



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



NAAC
NATIONAL ASSESSMENT AND
ACCREDITATION COUNCIL



HANDOUT

III Year II Semester

DEPARTMENT OF COMPUTER SCIENCE &
ENGINEERING (AIML)

ACADEMIC YEAR 2025-2026

HANDOUT

III Year AIML-Semester-II

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

ACADEMIC YEAR 2025-26

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

HANDOUT- INDEX

S. No	Contents
1	Vision, Mission, PEOs, POs, PSOs & COs
2	Institution AcademicCalendar
3	Department AcademicCalendar
4	Subject wise
i)	Syllabus Copy
ii)	Lesson Plan
iii)	Question Bank
iv)	EndExaminationQuestions(Previous3 Academic Year)
v)	Mid-1 & Mid-2 Questions (Previous 3 Academic Year)



SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY

B. TECH – DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

1. Institution Vision/ Mission

VISION OF THE INSTITUTE

To be a premier institution in engineering & technology and management for competency, values and social consciousness.

MISSION OF THE INSTITUTE

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

IM2: Promote continuous industry-institute interaction aimed at promoting employability, entrepreneurship, leadership and research aptitude among stakeholders.

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

DEPARTMENT VISION

To develop competent engineers in the domain of Artificial Intelligence & Machine Learning for note worthy contributions to the society.

DEPARTMENT MISSION

DM1: To strengthen academic collaborations for better exposure.

DM2: Promote professional environment to imbibe ethical values and entrepreneurial skills

DM3: Encourage research & development by emphasizing on innovation

DM4: Facilitate Industry-Institute collaboration for mutual benefit

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

PEO1: Generate contributions towards advancements in Artificial Intelligence and Machine Learning

PEO2: Promote design, research and implementation of products through strong communication skills, leadership and entrepreneurial skills.

PEO3: Apply basic principles and practices of AIML to successfully complete software related projects to meet customer business objectives and/or productively engage in research.

PROGRAM OUTCOMES (POS)

PO1: 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.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: 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)

PO4: 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).

PO5: 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)

PO6: 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).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: 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

PO10: 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.

PO11: 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)

COs MAPPING WITH POs & PSOs

SOFTWARE TESTING METHODOLOGIES (R22CSE3244)

C321.1.	Summarize a range of different software testing techniques and strategies for testing projects. (Understand)
C321.2.	List the characteristics of Dataflow, transaction flow testing and Domain testing methods. (Remember)
C321.3.	Categorize path products, expressions. (Analyze).
C321.4.	Develop and apply testing strategies for software applications (Create).
C321.5.	Evaluate various test cases for control flow and transaction flow graphs (Evaluate).
C321.6.	Illustrate appropriate software testing tools and techniques (Apply)

Course Articulation Matrix

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
C322.1	3	3	-	-	-	-	-	-	-	2	3	2	3	-
C322.2	3	3	-	2	-	-	-	-	-	2	2	1	3	-
C322.3	3	2	2	3	3	-	-	2	-	3	3	3	3	-
C322.4	3	3	-	-	-	-	-	-	-	2	2	1	2	-
C322.5	3	3	-	2	-	-	-	-	-	2	3	2	3	-
C322.6	3	3	3	2	3	-	-	-	-	3	3	3	3	-
	3	2.8	2.5	2.2	3			2		2.3	2.7	2	2.8	-



SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY
(An Autonomous Institution under UGC, New Delhi)
Recognized under 2(f) and 12(B) of UGC Act 1956
NBA & NAAC Accredited, Approved by AICTE and Permanently affiliated to JNTUH
Sheriguda (V), Ibrahimpatnam, R.R. Dist, Hyderabad - 501 510

D4

BR-22

ANNEXURE -II

Lr.No.SICET/AUTO/DAE/III B.Tech Academic Calendar/29/2025

B.TECH III-YEAR I-SEM & II-SEM ACADEMIC CALENDAR
(FOR ACADEMIC YEAR : 2025-26)

Academic Calendar for B.Tech – IIIrd Year Students (2023 - 24 Batch), BR-22 Regulation.

I – Semester

S.No.	EVENT	PERIOD	DURATION
1.	Commencement of class work.	14.07.2025 (Monday)	
2.	1 st Spell of Instructions for covering First Two and a half Units.	14.07.2025 – 13.09.2025	9 Weeks
3.	CRT Classes	15.09.2025 – 07.10.2025	2 Weeks 2 Days
4.	I Mid Term Examinations.	08.10.2025 – 10.10.2025	3 Days
5.	Submission of I Mid Term Examination Marks.	16.10.2025	
6.	2 nd Spell of Instructions for Remaining Two and a half Units.	13.10.2025 – 13.12.2025	9 Weeks
7.	II Mid Term Examinations.	15.12.2025 – 17.12.2025	3 Days
8.	Preparation Holidays and Practical Examinations	18.12.2025 – 27.12.2025	1 Week
9.	Submission of II Mid Term Examination Marks.	23.12.2025	
10.	I Semester End Examinations.	29.12.2025 – 10.01.2026	2 Weeks
Commencement of Class-Work for III B.Tech - II Semester 19.01.2026			

Note: No of Working/instructional days : 90

II – Semester

S.No.	EVENT	PERIOD	DURATION
1.	Commencement of class work.	19.01.2026 (Monday)	
2.	1 st Spell of Instructions for covering First Two and a half Units.	19.01.2026 – 14.03.2026	8 Weeks
3.	I Mid Term Examinations.	16.03.2026 – 18.03.2026	3 Days
4.	Submission of I Mid Term Examination Marks.	24.03.2026	
5.	2 nd Spell of Instructions for Remaining Two and a half Units.	20.03.2026 – 09.05.2026	7 Weeks
6.	Continuation of remaining 2 nd Spell of Instructions	25.05.2026 – 10.06.2026	1 Week
7.	II Mid Term Examinations.	11.06.2026 – 13.06.2026	3 Days
8.	Preparation Holidays and Project Evaluation.	15.06.2026 – 20.06.2026	1 Week
9.	Submission of II Mid Term Examination Marks.	18.06.2026	
10.	II Semester End Examinations	22.06.2026 – 04.07.2026	2 Weeks

Note: No of Working/instructional days : 93

ACE

CE

DIRECTOR

PRINCIPAL

Controller of Examination
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510

DIRECTOR
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510.

