South Indian Education Society’s

Graduate School of Technology

NAAC 'A+' ,NBA Accredited (EXTC,CE,IT)

Autonomous Institute Affiliated to University of Mumbai

Department of Information Technology Engineering

Curriculum Structure FE to B.E

and

Second Year Syllabi

Board of Studies
Department of Information Technology
Bachelor of Engineering

Academic Council
SIES Graduate School of Technology

Effective from: AY 2025-26

PREAMBLE

Greetings to All Stakeholders,

We are happy to share that the SIES Graduate School of Technology has been granted autonomous status by the University Grants Commission (UGC) for the academic year 2024-2025 and beyond. The autonomy will enable us to design and implement a more dynamic and learner-centric curriculum that responds effectively to the evolving needs of society and industry.

The Department of Information Technology (IT) at SIES Graduate School of Technology, Nerul, Navi Mumbai, was established in 2002, and has since been a cornerstone of our institution's commitment to delivering high-quality technical education. Our department is dedicated to providing students with a comprehensive education in IT, equipping them with the skills and knowledge needed to excel in a rapidly evolving technological landscape. The department has been consistently recognized for its academic excellence, with accreditation from the National Board of Accreditation (NBA) in 2011, 2021, and 2024, reflecting our dedication to maintaining high standards in education and infrastructure.

The curriculum for the Department of Information Technology is designed to address the dynamic needs of the industry while fostering a deep understanding of core IT concepts. Our academic offerings include core subjects covering a range of fundamental topics such as programming, data structures, algorithms, database management, networking. In response to the evolving IT landscape, our program integrates cutting-edge topics such as Artificial Intelligence, Machine Learning, Cyber security, Cloud Computing, and Blockchain Technology. Students have the opportunity to choose electives and specialize in areas of interest, allowing them to tailor their education to their career goals and emerging industry trends. We emphasize strong industry connections through Internships and Projects. The department is committed to the holistic development of students. Our goal is to instill a passion for lifelong learning in our students.

Empowering our students with this innovative curriculum will develop them into pioneering engineers, ethical leaders, and global citizens. Our aim is for them to drive technological advancement and contribute positively to society.

We extend our heartfelt thanks to all our stakeholders for their unwavering support and look forward to a transformative journey ahead with our new autonomous status and curriculum.

Chairperson
Board of Studies
Information Technology Engineering
SIES Graduate School of Technology

HEAD
Department of Infomation Technology
S. I. E. S. Graduate School of Technology
Sri Chandrasekarendra Saraswathy Vidyapuram
Plot-1-C & E, Sector-V, Nerul, Navi Mumbai-400706

Chairperson
Academic Council
SIES Graduate School of Technology

PRINCIPAL
S.I.E.S. GRADUATE SCHOOL OF TECHNOLOGY (AUTONOMOUS)
Plot 1C/D/E, Sri Chandrasekarendra Saraswathy Vidyapuram
Sector - V, Nerul, Navi Mumbai - 400 706.

SIES Graduate School of Technology (Autonomous)

Department of Information Technology

Bachelor of Engineering

Semester wise Credit distribution structure for Four Year UG Engineering

Program - Information Technology: One Major, One Minor

Semester		I	II	III	IV	V	VI	VII	VIII	Total Credits
Basic Science Course (BSC)	BSC/ESC	07	06		--	--	--	--	--	13
Engineering Science Course (ESC)		09	10		--	--	--	--	--	19
Programme Core Course (PCC)	Program Courses	--	--	17	12	12	11	04	--	56
Programme Elective Course (PEC)		--	--	--	--	03	04	06	--	13
Multidisciplinary Minor (MDM)	Multidisciplinary Courses		-	--	03	04	04	04	--	15
Open Elective (OE) Other than a particular program		--	--	--	--	--	--	03	03	06
Vocational and Skill Enhancement Course (VSEC)	Skill Courses	01	01	02	02	--	02	--	--	08
Ability Enhancement Course (AEC -01, AEC-02)	Humanities Social Science and Management (HSSM)		02	--	--	02	--	--	--	04
Entrepreneurship/Economics/ Management Courses		--		02	02	--	--	--	--	04
Indian Knowledge System (IKS)			02		--	--	--	--	--	02
Value Education Course (VEC)		--	--	--	02	--	--	--	--	02
Research Methodology (RM)	Experiential Learning Courses	--	--	--	--	--	--	_	03	03
Community Engagement Project (CEP) / Field Project (FP)		--	--	01	01	--	--	-	-	02
Project		--	--	--	--	01	01	02	02	06
Internship/On Job Training (OJT)		--	---		--	--	--	–	09	09
Co-curricular Courses (CC)	Liberal Learning Courses	04			--	--	--	--	-	04
Total Credits (Major)		**21**	**21**	**22**	**22**	**22**	**22**	**19**	**17**	**166**

CURRICULUM STRUCTURE

SECOND YEAR ENGINEERING

(Information Technology)

Academic Year 2025-26

Nomenclature of the courses in the curriculum	
Abbreviation	**Title**
BSC	Basic Science Courses
ESC	Engineering Science Courses
PCC	Program Core Courses
PEC	Program Elective Course
MDM	Multidisciplinary Minor
OE	Open Elective
VSEC	Vocational and Skill Enhancement Course
AEC	Ability Enhancement Course
CC	Cocurricular Courses
LLC	Liberal Learning Courses
VEC	Value Education Course
RM	Research Methodology
IKS	Indian Knowledge System
CEP/ FP	Community Engagement Project/ Field Project
ISE	In Semester Examination
MSE	Mid Semester Examination
ESE	End Semester Examination
CIAP	Continuous Internal Assessment Practical
ESEP	End Semester Examination Practical

Program Structure for First Year

W.E.F. A.Y. 2024-25

Semester I

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract	Tut.	Theory	Pract.	Tut.	Total
FEC101	Applied Mathematics -I	BSC	3	--	--	3	--	--	3
FEC1021/ FEC1022	Applied Physics/ Applied Chemistry @	BSC	3	--	--	3	--	--	3
FEC103	Basic Electrical & Electronics Engineering	ESC	2	--	--	2	--	--	2
FEC104	C-Programming	ESC	2	--	--	2	--	--	2
FEC105	Applied Mechanics and Robot Dynamics	ESC	2	--	--	2	--	--	2
FEL1011/ FEL1012	Applied Physics Lab/ Applied Chemistry Lab @	BSC	--	1	--	--	0.5	--	0.5
FEL102	Basic Electrical & Electronics Engineering Lab	ESC	--	2	--	--	1	--	1
FEL103	C-Programming Lab	ESC	--	2	--	--	1	--	1
FEL104	Applied Mechanics and Robot Dynamics Lab	ESC	--	2	--	--	1	--	1
FEL105	Engineering Workshop-I	VSEC	--	2	--	--	1	--	1
FEL106	Health, Wellness and Mindfulness	CC	--	$2^{\#}+2$	--	--	2	--	2
FEL107	Induction Cum Universal Human Values	CC	--	5*	--	--	2.5	--	2.5
	Total		**12**	**18**	**--**	**12**	**9**	**--**	**21**

Examination Scheme-FY Semester-I

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		$ESE^{\$}$	Exam Duration (Hrs.)			
		ISE	MSE					
FEC101	Applied Mathematics -I	20	20	60	3	--	--	100
FEC1021/ FEC1022	Applied Physics/ Applied Chemistry @	20	20	60	3	--	--	100
FEC103	Basic Electrical & Electronics Engineering	15	15	45	2	--	--	75
FEC104	C-Programming	15	15	45	2	--	--	75
FEC105	Applied Mechanics and Robot Dynamics	15	15	45	2	--	--	75
FEL1011/ FEL1012	Applied Physics Lab/ Applied Chemistry Lab@	--	--	--	--	25	--	25
FEL102	Basic Electrical & Electronics Engineering Lab	--	--	--	--	25	25	50
FEL103	C-Programming Lab	--	--	--	--	25	25	50
FEL104	Applied Mechanics and Robot Dynamics Lab	--	--	--	--	25	25	50
FEL105	Engineering Workshop-I	--	--	--	--	25	--	25
FEL106	Health, Wellness and Mindfulness	--	--	--	--	25	–	25
FEL107	Induction Cum Universal Human Values	--	--	--	--	25	--	25
	Total	**85**	**85**	**255**		**175**	**75**	**675**

@Physics/Chemistry in one semester.
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.
Theory 1 credit for 1 hour and practical 1 credit for 2 hours.
*Indicates workload of a learner for UHV. Faculty Load: ½ hour per week per four groups
# Two hours of practical class to be conducted for full class as demo/ discussion.
ISE: In Semester Examination**:** To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test etc. of 20/15 marks.
MSE: Mid Semester Examination: To be conducted as written examination for 20/15 marks of duration 1 Hr.
ESE: End Semester Examination
CIAP: Continuous Internal Assessment Practical. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).
ESEP: End Semester Examination Practical. Oral/Practical Examination will be conducted as End Semester Examination Practical (ESEP).

Program Structure for First Year

W.E.F. A.Y. 2024-25

Semester II

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
FEC201	Applied Mathematics -II	BSC	3	--	--	3	--	--	3
FEC2021/ FEC2022	Applied Physics/ Applied Chemistry @	BSC	3	--	--	3	--	--	3
FEC203	Engineering Graphics	ESC	2	--	--	2	--	--	2
FEC204	Digital System Design	ESC	3	--	--	3	--	--	3
FEC205	Professional Communication Techniques	AEC	2	--	--	2	--	--	2
FEL2011/ FEL2012	Applied Physics Lab/ Applied Chemistry Lab @	BSC	--	1	--	--	0.5	--	0.5
FEL202	Engineering Graphics Lab	ESC	--	2	--	--	1	--	1
FEL203	Digital System Design Lab	ESC	--	2	--	--	1	--	1
FEL204	Professional Communication Techniques Lab	AEC	--	1	--	--	0.5	--	0.5
FEL205	Object Oriented Programming Methodology Lab	ESC	--	2*+2	--	--	2	--	2
FEL206	Engineering Workshop-II	VSEC	--	2	--	--	1	--	1
FEL207	Indian Knowledge System	CC	--	2*+2	--	--	2	--	2
	Total		**13**	**16**	--	**13**	**8**	--	**21**

Examination Scheme-FY Semester-II

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
FEC201	Applied Mathematics -II	20	20	60	03	--	--	100
FEC2021/ FEC2022	Applied Physics/ Applied Chemistry @	20	20	60	03	--	--	100
FEC203	Engineering Graphics	15	15	45	03	--	--	75
FEC204	Digital System Design	20	20	60	03	--	--	100
FEC205	Professional Communication Techniques	15	15	45	02	--	--	75
FEL2011/ FEL2012	Applied Physics Lab/ Applied Chemistry Lab @	--	--	--	--	25	--	25
FEL202	Engineering Graphics Lab	--	--	--	--	25	25	50
FEL203	Digital System Design Lab	--	--	--	--	25	25	50
FEL204	Professional Communication Techniques Lab	--	--	--	--	25	--	25
FEL205	Object Oriented Programming Methodology Lab	--	--	--	--	25	25	50
FEL206	Engineering Workshop-II	--	--	--	--	25	--	25
FEL207	Indian Knowledge System	--	--	--	--	25	–	25
Total		**90**	**90**	**270**	--	**175**	**75**	**700**

@Physics/Chemistry in one semester.
* Two hours of practical class to be conducted for full class as demo/ discussion.
# Course evaluation is an activity based which may be an individual or group of students.
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.
Theory 1 credit for 1 hour and practical 1 credit for 2 hours.
ISE: In Semester Examination: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test etc. of 20/15 marks.
MSE: Mid Semester Examination: To be conducted as written examination for 20/15 marks of duration 1 Hr.
ESE: End Semester Examination
CIAP: Continuous Internal Assessment Practical. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).
ESEP: End Semester Examination Practical. Oral/Practical Examination will be conducted as End Semester Examination Practical (ESEP).

Program Structure for Second Year

W.E.F. A.Y. 2025-26

Semester III

Course Code	Course Name		Teaching Scheme			Credits Assigned			
		Category	(Contact Hours)						
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
ITC301	Applied Mathematics-III	PCC	3	-	-	3	-	-	3
ITC302	Data Structure and Analysis	PCC	3	-	-	3	-	-	3
ITC303	Database Management Systems	PCC	3	-	-	3	-	-	3
ITC304	Automata Theory	PCC	2	-	1	2	-	1	3
ITC305	Computer Organization and Architecture	PCC	3	-	-	3	-	-	3
ITC306	Engineering Economics	HSSM	2	-	-	2	-	-	2
ITL301	Data Structure Lab	PCC	-	2	-	-	1	-	1
ITL302	Structural Query Language Lab	PCC	-	2	-	-	1	-	1
ITL303	Skill Lab (Python)	VSEC	-	2*+2	-	-	2	-	2
ITM301	Mini Project 1A (Python)	CEP	-	2#	-	-	1	-	1
Total			**16**	**10**	**1**	**16**	**5**	**1**	**22**

Examination Scheme - IT Semester-III

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
ITC301	Applied Mathematics-III	20	20	60	3	-	-	100
ITC302	Data Structure and Analysis	20	20	60	3	-	-	100
ITC303	Database Management Systems	20	20	60	3	-	-	100
ITC304	Automata Theory	15	15	45	2	25	-	100
ITC305	Computer Organization and Architecture	20	20	60	3		-	100
ITC306	Engineering Economics	50	-	-	-	-	-	50
ITL301	Data Structure and Analysis Lab	-	-	-	-	25	25	50
ITL302	Structural Query Language Lab	-	-	-	-	25	25	50
ITL303	Skill Lab (Python)	-	-	-	-	25	25	50
ITM301	Mini Project 1A (Python)	-	-	-	-	25	25	50
Total		**120**	**120**	**285**	-	**125**	**100**	**750**

* Two hours of practical class to be conducted for full class as demo/ discussion.
# Indicates workload of a learner (Not faculty) for Mini Project 1A. Faculty Load: ½ hour per week per four groups
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

Program Structure for Second-Year

W.E.F.A.Y.2025-26

Semester IV

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
ITC401	Applied Mathematics-IV	PCC	3	-	-	3	-	-	3
ITC402	Computer Network	PCC	3	-	-	3	-	-	3
ITC403	Operating System	PCC	3		-	3		-	3
ITC404	Critical Thinking and Design	HSSM (AEC)	2	-	-	2	-	-	2
MDMC40X1	Multidisciplinary Minor (MDM-I)	MDM	3	-	-	3	-	-	3
ITL401	Computer Network Lab	PCC	-	2	-	-	1	-	1
ITL402	Operating System Lab	PCC	-	2	-	-	1	-	1
ITL403	Internet Programming Lab	PCC	-	2	-	-	1	-	1
ITL404	Skill Lab (MAD)	VSEC	-	2*+2	-	-	2	-	2
ITL405	Value Education (UHV)	HSSM (VEC)	-	4	-	-	2	-	2
ITM401	Mini Project-1 B (Web Technology)	CEP	-	2#	-	-	1	-	1
Total			**14**	**16**	-	**14**	**8**	-	**22**

Examination Scheme - IT Semester-IV

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
ITC401	Applied Mathematics-IV	20	20	60	3	-	-	100
ITC402	Computer Network	20	20	60	3			100
ITC403	Operating System	20	20	60	3			100
ITC404	Critical Thinking and Design	15	15	45	2	-	-	75
MDMC40X1	Multidisciplinary Minor -1	20	20	60	3			100
ITL401	Computer Network Lab	-	-	-	-	25	25	50
ITL402	Operating System Lab	-	-	-	-	25	25	50
ITL403	Internet Programming Lab					25	25	50
ITL404	Skill Lab (MAD)	-	-	-	-	25	25	50
ITL405	Value Education (UHV)	-	-	-	-	50	-	50
ITM401	Mini Project-1 B (Web Technology)	-	-	-	-	25	25	50
Total		**95**	**95**	**285**		**175**	**125**	**775**

UHV: Universal Human Values

* Two hours of practical class to be conducted for full class as demo/ discussion.UHV: Universal Human Values

# Indicates workload of a learner (Not faculty) for Mini Project 1B. Faculty Load: ½ hour per week per four groups

$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

Program Structure for Third Year

W.E.F.A.Y.2026-27

Semester V

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
ITC501	Cryptography and Network Security	PCC	3	-	-	3	-	-	3
ITC502	Software Engineering and Agile Software Development	PCC	3	-	-	3	-	-	3
ITC503	Cloud Computing	PCC	3	-	-	3	-	-	3
MDMC50X1	Multidisciplinary Minor (MDM-II)	MDM	3	-	-	3	-	-	3
ITPEC501X	Program Elective I	PEC	3	-	-	3	-	-	3
ITL501	Security Lab	PCC	-	2	-	-	1	-	1
ITL502	Software Engineering Lab	PCC	-	2	-	-	1	-	1
ITL503	Cloud Computing Lab	PCC	-	2	-	-	1	-	1
ITL504	Professional Communication and Ethics	HSSM (AEC)	-	2*+2	-	-	2	-	2
MDML50X1	Multidisciplinary Minor Lab (MDM-II)	MDM	-	2	-	-	1	-	1
ITM501	Mini Project-2A (Cloud Computing)	MP	-	2#	-	-	1	-	1
Total			**15**	**14**	**-**	**15**	**7**	**-**	**22**

Examination Scheme – IT Semester-V

Course Code	Course Name	Examination Scheme						
		Theory						Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)	CIAP	ESEP	
		ISE	MSE					
ITC501	Cryptography and Network Security	20	20	60	3	-	-	100
ITC502	Software Engineering and Agile Software Development	20	20	60	3	-	-	100
ITC503	Cloud Computing	20	20	60	3	-	-	100
MDMC50X1	Multidisciplinary Minor (MDM-II)	20	20	60	3	-	-	100
ITPEC501X	Program Elective I	20	20	60	3	-	-	100
ITL501	Security Lab	-	-	-	-	25	25	50
ITL502	Software Engineering Lab	-	-	-	-	25	25	50
ITL503	Cloud Computing Lab	-	-	-	-	25	-	25
ITL504	Professional Communication and Ethics	-	-	-	-	50	-	50
MDML50X1	Multidisciplinary Minor Lab (MDM-II)	-	-	-	-	25	25	50
ITM501	Mini Project-2A (Cloud Computing)	-	-	-	-	25	25	50
Total		**100**	**100**	**300**	-	175	**100**	**775**

* Two hours of practical class to be conducted for full class as demo/ discussion.

# Indicates workload of a learner (Not faculty) for Mini Project 2A. Faculty Load: ½ hour per week per four groups

$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

Program Elective – I

Technology Bucket			
IoT & Computing	**Data Engineering**	**Computer Graphics & Multimedia**	**Network and Security**
ITPEC5011: Embedded System	**ITPEC5012:** Advanced Database Management Techniques	**ITPEC5013:** Computer Graphics and Multimedia	**ITPEC5014:** Network Design & Management

Program Structure for Third Year

W.E.F.A.Y.2026-27

Semester VI

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)		Credits Assigned		
			Theory	Pract.	Theory	Pract.	Total
ITC601	Date Warehouse and Data Mining	PCC	3	-	3	-	**3**
ITC602	Cyber and Infrastructure Security	PCC	3	-	3	-	**3**
ITC603	Wireless and Sensor Technology	PCC	3	-	3	-	**3**
MDMC60X1	Multidisciplinary Minor (MDM-III)	MDM	3	-	3	-	**3**
ITPEC601X	Program Elective II	PEC	3		3	-	**3**
ITL601	Data Minig Lab	PCC	-	2	-	1	**1**
ITL602	Sensor Lab	PCC	-	2	-	1	**1**
ITL603	Skill Lab (DevOps Lab)	VSEC	-	2*+2	-	2	**2**
MDML60X1	Multidisciplinary Minor Lab (MDM-III)	MDM	-	2	-	1	**1**
ITPEL601X	Program Elective II Lab	PEC	-	2	-	1	**1**
ITM601	Mini Project -2B (ML/ DS/ WS)	MP	-	2#	-	1	**1**
Total			**15**	**14**	**15**	**7**	**22**

Examination Scheme - IT Semester-VI

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
ITC601	Date Warehouse and Data Mining	20	20	60	3	-	-	100
ITC602	Cyber and Infrastructure Security	20	20	60	3	-	-	100
ITC603	Wireless and Sensor Technology	20	20	60	3	-	-	100
MDMC60X1	Multidisciplinary Minor (MDM-III)	20	20	60	3	-	-	100
ITPEC601X	Program Elective II	20	20	60	3	-	-	100
ITL601	Data Minig Lab	-	-	-	-	25	25	50
ITL602	Sensor Lab	-	-	-	-	25	25	50
ITL603	Skill Lab (DevOps Lab)	-	-	-	-	25	25	50
MDML60X1	Multidisciplinary Minor Lab (MDM-III)	-	-	-	-	25	25	50
ITPEL601X	Program Elective II Lab	-	-	-		25		25
ITM601	Mini Project -2B (ML/ DS/ WS)	-	-	-	-	25	25	50
Total		**100**	**100**	**300**	-	**150**	**125**	**775**

* Two hours of practical class to be conducted for full class as demo/ discussion.

