SOMAIYA
VIDYAVIHAR

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

7th AC Agenda
No. 4.B.2
Dated 5th July 2024

Autonomy Syllabus Scheme III (2023-24)

(As per NEP 2020 Guidelines)

for

Four Year Multidisciplinary

Bachelors of Technology (B.Tech.) Program

in

Computer Engineering

with

Multiple Entry and Multiple Exit Options

Levels 4.5 - 6

(First Year Effective from A.Y. 2023-24,
Second Year Effective from A.Y. 2024-25,
Third Year Effective from A.Y. 2025-26,
Last Year Effective from A.Y. 2026-27)

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

Chairperson BoS Computer Engineering:

The National Education Policy 2020 (NEP 2020) introduced by the Government of India, aims to ensure that quality education is accessible to all, while fostering multidisciplinary research and integrating skill-based learning with advanced technology. The Department of Higher and Technical Education, Maharashtra, emphasizes key elements for an NEP 2020-aligned curriculum includes, Core and Elective Courses, Vocational and Skill Enhancement Courses, Ability Enhancement Courses, Indian Knowledge System and Value Education, Co-curricular Activities and Projects, Multidisciplinary Minor Courses and Flexible Degree Options.

Since attaining autonomy in the academic year 2021-22, the computer engineering department at KJSIT has proactively implemented progressive educational reforms. Our existing Syllabus Schemes I, II, and II B have already integrated many NEP 2020 principles, focusing on robust core courses and a diverse array of electives that cater to both major-related and interdisciplinary interests. To achieve full alignment with NEP 2020, KJSIT has introduced the Autonomy Syllabus Scheme III, effective from the Academic Year 2023-24 starting from first year. According to this it is my privilege to present the revised autonomy scheme-III as per NEP implementation and detailed syllabus of Bachler of Technology, BTech in computer engineering from academic year 2024-25, for second year students i.e. semester III and IV. The syllabus introduces mandatory Multidisciplinary Minor courses in critical areas such as Innovation and Entrepreneurship, Biotechnology, IoT and Cloud Computing, GIS and VLSI. These courses encourage interdisciplinary collaboration and equip students to tackle complex challenges through innovative solutions.

The curriculum includes Skill-Based Learning (SBL), Activity-Based Learning (ABL), and Technology-Based Learning (TBL) courses, collectively known as exposure (SAT) courses. We have enhanced these offerings by aligning SAT courses into specialized categories like Vocational Skill (VS-SAT), Skill Enhancement (SE-SAT), Ability Enhancement (AE-SAT), and Value Education (VE-SAT), catering to varied student interests and career aspirations.

Apart from this, scheme III maintains the robust Project-Based Learning (PBL) framework from Scheme II, with students engaging in Community Engagement and Field Projects throughout their academic journey.

Moreover, institute continues its initiative of offering Honours courses aligned with emerging technologies recognized by AICTE, including IoT, AI & ML, Cyber Security, AR & VR, Data Science, and Blockchain. These specialized tracks are designed to meet industry standards and provide students with extensive opportunities for specialization and career advancement.

The board of studies appreciates the excellent work and contributions of the Coordinators and Members of the committee. They have given many inputs and guidelines for making the syllabi what it is now.

In conclusion, KJSIT's proactive approach to integrating NEP 2020 guidelines not only enhances educational quality but also prepares students to excel in a competitive global landscape, fostering innovation, leadership, and societal impact through robust, forward-thinking curricular frameworks.

Dr. Sarita P. Ambadekar
HOD Computer Engineering Department
Dr. Sarita Ambadekar

Chairperson BoS, Head of the Department

Two Experts in the subject from outside the parent University to be nominated by the Academic Council

1. Dr. Ratnadeep Deshmukh
   Professor & Head, Dept of CS & IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad.

2. Dr. Preeti Patil
   Associate Professor and Head of IT dept.
   D. Y. Patil Pratishthans D.Y. Patil College of Engineering, Pune

One expert to be nominated by the Vice-Chancellor from a panel of six recommended by the College Principal

1. Dr. Sangita Chaudhari
   Professor & Head Information Technology,
   Ramrao Adik Institute of Technology. Dr D Y Patil Vidyanagar, Nerul Navi Mumbai

Spatial invitee, Representative from Industry/Corporate Sector/Allied Area relating to Placement

1. Mr. Pankaj Kumar Bhagat
   Program Head-Intellectual Property (IP) at Mastek Limited
   Chennai, Tamil Nadu, India

The entire faculty members of each specialization

Dr. Shyamal Virnodkar,

Ms. Mrunali Desai,

Ms. Kavita Bathe,

Dr. Jyoti Wadmare

Ms. Chitra Bhole,

Ms. Nisha Vanjari,

Ms. Shubhada Labde,

Dr. Madar Bivalkar,

Dr. Dhanshree Toradmalle,

Ms. Aarti Sahitya,

Ms. Pradnya Patil,

Dr. Madhura Phadke,

Dr. Shreya Patankar,

Ms. Pradnya Bhangale,

Ms. Minal Sonkar,

Mr. Abhijit Patil,

Dr. Nilesh Yadav,

Dr. Surekha Janrao

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
- MM- Multidisciplinary Minor

Programs Offered with Multiple Entry Multiple Exit Options

Level 4.5: UG Certificate in Technology

Disciplines:	• Information Technology • Computer Engineering • Artificial Intelligence & Data Science • Electronics and Telecommunication
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

Disciplines:	• Information Technology • Computer Engineering • Artificial Intelligence & Data Science • Electronics and Telecommunication
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.)

Disciplines:	• Information Technology • Computer Engineering • Artificial Intelligence & Data Science • Electronics and Telecommunication
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

<table>
<tr><td>Major Disciplines with Offered Multidisciplinary Minors:</td><td>
<table>
<tr><th>Minor / Major</th><th>Innovation and Entrepreneurship</th><th>Biotechnology</th><th>IoT and Cloud Computing</th><th>Geographical Information System</th><th>VLSI</th></tr>
<tr><td>Information Technology</td><td>√</td><td>√</td><td>√</td><td>√</td><td>√</td></tr>
<tr><td>Computer Engineering</td><td>√</td><td>√</td><td>√</td><td>√</td><td>√</td></tr>
<tr><td>Artificial Intelligence & Data Science</td><td>√</td><td>√</td><td>√</td><td>√</td><td>√</td></tr>
<tr><td>Electronics & Telecommunication</td><td>√</td><td>√</td><td>√</td><td>√</td><td>√</td></tr>
</table>
</td></tr>
<tr><td>Years of Study:</td><td>04 Years</td></tr>
<tr><td>Semesters:</td><td>Major – 1, 2, 3, 4, 5, 6, 7, 8
Multidisciplinary Minors – 4, 5, 6</td></tr>
<tr><td>Credits:</td><td>174</td></tr>
</table>

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

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

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

Major Disciplines with Offered Honors and Multidisciplinary Minors:

Major	Honors with Research
Information Technology	√
Computer Engineering	√
Artificial Intelligence & Data Science	√
Electronics and Telecommunication	√

Major \ Minor	Innovation and Entrepreneurship	Biotechnology	IoT and Cloud Computing*	Geographical Information System	VLSI
Information Technology	√	√	√	√	√
Computer Engineering	√	√	√	√	√
Artificial Intelligence & Data Science	√	√	√	√	√
Electronics & Telecommunication	√	√	√	√	√

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 Disciplines with Multidisciplinary Minors and Specialization Minors:

Multidisciplinary Minors:

Major \ Minor	Innovation and Entrepreneurship	Biotechnology	IoT and Cloud Computing*	Geographical Information System	VLSI
Information Technology	√	√	√	√	√
Computer Engineering	√	√	√	√	√
Artificial Intelligence & Data Science	√	√	√	√	√
Electronics & Telecommunication	√	√	√	√	√

Specialization Minors:

06 additional courses (of minimum 12 week each), in another Engg. / Tech. discipline / Emerging Areas through MOOC – SWAYAM

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 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: COMPUTER ENGINEERING

TEACHING SCHEME

Course Code	Course Name	Teaching Scheme (Contact Hours)		Credits Assigned		Course Category
		TH – P – TUT	Total	TH – P – TUT	Total	
C301	Applications of Mathematics in Engineering-I	3 – 0 – 1	04	3 – 0 – 1	04	BS
C302	Data Structure	3 – 0 – 0	03	3 – 0 – 0	03	PC
C303	Database Management System	3 – 0 – 0	03	3 – 0 – 0	03	PC
C304	Digital Logic & Computer Architecture	3 – 0 – 0	03	3 – 0 – 0	03	PC
C305	Discrete Structures and Graph Theory	3 – 0 – 0	03	3 – 0 – 0	03	PC
L302	Data Structure Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
L303	Database Management System Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
L304	Digital Logic & Computer Architecture Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
PR31	Community Engagement PBL – Mini Project I	0 – 2 – 0	02[$]	0 – 1 – 0	01	PBL
XS37	Skill Enhancement - SAT VII: Skill-Based Learning: Object Oriented Programming with JAVA	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	
C301	Applications of Mathematics in Engineering-I	20	20	40	60	2.5	25	-	-	-	125
C302	Data Structure	20	20	40	60	2.5	-	-	-	-	100
C303	Database Management System	20	20	40	60	2.5	-	-	-	-	100
C304	Digital Logic & Computer Architecture	20	20	40	60	2.5	-	-	-	-	100
C305	Discrete Structures and Graph Theory	20	20	40	60	2.5	-	-	-	-	100
L302	Data Structure Lab	-	-	-	-	-	25	-	-	25	50
L303	Database Management System Lab	-	-	-	-	-	25	-	-	25	50
L304	Digital Logic & Computer Architecture Lab	-	-	-	-	-	25	-	-	-	25
PR31	Community Engagement PBL – Mini Project I	-	-	-	-	-	25	-	-	25	50
XS37	Skill Enhancement - SAT VII: Skill-Based Learning: Object Oriented Programming with JAVA	-	-	-	-	-	25	-	-	-	25
Total		**100**	**100**	**200**	**300**	**-**	**150**	**-**	**-**	**75**	**725**

