SOMAIYA
VIDYAVIHAR
K J Somaiya Institute of Technology
An Autonomous Institute Permanently Affiliated to the University of Mumbai

Item No.: 4.A.1
A.C.: 05/07/2024

Autonomy Syllabus Scheme III (2023-24)

(As per NEP 2020 Guidelines)

for

Four Year Multidisciplinary

Bachelors of Technology (B.Tech.) Program

in Information Technology

with

Multiple Entry and Multiple Exit Options

Levels 4.5 - 6

(Second Year Effective from A.Y. 2024-25)

From the Principal's Desk:

To address the changing demands of the digital era, it is required to create a future-ready workforce that can navigate the complexities of an interconnected world, drive innovation, and contribute to the nation's growth. The **National Educational Policy 2020 (NEP 2020)** framed by the Government of India recommends a holistic, inclusive, and flexible approach to ensure equitable access to quality education across all levels, promote multidisciplinary research, and impart skill-based education with integration of technology. As per guidelines by the Department of Higher and Technical Education, Government of Maharashtra, the salient features of NEP 2020 aligned curriculum should include:

- Major (Core) Mandatory and Elective Courses
- Open Elective Courses
- Vocational and Skill Enhancement Courses
- Ability Enhancement Courses, Indian Knowledge System, and Value Education Courses
- Co-curricular Courses and Field Projects / Community Engagement Projects / Internship
- Multidisciplinary Minor Courses
- Option for Bachelor's Degree with Honours (based on Additional Credits)
- Option for Bachelor's Degree – Honours with Research (based on Additional Credits)
- Option for Bachelor's Degree with Double Minors (based on Additional Credits)
- Multiple Entry and Multiple Exit Options

Being an **autonomous institute** since the Academic Year 2021-22, **K. J. Somaiya Institute of Technology (KJSIT)**, has well-adapted newer approaches to reach higher levels of excellence in engineering education. Ahead of its time, the academic reforms at KJSIT have already addressed majority of these NEP 2020 aspects through its existing **Syllabus Scheme I, II, and II B** implemented under the academic autonomy. For a complete alignment with NEP 2020, the **KJSIT Autonomy Syllabus Scheme III** is introduced, to be effective from Academic Year 2023-24 across all the branches, progressively from First Year Engineering.

Specifically, the existing curriculum already comprise state-of-the-art **Major (Core) courses** in theory and practical. With an ideology that the root of innovation is 'interest', the curriculum offers wide range of Elective courses — grouped into **Major-related Electives** and **Inter-disciplinary / Open Electives**. At par with international engineering education, it follows a learner-centric approach as well as promotes MOOCs, where the students can choose to study courses concerning areas of their interests, and the same is continued in Scheme III.

Further, under the theme of "Learning by Doing", the existing curriculum includes Skill-Based Learning (SBL), Activity-Based Learning (ABL), and Technology-Based Learning (TBL) as eXposure (SAT) courses — that assure X factor in all the students of the institute. The SAT courses are practiced across the first three years of engineering, focusing on responsibilities towards society, problem-solving abilities, communication skills, ethics, leadership and teamwork, motivation for life-long learning, skills on emerging areas of technology, skills on different languages, etc. In the Syllabus Scheme III, these SAT courses are now aligned and offered as **Vocational Skill - SAT (VS - SAT) courses**, **Skill Enhancement - SAT (SE - SAT) courses**, **Ability Enhancement - SAT (AE - SAT) courses**, and **Value Education - SAT (VE - SAT) courses**.

Further, **Indian Knowledge System - SAT (IKS - SAT) course** is newly introduced in Scheme III that emphasizes on drawing insights from ancient wisdom to address modern challenges. Also, as an extension to the induction program for the First Year students, the introduced **Co-curricular - SAT (CC - SAT) course** aims to induct incumbents with the institutional practices, culture, and values, as well as encourage participation in co-curricular activities.

The component of **Project-Based Learning (PBL)** included in the Syllabus Scheme II is carried forward to Scheme III, wherein the students develop **Community Engagement / Field Projects** in Second, Third, and Last Year as Mini, Minor, and Major Projects respectively. Scheme III also retains the **Internship** component, offered with credits, to equip graduates with the industry trends, practices, and skills required at national and

global level. The duality of PBL and Internship enables student involvement in research, innovation, and entrepreneurship, which are the fulcrums of higher education.

As a new introduction in line with NEP 2020, the Syllabus Scheme III incorporates mandatory **Multidisciplinary Minor courses** in Innovation and Entrepreneurship, Biotechnology, IoT and Cloud Computing, Geographical Information System, Very Large Scale Integration (VLSI) and Artificial Intelligence. These courses promote interdisciplinary thinking and broaden the career prospects, enabling students to develop solutions to real-world problems by combining expertise from multiple domains.

Aligned with NEP 2020, the Scheme III retains the initiative taken through Scheme II / II B of offering **Honours courses** for students who are desirous of pursuing focused interest in 06 emerging areas of technology recognized by AICTE: Internet of Things, Artificial Intelligence & Machine Learning, Cyber Security, Virtual and Augmented Reality, Data Science, and Blockchain. These Honours courses correspond to high-end industry standards and offer multi-fold opportunities of specialization.

As per NEP 2020, the above curricular aspects of Four Years UG Engineering Programme shall be offered with **Multiple Entry and Multiple Exit options**, leading to the conferment of:

- **One Year UG Certificate in Technology:** Awarded after completing First Year of Engineering and acquiring additional 08 credits immediately after First Year.
- **Two Years UG Diploma in Technology:** Awarded after completing Second Year of Engineering and acquiring additional 08 credits immediately after Second Year.
- **Three Years Bachelor's Degree in Vocation (B.Voc.):** Awarded after completing Third Year of Engineering and acquiring additional 08 credits immediately after Third Year.
- **Four Years Bachelor's Degree in Technology (B.Tech.) with Multidisciplinary Minor:** Awarded after completing Fourth Year of Engineering.
- **Four Years Bachelor's Degree in Technology (B.Tech.) Honors with Multidisciplinary Minor:** Awarded after completing Fourth Year of Engineering and acquiring additional 18 credits through Honours courses in respective major discipline over Third & Fourth Year of Engineering.
- **Four Years Bachelor's Degree in Technology (B.Tech.) Honors with Research and Multidisciplinary Minor:** Awarded after completing Fourth Year of Engineering and acquiring additional 18 credits through a research project in respective major discipline during Fourth Year of Engineering.
- **Four Years Bachelor's Degree in Technology (B.Tech.) with Double Minors (Multidisciplinary & Specialization):** Awarded after completing Fourth Year of Engineering and acquiring additional 18 credits through additional courses in another Engg. / Tech. discipline during Second to Fourth Year of Engineering.

Through the implementation of Autonomy Syllabus Scheme III (as per NEP 2020 Guidelines), strategic planning, and joint efforts of all stakeholders, KJSIT is endeavouring to enhance the quality of engineering education and set a benchmark for all the autonomous institutes nationwide.

Dr. Vivek Sunnapwar
Principal and Chairman - Academic Council

Preface by Chairperson – Board of Studies (BoS) in Information Technology:

Information Technology is the backbone of modern innovation and digital transformation, driving societal advancement by revolutionizing processes, enhancing efficiency, and creating new opportunities. It enables breakthroughs in automation, data analysis, connectivity, and problem-solving, crucial for progress and development in the contemporary world. The **National Educational Policy 2020 (NEP 2020)** by Govt. of India propels us towards a holistic education system, integrating multidisciplinary research, community engagement, and skill-based learning to prepare a future-ready workforce.

To foster a generation of IT engineers capable of leveraging technology to tackle real-world challenges, advance the Sustainable Development Goals (SDGs) of the United Nations, and contribute effectively to the vision of Atmanirbhar Bharat, our curriculum for the B.Tech. Program in Information Technology has evolved from Scheme I to revised Schemes II, II B, and now to Scheme III—aligned with NEP 2020. The revision reflects reorganization and inclusion of state-of-the-art courses as well as Learning-by-Doing SAT courses, with objectives to: empower students in achieving better employability, start-ups and other avenues for higher education; holistic development as per NEP 2020; and attainment of all the program outcomes. The Scheme III introduced herewith, shall be effective for Second Year from Academic Year 2024-25, and progressively thereafter.

As per guidelines by the Department of Higher and Technical Education, Government of Maharashtra, the salient features of NEP 2020 aligned Scheme III curriculum for undergraduate engineering in Information Technology includes the following verticals:

- **Major (Core) Mandatory and Elective Courses**, which covers the cutting-edge technology courses in Information Technology, designed with consideration of current and futuristic trends in the industries. The elective courses at department-level are grouped into 04 major domains: a) Artificial Intelligence, b) Network & Security, c) Multimedia, and d) Optimization, which provides students with opportunities to have in-depth knowledge in the emerging areas concerning their own choice.
- **Open Elective Courses**, offered at institute-level provide students flexibility to explore diverse areas beyond their core curriculum, fostering interdisciplinary knowledge.
- **Vocational and Skill Enhancement Courses**, referred as Vocational Skill – SAT (VS-SAT) Courses and Skill Enhancement – SAT (SE-SAT) Courses respectively are designed to impart IT-specific vocational skills as well as enhance problem-solving skills and analytical thinking.
- **Ability Enhancement and Value Education Courses**, which cover Ability Enhancement – SAT (AE - SAT) Courses focusing on professional communication skills and learning Indian modern languages, and Value Education – SAT (VE - SAT) Courses to foster integrity, responsible decision-making, and professional conduct.
- **Community Engagement Projects and Internship**, with 5 Project-Based Learning (PBL) courses and a semester-long Internship enable students with exposure and proficiency in developing need-based or field projects to address the real-world issues of the society and/or industry.
- **Multidisciplinary Minor Courses**, which encourage students to explore diverse fields beyond their major, thus creating versatile individuals capable of contributing to multiple sectors and addressing global challenges.
- **Option for Bachelor's Degree with Honours**, promoting expertise and advanced knowledge, and enabling students to tailor their education to their interests and career aspirations.
- **Option for Bachelor's Degree – Honours with Research**, emphasizes research at UG level, thus encouraging research careers, innovation and intellectual growth.
- **Multiple Entry and Multiple Exit Options**, enables learners to earn certificates, diplomas, or degrees based on their completed credits, facilitates lifelong learning, and accommodates diverse learning needs, personal circumstances, and career goals.

The amalgamation of all these learning components in the curriculum, aligned with NEP 2020's vision of an equitable and dynamic education system, will nurture vast potential of the youths and contribute to the national development process in field of Information Technology.

The curriculum is the culmination of the efforts and meticulous work of all the members of the Board of Studies, subject-expert faculty members from other departments of the institute, external experts from academia, experienced professionals from IT companies, as well as the alumni working in IT companies across India and abroad.

We, the Board of Studies in Information Technology believe that the curriculum will meet the expectations of all the stakeholders and they shall take the advantage of the dynamic features of the curriculum—making the teaching-learning process an exalted experience for all.

Dr. Radhika Kotecha
Head – Department of Information Technology and Chairperson – BoS in Information Technology

Members of the Board of Studies (BoS) in Information Technology:

1. Dr. Radhika Kotecha (Chairperson),
   Professor and Head – Department of Information Technology
2. Dr. Hariram Chavan (Member – Internal Faculty),
   Professor – Department of Information Technology and Dean – Administration, KJSIT
3. Dr. Mansing Rathod (Member – Internal Faculty),
   Associate Professor – Department of Information Technology, KJSIT
4. Mr. Uday Rote (Member – Internal Faculty),
   Assistant Professor – Department of Information Technology and Dean – Student Welfare, KJSIT
5. Mrs. Seema Yadav (Member – Internal Faculty),
   Assistant Professor – Department of Information Technology
6. Dr. Vijaya Pinjarkar (Member – Internal Faculty),
   Assistant Professor – Department of Information Technology
7. Ms. Nasim Shah (Member – Internal Faculty),
   Assistant Professor – Department of Information Technology
8. Dr. Harsh Bhor (Member – Internal Faculty),
   Assistant Professor – Department of Information Technology
9. Dr. Reena Lokare (Member – Internal Faculty),
   Assistant Professor – Department of Information Technology
10. Mrs. Sarita Rathod (Member – Internal Faculty),
    Assistant Professor – Department of Information Technology
11. Dr. Lata Ragha (Member – External Faculty),
    Professor – Department of Information Technology and Dean Students Affairs & Alumni,
    Fr. C. Rodrigues Institute of Technology, Navi Mumbai
12. Dr. Vijay Raisinghani (Member – External Faculty),
    Professor, Professor (E-Business), WeSchool, Mumbai
13. Ms. Shilpa Karekeraa (Member – Industry Expert),
    Founder and CEO, Myraa Technologies (International)
14. Mr. Sunil Jain (Member – Industry Expert),
    Enterprise Data & Analytics Architect, Accenture, Mumbai

Nomenclature and Alignment of Verticals and Components

Verticals as per NEP 2020 Guidelines	Components Aligning with KJSIT Autonomy Syllabus Scheme I / II / II B	Nomenclature for KJSIT Autonomy Syllabus Scheme III Aligned with NEP 2020 Guidelines
Basic and Engineering Science Courses	Basic Science (BS) Course	Basic Science (BS) Courses
	Engineering Science (ES) Course	Engineering Science (ES) Courses
Major Courses	Professional Core (PC) Courses	Major / Professional Core (PC) Courses
	Professional Elective - Department-level (PE-DLC) Courses	Major / Professional Elective - Department-level (PE-DLC) Courses
Generic / Open Elective Courses	Open Elective - Institute-level (OE-ILC) Courses	Open Elective - Institute-level (OE-ILC) Courses
Multidisciplinary Minor Courses	-	Multidisciplinary Minor (MM) Courses
Vocational Skill Courses	Workshop I; Workshop II; SAT Courses – TBL	Vocational Skill - SAT (VS-SAT) Courses
Skill Enhancement Courses	SAT Courses – SBL (Program Specific)	Skill Enhancement - SAT (SE-SAT) Courses
Ability Enhancement Courses	Professional Communication Skills; SAT Course – SBL (Foreign and/or Indian Modern Languages)	Ability Enhancement - SAT (AE - SAT) Courses
Indian Knowledge System Courses	-	Indian Knowledge System - SAT (IKS - SAT) Courses
Value Education Courses	SAT Course – ABL (National, Global, Societal and Environmental Aspects); Business Communication & Ethics	Value Education - SAT (VE - SAT) Courses
Field Projects / Community Engagement Projects	PBL – Mini, Minor, Major	Community Engagement – Project-Based Learning (PBL)
Internship / Apprenticeship	Internship	Internship (INT)
Co-curricular Courses	Student Induction Program	Co-curricular - SAT (CC - SAT) Courses

Other Abbreviations:

- SAT – Skill/Activity/Technology-Based Learning (Exposure Courses)
- TH – Theory
- P – Practical
- TUT – Tutorial
- T1 – Test 1
- T2 – Test 2
- CA – Continuous Assessment Test (T = T1 + T2)
- ESE – End Semester Exam
- TW – Term Work
- O – Oral Exam
- P – Practical Exam
- P&O – Practical & Oral Exam

Programs Offered with Multiple Entry Multiple Exit Options

Level 4.5: UG Certificate in Technology

Major Discipline:	Information Technology
Years of Study:	01 Year
Semesters:	1 and 2
Credits:	42
Additional Requirements:	08 Credit Bridge Course Corresponding to Skill-Based Courses / Internship / Mini Projects in Major during Summer Vacation after 1st Year

Level 5: UG Diploma in Technology

Major Discipline:	Information Technology
Years of Study:	02 Years
Semesters:	1, 2, 3, 4
Credits:	85
Additional Requirements:	08 Credit Bridge Course Corresponding to Skill-Based Courses / Internship / Mini Projects in Major during Summer Vacation after 2nd Year

Level 5.5: Bachelor's Degree in Vocation (B. Voc.)