PRINCIPAL
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510

Sign:

Dr. M.V.S.S. Giridhar

Prof. of CEA; JNTUH Nominee

Sign:

Dr. T. Venu Gopal

Prof. of CSE; JNTUH Nominee

Sign:

Dr. D. Ramesh

Prof. of CSE; JNTUH Nominee

(Copy to DAP&E and Copy to all the Dept. Heads)

SUNDAY		Dec-25			1		1								Jun-26		
MONDAY	1				2		2							1			
TUESDAY	2				3	II YR CRT TRAINING	3	II YR CRT TRAINING		Apr-26				2	II YR TRAINING		Jul-26
WEDNESDAY	3			Jan-26	4		4	HOLI HOLIDAY	1					3		1	II-II & III-II SEM END EXAMS
THURSDAY	4		1	NEW YEAR HOLIDAY	5		5		2	II YR TRAINING		May-26	4		2	II-II & III-II SEM END EXAMS	
FRIDAY	5	AI CLUB MEETING	2		6	II YR CRT TRAINING	6	II YR CRT TRAINING	3	GOOD FRIDAY HOLIDAY	1	IV-II MID II EXAMS	5	II YR TRAINING	3	II-II & III-II SEM END EXAMS	
SATURDAY	6	DEPARTMENT MEETING	3		7	AI CLUB MEETING	7		4	AI CLUB MEETING	2	IV-II MID II EXAMS	6		4	II-II & III-II SEM END EXAMS	
SUNDAY	7	HOLIDAY	4	HOLIDAY	8	HOLIDAY	8	HOLIDAY	5	HOLIDAY	3	HOLIDAY	7	HOLIDAY	5	HOLIDAY	
MONDAY	8		5		9		9		6		4	IV-II PREPARATION HOLIDAYS	8		6		
TUESDAY	9		6		10	II YR CRT TRAINING	10	II YR CRT TRAINING	7	II YR TRAINING	5	IV-II PREPARATION HOLIDAYS	9		7		
WEDNESDAY	10		7		11		11		8		6	IV-II PREPARATION HOLIDAYS	10		8		
THURSDAY	11		8		12		12		9		7	SUBMISSION OF MID-II MARKS	11	II-II&III-II MID II EXAM	9		
FRIDAY	12		9		13	II YR CRT TRAINING	13	II YR CRT TRAINING	10	II YR TRAINING	8	IV-II PREPARATION HOLIDAYS	12	II-II&III-II MID II EXAM	10		
SATURDAY	13		10	DEPARTMENT MEETING	14		14		11	DEPARTMENT MEETING	9	IV-II PREPARATION HOLIDAYS	13	II-II&III-II MID II EXAM	11	DEPARTMENT MEETING	
SUNDAY	14	HOLIDAY	11	HOLIDAY	15	HOLIDAY	15	HOLIDAY	12	HOLIDAY	10	HOLIDAY	14	HOLIDAY	12	HOLIDAY	
MONDAY	15	IV-II COMMENCEMENT WORK	12		16	IV-II MID I EXAM	16	II & III-II MID I EXAM	13		11	IV-II SEM EXAMS	15	II-II&III-II PREP HOILDAYS & PROJECT EVALUATION	13		
TUESDAY	16		13	PONGAL HOLIDAYS	17	IV-II MID I EXAM& II YR TRAINING	17	II & III-II MID I EXAM,II YR TRAINING	14	DR.AMBEDKAR JAYANTI HOLIDAY	12	IV-II SEM EXAMS	16	II-II&III-II PREP HOILDAYS & PROJECT EVALUATION	14		
WEDNESDAY	17		14	PONGAL HOLIDAYS	18	IV-II MID I EXAM	18	II & III-II MID I EXAM	15		13	IV-II SEM EXAMS	17	II-II&III-II PREP HOILDAYS & PROJECT EVALUATION	15		
THURSDAY	18		15	PONGAL HOLIDAYS	19	IV-II 2ND SPELL	19		16		14	IV-II SEM EXAMS	18	II-II&III-II PREP HOILDAYS & PROJECT EVALUATION	16		
FRIDAY	19		16	PONGAL HOLIDAYS	20	II YR CRT TRAINING	20	II -II & III-II/ II SPELL INSTRUCTIONS	17	II YR TRAINING	15	IV-II SEM EXAMS,	19	II-II&III-II PREP HOILDAYS & PROJECT EVALUATION	17		
SATURDAY	20		17	PONGAL HOLIDAYS	21		21		18		16	IV-II SEM EXAMS	20	II-II&III-II PREP HOILDAYS & PROJECT EVALUATION	18		
SUNDAY	21	HOLIDAY	18	HOLIDAY	22	HOLIDAY	22	HOLIDAY	19	HOLIDAY	17	HOLIDAY	21	HOLIDAY	19	HOLIDAY	

MONDAY	22		19	II & III-II COMMENCEMENT WORK	23		23		20		18		22	II-II & III-II SEM END EXAMS	20	
TUESDAY	23		20		24	SUBMISSION OF IV-II MID I MARKS, II YR TRAINING	24	SUBMISSION OF III-II MID I MARKS	21	II YR TRAINING	19	II YR TRAINING	23	II-II & III-II SEM END EXAMS	21	

WEDNESDAY	24		21		25		25		22		20		24	II-II & III-II SEM END EXAMS	22	
THURSDAY	25	CHRISTMAS HOLIDAY	22		26		26		23		21		25	II-II & III-II SEM END EXAMS	23	
FRIDAY	26	BOXING DAY HOLIDAY	23	AI CLUB MEETING	27	II YR CRT TRAINING	27	II YR TRAINING	24	II YR TRAINING	22	II YR TRAINING	26	MUHARRAM HOLIDAY	24	
SATURDAY	27	DEPARTMENT MEETING	24		28	DEPARTMENT MEETING	28		25		23		27	II-II & III-II SEM END EXAMS	25	
SUNDAY	28	HOLIDAY	25	HOLIDAY		HOLIDAY	29	HOLIDAY	26	HOLIDAY	24	HOLIDAY	28	HOLIDAY	26	HOLIDAY
MONDAY	29		26	REPUBLIC DAY			30		27		25		29	II-II & III-II SEM END EXAMS	27	
TUESDAY	30		27	II & III YR CRT TRAINING			31	II YR TRAINING	28	II YR TRAINING	26	II YR TRAINING	30	II-II & III-II SEM END EXAMS	28	
WEDNESDAY	31	AI CLUB MEETING	28	II & III YR CRT TRAINING					29	AI CLUB MEETING	27	BAKRID HOLIDAY			29	
THURSDAY			29	II & III YR CRT TRAINING					30	IV-II MID II EXAMS	28				30	
FRIDAY			30	II & III YR CRT TRAINING							29	II YR TRAINING			31	
SATURDAY			31	II & III YR CRT TRAINING							30					
SUNDAY		HOLIDAY		HOLIDAY		HOLIDAY		HOLIDAY		HOLIDAY	31	HOLIDAY		HOLIDAY		HOLIDAY



Lr.No.SICET/AUTO/DAE/III B.Tech Academic Calendar/85/2024

B.TECH III-YEAR I-SEM & II-SEM ACADEMIC CALENDAR
(FOR ACADEMIC YEAR : 2024-25)

Academic Calendar for B.Tech – IIIrd Year Students (2022 - 23 Batch), BR-22 Regulation.