SEM IV: TEACHING SCHEME

Course Code	Course Name	Teaching Scheme (Contact Hours)		Credits Assigned		Course Category
		TH – P – TUT	Total	TH – P – TUT	Total	
C401	Applications of Mathematics in Engineering-II	3 – 0 – 1	04	3 – 0 – 1	04	BS
C402	Analysis of Algorithms	3 – 0 – 0	03	3 – 0 – 0	03	PC
C403	Operating System	3 – 0 – 0	03	3 – 0 – 0	03	PC
C404	Theory of Computer Science	3 – 0 – 0	03	3 – 0 – 0	03	PC
C405	Multidisciplinary Minor Course	3 – 0 – 0	03	3 – 0 – 0	03	MM
L402	Analysis of Algorithms Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
L403	Operating System Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
L404	Multidisciplinary Minor Lab	0 – 2 – 0	02	0 – 1 – 0	01	MM
PR42	Community Engagement PBL – Mini Project II	0 – 2 – 0	02[$]	0 – 1 – 0	01	PBL
XS48	Skill Enhancement – SAT VIII: Skill-Based Learning: Python Programming	0 – 2[*] – 0	02	0 – 1 – 0	01	SE-SAT
XS49	Ability Enhancement – SAT IX: Skill-Based Learning: Indian/Foreign Modern language	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	
C401	Applications of Mathematics in Engineering-II	20	20	40	60	2.5	25	-	-	-	125
C402	Analysis of Algorithms	20	20	40	60	2.5	-	-	-	-	100
C403	Operating System	20	20	40	60	2.5	-	-	-	-	100
C404	Theory of Computer Science	20	20	40	60	2.5	-	-	-	-	100
C405	Multidisciplinary Minor Course	-	-	-	-	-	50	50	-	-	100
L402	Analysis of Algorithms Lab	-	-	-	-	-	25	-	-	25	50
L403	Operating System Lab	-	-	-	-	-	25	-	25	-	50
L404	Multidisciplinary Minor Lab	-	-	-	-	-	25	-	-	-	25
PR42	Community Engagement PBL – Mini Project II	-	-	-	-	-	25	-	-	25	50
XS48	Skill Enhancement – SAT VIII: Skill-Based Learning: Python Programming	-	-	-	-	-	25	-	-	-	25
XS49	Ability Enhancement – SAT IX: Skill-Based Learning: Indian/Foreign Modern language	-	-	-	-	-	25	-	-	-	25
Total		**80**	**80**	**160**	**240**	-	**225**	**50**	**25**	**50**	**750**

Course Code	Course Name	Credits Assigned			
		TH	P	TUT	Total
C301	Applications of Mathematics in Engineering-I	03	0	01	04

Prerequisites:	1. Engineering Mathematics-I 2. Engineering Mathematics-II
Course Objectives:	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:	**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. & Name	Sub Topics	CO Mapped	Hrs	Total Hrs/ Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction	**-**	**02**	**02**
1. Laplace Transform	Definition of Laplace transform, Condition of Existence of Laplace transform.	**CO1**	**01**	**07**
	Laplace Transform (L) of Standard Functions like eat, (at), (at), $sinh(at)$, $cosh(at)$ and tn, $n \geq 0$.		**02**	
	Properties of Laplace Transform: Linearity, First Shifting theorem, Second Shifting Theorem, change of scale Property, multiplication by t, Division by t, Laplace Transform of derivatives and integrals (Properties without proof).		**02**	

	Evaluation of integrals by using Laplace Transformation.		**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.		**02**	
	Inverse Laplace transform using Convolution theorem (without proof).		**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 and $2l$.		**02**	
	Fourier series of even and odd functions.		**02**	
	Fourier Transform-Fourier sine transform and Fourier cosine transform.		**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).		**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.		**02**	
	Harmonic function, Harmonic conjugate and orthogonal trajectories.		**02**	
5. Statistical Techniques	Karl Pearson's coefficient of correlation (r)	**CO5**	**01**	**06**
	Spearman's Rank correlation coefficient (R) (with repeated and non-repeated ranks)		**01**	
	Lines of regression		**02**	
	Fitting of first- and second-degree curves.		**02**	
6.Probability	Definition and basics of probability, conditional probability.	**CO6**	**01**	**06**
	Total Probability theorem and Bayes' theorem.		**01**	
	Discrete and continuous random variable with probability distribution and probability density function.		**02**	
	Expectation, Variance, Moment generating function, Raw and central moments up to 4th order.		**02**	
ii. **Course Conclusion**	Recap of Modules, Outcomes, Applications, and Summarization.	**-**	**01**	**01**
Total Hours				**42**
Text Books:	1. Higher Engineering Mathematics, Dr. B. S. Grewal, Khanna Publication. 2. Advanced Engineering Mathematics, Erwin Kreyszig, Wiley Eastern Limited . 3. Probability, Statistics and Random Processes, T. Veerarajan, McGraw-Hill Education.			

Reference Books:	1. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Narosa publication. 2. Complex Variables and Applications, Brown and Churchill, McGraw-Hill Education. 3. Theory and Problems of Fourier Analysis with applications to BVP, Murray Spiegel, Schaum's Outline Series.
Useful Links:	1. e-PGPathshala (inflibnet.ac.in) 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)	1. Each Student has to write at least 6 class tutorials on entire syllabus. 2. Journal must include at least 2 assignments on content of theory of the course. The distribution of Term Work marks will be as follows – • Class Tutorials on entire syllabus: 20 marks • Assignment: 05 marks
Assessment:	
Continuous Assessment for 40 marks: 1. Test 1 – 20 marks 2. Test 2 – 20 marks	
End Semester Theory Examination will be of 60-Marks for 02 hrs 30 min duration.	

Course Code	Course Name	Credits (TH+P+TUT)
C302	Data Structure	3 - 0 – 0

Prerequisite:	C programming
Course Objectives:	1. To discuss types of different data structures and concept of Abstract Data Type 2. To discuss the concept of stack and queue and apply them to various applications. 3. To describe the concept of link list and apply it to various applications 4. To introduce the different kinds of trees. 5. To discuss graph related concepts and traversals along with application. 6. To teach various searching techniques.
Course Outcomes:	**After successful completion of this course, learner will be able to:** 1. Describe types of data structure and write ADT. 2. Implement stack and different types of queues using array and their applications 3. Perform various types of link list operations and their applications 4. Perform operations on Binary Search Tree, AVL tree, Btree and B+Tree 5. Implement Graph traversals BFS, DFS and application of Graph in topological sorting 6. Describe various Hashing functions, Collision techniques and compare various searching techniques Linear Search, Binary Search and Hashing

Module No. & Name	Sub Topics	CO Mapped	Hrs	Total Hrs/ Module
i. Prerequisites and Course outline	Prerequisite Concepts and Course Introduction.	**-**	**01**	**01**
1. Introduction to Data Structures	Introduction to Data Structures, Concept of ADT,	**CO1**	**01**	**02**
	Types of Data Structures-Linear and Nonlinear, Operations on Data Structures.		**01**	
2.Stack and Queues	Introduction, ADT of Stack, Operations on Stack, Array Implementation of Stack	**CO2**	**01**	**09**
	Applications of Stack-Well formedness of Parenthesis		**01**	
	Infix to Postfix Conversion		**01**	
	Postfix Evaluation		**01**	
	Recursion		**01**	
	Introduction, ADT of Queue, Operations on Queue, Array Implementation of Queue		**01**	
	Implementation of circular and Double Ended Queue, Priority Queue, Applications of Queue		**03**	
3. Linked List	Introduction, Representation of Linked List, Linked List v/s Array.	**CO3**	**01**	**10**

	Types of Linked List - Singly Linked List, Circular Linked List, Doubly Linked List, Operations on Singly Linked List and Doubly Linked List		**06**	
	Stack and Queue using Singly Linked List		**01**	
	Singly Linked List Application-Polynomial Representation and Addition		**02**	
4. Trees	Introduction, Tree Terminologies, Binary Tree, Binary Tree Representation, Types of Binary Tree	**CO4**	**01**	**11**
	Binary Tree Traversals		**02**	
	Binary Search Tree, Operations on Binary Search Tree		**04**	
	Applications of Binary Tree-Expression Tree, Huffman Encoding		**01**	
	Search Trees-AVL, rotations in AVL Tree, operations on AVL Tree, Introduction of B Tree, B+ Tree		**03**	
5. Graphs	Introduction, Graph Terminologies, Representation of Graph	**CO5**	**01**	**04**
	Graph Traversals-Depth First Search (DFS) and Breadth First Search (BFS)		**02**	
	Graph Application- Topological Sorting		**01**	
6. Searching Techniques	Linear Search, Binary Search, Hashing-Concept, Hash Functions	**CO6**	**02**	**04**
	Collision Resolution Techniques		**02**	
ii. Course conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	**--**		**01**
Total Hours				**42**

Books:

Textbooks:	1. Aaron M Tenenbaum, Yedidyah Langsam, Moshe J Augenstein, "Data Structures Using C", Pearson Publication. 2. Reema Thareja, "Data Structures using C", Oxford Press. 3. Richard F. Gilberg and Behrouz A. Forouzan, "Data Structures: A Pseudocode Approach with C", 2ndEdition, CENGAGE Learning. 4. Jean Paul Tremblay, P. G. Sorenson, "Introduction to Data Structure and Its Applications", McGraw-Hill Higher Education 5. Data Structures Using C, ISRD Group, 2ndEdition, Tata McGraw-Hill.
Reference Books:	1. Prof. P. S. Deshpande, Prof. O. G. Kakde, "C and Data Structures", DreamTech press. 2. E. Balagurusamy, "Data Structure Using C", Tata McGraw-Hill Education India. 3. Rajesh K Shukla, "Data Structures using C and C++", Wiley-India 4. GAV PAI, "Data Structures", Schaum's Outlines. 5. Robert Kruse, C. L. Tondo, Bruce Leung, "Data Structures and Program, Design in C", Pearson

Useful Links:	

1. https://nptel.ac.in/courses/106/102/106102064/
2. https://www.coursera.org/specializations/data-structures-algorithms
3. https://www.edx.org/course/data-structures-fundamentals
4. https://swayam.gov.in/nd1_noc19_cs67/preview

Assessment:

Continuous Assessment for 40 marks:

1. Test 1 – 20 marks
2. Test 2 – 20 marks

End Semester Theory Examination will be of 60 marks of 02 hrs min 30 duration.