Major Discipline:	Information Technology
Years of Study:	03 Years
Semesters:	1, 2, 3, 4, 5, 6
Credits:	130
Additional Requirements:	08 Credit Bridge Course Corresponding to Skill-Based Courses / Internship / Mini Projects in Major during Summer Vacation after 3rd Year

Level 6: B.Tech. in Technology with Multidisciplinary Minor

Major Discipline:	Information Technology
Offered Multidisciplinary Minors:	• Innovation and Entrepreneurship • Biotechnology • IoT and Cloud Computing • Geographical Information System • VLSI
Years of Study:	04 Years
Semesters:	**Major –** 1, 2, 3, 4, 5, 6, 7, 8 **Multidisciplinary Minors** – 4, 5, 6
Credits:	174

Level 6: B.Tech. in Technology - Honors and Multidisciplinary Minor

Major Discipline:	Information Technology
Offered Honors and Multidisciplinary Minors:	**Honors:** • Internet of Things* • Artificial Intelligence & Machine Learning • Cyber Security • Virtual and Augmented Reality • Data Science • Blockchain **Multidisciplinary Minors:** • Innovation and Entrepreneurship • Biotechnology • IoT and Cloud Computing* • Geographical Information System • VLSI * Can be chosen for either Honors or Minors, not both
Years of Study:	04 Years
Semesters:	**Major –** 1, 2, 3, 4, 5, 6, 7, 8 **Multidisciplinary Minors** – 4, 5, 6 **Honors** – 5, 6, 7, 8
Credits:	192 (= Major with Multidisciplinary Minors: 174 + Honors: 18)

Level 6: B.Tech. in Technology - Honors with Research and Multidisciplinary Minor

Major Discipline:	Information Technology
Offered Multidisciplinary Minors:	• Innovation and Entrepreneurship • Biotechnology • IoT and Cloud Computing • Geographical Information System • VLSI
Years of Study:	04 Years
Semesters:	**Major –** 1, 2, 3, 4, 5, 6, 7, 8 **Multidisciplinary Minors** – 4, 5, 6 **Honors with Research** – 7, 8
Credits:	192 (= Major with Multidisciplinary Minors: 174 + Honors with Research: 18)

Level 6: B.Tech. in Technology with Double Minors (Multidisciplinary & Specialization)

Major Discipline:	Information Technology
Offered Multidisciplinary Minors and Specialization Minors:	**Multidisciplinary Minors:** • Innovation and Entrepreneurship • Biotechnology • IoT and Cloud Computing • Geographical Information System • VLSI **Specialization Minors:** 06 additional courses (of minimum 12 week each), in another Engg. / Tech. discipline / Emerging Areas through MOOC
Years of Study:	04 Years
Semesters:	**Major –** 1, 2, 3, 4, 5, 6, 7, 8 **Multidisciplinary Minors** – 4, 5, 6 **Specialization Minors** – 3, 4, 5, 6, 7, 8
Credits:	192 (= Major with Multidisciplinary Minors: 174 + Specialization Minors: 18)

Credit Distribution Structure for Four Year Multidisciplinary B.Tech. Degree Program in Information Technology with Multiple Entry Multiple Exit Options

Level	Semester	Faculty: Science and Technology					Faculty: Any	Vocational Skills (VS) & Skill Enhancement (SE) Courses		Ability Enhancement (AE), Indian Knowledge System (IKS), Value Education (VE) Courses			Field Projects / Community Engagement (CE) Projects, Internship (INT), and Co-curricular (CC) Courses			Credits	Cumulative Credits
		Basic Science (BS) Courses	Engineering Science (ES) Courses	Major / Professional Core (PC) Courses	Major / Professional Elective - Department-level (PE-DLC) Courses	Multi-disciplinary Minor (MM) Courses	Open Elective - Institute-level (OE-ILC) Courses	VS - SAT Courses	SE - SAT Courses	AE - SAT Courses	IKS - SAT Courses	VE - SAT Courses	CE - Project-Based Learning (PBL)	INT	CC - SAT Courses		
Level 4.5	I	9	8					1				1			2	21	42
	II	9	8					1		2	1					21	
Exit Option with UG Certificate in Technology with Additional 08 Credit Bridge Course Corresponding to Skill-Based Courses / Internship / Mini Projects in Major																	
Level 5.0	III	4		15					1				1			21	85
	IV	4		11		4			1	1			1			22	
Exit Option with UG Diploma in Technology with Additional 08 Credit Bridge Course Corresponding to Skill-Based Courses / Internship / Mini Projects in Major																	
Level 5.5	V			11	4	3			1			2	1			22	130
	VI			8	4	3	3	2					3			23	
Exit Option with Bachelor's Degree in Vocation (B. Voc.) with Additional 08 Credit Bridge Course Corresponding to Skill-Based Courses / Internship / Mini Projects in Major																	
Level 6.0	VII			8	7		3						6			24	174
	VIII			8										12		20	
Total		**26**	**16**	**61**	**15**	**10**	**6**	**4**	**3**	**3**	**1**	**3**	**12**	**12**	**2**	**174**	

SEMESTER III

TEACHING SCHEME

Course Code	Course Name	Teaching Scheme (Contact Hours)		Credits Assigned		Course Category
		TH – P – TUT	Total	TH – P – TUT	Total	
ITC301	Applications of Mathematics in Engineering – I	3 – 0 – 1	04	3 – 0 – 1	04	BS
ITC302	Data Structures and Analysis	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC303	Database Management System	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC304	Computer Organization and Architecture	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC305	Software Engineering	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITL302	Data Structures Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITL303	SQL Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITL304	Microprocessor & Microcontroller Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITPR31	Community Engagement PBL – Mini Project I	0 – 2 – 0	02[$]	0 – 1 – 0	01	PBL
ITXS37	Skill Enhancement - SAT VII: Skill-Based Learning (Java Programming)	0 – 2[*] – 0	02	0 – 1 – 0	01	SE-SAT
Total		**15 – 10 – 1**	**26**	**15 – 5 – 1**	**21**	

*SAT can be conducted as TH or P or both as required.

[$]Load of learner, not the faculty.

EXAMINATION SCHEME

Course Code	Course Name	CA Marks			ESE		TW / O / P Marks				Total Marks
		T1	T2	T = T1 + T2	Marks	Duration (in Hrs)	TW	O	P	P&O	
ITC301	Applications of Mathematics in Engineering – I	20	20	40	60	2.5	25	-	-	-	125
ITC302	Data Structures and Analysis	20	20	40	60	2.5	-	-	-	-	100
ITC303	Database Management System	20	20	40	60	2.5	-	-	-	-	100
ITC304	Computer Organization and Architecture	20	20	40	60	2.5	-	-	-	-	100
ITC305	Software Engineering	20	20	40	60	2.5	-	-	-	-	100
ITL302	Data Structures Lab	-	-	-	-	-	25	-	-	25	50
ITL303	SQL Lab	-	-	-	-	-	25	-	-	25	50
ITL304	Microprocessor & Microcontroller Lab	-	-	-	-	-	25	-	-	-	25
ITPR31	Community Engagement PBL – Mini Project I	-	-	-	-	-	25	-	-	25	50
ITXS37	Skill Enhancement - SAT VII: Skill-Based Learning (Java Programming)	-	-	-	-	-	25	-	-	-	25
Total		**100**	**100**	**200**	**300**	**-**	**150**	-	-	**75**	**725**

SEMESTER IV

TEACHING SCHEME

Course Code	Course Name	Teaching Scheme (Contact Hours)		Credits Assigned		Course Category
		TH – P – TUT	Total	TH – P – TUT	Total	
ITC401	Applications of Mathematics in Engineering – II	3 – 0 – 1	04	3 – 0 – 1	04	BS
ITC402	Computer Network and Network Design	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC403	Operating Systems	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC404	Automata Theory	3 – 0 – 0	03	3 – 0 – 0	03	PC
MMC4051	Multidisciplinary Minor Course	3 – 0 – 0	03	3 – 0 – 0	03	MM
ITL402	Network Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITL403	Unix Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
MML4051	Multidisciplinary Minor Lab	0 – 2 – 0	02	0 – 1 – 0	01	MM
ITPR42	Community Engagement PBL – Mini Project II	0 – 2 – 0	02[$]	0 – 1 – 0	01	PBL
ITXS48	Skill Enhancement – SAT VIII: Skill-Based Learning (Python Programming)	0 – 2[*] – 0	02	0 – 1 – 0	01	SE-SAT
ITXS49	Ability Enhancement – SAT IX: Skill-Based Learning (Foreign and/or Indian Modern Languages)	0 – 2[*] – 0	02	0 – 1 – 0	01	AE-SAT
Total		**15 – 12 – 1**	**28**	**15 – 6 – 1**	**22**	

[*]SAT can be conducted as TH or P or both as required.

[$]Load of learner, not the faculty.

EXAMINATION SCHEME

Course Code	Course Name	CA Marks			ESE		TW / O / P Marks				Total Marks
		T1	T2	T = T1 + T2	Marks	Duration (in Hrs)	TW	O	P	P&O	
ITC401	Applications of Mathematics in Engineering – II	20	20	40	60	2.5	25	-	-	-	125
ITC402	Computer Network and Network Design	20	20	40	60	2.5	-	-	-	-	100
ITC403	Operating Systems	20	20	40	60	2.5	-	-	-	-	100
ITC404	Automata Theory	20	20	40	60	2.5	-	-	-	-	100
MMC405	Multidisciplinary Minor Course	-	-	-	-	-	50	50	-	-	100
ITL402	Network Lab	-	-	-	-	-	25	-	-	25	50
ITL403	Unix Lab	-	-	-	-	-	25	-	25	-	50
MML405	Multidisciplinary Minor Lab	-	-	-	-	-	25	-	-	-	25
ITPR42	Community Engagement PBL – Mini Project II	-	-	-	-	-	25	-	-	25	50
ITXS48	Skill Enhancement – SAT VIII: Skill-Based Learning (Python Programming)	-	-	-	-	-	25	-	-	-	25
ITXS49	Ability Enhancement – SAT IX: Skill-Based Learning (Foreign and/or Indian Modern Languages)	-	-	-	-	-	25	-	-	-	25
Total		**80**	**80**	**160**	**240**	-	**225**	**50**	**25**	**50**	**750**

SEMESTER III

TEACHING SCHEME

Course Code	Course Name	Teaching Scheme (Contact Hours)		Credits Assigned		Course Category
		TH – P – TUT	Total	TH – P – TUT	Total	
ITC301	Applications of Mathematics in Engineering – I	3 – 0 – 1	04	3 – 0 – 1	04	BS
ITC302	Data Structures and Analysis	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC303	Database Management System	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC304	Computer Organization and Architecture	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC305	Software Engineering	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITL302	Data Structures Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITL303	SQL Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITL304	Microprocessor & Microcontroller Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITPR31	Community Engagement PBL – Mini Project I	0 – 2 – 0	02[$]	0 – 1 – 0	01	PBL
ITXS37	Skill Enhancement - SAT VII: Skill-Based Learning (Java Programming)	0 – 2[*] – 0	02	0 – 1 – 0	01	SE-SAT
Total		**15 – 10 – 1**	**26**	**15 – 5 – 1**	**21**	

*SAT can be conducted as TH or P or both as required.

[$]Load of learner, not the faculty.