I – Semester

S.No.	EVENT	PERIOD	DURATION
1.	Commencement of class work.	22.07.2024 (Monday)	
2.	1 st Spell of Instructions for covering First Two and a half Units.(Including CRT training Classes)	22.07.2024 – 28.09.2024	10 Weeks
3.	I Mid Term Examinations.	30.09.2024 – 05.10.2024	5 Days
4.	Submission of I Mid Term Examination Marks.	09.10.2024	
5.	2 nd Spell of Instructions for Remaining Two and a half Units.	14.10.2024 – 14.12.2024	9 Weeks
6.	II Mid Term Examinations.	16.12.2024 – 18.12.2024	3 Days
7.	Preparation Holidays and Practical Examinations	19.12.2024 – 28.12.2024	1 Week 3 Days
8.	Submission of II Mid Term Examination Marks.	27.12.2024	
9.	I Semester End Examinations.	30.12.2024 – 18.01.2025	3 Weeks
Commencement of Class-Work for III B.Tech - II Semester 20.01.2025 (Monday)			

Note: No of Working/instructional days : 99

II – Semester

S.No.	EVENT	PERIOD	DURATION
1.	Commencement of class work.	20.01.2025 (Monday)	
2.	1 st Spell of Instructions for covering First Two and a half Units.	20.01.2025 – 22.03.2025	9 Weeks
3.	I Mid Term Examinations.	24.03.2025 – 26.03.2025	3 Days
4.	Submission of I Mid Term Examination Marks.	02.04.2025	
5.	2 nd Spell of Instructions for Remaining Two and a half Units.	27.03.2025 – 10.05.2025	6 Weeks 3 Days
6.	Summer Vacation	12.05.2025 – 24.05.2025	2 Weeks
7.	Continuation of 2 nd Spell of Instructions for Remaining Two and a half Units.	26.05.2025 – 11.06.2025	2 Weeks 3 Days
8.	II Mid Term Examinations.	12.06.2025 – 14.06.2025	3 Days
9.	Preparation Holidays and Practical Examinations	16.06.2025 – 21.06.2025	1 Week
10.	Submission of II Mid Term Examination Marks.	19.06.2025	
11.	II Semester End Examinations	23.06.2025 – 05.07.2025	2 Weeks
12.	Commencement of Class-Work for IV B.Tech - I Semester	07.07.2025(Monday)	

Note: No of Working/instructional days : 98

ACE

Controller of Examination
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510.

DIRECTOR
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510.

PRINCIPAL
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510.

Sign:
Dr. M.V.S.S. Giridhar
Prof. of CEA; JNTUH Nominee

Sign:
Dr. T. Venu Gopal
Prof. of CSE; JNTUH Nominee

Sign:
Dr. D. Ramesh
Prof. of CSE; JNTUH Nominee

(Copy to DAP&E and Copy to all the Dept. Heads)



Lr.No.SICET/AUTO/DAE/III B.Tech Academic Calendar/85/2024

B.TECH III-YEAR I-SEM & II-SEM ACADEMIC CALENDAR
(FOR ACADEMIC YEAR : 2024-25)

Academic Calendar for B.Tech – IIIrd Year Students (2022 - 23 Batch), BR-22 Regulation.

I – Semester

S.No.	EVENT	PERIOD	DURATION
1.	Commencement of class work.	22.07.2024 (Monday)	
2.	1 st Spell of Instructions for covering First Two and a half Units.(Including CRT training Classes)	22.07.2024 – 28.09.2024	10 Weeks
3.	1 Mid Term Examinations.	30.09.2024 – 05.10.2024	5 Days
4.	Submission of 1 Mid Term Examination Marks.	09.10.2024	
5.	2 nd Spell of Instructions for Remaining Two and a half Units.	14.10.2024 – 14.12.2024	9 Weeks
6.	1 Mid Term Examinations.	16.12.2024 – 18.12.2024	3 Days
7.	Preparation Holidays and Practical Examinations	19.12.2024 – 28.12.2024	1 Week 3 Days
8.	Submission of 1 Mid Term Examination Marks.	27.12.2024	
9.	1 Semester End Examinations.	30.12.2024 – 18.01.2025	3 Weeks
Commencement of Class-Work for III B.Tech - II Semester 20.01.2025 (Monday)			

Note: No of Working/Instructional days : 99

II – Semester

S.No.	EVENT	PERIOD	DURATION
1.	Commencement of class work.	20.01.2025 (Monday)	
2.	1 st Spell of Instructions for covering First Two and a half Units.	20.01.2025 – 22.03.2025	9 Weeks
3.	1 Mid Term Examinations.	24.03.2025 – 26.03.2025	3 Days
4.	Submission of 1 Mid Term Examination Marks.	02.04.2025	
5.	2 nd Spell of Instructions for Remaining Two and a half Units.	27.03.2025 – 10.05.2025	6 Weeks 3 Days
6.	Summer Vacation	12.05.2025 – 24.05.2025	2 Weeks
7.	Continuation of 2 nd Spell of Instructions for Remaining Two and a half Units.	26.05.2025 – 11.06.2025	2 Weeks 3 Days
8.	1 Mid Term Examinations.	12.06.2025 – 14.06.2025	3 Days
9.	Preparation Holidays and Practical Examinations	16.06.2025 – 21.06.2025	1 Week
10.	Submission of 1 Mid Term Examination Marks.	19.06.2025	
11.	1 Semester End Examinations	23.06.2025 – 05.07.2025	2 Weeks
12.	Commencement of Class-Work for IV B.Tech - I Semester	07.07.2025(Monday)	

Note: No of Working/Instructional days : 98

ACE

Controller of Examination
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510

DIRECTOR
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510

PRINCIPAL
Sri Indu College of Engineering & Technology
(An Autonomous Institution Under UGC)
Sheriguda (V), Ibrahimpatnam, R.R. Dist-501510

Sign:

Dr. M.V.S.S. Giridhar
Prof. of CEA; JNTUJ Nominee

Sign:

Dr. T. Venu Gopal
Prof. of CSE; JNTUJ Nominee

Sign:

Dr. D. Ramesh
Prof. of CSE; JNTUJ Nominee

(Copy to DAP&E and Copy to all the Dept. Heads)

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 – II

(R22CSE3244) SOFTWARE TESTING METHODOLOGIES

Course Objectives

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using latest tools.