# Indicates workload of a learner (Not faculty) for Mini Project 2B. Faculty Load: ½ hour per week per four groups

$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

Program Elective- II

Technology Bucket			
IoT & Computing	**Data Engineering**	**Computer Graphics & Multimedia**	**Network and Security**
ITPEC6011:	**ITPEC6012:**	**ITPEC6013:**	**ITPEC6014:**
High-Performance Computing	Data Analytics and Visualization	Image Processing & Computer Vision	Ethical Hacking and Forensics
ITPEL6011:	**ITPEL6012:**	**ITPEL6013:**	**ITPEL6014:**
High-Performance Computing Lab	Data Analytics and Visualization Lab	Image Processing & Computer Vision Lab	Ethical Hacking and Forensics Lab

Program Structure for Fourth Year

W.E.F.A.Y.2027-28
Semester VII

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)		Credits Assigned		
			Theory	Pract.	Theory	Pract.	Total
ITC701	Big Data Storage Processing and Management	PCC	3	-	3	-	3
MDMC70X1	Multidisciplinary Minor (MDM-IV)	MDM	3	-	3	-	3
ITPEC701X	Program Elective III	PEC	3		3		3
ITPEC702X	Program Elective IV	PEC	3		3		3
OEC701X	Open Elective I	OE	3	-	3	-	3
ITL701	Big Data Storage Processing and Management Lab	PCC	-	2	-	1	1
MDML70X1	Multidisciplinary Minor Lab (MDM-IV)	MDM	-	2	-	1	1
ITP701	Major Project -1	Project	-	4#	-	2	2
Total			15	8	15	4	19

Examination Scheme - IT Semester-VII

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
ITC701	Big Data Storage Processing and Management	20	20	60	3	-	-	100
MDMC70X1	Multidisciplinary Minor (MDM-IV)	20	20	60	3	-	-	100
ITPEC701X	Program Elective III	20	20	60	3	-	-	100
ITPEC702X	Program Elective IV	20	20	60	3	-	-	100
OEC701X	Open Elective I	20	20	60	3	-	-	100
ITL701	Big Data Storage Processing and Management Lab	-	-	-	-	25	25	50
MDML70X1	Multidisciplinary Minor Lab (MDM-IV)	-	-	-	-	25	25	50
ITP701	Major Project -1	-	-	-	-	25	25	50
Total		100	100	300	-	75	75	650

#Indicates workload of Learner (Not faculty), for Major Project
Project Guide Load = ½ hour per week per project group
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

Program Elective-III

Technology Bucket			
IOT & Computing	**Data Engineering**	**Computer Graphics & Multimedia**	**N/w and Security**
ITPEC7011: Internet of Everything	**ITPEC7012:** Web Analytics	**ITPEC7013:** Augmented Reality and Virtual Reality	**ITPEC7014:** Blockchain Technology

Program Elective-IV

Technology Bucket			
IoT & Computing	**Data Engineering**	**Computer Graphics & Multimedia**	**N/w and Security**
ITPEC7021: Quantum Computing	**ITPEC7022:** Reinforcement Learning	**ITPEC7023:** Immersive Game Development (Game Theory)	**ITPEC7024:** TICTH: Threat Intelligence and Cyber Threat Hunting

Open Elective -I

Course Code	Course Name
OEC7011	Project Management
OEC7012	Finance Management
OEC7013	Management Information System
OEC7014	Entrepreneurship Development and Management
OEC7015	Operation Research
OEC7016	Disaster Management and Mitigation Measures
OEC7017	Product Design

Program Structure for Fourth Year

W.E.F.A.Y.2027-28

Semester VIII

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)		Credits Assigned		
			Theory	Pract.	Theory	Pract.	Total
ITC801	Research Methodology	RM	3	-	3	-	3
OE801X	Open Elective II	OE	3	-	3	-	3
ITP801	Major Project -II	Project	-	4#	-	2	2
ITINT801	Internship/Project/Research	Internship	-	-	-	9	9
Total			**6**	**4**	**6**	**11**	**17**

Examination Scheme - IT Semester-VIII

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISC	MSC					
ITC703	Research Methodology	20	20	60	3	-	-	100
OEC801X	Open Elective II	20	20	60	3	-	-	100
ITP801	Major Project-II	-	-	-	-	100	50	150
ITINT801	Internship	-	-	-	-	200	_	200
Total		40	40	**120**	-	**300**	**50**	**550**

#Indicates workload of Learner (Not faculty), for Major Project
Project Guide Load = ½ hour per week per project group
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

Open Elective -II

Course Code	Course Name
OEC8011	Enterprise Resource Planning
OEC8012	Cyber Security and Laws
OEC8013	Energy Audit and Management
OEC8014	IPR and Patenting
OEC8015	Environmental Management
OEC8016	Digital Business Management
OEC8017	Human Resource Management

Multidisciplinary Minor (MDM)

Track	Minor Track	Partner Institute if any	Module	Code	Eligible
1	ML	SIES GST	Artificial Intelligence	MDMC4011	IT/EXTC/CSE IOT
			Machine Learning	MDMC5012	
			Natural Language Processing	MDMC6013	
			Deep Learning	MDMC7014	
2	DS	SIES GST	Statistical Foundation for Data Science	MDMC4021	ECS/CE/EXTC
			Data Analytics & Visualization	MDMC5022	
			Decision Making & Business Intelligence	MDMC6023	
			Big Data Analytics	MDMC7024	
3	Embedded Systems	SIES GST	Microprocessor and Microcontrollers	MDMC4031	CE/AIDS/AIML
			RTOs and Embedded systems	MDMC5032	
			Sensor Technology	MDMC6033	
			Industrial Internet of Things	MDMC7034	
4	Cyber Security	SIES GST	Computer Network	MDMC4041	AIDS/AIML
			Cryptography & System Security	MDMC5042	
			Cloud Computing and Security	MDMC6043	
			Digital Forensics	MDMC7044	
5	System Programming	SIES GST	Advance Data Structure	MDMC4051	CSEIOT/ECS/IT
			Advance Algorithm	MDMC5052	
			System Programming and Compiler Construction	MDMC6053	
			Distributed Systems	MDMC7054	
6	Management	SIESSBS	Cost Management	MDMC4061	EXTC/CE/IT/ECS/AIDS/AIML/CSE IOT
			Supply Chain Management	MDMC5062	
			HR & Organization	MDMC6063	
			Marketing Management	MDMC7064	

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC301	Applied Mathematics-III	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
ITC301	Applied Mathematics-III	20	20	60	--	--	100

Pre-requisite:

1. Applied Mathematics I
2. Applied Mathematics II

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions
4. PO4: Conduct Investigation of complex problems

Course Objectives:

1. Evaluate eigenvalues and eigenvectors and apply them to solve systems of linear equations and matrix diagonalization
2. Evaluate matrix operations, singular value decomposition (SVD), and their applications.
3. Understand fundamental concepts of logic, set theory, and functions to develop mathematical reasoning.
4. Apply combinatorial principles and counting techniques in problem-solving and Analyse number theory concepts and modular arithmetic in cryptographic applications.
5. Understand the concept of complex variables, C-R equations, harmonic functions and their conjugate and mapping in complex plan.
6. Evaluate line and contour integrals and construct the power series expansion of a complex-valued function.

Course Outcomes:

After successful completion of the course, students will be able to :

1. Evaluate matrix operations, eigenvalues, eigenvectors, and diagonalization properties.
2. Apply the concept of singular value decomposition (SVD) to decompose the matrix.
3. Explain concepts of propositional logic, set theory, and functions and their role in problem-solving.
4. Apply combinatorial techniques to solve counting and permutation problems, and analyse number theory concepts, modular arithmetic, and their cryptographic applications.
5. Apply the concept of complex numbers, complex functions, and analyze their significance

in data science and engineering.

6. Apply the concepts of Complex Integration to evaluate integrals, analyze and compute residues, and solve various contour integrals.

Module No.	Unit No.	Topic	Hours	Mapped CO
1	Matrices and Linear Algebra		7	CO1
	1.1	**Characteristic Equation, Eigenvalues and Eigenvectors:** Properties (without proof), Cayley-Hamilton theorem (without proof), verification and reduction of higher-degree polynomials.		
	1.2	**Similarity of Matrices:** Diagonalizable and non-diagonalizable matrices.		
	1.3	**Functions of Square Matrices:** Derogatory and non-derogatory matrices.		
		Self-Learning Topics: Applications of eigenvalues in image processing, Markov chains.		
2	Singular Value Decomposition (SVD) and Applications		6	CO2
	2.1	**Introduction to SVD:** Definition, properties, and applications in data science and image compression.		
	2.2	**Computation of SVD:** Decomposing matrices, rank approximation, and principal component analysis (PCA).		
		Self-Learning Topics: Deep learning applications of SVD, Latent Semantic Indexing in NLP.		
3	Logic and Set Theory		7	CO3
	3.1	**Logic and Proofs:** Propositional logic, logical equivalence, predicates, quantifiers, rules of inference, proof techniques (direct, contrapositive, contradiction, induction).		
	3.2	**Sets and Relations:** Sets, operations on sets, Venn diagrams, properties of relations, equivalence relations, partial orders.		
	3.3	**Functions:** Definition, types of functions (injective, surjective, bijective), composition, inverse functions,		
	3.4	**Combinatorics:** Basics of counting, pigeonhole principle, inclusion-exclusion principle		
		Self-Learning Topics: Advanced proof techniques (Resolution, Natural Deduction), fuzzy logic applications. Ramanujan’s combinatorial theories.		
4	Number Theory and Cryptographic Foundations		5	CO4
	4.1	**Divisibility and Primes:** Euclidean algorithm, greatest common divisor (GCD), prime factorization, fundamental		

		theorem of arithmetic.		
	4.2	**Modular Arithmetic and Cryptography:** Congruences, Fermat's theorem, Euler's theorem.		
		Self-Learning Topics: RSA and ElGamal encryption, Cryptanalysis techniques, Blockchain cryptographic principles.		
5	**Complex Variable**		**7**	**CO5**
	5.1	Function f(z) of complex variable, limit, continuity and differentiability of f(z) Analytic function, necessary and sufficient conditions for f(z) to be analytic (without proof).		
	5.2	Cauchy-Riemann equations in cartesian coordinates (without proof).		
	5.3	Milne-Thomson method to determine analytic function f(z) when real part (u) or Imaginary part (v) or its combination (u+v or u-v) is given.		
	5.4	Harmonic function, Harmonic conjugate and orthogonal trajectories.		
		Self-learning Topics: Conformal mapping, linear, bilinear mapping, cross ratio, fixed points and standard transformations.		
6	**Complex Integration**		**7**	**CO6**
	6.1	Line Integral, Cauchy's Integral theorem for simple connected and multiply connected regions (without proof), Cauchy's Integral formula (without proof).		
	6.2	Taylor's and Laurent's series (without proof).		
	6.3	Definition of Singularity, Zeroes, poles of f(z), Residues, Cauchy's Residue Theorem (without proof).		
		Self-Learning Topics: Application of Residue Theorem to evaluate real integrations		

Textbooks:

1. **Erwin Kreyszig**, *Advanced Engineering Mathematics*, Wiley Eastern Limited, 10th Edition 2023-24
2. **Dr. B. S. Grewal**, Higher Engineering Mathematics, , Khanna Publication, 45th edition
3. **Kenneth H. Rosen**, *Discrete Mathematics and Its Applications*, McGraw Hill, 7th Edition, 2011.
4. **C.L. Liu and D.P. Mohapatra**, *Elements of Discrete Mathematics*, McGraw Hill, 4th Edition, 2012.
5. **Gareth A. Jones, J. Mary Jones**, *Elementary Number Theory*, Springer, 2nd Edition, 2012.
6. **Hoffman and Kunze**, *Linear Algebra*, Pearson, 2nd Edition, 2015.
7. **Ponnusamy**, *Foundations of Complex Analysis*, S, Narosa Publications, 2011.

Reference Books:

1. **R.P. Grimaldi**, *Discrete and Combinatorial Mathematics: An Applied Introduction*, Pearson, 5th Edition, 2003.

2. **Niven, Zuckerman, and Montgomery**, *An Introduction to the Theory of Numbers*, Wiley, 5th Edition, 2008.
3. **S. Lang**, *Linear Algebra*, Springer, 3rd Edition, 2004.
4. **Stinson, D.R.**, *Cryptography: Theory and Practice*, CRC Press, 3rd Edition, 2005.

Online Resources:

1. **MIT OpenCourseWare – Discrete Mathematics**
   a. https://ocw.mit.edu/courses/mathematics/
   b. Covers **Logic, Proofs, Sets, Functions, and Number Theory**.
2. **Coursera – Discrete Mathematics by UC San Diego**
   a. https://www.coursera.org/specializations/discrete-mathematics
   b. Covers **Combinatorics, Set Theory, and Proof Techniques**.
3. **Cryptography by Stanford (Coursera)**
   a. https://www.coursera.org/learn/crypto
   b. Covers **Modular Arithmetic, RSA, ElGamal, and Cryptanalysis**.
4. **MIT Linear Algebra Video Lectures**
   a. https://ocw.mit.edu/courses/mathematics/18-06-linear-algebra-spring-2010/
   b. Useful for **Matrices and SVD Applications**.

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 05 marks attendance.

MSE: To be conducted as a written examination for 20 marks (on 50% syllabus).

End Semester Examination

ESE is of duration 03 hours and 80 marks and will be scaled down to 60.

The question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC302	Data Structure and Analysis	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
ITC302	Data Structure and Analysis	20	20	60	--	--	100

Pre-requisite:

1. FEL103, FEL205: Knowledge of Basic Programming

Program Outcome mapped:

1. PO1: Engineering Knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.

Course Objectives

1. To understand the fundamental concepts of data structures such as stacks, queues, linked lists, trees, and graphs, and their applications in problem-solving.
2. To analyze and implement various tree structures, including binary trees, binary search trees, AVL trees, and B-trees, and understand their use in efficient data organization and retrieval.
3. To explore graph representations, traversal techniques, and algorithms for solving real-world problems such as shortest path and minimum spanning tree.
4. To develop a deep understanding of recursion, storage management techniques, and their applications in dynamic memory allocation and garbage collection.
5. To master searching and sorting algorithms, hashing techniques, and their performance analysis for efficient data processing.
6. To apply data structures and algorithms to solve real-world problems such as polynomial operations, expression evaluation, scheduling, and Huffman coding.

Course Outcomes

After successful completion of the course, student will be able to…

1. Explain and implement linear data structures such as stacks, queues, and linked lists, and analyse their time and space complexity.
2. Construct and manipulate tree structures, including binary search trees, AVL trees, and B-trees, and apply them to solve problems like searching, insertion, and deletion.

3. Represent graphs using adjacency matrices and lists, and apply traversal algorithms (BFS, DFS) and shortest path algorithms (Dijkstra's) to solve problems.
4. Write recursive functions, analyse their performance, and implement storage management techniques such as memory allocation and garbage collection.
5. Implement and analyse searching algorithms (binary search, hashing) and sorting algorithms (quick sort, merge sort) for efficient data processing.
6. Apply data structures and algorithms to solve real-world problems such as polynomial operations, expression evaluation, and Huffman coding.

Theory Syllabus

Module No.	Unit No.	Topics	Hrs.	Mapped to CO
0		**Prerequisite**	**1**	
	0.1	Defining, Declaring, and Initialization of Structure Variables.		
	0.2	Accessing Members of a Structure, Array of Structures, Nested Structures, Pointers to Structures.		
	0.3	Passing Structure, Structure Members, Structure Arrays, and Pointer to Structure as Function Parameters, Self-Referential Structures.		
		Self-Learning Topics: Memory Layout of Structures in C/C++, Union vs Structure, Introduction to Object-Oriented Programming (Classes and Objects).		
1.		**Stacks, Queues, and Linked Lists**	**8**	**CO1**
	1.1	Introduction to Data Structures: Linear and Non-Linear, Static and Dynamic.		
	1.2	Types of Asymptotic Notations in Complexity Analysis of Algorithms.		
	1.3	Concept of Stack and Queue, Array Implementation of Stack and Queue, Circular Queue, Double-Ended Queue, Priority Queue.		
	1.4	Concept of Linked Lists: Singly, Doubly, and Circular Linked Lists.		
	1.5	Insertion, Deletion, Update, and Copying Operations with Linked Lists, Reversing a Singly Linked List.		
	1.6	Linked List Implementation of Stack and Queue.		
		Self-Learning Topics: Skip Lists, XOR Linked Lists, Comparison of Linked Lists in Python vs C/C++, Applications in Real-Time Systems.	-	
2.		**Trees**	**8**	**CO2**
	2.1	Introduction to Trees: Terminology, Types of Binary Trees.		
	2.2	Non-Recursive Preorder, Inorder, and Postorder Traversal.		
	2.3	Creation of Binary Trees from Traversal.		
	2.4	Binary Search Tree (BST): Traversal, Searching, Insertion, and Deletion.		

	2.5	Threaded Binary Tree: Inorder Successor and Predecessor, Insertion, and Deletion.		
	2.6	AVL Tree: Searching, Traversing, Rotations (Right, Left), Insertion, and Deletion.		
	2.7	B-Tree and B+ Tree: Searching, Insertion, Deletion.		
		Self-Learning Topics: Trie, Segment Trees, Tree Implementations in Python, Red-Black Trees, Applications in Databases and File Systems.	-	
3.		**Graphs**	5	CO3
	3.1	Introduction to Graphs: Undirected Graph, Directed Graph, Terminology.		
	3.2	Connectivity in Undirected and Directed Graphs, Spanning Tree.		
	3.3	Representation of Graphs: Adjacency Matrix, Adjacency List, Transitive Closure, and Path Matrix.		
	3.4	Traversals: Breadth-First Search (BFS), Depth-First Search (DFS).		
		Self-Learning Topics: Graph Applications in Social Networks, GPS, Topological Sorting, Graph Coloring, Graph Databases (Neo4j).	-	
4.		**Recursion and Storage Management**	5	CO4
	4.1	Recursion: Writing Recursive Functions, Flow of Control, Winding and Unwinding Phases.		
	4.2	Recursive Data Structures, Tail Recursion, Direct and Indirect Recursion.		
	4.3	Storage Management: Sequential Fit Methods (First Fit, Best Fit, Worst Fit).		
	4.4	Fragmentation, Freeing Memory, Boundary Tag Method.		
	4.5	Buddy Systems: Binary Buddy System, Fibonacci Buddy System.		
	4.6	Compaction, Garbage Collection.		
		Self-Learning Topics: Tail Call Optimization, Memory Pools, Garbage Collection Algorithms, Memory Management in Python and Java, Recursion in Dynamic Programming.	-	
5.		**Searching and Sorting**	6	CO5
	5.1	Searching: Sequential Search, Binary Search.		
	5.2	Hashing: Hash Functions (Truncation, Mid-Square, Folding, Division).		
	5.3	Collision Resolution: Open Addressing (Linear Probing, Quadratic Probing, Double Hashing), Separate Chaining.		
	5.4	Sorting: Insertion Sort, Selection Sort, Merge Sort, Quick Sort, Radix Sort.		
		Self-Learning Topics: External Sorting, Consistent Hashing, Sorting in Python, Timsort, Introsort, Hashing in Cryptography and Blockchain, Complexity Analysis.	-	

6.		**Applications of Data Structures**	**6**	**CO6**
	6.1	Applications of Linked Lists: Polynomial Addition and Multiplication.		
	6.2	Applications of Stacks: Reversal of a String, Expression Evaluation, Polish Notation (Infix, Prefix, Postfix).		
	6.3	Applications of Queues: Scheduling, Round Robin Scheduling.		
	6.4	Applications of Trees: Huffman Coding, Heap Sort.		
	6.5	Applications of Graphs: Shortest Path Algorithms (Dijkstra's, Bellman-Ford, Floyd-Warshall).		
	6.6	Minimum Spanning Tree: Prim's Algorithm, Kruskal's Algorithm.		
		Self-Learning Topics: Bloom Filters, Disjoint Set Union (DSU), Data Structures in Blockchain, Real-World Applications of Stacks, Queues, Graphs in ML.	-	
		Total	**39**	

Textbooks:

1. Adam Drozdek, *Data Structures and Algorithms in C/C++*, Cengage Learning.
2. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson.
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, *Introduction to Algorithms*, MIT Press.
4. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, *Data Structures and Algorithms in Python*, Wiley.
5. Reema Thareja, *Data Structures Using C*, Oxford University Press.

Reference Books:

1. Donald E. Knuth, *The Art of Computer Programming, Volume 1: Fundamental Algorithms*, Addison-Wesley.
2. Robert Sedgewick, Kevin Wayne, *Algorithms*, Addison-Wesley.
3. Narasimha Karumanchi, *Data Structures and Algorithms Made Easy*, CareerMonk Publications.
4. Robert Lafore, *Data Structures and Algorithms in Java*, Sams Publishing.
5. Bradley N. Miller, David L. Ranum, *Problem Solving with Algorithms and Data Structures Using Python*, Franklin, Beedle & Associates.

Online References:

1. https://www.geeksforgeeks.org/
2. https://www.tutorialspoint.com/
3. https://www.hackerrank.com/domains/tutorials/10-days-of-algorithms

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class test/ Case study etc. of 15 marks and 5 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus)

End Semester Examination

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

Question paper will comprise of 03 questions.

Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.

Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.

Question3(20 marks) :- Solve any 04 out of 06. All questions carry 05 marks each.