EXAMINATION SCHEME

Course Code	Course Name	CA Marks			ESE		TW / O / P Marks				Total Marks
		T1	T2	T = T1 + T2	Marks	Duration (in Hrs)	TW	O	P	P&O	
ITC301	Applications of Mathematics in Engineering – I	20	20	40	60	2.5	25	-	-	-	125
ITC302	Data Structures and Analysis	20	20	40	60	2.5	-	-	-	-	100
ITC303	Database Management System	20	20	40	60	2.5	-	-	-	-	100
ITC304	Computer Organization and Architecture	20	20	40	60	2.5	-	-	-	-	100
ITC305	Software Engineering	20	20	40	60	2.5	-	-	-	-	100
ITL302	Data Structures Lab	-	-	-	-	-	25	-	-	25	50
ITL303	SQL Lab	-	-	-	-	-	25	-	-	25	50
ITL304	Microprocessor & Microcontroller Lab	-	-	-	-	-	25	-	-	-	25
ITPR31	Community Engagement PBL – Mini Project I	-	-	-	-	-	25	-	-	25	50
ITXS37	Skill Enhancement - SAT VII: Skill-Based Learning (Java Programming)	-	-	-	-	-	25	-	-	-	25
Total		**100**	**100**	**200**	**300**	**-**	**150**	-	-	**75**	**725**

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC301	Applications of Mathematics in Engineering – I	03	-	01	04

Prerequisites:	Engineering Mathematics.
Course Objectives (COBs):	1. To learn the Laplace Transform, Inverse Laplace Transform of various functions, its applications. 2. To understand the concept of Fourier Series, its complex form and enhance the problem-solving skills. 3. To understand the concept of Complex Variables, C-R equations with applications. 4. To understand the basic techniques of statistics like Correlation, Regression, and Curve Fitting for Data Analysis, Machine learning, and AI. 5. To understand some advanced topics of Probability, Random Variables with their Distributions and Expectations.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Solve the real integrals in engineering problems using the concept of Laplace Transform. 2. Analyze engineering problems through the application of inverse Laplace transform of various functions. 3. Expand the periodic function by using the Fourier series for real-life problems and complex engineering problems. 4. Solve the problems of obtaining orthogonal trajectories and analytic functions by means of complex variable theory and application of harmonic conjugate. 5. Apply the concept of Correlation and Regression to the engineering problems in Data Science, Machine Learning, and AI. 6. Analyze the spread of data and distribution of probabilities by the concepts of probability and expectation.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	01	01
1. Laplace Transform	Definition of Laplace Transform, Condition of Existence of Laplace Transform.	CO1	01	07
	Laplace Transform *(L)* of Standard Functions like e^{at}, $sin(at)$, $cos(at)$, $sinh(at)$, $cosh(at)$ and t^n, $n \geq 0$.	CO1	02	
	Properties of Laplace Transform: Linearity, First Shifting Property, Second Shifting Property, Change of Scale Property, Multiplication by t, Division by t, Laplace Transform of Derivatives and Integrals (Properties without proof).	CO1	02	
	Evaluation of Integrals by using Laplace Transformation.	CO1	02	
2. Inverse Laplace Transform	Definition of Inverse Laplace Transform, Linearity Property, Inverse Laplace Transform of Standard Functions, Inverse Laplace Transform using Derivatives.	CO2	02	06

	Partial Fractions Method to find Inverse Laplace Transform.	CO2	02	
	Inverse Laplace Transform using Convolution Theorem (without proof).	CO2	02	
3. Fourier Series	Dirichlet's Conditions, Definition of Fourier Series and Parseval's Identity (without proof).	CO3	01	07
	Fourier Series of Periodic Function with Period 2π & $2l$.	CO3	02	
	Fourier Series of Even and Odd Functions.	CO3	02	
	Fourier Transform-Fourier Sine Transform and Fourier Cosine Transform.	CO3	02	
4. Complex Variables	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).	CO4	01	07
	Cauchy-Riemann Equations in Cartesian Coordinates (without proof).	CO4	02	
	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.	CO4	02	
	Harmonic Function, Harmonic Conjugate and Orthogonal Trajectories.	CO4	02	
5. Statistical Techniques	Karl Pearson's Coefficient of Correlation *(r)*.	CO5	01	07
	Spearman's Rank Correlation Coefficient *(R)* (with repeated and non-repeated Ranks).	CO5	01	
	Lines of Regression.	CO5	02	
	Fitting of First and Second-Degree Curves.	CO5	02	
6. Probability	Definition and Basics of Probability, Conditional Probability.	CO6	01	07
	Total Probability Theorem and Bayes' Theorem.	CO6	01	
	Discrete and Continuous Random Variable with Probability Distribution and Probability Density Function.	CO6	02	
	Expectation, Variance, Moment Generating Function, Raw and Central Moments up to 4^{th} order.	CO6	02	

Text Books:	1. B. Grewal, Higher Engineering Mathematics, Khanna Publications. 2. E. Kreyszig, Advanced Engineering Mathematics, Wiley. 3. T. Veerarajan, Probability, Statistics and Random Processes, McGraw Hill.
Reference Books:	1. R. Jain and S. Iyengar, Advanced Engineering Mathematics, Narosa Publication. 2. J. Brown and R. Churchill, Complex Variables and Applications, McGraw Hill. 3. M. Spiegel, Theory and Problems of Fourier Analysis with applications to BVP, Schaum's Outline Series.
Useful Links:	1. http://epgp.inflibnet.ac.in/Home/ViewSubject?catid=25 2. https://nptel.ac.in/noc/courses/111/ 3. https://www.coursera.org/courses?query=mathematics 4. https://ndl.iitkgp.ac.in/
Term Work (TW):	• Term work shall consist of 06 batch wise tutorials. • Journal must include at least 02 assignments on content of theory of the course.

	• Term work evaluation shall be for Total 25 Marks based on Tutorials (15 Marks) and Assignments (10 Marks).
Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC302	Data Structures and Analysis	03	-	-	03

Prerequisites:	Computer Programming (C / C++).
Course Objectives (COBs):	1. To introduce the concepts of data structures and analysis procedure. 2. To conceptualize linear data structures and its implementation for various real-world applications. 3. To provide the understanding of non-linear data structures and its applications in developing solutions to real-world problems. 4. To impart knowledge of sorting and searching algorithms. 5. To develop an ability to design and analyze algorithms using various data structures. 6. To design and implement various data structure algorithms for solving real-world problems.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Explain the fundamental concepts of data structures, analyse a given problem to identify suitable data structures applicable for solving it, and describe the complexities of algorithms designed for the same. 2. Apply the concepts of stacks and queues to develop real-world problem solutions. 3. Apply the concepts of singly, circular, or doubly linked list as per the requirements for solving real-world problems. 4. Apply the concepts of trees to develop real-world problem solutions. 5. Apply the concepts of graphs to develop real-world problem solutions. 6. Apply appropriate sorting/searching techniques for real-world problem-solving.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Introduction to Data Structures and Analysis	Introduction to Data Structures, Need of Data Structures, Types of Data Structures: Linear and Non-linear Data Structures, Static and Dynamic Data Structures.	CO1	02	04
	Introduction to Analysis, Algorithms, Characteristics of Algorithms, Time and Space Complexities, Order of Growth Functions, Asymptotic Notations.	CO1	02	
2. Stacks and Queues	Introduction to Stack, Stack as ADT, Operations on Stack, Polish Notation: Infix, Prefix, and Postfix Expressions, their Evaluation and Conversions.	CO2	04	10
	Applications of Stack: Reversal of a String, Checking Validity of Expressions with Nested Parenthesis.	CO2	01	
	Introduction to Queue, Queue as ADT, Operations on Queue, Linear Representation of Queue, Circular Queue, De-queue.	CO2	03	
	Priority Queue, Applications of Queues: Scheduling.	CO2	01	
	Analysis of Stack and Queue Complexities and their Suitability for Solving Different Real-world Problems.	CO1	01	

3. Linked List	Introduction to Linked Lists, Singly Linked Lists, Circular Linked Lists, Insertion, Deletion, and Update Operations with Singly and Circular Linked Lists.	CO3	04	09
	Doubly Linked Lists, Insertion, Deletion, and Update Operations with Doubly Linked Lists.	CO3	03	
	Linked List Representation of Stack and Queue, Analysis of Linked Lists and its Suitability for Solving Different Real-world Problems.	CO3, CO1	01	
	Applications of Linked Lists.	CO3	01	
4. Trees	Introduction to Trees, Tree Terminologies.	CO4	01	06
	Binary Tree Representation, Operations on Binary Trees, Traversal of Binary Trees, Threaded Binary Trees, Analysis of Trees and its Suitability for Solving Different Real-world Problems.	CO4, CO1	03	
	Application-oriented Introduction: Binary Search Trees, B-Trees, B+ Trees, Decision Trees, Expression Trees, etc.	CO4	01	
	Application of Trees: Huffman Encoding.	CO4	01	
5. Graphs	Introduction to Graphs, Graph Terminologies, Graph Representation, Type of Graphs.	CO5	01	05
	Graph Traversal: Depth First Search (DFS), Breadth First Search (BFS), Analysis of Graphs and its Suitability for Solving Different Real-world Problems.	CO5, CO1	02	
	Minimum Spanning Tree: Prim's & Kruskal's Shortest Path Algorithm, Applications of Graphs: Traversal.	CO5	02	
6. Sorting and Searching	Introduction to Sorting, Sorting Techniques: Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Quick Sort, Comparison of Sorting Technique Performances.	CO6, CO1	03	05
	Searching: Sequential Search, Binary Search, Hashing: Hash Functions – Truncation, Mid-square Method, Folding Method, Division Method.	CO6	01	
	Collision Resolution: Open Addressing - Linear Probing, Quadratic Probing, Double Hashing, Separate Chaining, Bucket Hashing, Analysis of all Searching Techniques.	CO6	01	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01
Text Books:	1. J. Tremblay and P. Sorenson, Introduction to Data Structure and its Applications, McGraw Hill. 2. R. Thareja, Data Structures using C, Oxford. 3. S. Srivastava, D. Srivastava, Data Structures through C in Depth, BPB Publications.			
Reference Books:	1. Y. Langsam, M. Augenstein, and A. Tenenbaum, Data Structures using C and C++, Pearson. 2. E. Horowitz and S. Sahni, Fundamentals of Data Structures, Galgotia Publications. 3. R. Shukla, Data Structures using C and C++, Wiley.			
Useful Links:	1. https://learndsa.kjsieit.in/ 2. https://nptel.ac.in/courses/106/102/106102064/ 3. https://www.coursera.org/learn/data-structures 4. https://www.codechef.com/			

Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC303	Database Management System	03	-	-	03

Prerequisites:	Computer Programming (C / C++), Basic Knowledge of Computer File System.
Course Objectives (COBs):	1. To learn the basics and understand the need of a Database Management System. 2. To construct conceptual data model for real world applications. 3. To build a Relational Model from ER/EER. 4. To introduce the concept of SQL to store and retrieve data efficiently. 5. To demonstrate notions of Normalization for Database Design. 6. To understand the concepts of Transaction Processing - Concurrency Control & Recovery Procedures.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Describe the basics and need of a database management system. 2. Design conceptual models for real life applications. 3. Create a Relational model from ER/EER. 4. Apply queries using SQL commands for databases. 5. Design normalized database by applying normalization process. 6. Explain the concept of transaction, concurrency and recovery.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Database System Concepts and Architecture	Introduction, Characteristics of Databases, File System v/s Database System, Data Abstraction and Data Independence.	CO1	03	05
	DBMS System Architecture, Database Administrator (DBA), Role of DBA.	CO1	02	
2. Entity Relationship Model	Conceptual Modelling of a Database, Entity Relationship (ER) Model, Entity Type, Entity Sets, Attributes and Keys, Relationship Types, Relationship Sets.	CO2	03	06
	Weak Entity Types, Generalization, Specialization and Aggregation, Extended Entity-Relationship (EER) Model.	CO2	03	
3. Relational Model & Relational Algebra	Introduction to Relational Model, Relational Model Constraints and Relational Database Schemas, Concept of Keys: Primary Key, Secondary Key, Foreign Key, Mapping the ER and EER Model to Relational Model.	CO3	03	05
	Introduction to Relational Algebra, Relational Algebra Expressions for Unary Relational Operations, Set Theory Operations, Binary Relational Operation, Relational Algebra Queries.	CO3	02	
4. Structured Query Language	Overview of SQL, Data Definition Commands, Set Operations, Aggregate Function, Null Values, Data Manipulation Commands, Data Control Commands,	CO4	03	09

(SQL) & Indexing	Complex Retrieval Queries using Group by.			
	Recursive Queries, Nested Queries, All Types of Joins, Introduction to PL-SQL, Integrity Constraints in SQL. Database Programming with JDBC, Security and Authorization: Grant & Revoke in SQL. Functions and Procedures in SQL and Cursors.	CO4	04	
	Indexing: Basic Concepts, Ordered Indices, Index Definition in SQL.	CO4	02	
5. Relational Database Design	Design Guidelines for Relational Schema, Functional Dependencies, Database Tables and Normalization, The Need for Normalization, The Normalization Process, Improving the Design.	CO5	05	07
	Definition of Normal Forms- 1NF, 2NF, 3NF & The Boyce-Codd Normal Form (BCNF), 4NF.	CO5	02	
6. Transactions Management, Concurrency and Recovery	Transaction Concepts, State Diagram, ACID Properties, Transaction Control Commands, Concurrent Executions, Serializability – Conflict and View.	CO6	04	07
	Concurrency Control: Lock-based-protocols, Deadlock Handling, Timestamp-Based Protocols, Recovery System: Recovery Concepts, Log Based Recovery.	CO6	03	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01

Text Books:	1. H. Korth, A. Silberchatz, S. Sudarshan, Database System Concepts, McGraw Hill. 2. R. Elmasri and S. Navathe, Fundamentals of Database Systems, Pearson. 3. R. Ramkrishnan and J. Gehrke, Database Management Systems, McGraw Hill.
Reference Books:	1. P. Rob and C. Coronel, Database Systems Design, Implementation and Management, Thomson Learning. 2. P. Deshpande, SQL & PL/SQL for Oracle 11g Black Book, Dreamtech Press. 3. G. Gupta, Database Management Systems, McGraw Hill.
Useful Links:	1. https://onlinecourses.nptel.ac.in/noc19_cs46/preview 2. https://onlinecourses.nptel.ac.in/noc21_cs04/preview 3. https://www.coursera.org/learn/database-management
Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC304	Computer Organization and Architecture	03	-	-	03