Course Outcomes: Design and develop the best test strategies in accordance to the development model.

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

UNIT - III

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 - IV

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

UNIT - V

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 Winrunner).

TEXT BOOKS:

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.

REFERENCE BOOKS:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

FACULTY INDIVIDUAL TIME TABLE

Name of the Faculty: Dr.Karunaiah Bonigala

AY: 2025-2026

Designation: Assoc. Professor

Year Semester: III/II

Time	9:40 – 10:40	10:40 – 11:40	11:40 – 12:40	12:40 To 1:20	1:20 – 2:15	2:15 – 3:10	3:10 – 4:00
Days	1	2	3		4	5	6
Monday		STM-III-II (AIML-A)		L U N C H		STM-III-II (AIML-A)	
Tuesday	STM-III-II (AIML-A)		STM-III-II (AIML-B)				STM-III-II (AIML-B)
Wednesday							
Thursday			STM-III-II (AIML-A)				
Friday	STM-III-II (AIML-A)						STM-III-II (AIML-B)
Saturday		STM-III-II (AIML-B)					

S.NO	SUBJECT CODE	SUBJECT	CLASS	NO OF HOURS IN A WEEK	ADDITIONAL WORK LOAD
1	STM(R22CSE3244)	Software Testing Methodologies	III (AIML A)	5	
2	STM(R22CSE3244)	Software Testing Methodologies	III (AIML B)	5	
	TOTAL HOURS IN A WEEK			10	

Faculty Signature

DEAN

Head of the Department

	SRI INDU COLLEGE OF ENGG & TECHNOLOGY		
	LESSON PLAN		
	(Regulation:R22)		
	Department of AIML		
Sub. Code & Title	R22CSE3244&SOFTWARE TESTING METHODOLOGIES		
AcademicYear:2025-2026	Year/Sem./Section	III-II	
Faculty Name & Designation	Dr.Karunaiah Bonigala, Assoc. Professor		

Unit / Item No.	Topic (s)	Book Reference	Page (s)		Teaching Methodology	Proposed No. of Periods	Proposed Date of Handling	CO/RBT
			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	Blackboard	01		CO-1, L2
1.2	Goals of testing, Phases in Tester’s Mental life	T-1	3	6	PPT	01		CO-1,L2
1.3	Test Design,Testing is Not Everything	T-1	7	9	PPT	01		CO-1,L2
1.4	Dichotomies	T-1	9	15	Blackboard	02		CO-1,L1
1.5	A Model for testing	T-1	15	20	Blackboard	01		CO-1,L2
1.6	Tests, Role of models	T-1	20	27	Blackboard	01		CO-1,L2
1.7	Consequences of bugs	T-1	27	28	PPT	01		CO-1,L2
1.8	How Bugs Affects us	T-1	28	33	PPT	01		CO-1,L2
1.9	Taxonomy of bugs	T-1	33	57	Blackboard	02		CO-1,L2
1.10	Basic concepts of path testing	T-1	59	90	PPT	02		CO-1,L2
1.11	Predicates, Path predicates and achievable paths	T-1	92	99	PPT	01		CO-1,L2
1.12	Path sensitizing, Path instrumentation	T-1	101	112	PPT	01		CO-1,L2
1.13	Application of path testing	T-1	115	117	PPT	01		CO-1,L2
II	TRANSACTION FLOW TESTING, DATAFLOW TESTING & DOMAIN					18		

	TESTING						
2.1	Transaction flow testing	T-1,R-1	122	131	PPT	02	CO-2,L3
2.2	Transaction flow testing Techniques	T-1,R-1	133	139	PPT	02	CO-2,L3
2.3	Dataflow testing: Basics of dataflow testing	T-1,R-1	145	157	Blackboard	02	CO-2,L3
2.4	Strategies in dataflow Testing	T-1,R-1	161	167	Blackboard	01	CO-2,L3
2.5	Applications of dataflow Testing	T-1	168	171	Blackboard	01	CO-2
2.6	Domains and paths	T-1,R-1	173	179	PPT	02	CO-3,L4
2.7	Nice & ugly domains	T-1	182	188	Blackboard	02	CO-3,L4
2.8	Domain testing	T-1	192	201	PPT	01	CO-3,L4
2.9	Domains and interfaces testing	T-1,R-1	202	206	PPT	02	CO-3,L4
2.10	Domains and testability	T-1,R-1	207	210	PPT	02	CO-3,L4
2.11	Revision of domain Testing	T-1			PPT	01	CO-3,L4

	Review	Signature of the HOD/Coordinator	
--	---------------	---	--

UNIT-III

III	PATHS, PATH PRODUCTS AND REGULAR EXPRESSIONS					14		
3.1	Path products & path Expressions	T-1,R-1	244	250	PPT	02		CO-4,L2
3.2	Reduction procedure	T-1,R-1	251	256	PPT	02		CO-4,L2
3.3	Applications	T-1,R-2	257	277	Blackboard	01		CO-4,L2
3.4	Regular expressions	T-1,R-2	278	280	PPT	01		CO-4,L2
3.5	Flow anomaly detection	T-1	281	282	Blackboard	01		CO-4,L2
3.6	Logic based testingOverview	T-1	320	322	Blackboard	01		CO-4,L2
3.7	Decision tables	T-1	322	329	Blackboard	02		CO-4,L2
3.8	Path expressions	T-1	332	341	Blackboard	02		CO-4,L2
3.9	KV charts	T-1	343	352	PPT	01		CO-4,L2
3.10	Specifications	T-1	352	357	PPT	01		CO-4,L2
	Review	Signature of the HOD/Coordinator						

UNIT-IV

IV	STATES,STATE GRAPHS,AND TRANSITION TESTING						12		
5.1	State graphs	T-1	363	369	Blackboard	03		CO-5,L2	
5.2	Good & Bad state graphs	T-1	373	386	PPT	03		CO-5,L2	
5.3	State testing	T-1	387	390	PPT	3		CO-5,	
5.4	Testability tips	T-1	391	394	Blackboard	3		CO-5,	
	Review	Signature of the HOD/Coordinator							
	UNIT-V								
V	GRAPH MATRICES AND APPLICATIONS						12		
5.5	Graph Matrices overview	T-1	397	398	PPT	3		CO-5,	
5.6	Matrix of graph, relations, power of a matrix	T-1	399	414	PPT	3		CO-5,	
5.7	Node reduction algorithm	T-1	415	420	PPT	3		CO-5,	
5.8	Building tools	T-1	421	426	PPT	3		CO-5,	
	Review	Signature of the HOD/Coordinator							

	SRI INDU COLLEGE OF ENGG & TECHNOLOGY LESSON PLAN (Regulation:R22) Department of Computer Science And Information Technology		
	Sub. Code &Title	R22CSE3244&SOFTWARE TESTING METHODOLOGIES	
	AcademicYear:2025-2026	Year/Sem./Section	III-II
	Faculty Name & Designation	Dr.Karunaiah Bonigala, Assoc. Professor	

LIST OF TEXT BOOKS AND REFERENCES

TEXT BOOKS:

- T1. Software Testing techniques – Boris Beizer, Dreamtech, second edition.
- T2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

REFERENCE BOOKS:

- R1. The craft of software testing – Brian Marick, Pearson Education.
- R2. Software Testing Techniques – SPD(Oreille)
- R3. Software Testing in the Real World – Edward Kit, Pearson.
- R4. Effective methods of Software Testing, Perry, John Wiley.
- R5. Art of Software Testing – Meyers, John Wiley.

