Agnel Charities

Fr. C. Rodrigues Institute of Technology

Sector 9A, Vashi, Navi Mumbai, 400703, Maharashtra, India

www.fcrit.ac.in

An Autonomous Institute Affiliated to the University of Mumbai

Department of Computer Engineering

Curriculum Structure FY to B. Tech.

&

Second Year Syllabus

Prepared by: Board of Studies for Computer Engineering

Approved By: Academic Council of Fr. C. Rodrigues Institute of Technology

Effective from :2025-26

Revision: 2024.1

PREAMBLE

Accelerating Towards Excellence: Unveiling a New Era in Education

Dear Students, Faculty, and Stakeholders,

It is with great pleasure and anticipation that we introduce the newly designed curriculum for autonomy at Agnel Charities' Fr. C. Rodrigues Institute of Technology. This pioneering initiative aims to revolutionize engineering education, ensuring our graduates are equipped with not only technical prowess but also the holistic skills necessary for thriving in today's dynamic professional landscape.

1. **Purpose of Autonomy:** Our commitment to autonomy is rooted in the imperative to bridge the gap between academia and industry. We envision education as a catalyst for individual growth, fostering self-sustainability and enhancing employability. Through our curriculum, we strive to nurture engineers who not only excel in their fields but also contribute meaningfully to society.

2. **Curriculum Design: A Top-to-Down Approach:** Our curriculum is meticulously crafted with a top-to-down approach, encompassing all 12 attributes of Program Outcomes mandated by regulatory bodies. Emphasizing a blend of theoretical knowledge and practical application, it is designed to cultivate well-rounded professionals capable of tackling real-world challenges with confidence and competence.

3. **Alignment with National Education Policy-2020:** In adherence to the guidelines laid out in the National Education Policy-2020, our curriculum embodies a multidisciplinary approach, offering a diverse array of core and elective courses. It integrates hands-on learning experiences such as mini and major projects, skill-based labs, and one-semester internships to nurture innovation and problem-solving skills. Additionally, the inclusion of value-added courses, honours, and minors ensures a comprehensive educational journey tailored to individual interests and aspirations.

4. **Opportunities for Teachers in Innovation:** We recognize the pivotal role of our faculty in shaping the educational experience. Our curriculum provides ample opportunities for teachers to innovate in teaching-learning methodologies and evaluation techniques. Through continuous professional development programs and collaborative platforms, we empower our educators to experiment with innovative pedagogies, leverage technology for enhanced learning outcomes, and implement novel assessment strategies. By fostering a culture of innovation among our faculty, we aim to enrich the learning experience and inspire a passion for lifelong learning among our students.

As we embark on this transformative journey, we invite all stakeholders to join us in shaping the future of engineering education. Together, let us strive towards excellence, innovation, and societal impact.

Sincerely,

Dean of Academics Agnel Charities' Fr. C. Rodrigues Institute of Technology

Preamble from Board of Studies

Dear Students and Stakeholders,

The Department of Computer Engineering at Fr. C. Rodrigues Institute of Technology, Vashi, was established in 1994 and the current intake capacity is 120.One of the notable achievements is accreditation by the National Board of Accreditation (NBA) in 2006, 2012, 2019 and 2021, which reflects our commitment in maintaining high standards of education and infrastructure. Our mission is to provide a dynamic learning environment where students not only acquire technical expertise but also develop critical thinking, problem-solving skills, and leadership qualities. Through rigorous academic activities, hands-on learning opportunities, and industry collaborations, we aim to prepare our graduates to thrive in today's rapidly evolving technological landscape.

It is with great pleasure and anticipation that the members of Board of Studies, Computer Engineering, at Agnel Charities' Fr. C. Rodrigues Institute of Technology are very happy to introduce the newly designed curriculum for autonomy with effect from AY 24-25.

This curriculum is tailored to shape well-rounded individuals by adopting a comprehensive educational approach. It prioritizes building a solid foundation in science, mathematics, and computer engineering, complemented by a diverse selection of elective courses aimed at honing expertise in specialized areas to gain practical experience through extensive industry or research internships. Collaboration with external entities ensures the integration of pertinent skill-building courses, while honors evaluations offer opportunities for advanced learners. Moreover, in alignment with the principles outlined in the National Education Policy-2020, mission and vision of Institute and Department, our curriculum embraces a multidisciplinary approach, integrating subjects of various disciplines to provide a distinct learning experience to students. Furthermore, the curriculum systematically fosters the development of soft skills and enriches social, physical, mental, and spiritual dimensions of personality through thoughtfully curated liberal and experiential learning. Our curriculum adopts a top-to-down approach, meticulously incorporating all 12 attributes of Program Outcomes mandated by regulatory bodies, with reference to NBA SAR-January 2016.

The salient features of the autonomy curriculum are:

1. Implementation of a credit structure that aims to afford students' more time for extracurricular activities, co-curricularactivities, innovation, internships, and research.