All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC303	Database Management Systems	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
ITC303	Database Management Systems	20	20	60	--	--	100

Pre-requisite: FEC104,FEL103, FEL205: Knowledge of Basic Programming

Program Outcomes addressed :

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions
4. PO4: Conduct investigations of complex problems
5. PO5: Engineering tool usage

Course Objectives:

1. To learn the foundational knowledge and understand the significance of database management systems.
2. To construct Entity Relationship(ER/EER) data model for real world applications
3. To gain a clear understanding of the Relational Data Model and the fundamentals of Relational Algebra.
4. To introduce the fundamentals for efficient data storage and retrieval.
5. To demonstrate the concepts of normalization for efficient database design.
6. To learn the concepts of transaction processing, including concurrency control and recovery mechanisms.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Identify the importance and necessity of a Database Management System (DBMS)
2. Design conceptual data models using ER/EER modeling for real-life applications.
3. Develop a Relational Model for real-world scenarios and formulate queries using Relational Algebra.
4. Formulate SQL queries for data manipulation and retrieval.
5. Implement normalization techniques for efficient relational database design.
6. Explain and demonstrate transaction processing, including concurrency control and recovery mechanisms.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Database System Concepts and Architecture**	**05**	**CO1**
	1.1	**Introduction:** Introduction, Characteristics of Databases, File system v/s Database system, Data Models, Data abstraction and Data Independence.		
	1.2	**Fundamental Concepts :**DBMS system architecture, Users of Database, Database Administrator (DBA), Role of DBA **Self-learning Topics**: Identify the types of Databases such as Object Oriented Database, Graph Database etc.		
2.0		**The Entity- Relationship Model**	**06**	**CO2**
	2.1	**Conceptual Modeling of a database** : The Entity-Relationship (ER) Model, Entity Type, Entity Sets, Attribute Types and Keys, Relationship Types, Relationship Sets, Weak entity Types.		
	2.2	**Extended ER:** Generalization, Specialization and Aggregation, Extended Entity-Relationship (EER) Model. **Self-learning Topics**: Design an ER model for any real time case study.		
3.0		**Relational Model & Relational Algebra**	**08**	**CO3**
3.0	**3.1**	**Relational Model:** Relational Model Constraints and Relational Database Schemas, Concept of Keys: Primary Kay, Secondary key, Foreign Key, Mapping the ER and EER Model to the Relational Model		
	3.2	**Relational Algebra:** Relational Algebra expressions for Unary Relational Operations, Set Theory operations, Binary Relational operation Relational Algebra Queries **Self-learning Topics**: Map the ER model to relational models for different real life application.		

4.0		**Structured Query Language (SQL) & Indexing**	**08**	**CO4**
	4.1	**Overview of SQL:** Data Definition Commands, Set operations, aggregate function, null values, Data Manipulation commands, Data Control commands, Complex Retrieval Queries using Group By, Order By, Recursive Queries, nested Queries, Integrity constraints in SQL.		
	4.2	**Database Programming:** JDBC, Security and authorization: Grant & Revoke in SQL Functions and Procedures in SQL and cursors. **Self-learning Topics**: Design of database for the relational model designed in module III and practice various queries.		
5.0		**Relational Database Design**	**06**	**CO5**
	5.1	**Design guidelines for relational Schema:** Functional Dependencies, Database tables and normalization, The need for normalization		
	5.2	**Normalization process:** Improving the design, Definition of Normal Forms- 1NF, 2NF, 3NF & The Boyce-Codd Normal Form (BCNF). **Self-learning Topics:** Consider any real time application and normalization upto 3NF/BCNF		
6.0		**Transactions Management and Concurrency and Recovery**	**06**	**CO6**
	6.1	**Transaction:** Transaction concept, State Diagram, ACID Properties, Transaction Control Commands, Concurrent Executions, Serializability – Conflict and View serializability.		
	6.2	**Concurrency and Recovery Control**: Lock-based-protocols, Timestamp-based protocols, Deadlock handling Recovery Concepts, Log based recovery. **Self-learning Topics**: Study the various deadlock situations which may occur for a database designed in module 5.		
		Total	**39**	

Textbooks:

1. Korth, Slberchatz, Sudarshan, Database System Concepts, 6th Edition, McGraw Hill
2. Elmasri and Navathe, Fundamentals of Database Systems, 6th Edition, Pearson education

3. Raghu Ramkrishnan and Johannes Gehrke, Database Management Systems, TMH
4. G. K. Gupta: "Database Management Systems", 3rd Edition, McGraw – Hill, 2018.

Reference books:

1. Peter Rob and Carlos Coronel, ― Database Systems Design, Implementation and Management‖,Thomson Learning, 9th Edition.
2. Dr. P.S. Deshpande, "SQL and PL/SQL for Oracle 11g Black Book", Dreamtech Press, 2012.
3. Lynn Beighley, "Head First SQL", O'Reilly Media, 2007.

Online References:

1. https://www.nptel.ac.in
2. https://www.coursera.org/
3. https://www.oreilly.com

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 5 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

Question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks) :- Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC304	Automata Theory	02	-	1	02	-	1	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
ITC304	Automata Theory	15	15	45	25	--	100

Pre-requisite:

Basic Mathematical Fundamentals: Sets, Logic, Relations, Functions.

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct Investigation of Complex problems
5. PO5: Engineering Tool Usage
6. PO11: Lifelong Learning

Course Objectives:

1. To understand the foundations of computation including Automata Theory.
2. To construct models of regular expressions and languages.
3. To understand the relation between Regular Languages, Contexts free Languages, PDA and TM.
4. To learn how to design PDA as Acceptor.
5. To learn how to design TM as Calculators.
6. To understand Applications of Automata Theory.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Design automata theory using Finite Automata
2. Show regular expressions for any pattern.
3. Analyze and design Context Free languages and Grammars.
4. Design different types of Push down Automata as Simple Parser.
5. Design different types of Turing Machines as Acceptor, Verifier, Translator and Basic computing machine.
6. Analyze the Limitations and Extensions of Automata.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Finite Automata**	**09**	**CO1**
	1.1	Alphabets, Strings, Languages, Finite State machine (FSM), Need for automata theory, Introduction to formal proof of Finite Automata (FA)		
	1.2	FSM without Output (Acceptor/Recognizer) - Deterministic Finite Automata (DFA), Non-deterministic Finite Automata (NFA), Finite Automata with Epsilon transitions, Equivalence of NFA and DFA, Equivalence of NFAs with and without ε-moves, Minimization of DFAs.		
	1.3	FSM with output- Moore Machine and Mealy machine, Equivalence -conversion of Moore and mealy, conversion of Mealy and Moore machine. **Self-Learning Topics:** Two-way finite Automata		
2.0		**Regular Expressions and Regular Languages**	**06**	**CO2**
	2.1	Regular Languages- Closure properties and Decision properties of regular languages, Regular expression – Equivalence of Finite Automata and regular expressions (FA to RE, RE to FA), Regular Grammar-RL and LL Grammar. **Self-Learning Topics:** Applications of Regular Grammar in compiler Design		
3.0		**Context Free Grammar**	**07**	**CO3**
	3.1	Context Free Languages-Types of Grammar, Chomsky 's hierarchy of languages, Context-Free Grammar (CFG) and Languages, Derivations and Parse trees, Ambiguity in grammars and languages, Simplification and Normalization- (CNF&GNF) Closure properties of Context Free Languages. **Self-Learning Topics:** Applications of CFG in NLP and Programming language Design		
4.0		**Push Down Automata**	**07**	**CO4**
	4.1	Push Down Automata (PDA): Definition, Moves, Instantaneous descriptions, Languages of pushdown automata. Equivalence of pushdown automata and CFG-CFG to PDA,PDA to CFG –Deterministic Pushdown Automata. **Self-Learning Topics:** Non-Deterministic Push Down Automata		

5.0		**Turing Machine**	**07**	**CO5**
	5.1	Turing Machine: Basic model, definition and representation – Instantaneous Description, Language acceptance by TM-Recursive and Recursive Enumerable Languages, Deterministic TM, Variants of TM, Halting Problem **Self-Learning Topics:** Undecidable Problems		
6.0		**Applications of Automata**	**03**	**CO6**
	6.1	Applications of Various Automata- FA, PDA, TM Limitations of Various Automata -FA, PDA, TM Real Time applications of Automata Theory **Self-Learning Topics:** Automata Theory in AI-NLP, Robotics and planning.		
		Total	**39**	

Textbooks:

1. John C Martin, "Introduction to Languages and the Theory of Computation", 4th Edition, Tata McGraw Hill, 2011.
2. Elaine Rich,"Automata, Computability and Complexity:Theory and Applications."
3. V. Aho, R. Shethi, Monica Lam, J.D. Ulman, "Compilers Principles, Techniques and Tools", Pearson Education.

Reference books:

1. K.L.P.Mishra and N.Chandrasekaran, "Theory of Computer Science: Automata Languages and Computation", 3rd Edition, Prentice Hall of India, 2006.
2. Daniel I.A. Cohen, "Introduction to Computer Theory", John Wiley & Sons.
3. Vivek Kulkarni," Theory of Computation", Oxford University

Online References:

1. https://onlinecourses.nptel.ac.in/noc19_cs79/preview
2. https://online.stanford.edu/courses/cs154-introduction-theory-computation
3. https://ocw.mit.edu/courses/18-404j-theory-of-computation-fall-2020/pages/lecture-notes/

Course Assessment:
ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 10 marks and 05 marks attendance.

MSE: To be conducted as written examination for 15 marks (on 50% syllabus)

Term Work:

Term Work should consist of at least 05 assignments (one assignment on each module).
Assignment -20 marks and Attendance 5 marks

End Semester Examination:

$ ESE of duration 2 hours are of 60 marks and scaled to 45.

Question paper will comprise of 3 questions.
Question1(15 marks): - Solve any 03 out of 04. All questions carry 05 marks each.
Question 2 (30 marks): - Solve any 03 out of 05. All questions carry 10 marks each.
Question3(15 marks) :- Solve any 03 out of 04. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC305	Computer Organization and Architecture	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
ITC305	Computer Organization and Architecture	20	20	60	--	--	100

Prerequisite:

FEC204 -Digital System Design

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/ development of solutions
4. PO4: Conduct investigations of complex problems
5. PO11: Lifelong Learning

Course Objectives:

Students will

1. Conceptualize the basics of organizational and architectural issues of a digital computer.
2. Analyze processor performance improvement using instruction level parallelism.
3. Learn the function of each element of a memory hierarchy.
4. Study various data transfer techniques in digital computer.
5. Articulate design issues in the development of processor or other components that satisfy design requirements and objectives.
6. Learn microprocessor architecture and study assembly language programming.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Describe basic organization of computer and the architecture of 8086 microprocessor.
2. Construct assembly language program for given task for 8086 microprocessor.
3. Demonstrate control unit operations and conceptualize instruction level parallelism.
4. Represent integer, real numbers and perform computer arithmetic operations on integer numbers.
5. Categorize memory organization and explain the function of each element of a memory hierarchy.
6. Identify and compare different methods for computer I/O mechanisms.

Module No.	Unit No.	Topics	Hrs.	CO
1.0		**Overview of Computer Architecture & Organization**	07	CO1
	1.1	Introduction of Computer Organization and Architecture. Basic organization of computer and block level description of the functional units. Evolution of Computers, Von Neumann model. Performance measure of Computer Architecture.		
	1.2	Architecture of 8086 microprocessor, Pin description of 8086 microprocessor, Minimum mode & Maximum mode of Operation.		
		Self-Learning Topics- Memory Banking of 8086 microprocessor		
2.0		**Programming 8086 Microprocessor**	07	CO2
	2.1	Addressing modes, Instruction Set, Assembly Language Programming,		
	2.2	Programs based on Loops, Strings, Procedures, Macros etc.		
		Self-Learning Topics - Interfacing of I/O devices with 8086(8255, ADC,DAC).		
3.0		**Processor Organization and Architecture**	07	CO3
	3.1	**CPU Architecture**: Register Organization, Instruction formats, basic instruction cycle. Instruction interpretation and sequencing.		
	3.2	**Control Unit:** Soft wired (Micro-programmed) and hardwired control unit design methods. Microinstruction sequencing and execution. Micro operations, concepts of nano programming. Introduction to parallel processing concepts, Flynn's classifications, pipeline processing, instruction pipelining, pipeline stages, pipeline hazards.		
		Self-Learning Topics -Study the examples on instruction pipelining for practice, Architecture of ARM processor		
4.0		**Data Representation and Arithmetic Algorithms**	06	CO4
	4.1	**Number representation:** Binary Data representation, two's complement representation. Basics of floating-point representation IEEE 754 floating point (Single & double precision) number representation.		
	4.2	**Integer Data arithmetic**: Addition, Subtraction. Multiplication: Unsigned & Signed multiplication- Add & Shift Method, Booth's algorithm. Division of integers: Restoring and non-restoring division		
		Self-Learning Topics - Signed division, Implement Booth's Algorithm and Division methods.		

5.0		**Memory Organization**	**08**	**CO5**
	5.1	**Introduction to Memory and Memory parameters:** Classifications of primary and secondary memories. Types of RAM and ROM, Allocation policies, Memory hierarchy and characteristics.		
	5.2	**Cache memory:** Concept, architecture (L1, L2, L3), mapping techniques. Cache Coherency, Interleaved and Associative memory.		
		Self-Learning Topics - Case study on Memory Organization, Numerical on finding EAT, Address mapping.		
6.0		**I/O Organization**	**04**	**CO6**
	6.1	**Input/output systems:** I/O modules and 8089 I/O processor.		
	6.2	**Types of data transfer techniques**: Programmed I/O, Interrupt driven I/O and DMA.		
		Self-Learning Topics: I/O Bus and Data Transfer Mechanisms, Modern Trends in I/O Systems		
		Total	**39**	

Text Books:

1. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, “Computer Organization”, Fifth Edition, Tata McGraw-Hill.
2. William Stallings, “Computer Organization and Architecture: Designing for Performance”, Eighth Edition, Pearson.
3. 8086/8088 family: Design Programming and Interfacing: By John Uffenbeck (Pearson Education)
4. Microprocessor and Interfacing: By Douglas Hall (TMH Publication).

Reference Books:

1. B. Govindarajulu, “Computer Architecture and Organization: Design Principles and Applications”, Second Edition, Tata McGraw-Hill.
2. Dr. M. Usha, T. S. Srikanth, “Computer System Architecture and Organization”, First Edition, Wiley-India.
3. John P. Hayes, “Computer Architecture and Organization”, McGraw-Hill.,Third Edition.
4. K Bhurchandi, “Advanced Microprocessors & Peripherals”, Tata McGraw-Hill Education

Online References:

1. https://www.coursera.org/learn/build-a-computer
2. https://onlinecourses.nptel.ac.in/noc23_cs113/preview
3. https://onlinecourses.swayam2.ac.in/ntr25_ed43/preview

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 05 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus).

End Semester Examination (ESE):

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

Question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC306	Engineering Economics	02	-	-	02	-	-	02

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
ITC306	Engineering Economics	50		--	--	--	50

Pre-requisite :
Principles of Basic Mathematics

Program Outcomes addressed:

1. PO1: Engineering knowledge.
2. PO2: Problem analysis.
3. PO10: Project Management and Finance.

Course Objectives:

1. To introduce students to the basic principles of economics and their application to engineering decision-making.
2. To explore the Role of Trade in a Modern Economy
3. To develop student’s analytical skills in assessing consumer behavior and the determinants of demand and supply across different market structures, including price elasticity.
4. To enable students to understand cost analysis, pricing, project evaluation.
5. To develop the ability to make informed decisions regarding engineering projects based on economic criteria.
6. To understand the Concept of Interest Rates and Their Role in the Economy

Course Outcomes: Upon completion of this course, learners will be able to…

1. Define the basic concept of micro and macroeconomics, engineering economics and their application in engineering economy.
2. Define and Explain the Concept of Market in the Modern Economy.
3. Evaluate the effects of changes in demand and supply on price determination of products and services.
4. Analyze the costs and benefits of various engineering solutions.
5. Develop the ability to account for time value of money using engineering economy factors and formulas.
6. Understand market dynamics and pricing strategies in different industrial sectors.

Module No.	Unit No.	Topics	Hrs.	CO
1.0		**Introduction to Economics**	03	CO1
	1.1	**Economics** - Nature, Scope, Basic problems of an economy, Micro Economics and Macro Economics, The three problems of Economics Organization. Introduction to Engineering Economics.		
		Self-Learning: Basic Economic Concepts: Cost, Benefit, Profit.		
2.0		**Market and Government in Modern Economy**	03	CO2
	2.1	**Modern Economy** - Market Definition, How market solve three economics problems, Trade, Money & Capital, The economic role of Government.		
		Self-Learning: Market Economy vs. Planned Economy, The Role of Private vs. Public Sectors		
3.0		**Supply, Demand and Product market**	06	CO3
	3.1	**Basic Elements of Supply and Demand -** The determination of Demand and Supply, The Demand Schedule, The Supply Schedule, Equilibrium of supply and demand. Application of Supply and Demand.		
	3.2	**Elasticity of Demand and Supply -** Price elasticity of Demand, Elasticity and Revenue, Price elasticity of Supply.		
	3.3	**Demand and Consumer behavior** - Choice and utility theory, Equimarginal principle, An alternative approach: substitution effect and income effect, From Individual to market demand.		
		Self-Learning: Case Study on demand and supply.		
4.0		**Production and Cost Theory**	05	CO4
	4.1	**Production** - Production function, Laws of returns: Law of variable proportion, Law of returns to scale.		
	4.2	**Cost and Revenue Concepts** - Total Costs, Fixed cost, Variable cost, Average cost and Marginal cost, The Link between production and costs, Analysis of cost minimization.		
		Self-Learning: Read case studies about businesses optimizing their production costs and making strategic production decisions.		
5.0		**Time value and Project evaluation with money**	04	CO5
	5.1	**Time Value of Money -** Interest - Simple and compound, nominal and effective rate of interest, Cash flow diagrams, Principles of economic equivalence.		
	5.2	**Evaluation of Engineering Projects** -Present worth method, Future worth method, Annual worth method, Internal rate of return method.		
		Self-Learning: Learn to use financial calculators or Excel		

		functions for quick calculations of TVM.		
6.0		**Money, Banking and Financial Markets**	**05**	**CO6**
	6.1	**Money and Interest Rates -** The Evolution of Money, Functions of Money, Interest rates, Price of Money, Demand for money.		
	6.2	**Banking and the supply of money** - Banking definition, Types of Banks, Banking as as a business, The process of Deposits creations.		
	6.3	**Financial Economics** - Financial assets, Risk and return on different assets, The stock market, Personal financial strategies.		
		Self learning: The evolution of financial market.		
		Total	**26**	

Textbooks:

1. Paul A. Samuelson and William D. Nordhaus, “Economics”, Tata McGraw Hill, 20th edition, 2019.
2. "Engineering Economy" by Leland Blank and Anthony Tarquin, 9th edition, 2024.

Reference books:

1 "Introduction to Engineering Economics" by John V. O'Connor, 5th edition, 2013.

2 W S Jawadekar, “Management Information Systems”, Tata McGraw Hill, 6th edition, 2020.

3 "Fundamentals of Engineering Economics" by Chan S. Park, 4th edition, 2018.

Online References:

1. https://www.mheducation.com/highered/product/Engineering-Economy-Blank.html.
2. https://archive.org/details/engineeringecono0000blan_t5b6.

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 25 marks .
ISE 25 marks = 05 marks for attendance + 20 marks for activities.

MSE: To be conducted as written examination for 25 marks (on 50 % syllabus).

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL301	Data Structure and Analysis Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITL301	Data Structure and Analysis Lab	--	--	--	25	25	50

Pre-requisite:

1. FEL103, FEL205: Knowledge of Basic Programming

Program Outcome mapped:

1. PO1: Engineering Knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.

Lab Objectives:

1. To develop hands-on skills in implementing and manipulating linear data structures such as stacks, queues, and linked lists.
2. To gain practical experience in implementing and analysing tree structures, including binary trees, binary search trees, and AVL trees.
3. To implement graph representations and traversal algorithms and apply them to solve problems like shortest path and minimum spanning tree.
4. To practice writing recursive functions and implementing storage management techniques such as memory allocation and garbage collection.
5. To implement and analyse searching and sorting algorithms and understand their applications in real-world scenarios.
6. To apply data structures and algorithms to solve real-world problems through mini-projects and case studies.

Lab Outcomes:

After successful completion of the course, student will be able to…

1. Implement and manipulate stacks, queues, and linked lists, and demonstrate their applications in problem-solving.
2. Implement and analyse tree structures, including binary search trees and AVL trees, and demonstrate their use in data organization.

3. Represent graphs using adjacency matrices and lists, and implement traversal algorithms (BFS, DFS) and shortest path algorithms (Dijkstra's).
4. Write recursive functions and implement storage management techniques such as memory allocation and garbage collection.
5. Implement and analyse searching algorithms (binary search, hashing) and sorting algorithms (quick sort, merge sort).
6. Apply data structures and algorithms to solve real-world problems through mini-projects and case studies.

Suggested List of Experiments

Module	Lab Activity	Mapped to LO
0	- Write programs to demonstrate the use of structures and pointers. - Implement basic operations using structures.	
1	- Implement stack operations (push, pop, peek) using arrays. - Implement queue operations (enqueue, dequeue) using arrays. - Implement circular queue and priority queue.	LO1
2	- Implement singly linked list operations (insertion, deletion, reversal). - Implement doubly linked list operations (insertion, deletion). - Implement circular linked list operations (insertion, deletion).	LO1
3	- Implement binary tree traversals (preorder, inorder, postorder). - Implement binary search tree operations (insertion, deletion, searching). - Implement AVL tree operations (insertion, deletion, rotations).	LO2
4	- Implement graph representation using adjacency matrix and adjacency list. - Implement BFS and DFS traversal. - Implement Dijkstra's shortest path algorithm.	LO3
5	- Write recursive functions for factorial and Fibonacci. - Implement hash tables with collision resolution (linear probing, chaining).	LO4
6	- Implement sorting algorithms (quick sort, merge sort). - Implement searching algorithms (binary search, hashing).	LO5
7	- Implement polynomial addition using linked lists. - Implement expression evaluation using stacks. - Implement Huffman coding using trees.	LO6

Textbooks:

1. Michael T. Goodrich, Roberto Tamassia, David M. Mount, *Data Structures and Algorithms in C++*, Wiley.
2. Michael T. Goodrich, Roberto Tamassia, *Data Structures and Algorithms in Java*, Wiley.
3. Benjamin Baka, *Python Data Structures and Algorithms*, Packt Publishing.
4. Yedidyah Langsam, Moshe J. Augenstein, Aaron M. Tenenbaum, *Data Structures Using C and C++*, Pearson.