Web links

- W1. <https://www.guru99.com/testing-methodology.html>
- W4. <https://www.youtube.com/watch?v=aFyK1zLFmXQ>
- W5. <https://www.youtube.com/watch?v=2xCMemD3Jyk&t=20s>
- W6. <http://elearningmodules4engg.blogspot.com/2015/06/k-v-charts-unit-6.html>

QUESTION BANK

	SRI INDU COLLEGE OF ENGG & TECHNOLOGY QUESTION BANK (Regulation :R22) Department of AIML		
	Sub. Code & Title	R22CSE3244 & SOFTWARE TESTING METHODOLOGIES	
	Academic Year: 2025-26	Year/Sem./Section	III B.TECH II-SEM
	Faculty Names& Designation	Dr.Karunaiah Bonigala , Assoc.Prof.(AI&ML)	

UNIT-I

Multiple Choice Questions		BT Level	Course Outcome
1C-1	What is the primary purpose of software testing? A) To debug code B) To ensure reliability C) To improve performance D) To reduce costs	1	CO1
1C-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
1C-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
1C-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
1C-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
1C-6	What are flow graphs used for in software testing?	2	CO1

	A) To model program flow B) To identify defects C) To prioritize testing D) To optimize performance		
1C-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
1C-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
1C-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
1C-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
1C-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
1C-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

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

1F -13	Path testing can be used to identify _____ in a program.	3	CO1
1F -14	The use of path testing can help to _____ the number of defects in a software system.	4	CO1
1F -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			
1M-1	i. Predicate	A) A condition that affects program flow	a)
	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	I
	iv. Path sensitizing	D) A technique used to identify conditions that affect program flow	
1M-2	i. Path testing	A) A technique that involves analyzing the flow of a program	b)
	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	II
	iv. Path sensitizing	D) A technique used to identify conditions that affect program flow	
1M-3	i. Path instrumentation	A) Used to monitor and control program execution	
	ii. Path sensitizing	B) Used to identify conditions that affect program flow	
	iii. Achievable paths	C) Used to identify paths that can be executed with valid input	III
	iv. Path predicates	D) Used to determine which path to take	
1M-4	i. Improved code coverage	A) Path testing	d)
	ii. Increased defect detection	B) Flow graph testing	
	iii. Reduced testing time	C) Path instrumentation	IV
	iv. Improved reliability	D) Path sensitizing	
1M-5	i. Integration of path testing with other testing techniques	A) To improve the overall quality of a software system	e)
	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 a software system	V
	iv. Use of path testing to reduce testing time	D) To reduce the time and cost of testing	

5 MARKS QUESTIONS			
1D-1	What is the primary purpose of testing in software development?	1	CO1
1D-2	Explain the concept of dichotomies in software testing.	2	CO1
1D-3	What is the consequence of a bug in a software system?	1	CO1
1D-4	Explain the concept of taxonomy of bugs in software testing.	2	CO1
1D-5	What is a predicate in the context of path testing?	1	CO1
1D-6	Explain the difference between a path predicate and an achievable path.	2	CO1
1D-7	Describe the process of path sensitizing in path testing.	3	CO1
1D-8	Describe the application of path testing in software development.	3	CO1
1D-9	Evaluate the effectiveness of path testing in identifying bugs in a software system.	6	CO1
1D-10	Design a simple path testing strategy for a given program.	5	CO1
1D-11	Compare and contrast different types of bugs that can occur in a software system.	4	CO1
1D-12	What is a model in the context of software testing?	1	CO1
1D-13	What is White-Box Testing? What are the advantages and disadvantages of white-box testing?	1	CO1
1D-14	What is Black-Box Testing? What are the advantages and disadvantages of Black-box testing?	1	CO1
1D-15	Define Software Testing and write down the different levels in software testing	1	CO1
UNIT-II			
Multiple Choice Questions		BT Level	Course Outcome
2C-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
2C-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
2C-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	1	CO2

	C) A testing technique that analyzes the control flow of a system D) A testing technique that analyzes the data flow of a system		
2C-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
2C-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
2C-7.	What is the relationship between domains and paths? A) Domains define the possible inputs to a system, while paths define the 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	4	CO2
2C-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
2C-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
2C-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
2C-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
2C-12.	How is dataflow testing used to identify defects in a program? A) By analyzing the flow of data through a program	3	CO2

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

2F-13	The application of dataflow testing can be used to improve the _____ of a system.	4	CO2								
2F-14	Dataflow testing can be used to improve the _____ of a system	5	CO2								
2F-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										
2M-1	<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>	i. Transaction flow testing	A) A testing technique that analyzes the flow of transactions through a system	ii. Dataflow testing	B) A testing technique that analyzes the flow of data through a program	iii. Domain testing	C) A testing technique that analyzes the domains of a system	iv. Interface testing	D) A testing technique that analyzes the interfaces of a system	i	CO2
i. Transaction flow testing	A) A testing technique that analyzes the flow of transactions through a system										
ii. Dataflow testing	B) A testing technique that analyzes the flow of data through a program										
iii. Domain testing	C) A testing technique that analyzes the domains of a system										
iv. Interface testing	D) A testing technique that analyzes the interfaces of a system										