Prerequisites:	Basics of Logic Design.
Course Objectives (COBs):	1. To conceptualize the basics of organizational and features of a digital computer. 2. To study microprocessor architecture and assembly language programming. 3. To study processor organization and parameters influencing performance of a processor. 4. To analyze various algorithms used for arithmetic operations. 5. To study the function of each element of memory hierarchy and various data transfer techniques used in digital computer. 6. To study microcontroller architecture and C language programming.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Describe basic organization of computer and the architecture of 8086 microprocessor and implement assembly language programming for 8086 microprocessors. 2. Describe control unit design methods and conceptualize instruction level parallelism. 3. Apply fundamentals of digital logic design to solve problem & perform various arithmetic operations using various algorithms. 4. Describe concept of memory organization and explain the function of each element of a memory hierarchy. 5. Explain different methods for computer I/O mechanism. 6. Describe the architecture of 8051 microcontroller and implement C language programming for 8051 microcontrollers.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Overview of Computer Architecture & Organization	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.	CO1	03	05
	Architecture of 8086 Family, Instruction Set, Addressing Modes.	CO1	02	
2. Processor Organization and Architecture	CPU Architecture, Instruction Formats, Basic Instruction Cycle with Interrupt Processing. Instruction Interpretation and Sequencing.	CO2	02	07
	Control Unit: Soft Wired (Microprogrammed) and Hardwired Control Unit.	CO2	03	
	Microinstruction Sequencing and Execution, Micro Operations, Concepts of Nano Programming, Introduction to Parallel Processing Concepts, Flynn's Classifications, Instruction Pipelining, Pipeline Hazards.	CO2	02	
3. Data Representati	Number Systems: Introduction to Number Systems, Binary Number Systems, Signed Binary Numbers,	CO3	02	08

on and Arithmetic Algorithms	Binary, Octal, Decimal and Hexadecimal Number and their Conversions, 1's and 2's Complement			
	Basics of Digital Circuits: NOT, AND, OR, NAND, NOR, EX-OR, EX-NOR Gates, Introduction to K-Mal	CO3	03	
	Booth's Algorithm, Division of Integers: Restoring and Non-Restoring Division, Signed Division.	CO3	03	
	Floating-Point Representation: IEEE 754 Floating Point (Single & Double Precision) Number Representation.	CO3	01	
4. Memory Organization	Introduction to Memory and Memory Parameters, Classifications of Primary and Secondary Memories, Types of RAM and ROM, Allocation Policies, Memory Hierarchy and Characteristics.	CO4	03	06
	Cache Memory: Concept, Architecture (L1, L2, L3), Mapping Techniques. Cache Coherency, Interleaved and Associative Memory.	CO4	03	
5. I/O Organization	Input/Output Systems, I/O Module-Need & Functions.	CO5	02	05
	Types of Data Transfer Techniques: Programmed I/O, Interrupt Driven I/O and DMA.	CO5	03	
6. Overview of 8051 Microcontroller	Introduction to Microcontroller, Difference between Microcontroller and Microprocessor.	CO6	04	08
	Architecture of 8051 Microcontroller, Pin Diagram of 8051, Instruction Set of 8051, C Language Programming, Interfacing of Ports.	CO6	04	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01
Text Books:	1. C. Hamacher, Z. Vranesic and S. Zaky, Computer Organization, McGraw Hill. 2. W. Stallings, Computer Organization and Architecture: Designing for Performance, Pearson. 3. J. Uffenbeck, 8086/8088 family: Design Programming and Interfacing, Pearson Education. 4. M. Mazidi, J. Mazidi and R. McKinlay, The 8051 Microcontroller & Embedded systems using Assembly and C, Pearson. 5. R. Jain, Modern Digital Electronic, McGraw-Hill Publication.			
Reference Books:	1. L. Das, Embedded systems an integrated approach, Pearson. 2. B. Govindarajulu, Computer Architecture and Organization: Design Principles and Applications. 3. J. Hayes, Computer Architecture and Organization, McGraw Hill.			
Useful Links:	1. https://nptel.ac.in/courses/106/105/106105163/ 2. https://www.udemy.com/course/computer-organization-and-architecture-j/ 3. https://www.udemy.com/course/computer-fundamental-computer-architecture/			
Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.			
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.			

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC305	Software Engineering	03	-	-	03

Prerequisites:	Fundamentals of Programming.
Course Objectives (COBs):	1. To explain the concepts of object-oriented paradigm. 2. To provide comprehensive knowledge of software engineering principles. 3. To describe and analyse requirements, and perform planning and scheduling. 4. To develop software solutions using Behavioural and Structured diagrams. 5. To utilize tools for creating UML diagrams. 6. To apply testing and ensure quality in software solutions.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Explain the concepts of the object-oriented paradigm and the Java programming language. 2. Explain software engineering principles and software development models. 3. Analyze requirements to prepare software plans, schedules, and track project progress. 4. Design UML diagrams based on software requirements. 5. Apply tools for creating UML diagrams. 6. Analyze the quality of software solutions through testing.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Object-Oriented Programming for Software Development	Introduction to Software Engineering, Nature of Software, Programming for Software Development, Principles of object-oriented programming: Classes, Objects, Abstraction, Encapsulation, Inheritance, Polymorphism, Features of Java Language.	CO1	02	07
	Constants, variables and data types, Operators and Expressions, Types of variables and methods.	CO1	02	
	Control Statements: If Statement, If-else, Nested if, switch Statement, break, continue. Iteration Statements: for loop, while loop, and do-while loop. Java methods, Overloading, Math class, Arrays in java.	CO1	03	
2. Software Engineering Foundations	Software Engineering Myths, Software Process, Generic Process Model, SDLC.	CO2	02	06
	Prescriptive Process Models: The Waterfall Model, Incremental Models, Evolutionary Process Models: RAD and Spiral Model.	CO2	04	
3. Requirement Analysis, Software Estimation and Scheduling	Software Requirements: Functional & Non-Functional	CO3	02	08
	Software Documentation: Analysis and Modelling, Software Requirement Specification (SRS).	CO3	02	
	Software Project Estimation: LOC, FP, and Cost Estimation Techniques.	CO3	02	
	Project Scheduling & Tracking, Gantt Chart, PERT/CPM	CO3	02	

4. UML Diagrams	Design Concepts, Data Flow Diagram, Use Case Diagrams, Activity Diagrams.	CO4	04	08
	State Charts, Sequence Diagrams, Class and Component Diagrams.	CO4	04	
5. Tools for UML Diagrams	Popular UML Tools	CO5	02	04
	Criteria for Choosing UML Tools, Tools for Collaborative UML Designing, Advanced Features in UML Tools	CO5	02	
6. Software Testing	Software Quality Testing: Strategic Approach, Strategies for Conventional Software.	CO6	03	06
	Types of Dynamic Testing: White Box and Black Box Testing, Alpha and Beta Testing	CO6	03	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01

Text Books:	1. R. Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill. 2. R. Mall, Fundamentals of Software Engineering, PHI.
Reference Books:	1. P. Jalote, An Integrated Approach to Software Engineering, Narosa Publication. 2. I. Sommerville, Software Engineering, Addison-Wesley.
Useful Links:	1. https://nptel.ac.in/courses/108/102/108102120/ 2. https://nptel.ac.in/courses/108/105/108105132/ 3. https://www.udemy.com/course/analog-communication/ 4. https://www.udemy.com/course/digital-communication-information-theory/
Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Lab Code	Lab Name	Credits			
		TH	P	TUT	Total
ITL302	Data Structures Lab	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	Turbo / Borland C Complier / Online C Compiler.
Prerequisites:	Computer Programming (C / C++).
Lab Objectives (LOBs):	1. To introduce the concepts of data structures and analysis procedure. 2. To conceptualize linear data structures and its implementation for various real-world applications. 3. To provide the understanding of non-linear data structures and its applications in developing solutions to real-world problems. 4. To impart knowledge of sorting and searching algorithms. 5. To develop an ability to design and analyze algorithms using various data structures. 6. To design and implement various data structure algorithms for solving real-world problems.
Lab Outcomes (LOs):	Upon completion of the course, the learners will be able to: 1. Apply the concepts of stacks for real-world applications. 2. Apply the concepts of queues for real-world applications. 3. Apply the concepts of singly, circular, and doubly linked list for real-world applications. 4. Implement tree and graph data structure for real-world applications. 5. Implement sorting and searching techniques for real-world applications. 6. Develop solutions to real-world problems and challenges in Data Structures in team as well as an individual.

Lab No.	Experiment Title	LOs Mapped	Hours
0	Lab Prerequisites	-	02
1	Implementation of Stack using Array for real-world application.	LO1	02
2	Implementation of Queue using Array for real-world application.	LO2	02
3	Implementations of Infix to Postfix Expression for real-world application.	LO1	02
4	Implementation of Double-ended Queue using Array for real-world application.	LO2	02
5	Implementation of Singly Linked List / Circular Singly Linked List and various operations for real-world.	LO3	02
6	Implementation of Doubly Linked List and various operation for real-world application.	LO3	04
7	Implementation of Binary Tree and its Traversal for real-world application.	LO4	02
8	Implementation of various operations on tree like – copying tree, mirroring a tree, counting the number of nodes in the tree, etc.	LO4	02
9	Implementation of any one Graph Traversal Technique for real-world application.	LO4	02
10	Implementation of any one Sorting Technique considering a real-world application.	LO5	02
11	Advancement through Data Structures: a. Creation of Git profile for source code management. b. Solving problems of Data Structures using HackerRank, etc. platforms.	LO6	04

Virtual Lab Links:	1. http://cse01-iiith.vlabs.ac.in/ 2. https://ds1-iiith.vlabs.ac.in/data-structures-1/ 3. https://ds2-iiith.vlabs.ac.in/data-structures-2/
Term Work (TW):	• Term work should consist of a minimum of 08 experiments, to be performed in C / C++ / Java / Python. • Journal must include at least 02 assignments on content of theory course "Data Structures and Analysis" and "Data Structures Lab". • Term work evaluation shall be for Total 25 Marks (Experiments: 20 Marks, Assignments: 05 Marks). • The final certification and acceptance of term work will be based on attendance in Theory and Lab sessions, satisfactory performance of laboratory work, and minimum passing marks in term work evaluation.
Practical & Oral (P&O):	P&O examination will be based on the experiment list for Total 25 Marks (Practical: 15 Marks and Oral: 10 Marks).

Lab Code	Lab Name	Credits			
		TH	P	TUT	Total
ITL303	SQL Lab	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	MySQL / Online SQL Editor, JDK.
Prerequisites:	Computer Programming (C / C++).
Lab Objectives (LOBs):	1. To identify and define problem statements for real life applications. 2. To construct conceptual data model for real life applications. 3. To build Relational Model from ER/EER and use relational algebra. 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 (LOs):	Upon completion of the course, the learners will be able to: 1. Construct conceptual model for real-world applications. 2. Create and populate a RDBMS using SQL. 3. Implement efficient information retrieval using SQL. 4. Implement view, triggers and procedures to demonstrate specific event handling. 5. Implement database connectivity using JDBC. 6. Demonstrate the concept of concurrent transactions.

Lab No.	Experiment Title	LOs Mapped	Hours
0	Lab Prerequisites.	-	02
1	Identify real world problems and develop the problem statement. Design an Entity-Relationship (ER) / Extended Entity-Relationship (EER) Model.	LO1	02
2	Mapping ER / EER to Relational schema model.	LO1	02
3	Create a database using DDL and apply integrity constraints.	LO2, LO3	02
4	Perform data manipulations operations on populated databases.	LO3	02
5	Perform Authorization using Grant and Revoke.	LO2, LO3	02
6	Implement Basic and complex SQL queries.	LO3, LO4	02
7	Implementation of Views and Triggers.	LO4	02
8	Demonstrate database connectivity by preparing a simple form in any scripting language.	LO5	04
9	Execute TCL commands.	LO4	02
10	Implement Functions and Procedures in SQL.	LO3, LO4	02
11	Implementation of Cursor.	LO3, LO4	02
12	Implementation and demonstration of Transaction and Concurrency Control techniques using Locks.	LO6	02

Virtual Lab Links:	1. http://vlabs.iitb.ac.in/bootcamp/labs/dbms/exp8/index.php 2. http://vlabs.iitb.ac.in/vlabs-dev/labs/dblab/index.php 3. https://dev.mysql.com/doc/refman/8.0/en/sql-data-definition-statements.html
Term Work (TW):	• Term work should consist of a minimum of 08 experiments. • Journal must include at least 02 assignments on content of theory course "Database Management Systems" and "SQL Lab". • Term work evaluation shall be for Total 25 Marks (Experiments: 20 Marks, Assignments: 05 Marks). • The final certification and acceptance of term work will be based on attendance in Theory and Lab sessions, satisfactory performance of laboratory work, and minimum passing marks in term work evaluation.
Practical & Oral (P&O):	P&O examination will be based on the experiment list for Total 25 Marks (Practical: 15 Marks and Oral: 10 Marks).

Lab Code	Lab Name	Credits			
		TH	P	TUT	Total
ITL304	Microprocessor and Microcontroller Lab	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	Tasm, Keil / Crossware.
Prerequisites:	Computer Programming (C / C++).
Lab Objectives (LOBs):	1. To get hands-on experience with Assembly Language Programming. 2. To study interfacing of peripheral devices with 8086 microprocessors. 3. To realize techniques for faster execution of instructions and improve speed of operation and performance of microprocessors. 4. To write and debug programs in TASM / hardware kits / Crossware / Keil. 5. To get hands on experience with C Language Programming with controller. 6. To study interfacing of peripheral devices with 8051 microcontrollers.
Lab Outcomes (LOs):	Upon completion of the course, the learners will be able to: 1. Execute the selected instructions to understand addressing modes of 8086. 2. Execute assembly language programs on microprocessor using arithmetic and logical instructions of 8086 microprocessors. 3. Execute assembly language programs using loop instructions of 8086 microprocessors. 4. Execute the selected instructions to understand addressing modes of 8051. 5. Implement C language programs using instruction set of 8051. 6. Implement C language programs for interfacing different devices with 8051.

Lab No.	Experiment Title	LOs Mapped	Hours
0	Lab Prerequisites.	-	02
1	Simulation of selected instructions to understand the addressing modes and instruction set of 8086 microprocessors.	LO1	02
2	Implementation of Arithmetic and Logical operations using Assembly Language Programming. a. Program to perform arithmetic operations on 16-bit data. b. Program to evaluate given logical expression. c. Convert two-digit Packed BCD to Unpacked BCD.	LO2	02
3	Implementations of loop operations using Assembly Language Programming. a. Program to move set of numbers from one memory block to another. b. Program to count number of 1's and 0's in a given 8-bit number. c. Program to find even and odd numbers from a given list. d. Program to search for a given number.	LO3	02
4	Implementation of String Operations using Assembly Language Programming. a. Check whether a given string is a Palindrome or not. b. Compute the factorial of a positive integer 'n' using procedure. c. Generate the first 'n' Fibonacci numbers.	LO3	02
5	Simulation of selected instructions to understand the addressing modes and instruction set of 8051 Microcontroller.	LO4	02

6	Implementation of Arithmetic and Logical operations using C Language Programming. a. Program to perform arithmetic operations on 16-bit data. b. Program to evaluate given logical expression. c. Convert two-digit Packed BCD to Unpacked BCD.	LO5	02
7	Implementations of loop operations using C Language Programming. a. Program to move set of numbers from one memory block to another. b. Program to count number of 1's and 0's in a given 8-bit number. c. Program to find even and odd numbers from a given list. d. Program to search for a given number.	LO5	02
8	Interfacing of 8051 Microcontroller. a. Program to toggle bits of port P0, P1, P2, P3. b. Program to interface Stepper Motor. c. Program to perform serial communication.	LO6	06
9	Implementation of interfacing of LCD with the 8051 Microcontroller using C language programming.	LO6	02
10	Interfacing with 8051 Microcontroller. a. Interfacing Seven Segment Display. b. Interfacing Keyboard Matrix. c. Interfacing DAC.	LO6	04

Virtual Lab Links:	http://vlabs.iitkgp.ac.in/coa/
Term Work (TW):	• Term work should consist of a minimum of 08 experiments. • Journal must include at least 02 assignments on content of theory course "Computer Organization and Architecture" and "Microprocessor and Microcontroller Lab". • Term work evaluation shall be for Total 25 Marks (Experiments: 20 Marks, Assignments: 05 Marks). • The final certification and acceptance of term work will be based on attendance in Theory and Lab sessions, satisfactory performance of laboratory work, and minimum passing marks in term work evaluation.