Course Code	Course Title		Credits (TH+P+TUT)	
C303	**Database Management System**		**3 - 0 - 0**	
Prerequisite:	Data Structures			
Course Objectives:	1. Develop entity relationship data model and its mapping to relational model 2. Learn relational algebra and Formulate SQL queries 3. Apply normalization techniques to normalize the database 4. Understand the concept of transaction, concurrency control and recovery techniques.			
Course Outcomes:	**After completion of the course students will be able to..** 1. Recognize the need of database management system 2. Design ER and EER diagram for real life applications 3. Construct relational models and write relational algebra queries. 4. Formulate SQL queries 5. Apply the concept of normalization to relational database design. 6. Describe the concept of transaction, concurrency and recovery.			
Module No. & Name	**Sub-Topics**	**CO mapped**	**Hrs / Sub Topics**	**Hrs/ Module**
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction	**-**	**01**	**01**
1.Introduction Database Concepts	Introduction, Characteristics and applications of databases, File system v/s Database system,	**CO1**	**01**	**03**
	Data abstraction and data Independence, DBMS system architecture, Database Administrator		**02**	
2. Entity–Relationship Data Model	The Entity-Relationship (ER) Model: Entity types: Weak and strong entity sets, Entity sets, Types of Attributes, Keys	**CO2**	**03**	**06**
	Relationship constraints: Cardinality and Participation, Extended Entity-Relationship (EER) Model: Generalization, Specialization and Aggregation		**03**	
3. Relational Model and relational Algebra	Introduction to the Relational Model, relational schema and concept of keys.	**CO3**	**02**	**08**
	Mapping the ER and EER Model to the Relational Model		**03**	
	Relational Algebra-operators, Relational Algebra Queries		**03**	
4.Structured Query Language (SQL)	Overview of SQL, Data Definition Commands, Integrity constraints: key constraints, Domain Constraints, Referential integrity, check constraints	**CO4**	**02**	**07**
	Data Manipulation commands, Data Control commands		**01**	

	Set and string operations, aggregate function-group by, having, Views in SQL, joins, Nested and complex queries, Triggers, PL/SQL		**04**	
5.Relational-Database Design	Pitfalls in Relational-Database designs, Concept of normalization, Function Dependencies	**CO5**	**03**	**06**
	First Normal Form, 2NF, 3NF, BCNF, 4NF (Conversion of Normalization forms)		**03**	
6.Transactions Management and Concurrency and Recovery	Transaction concept, Transaction states, ACID properties, Transaction Control Commands	**CO6**	**02**	**10**
	Concurrent Executions, Serializability-Conflict and View, Concurrency Control: Lock-based		**04**	
	Timestamp-based protocols, Recovery System: Log based recovery, Deadlock handling		**04**	
ii. Course Conclusion:	Recap of Modules, Outcomes, Applications and Summarization.	**-**	**01**	**01**
Total Hrs				**42**

Textbooks:	1. Korth, Slberchatz, Sudarshan, Database System Concepts, 6thEdition, McGraw Hill 2. Elmasri and Navathe, Fundamentals of Database Systems, 5thEdition, Pearson Education 3. Raghu Ramkrishnan and Johannes Gehrke, Database Management Systems, TMH
References:	1. Peter Rob and Carlos Coronel, Database Systems Design, Implementation and Management‖, Thomson Learning, 5th Edition. 2. Dr. P.S. Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press. 3. G. K. Gupta, Database Management Systems, McGraw Hill, 2012
Useful Links	1. .https://nptel.ac.in/courses/106/105/106105175/ 2. .https://swayam.gov.in/nd1_noc19_cs46/preview 3. https://www.classcentral.com/course/swayam-database-management-system-9914 4. .https://www.mooc-list.com/tags/dbms

Assessment:

Continuous Assessment for 40 marks:

1. Test 1– 20 marks
2. Test 2 – 20 marks

End Semester Theory Examination will be of 60 marks for 02 hrs 30 min duration.

Course Code	Course Name	Credits (TH+P+TUT)
C304	Digital Logic & Computer Architecture	3-0-0

Prerequisite:	Knowledge on number systems
Course Objectives:	1. To have the rough understanding of the basic structure and operation of basic digital circuits and a digital computer. 2. To discuss in detail arithmetic operations in digital systems. 3. To discuss generation of control signals and different ways of communication with I/O devices. 4. To study the hierarchical memory and principles of advanced computing.
Course Outcomes:	**After the successful completion of this course, learner will be able to:** 1. Learn different number systems and basic structure of computer systems. 2. Demonstrate the arithmetic algorithms. 3. Describe the basic concepts of digital components and processor organization. 4. Explain the generation of control signals of computers. 5. Demonstrate the memory organization. 6. Describe the concepts of parallel processing and different Buses.

Module No. & Name	Sub Topics	CO mapped	Hrs / Sub Topics	Total Hrs/ Module
i. Prerequisites and Course outline	Prerequisite Concepts and Course Introduction.	**-**	**02**	**02**
1. Computer Fundamentals	Introduction to Number System and Codes Number Systems : Binary, Octal, Decimal, Hexadecimal	**CO1**	**01**	**06**
	Codes: Grey, BCD, Excess-3, ASCII, Boolean Algebra		**02**	
	Logic Gates: AND, OR, NOT, NAND, NOR, EX-OR		**01**	
	Overview of computer organization and architecture. Basic Organization of Computer and Block Level functional Units, Von- Neumann Model		**02**	
Data Representation and Arithmetic Algorithms	Binary Arithmetic: Addition, Subtraction, Multiplication.	**CO1, CO2**	**01**	**08**
	Division using Sign Magnitude, 1's and 2's compliment		**02**	
	BCD and Hex Arithmetic Operation		**01**	
	Booths Multiplication Algorithm, Restoring and Non-restoring Division Algorithm. IEEE-754 Floating point Representation		**04**	
3.Processor Organization and Architecture	Introduction: Half adder, Full adder, MUX, DMUX, Encoder, Decoder(IC level)	**CO3**	**02**	**06**
	Introduction to Flip Flop: SR, JK, D, T (Truth table)		**02**	

	Register Organization, Instruction Formats, Addressing modes, Instruction Cycle, Interpretation and sequencing		**02**	
4.Control Unit Design	Hardwired Control Unit: State Table Method, Delay Element Methods	**CO4**	**03**	**06**
	Microprogrammed Control Unit: Micro Instruction-Format, Sequencing and execution, Micro operations, Examples of microprograms		**03**	
5. Memory Organization	Introduction and characteristics of memory, Types of RAM and ROM, Memory Hierarchy, 2-level Memory Characteristic	**CO5**	**03**	**06**
	Cache Memory: Concept, locality of reference, Design problems based on mapping techniques, Cache coherence and write policies. Interleaved and Associative Memory		**03**	
6. Principles of Advanced Processor and Buses	Basic Pipelined Data path and control, data Dependencies	**CO6**	**02**	**08**
	Data hazards, branch hazards, delayed branch, and branch prediction, Performance measures-CPI, Speedup, Efficiency, throughput		**02**	
	Amdhal's law. Flynn's Classification, Introduction to multicore architecture		**02**	
	Introduction to buses: ISA, PCI, USB. Bus Contention and Arbitration		**02**	
ii.Course conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	**--**	**01**	**01**
Total Hours				**42**

Books:	
Textbooks	1. R. P. Jain, "Modern Digital Electronic", McGraw-Hill Publication, 4thEdition. 2. William Stalling, "Computer Organization and Architecture: Designing and Performance", Pearson Publication 10TH Edition. 3. John P Hayes, "Computer Architecture and Organization", McGraw-Hill Publication, 3RD Edition. 4. Dr. M. Usha and T. S. Shrikanth, "Computer system Architecture and Organization", Wiley publication.
Reference Books	1. Andrew S. Tanenbaum, "Structured Computer Organization", Pearson Publication. 2. B. Govindarajalu, "Computer Architecture and Organization", McGraw-Hill Publication. 3. Malvino, "Digital computer Electronics", McGraw-Hill Publication, 3rd edition. 4. Smruti Ranjan Sarangi, "Computer Organization and Architecture", McGraw-Hill Publication.
Useful Links:	1. https://www.classcentral.com/course/swayam-computer-organization-and-architecture-a- pedagogical- aspect-9824 2. https://nptel.ac.in/courses/106/103/106103068/ 3. https://www.coursera.org/learn/comparch

	4. https://www.edx.org/learn/computer-architecture
Assessment:	
Continuous Assessment for 40 marks: 1. Test 1 – 20 marks 2. Test 2 – 20 marks	
End Semester Theory Examination will be of 60 marks of 02 hrs 30 min duration.	

Course Code	Course Name	Credits (TH)
C305	Discrete Structures and Graph Theory	3

Prerequisite:	1. Basic Mathematics
Course Objectives:	1. Cultivate clear thinking and creative problem solving. 2. Thoroughly train in the construction and understanding of mathematical proofs. Exercise common mathematical arguments and proof strategies. 3. To apply graph theory in solving practical problems. 4. Thoroughly prepare for the mathematical aspects of other Computer Engineering courses.
Couse Outcomes:	**On successful completion, of course, learner will be able to:** 1. Analyze the Problems and its statements logically. 2. Apply the relations, functions, Diagraph and Lattice. 3. Apply the notion of mathematical thinking, mathematical proofs and to apply them in problem solving. 4. Identify problems concepts of graph theory in solving real world problems 5. Examine the groups and codes in Encoding-Decoding. 6. Analyze a complex computing problem and apply principles of discrete mathematics to identify solutions

Module No	Topics	CO mapped	Hrs /Subtopic	Total Hrs/ Module
i. Prerequisites and Course outline	Prerequisite Concepts and Course Introduction	**-**	**01**	**01**
1. Logic	Propositional Logic, Predicate Logic, Laws of Logic, Quantifiers	**CO1**	**03**	**06**
	Normal Forms, Inference Theory of Predicate Calculus, First order logic, Mathematical Induction.		**03**	
2.Relations and Functions	Basic concepts of Set Theory	**CO2**	**01**	**06**
	Relations: Definition, Types of Relations, Representation of Relations, Closures of Relations		**02**	
	Warshall's algorithm, Equivalence relations and Equivalence Classes		**01**	
	Functions: Definition, Types of functions, Composition of functions, Identity and Inverse function		**02**	
3.. Posets and Lattice	Partial Order Relations, Poset, Hasse Diagram	**CO3**	**02**	**05**
	Chain and Antichains, Lattice, Types of Lattice, Sub lattice		**03**	
4. Counting	Basic Counting Principle- , Product Rule, Inclusion-Exclusion Principle, Pigeon hole Principle	**CO4**	**03**	**06**
	Recurrence relations, Solving recurrence relations, types		**03**	

5. **Algebraic Structures**	**Algebraic structures with one binary operation:** Semi group, Monoid, Groups, Subgroups, Abelian Group, Cyclic group, Isomorphism.	**CO5**	**06**	**08**
	Algebraic structures with two binary operations: Ring.		**02**	
	Coding Theory: Coding, binary information and error detection, decoding and error correction.		**02**	
6. **Graph Theory**	Types of graphs, Graph Representation, Sub graphs, Operations on Graphs, Walk, Path, Circuit	**CO6**	**04**	**08**
	Connected Graphs, Disconnected Graph, Components, Homomorphism and Isomorphism of Graphs,		**02**	
	Euler and Hamiltonian Graphs, Planar Graph, Cut Set, Cut Vertex, Applications.		**02**	
ii.Course conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	**--**	**01**	**01**
Total Hours				**42**

Books:

Text Books	1 Bernad Kolman, Robert Busby, Sharon Cutler Ross, Nadeem -ur Rehman, "Discrete Mathematical Structures", Pearson Education. 2 C.L.Liu" Elements of Discrete Mathematics", second edition 1985, McGraw-Hill Book Company. Reprinted 2000. 3 K.H.Rosen," Discrete Mathematics and applications", fifth edition2003,Tata McGraw Hill Publishing Company
Reference Books	1 Y N Singh," Discrete Mathematical Structures", Wiley-India. 2 J.L.Mott, A.Kandel, T.P.Baker," Discrete Mathematics for Computer Scientists and Mathematicians", Second Edition 1986, Prentice Hall of India. 3. J.P.Trembley, R.Manohar" Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill Publishing Company 4. Seymour Lipschutz, Marc Lars Lipson, "Discrete Mathematics" Schaum"s Outline, McGraw Hill Education. 5. Narsing Deo, "Graph Theory with applications to engineering and computer science", PHI Publications. 6. P.K. Bisht , H.S.Dhami, "Discrete Mathematics", Oxford press.