2M-2	<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>	i. Transaction flow testing	A) To ensure that all transactions are processed correctly	ii. Dataflow testing	B) To ensure that all data flows are correct	iii. Domain testing	C) To ensure that all domains are correctly defined	iv. Interface testing	D) To ensure that all interfaces are working correctly	ii	CO2
i. Transaction flow testing	A) To ensure that all transactions are processed correctly										
ii. Dataflow testing	B) To ensure that all data flows are correct										
iii. Domain testing	C) To ensure that all domains are correctly defined										
iv. Interface testing	D) To ensure that all interfaces are working correctly										
2M-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>	i. Nice domains	A) Used to test valid inputs	ii. Ugly domains	B) Used to test invalid inputs	iii. Dataflow testing	C) Used to analyze the flow of data through a program	iv. Domain testing	D) Used to analyze the domains of a system	iii	CO2
i. Nice domains	A) Used to test valid inputs										
ii. Ugly domains	B) Used to test invalid inputs										
iii. Dataflow testing	C) Used to analyze the flow of data through a program										
iv. Domain testing	D) Used to analyze the domains of a system										
2M-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>	i. Improved reliability	A) Transaction flow testing	ii. Increased testability	B) Dataflow testing	iii. Reduced defects	C) Domain testing	iv. Improved performance	D) Interface testing	iv	CO2
i. Improved reliability	A) Transaction flow testing										
ii. Increased testability	B) Dataflow testing										
iii. Reduced defects	C) Domain testing										
iv. Improved performance	D) Interface testing										

2M-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>	i. Integration of transaction flow testing with dataflow testing	A) To improve the overall quality of a system	ii. Use of domain testing to improve testability	B) To reduce the number of defects in a system	iii. Use of dataflow testing to analyze the flow of data	C) To improve the reliability of a system	iv. Use of interface testing to ensure correct interfaces	D) To improve the testability of a system	v	CO2
	i. Integration of transaction flow testing with dataflow testing	A) To improve the overall quality of a system									
	ii. Use of domain testing to improve testability	B) To reduce the number of defects in a system									
	iii. Use of dataflow testing to analyze the flow of data	C) To improve the reliability of a system									
iv. Use of interface testing to ensure correct interfaces	D) To improve the testability of a system										

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
3C-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	1	CO3

	D) A representation of a flowchart		
3C-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
3C-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
3C-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
3C-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
3C-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
3C-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
3C-8	How can logic-based testing 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	CO3
3C-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
3C-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
3C-11	What is the purpose of a Karnaugh chart (K-map) in logic-based testing?	2	CO4

	A) To simplify complex logic B) To complicate simple logic C) To represent conditional statements D) To represent loops		
3C-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
3C-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
3C-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
3C-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			
3F-1	Path products are used in software testing to represent the _____ of a path through a program.	1	CO3
3F-2	The reduction procedure is used in path products to _____ the path product.	2	CO3
3F-3	Regular expressions are used in software testing to detect _____ in the flow of a program.	3	CO3
3F-4	Logic-based testing is a software testing technique that uses _____ to represent the logic of a program.	1	CO3
3F-5	Decision tables are used in logic-based testing to represent _____ logic.	2	CO3
3F-6	Path expressions are used in logic-based testing to represent the _____ of a path through a program.	3	CO3
3F-7	Karnaugh charts (K-maps) are used in logic-based testing to _____ complex logic.	4	CO3

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

3M-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>		i. Path products	A) Used to detect flow anomalies in a program	ii. Regular expressions	B) Used to represent complex logic in a program	iii. Decision tables	C) Used to analyze the flow of a program	iv. Logic-based testing	D) Used to improve the reliability of a software system	II	CO3
i. Path products	A) Used to detect flow anomalies in a program											
ii. Regular expressions	B) Used to represent complex logic in a program											
iii. Decision tables	C) Used to analyze the flow of a program											
iv. Logic-based testing	D) Used to improve the reliability of a software system											
3M-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>		i. Path expressions	A) Improves the coverage of test cases	ii. Decision tables	B) Reduces the complexity of test cases	iii. Karnaugh charts	C) Improves the reliability of a software system	iv. Regular expressions	D) Detects flow anomalies in a program	III	CO4
i. Path expressions	A) Improves the coverage of test cases											
ii. Decision tables	B) Reduces the complexity of test cases											
iii. Karnaugh charts	C) Improves the reliability of a software system											
iv. Regular expressions	D) Detects flow anomalies in a program											
3M-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>		i. Path products	A) Can be complex to implement	ii. Regular expressions	B) Can be difficult to maintain	iii. Decision tables	C) May not detect all defects	iv. Logic-based testing	D) Can be time-consuming to test	IV	CO4
i. Path products	A) Can be complex to implement											
ii. Regular expressions	B) Can be difficult to maintain											
iii. Decision tables	C) May not detect all defects											
iv. Logic-based testing	D) Can be time-consuming to test											

3M-5	i. Integration of path products with regular expressions	A) To improve the overall quality of a software system	V	CO4
	ii. Use of decision tables to improve test coverage	B) To reduce the number of defects in a software system		
	iii. Use of Karnaugh charts to simplify complex logic	C) To improve the reliability of a software system		
	iv. Use of logic-based testing to improve reliability	D) To improve the testability of a software system		

5 MARKS QUESTIONS

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

UNIT IV

Multiple Choice Questions		Course Outcome
---------------------------	--	----------------

4C-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
4C-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
4C-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
4C-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
4C-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 D) A state graph that has a small number of transitions	2	CO5
4C-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
4C-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
4C-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
4C-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
4C-10	What is transition testing in software testing? A) A testing technique that involves testing all possible transitions between states	1	CO5

	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		
4C-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
4C-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
4C-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
4C-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
4C-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			
4F-1	A state graph is a graphical representation of a program's _____ and transitions.	1	CO5

4F-2	The purpose of a state graph is to identify all possible _____ of a program.	2	CO5								
4F-3	A good state graph should have a small number of _____ and transitions.	3	CO5								
4F-4	A bad state graph is one that has a large number of _____ and transitions.	1	CO5								
4F-5	State testing involves testing all possible _____ of a program.	2	CO5								
4F-6	Transition testing involves testing all possible _____ between states.	3	CO5								
4F-7	The advantage of using state graphs is that they improve the _____ of test cases.	4	CO5								
4F-8	State graphs can be used to improve the _____ of a software system.	5	CO5								
4F-9	The effectiveness of state graphs in identifying defects can be evaluated by analyzing the _____ of the system.	6	CO5								
4F-10	A state graph consists of nodes that represent _____ and edges that represent transitions.	1	CO5								
4F-11	The nodes in a state graph are also known as _____.	2	CO5								
4F-12	The edges in a state graph are also known as _____.	3	CO5								
4F13	State graphs can be used to identify _____ in a software system.	4	CO5								
4F14	State graphs can be used to improve the _____ of a software system.	5	CO5								
4F15	The use of state graphs can be evaluated by analyzing the _____ of the system.	6	CO5								
	Match The Following Questions		CO5								
4M-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. State graph	A) A graphical representation of a program's states and transitions	ii. State testing	B) A testing technique that involves testing all possible states of a program	iii. Transition testing	C) A testing technique that involves testing all possible transitions between states	iv. Testability	D) The ease with which a program can be tested	I	CO5
i. State graph	A) A graphical representation of a program's states and transitions										
ii. State testing	B) A testing technique that involves testing all possible states of a program										
iii. Transition testing	C) A testing technique that involves testing all possible transitions between states										
iv. Testability	D) The ease with which a program can be tested										

