Karnaugh charts</td><td>C) Improves the reliability of a software system</td></tr><tr><td>iv. Regular expressions</td><td>D) Detects flow anomalies in a program</td></tr></table>	III	CO4
M-4	<table><tr><td>i. Path products</td><td>A) Can be complex to implement</td></tr><tr><td>ii. Regular expressions</td><td>B) Can be difficult to maintain</td></tr><tr><td>iii. Decision tables</td><td>C) May not detect all defects</td></tr><tr><td>iv. Logic-based testing</td><td>D) Can be time-consuming to test</td></tr></table>	IV	CO4
M-5	<table><tr><td>i. Integration of path products with regular expressions</td><td>A) To improve the overall quality of a software system</td></tr><tr><td>ii. Use of decision tables to improve test coverage</td><td>B) To reduce the number of defects in a software system</td></tr><tr><td>iii. Use of Karnaugh charts to simplify complex logic</td><td>C) To improve the reliability of a software system</td></tr><tr><td>iv. Use of logic-based testing to improve reliability</td><td>D) To improve the testability of a software system</td></tr></table>	V	CO4

5 MARKS QUESTIONS

D-1	Write Short Notes on i. Distributive Laws ii. Absorption Rules iii. Loop iv. Identity elements	3	CO3
D-2	Explain Huang's theorem with an example?	1	CO3
D-3	Explicate Regular Expressions and Flow Anomaly detection?	2	CO3
D-4	Discuss Path Sums and Path Product?	6	CO3
D-5	Discuss in detail about applications of path expressions?	6	CO3
D-6	Discuss mean processing time and push/pop node reduction methods?	6	CO3
D-7	Set down the concept of KV chart? Draw the k-map for 3 variable functions?	3,5	CO3
D-8	What are decision tables? Illustrate the applications of decision tables. How is a decision table Useful in testing?	1	CO3
D-9	Write in detail about logical based testing?	2	CO4
D-10	Explain condition stub and action stub?	1	CO4
D-11	Explain in detail Path Expressions in Logic based testing?	1	CO4
D-12	Compare and contrast decision tables and path expressions in logic-based testing.	4	CO4
D-13	Compare and contrast path products and path expressions.	4	CO4
D-14	Describe the reduction procedure for path expressions?	3	CO4
D-15	Explain in detail KV-Charts?	1	CO4

UNIT IV

Multiple Choice Questions			Course Outcome
C-1	What is a state graph in software testing? A) A graphical representation of a program's states and transitions B) A graphical representation of a program's inputs and outputs C) A graphical representation of a program's control flow D) A graphical representation of a program's data flow	1	CO5
C-2	What is the purpose of a state graph in software testing? A) To identify all possible states of a program B) To identify all possible transitions between states C) To identify all possible inputs to a program D) To identify all possible outputs of a program	2	CO5
C-3	How are state graphs used in state testing? A) To identify all possible states of a program B) To identify all possible transitions between states C) To identify all possible inputs to a program D) To identify all possible outputs of a program	3	CO5
C-4	What is a good state graph in software testing? A) A state graph that has a small number of states B) A state graph that has a large number of states C) A state graph that has a small number of transitions D) A state graph that has a large number of transitions	1	CO5
C-5	What is a bad state graph in software testing? A) A state graph that has a large number of states B) A state graph that has a small number of states C) A state graph that has a large number of transitions	2	CO5

	D) A state graph that has a small number of transitions		
C-6	How can a state graph be used to improve the testability of a program? A) By identifying all possible states of a program B) By identifying all possible transitions between states C) By identifying all possible inputs to a program D) By identifying all possible outputs of a program	3	CO5
C-7	What is the advantage of using state graphs in software testing? A) It reduces the number of test cases B) It increases the number of test cases C) It improves the coverage of test cases D) It reduces the complexity of test cases	4	CO5
C-8	How can state testing be used to improve the reliability of a software system? A) By identifying all possible states of a program B) By identifying all possible transitions between states C) By identifying all possible inputs to a program D) By identifying all possible outputs of a program	5	CO5
C-9	Evaluate the effectiveness of state graphs in identifying defects in a software system. What is the primary benefit of using state graphs? A) It ensures that all possible states are tested B) It ensures that all possible transitions are tested C) It ensures that all possible inputs are tested D) It ensures that all possible outputs are tested	6	CO5
C-10	What is transition testing in software testing? A) A testing technique that involves testing all possible transitions between states B) A testing technique that involves testing all possible states of a program C) A testing technique that involves testing all possible inputs to a program D) A testing technique that involves testing all possible outputs of a program	1	CO5
C-11	What is the purpose of transition testing in software testing? A) To identify all possible transitions between states B) To identify all possible states of a program C) To identify all possible inputs to a program D) To identify all possible outputs of a program	2	CO5
C-12	How is transition testing used in software testing? A) To identify all possible transitions between states B) To identify all possible states of a program C) To identify all possible inputs to a program D) To identify all possible outputs of a program	3	CO5
C-13	What is the advantage of using transition testing in software testing? A) It reduces the number of test cases B) It increases the number of test cases C) It improves the coverage of test cases D) It reduces the complexity of test cases	4	CO5