Useful Links:

1.https://www.edx.org/learn/discrete-mathematics

2.https://www.coursera.org/specializations/discrete-mathematics

3.https://nptel.ac.in/courses/106/106/106106094/

4.https://swayam.gov.in/nd1_noc19_cs67/preview

Test (T1& T2):

Two tests must be conducted of which should cover at least 80% of syllabus.

Continuous Assessment for 40 marks:

1. Test 1 – 20 marks
2. Test 2 – 20 marks

End Semester Theory Examination will be of 60 marks of 02 hrs 30 min duration.

Lab Code	Lab Name	Credits (P+TUT)
L302	**Data Structures Lab**	0- 1-0

Lab Prerequisite:	C Programming
Lab Objectives:	1. To implement basic data structures such as linked lists, stacks and queues 2. To solve problem involving graphs and trees 3. To choose appropriate data structure and apply it to various problems
Lab Outcomes (LOs):	**At the end of the course, the student will be able to** 1. Implement linear data structures & be able to handle operations like insertion, deletion, searching and traversing on them. 2. Implement nonlinear data structures & be able to handle operations like insertion, deletion, searching and traversing on them 3. Choose appropriate data structure and apply it in various problems 4. Select appropriate searching techniques for given problems. 5. Apply ethical principles like timeliness and adhere to the rules of the laboratory.

Expt. No.	Experiment Title	LO mapped	Hrs / Lab
0	Prerequisite	**-**	**02**
1	Implement Stack ADT using array.	**LO1, LO5**	**02**
2	Convert an Infix expression to Postfix expression using stack ADT.	**LO1, LO3, LO5**	**02**
3	Evaluate Postfix Expression using Stack ADT.	**LO1, LO3, LO5**	**02**
4*	At least 2 applications of Stack from the **useful links/any other** given below.	**LO1, LO3, LO5**	**02**
5	Implement Linear Queue ADT using array.	**LO1, LO3, LO5**	**02**
6	Implement Circular/Double ended Queue ADT using array.	**LO1, LO3, LO5**	**02**
7	Implement Priority Queue ADT using array.	**LO1, LO3, LO5**	**02**
8	Implement Singly Linked List ADT.	**LO1, LO3, LO5**	**02**
9	Implement Circular Linked List ADT.	**LO1, LO3, LO5**	**02**
10	Implement Doubly Linked List ADT.	**LO1, LO3, LO5**	**02**
11	Implement Stack / Linear Queue ADT using Linked List.	**LO1, LO3, LO5**	**02**
12*	Implement Binary Search Tree ADT using Linked List.	**LO2, LO3, LO5**	**02**
13*	Implement Graph Traversal techniques:) Depth First Search b) Breadth First Search	**LO2, LO3, LO5**	**02**
14*	At least 2 applications of Binary Search Technique from the **useful links/any other** given below	**LO4, LO5**	**02**

Useful Links:

1. www.leetcode.com

Lab Code	Lab Name	Credits (P+TUT)
L303	**Database Management System Lab**	**0-1-0**

Prerequisite:	Data structures
Lab Objectives:	1. To explore design and develop of relational model 2. To present SQL and procedural interfaces to SQL comprehensively 3. To introduce the concepts of transactions and transaction processing
Lab Outcomes (LOs):	**At the end of the course, the students will be able to** 1. Design ER /EER diagram and convert it to a relational model for the real world application. 2. Apply DDL, DML, DCL and TCL commands 3. Write simple and complex queries 4. Use PL / SQL Constructs. 5. Demonstrate the concept of concurrent transactions execution and frontend-backend connectivity 6. Apply ethical principles like timeliness and adhere to the rules of the laboratory.

Suggested List of Experiments

Lab No.	Title of Experiment	LO Mapped	Hrs / Lab
0	Prerequisite	**-**	**02**
1	Identify the case study and detailed statement of the problem. Design an Entity-Relationship (ER) / Extended Entity-Relationship (EER) Model.	**LO1, LO6**	**02**
2	Mapping ER/EER to Relational schema model.	**LO1, LO6**	**02**
3	Create a database using Data Definition Language (DDL) and apply integrity constraints for the specified System	**LO2, LO6**	**02**
4	Apply DML Commands for the specified system	**LO2, LO6**	**02**
5	Perform Simple queries, string manipulation operations and aggregate functions.	**LO3, LO6**	**02**
6	Implement various Join operations.	**LO3, LO6**	**02**
7	Perform Nested and Complex queries	**LO3, LO6**	**02**
8	Perform DCL and TCL commands	**LO2, LO6**	**02**
9	Implement procedure and functions	**LO4, LO6**	**02**
10	Execution of CRUD operations from front end using Database connectivity.	**LO5, LO6**	**02**
11	Implementation of Views and Triggers.	**LO4, LO6**	**02**
12	Implementation and demonstration of Transaction and Concurrency control techniques using locks.	**LO5, LO6**	**02**

2. www.hackerrank.com
3. www.cs.usfca.edu/~galles/visualization/Algorithms.html
4. www.codechef.com
5. https://learndsa.kjsieit.in/

Term work:

1. Term work should consist of 10 experiments.
2. Journal must include at least 2 assignments.
3. The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work.
4. Total 25 Marks (Experiments: 20-marks, Assignments: 05-marks)

Oral & Practical Exam:

Oral & Practical Exam will be based on the entire syllabus of C303 and L303

Term Work:

Term work should consist of 10 experiments.

Journal must include at least 2 assignments on content of theory and practical of "Database Management System"

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

Total 25 Marks (Experiments: 20-marks, Assignments: 05-marks)

Lab Code	Lab Name	Credits(P+TUT)
L304	**Digital Logic & Computer Architecture Lab**	**0-1-0**

Lab Prerequisite:	C Programming Language
Lab Objectives:	1. To implement operations of the arithmetic unit using algorithms. 2. Design and simulate different digital circuits. 3. To design memory subsystems including cache memory. 4. To demonstrate CPU and ALU design.
Lab Outcomes (LOs):	**At the end of the course, the student will be able to** 1. Describe the basics of digital components 2. Design the basic building blocks of a computer: ALU, registers, CPU and memory 3. Recognize the importance of digital systems in computer architecture 4. Implement various algorithms for arithmetic operations. 5. Apply ethical principles like timeliness and adhere to the rules of the laboratory.

Lab No.	Experiment Title	LO mapped	Hrs/Lab
0	Prerequisite	**-**	**02**
1	To verify the truth table of various logic gates using ICs.	**LO1, LO5**	**02**
2	To realize the gates using universal gates	**LO1, LO5**	**02**
3	Code conversion.	**LO1, LO5**	**02**
4	To realize half adder and full adder.	**LO2, LO5**	**02**
5	To implement logic operation using MUX IC.	**LO3, LO5**	**02**
6	To implement logic operation decoder IC.	**LO3, LO5**	**02**
7	Study of flip flop IC.	**LO3, LO5**	**02**
8	To implement ripple carry adder.	**LO3, LO5**	**02**
9	To implement carry look ahead adder.	**LO3, LO5**	**02**
10	To implement Booth's algorithm.	**LO4, LO5**	**02**
11	To implement a restoring division algorithm.	**LO4, LO5**	**02**
12	To implement non restoring division algorithm.	**LO4, LO5**	**02**
13	To implement ALU design.	**LO2, LO5**	**02**
14	To implement CPU design.	**LO2, LO5**	**02**
15	To implement memory design.	**LO2, LO5**	**02**
16	To implement cache memory design.	**LO2, LO5**	**02**

Notes:

1. Any Four experiments from Exp. No. 1 to Exp. No. 7 using hardware.
2. Any Six experiments from Exp. No. 8 to Exp. No. 16 using Virtual Lab, expect Exp. No. 10,11 and 12.
3. Exp. No. 10 to Exp. No. 12 using Programming language.

Useful Link: http://cse10-iitkgp.virtual-labs.ac.in/

Term work:

1. Term work should consist of minimum 10 experiments
2. Journal must include at least 2 assignments on content of theory and practical of the course "Digital Logic & Computer Organization and Architecture"

3. The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work.
4. Total 25 Marks (Experiments: 20-marks, Assignments: 05-marks)

Virtual Lab Link:

http://vlabs.iitb.ac.in/vlabs-dev/vlab_bootcamp/bootcamp/bots_with_dots/labs/index.html

Assessment:

Term Work for 25 Marks:

Programming labs to be conducted as 2hrs continuous theory + hands-on session.
The assessment will be
• An online quiz conducted at the end of every 2-hr session consisting of 5 questions for a total of 10 marks.
The average of best 10 quizzes will be considered toward 10 marks out of 25.
Students should perform minimum 12 experiments.
The programs performed along with the screenshot of output have to be submitted.
A cover page will be attached stating the aims and objectives. This will be considered towards 10 marks

Course code	Course Name	Credits
PR31	**Community Engagement//Field Project- Mini Project -I**	**0-1-0**

PBL Objectives:	1. To acquaint with the process of identifying the needs and converting it into the problem. 2. To familiarize the process of solving the problem in a group. 3. To acquaint with the process of applying basic engineering fundamentals to attempt solutions to the problems. 4. To inculcate the process of self-learning and research.
PBL Outcomes (PROs):	**At the end of the course, the student will be able to:** 1. Identify problems based on societal /research needs. 2. Apply Knowledge and skill to solve societal problems in a group. 3. Develop interpersonal skills to work as member of a group or leader. 4. Analyze the impact of solutions in societal and environmental context for sustainable development. 5. Excel in written and oral communication. 6. Demonstrate capabilities of self-learning in a group, which leads to lifelong learning. 7. Demonstrate project management principles during project work.

Guidelines for Mini Project

1	Project based learning Mini Project Lab-1 should be implemented preferably using Java programming (CEXS33)
2	Students shall form a group of 2 to 3 students, while forming a group shall not be allowed less than two or more than three students, as it is a group activity.
3	Students should do survey and identify needs, which shall be converted into problem statement for mini project in consultation with faculty supervisor/internal committee of faculties.
4	Students shall submit implementation plan in the form of Gantt/PERT/CPM chart, which will cover weekly activity of mini project.
5	A logbook to be prepared by each group, wherein group can record weekly work progress, guide/supervisor can verify and record notes/comments.
6	Faculty supervisor may give inputs to students during mini project activity; however, focus shall be on self-learning.
7	Students in a group shall understand problem effectively, propose multiple solution and select best possible solution in consultation with guide/ supervisor.
8	Students shall convert the best solution into working model using Java programming.
9	The solution to be validated with proper justification and report to be compiled in standard format of the college.