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

4M-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>	i. Integration of state testing with transition testing	A) To improve the overall quality of a software system	ii. Use of state graphs to improve testability	B) To reduce the number of defects in a software system	iii. Use of testability metrics to evaluate the effectiveness of testing	C) To improve the reliability of a software system	iv. Use of state testing to improve the reliability of a software system	D) To improve the maintainability of a software system	V	CO5
	i. Integration of state testing with transition testing	A) To improve the overall quality of a software system									
	ii. Use of state graphs to improve testability	B) To reduce the number of defects in a software system									
	iii. Use of testability metrics to evaluate the effectiveness of testing	C) To improve the reliability of a software system									
iv. Use of state testing to improve the reliability of a software system	D) To improve the maintainability of a software system										

5 MARKS QUESTIONS

4D-1	Write short notes on a. Transition Bugs b. Dead States c. State Bugs d. Encoding Bugs	2	CO5
4D-2	What are the principles of state testing? Discuss advantages and disadvantages?	2	CO5
4D-3	Write the design guidelines for building finite state machine into code?	2	CO5
4D-4	Discuss about good and bad state graphs?	2	CO5
4D-5	What are the software implementation issues in state testing?	2	CO5
4D-6	Explain state testing in software testing?	1	CO5
4D-7	Define State Graph, Dead states and Equivalence states and Explain?	1	CO5
4D-8	Explain State transition table in detail?	3	CO5
4D-9	Explain impact of bugs in state testing?	3	CO5
4D-10	Explain switches, tags and Unachievable paths?	1	CO5
4D-11	What is Essential and Inessential Finite-State Behavior?	2	CO5
4D-12	Explain in detail Limitations and Extensions in State Testing?	2	CO5
4D-13	Distinguish between Time versus Sequence.	3	CO5
4D-14	Explain Inputs and Transitions in State Graphs?	2	CO5

4D-15	Explain State, state graphs and Transition Testing?	1	CO5
UNIT V			
Multiple Choice Questions			Course Outcome
5C-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
5C-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
5C-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
5C-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
5C-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
5C-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
5C-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
5C-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
5C-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	6	CO6

	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		
5C-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
5C-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 C) To represent the control flow of a program D) To represent the data flow of a program	2	CO5
5C-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
5C-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
5C-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
5C-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			
5F-1	A graph matrix is a _____ representation of a graph.	1	CO6
5F-2	The matrix of a graph is used to represent the _____ between nodes.	2	CO6
5F-3	The power of a matrix is used to calculate the _____ of a graph.	3	CO6
5F-4	A relation in graph theory is a _____ between two nodes.	1	CO6
5F-5	The node reduction algorithm is used to _____ the number of nodes in a graph.	2	CO6
5F-6	Graph matrices are used in software testing to identify all possible _____ through a graph.	3	CO6

5F-7	The use of graph matrices can improve the _____ of a software system.	4	CO6								
5F-8	Graph matrices can be used to simplify the _____ of a graph.	5	CO6								
5F-9	The effectiveness of graph matrices in identifying defects can be evaluated by analyzing the _____ of the system.	6	CO6								
5F-10	A graph matrix is a square _____ with rows and columns representing nodes.	1	CO6								
5F-11	The entries in a graph matrix represent the _____ between nodes.	2	CO6								
5F-12	The power of a matrix is calculated by _____ the matrix by itself.	3	CO6								
5F-13	The node reduction algorithm is used to reduce the _____ of a graph	4	CO6								
5F-14	Graph matrices can be used to improve the _____ of a software system.	5	CO6								
5F-15	The use of graph matrices can be evaluated by analyzing the _____ of the system.	6	CO6								
	Match The Following Questions										
5M-1	<table><tr><td>i. Graph matrix</td><td>A) A square matrix with rows and columns representing nodes</td></tr><tr><td>ii. Relation</td><td>B) A connection between two nodes in a graph</td></tr><tr><td>iii. Power of a matrix</td><td>C) The result of multiplying a matrix by itself</td></tr><tr><td>iv. Node reduction algorithm</td><td>D) A technique used to reduce the number of nodes in a graph</td></tr></table>	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	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										
5M-2	<table><tr><td>i. Graph matrix</td><td>A) Used to represent the structure of a graph</td></tr><tr><td>ii. Relation</td><td>B) Used to identify all possible paths through a graph</td></tr><tr><td>iii. Power of a matrix</td><td>C) Used to calculate the number of cycles in a graph</td></tr><tr><td>iv. Node reduction algorithm</td><td>D) Used to simplify the structure of a graph</td></tr></table>	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	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										

5M-3	<table><tr><td>i. Graph matrix</td><td>A) Improves the coverage of testing</td></tr><tr><td>ii. Relation</td><td>B) Reduces the complexity of testing</td></tr><tr><td>iii. Power of a matrix</td><td>C) Improves the reliability of a software system</td></tr><tr><td>iv. Node reduction algorithm</td><td>D) Simplifies the structure of a graph</td></tr></table>		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	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											
5M-4	<table><tr><td>i. Graph matrix</td><td>A) Used to identify all possible defects in a software system</td></tr><tr><td>ii. Relation</td><td>B) Used to simplify the structure of a graph</td></tr><tr><td>iii. Power of a matrix</td><td>C) Used to calculate the number of cycles in a graph</td></tr><tr><td>iv. Node reduction algorithm</td><td>D) Used to represent the structure of a graph</td></tr></table>		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	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											
5M-5	<table><tr><td>i. Integration of graph matrices with node reduction algorithms</td><td>A) To improve the overall quality of a software system</td></tr><tr><td>ii. Use of relations to improve the coverage of testing</td><td>B) To reduce the number of defects in a software system</td></tr><tr><td>iii. Use of power of a matrix to simplify the structure of a graph</td><td>C) To improve the reliability of a software system</td></tr><tr><td>iv. Use of graph matrices to improve the reliability of a software system</td><td>D) To simplify the structure of a graph</td></tr></table>	i. Integration of graph matrices with node reduction algorithms	A) To improve the overall quality of a software system	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	V	CO6	
	i. Integration of graph matrices with node reduction algorithms	A) To improve the overall quality of a software system										
	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

BR-16

SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY

D4

(An Autonomous Institution under UGC, New Delhi)

Recognized under 2(f) and 12(B) of UGC Act 1956

III B.Tech - II Semester –End Examinations (Regular) May - 2019

(R16CSE1121) SOFTWARE TESTING METHODOLOGIES

(Common to CSE, IT)

Duration: 3 Hrs

04.05.2019

Max Marks: 70M

Section – A

Answer All the following questions

Marks: 5Qx4M = 20M

1. Compare among testing and debugging.
2. Mention the applications of data flow testing.
3. Describe about nice and ugly domains.
4. What is data flow anomaly problem?
5. What are the principles of state testing?