Reference Books:

1. Robert Sedgewick, *Algorithms in C++*, Addison-Wesley.
2. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, *Data Structures and Algorithms in Python*, Wiley.

3. Robert Lafore, *Data Structures and Algorithms in Java*, Sams Publishing.

Web Links:

1. https://www.geeksforgeeks.org/
2. https://www.hackerrank.com/domains/tutorials/10-days-of-algorithms
3. https://leetcode.com/
4. https://www.codecademy.com/

Assessment and Evaluation

Term Work:

The term work should include 10 experiments covering all Lab Outcomes. At least 02 assignments covering the entire syllabus based on modules of subject **ITC302,** and practical of “Data Structures and Algorithm Lab”. The assignments should be students’ centric, and an attempt should be made to make assignments more meaningful, interesting and innovative. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks:

25 Marks (Total Marks) =15 Marks (Experiment) + 05 Marks (Assignments) + 05 Marks (Attendance)

The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

End-semester Practical and oral exam will be held based on the above and modules of subject ITC302 syllabus and will be conducted as End Semester Examination Practical (ESEP).

SIES Graduate School of Technology

Department of Information Technology

Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL302	SQL Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITL302	SQL Lab	--	--	--	25	25	50

Pre-requisite:

FEC104,FEL103, FEL205: Knowledge of Basic Programming

Program Outcomes addressed:

1. PO1: Engineering Knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.

Lab Objectives:

1. To identify, define, and model real-life application problems using conceptual data models.
2. To design and construct relational models from ER/EER diagrams for real-life applications.
3. To demonstrate usage of relational algebra operations.
4. To apply SQL to store and retrieve data efficiently
5. To implement database connectivity using JDBC
6. To understand the concepts of transaction processing- concurrency control & recovery procedures.

Lab Outcomes:

After successful completion of this course, the students will be able to:

1. Define problem statement and Construct the conceptual model for real life application.
2. Design ER/EER and convert it into relational model and incorporate relational algebra operations.
3. Create and populate a RDBMS using SQL and write SQL queries for efficient information retrieval.
4. Apply view, triggers and procedures to demonstrate specific event handling.
5. Demonstrate database connectivity using JDBC.
6. Determine the need for advance databases.

Detailed Lab Syllabus

Module No	Topics to be Covered	Mapped to LO
1	**Conceptual Data Modeling** - Entity-Relationship (ER) Model: Entities, attributes, relationships - Extended ER (EER) Model: Specialization, generalization, aggregation - Steps to design an ER/EER diagram. - Conversion of ER/EER diagrams to relational schema	LO1,LO2
2	**SQL Data Definition and Manipulation** - SQL syntax and basic commands -Data Definition Language (DDL): CREATE, ALTER, DROP - Data Manipulation Language (DML): INSERT, UPDATE, DELETE - Data Query Language (DQL): SELECT with WHERE, ORDER BY, GROUP BY, HAVING , -integrity constraints	LO3
3	**Advanced SQL Queries , Database Security** - Joins: INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN - Subqueries and nested queries - Set operations: UNION, INTERSECT, EXCEPT **Database Security and Authorization** - Database security concepts - SQL authorization: GRANT and REVOKE - Roles and privileges	LO3
4	**Views, Triggers, and Stored Procedures** - Views: Creation, updating, and deletion - Triggers: Definition, types, and usage - Stored procedures: Creation and execution	LO4
5	**Database Connectivity using JDBC** - Introduction to JDBC - JDBC architecture and components - Connecting to a database using JDBC	LO5
6	**Advance Databases** NoSql Databases , Mobile Databses, Cloud databases	LO6

Suggested List of Experiments:		
Sr. No.	**Title of Experiments**	**LO Mapped**
1	Identify a real-world problem and formulate a problem statement. Design an Entity-Relationship (ER) / Extended Entity-Relationship (EER) Model for the proposed	LO1

	system.	
2	Convert the ER/EER model into a Relational Schema.	LO2
3	Create a database using DDL and enforce integrity constraints.	LO2, LO3
4	Perform data manipulation operations (DML) on the populated database.	LO2, LO3
5	Implement authorization mechanisms using GRANT and REVOKE.	LO2, LO3
6	Develop and execute basic and complex SQL queries.	LO3
7	Implement Views and Triggers to enhance database functionality.	LO4
8	Establish database connectivity using JDBC for application integration.	LO4
9	Apply Transaction Control Language (TCL) commands for managing transactions.	LO5
10	Create and execute user-defined functions and stored procedures in SQL.	LO5
11	Creating and Managing NoSql Database using MongoDB.	LO6

Reference books:

1. Peter Rob and Carlos Coronel, ― Database Systems Design, Implementation and Management‖,Thomson Learning, 9th Edition.
2. Dr. P.S. Deshpande, “SQL and PL/SQL for Oracle 11g Black Book”, Dreamtech Press, 2012.
3. Lynn Beighley, “Head First SQL”, O'Reilly Media, 2007.

Online References:

1. https://www.nptel.ac.in
2. https://www.coursera.org/
3. https://www.w3schools.com/

Term Work:

Term Work shall consist of at least 10 Practical’s based on the above list, but not limited to. Also, Term work Journal must include at least 2 assignments. The first assignment may be based on: Relational Algebra and Second may be based on Transactions.

Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks:

25 Marks (Total Marks) =15 Marks (Experiment) + 05 Marks (Assignments) + 05 Marks (Attendance)

The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

End-semester Practical and oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP)

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL303	Skill Lab (Python Lab)	--	2*+2	--	--	02	--	02
* Theory class to be conducted for full class								

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITL303	Skill Lab (Python Lab)	--	--	--	25	25	50

Prerequisite:

Knowledge of C and Java programming

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct Investigation of Complex problems
5. PO5: Engineering Tool Usage
6. PO6: The Engineer and The World
7. PO8: Ethics
8. PO11: Lifelong Learning

Lab Objectives:

1. To learn the fundamentals of python
2. To learn Functions and Data structures
3. To learn and Understand Object oriented Programming concepts in python
4. To study concepts of Concepts of packages, multithreading and exception handling
5. To Study the concepts of File handling, GUI & database programming
6. To Understand Data visualization using Matplotlib and Data analysis using Pandas

Lab Outcomes: At the end of the course, the students should be able to

1. Interpret and apply the fundamentals of python.
2. Implement functions and data structures concepts in python.
3. Apply Object oriented Programming concepts in python for problem solving.
4. Illustrate the Concepts of packages, multithreading and exception handling.
5. Outline the concepts of File handling, and Design GUI with database connectivity.
6. Formulate Data visualization using Matplotlib and Data analysis using Pandas.

Module		Detailed Content	Hours	LO Mapped
0		**Prerequisite**	01	
		Study Different Python IDE(VScode, PyCharm, Atom, IDLE), installation of any one IDE , Environment setup		
1		**Fundamentals of Python**	04	LO1
	1.1	Python Features, Python building blocks: Identifiers, Keywords, Indention, Variables and Comments, Basic data types, Operators, Control flow statements: Conditional statements, Looping in Python (while loop, for loop, nested loops) Loop manipulation -continue, pass, break, Input/output Functions, Decorators, Iterators and Generators. Advanced Data Types-List, Tuples, dictionaries, set, string. **Self-Learning Topic:** Implement Programs based on nested loops, pattern based		
2		**Arrays, Functions and Data structures**	06	LO2
	2.1	Arrays: Single dimensional Arrays,Multi-dimensional Arrays using NumPy: Mathematical operations, Matrix operations, aggregate and other Built-in functions. Functions: Built-in functions in python, defining function, calling function, returning values, passing parameters, Nested and Recursive functions. **Self-Learning Topic:** Anonymous Functions (Lambda, Map, Reduce, Filter)		
	2.2	Data Structures-Link List, Stack, Queues, Dequeues **Self-Learning Topic:** Implement Graph data structure using dictionary		
3		**Object Oriented Programming**	03	LO3
	3.1	Object-oriented programming concepts, Creating Classes and Objects, Constructors. Inheritance: Types of Inheritance, Super()method, Method overloading, Method overriding, Abstract class, Abstract method. **Self-Learning Topic:** Interface/Wrapper class		
4		**Packages, multithreading and exception handling**	04	LO4
	4.1	Packages: creating user defined packages and importing packages. Multi-threading: Thread and Process, starting a thread, threading module, Synchronizing threads. Exception handling: Compile time errors, Runtime errors, exceptions, types of exception, try statement, except block, User-Defined Exceptions. **Self-Learning Topic:** Implement deadlock program		
5		**File handling, GUI & database programming**	04	LO5
	5.1	File Handling: File operations (read, write, rename, delete), file methods, File Exceptions. Graphical user interface (GUI): different GUI tools in python (Tkinter, PyQt,), Working with containers, Canvas, Frame, Widgets (Button, Label, Text, Scrollbar, Check button, Radio button, Entry, Spinbox, Message etc.) Database connectivity: to perform CRUD operations. (SQLite, MySQL, Oracle, PostgreSQL etc.) **Self-Learning:** GUI frameworks Kivy, PySimpleGUI, Libavg		

6		**Data visualization and analysis**	**04**	**LO6**
	6.1	Visualization using Matplotlib: Matplotlib with Numpy, working with plots (line plot, bar graph, histogram, scatter plot, area plot, pie chart etc.) Data manipulation and analysis using Pandas: Introduction to Pandas, importing data into Python, series, data frames, indexing data frames, basic operations with data frame, filtering, combining and merging data frames, Removing Duplicates.		
	6.2	Web Frameworks for Python, Popular Full-Stack Frameworks-Django, Flask, Fast API. **Self-learning:** Framework Types (Full stack, Micro, Asynchronous).		
		Total	**26**	

Hardware & Software Requirements

Hardware Requirements	**Software Requirements**	**Other Requirements**
PC With following Configuration 1. Intel Dual core Processor or higher 2. Minimum 2 GB RAM 3. Minimum 40 GB Hard disk 4. Network interface card	1.Windows or Linux Desktop OS 2. Python 3.6 or higher 3. Notepad ++ 4.Python IDEs like IDLE, Pycharm, Pydev, Netbeans or Eclipse, VScode 5. Mysql ,Oracle,PostgreSQL	1. Internet Connection for installing additional packages if required /Explore any advanced concept to learn

Textbooks:

1 Dr. R. Nageswara Rao,” Core Python Programming”, Dreamtech Press, Wiley Publication
2 M. T. Savaliya , R. K. Maurya, “Programming through Python”, StarEdu Solutions.
3 E Balagurusamy, “Introduction to computing and problem-solving using python”, McGraw Hill Publication.

Reference Books:

1 Zed A. Shaw, “Learn Python 3 the Hard Way”, Zed Shaw's Hard Way Series.
2 Martin C. Brown,” Python: The Complete Reference”, McGraw-Hill Publication.
3 Paul Barry,” Headfirst Python”, 2nd Edition, O'Reilly Media, Inc.

Online Resources:

1. https://www.python.org/about/gettingstarted/
2. https://www.learnpython.org/
3. https://www.kaggle.com/learn/python
4. https://www.pythontutorial.net/
5. https://www.datacamp.com/tutorial/data-structures-python
6. https://www.geeksforgeeks.org/python-programming-language-tutorial/
7. https://onlinecourses.nptel.ac.in/noc24_cs57/preview

Suggested List of Programming Assignments/laboratory Work:		
Sr. No.	**Name of the Experiment**	**LO Mapped**
1	Write python programs to understand a) Basic data types, Operators, expressions and Input Output Statements b) Control flow statements: Conditional statements (if, if…else, nested if) c) Looping in Python (while loop, for loop, nested loops) d) Decorators, Iterators and Generators.	LO1
2	Write python programs to understand a) List, Tuples and dictionaries data types and built in methods b) Set and String Data types with its built-in functions	LO1
3	Write python programs to understand a) Basic Array operations using Numpy library b) Implement User defined Functions	LO2
4	Implement Stack and Queue data structure using list	LO2
5	Write Python program to define Class, Object and include Constructor also.	LO3
6	Write python program to implement a) Different types of Inheritance b) Polymorphism using Operator overloading, Method overloading, Method overriding c)Abstract class, Abstract method in Python	LO3
7	Write python program to understand a) Creating User-defined modules/packages and import them in a program b) Create a multithreaded application with thread and synchronization concept.	LO4
8	Creating a menu driven application which should cover all the built-in exceptions in python.	LO4
9	Write python program to understand different File Handling operations.	LO5
10	Design a Registration page by Graphical user interface (GUI) using built-in tools in python (Tkinter/PyQt)	LO5
11	Implement GUI database connectivity to perform CRUD operations in python (Use any one database like SQLite, MySQL, Oracle, PostgreSQL etc.)	LO5
12	Write python programs to implement different types of plots using NumPy and Matplotlib	LO6
13	Write Program to demonstrate Data Series and Data Frames using Pandas	LO6
*	**Content Beyond Syllabus Experiment** Learn key Pandas SQL operations, including reading and writing data between Pandas and SQL databases, and handling data types effectively.	

Term Work:

1 Term work should consist of minimum10 experiments by covering all main concepts given in syllabus and Journal must include at least 2 assignments.

2 The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

3 Term work is assessed as **Continuous Internal Assessment Practical (CIAP)**.

4 Total 25-Marks (Experiments: 15-marks, Attendance: 05-marks,Assignment: -5-marks)

Oral & Practical exam

Based on the entire syllabus End Semester Examination Practical (ESEP) will be conducted.

End-Semester Practical and Oral Exam (ESEP): 25 Marks (2 hours).

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITM301	Mini Project 1A (Python)	--	02#	--	--	1	--	1

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITM301	Mini Project 1A (Python)	--	--	--	25	25	50

# Indicates workload of a learner (Not Faculty) for Mini Project 1A. Faculty Load: 1 hour per week per four groups.

Pre-requisite:

1. FEL103, FEL205: Knowledge of Basic Programming

Program Outcome mapped:

1. PO1: Engineering Knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.
7. PO7: Ethics.
8. PO8: Individual and Collaborative Teamwork.
9. PO9: Communication.
10. PO10: Project Management and Finance.
11. PO11: Life-Long Learning.

Course Objectives:

1. To acquaint students with the process of identifying societal or research needs and converting them into computational problems.
2. To familiarize students with Python programming for problem-solving and data analysis.
3. To develop skills in applying Python libraries and frameworks to solve real-world problems.
4. To inculcate self-learning and research capabilities in Python-based projects.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Identify and Analyze computational problems based on societal, research, or industry needs and apply Python programming skills to develop appropriate solutions.
2. Demonstrate teamwork and interpersonal skills by collaborating effectively on Python-based projects, showcasing leadership and coordination abilities.
3. Analyze and Interpret data using Python libraries (e.g., NumPy, Pandas, Matplotlib) to derive meaningful inferences and support decision-making.
4. Design and develop sustainable and innovative solutions using Python while adhering to standard coding practices and documentation guidelines.
5. Communicate Python-based solutions effectively through structured written reports, presentations, and demonstrations.
6. Demonstrate self-learning and adaptability by exploring and integrating new Python tools, libraries, and frameworks for lifelong learning and professional growth.

Guidelines for Mini Project 1A (Python):

- **Group Formation:** 3-4 students per group.
- **Problem Identification:** Identify a societal or research need and convert it into a Python-based problem statement.
- **Implementation Plan:** Prepare a Gantt/PERT/CPM chart for weekly activities.
- **Log Book:** Maintain a log book to record weekly progress, verified by the supervisor.
- **Self-Learning:** Focus on self-learning with minimal guidance from the supervisor.
- **Solution Development:** Propose multiple solutions, select the best one, and implement it using Python.
- **Validation:** Validate the solution with proper justification and compile a report as per University of Mumbai guidelines.
- **Duration:** The project can span one or two semesters based on complexity.

General Guidelines for Both Mini Projects:

- **Review Committee:** A committee will evaluate progress through two reviews per semester.
- **Term Work Marks Distribution:**
  - Log Book (Guide/Supervisor): 10 marks
  - Review Committee: 10 marks
  - Quality of Project Report: 5 marks
- **Final Assessment:** Presentation and demonstration of the working model to a panel of internal and external examiners.
- **Publication:** Students are encouraged to publish their work in conferences or student competitions.

This syllabus ensures that students gain hands-on experience in Python and Web Technology while addressing real-world problems, fostering innovation, teamwork, and lifelong learning.

Term Work Marks Distribution:

1. Log Book (Guide/Supervisor): 10 Marks

Criteria	Marks Allocation	Description
Problem Identification	**2 Marks**	- Clarity and relevance of the problem statement.
		- Alignment with societal, research, or environmental needs.
Implementation Plan	**2 Marks**	- Quality of Gantt/PERT/CPM chart and planning.
		- Realistic and well-structured timeline for activities.
Weekly Progress and Updates	**3 Marks**	- Regular updates in the log book with detailed progress.
		- Supervisor verification of weekly activities.
Self-Learning and Adaptability	**3 Marks**	- Evidence of self-learning and minimal guidance from the supervisor.
		- Exploration of new tools, technologies, or frameworks.

2. Review Committee: 10 Marks

Criteria	Marks Allocation	Description
Innovation and Creativity	**2 Marks**	- Innovativeness in proposing web-based solutions.
		- Creativity in addressing the problem.
Functionality and Responsiveness	**3 Marks**	- Functionality of the web application (features, usability, and performance).
		- Responsiveness across devices (desktop, tablet, mobile).
Use of Web Technologies	**3 Marks**	- Effective use of web development tools, frameworks, and technologies.
		- Adherence to standard coding practices and norms.
Societal Impact and Sustainability	**2 Marks**	- Analysis of societal and environmental impact.
		- Incorporation of sustainable practices in the solution.

3. Quality of Project Report: 5 Marks

Criteria	Marks Allocation	Description
Clarity and Completeness	**2 Marks**	- Clear and detailed explanation of the problem, solution, and implementation.
		- Adherence to University of Mumbai guidelines for project reports.
Documentation and Communication	**2 Marks**	- Quality of written documentation (structure, grammar, and technical accuracy).
		- Effective use of diagrams, charts, and visuals.
Oral Presentation	**1 Mark**	- Clarity and confidence during the presentation.
		- Ability to answer questions and justify the solution.

Summary of Marks Distribution:

Component	**Mark**
Log Book (Guide/Supervisor)	10
Review Committee	10
Quality of Project Report	5
Total	**25**

Key Features:

1. **Log Book (10 Marks)**: Focuses on problem identification, planning, progress tracking, and self-learning.
2. **Review Committee (10 Marks)**: Evaluates innovation, functionality, use of technologies, and societal impact.
3. **Project Report (5 Marks)**: Assesses the quality of documentation, clarity, and oral presentation.

ESEP Evaluation Criteria for Project Presentation (25 Marks)

Criteria	**Marks Allocation**	**Description**
Problem Identification & Analysis	**5 Marks**	- Clarity and relevance of the problem statement. - Justification of the problem's significance in societal, research, or industry needs.
Technical Implementation & Solution	**5 Marks**	- Use of Python programming concepts and frameworks. - Efficiency and correctness of the solution.
Innovation & Creativity	**3 Marks**	- Novelty and uniqueness of the solution. - Creative approach in problem-solving.
Functionality & Usability	**3 Marks**	- Working model demonstration with minimal errors. - Usability, efficiency, and user experience of the project.
Presentation & Communication Skills	**5 Marks**	- Clarity, confidence, and structure of the presentation. - Effective demonstration and ability to answer queries.
Project Report Quality	**2 Marks**	- Well-structured documentation with appropriate diagrams, charts, and visuals.
Societal Impact & Sustainability	**2 Marks**	- Explanation of the project's impact on society or environment. - Consideration of sustainability and ethical practices.

Final Mark Distribution:

- Project Presentation & Demonstration → 15 Marks
- Project Report & Documentation → 5 Marks
- Societal & Sustainable Impact → 5 Marks
- Total → 25 Marks

Internal Assessment:

For 03 credit - 80 marks subject

Assessment consists of one Mid Semester Examination (MSE) of 20 marks and In Semester Examination (ISE) of 20 marks. The MSE to be conducted based on 50 % syllabus with duration of one hour.

For 02 credit - 60 marks subject

Assessment consists of one Mid Semester Examination (MSE) of 15 marks and In Semester Examination (ISE) of 15 marks. The MSE to be conducted based on 50 % syllabus with duration of one hour.

In Semester Examination (ISE)

SE 20 marks = 05 marks attendance +15 marks for Activities.

ISE 15 marks = 05 marks attendance +10 marks for Activities.

The Rubrics for activities are as follows. The activities will be decided by course in charge and approved by HoD.