C-14	How can transition testing be used to improve the reliability of a software system? A) By identifying all possible transitions between states B) By identifying all possible states of a program C) By identifying all possible inputs to a program D) By identifying all possible outputs of a program	5	CO5
C-15	Evaluate the effectiveness of transition testing in identifying defects in a software system. What is the primary benefit of using transition testing? A) It ensures that all possible transitions are tested B) It ensures that all possible states are tested C) It ensures that all possible inputs are tested D) It ensures that all possible outputs are tested	6	CO5
	Filling The Blank Questions		
F-1	A state graph is a graphical representation of a program's _____ and transitions.	1	CO5
F-2	The purpose of a state graph is to identify all possible _____ of a program.	2	CO5
F-3	A good state graph should have a small number of _____ and transitions.	3	CO5
F-4	A bad state graph is one that has a large number of _____ and transitions.	1	CO5
F-5	State testing involves testing all possible _____ of a program.	2	CO5
F-6	Transition testing involves testing all possible _____ between states.	3	CO5
F-7	The advantage of using state graphs is that they improve the _____ of test cases.	4	CO5
F-8	State graphs can be used to improve the _____ of a software system.	5	CO5
F-9	The effectiveness of state graphs in identifying defects can be evaluated by analyzing the _____ of the system.	6	CO5
F-10	A state graph consists of nodes that represent _____ and edges that represent transitions.	1	CO5
F-11	The nodes in a state graph are also known as _____.	2	CO5
F-12	The edges in a state graph are also known as _____.	3	CO5
F13	State graphs can be used to identify _____ in a software system.	4	CO5
F14	State graphs can be used to improve the _____ of a software system.	5	CO5
F15	The use of state graphs can be evaluated by analyzing the _____ of the system.	6	CO5
	Match The Following Questions		CO5

M-1	<table><tr><td>i. State graph</td><td>A) A graphical representation of a program's states and transitions</td></tr><tr><td>ii. State testing</td><td>B) A testing technique that involves testing all possible states of a program</td></tr><tr><td>iii. Transition testing</td><td>C) A testing technique that involves testing all possible transitions between states</td></tr><tr><td>iv. Testability</td><td>D) The ease with which a program can be tested</td></tr></table>	I	CO5
M-2	<table><tr><td>i. State testing</td><td>A) To identify all possible states of a program</td></tr><tr><td>ii. Transition testing</td><td>B) To identify all possible transitions between states</td></tr><tr><td>iii. State graphs</td><td>C) To represent a program's states and transitions graphically</td></tr><tr><td>iv. Testability</td><td>D) To improve the ease with which a program can be tested</td></tr></table>	II	CO5
M-3	<table><tr><td>i. State graphs</td><td>A) Improves the coverage of test cases</td></tr><tr><td>ii. State testing</td><td>B) Reduces the complexity of test cases</td></tr><tr><td>iii. Transition testing</td><td>C) Improves the reliability of a software system</td></tr><tr><td>iv. Testability</td><td>D) Improves the maintainability of a software system</td></tr></table>	III	CO5
M-4	<table><tr><td>i. State testing</td><td>A) Can be time-consuming to test all possible states</td></tr><tr><td>ii. Transition testing</td><td>B) Can be difficult to identify all possible transitions</td></tr><tr><td>iii. State graphs</td><td>C) Can be complex to create and maintain</td></tr><tr><td>iv. Testability</td><td>D) Can be limited by the complexity of the program</td></tr></table>	IV	CO5