Section – B

Answer any FIVE questions choosing at least one from each Unit

Marks: 5Qx10M = 50M

UNIT-I

6. Discuss about the taxonomy of bugs in detail.

(OR)

7. Discuss about path sensitizing and path instrumentation.

UNIT-II

8. Write short notes on the following:

i) Transaction Flow Graphs ii) Data flow testing.

(OR)

9. Discuss about various Transaction flow testing techniques.

UNIT-III

10. Explain about domain bugs and how to test them.

(OR)

11. Write how to perform interface testing.

UNIT-IV

12. Explain about KV charts in detail.

(OR)

13. Discuss about path products and path expressions.

UNIT-V

14. Explain about the Partitioning Algorithm.

(OR)

15. Discuss about the Node Reduction Algorithm.

SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY **D4**
(An Autonomous Institution Under 2(f) and 12(B) of UGC Act 1956, New Delhi)
II B.Tech - II Semester - II Mid Term Examinations
(R16CSE1121) SOFTWARE TESTING METHODOLOGIES
(Common to CSE, IT)

Duration: 90Mins

Date: 15.04.2019 AN

Max Marks: 25M

Section – A

Answer All the questions

Marks: 5Qx1M = 5M

1. Define linear predicates.
2. Define Path expression.
3. What is path Sum?
4. Demonstrate Dead States.
5. Describe loop reduction.

Section – B

Answer any FOUR questions

Marks: 4Qx5M = 20M

6. Explicate how one-dimensional domains are tested.
7. Differentiate in detail the domains and interface testing.
8. Example Huang's theorem with example.
9. Discuss Path Sums and Path Product.
10. Discuss about good and bad state graphs.
11. Explicate various types of relations related to matrix.

III B.Tech- II Semester –End Examinations (Regular/Suppl.) May - 2018
SOFTWARE TESTING METHODOLOGIES
(Common to CSE, IT)

Duration: 3 Hrs**Max Marks: 70M****Section – A****Answer All the following questions****Marks: 5Qx4M = 20M**

1. Explain the testing process.
2. Differentiate between fault and failure.
3. What are the different testing features?
4. Define directed and Undirected graphs with example.
5. Explain about testing categorization.

Section – B**Answer any FIVE questions choosing at least one from each Unit****Marks: 5Qx10M = 50M****UNIT - I**

6. a) Why testing is important in software development?
b) Mention the purpose of testing.

(OR)

7. a) Mention the applications of path testing
b) Explain the basic concepts of path testing.

UNIT - II

8. a) What is meant by transaction flow. How they are different from data flow diagram.
b) Explain the transaction flow testing techniques.

(OR)

9. a) What are the different strategies in data flow testing.
b) Explain Object Class testing with examples.

UNIT - III

10. a) What are domains and Paths. Discuss about Domain testing.
b) Describe about Domain and Interface Testing.

(OR)

11. a) What are the objectives of Integration Testing.
b) What is meant by testability? Explain various properties of testability.

UNIT - IV

12. a) Differentiate between path product and path expression.
b) Discuss about Logic based testing.

(OR)

13. Explain the Huang's theorem with example.

UNIT - V

14. a) What is State Graphs. Discuss about its features.
b) Differentiate between Good and Bad State Graphs.

(OR)

15. What is Node Reduction Algorithm.

**BR-14
D4**

SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY

(An Autonomous Institution Under 2(f) and 12(B) of UGC Act 1956, New Delhi)

III B.Tech - II Semester - II Mid Term Examinations

(R14CSE1121) SOFTWARE TESTING METHODOLOGIES

(Common to CSE, IT)

Duration: 90Mins

Date: 09.04.2018 AN

Max

Marks: 25M

Section – A

Answer All the questions

Marks:

1Mx5Q = 5M

1. Define nice domains.
2. Define condition stub.
3. Define action stub.
4. Define Finite State Machine.
5. Define State Graph.

Section – B

Answer any FOUR questions

Marks: 5Mx4Q = 20M

6. Elucidate domain testing. Discuss the various applications of domain testing.
7. With a neat diagram, explicate the schematic representation of domain testing.
8. Discuss Path Sums and Path Product.
9. Explicate about the ambiguities and contradictions in specifications.
10. Discuss node reduction algorithm.
11. Write about loops in matrix representation.

BR-14

SRI INDU COLLEGE OF ENGINEERING & TECHNOLOGY

D4

(An Autonomous Institution under UGC, New Delhi)

Recognized under 2(f) and 12(B) of UGC Act 1956

III B.Tech - II Semester –End Examinations (Regular) April/May - 2017

(R14CSE1121) SOFTWARE TESTING METHODOLOGIES

(Common to CSE, IT)

Duration: 3 Hrs

Max Marks: 70M

Section – A

Answer Allthe following questions

Marks: 5x4 = 20M

1. Explain the Traversal marker form for path instrumentation.
2. Write a short note on transaction flows. Write its complications.
3. What is the purpose of domain testing? Give its schematic representation.
4. What is the significance of decision tables in logic based testing? Write its applications.
5. State and explain principles of state testing. What is the impact of Bugs in State Testing?

Section – B

Answer any FIVE of the following questions

Marks: 5x10 = 50M

UNIT-I

- 6.a) Explain Heuristic procedures for sensitizing paths.
- b) What is predicate coverage? Explain with example.

(OR)

7. a) State and explain various dichotomies in software testing.
- b) Draw and explain model of testing. Is complete testing possible? Discuss.

UNIT-II

8. a) The transaction flows are often ill structured. Discuss its reasons.
- b) Illustrate the applications of data flow testing.

(OR)

9. a) What is meant by transaction flow testing? Discuss its significance.
- b) Explain the application tools and effectiveness of data flow testing.

UNIT-III

10. a) What are domain bugs? Illustrate how to test them.
- b) State and explain with suitable examples various two-dimensional domain bugs.

(OR)

- 11.a) What is meant by domain dimensionality? Explain.
- b) State and explain various restrictions at domain testing processes.

UNIT-IV

12. State Huang's theorem and explain its implementation. Also explain its generalizations and limitations.

(OR)

13. What are the decision tables? Illustrate about decision table processors and Kv-charts.

UNIT-V

14. a) Discuss briefly with an example about good state graphs and bad state graphs.
- b) Explain about the node reduction algorithm.

(OR)

15. Discuss briefly about the power of a matrix. Write relative merits and demerits of different Graph Matrix representations.