10	With the focus on the self-learning, innovation, addressing societal problems and entrepreneurship quality development within the students through the Mini Projects, it is preferable that a single project of appropriate level and quality to be carried out in two semesters by all the groups of the students. i.e. Mini Project 1 in semester III and IV.
11	However, based on the individual students or group capability, with the mentor's recommendations, if the proposed Mini Project adhering to the qualitative aspects mentioned above gets completed in odd semester, then that group can be allowed to work on the extension of the Mini Project with suitable improvements/modifications or a completely new project idea in even semester. This policy can be adopted on case by case basis.

Term Work:

The review/ progress monitoring committee shall be constituted by senior faculty members. The progress of mini project to be evaluated on continuous basis, minimum two reviews in each semester. Assessment also considers peer review and ethics observed by faculties and participation involvement.

Continuous Assessment:

In continuous assessment focus shall also be on each individual student, log book maintained and weekly meeting based on the same.

Review / progress monitoring committee may consider following points for assessment based on project as mentioned in general guidelines

1	Students' group shall complete project in all aspects including, a. Identification of need/problem b. Proposed final solution c. Procurement of components/system d. Building prototype and testing
2	Continuous assessment will be weekly based on logbook. Two presentations will be conducted for review before a panel. First shall be for finalization of problem and proposed solution

Distribution of Term work marks for both semesters shall be as below:	**Practical Marks**
Marks awarded by guide/supervisor based on implementation	10
Peer assessment by team members	5
Marks awarded by review committee	5
Quality of Project report	5

Course Code	Course Name	Credits
XS37	Skill Enhancement - SAT VII: Skill Based Learning Value Education : Object Oriented Programming with JAVA	0-1-0

Prerequisite:	Structured Programming Approach
Skill Objectives:	1. To learn the basic concepts of object-oriented programming 2. To study JAVA programming language 3. To study various concepts of JAVA programming like multithreading, exception Handling, packages, etc. 4. To explain components of GUI based programming.
Skill Outcomes (SOs):	**At the end of the course, the student will be able to** 1. Apply fundamental programming constructs. 2. Implement the concept of classes and objects, inheritance and interfaces. 3. Implement the concept of strings, arrays, vectors and packages 4. Implement the concept of exception handling and multithreading. 5. Develop GUI based application. 6. Apply ethical principles like timeliness and adhere to the rules of the laboratory

Lab No.	Experiment Title	SO mapped	Hrs /Lab
1	**Title:** Write a program to implement basic programming constructs like branching and looping. **Concepts:** Introduction to Java, Object Oriented Concepts, Java Virtual Machine, Basic programming constructs: variables, data types, and operators, expressions, branching and looping.	**SO1, SO6**	**02**
2	Write a program to demonstrate different ways of accepting user input in Java. **Concepts:** Class, object, data members, member functions, Command Line Argument, Input and output functions in Java, Buffered reader class, Scanner class.	**SO1, SO6**	**02**
3	Write a program to implement the concept of 1. Method overloading 2. Constructor overloading. **Concepts:** Method, how to pass parameters, Method overloading, static members and functions, Introduction to Constructors, Constructor types, Constructor overloading.	**SO2, SO6**	**02**

4	Write a program implement the concept of 2D array and String Manipulation functions in Java. **Concepts:** Array, Strings, String Buffer	**SO3, SO6**	**02**
5	Write a program to implement the concept of Inheritance. **Concepts:** Inheritance, Types of inheritance, extends keyword , super keyword, Access Modifiers	**SO2, SO6**	**02**
6	Write a program to implement the concept of Method Overriding. **Concepts:** Inheritance, Method Overriding.	**SO2, SO6**	**02**
7	Write a program to implement the concept of abstract class and abstract method. **Concepts:** Abstract class and abstract method	**SO2, SO6**	**02**
8	Write a program to implement the concept of package. **Concepts:** Introduction to Packages, Types of Packages-Built-in packages, User defined packages	**SO3, SO6**	**02**
9	Write a program to implement the concept of Exception handling **Concepts:** Exception handling using try, catch, finally, throw and throws, Multiple try and catch blocks, User Defined Exceptions	**SO4, SO6**	**02**
10	Write a program to implement the concept of Multithreading **Concepts:** Introduction to Multithreading, Thread lifecycle, thread class methods, creating threads using extends and implements keyword.	**SO4, SO6**	**02**
11	Design form for Admission process management application system using AWT or Java Swing **Concepts:** Applet and applet life cycle, creating applets, graphics class functions, parameter passing to applet, Font and color class. Event handling using event class AWT: working with windows, using AWT controls for GUI design Swing class in JAVA.	**SO5, SO6**	**02**
12	Study and Implement the concept of JDBC and Perform CRUD Operation on the form created in 11 using Java Database Connectivity **Concepts:** Introduction to JDBC, JDBC-ODBC connectivity, JDBC architecture.	**SO5, SO6**	**02**

Textbooks

1. Herbert Schildt, 'JAVA: The Complete Reference', Ninth Edition, Oracle Press.
2. E. Balagurusamy, 'Programming with Java', McGraw Hill Education.

Reference Books

1. “JAVA Programming‟, Black Book, Dreamtech Press
2. Dietaland Dietal, “Java: How to Program‟, 8th Edition, PHI
3. Ivor Horton, “Beginning JAVA‟, Wiley India
4. “Learn to Master Java programming‟, Staredu Solutions

Useful Links:

1. www.nptelvideos.in
2. www.w3schools.com
3. www.tutorialspoint.com
4. https://starcertification.org/Certifications/Certificate/securejava

SEM IV: TEACHING SCHEME

Course Code	Course Name	Teaching Scheme (Contact Hours)		Credits Assigned		Course Category
		TH – P – TUT	Total	TH – P – TUT	Total	
C401	Applications of Mathematics in Engineering-II	3 – 0 – 1	04	3 – 0 – 1	04	BS
C402	Analysis of Algorithms	3 – 0 – 0	03	3 – 0 – 0	03	PC
C403	Operating System	3 – 0 – 0	03	3 – 0 – 0	03	PC
C404	Theory of Computer Science	3 – 0 – 0	03	3 – 0 – 0	03	PC
C405	Multidisciplinary Minor Course	3 – 0 – 0	03	3 – 0 – 0	03	MM
L402	Analysis of Algorithms Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
L403	Operating System Lab	0 – 2 – 0	02	0 – 1 – 0	01	PC
L404	Multidisciplinary Minor Lab.	0 – 2 – 0	02	0 – 1 – 0	01	MM
PR42	Community Engagement PBL – Mini Project II	0 – 2 – 0	02[$]	0 – 1 – 0	01	PBL
XS48	Skill Enhancement – SAT VIII: Skill-Based Learning: Python Programming	0 – 2[*] – 0	02	0 – 1 – 0	01	SE-SAT
XS49	Ability Enhancement – SAT IX: Skill-Based Learning: Indian/Foreign Modern language	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	
C401	Applications of Mathematics in Engineering-II	20	20	40	60	2.5	25	-	-	-	125
C402	Analysis of Algorithms	20	20	40	60	2.5	-	-	-	-	100
C403	Operating System	20	20	40	60	2.5	-	-	-	-	100
C404	Theory of Computer Science	20	20	40	60	2.5	-	-	-	-	100
C405	Multidisciplinary Minor Course	-	-	-	-	-	50	50	-	-	100
L402	Analysis of Algorithms Lab	-	-	-	-	-	25	-	-	25	50
L403	Operating System Lab	-	-	-	-	-	25	-	25	-	50
L404	Multidisciplinary Minor Lab.	-	-	-	-	-	25	-	-	-	25
PR42	Community Engagement PBL – Mini Project II	-	-	-	-	-	25	-	-	25	50
XS48	Skill Enhancement – SAT VIII: Skill-Based Learning: Python Programming	-	-	-	-	-	25	-	-	-	25
XS49	Ability Enhancement – SAT IX: Skill-Based Learning: Indian/Foreign Modern language	-	-	-	-	-	25	-	-	-	25
Total		**80**	**80**	**160**	**240**	-	**225**	**50**	**25**	**50**	**750**

Course Code	Course Name	Credits Assigned			
		TH	P	TUT	Total
C401	Applications of Mathematics in Engineering-II	03	-	01	04

Prerequisites:	1. Engineering Mathematics-I 2. Engineering Mathematics-II 3. Applications of Mathematics in Engineering-I
Course Objectives (COBs):	1. Matrix algebra to understand engineering problems. 2. Line and Contour integrals and expansion of a complex valued function in a power series. 3. To understand the concepts of vector spaces used in the field of machine learning and engineering problems. 4. The concepts of probability distributions and sampling theory for small samples. 5. Linear and Non-linear programming problems of optimization.
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. & Name	Sub-Topics	CO Mapped	Hrs / Sub Topics	Total Hrs/ module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	02	02
1. Linear Algebra (Theory of Matrices)	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.		02	
	Similarity of Matrices, Diagonalizable and Non-Diagonalizable Matrices.		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		**02**	
	Singular Value Decomposition		**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)		**02**	
	Covariance and Correlations (Kendall rank correlation).		**02**	
4. Probability Distribution	Discrete Probability Distribution: Binomial distribution, Poisson distribution	**CO4**	**02**	**07**
	Continuous Probability Distribution: Normal Distribution, Exponential distributions, Webill distribution		**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.		**03**	
	ANOVA test		**02**	
6.Linear & Non Linear Programming Problems	Types of Solutions, Standard and Canonical of LPP, Basic and Feasible solutions, Slack Variables, Surplus Variables, Simplex Method.	**CO6**	**03**	**06**
	Unconstrained & constrained NLPP using Method of Lagrange's Multiplier (with one-equality constraint with two and three variables)		**03**	
ii.Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	**-**	**01**	**01**
Total Hours				**42**
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.			

<table>
<tr><td>Reference Books:</td><td>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.
5. B. L. Agarwal, Basic Statistics, New Age International publishers.
6. H. K. Dass, Advance Engineering Mathematics, S. Chand and Company ltd</td></tr>
<tr><td>Useful Links:</td><td>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</td></tr>
<tr><td>Term Work (TW):</td><td>1. Term work should consist of 6 batch wise tutorials.
2. Journal must include at least 2 assignments on content of theory of the course.
The distribution of term work marks will be as follows
<table><tr><td>1</td><td>Tutorials</td><td>20</td></tr><tr><td>2</td><td>Assignment</td><td>05</td></tr></table></td></tr>
<tr><td colspan="2">Assessment:</td></tr>
<tr><td colspan="2">Continuous Assessment for 40 marks:
1. Test 1 – 20 marks
2. Test 2 – 20 marks</td></tr>
<tr><td colspan="2">End Semester Theory Examination will be of 60 marks for 02 hrs 30 min duration</td></tr>
</table>

Course Code	Course Name	Credits (TH+P+TUT)
C402	Analysis of Algorithms	3 - 0 - 0

Prerequisite:	1. Data structure concepts 2. Discrete structures
Course Objectives:	1. To provide mathematical approaches for Analysis of Algorithms 2. To understand and solve problems using various algorithmic approaches 3. To analyze algorithms using various methods
Course Outcomes:	**At the end of the course, the students should be able to..** 1. Analyze the running time and space complexity of algorithms 2. Describe, apply and analyze the complexity of divide and conquer strategy. 3. Describe, apply and analyze the complexity of greedy strategy. 4. Describe, apply and analyze the complexity of dynamic programming strategy. 5. Explain and apply backtracking, branch and bound. 6. Explain and apply string matching techniques.