Sr. No	Rubrics	Marks
1	Multiple Choice Questions (Quiz)	05 Marks
2	Literature review of papers/journals	05 Marks
3	Participation in event/ workshop/ talk / competition followed by small report and certificate of participation relevant to the subject	05 Marks
4	Extra Experiments/ Virtual Lab	05 marks
5	Content beyond syllabus presentation	05 marks
6	Wins in the event/competition/hackathon pertaining to the course	10 Marks
7	Case study, Presentation, group discussion, technical debate on recent trends in the said course	10 Marks
8	NPTEL/ Coursera/ Udemy/any MOOC Certificate course for 4 weeks or more	10 Marks
9	Creating Proof of Concept	10 Marks
10	Mini Project	10 Marks
11	GATE Based Assignment test/Tutorials etc	10 Marks

*For sr.no.8, the date of certification exam should be within the term and in case a student is unable complete the certification, the grading has to be done accordingly.

=

SIES Graduate School of Technology
Department of Information Technology
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC401	Applied Mathematics-IV	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITC401	Applied Mathematics-IV	20	20	60	--	--	100

Pre-requisite: Knowledge of

1. FEC101- Applied Mathematics-I
2. FEC102- Applied Mathematics-II
3. ITC301- Applied Mathematics-III

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/ development of solution
4. PO4: Conduct investigation of complex problems

Course Objectives:

1. To develop a strong understanding of statistical techniques for data analysis.
2. To provide a foundation in probability theory and its applications.
3. To introduce probability distributions and sampling theories for real-world applications.
4. To impart knowledge of hypothesis testing and ANOVA techniques.
5. To familiarize students with linear programming methods and optimization techniques.
6. To equip students with nonlinear programming methods and advanced optimization techniques.

Course Outcomes (COs):

After completing this course, students will be able to:

1. Apply statistical techniques for data analysis and problem-solving.
2. Understand and implement probability theories in decision-making.
3. Utilize probability distributions and sampling theory for data interpretation.
4. Perform hypothesis testing and ANOVA for statistical inferences.

5. Solve real-world optimization problems using linear programming methods.
6. Apply nonlinear programming techniques for constrained optimization problems.

Module No.	Unit No.	Topic	No. of Hrs	Mapped CO
1	**Statistical Techniques**		6	CO1
	1.1	Karl Pearson's Coefficient of correlation (r) and related concepts with problems.		
	1.2	Spearman's Rank correlation coefficient (R) (Repeated & non-repeated ranks with problems).		
	1.3	Lines of regression.		
		Self-Learning Topics: Covariance.		
2	**Probability Theory**		6	CO2
	2.1	Total Probability theorem and Bayes' theorem.		
	2.2	Discrete and continuous random variables with probability distribution and probability density function.		
	2.3	Expectation, Variance, Laws of expectation.		
	2.4	Moment generating function, Raw and central moments up to 4th order.		
		Self-Learning Topics: Skewness and Kurtosis of distribution (data).		
3	**Probability Distribution and Sampling Theory**		7	CO3
	3.1	Probability Distribution: Poisson and Normal distribution.		
	3.2	Sampling distribution, Testing of Hypothesis, Level of Significance, Critical region, One-tailed, and two-tailed test, Degree of freedom.		
	3.3	Large Sampling with test of single mean and difference of means.		
	3.4	Students' t-distribution (Small sample). Test the significance of mean and difference between the means of two samples.		
		Self-Learning Topics: Large sampling with testing for parameters.		
4	**Test of Hypothesis - Chi-square Distribution and ANOVA**		7	CO4
	4.1	Chi-Square Test: Test of goodness of fit.		
	4.2	Independence of attributes, Contingency table.		

	4.3	Analysis of Variance (F-Test): One-way classification, Two-way classification (short-cut method).		
		Self-Learning Topics: Other types of non-parametric tests.		
5		**Linear Programming Problems**	7	CO5
	5.1	Types of solutions, Standard and Canonical of LPP, Basic and Feasible solutions, slack variables, surplus variables, Simplex method.		
	5.2	Artificial variables, Big-M method (Method of penalty).		
	5.3	Dual Simplex Method.		
		Self-Learning Topics: Principle of Duality, Dual of LPP.		
6		**Nonlinear Programming Problems**	6	CO6
	6.1	NLPP with one equality constraint (two or three variables) using the method of Lagrange's multipliers.		
	6.2	NLPP with One inequality constraint: Kuhn-Tucker conditions.		
	6.3	NLPP with two inequality constraints: Kuhn-Tucker conditions.		
		Self-Learning Topics: NLPP with two equality constraints.		

Textbooks:

1. **S.C. Gupta, V.K. Kapoor**, *Fundamentals of Mathematical Statistics*, Sultan Chand & Sons, 11th Edition, 2014.
2. **Sheldon M. Ross**, *Introduction to Probability and Statistics for Engineers and Scientists*, Academic Press, 6th Edition, 2020.
3. **Montgomery, D.C., and Runger, G.C.**, *Applied Statistics and Probability for Engineers*, Wiley, 6th Edition, 2014.
4. **Hamdy A. Taha**, *Operations Research: An Introduction*, Pearson, 10th Edition, 2016.
5. **K.S. Sivakumar**, *Operations Research**, Pearson, 2nd Edition, 2018.

Reference Books:

1. **J. Medhi**, *Statistical Methods: An Introductory Text*, New Age International, 2nd Edition, 2018.
2. **A. Papoulis, S. U. Pillai**, *Probability, Random Variables, and Stochastic Processes*, McGraw Hill, 4th Edition, 2002.
3. **Ravindran, Phillips, and Solberg**, *Operations Research: Principles and Practice*, Wiley, 2nd Edition, 2017.
4. **Casella, G., and Berger, R.L.**, *Statistical Inference*, Cengage Learning, 2nd Edition, 2017.
5. **F. S. Hillier, G. J. Lieberman**, *Introduction to Operations Research*, McGraw Hill, 10th Edition, 2021.

Online Resources:

1. **MIT Open Course Ware – Probability and Statistics**
   a. https://ocw.mit.edu/courses/mathematics/
   b. Covers **Probability Theory, Bayes' Theorem, and Statistical Methods**.
2. **Coursera – Probability and Statistics (Stanford University)**
   a. https://www.coursera.org/learn/probability-statistics
   b. Covers **Probability Distributions and Hypothesis Testing**.
3. **NPTEL – Operations Research (IIT Kharagpur)**
   a. https://nptel.ac.in/courses/111107128
   b. Covers **LPP, Duality, and Nonlinear Programming Problems**.
4. **MIT Open Course Ware – Linear Programming and Optimization**
   a. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-251j-introduction-to-mathematical-programming-fall-2009/
   b. Covers **Simplex Method, Big-M Method, and Kuhn-Tucker Conditions**.

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 5 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus)

End Semester Examination

ESE duration of 03 hours and 80 marks and to be scaled down to 60.

The question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC402	Computer Network	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
ITC402	Computer Network	20	20	60	--	--	100

Pre- requisite:

FEC204: Digital System Design

Program Outcomes Addressed

PO1: Engineering Knowledge
PO2: Problem analysis
PO3: Design/Development of Solutions
PO4: Conduct investigation of complex problems
PO5: Engineering Tool Usage
PO6: The Engineer and the world
PO11: Life Long Learning

Course Objectives:

1. Build an understanding about the concepts and fundamentals of computer networks.
2. Perceive comprehensive knowledge about the principles, protocols, reference models (OSI and TCP/IP) and its functionalities.
3. Analyze Client/server model in application layer protocols.
4. Understand data transportation issues and related protocols for end-to-end delivery of data.
5. Implement various routing algorithms and analyze them
6. Apply the concepts of Error detection and error correction to identify the errors in data.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Describe role of each layer of the OSI and TCP/IP models.
2. Analyze the standard client server applications of the application layer.
3. Implement Transport Layer protocols.

4. Implement appropriate routing algorithms for network-layer packet delivery
5. Illustrate the data link layer services & multiple access techniques
6. Categorize various transmission media.

Module No.	Unit No.	Topics	Hrs.	Mapped to CO
1.0		**Introduction to Networking**	**4**	**CO1**
	1.1	Introduction to computer network, network application, network software and hardware components (Interconnection networking devices), Network Topology.		
	1.2	Reference models: OSI Reference model, TCP/IP suite Comparison of OSI and TCP/IP.		
		Self-Learning: Identify the different devices used in network connection in college campus		
2.0		**Application Layer**	**6**	**CO2**
	2.1	Introduction: Providing Services, Application layer Paradigms, client-Server Paradigms, Application Programming Interface, Using Services of the Transport Layer.		
	2.2	Standard Client Server applications: World Wide Web and HTTP, FTP, Electronic Mail, TELNET, Secure Shell (SSH), Domain Name System (DNS).		
		Self-Learning: Difference between HTTP and FTP Protocol. Various protocols in session and presentation layer in OSI Model		
3.0		**Transport Layer**	**6**	**CO3**
	3.1	The Transport Service: Transport Layer Protocols, Simple Protocol, Stop-and-Wait Protocol, GoBack-N Protocol (GBN), Selective-Repeat Protocol, Bidirectional Protocols: Piggybacking		
	3.2	UDP and TCP Protocol: User Datagram, UDP Services, UDP Applications, TCP Services, TCP Features, TCP Segment, A TCP Connection, Flow Control, Error Control, TCP Congestion Control, TCP Timers		
		Self-Learning: List real time example of UDP and TCP		
4.0		**Network layer**	**10**	**CO4**
	4.1	Introduction: Network-Layer Services, Packet Switching, Network Layer Protocols: IPv4 Datagram Format, IPv4 Addresses, Subnetting, Supernetting, Forwarding of IP Packets, ICMPv4. IPv6 packet format, Transition from IPv4 to IPv6.		
	4.2	Routing algorithms: Shortest Path (Dijkastra’s, Bellmanford		

		Algorithm), Distance Vector Routing (RIP), Link state routing (OSPF), BGP, QoS, Network Layer Congestion.		
		Self-Learning Study difference between IPV4 and IPV6. Network Class A, B, C, D, E and subnet mask		
5.0		**Data Link Layer**	**10**	**CO5**
	5.1	DLL Design Issues (Services, Framing, Error Control, Flow Control), Error Detection and Correction (Hamming Code, CRC, Checksum), Elementary Data Link protocols, Stop and Wait, Sliding Window (Go Back N, Selective Repeat)		
	5.2	Medium Access Control Sublayer **,** Channel Allocation problem, Multiple access Protocol (Aloha, Carrier Sense Multiple Access (CSMA/CD). Ethernet Protocol: Standard Ethernet, Fast Ethernet (100 Mbps), Gigabit Ethernet, 10-Gigabit Ethernet.		
		Self-Learning: Link Layer Security		
6.0		**Physical Layer**	**3**	**CO6**
	6.1	Unguided Transmission Media: Radio wave, Microwave, Infrared. Guided Transmission Media: Twisted pair, Coaxial, Fiber optics		
	6.2	Switching: Circuit- Switched Networks, Packet switching, Structure of a switch.		
		Self-Learning Topics: Compare and contrast various transmission media.		
		Total	**39**	

Textbooks:

1. James F. Kurose, Keith W. Ross, **Computer Networking, A Top-Down Approach Featuring the Internet**,6th edition, Addison Wesley
2. A.S. Tanenbaum, **Computer Networks**,4th edition Pearson Education
3. B.A. Forouzan, **Data Communications and Networking**, 5th edition, TMH

Reference books:

1 S.Keshav,**An Engineering Approach To Computer Networking**, Pearson

2 Natalia Olifer & Victor Olifer,**Computer Networks: Principles, Technologies & Protocols for Network Design**, Wiley India, 2011.

3 Larry L.Peterson, Bruce S.Davie, **Computer Networks: A Systems Approach**, Second
Edition ,The Morgan Kaufmann Series in Networking

Online References:

1 https://www.netacad.com/courses/networking/networking-essentials
2 https://www.coursera.org/learn/computer-networking
3 https://nptel.ac.in/courses/106/105/106105081
4 https://www.edx.org/course/introduction-to-networking

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 5 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus)

End Semester Examination

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

The question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks): Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC403	Operating System	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
ITC403	Operating System	20	20	60	--	--	100

Pre-requisite:

1. FEL103, FEL205: Knowledge of Basic Programming
2. ITC305 : Computer Organization and Architecture

Program Outcome mapped:

1. PO1: Engineering Knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.

Course Objectives:

1. To provide a comprehensive understanding of the core components, functions, and services of an Operating System (OS).
2. To introduce the concepts of process management, including process states, scheduling, and multithreading, and their relevance in modern computing environments.
3. To enable learners to apply synchronization techniques, analyze deadlock scenarios, and evaluate solutions for inter-process communication (IPC).
4. To explore memory management techniques, including virtual memory, paging, segmentation, and page replacement algorithms, and their impact on system performance.
5. To analyze storage management strategies, including file systems, disk scheduling, and I/O management, in the context of modern storage technologies.
6. To study the architecture and functionalities of special-purpose operating systems, including distributed, real-time, embedded, and cloud-based systems, and their applications in emerging technologies.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Explain the fundamental concepts, components, and services of an Operating System and its role in computer system organization.
2. Analyze process management techniques, including process scheduling, multithreading, and CPU scheduling algorithms, and evaluate their performance in different scenarios.
3. Apply synchronization primitives, analyze deadlock conditions, and propose solutions for inter-process communication (IPC) in a multi-process environment.
4. Evaluate memory management techniques, including paging, segmentation, and virtual memory, and implement page replacement algorithms to optimize system performance.
5. Design and analyze storage management strategies, including file systems, disk scheduling, and I/O management, for modern storage technologies.
6. Compare and contrast the functionalities of special-purpose operating systems, including distributed, real-time, embedded, and cloud-based systems, and their applications in emerging technologies.

Detailed Syllabus:

Module No.	Unit No.	Topics	Hrs.	Mapped to CO
0	**Prerequisite**		**1**	
	0.1	Programming Language execution, and process Basics of Hardware (ALU, RAM, ROM, HDD, etc.)		
I	**Fundamentals of Operating System**		**4**	**CO1**
	1.1	Programming Language Execution, Basics of Hardware (ALU, RAM, ROM, HDD, etc.)		
	1.2	Introduction to Operating Systems: OS Structure and Operations, Functions of OS, OS Services and Interface		
	1.3	System Calls and Types, System Programs, OS Structure, System Boot		
		Self-learning Topics: Study of Linux, Windows, and macOS system calls and their differences.		
II	**Process Management**		**8**	**CO2**
	2.1	Basic Concepts of Process, Operations on Process, Process State Model and Transition, Process Control Block, Context Switching		
	2.2	Multithreading: Thread Types and Models		

	2.3	Uni-Processor Scheduling: Types, Criteria, and Algorithms (FCFS, SJF, Round Robin, Priority Scheduling, Multilevel Queue Scheduling)		
		Self-learning Topics: Performance comparison of scheduling algorithms; Real-time scheduling in IoT and embedded systems.		
III	**Process Synchronization and deadlock**		**9**	**CO3**
	3.1	Inter-process Communication (IPC): Concepts and Techniques, Race Conditions, Critical Section Problem, Peterson’s Solution		
	3.2	Synchronization: Hardware Support, Semaphores, Mutex, Classic Synchronization Problems (Producer-Consumer, Reader-Writer, Dining Philosophers)		
	3.3	Deadlocks: Characterization, Detection, Prevention, Avoidance (Banker’s Algorithm)		
		Self-learning Topics:Case study on deadlock detection and recovery in distributed systems.		
IV	**Memory Management**		**7**	**CO4**
	4.1	Basic Concepts of Memory Management: Swapping, Contiguous Memory Allocation		
	4.2	Paging and Page Table Structure, Segmentation, Virtual Memory: Demand Paging, Copy-on-Write		
	4.3	Page Replacement Algorithms (FIFO, LRU, Optimal), Thrashing		
		Self-learning Topics: Memory management in Linux; Implementation of page replacement algorithms.		
V	**Storage Management**		**6**	**CO5**
	5.1	File Systems: Concepts, Access Methods, Directory Structure, Implementation		
	5.2	Allocation Methods (Contiguous, Linked, Indexed), Free Space Management		
	5.3	Disk Structure and Scheduling (FCFS, SSTF, SCAN, C-SCAN, LOOK), RAID Structures, I/O Systems		
		Self-learning Topics: Comparison of file systems in Linux (ext4) and Windows (NTFS); I/O management in cloud environments.		
VI	**Special-purpose Operating Systems**		**4**	**CO6**
	6.1	Open-source vs. Proprietary OS, Distributed OS, Network OS		
	6.2	Embedded OS, Real-Time OS (RTOS), Mobile OS (Android, iOS), Cloud and IoT OS, Multimedia OS		

	Self-learning Topics: Case study on a real-time operating system (e.g., FreeRTOS) or a cloud-based OS (e.g., ChromeOS) or vehicle OS

Text Books:

1. A. Silberschatz, P. Galvin, G. Gagne, *Operating System Concepts*, 10th ed., Wiley, 2018.
2. W. Stallings, *Operating Systems: Internal and Design Principles*, 9th ed., Pearson, 2018.
3. A. Tanenbaum, *Modern Operating Systems*, 4th ed., Pearson, 2015.

Reference Books:

1. N. Chauhan, *Principles of Operating Systems*, 1st ed., Oxford University Press, 2014.
2. A. Tanenbaum and A. Woodhull, *Operating System Design and Implementation*, 3rd ed., Pearson.
3. R. Arpaci-Dusseau and A. Arpaci-Dusseau, *Operating Systems: Three Easy Pieces*, 1st ed., CreateSpace Independent Publishing Platform, 2018.

Online References:

1.https://www.nptel.ac.in/
2. https://swayam.gov.in/
3.https://www.coursera.org/

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 5 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus)

End Semester Examination

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

Question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks) :- Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC405	Critical Thinking and Design	02	-	-	02	-	-	02

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
ITC405	Critical Thinking and Design	15	15	45	--	--	75

Pre- requisite: None

Program Outcomes Addressed

PO2: Problem analysis

PO3: Design/development of solutions

PO5: Engineering Tool Usage

PO6: The Engineer and the world

PO7: Ethics

PO8: Individual and Collaborative Team Work

PO9: Communication

PO10: Project Management and Finance.

PO11: Life Long Learning

Course Objectives:

1. To describe the fundamentals of critical thinking and fair-minded reasoning for effective decision-making.
2. To differentiate personal thinking stages and implement structured strategies for continuous cognitive growth.
3. To analyze key elements of thought and intellectual standards to enhance logical reasoning.
4. To examine the principles of design thinking and apply them to solve real-world problems through an iterative, user-centered approach.
5. To demonstrate hands-on experience with idea generation, customer insights, and problem framing to drive innovation.

6. To employ creative problem-solving techniques such as brainstorming, prototyping, and hypothesis validation to design user-centric solutions.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Interpret the fundamentals of critical thinking and fair-minded reasoning for effective decision-making.
2. Identify their cognitive development stage and implement structured strategies to progress as a critical thinker.
3. Apply intellectual standards like clarity, accuracy, and logic to improve reasoning and problem-solving skills.
4. Integrate design thinking principles to create innovative, balanced, and user-centered solutions.
5. Develop a broad perspective in understanding customer needs and effectively define problem statements using diverse methodologies.
6. Implement creative solutions and enhance ideas through iterative prototyping and user feedback using brainstorming techniques.