PBL Course Code	PBL Course Name	Credits			
		TH	P	TUT	Total
ITPR31	Community Engagement PBL – Mini Project I	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	JDK, MySQL.
Prerequisites:	Basics of Computer Programming.
PBL Objectives (PROBs):	1. To create awareness among the students of the characteristics of several domain areas where IT can be effectively used. 2. To engage in community service, practice the process of identifying the needs and converting it into a problem statement. 3. To apply engineering knowledge and modern tools/technologies for deriving solutions to the real-world problems. 4. To inculcate the process of self-learning and research. 5. To be acquainted with solving the problem in a group. 6. To improve communication, management and report-writing skills of the students.
PBL Outcomes (PROs):	Upon completion of the course, the learners will be able to: 1. Identify societal / research needs through community engagement, formulate problem statements, review research literature, and analyze complex engineering problems. 2. Design suitable solutions for the problems including scope, objectives, timeline, system flow, user interface, algorithms, etc. 3. Gather, analyze, and interpret data — and apply knowledge of engineering fundamentals, modern tools / technologies for development of solutions. 4. Analyze sustainability and scalability of the developed solution and its impact in terms of environmental, societal, safety, legal, cultural, health, etc. aspects. 5. Apply ethical principles, excel in written and oral communication, and engage in independent and life-long learning. 6. Interact efficiently and effectively as an individual with the team members or leader for timely and professional management of projects.
Guidelines for Project-Based Learning (PBL):	1. Students have to form a team of minimum 02 and maximum 04 members, based on their area of interest and size of project. Interdisciplinary (inter-branch) teams are encouraged. 2. Students should carry out a field survey for community engagement, and identify needs, which shall be converted into problem statement for Mini Project in consultation with Faculty Guide, Internal committee of faculties, and the Head of Department. 3. Students should develop a Desktop / Web / Mobile Application with a proper user interface using any suitable technology like HTML5, CSS, etc. for front end and Java / Go at backend. 4. Projects should compulsorily be based on societal contribution (healthcare, agriculture, etc.) and reflecting role of engineer in the society. Students should try to take up need-based live projects so as to get exposure to communication with beneficiaries and skills for understanding client requirements. 5. Based on the idea presentation as well as discussion on feasibility, novelty, and contribution of the idea, a project definition will be finalized. 6. Students shall submit their implementation plan in the form of Gantt / PERT / CPM chart, which will cover weekly activity of the Mini project. 7. A log book is to be prepared by each group, wherein the group can record weekly work progress and the Faculty Guide can verify and record notes / comments.

	8. Faculty Guide may give inputs to students during Mini Project activity; however, focus shall be on self-learning. 9. Students in a group shall understand the problem effectively, propose multiple solutions, and select the best possible solution in consultation with their guide. 10. Students shall convert the best solution into a working model using various components of their domain areas and demonstrate. 11. The solution is to be validated with proper justification and report is to be compiled in standard format of the Department. 12. With the focus on self-learning, innovation, addressing societal problems and entrepreneurship quality development within the students through the Mini Projects, if the problem statement is huge and significant, a same problem statement can be worked upon for 02 semesters, i.e., same Mini Project in Semesters III and IV. Projects with a very large scope can also be taken forward to higher semesters, in consultation with the Head of the Department. 13. Students are encouraged to participate in Technical Paper Presentation competitions. 14. The students' group shall complete a project in all aspects including: Identification of need / problem, proposed final solution, Procurement of components / systems / data, Building prototype and testing. 15. Three reviews will be conducted for continuous assessment: one shall be for finalization of the problem and proposed solution, second shall be for evaluation of work progress, and third shall be for evaluation of implementation and testing of solutions. 16. Mini Project shall be assessed based on following parameters: • Attainment of Course Outcomes. • Technical efficiency and quality of developed solution. • Innovativeness in solutions. • Impact on environment. • Cost effectiveness. • Sustainability analysis. • Societal impact. • Effective use of standard engineering norms. • Contribution of an individual as member or leader. • Clarity in written and oral communication. 17. Students are encouraged to publish a paper based on the work in Conferences / Student competitions.
Useful Learning Links:	1. https://onlinecourses.nptel.ac.in/noc21_cs56/preview 2. https://www.coursera.org/specializations/core-java 3. https://www.udemy.com/course/java-se-programming/
Term Work (TW):	• Term Work shall be granted based individual's contribution in group activity, their understanding and response to questions. • Term Work evaluation shall be for Total 25 Marks — based on the following evaluation: ○ Presentation in Review 1 ○ Presentation in Review 2 ○ Presentation in Review 3 ○ Project Report and Log Book
Practical & Oral (P&O):	P&O examination will be of Total 25 Marks and shall be based on the Project Demonstration, Presentation, and Report.

Exposure Course Code	Exposure Course Name	Credits			
		TH	P	TUT	Total
ITXS37	Skill Enhancement - SAT VII: Skill-Based Learning (Java Programming)	-	02	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	JDK, NetBeans, Eclipse
Skill Prerequisites:	Basics of Computer Programming
Skill Objectives (SOBs):	1. To understand the concepts of object-oriented paradigm in the Java programming language. 2. To understand the importance of Classes & objects along with constructors, Arrays, Strings and vectors 3. To learn the principles of inheritance, interface, and packages and demonstrate the concept of reusability for faster development. 4. To recognize usage of Exception Handling, Multithreading, Input Output streams in various applications 5. To learn designing, implementing, testing, and debugging graphical user interfaces in Java using Swings and AWT components that can react to different user events. 6. To develop graphical user interfaces using JavaFX controls.
Skill Outcomes (SOs):	Upon completion of the course, the learners will be able to: 1. Apply the fundamental concepts of Java Programing. 2. Apply the concepts of classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem. 3. Apply the concepts of Inheritance, Interface and Packages. 4. Construct robust and faster programmed solutions to problems using concept of Multithreading, exceptions and file handling 5. Develop Graphical User Interface using Abstract Window Toolkit and Swings along with response to the events and database connectivity. 6. Develop Graphical User Interface by exploring JavaFX framework based on MVC architecture.

Module No. and Name	Subtopics	SOs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Java Fundamentals	Overview of Procedure and Object-Oriented Programming	SO1	01	03
	Fundamentals and Practice of Java Programming	SO1	02	
2. Classes, Objects, Arrays, and Strings	Classes & Objects: Reference Variables, Passing Parameters to Methods and Returning Parameters from the Methods, Static Members, Non-Static Members Nested and Inner Classes. Static	SO1, SO2	01	03

	Initialization Block (SIB), Instance Initialization Block (IIB).			
	Constructors: Parameterized Constructors, Chaining of Constructor, finalize () Method, Method Overloading, Constructors Overloading. Recursion, Command-Line Arguments. Wrapper Classes, InputBufferReader, OutputBufferReader, StringBuffer Classes, String Functions.	SO1, SO2	01	
	Arrays & Vectors: One and Two-Dimensional Arrays, Irregular Arrays, Dynamic Arrays, Array List and Array of Object.	SO2	01	
3. Inheritance, Packages and Interfaces	Inheritance: Inheritance Basics, Types of Inheritance in Java, Member Access, Using Super- to call Superclass Constructor, to access Member of Super Class (Variables and Methods), Creating Multilevel Hierarchy, Constructors in Inheritance, Method Overriding, Abstract Classes and Methods, using Final, Dynamic Method Dispatch.	SO1, SO3	02	04
	Packages: Defining Packages, Creating Packages, Importing and Accessing Packages.	SO1, SO3	01	
	Interfaces: Defining, Implementing and Extending Interfaces, Variables in Interfaces, Default Method in Interface, Static Method in Interface, Abstract Classes versus Interfaces.	SO1, SO3	01	
4. Exception Handling, Multithreading, Input Output Streams	Exception Handling: Exception-Handling Fundamentals, Exception Types, Exception Class Hierarchy, Using Try and Catch, Multiple Catch Clauses, Nested Try Statements, Throw, Throws, Finally, Java's Built-In Exceptions, Creating Your Own Exception Subclasses.	SO1, SO3, SO4	02	06
	Multithreaded Programming: The Java Thread Model and Thread Life Cycle, Thread Priorities, Creating a Thread, Implementing Runnable, Extending Thread, Creating Multiple Threads.	SO1, SO3, SO4	02	
	Synchronization: Using Synchronized Methods, The Synchronized Statement	SO1, SO3, SO4	01	
	I/O Streams: Streams, Byte Streams and Character, The Predefined Streams, Reading Console Input, Reading Characters, Reading Strings, Writing Console Output, Reading and Writing Files.	SO1, SO3, SO4	01	
5. GUI Programming - I (AWT, Event Handling, Swing)	Designing Graphical User Interfaces in Java: Components and Containers, Basics of Components, Using Containers, Layout Managers, AWT Components, Adding a Menu to Window, Extending GUI Features.	SO1, SO4, SO5	02	06

	Event-Driven Programming in Java: Event-Handling Process, Event-Handling Mechanism, Delegation Model of Event Handling, Event Classes, Event Sources, Event Listeners, Adapter Classes as Helper Classes in Event Handling.	SO1, SO4, SO5	02	
	Introducing Swing: AWT vs Swings, Components and Containers, Swing Packages, A Simple Swing Application, Painting in Swing, Designing Swing GUI Application using Buttons, JLabels, Checkboxes, Radio Buttons, JScrollPane, JList, JComboBox, Trees, TablesScroll Pane Menus and Toolbar, Database Connectivity.	SO1, SO4, SO5	02	
6. GUI Programming - II (JavaFX)	JavaFX Basic Concepts, JavaFX Application Skeleton, Compiling and Running JavaFX Program.	SO1, SO5, SO6	02	03
	Simple JavaFX Control: Label, Using Buttons and Events, Drawing directly on Canvas.	SO1, SO5, SO6	01	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01
Text Books:	1. H. Schildt, Java - The Complete Reference, Tenth Edition, McGraw Hill. 2. E. Balguruswamy, Programming with Java A primer, McGraw Hill. 3. A. Seth, B. Juneja, Java One Step Ahead, Oxford University Press.			
Reference Books:	1. D. Editorial Services, Java 8 Programming Black Book, Dreamtech Press. 2. Y. Kanetkar, Let Us Java, BPB Publications.			
Useful Learning Links:	1. https://onlinecourses.nptel.ac.in/noc21_cs03/preview 2. https://onlinecourses.swayam2.ac.in/aic20_sp13/preview 3. https://www.coursera.org/projects/introduction-to-java-programming-java-fundamental-concepts 4. https://www.udemy.com/course/core-java-from-scratch/ 5. https://java-iitd.vlabs.ac.in/			
Guidelines for Skill-Based Learning (SBL):	• Programming labs shall be conducted as 02 Hours of blended theory and hands-on session. • The classes can be conducted as a flipped classroom, where students have to attend class after reviewing the lessons provided to them beforehand. • Discussion on the topics and implementation of programs involving the concepts mentioned will be performed during the assigned lab hours.			
Term Work (TW):	• Term Work evaluation shall be for Total 25 Marks based on Practical Performance. • The final certification and acceptance of term work will be based on satisfactory performance of laboratory work, and minimum passing marks in term work evaluation.			

SEMESTER IV

TEACHING SCHEME

Course Code	Course Name	Teaching Scheme (Contact Hours)		Credits Assigned		Course Category
		TH – P – TUT	Total	TH – P – TUT	Total	
ITC401	Applications of Mathematics in Engineering – II	3 – 0 – 1	04	3 – 0 – 1	04	BS
ITC402	Computer Network and Network Design	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC403	Operating Systems	3 – 0 – 0	03	3 – 0 – 0	03	PC
ITC404	Automata Theory	3 – 0 – 0	03	3 – 0 – 0	03	PC
MMC4051	Multidisciplinary Minor Course	3 – 0 – 0	03	3 – 0 – 0	03	MM
ITL402	Network Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
ITL403	Unix Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
MML4051	Multidisciplinary Minor Lab	0 – 2 – 0	02	0 – 1 – 0	01	MM
ITPR42	Community Engagement PBL – Mini Project II	0 – 2 – 0	02[$]	0 – 1 – 0	01	PBL
ITXS48	Skill Enhancement – SAT VIII: Skill-Based Learning (Python Programming)	0 – 2[*] – 0	02	0 – 1 – 0	01	SE-SAT
ITXS49	Ability Enhancement – SAT IX: Skill-Based Learning (Foreign and/or Indian Modern Languages)	0 – 2[*] – 0	02	0 – 1 – 0	01	AE-SAT
Total		**15 – 12 – 1**	**28**	**15 – 6 – 1**	**22**	

[*]SAT can be conducted as TH or P or both as required.

[$]Load of learner, not the faculty.