Module No. & Name	Sub-Topics	CO mapped	Hrs / Sub Topics	Total Hrs/Module
i. Prerequisites and Course Outline	Prerequisite Concepts and Course Introduction.	-	01	01
1.Introduction	Performance analysis, space and time complexity, Growth of function, Big- Oh, Omega Theta notation. Mathematical background for algorithm analysis.	CO1	03	09
	Complexity class: Definition of P, NP, NP-Hard, NP-Complete		01	
	Analysis of selection sort, insertion sort		02	
	Recurrences: The substitution method, Recursion tree method, Master method		03	
2. Divide and Conquer Approach	General method, Merge sort, Quick sort, Finding minimum and maximum algorithms and their Analysis, Analysis of Binary search.	CO2	05	05
3. Greedy Method Approach	General Method, Single source shortest path: Dijkstra Algorithm Fractional Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees: Kruskal and Prim's algorithms	CO3	06	06
4. Dynamic Programming Approach	General Method, Multistage graphs, Single source shortest path: Bellman Ford Algorithm All pair shortest path: Floyd Warshall Algorithm	CO4	05	10

	Assembly-line scheduling Problem, 0/1 knapsack Problem, Travelling Salesperson problem, Longest common subsequence		**05**	
5.Backtracking and Branch and bound	General Method, Backtracking: N-queen problem, Sum of subsets, Graph coloring	**CO5**	**03**	**06**
	Branch and Bound: Travelling Salesperson Problem, 15 Puzzle problem		**03**	
6. String Matching Algorithms	The Naïve string-matching algorithm, The Rabin Karp algorithm, The Knuth-Morris-Pratt algorithm	**CO6**	**04**	**04**
ii. Course Conclusion	Recap of Modules, Outcomes, Application and Summarization.	**-**	**01**	**01**
Total Hours				**42**

Books:

Text Books	1. T. H. Cormen, C.E. Leiserson,R.L. Rivest, and C. Stein, "Introduction to algorithms", 2nd Edition, PHI Publication 2005. 2. Ellis Horowitz, Sartaj Sahni, S. Rajsekaran. "Fundamentals of computer algorithms" University Press.
Reference Books	1. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw- Hill Edition. **2.** S. K. Basu, "Design Methods and Analysis of Algorithm", PHI**.**

Useful Links:

1. https://nptel.ac.in/courses/106/106/106106131/
2. https://swayam.gov.in/nd1_noc19_cs47/preview
3. https://www.coursera.org/specializations/algorithms
4. https://www.mooc-list.com/tags/algorithms

Assessment:

Continuous Assessment for 40 marks:

1. Test 1 – 20 marks
2. Test 2 – 20 marks

End Semester Theory Examination will be of 60 marks for 02 hrs 30 min duration

Course Code	Course Name	Credits (TH+P+TUT)
C403	Operating Systems	3-0-0

Prerequisite:	1. Data structures 2. Computer architecture
Course Objectives:	1. To introduce basic concepts and functions of operating systems. 2. To understand the concept of process, thread and resource management. 3. To understand the concepts of process synchronization and deadlock. 4. To understand various Memory, I/O and File management techniques.
Course Outcomes:	**After the successful completion of this course, learner will be able to:** 1. Describe the objectives, functions and structure of OS 2. Analyze the concept of process management and evaluate performance of process scheduling algorithms. 3. Apply the concepts of synchronization and deadlocks 4. Evaluate performance of Memory allocation and replacement policies 5. Explain the concepts of file management. 6. Apply concepts of I/O management and analyze techniques of disk scheduling.

Module No & Name	Sub-Topics	CO mapped	Hrs /Sub Topics	Total Hrs/ Module
i. Prerequisite	Prerequisites concepts and course introduction	--	**01**	**01**
1. Operating system Overview	Introduction, Objectives, Functions and Evolution of Operating System	**CO1**	**02**	**05**
	Operating system structures: Layered, Monolithic and Microkernel		**01**	
	Linux Kernel, Shell and Shell Programming, System Calls		**02**	
2. Process and Process Scheduling	Concept of a Process, Process States, Process Description, Process Control Block.	**CO2**	**02**	**09**
	Uniprocessor Scheduling-Types: Preemptiveand Non-preemptive , scheduling algorithms (FCFS, SJF, SRTN, Priority, RR)		**04**	
	Threads: Definition and Types, Concept of Multithreading		**03**	
3. Process Synchronizat - ion and Deadlocks	Concurrency: Principles of Concurrency,Inter- Process Communication, Process Synchronization	**CO3**	**02**	**09**
	Mutual Exclusion: Requirements Hardware Support (TSL), Operating System Support (Semaphores), Producer and Consumer problem		**03**	

	Principles of Deadlock: Conditions and Resource, Allocation Graphs, Deadlock Prevention, Deadlock Avoidance: Banker's Algorithm		**02**	
	Deadlock Detection and Recovery, Dining Philosophers Problem		**02**	
4. Memory Management	Memory Management Requirements, Memory Partitioning: Fixed, Partitioning, Dynamic Partitioning	**CO4**	**02**	**09**
	Memory Allocation Strategies: Best-Fit, First Fit, Worst Fit		**02**	
	Paging and Segmentation, TLB		**02**	
	Virtual Memory: Demand Paging, Page Replacement Strategies: FIFO, Optimal, LRU, Thrashing		**03**	
5. File Management	Overview, File Organization and Access	**CO5**	**02**	**04**
	File Directories		**01**	
	File Sharing		**01**	
6. IO Management	I/O devices, Organization of the I/O Function, Disk Organization	**CO6**	**01**	**04**
	I/O management		**01**	
	Disk Scheduling: FCFS, SSTF, SCAN, CSCAN, LOOK, C-LOOK		**02**	
ii. Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	**--**	**01**	**01**
Total Hours				**42**

Books:

Text Books	**1.** William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition, 2014, ISBN-10: 0133805913, ISBN-13: 9780133805918. 2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, John Wiley &Sons, Inc., 9th Edition, 2016, ISBN 978-81-265-5427-0
Reference Books	1. Andrew Tannenbaum, Operating System Design and Implementation, Pearson, 3rd Edition. 2. Achyut Godbole and Atul Kahate, Operating Systems, McGraw Hill Education, 3rd Edition 3. Maurice J. Bach, "Design of UNIX Operating System", PHI 4. Sumitabha Das, "UNIX: Concepts and Applications", McGraw Hill, 4th Edition

Useful Links:

1. https://swayam.gov.in/nd1_noc19_cs50/preview
2. https://nptel.ac.in/courses/117/106/117106113/
3. https://nptel.ac.in/courses/117/106/117106113/

4. https:/www.classcentral.com/course/swayam-introduction-to-operating-systems-6559

5. http://vlabs.iitb.ac.in/vlabs-dev/vlab_bootcamp/bootcamp/CRUX/labs/exp1/theory.html

Assessment:

Continuous Assessment for 40 marks:

1. Test 1 – 20 marks
2. Test 2 – 20 marks

End Semester Theory Examination will be of 60 marks for 02 hrs 30 min duration.

Course Code	Course Name	Credits (TH+P+TUT)
C404	**Theory of Computer Science**	**3- 0 – 0**
Prerequisite:	Discrete Structure	
Course Objectives:	1. Acquire conceptual understanding of fundamentals of grammars and languages. 2. Build concepts of theoretical design of deterministic and non-deterministic finite automata and push down automata. 3. Develop understanding of different types of Turing machines and applications. 4. To develop the knowledge and skills necessary to apply these models to solve real world problems.	
Course Outcomes:	**After the successful completion of this course, learner will be able to:** 1. Describe concepts of Theoretical Computer Science, difference and equivalence of DFA and NFA. 2. Discuss key notions of regular expression and pumping lemma for regular language. 3. Design Context free and regular grammar to recognize the language. 4. Solve problems on push down Automata. 5. Develop an understanding of computation through Turing Machine. 6. Acquire fundamental understanding of decidability and undecidability.	

Module No. & Name	Sub Topics	CO mapped	Hrs / Sub Topics	Total Hrs/ Module
i. Prerequisites and Course outline	Prerequisite Concepts and Course Introduction	**-**	**01**	**01**
1. Basic Concepts of a Finite Automata	Importance of TCS, Alphabets, Strings, Languages, Closure properties, Finite Automata (FA) and Finite State machine (FSM).	**CO1**	**03**	**09**
	Deterministic Finite Automata (DFA) and Nondeterministic Finite Automata (NFA): Definitions, transition diagrams and Language recognizers, Equivalence between NFA with and without ε- transitions, NFA to DFA Conversion, Minimization of DFA, FSM with output: Moore and Mealy machines, Applications and limitations of FA.		**06**	
2. Regular Expressions and Languages	Regular Expression (RE), Equivalence of RE and FA, Arden's Theorem, RE Applications	**CO2**	**03**	**07**
	Regular Language (RL), Closure properties of RLs, Decision properties of RLs, Pumping lemma for RLs		**04**	
3. Grammar	Grammars and Chomsky hierarchy	**CO3**	**02**	**08**
	Regular Grammar (RG), Equivalence of Left and Right linear grammar, Equivalence of RG and FA		**03**	
	Context Free Grammars (CFG) Definition, Sentential forms, Leftmost and Rightmost derivations, Parse tree, Ambiguity, Simplification and Applications, Normal Forms: Chomsky Normal Forms (CNF) and Greibach Normal Forms (GNF), Context Free language (CFL) - Pumping lemma, Closure properties.		**03**	

4. Pushdown Automata (PDA)	Definition, Language of PDA, PDA as generator, decider and acceptor of CFG.	**CO4**	**02**	**04**
	Deterministic PDA, Non- Deterministic PDA, Application of PDA.		**02**	
5. Turing Machine (TM)	Definition, Design of TM as generator, decider and acceptor	**CO5**	**04**	**09**
	Variants of TM: Multitrack, Multitape, Universal TM, Applications, Power and Limitations of TMs.		**05**	
6. Undecidability	Decidability and Undecidability, Recursive and Recursively Enumerable Languages.	**CO6**	**01**	**03**
	Halting Problem, Rice's Theorem, Post Correspondence Problem		**02**	
ii. Course conclusion	Recap of Modules, Outcomes, Applications, and Summarization.	**--**	**01**	**01**
Total Hours				**42**

Books:

Text Books	1. John E. Hopcroft, Rajeev Motwani, Jeffery D. Ullman, "Introduction to Automata Theory, Languages and Computation", 3rd Edition, Pearson Education, 2008. 2. Michael Sipser, "Theory of Computation", 3rd Edition, Cengage learning. 2013. 3. Vivek Kulkarni, "Theory of Computation", Illustrated Edition, Oxford University Press, (12 April 2013) India
Reference Books	1. J. C. Martin, "Introduction to Languages and the Theory of Computation", 4 th Edition, Tata McGraw Hill Publication. 2. Kavi Mahesh, "Theory of Computation: A Problem Solving Approach", Kindle Edition, Wiley-India, 2011.