M-5	<table><tr><td>i. Integration of state testing with transition testing</td><td>A) To improve the overall quality of a software system</td></tr><tr><td>ii. Use of state graphs to improve testability</td><td>B) To reduce the number of defects in a software system</td></tr><tr><td>iii. Use of testability metrics to evaluate the effectiveness of testing</td><td>C) To improve the reliability of a software system</td></tr><tr><td>iv. Use of state testing to improve the reliability of a software system</td><td>D) To improve the maintainability of a software system</td></tr></table>	V	CO5
	5 MARKS QUESTIONS		
-D-1	Write short notes on a. Transition Bugs b. Dead States c. State Bugs d. Encoding Bugs	2	CO5
-D-2	What are the principles of state testing? Discuss advantages and disadvantages?	2	CO5
-D-3	Write the design guidelines for building finite state machine into code?	2	CO5
-D-4	Discuss about good and bad state graphs?	2	CO5
-D-5	What are the software implementation issues in state testing?	2	CO5
-D-6	Explain state testing in software testing?	1	CO5
-D-7	Define State Graph, Dead states and Equivalence states and Explain?	1	CO5
-D-8	Explain State transition table in detail?	3	CO5
-D-9	Explain impact of bugs in state testing?	3	CO5
-D-10	Explain switches, tags and Unachievable paths?	1	CO5
-D-11	What is Essential and Inessential Finite-State Behavior?	2	CO5
-D-12	Explain in detail Limitations and Extensions in State Testing?	2	CO5
-D-13	Distinguish between Time versus Sequence.	3	CO5
-D-14	Explain Inputs and Transitions in State Graphs?	2	CO5
-D-15	Explain State, state graphs and Transition Testing?	1	CO5
	UNIT V		
	Multiple Choice Questions		**Course Outcome**

C-1	What is a graph matrix in software testing? A) A matrix representation of a graph B) A graphical representation of a matrix C) A table representation of a graph D) A tree representation of a matrix	1	CO6
C-2	What is the purpose of a graph matrix in software testing? A) To represent the structure of a graph B) To represent the functionality of a program C) To represent the control flow of a program D) To represent the data flow of a program	2	CO6
C-3	How is a graph matrix used in software testing? A) To identify all possible paths through a graph B) To identify all possible cycles in a graph C) To identify all possible nodes in a graph D) To identify all possible edges in a graph	3	CO6
C-4	What is the power of a matrix in graph theory? A) The number of times a matrix is multiplied by itself B) The number of rows in a matrix C) The number of columns in a matrix D) The number of elements in a matrix	1	CO6
C-5	What is the node reduction algorithm in graph theory? A) An algorithm used to reduce the number of nodes in a graph B) An algorithm used to increase the number of nodes in a graph C) An algorithm used to identify all possible paths through a graph D) An algorithm used to identify all possible cycles in a graph	2	CO6
C-6	How is the node reduction algorithm used in software testing? A) To simplify the structure of a graph B) To complicate the structure of a graph C) To identify all possible paths through a graph D) To identify all possible cycles in a graph	3	CO6
C-7	What is the advantage of using graph matrices in software testing? A) It reduces the complexity of testing B) It increases the complexity of testing C) It improves the coverage of testing D) It reduces the coverage of testing	4	CO6
C-8	How can graph matrices be used to improve the reliability of a software system? A) By identifying all possible paths through a graph B) By identifying all possible cycles in a graph C) By simplifying the structure of a graph D) By complicating the structure of a graph	5	CO6
C-9	Evaluate the effectiveness of graph matrices in identifying defects in a software system. What is the primary benefit of using graph matrices? A) It ensures that all possible paths are tested B) It ensures that all possible cycles are tested C) It simplifies the structure of a graph D) It complicates the structure of a graph	6	CO6
C-10	What is a relation in graph theory? A) A connection between two nodes in a graph B) A connection between two edges in a graph C) A connection between two graphs D) A connection between two matrices	1	CO6
C-11	What is the purpose of a relation in graph theory? A) To represent the structure of a graph B) To represent the functionality of a program	2	CO5