EXAMINATION SCHEME

Course Code	Course Name	CA Marks			ESE		TW / O / P Marks				Total Marks
		T1	T2	T = T1 + T2	Marks	Duration (in Hrs)	TW	O	P	P&O	
ITC401	Applications of Mathematics in Engineering – II	20	20	40	60	2.5	25	-	-	-	125
ITC402	Computer Network and Network Design	20	20	40	60	2.5	-	-	-	-	100
ITC403	Operating Systems	20	20	40	60	2.5	-	-	-	-	100
ITC404	Automata Theory	20	20	40	60	2.5	-	-	-	-	100
MMC405	Multidisciplinary Minor Course	-	-	-	-	-	50	50	-	-	100
ITL402	Network Lab	-	-	-	-	-	25	-	-	25	50
ITL403	Unix Lab	-	-	-	-	-	25	-	25	-	50
MML405	Multidisciplinary Minor Lab	-	-	-	-	-	25	-	-	-	25
ITPR42	Community Engagement PBL – Mini Project II	-	-	-	-	-	25	-	-	25	50
ITXS48	Skill Enhancement – SAT VIII: Skill-Based Learning (Python Programming)	-	-	-	-	-	25	-	-	-	25
ITXS49	Ability Enhancement – SAT IX: Skill-Based Learning (Foreign and/or Indian Modern Languages)	-	-	-	-	-	25	-	-	-	25
Total		**80**	**80**	**160**	**240**	-	**225**	**50**	**25**	**50**	**750**

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC401	Applications of Mathematics in Engineering – II	03	-	01	04

Prerequisites:	Engineering Mathematics, Basics of Applications of Mathematics in Engineering
Course Objectives (COBs):	1. To analyze characteristics of matrices like Eigenvalues and Eigen vectors. 2. To analyze characteristics of matrices like Nullity and factorization of matrices. 3. To study statistics for data science. 4. To introduce concepts of probability distributions 5. To introduce concepts of sampling theory 6. To use the theory of Linear and Non-linear programming in engineering problems.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Determine eigenvalues, eigenvectors of matrices and study diagonalization. 2. Find nullity of the matrix as well as the factorization of the matrix. 3. Find the estimate of location, variability, covariance and correlation. 4. Evaluate probability distribution. 5. Use sampling theory in decision making problems. 6. Solve optimization problems using techniques of Linear and Non-Linear Programming.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Linear Algebra: Theory of Matrices I	Characteristic Equation, Eigenvalues and Eigenvectors, and Properties (without proof).	CO1	02	06
	Cayley-Hamilton Theorem (without proof-state and verify), Verification and Reduction of Higher Degree Polynomials.	CO1	02	
	Similarity of Matrices, Diagonalizable and Non-Diagonalizable Matrices.	CO1	02	
2. Linear Algebra: Theory of Matrices II	(Recall: Trace, determinant of matrices, Rank of the matrix), Nullity of the matrices (upto 4×4 matrices)	CO2	02	07
	Matrix factorization : LU factorization- Cholesky factorization	CO2	02	
	Singular Value Decomposition	CO2	03	
3. Statistics for Data Analysis	Estimates of locations (Mean, Median, Mode, Quartiles (Q1, Q2, Q3))	CO3	02	06
	Estimates of variability (Range, Inter quartile range, standard deviation, variance)	CO3	02	
	Covariance and Correlations (Kendall rank correlation).	CO3	02	
4. Probability Distribution	Discrete Probability Distribution: Binomial distribution, Poisson distribution	CO4	02	07
	Continuous Probability Distribution: Normal Distribution, Exponential Distribution, Weibull Distribution	CO4	05	

5. Sample Testing	Sampling Distribution, Test of Hypothesis, Level of Significance, Critical Region, One-tailed, and Two-tailed Test, Degree of Freedom.	CO5	02	07
	Students' t-distribution (Small Sample), Test Significance of Mean and Difference between the Means of Two Samples, Chi-Square Test: Test of Goodness of Fit and Independence of Attributes, Contingency Table.	CO5	03	
	ANOVA test	CO5	02	
6. Nonlinear Programming Problems	NLPP with One Equality Constraint (Two or Three Variables) using the Method of Lagrange's Multipliers.	CO6	02	07
	NLPP with Two Equality Constraints.	CO6	02	
	NLPP with Inequality Constraint: Kuhn-Tucker Conditions.	CO6	03	

Text Books:	1. E. Kreyszig, Advanced Engineering Mathematics, Wiley. 2. R. Jain and S. Iyengar, Advanced Engineering Mathematics, Narosa Publication. 3. J. Brown and R. Churchill, Complex Variables and Applications, McGraw Hill.
Reference Books:	1. T. Veerarajan, Probability, Statistics and Random Processes, McGraw Hill. 2. H. Taha, Operations Research: An Introduction, Pearson. 3. S. Rao, Engineering Optimization: Theory and Practice, Wiley. 4. D. Hira and P. Gupta, Operations Research, S. Chand and Sons.
Useful Links:	1. https://nptel.ac.in/courses/111/108/111108066/ 2. https://nptel.ac.in/courses/111/103/111103070/ 3. https://nptel.ac.in/courses/111/104/111104071/ 4. https://nptel.ac.in/courses/111/105/111105041/ 5. https://www.coursera.org/learn/complex-analysis 6. NPTEL :: Biotechnology - NOC:Data Analysis for Biologists 7. https://nptel.ac.in/courses/111101165 8. https://nptel.ac.in/courses/104106121
Term Work (TW):	• Term work shall consist of 06 batch wise tutorials. • Journal must include at least 02 assignments on content of theory of the course. • Term work evaluation shall be for Total 25 Marks based on Tutorials (15 Marks) and Assignments (10 Marks).
Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC402	Computer Network and Network Design	03	-	-	03

Prerequisites:	Fundamentals of Communication.
Course Objectives (COBs):	1. To explain the division of network functionalities into layers. 2. To describe the types of transmission media along with data link layer concepts, design issues and protocols. 3. To analyze the strength and weaknesses of routing protocols and gain knowledge about IP addressing. 4. To evaluate the data transportation, issues and related protocols for end-to-end delivery of data. 5. To examine the data presentation techniques used in presentation layer & client/server model in application layer protocols. 6. To design a network for an organization using networking concepts.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Describe the functionalities of each layer of the models and compare the models. 2. Categorize the types of transmission media and explain data link layer concepts, design issues and protocols. 3. Analyze the routing protocols and assign IP address to networks. 4. Explain the data transportation and session management issues and related protocols used for end-to-end delivery of data. 5. Explain the data presentation techniques and illustrate the client/server model in application layer protocols. 6. Apply networking concepts of IP address, routing, and application services to design a network for an organization.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Introduction to Computer Networks	IT Infrastructure, Use of Computer Networks, Network Devices, Network Software, Protocol Layering.	CO1	02	04
	Reference Models: OSI, TCP/IP, Comparison of OSI & TCP/IP.	CO1	02	
2. Physical Layer & Data Link Layer	Physical Layer: Guided Media, Unguided Media, Wireless Transmission: Electromagnetic Spectrum, Switching: Circuit-Switched Networks, Packet Switching, Structure of a Switch.	CO2	04	10
	DLL Design Issues (Services, Framing, Error Control, Flow Control), Error Detection and Correction (Hamming Code, Parity, CRC, Checksum), Elementary Data Link protocols: Stop and Wait, Sliding Window (Go Back N, Selective Repeat), Piggybacking, HDLC.	CO2	04	
	Medium Access Protocols: Random Access, Controlled Access, Channelization, Ethernet Protocol: Standard	CO2	02	

	Ethernet, Fast Ethernet (100 Mbps), Gigabit Ethernet, 10-Gigabit Ethernet.			
3. Network Layer	Network Layer Services, Packet Switching, Network Layer Performance, IPv4 Addressing (Classful and Classless), Subnetting, Supernetting, IPv4 Protocol, DHCP, Network Address Translation (NAT).	CO3	03	08
	Routing Algorithms: Distance Vector Routing, Link State Routing, Path Vector Routing.	CO3	02	
	Protocols – RIP, OSPF, BGP.	CO3	02	
	Next Generation IP: IPv6 Addressing, IPv6 Protocol, Transition from IPV4 to IPV6.	CO3	01	
4. Transport Layer & Session Layer	Transport Layer: Transport Layer Services, Connectionless & Connection-Oriented Protocols.	CO4	01	07
	Transport Layer Protocols: User Datagram Protocol: UDP Services, UDP Applications, Transmission Control Protocol: TCP Services, TCP Features, Segment, A TCP Connection, Windows in TCP, Flow Control, Error Control, TCP Congestion Control, TCP Timers.	CO4	04	
	Session Layer: Session Layer Design Issues, Session Layer Protocol - Remote Procedure Call (RPC).	CO4	02	
5. Presentation Layer & Application Layer	Presentation Layer: Compression: Comparison between Lossy Compression and Lossless Compression, Huffman Coding, Speech Compression, LZW, RLE, Image Compression – GIF, JPEG.	CO5	03	05
	Application Layer: Standard Client-Server Protocols: World Wide Web, HTTP, FTP, Electronic Mail, Domain Name System (DNS), SNMP.	CO5	02	
6. Network Design Concepts	Introduction to VLAN, VPN.	CO6	02	05
	Case Study to Design a Network for an Organization Meeting the following Guidelines: Networking Devices, IP Addressing: Subnetting, Supernetting, Routing Protocols to be used, Services to be used: TELNET, SSH, FTP Server, Web Server, File Server, DHCP Server and DNS Server.	CO6	03	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01
Text Books:	1. A. Tanenbaum, Computer Networks, Pearson. 2. B. Forouzan, Data Communications and Networking, McGraw Hill.			
Reference Books:	1. S. Keshav, An Engineering Approach to Computer Networks, Pearson. 2. B. Forouzan, TCP/IP Protocol Suite, McGraw Hill. 3. R. Bose, Information Theory, Coding and Cryptography, McGraw Hill. 4. K. Sayood and M. Kaufman, Introduction to Data Compression, Elsevier.			
Useful Links:	1. https://nptel.ac.in/courses/106/105/106105183/ 2. https://nptel.ac.in/courses/106/105/106105080/ 3. https://www.coursera.org/learn/tcpip 4. https://www.coursera.org/learn/fundamentals-network-communications			

Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC403	Operating Systems	03	-	-	03

Prerequisites:	Computer Programming (C / C++), Basic of Hardware i.e., ALU, RAM, ROM, HDD, etc., Computer-System Organization
Course Objectives (COBs):	1. To understand the major components of Operating System & their functions. 2. To introduce the notion of a process and its management like transition, scheduling, etc. 3. To understand basic concepts related to Inter-Process Communication (IPC) like mutual exclusion, deadlock, etc. and role of Operating System in IPC. 4. To understand the concepts and implementation of memory management policies and virtual memory. 5. To understand functions of Operating System for storage management and device management. 6. To study the need and fundamentals of special-purpose Operating System with the advent of new emerging technologies.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Explain the basic concepts related to Operating System. 2. Describe the process management policies and illustrate the scheduling of processes by CPU. 3. Apply synchronization primitives and evaluate deadlock conditions as handled by Operating System. 4. Explain the memory allocation and management functions of Operating Systems. 5. Explain the services provided by Operating System for storage management. 6. Compare the functions of various special-purpose Operating Systems.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Fundamentals of Operating System	Introduction to Operating Systems, Operating System Structure and Operations, Functions of Operating Systems.	CO1	01	03
	Operating System Services and Interface, System Calls and its Types, System Programs, Operating System Structure, System Boot.	CO6	02	
2. Process Management	Basic Concepts of Process, Operation on Process, Process State Model and Transition, Process Control Block, Context Switching.	CO2	04	10
	Introduction to Threads, Types of Threads, Thread Models.	CO2	01	
	Basic Concepts of Scheduling, Types of Schedulers, Scheduling Criteria, Scheduling Algorithms.	CO2	05	
3. Process Coordination	Basic Concepts of Inter-process Communication and Synchronization, Race Condition, Critical Region and Problem, Peterson's Solution, Synchronization Hardware	CO3	04	08

	and Semaphores, Classic Problems of Synchronization, Message Passing.			
	Introduction to Deadlocks, System Model, Deadlock Characterization, Deadlock Detection and Recovery, Deadlock Prevention, Deadlock Avoidance.	CO3	04	
4. Memory Management	Basic Concepts of Memory Management, Swapping, Contiguous Memory Allocation, Paging, Structure of Page Table, Segmentation.	CO4	05	09
	Basic Concepts of Virtual Memory, Demand Paging, Copy-on Write, Page Replacement Algorithms, Thrashing.	CO4	04	
5. Storage Management	Basic Concepts of File System, File Access Methods, Directory Structure, File System Implementation, Allocation Methods, Free Space Management.	CO5	03	06
	Overview of Mass-Storage Structure, Disk Structure, Disk Scheduling, RAID Structure, Introduction to I/O Systems.	CO5	03	
6. Special-Purpose Operating Systems	Open-source and Proprietary Operating System, Fundamentals of Distributed Operating System, Network Operating System, Embedded Operating Systems, Cloud and IoT Operating Systems, Real-Time Operating System, Mobile Operating System, Multimedia Operating System.	CO6	02	03
	Comparison between Functions of various Special-purpose Operating Systems.	CO6	01	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01

Text Books:	1. A. Silberschatz, P. Galvin, G. Gagne, Operating System Concepts, Wiley. 2. W. Stallings, Operating Systems: Internal and Design Principles, Pearson. 3. A. Tanenbaum, Modern Operating Systems, Pearson.
Reference Books:	1. N. Chauhan, Principles of Operating Systems, Oxford University Press. 2. A. Tanenbaum and A. Woodhull, Operating System Design and Implementation, Pearson. 3. R. Arpaci-Dusseau and A. Arpaci-Dusseau, Operating Systems: Three Easy Pieces, CreateSpace Independent Publishing Platform.
Useful Links:	1. https://nptel.ac.in/courses/106/106/106106144/ 2. https://onlinecourses.nptel.ac.in/noc21_cs44/preview 3. https://www.coursera.org/learn/os-power-user
Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Course Code	Course Name	Credits			
		TH	P	TUT	Total
ITC404	Automata Theory	03	-	-	03

Prerequisites:	Basic Mathematical Fundamentals: Sets, Logic, Relations, Functions.
Course Objectives (COBs):	1. To learn fundamentals of Regular and Context Free Grammars and Languages. 2. To understand the relation between Regular Language and Finite Automata and Machines. 3. To learn how to design Automata as Acceptors, Verifiers and Translators. 4. To understand the relation between Regular Languages, Contexts Free Languages, PDA and TM. 5. To learn how to design PDA as acceptor and TM as Calculators. 6. To learn applications of Automata Theory.
Course Outcomes (COs):	Upon completion of the course, the learners will be able to: 1. Explain, analyze and design Regular languages, Expression and Grammars. 2. Design different types of Finite Automata and Machines as Acceptor, Verifier and Translator. 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. Explain applications of various Automata.