Module No.	Unit No.	Topics	Hrs.	Mapped to CO
1.0		**Introduction to Critical Thinking**	**4**	**CO1**
	1.1	**Introduction:** Start-up definition of Critical Thinking, How skilled are you as a Thinker? Hard Work, Concept of Critical Thinking, Establish new habits of thoughts, Develop confidence		
	1.2	**Fairminded Thinker:** Weak Vs Strong Critical Thinking Requirement of Fairmindedness Intellectual: Humility, Courage, Empathy, Integrity, Perseverance, Autonomy Interdependence of Intellectual Virtues		
		Self-Learning Topics: Role of Intellectual Humility in Decision-Making, How to Overcome Cognitive Biases for Stronger Reasoning, Practical Techniques to Develop Fair-Minded Thinking. Case Study: The Challenger Disaster: How Ignoring Critical Thinking Led to Catastrophe		
2.0		**Four Stages of Development, Game Plan**	**3**	**CO2**
	2.1	**Four Stages of Development:** Stage 1: Unreflective thinker, Stage 2: Challenged thinker, Stage 3: Beginning thinker, Stage 4: Practicing thinker		
	2.2	**Game Plan:** Purpose & Key Components of Game Plan, Integrating of Game Plan Strategies		

		Self-Learning Topics: Characteristics and challenges at each stage, Common obstacles and how to address them, Practical ways to enhance critical thinking in work and academics. Case Study: Explores how a student progresses through four stages using self-reflection& discipline.		
3.0		**Self-Understanding, Parts & Universal Standards**	**3**	**CO3**
	3.1	**Three Distinctive Functions:** Recognize the Mind’s Three Distinctive Functions; Special Relationship		
	3.2	**Thoughts & Intellectual Standards:** Fundamental structures of thought, The elements of thought, Universal Intellectual Standards: Clarity, Accuracy, Precision, Relevance, Depth, Breadth, Logic, Significance, Fairness		
		Self-Learning: Recognizing biases and promoting ethical decision-making, Case Study: Analyzes how a company applied intellectual standards to refine its business strategies.		
4.0		**Design Thinking & its Key Tenets**	**5**	**CO4**
	4.1	**Design Thinking Basics:** Traditional Model vs. Design Thinking, Five Stages: Inspire, Empathize, Define, Ideate, Prototype & Test Scale Thinking: Lean Thinking, Critical Thinking, Lateral Thinking, Design Thinking		
	4.2	**Key Tenets:** Customer-Centric Approach, Thinking Beyond Products, Balancing Desirability, Feasibility & Viability, Broad & Compartmentalized Thinking, Visual Thinking & Hands-on Approach		
		Self-Learning: Understanding the shift from conventional problem-solving to iterative design processes, Designing solutions with user needs at the core while balancing business feasibility. Case Study: How a global brand used design thinking to enhance customer experience and increase engagement.		
5.0		**Inspire, Empathize and Define**	**5**	**CO5**
	5.1	**Generating & Broadening Ideas:** Creating Stretch Goals, Power of Metaphors & Widening Perspectives, Importance of Diversity in Ideation		
	5.2	**Empathize & Define:** New Channels for Customer Insights, Deep Customer Empathy & Stakeholder Analysis, Leveraging		

		Technology for Insights, Mind Mapping: Stakeholders, Journey Mapping, Problem Framing		
		Self-Learning: Using metaphors and ideation techniques to expand creative possibilities, How diverse teams enhance innovation and problem-solving,Visualizing stakeholder journeys and structuring problem statements for better solutions. Case Study: How Airbnb used empathy mapping and customer insights to redefine its business model.		
6.0		**Ideate, Prototype and Test**	**6**	**CO6**
	6.1	Ideate: Brainstorming & Hybrid Ideation Techniques, Challenging Assumptions & Breaking Patterns, Cross-Industry Inspiration (Analogous Design), Designing for Extreme Users & Ideation Triggers		
	6.2	Prototype & Test: Rapid Prototyping & Hypothesis Validation, Storyboarding & Scenario Visualization, Collecting Feedback & Managing Failed Prototypes		
		Self-Learning: Exploring structured and unstructured brainstorming approaches, Testing ideas quickly through prototypes and data-driven validation, Using visual storytelling to map user experiences and refine concepts. Case Study: Explore Apple's iterative prototyping process in designing user-friendly products.		
		Total	**26**	

Textbooks:

1. Richard Paul, Linda Elder, “Critical Thinking: Tools for Taking Charge of Your Learning and Your Life”,Authors: Richard Paul, Linda Elder,2013,Pearson Education
2. Pavan Soni, “Design Your Thinking: The Mindsets, Toolsets, and Skill Sets for Creative Problem-solving”,2020, Penguin Random House India Private Limited

Reference books:

1. Roger L. Martin,“The Design of Business: Why Design Thinking is the Next Competitive Advantage”,2009,Harvard Business Press
2. Richard Paul, Robert Niewoehner, Linda Elde,”The Thinker's Guide to Engineering Reasoning, 2013, Foundation for Critical Thinking Design
3. Tilmann Lindberg, Christoph Meinel, Ralf Wagner, Christo, “Design Thinking: Creating a Culture of Innovation”,Springer
4. Brooke Noel Moore & Richard Parker,”Critical Thinking”13th Edition,2020,McGraw-Hill Education

Online References:

1. https://onlinecourses.nptel.ac.in/noc19_mg60/preview
2. https://onlinecourses.nptel.ac.in/noc20_de03/preview
3. https://onlinecourses.swayam2.ac.in/imb24_mg37/preview
4. https://www.coursera.org/learn/uva-darden-design-thinking-innovation

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 10 marks and 5 marks attendance.

MSE: To be conducted as written examination for 15 marks (on 50% syllabus)

End Semester Examination:

Question paper will comprise of 3 questions.
Question1(15 marks): - Solve any 03 out of 04. All questions carry 05 marks each.
Question 2 (30 marks): - Solve any 03 out of 05. All questions carry 10 marks each.
Question3(15 marks) :- Solve any 03 out of 04. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC4011	Artificial Intelligence	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
MDMC4011	Artificial Intelligence	20	20	60	--	--	100

Pre-requisite:

1. FEC101: Applied Mathematics -I
2. FEC201: Applied Mathematics -II

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solutions
4. PSO2: Become technocrats capable of working in multi-disciplinary fields

Course Objectives:

1. To conceptualize the basic ideas and techniques underlying the design of intelligent systems.
2. To make students understand and explore the mechanism of mind that enables intelligent thought and

action.

3. To make students understand advanced representation formalism and search techniques.
4. To make students understand how to deal with uncertain and incomplete information.

Course Outcomes:

After successful completion of the course, students will be able to:

1. To understand the fundamental concepts, evolution, and applications of AI.
2. To analyze intelligent agents, their structures, and problem-solving approaches using search methods.
3. To evaluate different AI problem-solving techniques, including uninformed, informed, and optimization-based search algorithms.
4. To apply logical reasoning and knowledge representation techniques for AI-based inference and decision-making.
5. To understand AI planning methods and different learning paradigms, including reinforcement learning.
6. To explore AI applications in real-world domains such as NLP, robotics, healthcare, retail, and banking.

Module No.	Unit No.	Topics	Hrs.	CO
1.0		**Introduction to AI**	**05**	**CO1**
	1.1	Introduction to Artificial Intelligence, Brief history and evolution of AI, Intelligent Systems: Categorization of Intelligent System, Components of AI Program, Foundations of AI, Sub-areas of AI		
		Self learning: Applications of AI, Current trends in AI		
2.0		**Intelligent Agents and Environments**	**09**	**CO2**
	2.1	Definition of an agent and its environment, Structure of Intelligent Agents, Types of agents, Learning Agent.		
	2.2	Solving problem by Searching: Problem Solving		

		Agent, Formulating Problems, Example Problems.		
		Self learning: The concept of rationality		
3.0		**Problem Solving Techniques in AI**	**07**	**CO3**
	3.1	Uninformed search methods: Breadth-First Search (BFS) and Depth-First Search (DFS), Depth Limited Search, Depth First Iterative Deepening (DFID)		
	3.2	Informed Search Methods: Greedy best first Search, A* Search, Memory bounded heuristic Search.		
		Local Search Algorithms and Optimization Problems: Hill climbing search Simulated annealing, Genetic algorithms.		
		Self learning: Adversarial Search: Game Playing, Min-Max Search, Alpha Beta Pruning		
4.0		**Knowledge Representation and Logical Reasoning**	**07**	**CO4**
		Logical Agents: Knowledge based Agents, Fundamentals of logic: propositional logic basics, Representation of knowledge using rules, First Order Logic (FOL): Syntax and Semantic, Basic inference techniques: forward chaining and backward chaining, Simple rule-based systems and examples		
		Knowledge Engineering in First-Order Logic, Propositional vs. First-Order Inference, Unification, Resolution		
		Self Learning: Representing knowledge in an uncertain domain, The semantics of belief network		
5.0		**Planning and Learning**	**05**	**CO5**
		The planning problem, Planning with state space search, Partial order planning, Hierarchical planning, Conditional Planning.		
		Learning: Forms of Learning, Theory of Learning, PAC learning. Introduction to statistical learning		

		(Introduction only) Introduction to reinforcement learning: Learning from Rewards		
		Self-learning: Passive and Active Reinforcement Learning		
6.0		**AI Applications**	**06**	**CO6**
		Introduction to NLP- Language models, Grammars, Parsing Robotics - Robots, Robot hardware, Problems Robotics can solve AI applications in Healthcare, Retail, Banking		
		Self-learning: AI applications in Retail, Banking		
		Total	**39**	

Textbooks:

1. Stuart J. Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", 4th Edition" Pearson Education, 2020.
2. George F Luger, "Artificial Intelligence" Low Price Edition, Fourth edition, Pearson Education, 2005
3. Lavika Goel, "Artificial Intelligence: Concepts and Applications," Wiley 2021.

Reference books:

1. Saroj Kaushik, "Artificial Intelligence", Cengage Learning, First edition, 2011
2. Elaine Rich and Kevin Knight, "Artificial Intelligence", Third Edition, McGraw Hill Education,2017.
3. Nils J. Nilsson, Principles of Artificial Intelligence, Narosa Publication.
4. Deepak Khemani, A First Course in Artificial Intelligence, McGraw Hill Publication
5. Patrick H. Winston, Artificial Intelligence, 3rd edition, Pearson Education.

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 5 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus)

End Semester Examination

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

Question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks) :- Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC 4051	Advance Data Structure	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
MDMC 4051	Advance Data Structure	20	20	60	--	--	100

Pre-requisite:

1. Data Structures and Algorithms

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/ development of solutions:
4. PO4: Conduct investigations of complex problems:
5. PO5: Modern tool usage:
6. PO6: The engineer and society

Course Objectives:

1. Understand advanced data structures and their applications in solving computational problems.
2. Analyze and implement efficient hashing techniques for optimized search operations.
3. Explore priority queues and balanced trees to improve data retrieval and manipulation.
4. Apply graph algorithms for solving network, optimization, and traversal problems.
5. Design efficient string processing and pattern matching algorithms for text-based applications.
6. Implement real-world applications using advanced data structures for improved system performance.

Course Outcomes:

After successful completion of the course student will be able to…

1. Apply and analyze advanced hashing techniques for fast data retrieval and collision handling.
2. Implement and evaluate priority queues and balanced trees for efficient search and storage.
3. Solve problems using graph algorithms like Topological Sorting, Shortest Paths, and Maximum Flow.
4. Design efficient pattern matching algorithms using KMP, Rabin-Karp, and Suffix Trees.
5. Develop optimized memory management techniques using Tries, Skip Lists, and Segment Trees.
6. Implement real-world applications using advanced data structures in domains like AI, Blockchain, and Networking.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1		**Advanced Hashing Techniques**	**6**	**CO1**
	1.1	Hashing Fundamentals: Hash functions, Properties, Load Factor.		
	1.2	Collision Handling: Separate Chaining, Open Addressing (Linear, Quadratic, Double Hashing).		
	1.3	Advanced Hashing: Universal Hashing, Extendible Hashing, Perfect Hashing.		
	1.4	Applications: Hashing in Databases, Cryptographic Hashing, Hash Tables in Standard Libraries.		
		Self-Learning Topics: Consistent Hashing, Cuckoo Hashing, Hashing in Blockchain.		
2		**Priority Queues and Balanced Trees**	**8**	**CO2**
	2.1	Priority Queues: Definition, Binary Heaps,Heap Order property, Heap Operations insert, delete Percolate down .		
	2.2	Binomial Heaps: Structure, Operations, Implementation.		
	2.3	Multi-Way Search Trees: B-Trees, B+ Trees, 2-3 Trees.		
	2.4	Red-Black Trees: Properties, Rotations, Insertions, Deletions.		
		Self-Learning Topics: Fibonacci Heaps, Treaps, Applications in Databases.		
3		**Graph Algorithms**	**8**	**CO3**
	3.1	Graph Representations: Adjacency Matrix, Adjacency List, Incidence Matrix.		
	3.2	Graph Traversals: BFS, DFS, Connected Components, Bridges, Articulation Points.		
	3.3	Graph Applications: Topological Sorting, Shortest Paths (Dijkstra's, Bellman-Ford, Floyd-Warshall).		
	3.4	Minimum Spanning Trees: Kruskal's, Prim's Algorithm.		
		Self-Learning Topics: Graph Coloring, Euler Circuits, Graph Databases (Neo4j).		
4		**String Processing and Pattern Matching**	**6**	**CO4**
	4.1	Naïve String Matching: Brute Force Approach.		
	4.2	Efficient Pattern Matching: Knuth-Morris-Pratt (KMP), Rabin-Karp Algorithm.		
	4.3	Suffix Trees and Arrays: Construction, Applications.		
	4.4	Aho-Corasick Algorithm: Multi-Pattern Matching.		
		Self-Learning Topics: Boyer-Moore Algorithm, Text Searching in Big Data.		
5		**Applications of Advanced Data Structures**	**7**	**CO5**

	5.1	Tries: Standard Tries, Compressed Tries, Suffix Tries.		
	5.2	Skip Lists: Structure, Search, Insert, Delete.		
	5.3	Segment Trees: Range Queries, Lazy Propagation.		
	5.4	Disjoint Sets (Union-Find): Path Compression, Applications in Kruskal’s Algorithm.		
		Self-Learning Topics: Fenwick Trees (Binary Indexed Trees), Bloom Filters, Memory Pools.		
6		**Applications of Advanced Data Structures**	**4**	**CO6**
	6.1	Blockchain Data Structures: Merkle Trees, DAGs in Cryptocurrencies.		
	6.2	Big Data Processing: Hadoop Data Structures, Trie-Based Indexing.		
		Self-Learning Topics: Real-World Implementations in Cloud Computing, Networking, and Bioinformatics.		
	Total		**39**	

Textbooks

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford SteinCormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C, Introduction to Algorithms (Fourth Edition) 2024. MIT Press.
2. Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2014). Data Structures and Algorithms in Java (6th ed.). Wiley.
3. Sahni, S. (2005). Data Structures, Algorithms, and Applications in C++ (2nd ed.). Universities Press.
4. Weiss, M. A. (2013). Data Structures and Algorithm Analysis in C++ (4th ed.). Pearson.

Reference Books

1. Knuth, D. E. (1997). The Art of Computer Programming, Volume 3: Sorting and Searching (2nd ed.). Addison-Wesley.
2. Kleinberg, J., & Tardos, É. (2005). Algorithm Design. Pearson.
3. Sedgewick, R., & Wayne, K. (2011). Algorithms (4th ed.). Addison-Wesley.
4. Dasgupta, S., Papadimitriou, C. H., & Vazirani, U. (2006). Algorithms. McGraw-Hill.

Online References:

NPTEL Courses:

1. https://nptel.ac.in/courses/106102064
2. https://nptel.ac.in/courses/106106133
3. https://onlinecourses.nptel.ac.in/noc22_cs92/preview

Coursera Courses:

1. https://www.coursera.org/specializations/data-structures-algorithms
2. https://www.coursera.org/learn/advanced-data-structures-rsa-and-quantum-algorithms

3. https://www.coursera.org/learn/advanced-data-structures

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 05 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus).

End Semester Examination

$ ESE is of duration 03 hours and 80 marks and will be scaled down to 60.

The question paper will comprise of 03 questions.
Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDM C4061	Cost Management	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$^{\$}$			
		ISE	MSE				
MDM C4061	Cost Management	20	20	60	--	--	100

Pre-requisite: Basic Accounting principles, Quantitative skills etc.

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO11:Life-Long Learning

Course Objectives: To acquire knowledge and understanding of the concepts, techniques, and practices of cost and management accounting and to develop skills for decision making.

Course Outcomes: Upon completion of this course, learners will be able to…

CO1: To understand and analyze different cost concept and methods.
CO2: To understand the Elements of Cost & Cost classification.
CO3: To apply various material concepts & classifications for preparation of cost sheet.
CO4: To analyze various techniques of costing and its application in Finance, budgets and budgetary control.
CO5: To develop requisite data for cost control and cost reduction.
CO6: To evaluate marginal costing techniques for decision making.

Module No.	Unit No.	Topics	Hrs.	CO
1.0	1	**Module 1: Introduction to Cost Accounting**	04	
		Meaning of Cost, Cost Accounting & its Objectives, Comparison between Cost accounting and Financial Accounting, Comparison between Cost Accounting and Management Accounting, Types of cost, Methods of costing & Techniques of costing.		**CO1**
		Self-Learning: Basic cost accounting concepts		
2.0	2	Classification of Costs and Cost Sheet	05	
		Elements of Cost, Classification of Costs, Cost center and cost unit, Preparation of Cost Sheet & Estimated Cost Sheet.		**CO2**
		Self-Learning: Purpose and importance of cost sheet.		
3.0		**Material Management and Accounting for materials**	06	
		Managing Purchase Functions, Cost of Material, Storing of materials – Inventory control methods, Costs associated with storing and ordering material, Economic Order Quantity, Fixation of levels and calculation of the same, Issue control-Pricing issues (LIFO, FIFO, Weighted Average), Material control - Objectives in Material Control, Stock Turnover, Material losses wastage, scrap, spoilage, defectives.		**CO3**
		Self-Learning: Basic flowchart for material flow in a company.		
4.0		**Accounting for labour and Overheads**	08	
		Accounting for labour: Types of Labour Cost, Methods of Remuneration, Treatment of overtime, fringe benefits, idle time etc. Accounting for overheads: Production overheads – Collection, Distribution to Production and service departments, Computation of Overheads Rate based on Machine Hour Rate method, Allocations and Apportionment, Absorption of overheads.		**CO4**
		Self-Learning: Types of labour, classification of overheads.		
5.0		**Cost Control and Cost Reduction**	10	
		Introduction, Comparison between cost control & cost reduction, Budgets and Budgetary Control, Meaning and Purpose		**CO5**

		of Budget, Objectives of Budgetary Control, Dangers of budget, Types of Budgets- Flexible Budget Standard Costing, Concept and development of standard costing, Variance analysis for cost, Direct Material variance- Cost, Price, usage, mix and yield variance Direct Labour Variance- Cost, Efficiency, usage, mix, yield and idle-time variance, Overhead Variance – Variable & Fixed Overhead variance, Sales variances – Value, rate, volume and mix variance.		
		Self-Learning: Differences and Interplay Between Cost Control and Cost Reduction.		
6.0		Marginal Costing & CVP Analysis	**06**	
		Nature and scope of Marginal Costing, Marginal Cost equation, Cost Profit volume analysis, Break Even point and Break-Even Analysis, Relevant cost analysis for decision making.		**CO6**
		Self-learning: Applications of Marginal Costing in Decision Making.		
		Total	**39**	

Textbooks:

1. B. Banerjee, *Cost Accounting: Theory and Practice*, 14th ed. New Delhi, India: PHI Learning Pvt. Ltd., 2021.
2. M. Y. Khan and P. K. Jain, *Management Accounting*, 8th ed. New Delhi, India: McGraw-Hill Education, 2021.

Reference books:

1. P. Shah, *Management Accounting*, 6th ed. New Delhi, India: Oxford University Press, 2015.
2. C. Drury, *Management and Cost Accounting*, 12th ed. Andover, U.K.: Cengage Learning, 2024.

Online References:

1. https://dynamicstudyhub.com/cost-management.
2. https://www.wallstreetmojo.com/cost-management

Course Assessment:

ISE: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 15 marks and 5 marks attendance.

MSE: To be conducted as written examination for 20 marks (on 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3 (20 marks) :- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL401	Computer Network Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme						
		Theory Marks			CIAP	ESEP	Total	
		Course Assessment		ESE				
		ISE	MSE					
ITL401	Computer Network Lab	--	--	--	25	25	50	

Prerequisite:

1.FEL103 C-Programming Lab
2. FEL203 Digital System Design Lab

Program Outcomes addressed:

PO1: Engineering knowledge
PO2: Problem analysis
PO3: Design/development of solutions
PO5: Modern Tool usage
PO6: The engineer and the world
PO11: Life Long Learning

Lab Objectives:

1. Build an understanding about fundamental concepts of computer network, protocols, architecture and applications
2. Demonstrate hands-on experience of computer network simulation and modelling techniques using simulation software
3. To understand the network simulator environment and visualize network topology and observe its performance
4. Implement client-server socket programming.
5. Demonstrate and interpret the traffic flow and the contents of protocol frames
6. Design and configure a network for an organization

Lab Outcomes:

After successful completion of the course, students will be able to:

1. Understand various network administration commands and analyze network equipment's used for internetworking.
2. Demonstrate the installation and configuration of network simulator and measure different network scenarios and their network performance using NS2.
3. Analyze the packet formats of each layer in TCP/IP model and traffic flow of different protocols
4. Implement the socket programming for client server architecture.
5. Illustrate the working of application layer protocol
6. Design a network for an organization using a network design tool

Suggested List of Experiments:		
Sr. No.	**Title of Experiments**	**Mapped to LO**
1.	To Install Network Operating system	LO1
2.	To study basic networking commands like ping, tracert, nslookup, netstat, ARP, RARP, ipconfig, ifconfig, dig, traceroute, nslookup, netstat.	LO1
3.	Analyse Hands-on on network equipment. Switches, Router Hardware Firewall	LO1
4.	Installation and configuration of NS-2 simulator. Write TCL scripts to create topologies.	LO2
5.	Write TCL scripts for topology with graphical simulation of traffic consideration (TCP, UDP) using NAM and plot the graph using NS2.	LO2
6.	Implement distance vector routing protocol in NS2.	LO3
7.	To install Wireshark and study the packet headers.	LO3
8.	To study and Implement Socket Programming using TCP and UDP	LO4
9	Perform remote login using Telnet Server.	LO5
10	Perform File Transfer and Access using FTP.	LO5
11	Design any organization network using Packet Tracer software. Eg. College Network or Campus network, using the concepts of Addressing (IP Address Assignment), Naming (DNS) and Routing. Also mention the internetworking devices used.	LO6

Text Books:

1. Computer Network Simulation in NS2 Basic Concepts and Protocol Implementation. -Prof Neeraj Bhargava, Pramod Singh Rathore, Dr. Ritu Bhargava, Dr. Abhishek Kumar, First Edition. BPB Publication.

2. Packet analysis with Wire shark, Anish Nath, PACKT publishing

3. TCP/IP Protocol Suite 4th Edition by Behrouz A. Forouzan

References:

1. NS2.34 Manual

2. Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems by Chris Sanders.

Online References:

1 https://www.netacad.com/courses/networking/networking-essentials
2 https://www.coursera.org/learn/computer-networking
3 https://nptel.ac.in/courses/106/105/106105081
4 https://www.edx.org/course/introduction-to-networking

Course Code	urse Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL402	Operating Systems Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITL402	Operating Systems Lab	--	--	--	25	25	50

Pre-requisite:

1. FEL103, FEL205: Knowledge of Basic Programming
2. ITC305 : Computer Organization and Architecture

Program Outcome mapped:

1. PO1: Engineering Knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.

Lab Objectives:

1. Demonstrate proficiency in Linux commands, system calls, and shell scripting for process and system management.
2. Implement system calls and APIs to develop custom Linux commands and utilities for file and process management.
3. Apply process scheduling and synchronization techniques to manage concurrency and resource sharing in operating systems.
4. Implement memory management strategies and deadlock handling mechanisms to optimize system performance and resource utilization.
5. Simulate virtual memory, demand paging, and file system operations to understand storage management in operating systems.
6. Develop network-based applications using socket programming and implement interprocess communication in distributed systems.