Useful Links:

1. www.jflap.org
2. https://nptel.ac.in/courses/106/104/106104028/
3. https://nptel.ac.in/courses/106/104/106104148/

Assessment:

Continuous Assessment for 40 marks:

1. Test 1 – 20 marks
2. Test 2 – 20 marks

End Semester Theory Examination will be of 60-Marks for 02 hrs 30 min duration.

Course Code	Course Name	Credits (TH+P+TUT)
C405	**Connecting IoT Gateway using AWS Services**	**3-0-0**

Prerequisite:	Science and Maths till 12th STD or Diploma in Engineering and Fundamentals from earlier semester.
Course Objectives:	The course aims to equip students with a comprehensive understanding of IoT architecture, focusing on the integration of microcontrollers and gateways, communication protocols, cloud services using AWS, and practical applications in various industries.
Course Outcomes:	Upon completion of this course, students will be able to: 1. Design and implement IoT solutions using microcontrollers and gateways. 2. Analyze and apply various IoT communication protocols. 3. Utilize AWS cloud services for IoT data storage, processing, and analytics. 4. Develop and deploy practical IoT applications across different sectors.

Module Number & Name	Sub Topics	CO mapped	Hrs / Sub topics	Total Hrs / Module
1. IoT Gateway and Microcontrollers	Know Your IoT Gateway		**12**	**19**
	Microcontrollers - Arduino		**7**	
2. IoT Communication Protocols	IoT Communication Protocols		**5**	**05**
	Video Resources (Self - Study)		**NA**	
3. IoT Services and Cloud Computing using AWS	Unlocking Power of IoT using AWS		**5**	**25**
	Mobile Browser to IoT Gateway Communication		**5**	
	Cloud Connectivity For IoT Applications		**5**	
	How to Open AWS Account		**5**	
	AWS Cost Management		**5**	
4. Applications	Home Security Solution- Digital Lock			**10**
Course Conclusion	Recap of Modules, Outcomes, Applications, and Summarization.		**1**	**01**
			Total Hours	**60**
TW: 50 Marks **Oral : 50 Marks**				

Lab Code	Lab Name	Credits (P+TUT)
L402	**Analysis of Algorithms Lab**	**0-1-0**

Prerequisite:	Basic knowledge of programming and data structure
Lab Objectives:	1. To introduce the methods of designing and analyzing algorithms 2. Design and implement efficient algorithms for a specified application 3. Strengthen the ability to identify and apply the suitable algorithm for the given real-world problem. 4. Analyze worst-case running time of algorithms and understand fundamental algorithmic problems.
Lab Outcomes (LOs):	**At the end of the course, the student will be able to** 1. Implement the algorithms using different approaches 2. Analyze the complexities of various algorithms 3. Compare the complexity of the algorithms for specific problems 4. Apply ethical principles like timeliness and adhere to the rules of the laboratory

Lab No.	Experiment Title	LO mapped	Hrs/Lab
0	Lab Prerequisites	**-**	**02**
1	**Introduction:(Implement any 2)** Selection sort, Insertion sort	**LO1, LO2, LO3, LO4**	**04**
2	**Divide and Conquer Approach :(Implement any 2)** Finding Minimum and Maximum, Merge sort, Quick sort, Binary search	**LO1, LO2, LO3, LO4**	**04**
3	**Greedy Method Approach :(Implement any 2)** Single source shortest path- Dijkstra Fractional Knapsack problem Job sequencing with deadlines Minimum cost spanning trees-Kruskal and Prim's algorithm	**LO1, LO3, LO4**	**04**
4	**Dynamic Programming Approach:(Implement any 2)** Single source shortest path- Bellman Ford All pair shortest path- Floyd Warshall , 0/1 knapsack, Travelling salesperson problem Longest common subsequence	**LO1, LO4**	**04**
5	**Backtracking and Branch and bound:(Implement any 2)** N-queen problem Sum of subsets Graph coloring	**LO1, LO4**	**04**
6	**String Matching Algorithms:(Implement any 2)** The Naïve string-matching Algorithms The Rabin Karp algorithmThe Knuth-Morris-Pratt algorithm	**LO1, LO4**	**06**

Text Books	1. T. H. Cormen, C.E. Leiserson,R.L. Rivest, and C. Stein, "Introduction to algorithms", 2nd Edition, PHI Publication 2005. 2. Ellis Horowitz, Sartaj Sahni, S. Rajsekaran. "Fundamentals of computer algorithms" University Press.
Reference Books	1. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw- Hill Edition. **2.** S. K. Basu, "Design Methods and Analysis of Algorithm", PHI**.**

Useful Links:

1. https://nptel.ac.in/courses/106/106/106106131/
2. https://swayam.gov.in/nd1_noc19_cs47/preview
3. https://www.coursera.org/specializations/algorithms
4. https://www.mooc-list.com/tags/algorithms

Term work:

1. Term work should consist of at least 10 experiments
2. Journal must include at least 2 assignments on content of theory and practical of the course "Analysis of Algorithms"
3. The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work.
4. Total 25 Marks (Experiments: 20-marks, Assignments: 05-marks)

Oral & Practical Exam:

Oral & practical examination will be based on entire syllabus of CEC402 and CEL402

Lab Code	Lab Name	Credits (P+TUT)
L403	**Operating Systems Lab**	**0-1-0**

Prerequisite:	1. Computer Organization 2. Data Structures and Algorithms
Lab Objectives:	1. To gain practical experience with designing and implementing concepts of operating systems such as system calls, CPU scheduling, process management, memory management, file systems and deadlock handling using C language in Linux environment. 2. To familiarize students with the architecture of Linux OS. 3. To provide necessary skills for developing and debugging programs in Linux environment. 4. To learn programmatically to implement simple operation system mechanisms
Lab Outcomes (LOs):	**At the end of the course, the student will be able to:** 1. Demonstrate basic Operating system Commands, Shell scripts, System Calls and API with respect to Linux 2. Implement various process scheduling algorithms and evaluate their performance. 3. Implement and analyze concepts of synchronization and deadlocks. 4. Implement various Memory Management techniques and evaluate their performance. 5. Implement and analyze concepts of virtual memory, concepts of file management and I/O management techniques. 6. Apply ethical principles like timeliness and adhere to rules of laboratory.

Lab No.	Experiment Title	LO mapped	Hrs/ Lab
0	Prerequisite	**-**	**02**
1	**<u>Explore Linux Commands</u>** **Explore usage of basic Linux Commands and system calls for file, directory and process management.** Commands: mkdir, chdir, cat, ls, chown, chmod, chgrp, ps etc. System Calls: open, read, write, close, getpid, setpid, getuid, getgid, getegid, geteuid. sort, grep, awk, etc.	**LO1, LO6**	**02**
2	**<u>Linux shell script</u>** Write shell scripts to do the following: a. Display OS version, release number, kernel version b. Display top 10 processes in descending order c. Display processes with highest memory usage. d. Display current logged in user and log name. e. Display current shell, home directory, operating system type, current path setting, current working directory	**LO1, LO6**	**02**

3	**Linux- API** Implement any one basic commands of Linux like ls, cp, mv and others using kernel APIs.	**LO1, LO6**	**02**
4	**Linux- Process** a. Create a child process in Linux using the fork system call. From the child process obtain the process ID of both child and parent by using getpid and getppid system call. b. Explore wait and waitpid before termination of process.	**LO2, LO6**	**02**
5	**Process Management: Scheduling** a. Write a program to demonstrate the concept of non-preemptive scheduling algorithms. b. Write a program to demonstrate the concept of preemptive scheduling algorithms	**LO2, LO6**	**02**
6	**Process Management: Synchronization** a. Write a C program to implement solution of Producer consumer problem through Semaphore b. Write a C program to implement solution of Reader's Writer's problem through Semaphore	**LO3, LO6**	**02**
7	**Process Management: Deadlock** a. Write a program to demonstrate the concept of deadlock avoidance through Banker's Algorithm b. Write a program demonstrate the concept of Dining Philosopher's Problem c. Simulate deadlock detection using CPU-OS Simulator	**LO3, LO6**	**02**
8	**Memory Management** a. Write a program to demonstrate the concept of MVT and MFT memory management techniques b. Write a program to demonstrate the concept of dynamic partitioning placement algorithms i.e., Best Fit, First Fit, Worst- Fit etc.	**LO4, LO6**	**02**
9	**Memory Management: Virtual Memory** a. Write a program to demonstrate the concept of demand paging for simulation of Virtual Memory implementation b. Write a program in C demonstrate the concept of page replacement policies for handling page faults eg: FIFO, LRU etc.	**LO5, LO6**	**02**
10	**File Management & I/O Management** a. Write a C program to simulate File allocation strategies typically sequential, indexed and linked files b. Write a C program to simulate file organization of multi-level directory structure. c. Write a program in C to do disk scheduling - FCFS, SCAN, C-SCAN	**LO5, LO6**	**02**
Virtual Lab Links:			
1. http://vlabs.iitb.ac.in/vlabs-dev/vlab_bootcamp/bootcamp/CRUX/labs/exp1/theory.html			

Term work:

1. Term work should consist of a minimum of 10 experiments covering all modules.
2. Journal must include at least 2 assignments on content of theory and practical of the course "Operating Systems"
3. The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work.
4. Total 25 Marks (Experiments: 20-marks, Assignments: 05-marks)

Practical & Oral Exam:

Practical Exam will be conducted based on the entire syllabus of C403 and L403

Course code	Course Name	Credits
PR42	**Community Engagement Project / Field Project –Mini Project-II**	**0-2-0**

Objectives:	1. To acquaint yourself with the process of identifying the needs and converting it into the problem. 2. To familiarize the process of solving the problem in a group. 3. To acquaint yourself with the process of applying basic engineering fundamentals to attempt solutions to the problems. 4. To inculcate the process of self-learning and research.
Outcome:	**After successful completion of this course learner will be able to…** 1. Identify problems based on societal /research needs. 2. Design solutions or system components or processes that meet the specified needs 3. Select appropriate tools to implement the project. 4. Develop interpersonal skills to work as a member of a group or leader 5. Excel in written and oral communication. 6. Demonstrate project management principles during project work. 7. Demonstrate capabilities of investigation and self-learning by oneself or as a team gaining life skills

Guidelines for Mini Project

1	Project based learning Mini Project Lab-1 should be implemented preferably using Python programming (CEXS45)
2	Students shall form a group of 2 to 3 students, while forming a group shall not be allowed less than two or more than three students, as it is a group activity.
3	Students should do survey and identify needs, which shall be converted into problem statements for mini project in consultation with faculty supervisor/internal committee of faculties.
4	Students shall submit implementation plan in the form of Gantt/PERT/CPM chart, which will cover weekly activity of mini project.
5	A logbook to be prepared by each group, wherein group can record weekly work progress, guide/supervisor can verify and record notes/comments.
6	Faculty supervisor may give inputs to students during mini project activity; however, focus shall be on self-learning.
7	Students in a group shall understand problem effectively, propose multiple solution and select best possible solution in consultation with guide/ supervisor.
8	Students shall convert the best solution into working model using Python Programming.