Module No. and Name	Subtopics	COs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Introduction and Regular Languages	Languages: Alphabets and Strings, Regular Languages: Regular Expressions, Regular Languages.	CO1	03	06
	Regular Grammars, RL and LL Grammars.	CO1	02	
	Closure Properties.	CO1	01	
2. Finite Automata	Finite Automata: FA as Language Acceptor or Verifier.	CO2	02	09
	NFA (with and without ε).	CO2	01	
	DFA, RE to NFA, NFA to DFA, Reduced DFA, NFA-DFA equivalence, FA to RE.	CO2	04	
	Finite State Machines with output: Moore and Mealy Machines. Moore and Mealy M/C Conversion. Limitations of FA.	CO2	02	
3. Context Free Grammars	Context Free Languages: CFG.	CO2	03	08
	Leftmost and Rightmost derivations, Ambiguity.	CO3	02	
	Simplification and Normalization (CNF & GNF) and Chomsky Hierarchy (Types 0 to 3).	CO3	03	
4. Push Down Automata	Push Down Automata: Deterministic (Single Stack) PDA.	CO4	04	06
	Equivalence between PDA and CFG. Power and Limitations of PDA.	CO4	02	

5. Turing Machine	Turing Machine: Deterministic TM.	CO5	04	07
	Variants of TM, Halting problem, Power of TM.	CO5	03	
6. Applications of Automata	Applications of FA.	CO2	01	03
	Applications of CFG.	CO3	01	
	Applications of PDA.	CO4		
	Applications of TM.	CO5	01	
	Introduction to Compiler & Its phases.	CO6		
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01

Text Books:	1. J. Martin, Introduction to languages and the Theory of Computation, McGraw Hill. 2. K. Mahesh, Theory of Computation: A Problem-Solving Approach, Wiley. 3. A. Aho, R. Shethi, M. Lam and J. Ulman, Compilers Principles, Techniques and Tools, Pearson.
Reference Books:	1. J. Hopcroft, R. Motwani and J. Ullman, Introduction to Automata Theory, Languages and Computation, Pearson. 2. D. Cohen, Introduction to Computer Theory, Wiley. 3. V. Kulkarni, Theory of Computation, Oxford University Press. 4. N. Chandrashekhar, K. Mishra, Theory of Computer Science, Automata Languages & Computations, PHI. 5. J. Donovan, Systems Programming, McGraw Hill. 6. S. Agrawal, Theoretical Computer Science, Vikas Publications.
Useful Links:	1. https://nptel.ac.in/courses/111/103/111103016/ 2. https://online.stanford.edu/courses/soe-ycsautomata-automata-theory 3. http://www.jflap.org/
Continuous Assessment (CA):	• Continuous Assessment shall be conducted for Total 40 Marks, including 02 Tests of 20 marks each. • Duration of each Test will be 1 Hour and addition of scores in both the tests will be considered for passing.
End Semester Examination (ESE):	• End Semester Exam shall be conducted for Total 60 Marks. • Duration of End Semester Exam shall be 02 Hours and 30 Minutes.

Lab Code	Lab Name	Credits			
		TH	P	TUT	Total
ITL402	Network Lab	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	TCL, NS2.35, Ubuntu Operating System, Protocol Analyzer like Wireshark.
Prerequisites:	Computer Programming (C / C++).
Lab Objectives (LOBs):	1. To get familiar with the basic network administration commands. 2. To install and configure network simulator and learn basics of TCL scripting. 3. To understand the network simulator environment and visualize a network topology and observe its performance. 4. To implement client-server socket programs. 5. To observe and study the traffic flow and the contents of protocol frames. 6. To design and configure a network for an organization.
Lab Outcomes (LOs):	Upon completion of the course, the learners will be able to: 1. Execute and evaluate network administration commands and demonstrate their use in different network scenarios. 2. Demonstrate the installation and configuration of network simulator. 3. Demonstrate and measure different network scenarios and their performance behavior. 4. Implement the socket programming for client server architecture. 5. Analyze the traffic flow of different protocols. 6. Design a network for an organization using a network design tool.

Lab No.	Experiment Title	LOs Mapped	Hours
0	Lab Prerequisites.	-	02
1	Execute and analyze basic networking commands: ifconfig, ip, traceroute, tracepath, ping, netstat, ss, dig, nslookup, route, host, arp, hostname, curl or wget, mtr, whois, tcpdump.	LO1	02
2	Installation and configuring of NS-2 simulator and introduction to TCL using Hello program.	LO2	02
3	Write TCL scripts to create topologies.	LO2	02
4	Analysis of network performance for quality-of-service parameters such as packet-delivery-ratio, delay and throughput by plotting xgraph.	LO3	02
5	Implement Distance Vector Routing Protocols.	LO3	02
6	Implement Link State Routing Protocols.	LO3	02
7	Installation and configuring of Graphical Network Simulator GNS- 3.	LO2	02
8	Implement Topology in GNS - 3.	LO3	02
9	Implement Socket Programming using TCP with C/Java/python: TCP Client, TCP Server.	LO4	02
10	Implement Socket Programming using UDP with C/Java/python: UDP Client, UDP Server.	LO4	02
11	Install one of the Network Protocol Analyser Tools and Analyse the Traffic.	LO5	02
12	Network Design for an organization using the following concepts: 1. Addressing (IP Address Assignment); 2. Naming (DNS); 3. Routing	LO6	04

Virtual Lab Links:	1. http://vlabs.iitb.ac.in/vlabs-dev/labs_local/computer-networks/ 2. http://vlabs.iitb.ac.in/vlabs-dev/labs/mit_bootcamp/comp_networks_sm/
Term Work (TW):	1. Term work should consist of a minimum of 08 experiments. 2. Journal must include at least 02 assignments on content of theory course "Computer Network & Network Design" and "Network Lab". 3. Term work evaluation shall be for Total 25 Marks (Experiments: 20 Marks, Assignments: 05 Marks). 4. The final certification and acceptance of term work will be based on attendance in Theory and Lab sessions, satisfactory performance of laboratory work, and minimum passing marks in term work evaluation.
Practical & Oral (P&O):	P&O examination will be based on the experiment list for Total 25 Marks (Practical: 15 Marks and Oral: 10 Marks).

Lab Code	Lab Name	Credits			
		TH	P	TUT	Total
ITL403	Unix Lab	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	Unix / Ubuntu, Editor, VirtualBox.
Prerequisites:	Computer Programming (C / C++).
Lab Objectives (LOB):	1. To understand architecture and installation of Unix Operating System. 2. To learn Unix general purpose commands and programming in Unix editor environment. 3. To understand file system management and user management commands in Unix. 4. To understand process management and memory management commands in Unix. 5. To learn basic shell scripting. 6. To learn scripting using Awk and Perl languages.
Lab Outcomes (LO):	Upon completion of the course, the learners will be able to: 1. Explain the functioning of Unix, and use various PC OS alternatives like CPU OS Simulator, Cloud OS, etc. 2. Apply the Unix general purpose commands. 3. Apply Unix commands for system administrative tasks such as file system management and user management. 4. Apply Unix commands for system administrative tasks such as process management and memory management. 5. Implement basic shell scripts for different applications. 6. Implement advanced scripts using Awk & Perl languages and grep, sed, etc. commands for performing various tasks.

Lab No.	Experiment Title	LOs Mapped	Hours
0	Lab Prerequisites.	-	02
1	a. Case Study: Brief History of Unix, Unix Architecture; Installation of Unix Operating System. b. Installation and hands-on alternates for execution of Unix utilities: VirtualBox, VMware, etc.	LO1	02
2	Study and hands-on with various editors like Vi, Vim, nano, pico, etc.	LO1	02
3	Execution of Unix General Purpose Utility Commands like echo, clear, exit, date, time, uptime, cal, cat, tty, man, which, history, id, pwd, whoami, ping, ifconfig, pr, lp, lpr, lpstat, lpq, lprm, cancel, mail, etc.	LO2	02
4	a. Study of Unix file system (tree structure), file and directory permissions, single and multiuser environment. b. Execution of File System Management Commands like ls, cd, pwd, cat, mkdir, rmdir, rm, cp, mv, chmod, wc, piping and redirection, grep, tr, echo, sort, head, tail, diff, comm, less, more, file, type, wc, split, cmp, tar, find, vim, gzip, bzip2, unzip, locate, etc.	LO3	02
5	Execution of User Management Commands like who, whoami, su, sudo, login, logout, exit, passwd, useradd/adduser, usermod, userdel, groupadd, groupmod, groupdel, gpasswd, chown, chage, chgrp, chfn, etc.	LO3	02

6	a. Execution of Process Management Commands like ps, pstree, nice, kill, pkill, killall, xkill, fg, bg, pgrep, renice, etc. b. Execution of Memory Management Commands like free, /proc/meminfo, top, htop, df, du, vmstat, demidecode, sar, pagesize, etc.	LO4	02
7	Implementation of Scheduling Algorithms using CPU OS Simulator.	LO1, LO4	02
8	a. Study of Shell, Types of Shell, Variables and Operators b. Execute the following Scripts (at least 6): i) Write a shell script to perform arithmetic operations. ii) Write a shell script to calculate simple interest. iii) Write a shell script to determine the largest among three integer numbers. iv) Write a shell script to determine if a given year is leap year or not. v) Write a shell script to print the multiplication table of given numbers using while statement. vi) Write a shell script to search whether an element is present in the list or not. vii) Write a shell script to compare two strings. viii) Write a shell script to read and check if the directory / file exists or not, if not make the directory / file. ix) Write a shell script to implement a menu-driven calculator using case statements. x) Write a shell script to print following pattern: * * * * * * * xi) Write a shell script to perform operations on directory like: display name of current directory, display list of directory contents, create another directory — write contents on that and copy it to a suitable location in your home directory, etc.	LO5	06
9	Execute the following scripts using grep / sed commands: i) Write a script using grep command to find the number of words character, words and lines in a file. ii) Write a script using egrep command to display a list of specific types of files in the directory. iii) Write a script using sed command to replace all occurrences of a particular word in a given file. iv) Write a script using sed command to print duplicate lines in input.	LO5	04
10	a. Execute the following scripts using Awk / Perl languages: i) Write an Awk script to print all even numbers in a given range. ii) Write an Awk script to develop a Fibonacci series (take user input for number of terms). iii) Write a Perl script to sort elements of an array. b. Write a Perl script to check a number is prime or not.	LO6	02
Virtual Lab Links /	1. https://www.ee.iitb.ac.in/~vlabsync/ 2. http://www.ee.surrey.ac.uk/Teaching/Unix/unix2.html		

Learning Resources:	3. https://www.hackerrank.com/domains/shell 4. S. Das, Unix Concepts and Applications, McGraw Hill. 5. R. Michael, Mastering Unix Shell Scripting, Wiley. 6. D. Ambawade, D. Shah, Linux Labs and Open-Source Technologies, Dreamtech Press. 7. Y. Kanetkar, Unix Shell Programming, BPB Publications. 8. B. Forouzan and R. Gilberg, Unix and Shell Programming, Cengage Learning.
Term Work (TW):	1. Term work should consist of a minimum of 08 experiments. 2. Journal must include at least 02 assignments on content of the theory course "Operating Systems" and "Unix Lab". 3. Term work evaluation shall be for Total 25 Marks (Experiments: 20 Marks, Assignments: 05 Marks). 4. The final certification and acceptance of term work will be based on attendance in Theory and Lab sessions, satisfactory performance of laboratory work, and minimum passing marks in term work evaluation.
Practical (P):	Practical Examination will be based on the experiment list for Total 25 Marks.

PBL Course Code	PBL Course Name	Credits			
		TH	P	TUT	Total
ITPR42	Community Engagement PBL – Mini Project II	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	Python, MySQL.
Prerequisites:	Computer Programming (C / C++), Fundamentals of Python.
PBL Objectives (PROBs):	1. To create awareness among the students of the characteristics of several domain areas where IT can be effectively used. 2. To engage in community service, practice the process of identifying the needs and converting it into a problem statement. 3. To apply engineering knowledge and modern tools/technologies for deriving solutions to the real-world problems. 4. To inculcate the process of self-learning and research. 5. To be acquainted with solving the problem in a group. 6. To improve communication, management and report-writing skills of the students.
PBL Outcomes (PROs):	Upon completion of the course, the learners will be able to: 1. Identify societal / research needs through community engagement, formulate problem statements, review research literature, and analyze complex engineering problems. 2. Design suitable solutions for the problems including scope, objectives, timeline, system flow, user interface, algorithms, etc. 3. Gather, analyze, and interpret data — and apply knowledge of engineering fundamentals, modern tools / technologies for development of solutions. 4. Analyze sustainability and scalability of the developed solution and its impact in terms of environmental, societal, safety, legal, cultural, health, etc. aspects. 5. Apply ethical principles, excel in written and oral communication, and engage in independent and life-long learning. 6. Interact efficiently and effectively as an individual with the team members or leader for timely and professional management of projects.
Guidelines for Project-Based Learning (PBL):	1. Students have to form a team of minimum 02 and maximum 04 members, based on their area of interest and size of project. Interdisciplinary (inter-branch) teams are encouraged. 2. Students should carry out a field survey for community engagement, and identify needs, which shall be converted into problem statement for Mini Project in consultation with Faculty Guide, Internal committee of faculties, and the Head of Department. 3. Students should develop a Web / Mobile Application with a proper user interface using any suitable technology like HTML5, CSS, etc. for front end and Python at backend. 4. Projects should compulsorily be based on societal contribution (healthcare, agriculture, etc.) and reflecting role of engineer in the society. Students should try to take up need-based live projects so as to get exposure to communication with beneficiaries and skills for understanding client requirements. 5. Based on the idea presentation as well as discussion on feasibility, novelty, and contribution of the idea, a project definition will be finalized. 6. Students shall submit their implementation plan in the form of Gantt / PERT / CPM chart, which will cover weekly activity of the Mini project. 7. A log book is to be prepared by each group, wherein the group can record weekly work progress and the Faculty Guide can verify and record notes / comments.