Lab Outcomes:

After successful completion of the course, students will be able to:

1. Explore and demonstrate the usage of Linux commands and system calls for file, directory, and

process management.
2. Develop shell scripts for system monitoring and process management tasks.
3. Implement and analyze process scheduling, synchronization, and deadlock handling techniques.
4. Implement memory management techniques and page replacement algorithms.
5. Simulate file allocation, file organization, and disk scheduling strategies.
6. Develop and implement interprocess communication using message passing and RPC mechanisms.

Detailed Syllabus

Sr. No.	**Topic**	**Tools**	**Mapped LO**
1	**Title**: Linux Commands & System Calls **Objective**: Learn fundamental Linux commands and system calls for file management, process handling, and networking. **Tasks**: 1. Learn basic Linux commands: ls, pwd, cd, mkdir, rm, cp, mv, cat, nano, chmod, ps, top, ping, uname, apt. 2. Write a shell script to list all files in a directory with detailed permissions. 3. Implement system calls to create, write, and read a file. 4. Demonstrate process creation using fork() and exec(). 5. Display details of the current logged-in user using system commands.	Ubuntu Terminal, Bash Shell	LO1
2	**Title**: Shell Scripting & Process Information **Objective**: Automate system monitoring and display process-related information using shell scripts. **Tasks**: 1. Write a script to display OS version, release number, and kernel version. 2. Write a script to display the top 10 processes consuming the most CPU. 3. Write a script to display the current logged-in user, shell, and home directory. 4. Automate system monitoring by checking memory usage and process count.	Linux Terminal, GDB Debugger	LO1
3	**Title**: Linux API Implementation **Objective**: Implement Linux commands using system calls and file handling APIs. **Tasks**:	CPU Scheduling Simulator, Python	LO3

	1. Implement the ls command using system calls. 2. Implement the cp command to copy a file using system calls. 3. Implement a custom mv command using file handling APIs. 4. Implement a basic cat command to display file contents.		
4	**Title**: Process Management - System Calls & Scheduling **Objective**: Implement process creation, scheduling algorithms, and process synchronization. **Tasks**: 1. Write a program to create a child process using fork() and execute another program using exec(). 2. Implement FCFS (First Come First Serve) and SJF (Shortest Job First) scheduling algorithms. 3. Implement Round Robin and Priority (preemptive) scheduling. 4. Use getpid() and getppid() to print process IDs of parent and child.	POSIX Semaphores, C/C++	LO3
5	**Title**: Process Synchronization **Objective**: Solve classic synchronization problems using semaphores, mutex locks, and interprocess communication. **Tasks**: 1. Implement the Producer-Consumer problem using semaphores. 2. Implement the Readers-Writers problem using mutex locks. 3. Implement process synchronization using message queues. 4. Demonstrate interprocess communication using shared memory.	Python, C++ Console Program	LO3
6	**Title**: Deadlock Detection & Avoidance **Objective**: Implement deadlock detection and avoidance algorithms. **Tasks**: 1. Implement the Banker's Algorithm for deadlock avoidance. 2. Implement a program to detect deadlocks using a resource allocation graph. 3. Solve the Dining Philosopher's problem using semaphores. 4. Simulate a scenario where deadlock occurs and recover from it.	EduPack Memory Management Simulator, Python	LO4
7	**Title**: Memory Management & Page Replacement **Objective**: Simulate memory partitioning, dynamic allocation, and page replacement algorithms. **Tasks**: 1. Simulate Memory Partitioning: MVT (Multiple Variable Tasks) and MFT (Multiple Fixed Tasks). 2. Implement Best Fit, First Fit, and Worst Fit dynamic memory allocation. 3. Compare FIFO and LRU page replacement algorithms.	Page Replacement Simulator, MATLAB, Python	LO4

	4. Simulate memory allocation and fragmentation in an OS.		
8	**Title**: Virtual Memory & Demand Paging **Objective**: Simulate demand paging, page faults, and swapping in virtual memory. **Tasks**: 1. Simulate demand paging and page faults in virtual memory. 2. Implement LRU, FIFO, and Optimal page replacement policies. 3. Write a program to calculate page fault rate for different page sizes. 4. Demonstrate swapping of processes in and out of memory.	Disk Scheduling Simulator, Java, Python	LO5
9	**Title**: File & Disk Management **Objective**: Implement file allocation strategies, disk scheduling algorithms, and analyze file systems. **Tasks**: 1. Implement file allocation strategies: Sequential, Indexed, and Linked. 2. Simulate file organization for a multi-level directory structure. 3. Implement disk scheduling algorithms: FCFS, SCAN, and C-SCAN. 4. Analyze file system structures of Linux (ext4) and Windows (NTFS).	C, Java File System Simulator	LO5
10	**Title**: Distributed Systems & Socket Programming and types of OS. **Objective**: Develop client-server systems and simulate distributed mutual exclusion. **Tasks**: 1. Implement a basic client-server message passing system using sockets. 2. Develop an RPC (Remote Procedure Call) application for data exchange. 3. Implement inter-process communication using named pipes (FIFO). 4. Simulate a distributed mutual exclusion mechanism.	NS3, Mininet, Python Sockets	LO6

Suggested List of Experiments

Experiment No.	**Experiment Title**	**Mapped LO**
1	Linux Commands & System Calls	LO1
2	Shell Scripting & Process Information	LO1
3	Linux API Implementation	LO3
4	Process Management - System Calls & Scheduling	LO3
5	Process Synchronization	LO3

6	Deadlock Detection & Avoidance	LO4
7	Memory Management & Page Replacement	LO4
8	Virtual Memory & Demand Paging	LO5
9	File & Disk Management	LO5
10	Distributed Systems & Socket Programming	LO6
11	Case study on Real time OS and Android OS.	LO6

Textbooks:

1. A. Silberschatz, P. Galvin, G. Gagne, *Operating System Concepts*, 10th ed., Wiley, 2018.
2. W. Stallings, *Operating Systems: Internal and Design Principles*, 9th ed., Pearson, 2018.
3. A. Tanenbaum, *Modern Operating Systems*, 4th ed., Pearson, 2015.

Reference Books:

1. N. Chauhan, *Principles of Operating Systems*, 1st ed., Oxford University Press, 2014.
2. A. Tanenbaum, A. Woodhull, *Operating System Design and Implementation*, 3rd ed., Pearson.
3. R. Arpaci-Dusseau, A. Arpaci-Dusseau, *Operating Systems: Three Easy Pieces*, 1st ed., CreateSpace Independent Publishing Platform, 2018.

Online References:

1. https://nptel.ac.in/courses/106105151
2. https://swayam.gov.in/
3. https://www.coursera.org/
4. https://riot-os.org/

Assessment and Evaluation

Term Work:

The term work should include a minimum 10 of experiments covering all Lab Outcomes. At least 02 assignments covering the entire syllabus based on modules of subject **ITC403,** and practical of “Operating Systems Lab”. The assignments should be students’ centric, and an attempt should be made to make assignments more meaningful, interesting and innovative. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks:

25 Marks (Total Marks) =15 Marks (Experiment) + 05 Marks (Assignments) + 05 Marks (Attendance)

The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

End-semester Practical and oral exams will be held based on the above and modules of subject **ITC403** syllabus and will be conducted as End Semester Examination Practical (ESEP).

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL403	Internet Programming Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITL403	Internet Programming Lab	--	--	--	25	25	50

Prerequisite:

1. FEL205 Object-Oriented Programming Methodology Lab

Program Outcomes addressed:

1. PO1: Engineering Knowledge

2. PO2: Problem Analysis

3. PO3: Design/development of solutions

4: LO5: Modern Tool Usage

Lab Objectives:

1. **Understand** the fundamental concepts of web development, including front-end and back-end technologies.
2. **Apply** HTML, CSS, and JavaScript to design responsive and interactive web pages.
3. **Analyze** client-server architecture and implement RESTful APIs using Node.js and Express.js.
4. **Evaluate** and integrate database technologies like MongoDB and PostgreSQL for backend development.
5. **Develop** full-stack applications using React and React Native with user authentication and state management.
6. **Deploy** web applications on cloud platforms like AWS, GCP, Vercel, and Heroku.

Lab Outcomes (LOs):

After successful completion of the course, students will be able to:

1. **Understand** the fundamental concepts of front-end technologies, including HTML, CSS, and Bootstrap, to create structured and styled web pages.
2. **Apply** JavaScript ES6 for dynamic web page development, including event handling, form validation, and asynchronous programming.
3. **Develop** a RESTful API using Node.js and Express.js to handle client-server communication.
4. **Integrate** MongoDB and PostgreSQL databases to store, retrieve, and manipulate data efficiently.
5. **Design and Implement** full-stack applications using React and React Native with proper state management and user authentication.
6. **Deploy** full-stack web applications on cloud platforms such as AWS, GCP, Vercel, and Heroku.

Detailed Syllabus

Sr No	Module	Detailed Content	Lab Outcome (LO)
1	**Frontend - HTML5**	Elements, Attributes, Head, Body, Hyperlink, Formatting, Images, Tables, Lists, Frames, Forms, Multimedia	LO1
2	**Frontend - CSS3 & Bootstrap**	CSS Syntax, Inclusion, Colors, Background, Fonts, Tables, Lists, CSS3 Selectors, Pseudo-classes, Pseudo-elements. Bootstrap: Grid System, Forms, Buttons, Navbar, Breadcrumb, Jumbotron	LO1
3	**Frontend - JavaScript ES6**	Variables, Operators, Conditions, Loops, Functions, Events, Classes & Objects, Inheritance, Error Handling, Validations, Arrays, Strings, Date, Promises, async/await	LO2
4	**Frontend - React**	Installation and Configuration, JSX, Components, Props, State, Forms, Events, Routers, Refs, Keys	LO5
5	**Mobile Development - React Native**	Environment Setup, Sample "Hello World" Example, Components, Navigation, State Management, API Calls	LO5
6	**Backend - Node.js & Express.js**	Installation and Configuration, Callbacks, Event Loop, Creating Express App, Middleware, REST API Development	LO3
7	**Database - MongoDB & PostgreSQL**	Introduction to NoSQL & SQL, CRUD Operations, Data Modeling, Indexing, Relationships, Aggregation	LO4
8	**Authenticatio**	JWT, OAuth, Session-based Authentication, Role-based Access Control	LO5

	n & Security	(RBAC), Encryption & Hashing	
9	**Hosting**	Hosting Web Applications on AWS, GCP, Vercel, and Heroku	LO6

Updated List of Experiments with LO Mapping

Sr No	**Experiment Title**	**LO Mapping**
1	Create a simple webpage using HTML5 elements and attributes.	LO1
2	Design a responsive webpage using CSS3 and Bootstrap.	LO1
3	Implement JavaScript functions for form validation and event handling.	LO2
4	Build a React application with functional components and props.	LO5
5	Create a simple to-do list app using React with state management.	LO5
6	Develop a basic mobile app using React Native with navigation.	LO5
7	Setup a Node.js server and create a basic REST API using Express.js.	LO3
8	Connect a Node.js application to MongoDB and perform CRUD operations.	LO4
9	Implement user authentication in a full-stack application using JWT.	LO5
10	Host a full-stack web application on AWS/GCP/Vercel/Heroku.	LO6
11	Implement role-based access control (RBAC) and secure API endpoints.	LO5
12	Build a file upload system in a React + Node.js application using Multer.	LO4
13	Integrate third-party APIs (e.g., weather API, payment gateway) into a full-stack application.	LO3
14	Implement server-side rendering (SSR) using Next.js for performance optimization.	LO5
15	Deploy a full-stack e-commerce application with a database and authentication.	LO6

Textbooks:

1. **Jon Duckett** – *HTML & CSS: Design and Build Websites*, John Wiley & Sons, 2011.
2. **Jon Duckett** – *JavaScript and JQuery: Interactive Front-End Web Development*, John Wiley & Sons, 2014.
3. **Robin Nixon** – *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*, O'Reilly Media, 2018.
4. **Ethan Brown** – *Learning JavaScript Design Patterns*, O'Reilly Media, 2021.
5. **Carlos Santana Roldán** – *React 18 Design Patterns and Best Practices*, Packt Publishing, 2023.
6. **Kyle Simpson** – *You Don't Know JS (book series)*, O'Reilly Media, 2015.

Reference Books:

1. **Eric Elliott** – *Programming JavaScript Applications*, O'Reilly Media, 2014.
2. **Flavio Copes** – *The React Handbook*, self-published, 2020.
3. **David Flanagan** – *JavaScript: The Definitive Guide*, O'Reilly Media, 2020.
4. **Mark Lutz** – *Learning Python*, O'Reilly Media, 2013.
5. **Narasimha Karumanchi** – *Data Structures and Algorithms Made Easy in JavaScript*, CareerMonk Publications, 2021.

Web References:

1. https://developer.mozilla.org/
2. https://www.w3schools.com/
3. https://react.dev/
4. https://nodejs.org/en/docs
5. https://expressjs.com/
6. https://www.mongodb.com/docs/
7. https://www.postgresql.org/docs/
8. https://lab.github.com/
9. https://aws.amazon.com/free/
10. https://vercel.com/docs

Term Work:

Term Work: Term Work shall consist of at least 10 to 12 practical list based on the above list. Also, Term work Journal must include at least 2 assignments.

Term work Marks:

25 Marks (Total Marks) =15 Marks (Experiment) + 05 Marks (Assignments) + 05 Marks attendance)

The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

End-semester Practical and oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP).

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL404	Skill Lab (MAD)	--	2*+2	--	--	02	-	02

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITL404	Skill Lab(MAD)	--	--	--	25	25	50

Prerequisite:

1. FEL205 Object Oriented Programming Methodology Lab
2. ITL303 Skill Lab Python

Program Outcomes addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/development of solutions

Lab Objectives:

1. Understand the fundamentals of the Flutter framework and its application in cross-platform mobile development.
2. Design and develop interactive Flutter applications using widgets, layouts, gestures, and animations.
3. Integrate backend services such as Firebase and Fire store to create production-ready Flutter applications.
4. Explore Progressive Web App (PWA) concepts and frameworks to build responsive and offline-capable web applications.
5. Develop responsive user interfaces using modern web technologies like jQuery Mobile, Material UI, Angular UI, and React UI.
6. Implement and deploy PWAs using service workers, web app manifests, and static hosting solutions like GitHub Pages.

Lab Outcomes:

After successful completion of the course, students will be able to:

1. Apply the Flutter framework for cross platform mobile application development
2. Design & develop interactive flutter applications using widgets, layouts, gestures and animation
3. Build production ready flutter application by integrating backend services & deploying them on Android /iOS platform

4. Design the core concepts of Progressive Web Apps (PWAs) and their requirements, including service workers, web app manifests, and HTTPS
5. Design and develop responsive user interfaces using modern web technologies like jQuery Mobile, Material UI, Angular UI, and React UI.
6. Implement, test, and deploy PWAs using service workers, web app manifests, and static hosting solutions like GitHub Pages.

Detailed Syllabus

Sr. No	Module	Detailed Contents	Hours	LO Mapping
1	Basics of Flutter Programming	Introduction to Flutter, Dart Basics, Widget Tree, and Element Tree. **Tools Used:** Flutter SDK, Android Studio/VS Code	02	LO1
2	Developing Flutter UI: Widgets, Layouts, Gestures, Animation	Using Common Widgets (SafeArea, AppBar, Column, Row, Container, Buttons, Text, etc.)., Building Layouts (Types of layout widgets), Applying Gestures (Gesture Detector, Draggable, DragTarget), Adding Animation (AnimatedContainer, AnimatedCrossFade, AnimationController) **Tools Used:** Flutter SDK, Android Studio/VS Code	06	LO2
3	Creating Production-Ready Apps	Working with Files (Reading/Writing files, JSON), Using Firebase with Flutter (Firestore, Firebase Authentication), Testing and Deploying Flutter Apps. **Tools Used:** Flutter SDK, Firebase, Android Studio	04	LO3
4	Introduction to Progressive Web Apps (PWAs)	PWA Concepts (Service Workers, Web App Manifest, HTTPS), PWA Frameworks and Tools. **Tools Used:** VS Code, Web Browser	02	LO4
5	Creating Responsive UI	Responsive Web Design (jQuery Mobile, Material UI, Angular UI, React UI). **Tools Used:** VS Code, Web Browser	06	LO5
6	Web App Manifest & Service Workers	Web App Manifest (Properties, Installation), Service Workers (Lifecycle, Caching, Offline Functionality), Deploying PWAs (GitHub Pages, Lighthouse Analysis) **Tools Used:** GitHub, VS Code, Web Browser, Google Lighthouse	06	LO6

Suggested List of Experiments:		
Sr. No.	Title of Experiments	LO Mapped
1	Install and configure the Flutter environment.	LO1
2.	Design a basic Flutter app using Dart language	LO1
3	Create a "Hello World" Flutter app.	LO1
4	Design a Flutter UI by including common widgets.	LO2
5	Design a layout of a Flutter app using layout widgets.	LO2
6	Apply navigation, routing in a Flutter app	LO2
7	Build an interactive app by including Flutter gestures and animations.	LO2
8	Include files and JSON data in a Flutter app	LO3
9	Integrate Firebase and Firestore backend services into a Flutter app.	LO3
10	Test and deploy a production-ready Flutter app on Android/iOS platforms.	LO3
11	Write metadata of an e-commerce PWA in a web app manifest file.	LO4
12	WAP to Code and register a service worker for an e-commerce PWA.	LO6
13	Create a responsive user interface for an e-commerce application using jQuery Mobile/Material UI/Angular UI/React UI.	LO5
14	Implement service worker events like fetch, sync, and push for an e-commerce PWA.	LO6
15	Study and implement the deployment of an e-commerce PWA to GitHub Pages	LO6
16	Use Google Lighthouse PWA analysis tool to test the PWA functioning	LO6
17	Deploy an e-commerce PWA using an SSL-enabled static hosting solution	LO6

Assignment List:

Assignment 1:

1. Study the basics of the Dart language and design a basic Flutter app. (LO1)
2. Include files and JSON data in a Flutter app. (LO3)
3. Build an interactive app by including Flutter gestures and animations. (LO2)

Assignment 2:

1. Create a responsive user interface for an e-commerce application using jQuery Mobile/Material UI/Angular UI/React UI. (LO5)
2. Implement a service worker for an e-commerce PWA and enable offline functionality. (LO4, LO6)

Text Books:

1. Beginning Flutter: A Hands-on Guide to App Development, Marco L. Napoli, Wiley, 2020.
2. Beginning App Development with Flutter: Create Cross-Platform Mobile Apps, Rap Payne, 2019.
3. Progressive Web Application Development by Example: Develop Fast, Reliable, and Engaging User Experiences for the Web, Packt Publishing Limited, 2018.
4. Building Progressive Web Apps, O'Reilly, 2017.
5. Progressive Web Apps with Angular: Create Responsive, Fast, and Reliable PWAs Using Angular, Apress, 1st ed., 2019.

Reference books:

1. Flutter in Action, Eric Windmill, Manning, 2019.
2. Google Flutter Mobile Development Quick Start Guide, Packt, 2019.
3. Learning Progressive Web Apps: Building Modern Web Apps Using Service Workers, Addison Wesley Professional, 2020.

Online References:

1. https://docs.flutter.dev/codelabs
2. https://www.tutorialpoints.com/flutter/index.htm
3. https://www.w3schools.com/css/css_rwd_intro.asp
4. https://angular.io/docs
5. https://flaviocopes.com/service-workers
6. https://blog.logrocket.com/how-to-build-a-progressive-web-app-pwa-with-node-js/

Term Work:

Term Work: Term Work shall consist of at least 10 to 12 practical list based on the above list. Also, Term work Journal must include at least 2 assignments.

Term work Marks:

25 Marks (Total Marks) =15 Marks (Experiment) + 05 Marks (Assignments) + 05 Marks attendance)

The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

End-semester Practical and oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP).

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL405	Value Education (UHV)	--	04	--	--	02	-	02

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITL405	Value Education (UHV)	--	--	--	50	--	50

Pre-requisite:

1. Not Applicable

Program Outcome mapped:

1. **PO1**: Engineering Knowledge.
2. **PO2**: Problem Analysis.
3. **PO3**: Design/Development of Solutions.
4. **PO4**: Conduct Investigations of Complex Problems.
5. **PO5**: Engineering Tool Usage.
6. **PO6**: The Engineer and The World.
7. **PO7**: Ethics.
8. **PO8**: Individual and Collaborative Teamwork.
9. **PO9**: Communication.
10. **PO10**: Project Management and Finance.
11. **PO11**: Life-Long Learning.

Course Objectives:

1. **To introduce the fundamental concepts of human values**, including intrinsic and extrinsic values, and their relevance to personal and professional development in the context of IT engineering.
2. **To explore the principles of Universal Human Values (UHV)**, emphasizing self-awareness, self-exploration, and the application of tools like the JOHARI window and SWOT analysis in the IT profession.
3. **To study the different levels of harmony**—within oneself, in the family, society, and nature—and apply these concepts to achieve a balanced and fulfilling life, especially in the fast-paced IT industry.
4. **To comprehend the key aspects of professional ethics in IT**, including ethical standards, work ethics, and moral issues such as data privacy, cybersecurity, and AI ethics.
5. **To develop foundational values** such as integrity, impartiality, nonpartisanship,

and objectivity, and cultivate empathy, tolerance, and compassion in both personal and professional contexts, particularly in IT-related decision-making.