9	The solution to be validated with proper justification and report to be compiled in standard format of the college.
10	With the focus on the self-learning, innovation, addressing societal problems and entrepreneurship quality development within the students through the Mini Projects, it is preferable that a single project of appropriate level and quality to be carried out in two semesters by all the groups of the students. i.e. Mini Project 1 in semester III and IV.
11	However, based on the individual students or group capability, with the mentor's recommendations, if the proposed Mini Project adhering to the qualitative aspects mentioned above gets completed in odd semester, then that group can be allowed to work on the extension of the Mini Project with suitable improvements/modifications or a completely new project idea in even semester. This policy can be adopted on case by case basis.

Term Work

The review/ progress monitoring committee shall be constituted by senior faculty members. The progress of mini project to be evaluated on continuous basis, minimum two reviews in each semester. Assessment also considers peer review and ethics observed by faculties and participation involvement.

Continuous Assessment

In continuous assessment focus shall also be on each individual student, log book maintained and weekly meeting based on the same.

	Distribution of Term work marks for both semesters shall be as below:	**Practical Marks**
1	Marks awarded by guide based on implementation	10
2	Peer assessment by team members	05
3	Marks awarded by review committee for presentation	05
4	Quality of Project report	05

Review / progress monitoring committee may consider following points for assessment based on project as mentioned in general guidelines

Project:

1	In this case in one semester students' group shall complete project in all aspects including, a. Identification of need/problem b. Proposed final solution c. Procurement of components/systems d. Building prototype and testing
2	Continuous assessment will be weekly based on logbook. Two presentations will be conducted for review before a panel. a. First shall be for finalization of problem and proposed solution b. Second shall be for implementation and testing of solution.

Assessment criteria of Mini Project.

Mini Project shall be assessed based on following criteria:

1	Quality of survey and identification of problem statement

2	Innovativeness in solutions
3	Implementation
4	Team work
5	Project report
Guidelines for Assessment of Mini Project Practical/Oral Examination:	
1	Report should be prepared as per the guidelines issued by the University of Mumbai.
2	Mini Project shall be assessed through a presentation and demonstration of working model by the student project group to a panel of Internal and External Examiners preferably from industry or research organizations having experience of more than five years approved by head of Institution.
3	Students shall be motivated to publish a paper based on the work in Conferences/students competitions.
Mini Project shall be assessed based on following points:	
1	Quality of problem and Clarity
2	Innovativeness in solutions
3	Cost effectiveness and Societal impact
4	Full functioning of working model as per stated requirements
5	Effective use of skill sets
6	Effective use of standard engineering norms
7	Contribution of an individual's as member or leader
8	Clarity in written and oral communication

Total Marks = Term work +Oral & Practical = (25+25)

25 marks of Term work will be given on the basis of evaluation of project practical marks and Log book which is filled weekly by students as per their weekly progress.

25 marks of Oral and practical will be based on a project implementation.

Course Code	Course Name	Credits (TH+P+TUT)
XS48	Skill Based learning: Python Programming (SAT-V)	0 + 1 + 0

Prerequisite:	Knowledge of programming language like C and Java
Skill Objectives:	1. Basics of Python programming 2. Decision Making, Data structure and Functions in Python 3. Object Oriented Programming using Python 4. Web framework for developing
Skill Outcomes:	**After successful completion of this course learner will be able to…** 1. To understand basic concepts in python. 2. To explore contents of files, directories and text processing with python 3. To develop program for data structure using built in functions in python. 4. To explore django web framework for developing python-based web application and basics of NumPy and Pandas 5. To understand Multithreading concepts using python. 6. Apply ethical principles like timeliness and adhere to the rules of the laboratory.

Module	Sub Topics	SO mapped	Hrs / Sub topics	Total Hrs / Module
i. Prerequisitesand Course Outline	**Introduction to python, Features, Applications, Comparison with C and Java**			**02**
1. Python basics	Data types in python, Operators in python, Input and Output	**SO1, SO6**	**01**	**04**
	Control statement, Arrays in python		**01**	
	String and Character in python, Functions, List and Tuples, Dictionaries Exception		**01**	
	Introduction to OOP, Classes, Objects, Interfaces, Inheritance		**01**	
2. Advanced Python	Files in Python, Directories	**SO2, SO6**	**01**	**04**
	Building Modules		**01**	
	Packages, Text Processing		**01**	
	Regular expression in python		**01**	
3. Data Structure inPython	Link List, Stack	**SO3, SO6**	**02**	**04**
	Queues, Dequeues		**02**	
4. Python Integration Primer	Graphical User interface, Networking in Python	**SO4, SO6**	**01**	**04**
	Python database connectivity		**01**	
	Introduction to Django		**02**	
5.Multithreading	Thread and Process, Starting a thread	**SO5, SO6**	**01**	**04**
	Threading module, Synchronizing threads		**02**	
	Multithreaded Priority Queue		**01**	

6. NumPy and Pandas	Creating NumPy arrays, Indexing and slicing in NumPy, creating multidimensional arrays, NumPy Data types	**SO4, SO6**	**02**	**06**
	Array Attribute, Indexing and Slicing, Creating array views copies, Manipulating array shapes I/O		**02**	
	Basics of Pandas, Using multilevel series, Series andData Frames, Grouping, aggregating, Merge DataFrames		**02**	
Total Hours				**28**

Books:	
Text Books	1. Dr. R. Nageswara Rao, "Core Python Programming", Dreamtech Press 2. Beginning Python: Using Python 2.6 and Python 3.1. James Payne, Wrox Publication 3. Anurag Gupta, G. P. Biswas, "Python Programming", McGraw-Hill 4. E. Balagurusamy, "Introduction to computing and problem-solving using python", McGraw Hill Education
Reference Books	1. Zed A. Shaw, "Learn Python 3 the Hard Way", Zed Shaw's Hard Way Series 2. Martin C. Brown," Python: The Complete Reference", McGraw-HillPublication. 3. Laura Cassell, Alan Gauld, "Python Projects", Wrox Publication
Useful Links:	1. "The Python Tutorial", http://docs.python.org/release/3.0.1/tutorial/ 2. Beginning Perl, https://www.perl.org/books/beginning-perl/ 3. http://spoken-tutorial.org 4. https://starcertification.org/Certifications/Certificate/python

Suggested experiments using Python:

Sr. No.	Title of Experiments
1	Exploring basics of python like data types (strings, list, array, dictionaries, set, tuples) and control statements
2	Creating functions, classes and objects using python. Demonstrate exception handling and inheritance.
3	Exploring Files and directories a. Python program to append data to existing file and then display the entire file b. Python program to count number of lines, words and characters in a file. c. Python program to display file available in current directory
4	Creating GUI with python containing widgets such as labels, textbox, radio, checkboxes and custom dialog boxes.
5	Menu driven program for data structure using built in function for link list, stack and queue.
6	Program to demonstrate CRUD (**create, read, update and delete)** operations on database (SQLite/ MySQL) using python.
7	Creation of simple socket for basic information exchange between server and client.
8	Creating web application using Django web framework to demonstrate functionality of user login and registration (also validating user detail using regular expression).

9	Programs on Threading using python.
10	Exploring basics of NumPy Methods.
11	Program to demonstrate use of NumPy: Array objects.
12	Program to demonstrate Data Series and Data Frames using Pandas.
13	Program to send email and read content of URL.

Term Work for 25 Marks:

Programming labs to be conducted as 2 hrs continuous (theory + hands-on) session.The assessment will be

- An online quiz conducted at the end of every 2-hr session consisting of 5 questions for a totalof 10 marks. The average of best 10 quizzes will be considered toward 10 marks.

- Students should perform minimum 10 experiments. The programs performed along with the screenshot of output have to be submitted within two days. A cover page will be attached stating the aims and objectives. This will be considered towards 10 marks.
- Attendance= 05 marks

	Spoken Tutorial Test	**Lab Submission**	**Total**
Marks Allotted	10	15	25

Course Code	Exposure Course Name	Credits			
		TH	P	TUT	Total
XA410	**Ability Enhancement – SAT X: Skill Based Learning (Indian/Foreign Modern language)**	-	01	-	01

SBL 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.
SBL Outcome (SOs):	Upon completion of the course, the learners will be able to: 1. Demonstrate 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 suggestive given list of course but are not limited to the list as it's a learner's choice for the interested course in the given semester time frame.

Sr No.	Suggestive list of Courses-
1	Introduction to Japanese Language and Culture
2	German – 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

Sr No	Suggestive Courses Link but are not limited to following resources 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 https://www.coursera.org/learn/spanish-vocabulary-cultural-experience https://www.coursera.org/learn/spanish-vocabulary-sports-travel-home https://www.coursera.org/learn/spanish-vocabulary-careers https://www.coursera.org/learn/spanish-vocabulary-project
5	https://www.coursera.org/learn/korean-beginners https://www.coursera.org/learn/learn-korean https://www.coursera.org/learn/learn-speak-korean1 https://www.coursera.org/learn/the-korean-alphabet-an-introduction-to-hangeul
6	https://www.udemy.com/course/complete-french-course/
7	https://www.udemy.com/course/complete-german-course-learn-german-for-beginners/
8	https://www.udemy.com/course/spanish-101-beginning-spanish-spanish-for-beginners/
9	https://www.udemy.com/course/complete-japanese-course-learn-japanese-for- beginners-lvl-1/
10	https://www.udemy.com/course/complete-korean-course-learn-korean-for-beginners- level-1/
11	https://www.udemy.com/course/the-complete-russian-language-course/
12	https://onlinecourses.nptel.ac.in/noc22_hs114/preview
13	https://onlinecourses.nptel.ac.in/noc22_hs85/preview
14	https://onlinecourses.nptel.ac.in/noc22_hs139/preview