	8. Faculty Guide may give inputs to students during Mini Project activity; however, focus shall be on self-learning. 9. Students in a group shall understand the problem effectively, propose multiple solutions, and select the best possible solution in consultation with their guide. 10. Students shall convert the best solution into a working model using various components of their domain areas and demonstrate. 11. The solution is to be validated with proper justification and report is to be compiled in standard format of the Department. 12. With the focus on self-learning, innovation, addressing societal problems and entrepreneurship quality development within the students through the Mini Projects, if the problem statement is huge and significant, a same problem statement can be worked upon for 02 semesters, i.e., same Mini Project in Semesters III and IV. Projects with a very large scope can also be taken forward to higher semesters, in consultation with the Head of the Department. 13. Students are encouraged to participate in Technical Paper Presentation competitions. 14. The students' group shall complete a project in all aspects including: Identification of need / problem, proposed final solution, Procurement of components / systems / data, Building prototype and testing. 15. Three reviews will be conducted for continuous assessment: one shall be for finalization of the problem and proposed solution, second shall be for evaluation of work progress, and third shall be for evaluation of implementation and testing of solutions. 16. Mini Project shall be assessed based on following parameters: • Attainment of Course Outcomes. • Technical efficiency and quality of developed solution. • Innovativeness in solutions. • Impact on environment. • Cost effectiveness. • Sustainability analysis. • Societal impact. • Effective use of standard engineering norms. • Contribution of an individual as member or leader. • Clarity in written and oral communication. 17. Students are encouraged to publish a paper based on the work in Conferences / Student competitions.
Useful Learning Links:	1. https://onlinecourses.nptel.ac.in/noc21_cs75/preview 2. https://www.coursera.org/specializations/python 3. https://www.udemy.com/course/the-complete-python-course/
Term Work (TW):	• Term Work shall be granted based individual's contribution in group activity, their understanding and response to questions. • Term Work evaluation shall be for Total 25 Marks — based on the following evaluation: o Presentation in Review 1 o Presentation in Review 2 o Presentation in Review 3 o Project Report and Log Book
Practical & Oral (P&O):	P&O examination will be of Total 25 Marks and shall be based on the Project Demonstration, Presentation, and Report.

Exposure Course Code	Exposure Course Name	Credits			
		TH	P	TUT	Total
ITXS48	Skill Enhancement – SAT VIII: Skill-Based Learning (Python Programming)	-	01	-	01

Hardware Requirements:	PC with i3 Processor or above.
Software Requirements:	Python, MySQL.
Skill Prerequisites:	Computer Programming (C / C++, Java).
Skill Objectives (SOBs):	1. To understand basics of Python including data types, operator, conditional statements, looping statements, input and output functions in Python. 2. To understand list, tuple, set, dictionary, string, array and functions in Python. 3. To impart knowledge of Object-Oriented Programming concepts in Python. 4. To explain concepts of modules, packages, multithreading and exception handling. 5. To understand knowledge of File handling, GUI & Database Programming. 6. To learn data visualization using Matplotlib, Data Analysis using Pandas and Web Programming using Flask.
Skill Outcomes (SOs):	Upon completion of the course, the learners will be able to: 1. Describe the structure, syntax, and semantics of the Python language. 2. Interpret advanced data types and functions in Python. 3. Illustrate the concepts of object-oriented programming as used in Python. 4. Develop Python applications using modules, packages, multithreading and exception handling. 5. Create solution with suitable GUI, File Handling functionalities and suitable database operations. 6. Develop cost-effective robust applications using the latest Python trends and technologies.

Module No. and Name	Subtopics	SOs Mapped	Hours / Subtopic	Total Hours / Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Basics of Python	Introduction, Features, Python building blocks – Identifiers, Keywords, Indention, Variables and Comments, Basic Data types (Numeric, Boolean, Compound).	SO1	01	03
	Operators: Arithmetic, Comparison, Relational, Assignment, Logical, Bitwise, Membership, Identity Operators, Operator Precedence.	SO1	01	
	Control Flow Statements: Conditional Statements (if, if…else, nested if) Looping in Python (while loop, for loop, nested loops) Loop Manipulation using continue, pass, break. Input / Output Functions, Decorators, Iterators and Generators.	SO1	01	

2. Advanced Datatypes and Functions	Lists: a) Defining lists, accessing values in List, deleting Values in List, Updating Lists b) Basic List Operations c) Built-in List Functions.	SO2	01	07
	Tuples: a) Accessing values in Tuples, deleting values in Tuples and updating Tuples b) Basic Tuple Operations c) Built-in Tuple Functions.	SO2	01	
	Dictionaries: a) Accessing values in Dictionary, deleting values in Dictionary and updating Dictionary. b) Basic Dictionary Operations c) Built-in Dictionary Functions.	SO2	01	
	Sets: a) Accessing values in Set, deleting values in Set, updating Sets b) Basic Set Operations. c) Built-in Set Functions.	SO2	01	
	Strings: a) String Initialization, Indexing, Slicing, Concatenation, Membership & Immutability b) Built-in String Functions.	SO2	01	
	Arrays: a) Working with Single dimensional Arrays: Creating, Importing, Indexing, Slicing, Copying and Processing Arrays. b) Working with Multi-Dimensional Arrays using Numpy: Mathematical Operations, Matrix Operations, Aggregate and other Built-in Functions.	SO2	01	
	Functions: a) Built-in Functions in Python. b) Defining Function, Calling Function, Returning Values, Passing Parameters. c) Nested and Recursive Functions d) Anonymous Functions (Lambda, Map, Reduce, Filter).	SO2	01	
3. Object-Oriented Programming	Overview of Object-oriented Programming, Creating Classes and Objects, Self-Variable, Constructors, Inner class, Static method, Namespaces.	SO3	01	03
	Inheritance: Types of Inheritance (Single, Multiple, Multi -level, Hierarchical), super() Method, Constructors in Inheritance, Operator Overloading, Method Overloading, Method Overriding.	SO3	01	
	Abstract Class, Abstract Method, Interfaces in Python.	SO3	01	
4. Modules, Packages, Multithreading and Exception Handling	Modules: Writing Modules, Importing Objects from Modules, Python Built-in Modules (e.g. Numeric and Mathematical Module, Functional Programming Module, Regular Expression Module), Namespace and Scoping.	SO4	01	04
	Packages: Creating User Defined Packages and Importing Packages.	SO4	01	
	Multi -Threading: Process Vs Thread, use of Threads, Types of Threads, Creating Threads in Python, Thread Synchronization, Deadlock of Threads.	SO4	01	
	Exception Handling: Compile Time Errors, Runtime Errors, Exceptions, Types of Exception, Try Statement, Except Block, Raise Statement, Assert Statement, User - Defined Exceptions.	SO4	01	

5. File Handling, GUI & Database Programming	File Handling: Opening File in Different Modes, Closing A File, Writing to A File, Accessing File Contents Using Standard Library Functions, Reading from A File – read (), readLine (), readLines (), Renaming and Deleting a File, File Exceptions, Pickle in Python.	SO5	01	03
	Graphical User Interface (GUI): Different GUI Tools in Python (Tkinter, Pyqt, Kivy, etc.), Working with Containers, Canvas, Frame, Widgets (Button, Label, Text, Scrollbar, Check Button, Radio Button, Entry, Spinbox, Message, etc.) Connecting GUI with Databases to Perform CRUD Operations. (On Supported Databases Like Sqlite, Mysql, Oracle, Postgresql, etc.).	SO5	02	
6. Data Visualization, Analysis and Web Programming using Python	Visualization Using Matplotlib: Matplotlib with Numpy, Working with Plots (Line Plot, Bar Graph, Histogram, Scatter Plot, Area Plot, Pie Chart, etc.), Working with Multiple Figures.	SO6	01	05
	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.	SO6	02	
	Scipy: Linear Algebra Functions using Numpy & Scipy.	SO6	01	
	Web Programming: Introduction to Flask, creating a Basic Flask Application, build a Simple REST API using Flask.	SO6	01	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	-	01	01
Text Books:	1. R. Nageswara Rao, Core Python Programming, Dreamtech Press, Wiley. 2. M. Savaliya, R. Maurya, Programming through Python, StarEdu Solutions. 3. E. Balagurusamy, Introduction to Computing and Problem-solving using Python, McGraw Hill.			
Reference Books:	1. Z. Shaw, Learn Python 3 the Hard Way, Zed Shaw's Hard Way Series. 2. M. Brown, Python: The Complete Reference, McGraw Hill. 3. P. Barry, Head First Python, 2nd Edition, O'Reilly Media.			
Useful learning Links:	1. https://docs.scipy.org/doc/numpy/user/quickstart.html 2. https://matplotlib.org/tutorials/ 3. https://pandas.pydata.org/docs/getting_started/ 4. https://www.geeksforgeeks.org/python-build-a-rest-api-using-flask/ 5. https://python-iitk.vlabs.ac.in/			
Guidelines for Skill-Based Learning (SBL):	• Programming labs shall be conducted as 02 Hours of blended theory and hands-on session. • The classes may be conducted as a flipped classroom, where students have to attend class after reviewing the lessons provided to them beforehand.			

	• Discussion on the topics and implementation of programs involving the concepts mentioned will be performed during the assigned lab hours.
Term Work (TW):	• Term Work evaluation shall be for Total 25 Marks based on Practical Performance. • The final certification and acceptance of term work will be based on satisfactory performance of laboratory work, and minimum passing marks in term work evaluation

Exposure Course Code	Exposure Course Name	Credits			
		TH	P	TUT	Total
ITXS49	Ability Enhancement – SAT IX: Skill-Based Learning (Foreign and/or Indian Modern Languages)	-	01	-	01

Skill Objectives (SOBs):	1. Acquire reading and writing proficiency in the target language 2. Understand the common heritage of, and diversity among, countries that speak the target language. 3. Communicate and interact effectively with citizens of the target cultures
Skill Outcomes (SOs):	Upon completion of the course, the learners will be able to: 1. Demonstrate of communicative proficiency in the target language. 2. Write the target language in formal expository prose that impede communication. 3. Learn through MOOC online courses to adopt hybrid mode of learning.
Guidelines for Skill-Based Learning (SBL):	Each student has to complete any one Foreign and/or Indian Language MOOC course from NPTEL / Coursera / Udemy, etc. sites referring the given suggestive list of courses, but not limited to the list as it is a learner's choice for the interested course, to be completed during the semester time frame.

Sr. No.	Suggestive List of Courses
1	Introduction to Japanese Language and Culture
2	German – I, II, III
3	The Psychology of Language
4	Spanish Vocabulary: Meeting People, Cultural Experience, Sports, Travel, and the Home, Careers and Social Events, Spanish Vocabulary Project
5	A Bridge to the World: Korean Language for Beginners, First Step Korean, Learn to Speak Korean 1, The Korean Alphabet: An Introduction to Hangeul
6	Complete French Course: Learn French for Beginners
7	Complete German Course: Learn German for Beginners
8	Spanish 1-4: Beginner, Elementary, Intermediate and Advanced
9	Complete Japanese Course: Learn Japanese for Beginners
10	Complete Korean Course: Learn Korean for Beginners
11	The Complete Russian Language Course
12	Spoken Sanskrit: Basic and Intermediate Levels
13	Applied Linguistics
14	Fundamental Concepts in Sociolinguistics
15	Introduction to Basic Spoken Sanskrit and Intermediate level to Basic Spoken Sanskrit

Learning Resources (Suggestive Courses Links but not limited to these only):	1. https://onlinecourses.nptel.ac.in/noc22_hs84/preview 2. https://onlinecourses.nptel.ac.in/noc22_hs89/preview 3. https://onlinecourses.nptel.ac.in/noc22_hs123/preview 4. https://www.coursera.org/learn/spanish-vocabulary-meeting-people 5. https://www.coursera.org/learn/spanish-vocabulary-cultural-experience 6. https://www.coursera.org/learn/spanish-vocabulary-sports-travel-home 7. https://www.coursera.org/learn/spanish-vocabulary-careers 8. https://www.coursera.org/learn/spanish-vocabulary-project 9. https://www.coursera.org/learn/korean-beginners 10. https://www.coursera.org/learn/learn-korean

	11. https://www.coursera.org/learn/learn-speak-korean1 12. https://www.coursera.org/learn/the-korean-alphabet-an-introduction-to-hangeul 13. https://www.udemy.com/course/complete-french-course/ 14. https://www.udemy.com/course/complete-german-course-learn-german-for-beginners/ 15. https://www.udemy.com/course/spanish-101-beginning-spanish-spanish-for-beginners/ 16. https://www.udemy.com/course/complete-japanese-course-learn-japanese-for-beginners-lvl-1/ 17. https://www.udemy.com/course/complete-korean-course-learn-korean-for-beginners-level-1/ 18. https://www.udemy.com/course/the-complete-russian-language-course/ 19. https://onlinecourses.nptel.ac.in/noc22_hs114/preview 20. https://onlinecourses.nptel.ac.in/noc22_hs85/preview 21. https://onlinecourses.nptel.ac.in/noc22_hs139/preview
Term Work (TW):	Term Work evaluation shall be for Total 25 Marks based on progress and completion of the course.

Baskets for Minors and Exit Courses

Multidisciplinary Minor (MM) Courses				
MM 1: Innovation and Entrepreneurship	**MM 2: Biotechnology**	**MM 3: IoT and Cloud Computing**	**MM 4: Geographical Information System**	**MM 5: Very-Large-Scale Integration (VLSI)**
Entrepreneurial Mindset	Introduction to Biotechnology	Introduction to Internet of Things	Spatial Computing Technologies	Processor Architecture and FPGA Design
Design Thinking	Biology, Society and Biomedical Issues	Connecting IoT Gateway using AWS Services	Digital Image Processing	Analog and Mixed-Signal IP Design
Fundraising, Finance, Due Diligence and Risk Management	Bioinformatics & Omics	Create Your Own IoT Solution	Geo-informatics and Technology	SoC Design and Implementation
Crafting Agreements, Negotiations and Pitching to Investors	Industrial Biotechnology	Building Industry IoT applications and Application Bank	Remote Sensing and Technology	Low Power VLSI Design
Design and Innovation of Business Models	Molecular Biology & Genetic Engineering	Cloud Computing	Geomatics	Chip Testing and Product Development
Ideation and Conceptualization using AI	Genomic Data Analysis	Automation using IoT	Remote Sensing and Sensors	Advanced VLSI CAD

Multiple Exit Courses*		
UG Certificate Exit Courses (04 Credits Each)	**UG Diploma Exit Courses (04 Credits Each)**	**Bachelor's in Vocation Exit Courses (04 Credits Each)**
MS Office	Multimedia and Animation	Software Testing and Quality Assurance
Digital Marketing	Database Administration	UI/UX Design
Network Administration	System Administration	Data Analytics Tools (Tableau, PowerBI, etc.)
Computer Hardware Maintenance	Go Programming	Mobile Application Development (Android / iOS)
Python Programming	Basic Web Development (PHP, HTML, CSS, etc.)	Advanced Full Stack Development (MERN, MEAN, etc.)
Mini Project	Mini Project	Mini Project
04 Weeks Internship	04 Weeks Internship	04 Weeks Internship
OR 06-08 Week Internship		

**To pursue 02 Courses of 04 Credits each OR 01 course of 04 Credits and 04 Week's Internship of 04 Credits OR 06-08 Week's Internship of 08 Credits.*