	C) To represent the control flow of a program D) To represent the data flow of a program		
C-12	How is a relation used in software testing? A) To identify all possible paths through a graph B) To identify all possible cycles in a graph C) To simplify the structure of a graph D) To complicate the structure of a graph	3	CO6
C-13	What is the advantage of using relations in software testing? A) It reduces the complexity of testing B) It increases the complexity of testing C) It improves the coverage of testing D) It reduces the coverage of testing	4	CO6
C-14	How can relations be used to improve the reliability of a software system? A) By identifying all possible paths through a graph B) By identifying all possible cycles in a graph C) By simplifying the structure of a graph D) By complicating the structure of a graph	5	CO6
C-15	Evaluate the effectiveness of relations in identifying defects in a software system. What is the primary benefit of using relations? A) It ensures that all possible paths are tested B) It ensures that all possible cycles are tested C) It simplifies the structure of a graph D) It complicates the structure of a graph	6	CO6
	Filling The Blank Questions		
F-1	A graph matrix is a _____ representation of a graph.	1	CO6
F-2	The matrix of a graph is used to represent the _____ between nodes.	2	CO6
F-3	The power of a matrix is used to calculate the _____ of a graph.	3	CO6
F-4	A relation in graph theory is a _____ between two nodes.	1	CO6
F-5	The node reduction algorithm is used to _____ the number of nodes in a graph.	2	CO6
F-6	Graph matrices are used in software testing to identify all possible _____ through a graph.	3	CO6
F-7	The use of graph matrices can improve the _____ of a software system.	4	CO6
F-8	Graph matrices can be used to simplify the _____ of a graph.	5	CO6
F-9	The effectiveness of graph matrices in identifying defects can be evaluated by analyzing the _____ of the system.	6	CO6
F-10	A graph matrix is a square _____ with rows and columns representing nodes.	1	CO6
F-11	The entries in a graph matrix represent the _____ between nodes.	2	CO6
F-12	The power of a matrix is calculated by _____ the matrix by itself.	3	CO6
F-13	The node reduction algorithm is used to reduce the _____ of a graph	4	CO6
F-14	Graph matrices can be used to improve the _____ of a software system.	5	CO6
F-15	The use of graph matrices can be evaluated by analyzing the _____ of the system.	6	CO6
	Match The Following Questions		

M-1		I	CO6

i. Graph matrix	A) A square matrix with rows and columns representing nodes
ii. Relation	B) A connection between two nodes in a graph
iii. Power of a matrix	C) The result of multiplying a matrix by itself
iv. Node reduction algorithm	D) A technique used to reduce the number of nodes in a graph

M-2		II	CO6

i. Graph matrix	A) Used to represent the structure of a graph
ii. Relation	B) Used to identify all possible paths through a graph
iii. Power of a matrix	C) Used to calculate the number of cycles in a graph
iv. Node reduction algorithm	D) Used to simplify the structure of a graph

M-3		III	CO6

i. Graph matrix	A) Improves the coverage of testing
ii. Relation	B) Reduces the complexity of testing
iii. Power of a matrix	C) Improves the reliability of a software system
iv. Node reduction algorithm	D) Simplifies the structure of a graph

M-4		IV	CO6

i. Graph matrix	A) Used to identify all possible defects in a software system
ii. Relation	B) Used to simplify the structure of a graph
iii. Power of a matrix	C) Used to calculate the number of cycles in a graph
iv. Node reduction algorithm	D) Used to represent the structure of a graph

M-5		i. Integration of graph matrices with node reduction algorithms	A) To improve the overall quality of a software system	V	CO6
		ii. Use of relations to improve the coverage of testing	B) To reduce the number of defects in a software system		
		iii. Use of power of a matrix to simplify the structure of a graph	C) To improve the reliability of a software system		
		iv. Use of graph matrices to improve the reliability of a software system	D) To simplify the structure of a graph		

5 MARKS QUESTIONS

5D-1	Define Graph? And Explain?	2	CO5
5D-2	Define Reflexive Relations? Define directed & undirected Graph?	2	CO6
5D-3	Explain in detail Node-Reduction Algorithm?	2	CO6
5D-4	What are the applications of Node-Reduction Algorithm?	3	CO6
5D-5	Discuss about Matrix representation in software in building tools?	6	CO6
5D-6	Discuss Node Reduction Optimization of building tools?	6	CO6
5D-7	Explain about the Partitioning Algorithm.	3	CO6
5D-8	What is a graph matrix, and how is it used to represent a graph?	1	CO6
5D-9	Analyze the impact of node reduction in simplifying the representation of a graph.	5	CO6
5D-10	Explain in details Relations in Graph Matrices?	3	CO6
5D-11	What is Matrix Powers and Products?	2	CO6
5D-12	How an Algorithm is partitioned? Explain?	4	CO6
5D-13	What are the principles of Powers of Matrix and Explain Loops?	3	CO6
5D-14	Explain Properties of Relations?	4	CO6
5D-15	Discuss the problem with pictorial graphs?	5	CO6