6. **To integrate human values into IT practices**, focusing on ethical decision-making, sustainable technology development, and responsible innovation.

Course Outcomes:
After successful completion of the course, students will be able to:

1. **Understand and explain** the basic concepts of human values and their significance in personal and professional contexts, particularly in the IT industry.
2. **Explore and internalize** human values to guide personal behavior and professional conduct in IT roles such as software development, data analysis, and cybersecurity.
3. **Analyze and apply** the concept of harmony at various levels of existence to achieve a balanced life, even in high-pressure IT environments.
4. **Identify and evaluate** ethical issues in the IT profession, including data privacy, cybersecurity, AI ethics, and intellectual property rights, using appropriate ethical theories and standards.
5. **Demonstrate and Uphold** integrity and ethical principles in professional and public service contexts, fostering empathy and compassion in IT projects that impact society.
6. **Integrate and implement** human values into IT practices, ensuring that technology development aligns with ethical, social, and environmental considerations.

Course Modules and Topics:

Module No.	Unit No.	Topics	Mapped Learning Outcome (LO)
1.0		Introduction to Human Values and Their Relevance in IT	LO1
	1.1	Definition, Intrinsic & Extrinsic values, Shalom Schwartz's Theory of Basic Human Values, Value education: Need, Basic Guidelines and Scope, Self-exploration, Happiness and Prosperity, Harmony, Self-awareness: JOHARI window and SWOT analysis	LO1
	1.2	Ethical Challenges in IT: Data privacy, cybersecurity, AI ethics, and intellectual property rights	LO4
2.0		Understanding Human Beings and Harmony at Various Levels of Existence	LO3

Module No.	Unit No.	Topics	Mapped Learning Outcome (LO)
	2.1	Human beings as a combination of the conscious 'I' and material body, Abraham Maslow's Hierarchy of Needs, Classification between I & Body, Co-existence, Harmony in Self: Swasthya and Sanyama	LO3
	2.2	Harmony in the Family -- Understanding Values in Human Relationships, Differentiation in relationships, Values in relationships	LO3
	2.3	Harmony in the Society -- From Family order to World Family Order, Comprehensive Human Goal, Harmony in Nature -- Understanding the Interconnectedness and Mutual Fulfilment, Understanding the Four Orders of Nature	LO3, LO6
3.0		Professional Ethics in IT	LO4
	3.1	Definition, Characteristics, Profession, Professionalism, Morality, Moral issues in the IT profession, Understanding Ethics, Ethical Standards, Work Ethics, Engineering Ethics	LO4
	3.2	Types of Inquiries, Kohlberg's Theory, Heinz Dilemma, Gilligan's Theory, and Ethical Theories	LO4
	3.3	Ethical Challenges in IT: Data privacy, cybersecurity, AI ethics, and intellectual property rights	LO4
4.0		Ethics, Integrity, and Aptitude in IT	LO5
	4.1	Essence, determinants, and consequences of ethics in human actions, Dimensions of ethics, Ethics in private and public relationships	LO5
	4.2	Key contributions from Indian and global moral thinkers and philosophers, emphasizing integrity, impartiality, and non-partisanship in professional settings	LO5
	4.3	Upholding objectivity and dedication to public service, Cultivating empathy, tolerance, and compassion, with a focus on their application in IT and public welfare	LO5
5.0		Understanding Harmony in Nature and Sustainable IT Practices	LO6
	5.1	Concept of harmony in Nature: Meaning of harmony in nature, Disharmony with Nature -- causes, Implications of disharmony with nature	LO6

Module No.	Unit No.	Topics	Mapped Learning Outcome (LO)
	5.2	Maintaining harmony with nature: Harmony through mutual fulfilment of the four orders in nature, Harmony through symbiotic relationship with nature, Achieving competence in maintaining harmony with nature in professional life	LO6
	5.3	Sustainable IT Practices: Green computing, energy-efficient algorithms, and eco-friendly technology development	LO6
6.0		Practicum Project -- Community Engagement and IT for Social Good	LO2, LO5, LO6
	6.1	Students carry out a community engagement project to benefit the local community through IT-based initiatives (e.g., developing apps for social causes, organizing digital literacy camps, or creating awareness about cybersecurity).	LO2, LO5, LO6
	6.2	Students write a reflective report on how the understanding of universal human values has been integrated into their IT project.	LO5, LO6

Textbooks:

1. **Naagarazan, R. S.** *A Textbook on Professional Ethics and Human Values*. 4th Edition. New Age International Publishers, 2021.
2. **Gaur, R.R., Sangal, R., & Bagaria, G.P.** *A Foundation Course in Human Values and Professional Ethics*. 3rd Edition. Excel Books, 2019.
3. **Khosla, Vaishali R., & Bhagat, Kavita.** *Human Values and Professional Ethics*. 2nd Edition. Macmillan Education, 2020.
4. **Harris, C.E., Pritchard, M.S., & Rabins, M.J.** *Engineering Ethics: Concepts and Cases*. 6th Edition. CENGAGE Learning, 2019.
5. **Murthy, PSR.** *Indian Culture, Values and Professional Ethics*. 4th Edition. BS Publications, 2022.

Reference Books:

1. **Kumar, Niraj.** *Lexicon for Ethics, Integrity & Aptitude for IAS General Studies Paper IV*. 2nd Edition. McGraw Hill Education, 2023.
2. **Subba Rao, G., & Roy Chowdhury, P. N.** *Ethics, Integrity & Aptitude*. 3rd Edition. McGraw Hill Education, 2020.

Online References:

1. https://fdp-si.aicte-india.org/index.php
2. https://example.com/

Course Assessment:

Internal Assessment Method (With Rubrics)

The internal assessment will consist of **Continuous Internal Assessment (CIAP) = 50 marks** based on **Assignments, Case Studies, Presentations, and Practicum Projects**.

Assessment Component	**Weightage (%)**	**Evaluation Criteria (Rubrics)**
Assignment on Human Values	20%	- Excellent (5): Demonstrates deep understanding with real-life examples - Good (4): Good understanding with relevant examples - Satisfactory (3): Basic understanding with minimal examples - Needs Improvement (2): Partial understanding with errors - Poor (1): Little to no understanding
Case Study on Ethical Issues in IT	20%	- Excellent (5): In-depth analysis with ethical theories and solutions - Good (4): Covers major ethical aspects with examples - Satisfactory (3): Identifies ethical concerns with some analysis - Needs Improvement (2): Limited understanding with minor errors - Poor (1): Lacks analysis and ethical reasoning
Presentation on Sustainability in IT	20%	- Excellent (5): Well-structured, engaging, innovative ideas - Good (4): Clear and logical presentation with good insights - Satisfactory (3): Covers major points but lacks depth - Needs Improvement (2): Some points missing, lacks clarity - Poor (1): Unstructured, lacks coherence
Reflection Report on Practicum Project	20%	- Excellent (5): Thoughtful reflection, well-articulated impact - Good (4): Covers personal learning and impact clearly

		- Satisfactory (3): General reflection with limited depth - Needs Improvement (2): Superficial understanding - Poor (1): Minimal effort, lacks insight
Participation in Discussion & Engagement	20%	- Excellent (5): Actively participates, provides insightful contributions - Good (4): Engaged, contributes relevant thoughts - Satisfactory (3): Participates but with limited contribution - Needs Improvement (2): Rarely participates, minimal effort - Poor (1): No participation

Examples of Practicum Projects for Community Engagement and IT for Social Good

The practicum project aims to encourage students to apply **Universal Human Values (UHV)** and **Ethics in IT** to solve real-world societal challenges. Below are some project ideas along with explanations of how they integrate **human values and ethics**:

Digital Literacy Program for Underprivileged Communities

Objective: Create and conduct workshops to educate marginalized communities about basic computer skills, cybersecurity awareness, and digital payments.

Implementation:

- Design an easy-to-understand curriculum on digital literacy.
- Conduct workshops/webinars in rural schools or community centers.
- Develop a simple mobile/web application for learning digital skills.
- Educate participants about data privacy, cyber threats, and ethical internet use.

Human Values & Ethics Integration:

- **Empathy & Compassion** – Address digital divide and empower underprivileged individuals.
- **Integrity & Responsibility** – Teach ethical use of technology and responsible online behavior.
- **Public Welfare** – Ensure safe digital access for vulnerable communities.

AI-Based Cyberbullying Detection for Schools & Colleges

Objective: Develop an AI model to identify cyberbullying in chat messages and social media posts, ensuring a safer digital environment.

Implementation:

- Collect and train data on cyberbullying-related words & phrases.

- Implement a Natural Language Processing (NLP)-based chatbot to detect abusive content.
- Educate students on ethical social media behavior and reporting mechanisms.
- Partner with schools/colleges to deploy the model in their IT systems.

Human Values & Ethics Integration:

- **Respect & Non-Partisanship** – Encourage online respectful interactions.
- **Fairness & Objectivity** – Ensure non-biased AI in content moderation.
- **Safety & Privacy** – Protect users' personal data and identity.

Green Computing Awareness & E-Waste Management App

Objective: Develop an app to educate users on sustainable IT practices and provide an e-waste collection service.

Implementation:

- Create an app that guides users on green computing practices.
- Provide nearby e-waste collection centers and reward users for recycling.
- Conduct IT industry awareness campaigns on energy-efficient computing.
- Promote the use of renewable energy in data centers.

Human Values & Ethics Integration:

- **Environmental Sustainability** – Encourage eco-friendly IT solutions.
- **Social Responsibility** – Spread awareness about ethical e-waste disposal.
- **Harmony in Nature** – Minimize IT sector's negative impact on nature.

Cybersecurity Awareness Chatbot for Senior Citizens

Objective: Build a WhatsApp or Telegram chatbot that assists senior citizens in identifying and avoiding online scams, phishing, and frauds.

Implementation:

- Develop an AI chatbot that explains common online scams.
- Create step-by-step tutorials on safe internet banking and social media usage.
- Partner with local community centers and NGOs to spread awareness.
- Ensure chatbot provides real-time support and automated alerts.

Human Values & Ethics Integration:

- **Compassion & Empathy** – Assist vulnerable groups in safe internet use.
- **Integrity & Awareness** – Promote honest and secure online transactions.
- **Public Welfare** – Reduce cyber frauds targeting elderly people.

AI Ethics Awareness in IT Companies & Colleges

Objective: Develop an interactive website or mobile app to educate IT professionals and students on ethical AI usage and biases in AI systems.

Implementation:

- Provide interactive case studies on AI bias, privacy, and ethical dilemmas.

- Conduct quiz-based learning to test AI ethical understanding.
- Collaborate with IT professionals and faculty to design real-world scenarios.
- Ensure alignment with global AI ethics standards (e.g., IEEE, EU AI Act).

Human Values & Ethics Integration:

- **Integrity & Fairness** – Ensure unbiased AI algorithms.
- **Public Interest** – Educate developers on responsible AI implementation.
- **Transparency** – Promote explainable and fair AI decision-making.

Mobile App for Volunteer & Donation Matching

Objective: Develop a volunteer-matching platform that connects IT professionals and students with social organizations in need of technical assistance.

Implementation:

- Allow users to register their skills (app development, cybersecurity, etc.).
- Connect them with NGOs or community projects that require IT support.
- Enable secure crowdfunding and donation tracking for transparency.
- Promote projects focused on digital inclusion and education.

Human Values & Ethics Integration:

- **Social Responsibility** – Encourage IT professionals to give back to society.
- **Transparency & Trust** – Maintain fair donation tracking.
- **Empathy & Compassion** – Align IT skills with community development.

Ethical Hacking & Cybersecurity Training for Students

Objective: Conduct a hands-on ethical hacking workshop to educate students on ethical penetration testing and cybersecurity best practices.

Implementation:

- Develop training modules on ethical hacking, cryptography, and network security.
- Conduct capture-the-flag (CTF) cybersecurity challenges for hands-on learning.
- Educate students on responsible disclosure of vulnerabilities.
- Partner with cybersecurity firms for internships and projects.

Human Values & Ethics Integration:

- **Ethical Responsibility** – Train IT students to prevent cyber crimes.
- **Accountability** – Promote responsible ethical hacking practices.
- **Public Safety** – Improve cybersecurity awareness in college networks.

AI-Powered Sign Language Recognition System

Objective: Develop an AI-based sign language recognition system to help hearing-impaired individuals communicate using real-time gesture recognition.

Implementation:

- Train a machine learning model on Indian Sign Language (ISL).
- Develop a mobile/web app that converts sign language gestures into text/speech.

- Partner with special education institutes and NGOs for deployment.
- Ensure open-source availability for future development.

Human Values & Ethics Integration:

- **Inclusion & Accessibility** – Bridge communication gaps for disabled individuals.
- **Fairness & Transparency** – Ensure AI is unbiased across different sign languages.
- **Social Welfare** – Enhance digital accessibility for differently-abled people.

- **Project Submission & Reflection Report**

After completing the practicum project, students will submit a reflective report covering:

1. **Project Objective & Problem Statement**
2. **Implementation Details & Challenges Faced**
3. **Human Values & Ethics Integrated**
4. **Impact Assessment & Learning Outcomes**
5. **Future Improvements & Scalability**

Evaluation Rubric:

- **Excellent (5):** Clear objectives, strong ethical integration, significant social impact.
- **Good (4):** Good ethical integration, minor improvement areas.
- **Satisfactory (3):** Basic implementation, lacks depth in ethical application.
- **Needs Improvement (2):** Minimal social impact, weak ethical connection.
- **Poor (1):** Unclear project execution, little relevance to human values.

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITM401	**Mini Project 1B (Python)**	--	02#	--	--	1	--	1

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
ITM401	**Mini Project 1B (Python)**	--	--	--	25	25	50

# Indicates workload of a learner (Not Faculty) for Mini Project 1A. Faculty Load: 1 hour per week per four groups.

Pre-requisite:

1. FEL103, FEL205: Knowledge of Basic Programming

Program Outcome mapped:

1. PO1: Engineering Knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.
7. PO7: Ethics.
8. PO8: Individual and Collaborative Teamwork.
9. PO9: Communication.
10. PO10: Project Management and Finance.
11. PO11: Life-Long Learning.

Course Objectives:

1. To acquaint students with the process of identifying societal or research needs and converting them into web-based solutions.
2. To familiarize students with web development technologies (HTML, CSS, JavaScript, etc.).
3. To develop skills in designing and implementing web applications for real-world problems.
4. To inculcate self-learning and research capabilities in web technology.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Identify and Analyse web-based problems based on societal, research, or environmental needs and apply web development skills to develop appropriate solutions.
2. Demonstrate teamwork and interpersonal skills by collaborating effectively on web-based projects, showcasing leadership and coordination abilities.
3. Design and implement responsive, user-friendly web applications while adhering to standard web development practices and norms.
4. Evaluate the societal and environmental impact of web solutions and incorporate sustainable practices in web development.
5. Communicate web-based solutions effectively through structured written reports, presentations, and demonstrations.
6. Demonstrate self-learning and adaptability by exploring and integrating new web technologies, tools, and frameworks for lifelong learning and professional growth.

Guidelines for Mini Project 1B (Web Technology):

- **Group Formation:** 3-4 students per group.
- **Problem Identification:** Identify a societal or research need and convert it into a web-based problem statement.
- **Implementation Plan:** Prepare a Gantt/PERT/CPM chart for weekly activities.
- **Log Book:** Maintain a log book to record weekly progress, verified by the supervisor.
- **Self-Learning:** Focus on self-learning with minimal guidance from the supervisor.
- **Solution Development:** Propose multiple solutions, select the best one, and implement it using web technologies.
- **Validation:** Validate the solution with proper justification and compile a report as per University of Mumbai guidelines.
- **Duration:** The project can span one or two semesters based on complexity.

General Guidelines:

- **Review Committee:** A committee will evaluate progress through two reviews per semester.
- **Term Work Marks Distribution:**
  - Log Book (Guide/Supervisor): 10 marks
  - Review Committee: 10 marks
  - Quality of Project Report: 5 marks
- **Final Assessment:** Presentation and demonstration of the working model to a panel of internal and external examiners.

- **Publication:** Students are encouraged to publish their work in conferences or student competitions.

This syllabus ensures that students gain hands-on experience in Python and Web Technology while addressing real-world problems, fostering innovation, teamwork, and lifelong learning.

Term Work Marks Distribution:

1. Log Book (Guide/Supervisor): 10 Marks

Criteria	Marks Allocation	Description
Problem Identification	**2 Marks**	- Clarity and relevance of the problem statement.
		- Alignment with societal, research, or environmental needs.
Implementation Plan	**2 Marks**	- Quality of Gantt/PERT/CPM chart and planning.
		- Realistic and well-structured timeline for activities.
Weekly Progress and Updates	**3 Marks**	- Regular updates in the log book with detailed progress.
		- Supervisor verification of weekly activities.
Self-Learning and Adaptability	**3 Marks**	- Evidence of self-learning and minimal guidance from the supervisor.
		- Exploration of new tools, technologies, or frameworks.

2. Review Committee: 10 Marks

Criteria	Marks Allocation	Description
Innovation and Creativity	**2 Marks**	- Innovativeness in proposing web-based solutions.
		- Creativity in addressing the problem.
Functionality and Responsiveness	**3 Marks**	- Functionality of the web application (features, usability, and performance).
		- Responsiveness across devices (desktop, tablet, mobile).
Use of Web Technologies	**3 Marks**	- Effective use of web development tools, frameworks, and technologies.
		- Adherence to standard coding practices and norms.
Societal Impact and	**2 Marks**	- Analysis of societal and environmental impact.

Sustainability		- Incorporation of sustainable practices in the solution.

3. Quality of Project Report: 5 Marks

Criteria	**Marks Allocation**	**Description**
Clarity and Completeness	**2 Marks**	- Clear and detailed explanation of the problem, solution, and implementation. - Adherence to University of Mumbai guidelines for project reports.
Documentation and Communication	**2 Marks**	- Quality of written documentation (structure, grammar, and technical accuracy). - Effective use of diagrams, charts, and visuals.
Oral Presentation	**1 Mark**	- Clarity and confidence during the presentation. - Ability to answer questions and justify the solution.

Summary of Marks Distribution:

Component	**Mark**
Log Book (Guide/Supervisor)	10
Review Committee	10
Quality of Project Report	5
Total	**25**

Key Features:

1. **Log Book (10 Marks)**: Focuses on problem identification, planning, progress tracking, and self-learning.
2. **Review Committee (10 Marks)**: Evaluate innovation, functionality, use of technologies, and societal impact.
3. **Project Report (5 Marks)**: Assesses the quality of documentation, clarity, and oral presentation.

ESEP Evaluation Criteria for Project Presentation (25 Marks)

Criteria	**Marks Allocation**	**Description**
Problem Identification & Analysis	**5 Marks**	- Clarity and relevance of the problem statement. - Justification of the problem's significance in societal, research, or industry needs.
Technical Implementation & Solution	**5 Marks**	- Use of Python programming concepts and frameworks. - Efficiency and correctness of the solution.

Innovation & Creativity	**3 Marks**	- Novelty and uniqueness of the solution. - Creative approach in problem-solving.
Functionality & Usability	**3 Marks**	- Working model demonstration with minimal errors. - Usability, efficiency, and user experience of the project.
Presentation & Communication Skills	**5 Marks**	- Clarity, confidence, and structure of the presentation. - Effective demonstration and ability to answer queries.
Project Report Quality	**2 Marks**	- Well-structured documentation with appropriate diagrams, charts, and visuals.
Societal Impact & Sustainability	**2 Marks**	- Explanation of the project's impact on society or environment. - Consideration of sustainability and ethical practices.

Final Mark Distribution:

- Project Presentation & Demonstration → 15 Marks
- Project Report & Documentation → 5 Marks
- Societal & Sustainable Impact → 5 Marks
- Total → 25 Marks

Internal Assessment:

For 03 credit - 80 marks subject

Assessment consists of one Mid Semester Examination (MSE) of 20 marks and In Semester Examination (ISE) of 20 marks. The MSE to be conducted based on 50 % syllabus with duration of one hour.

For 02 credit - 60 marks subject

Assessment consists of one Mid Semester Examination (MSE) of 15 marks and In Semester Examination (ISE) of 15 marks. The MSE to be conducted based on 50 % syllabus with duration of one hour.

In Semester Examination (ISE)

ISE 20 marks = 05 marks attendance +15 marks for Activities.

ISE 15 marks = 05 marks attendance +10 marks for Activities.

The Rubrics for activities are as follows. The activities will be decided by course in charge and approved by HoD.

Sr. No	Rubrics	Marks
1	Multiple Choice Questions (Quiz)	05 Marks
2	Literature review of papers/journals	05 Marks
3	Participation in event/ workshop/ talk / competition followed by small report and certificate of participation relevant to the subject	05 Marks
4	Extra Experiments/ Virtual Lab	05 marks
5	Content beyond syllabus presentation	05 marks
6	Wins in the event/competition/hackathon pertaining to the course	10 Marks
7	Case study, Presentation, group discussion, technical debate on recent trends in the said course	10 Marks
8	NPTEL/ Coursera/ Udemy/any MOOC Certificate course for 4 weeks or more	10 Marks
9	Creating Proof of Concept	10 Marks
10	Mini Project /	10 Marks
11	GATE Based Assignment test/Tutorials etc	10 Marks

*For sr.no.8, the date of certification exam should be within the term and in case a student is unable complete the certification, the grading has to be done accordingly.