2. Program elective courses within the department, open elective courses and honours/minor courses will offer specialized knowledge in the respective domain to students motivating them to remain updated on emerging trends, actively participate in continuousprofessional development.

3.The inclusion of Skill-Based Labs, Mini Projects, Major projects and Internships as a part of the curriculum provides students with a platform to demonstrate their talents through innovative projects, thereby strengthening their profiles and significantly improving their employability prospects in the product, service and consultancy organizations.

We are sure you will find this curriculum interesting, challenging, fulfilling the needs and expectations of Industry, Research and Academics. We are committed to fostering holistic development and ensuring that our students are well-equipped to succeed in their chosen careers. We take immense pride in serving as a beacon of academic excellence and innovation, fueled by our unwavering commitment.

Join us as we embark on this journey of autonomy, academic excellence and innovation in the field of Computer Engineering.

Sincerely,

Chairman, Board of Studies- Computer Engineering, Fr. C. Rodrigues Institute of Technology

Contents

A. Abbreviations

AEC	Ability Enhancement Course
AU	Audit Course
BSC	Basic Science Course including Mathematics
BSL	Basic Science Laboratory Course
ELC	Experiential Learning Course
ESC	Engineering Sciences Course
ESL	Engineering Sciences Laboratory Course
HMC	Honours or Minor Core Course
HML	Honours or Minor Laboratory
HMP	Honours or Minor Mini Project
HSS	Humanities Social Sciences and Management Course
IKS	Indian Knowledge System Course
INT	Internship
L	Lecture
LBC	Laboratory Course
LLC	Liberal Learning Course
MDM	Multidisciplinary Minor Course
MDL	Multidisciplinary Laboratory Course
MJP	Major Project
MNP	Mini Project
OEC	Open Elective Course
P	Practical
PCC	Program Core Course
PEC	Program Elective Course
RPC	Research Project Coursework
RPR	Research Project
SBL	Skill Based Laboratory
SEC	Skill Enhancement Course
T	Tutorial
VEC	Value Education Course

B. Credit Structure

1. B. Tech in Computer Engineering											
Type of Course	**Semester-wise Credit Distribution**									**FCRIT Credit Distribution**	**DTE Credit Distribution**
	I	**II**	**III**	**IV**	**V**	**VI**	**VII**	**VIII**	**Total**		
Basic Science Course (BSC)	08	08	--	--	--	--	--	--	**16**	**18**	**14-18**
Basic Science Laboratory Course (BSL)	01	01	--	--	--	--	--	--	**02**		
Engineering Science Course (ESC)	05	02	--	--	--	--	--	--	**07**	**16**	**12-16**
Engineering Science Laboratory Course (ESL)	04	05	--	--	--	--	--	--	**09**		
Program Core Course (PCC)	--	--	14	13	06	03	03	--	**39**	**50**	**44-56**
Laboratory Course (LBC)	--	--	02	03	02	02	02	--	**11**		
Program Elective (PEC)	--	--	--	--	03	03	06	03	**15**	**15**	**20**
Multidisciplinary Minor (MDM)	--	--	03	03	03	04	--	--	**13**	**13**	**14**
Multidisciplinary Laboratory Course (MDL) †	--	--	--	--	01	--	--	--	**01**	**01**	
Open Elective (OEC)	--	--	--	--	--	--	03	03	**06**	**06**	**08**
Skill Enhancement Course (SEC)	01	01	--	--	--	--	--	--	**02**	**08**	**08**
Skill Based Laboratory (SBL)	--	--	02	02	--	02	--	--	**06**		
Ability Enhancement Course (AEC)	--	03	--	--	02	--	--	--	**05**	**05**	**04**
Humanities Social Sciences and Management (HSS)	--	--	02	--	02	--	02	--	**06**	**06**	**04**
Indian Knowledge System (IKS)	--	02	--	--	--	--	--	--	**02**	**02**	**02**
Value Education Course (VEC)	02	--	--	02	--	--	--	--	**04**	**04**	**04**
Experiential Learning Course (ELC)	--	--	--	--	--	02	--	--	**02**	**02**	**04**
Mini Project (MNP)	--	--	01	01	01	01	--	--	**04**	**10**	**04**
Major Project (MJP)	--	--	--	--	--	--	02	04	**06**		
Internship (INT)	--	--	--	--	--	--	--	08	**08**	**08**	**12**
Liberal Learning Course (LLC)	--	--	--	--	--	02	--	---	**02**	**02**	**04**
Total Credits	**21**	**22**	**24**	**24**	**20**	**19**	**18**	**18**	**166**	**166**	**160-176**

† NOTE: The Multidisciplinary Laboratory Course can be moved to the sixth semester if the department prefers.

C Curriculum Structure and Examination Scheme for B. Tech in Computer Engineering

(FY and SY with Effect from AY 2024-2025)
Curriculum Structure – FY Semester-I

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
BSC101	Engineering Mathematics I	3	--	1	3	--	1	4
BSC102	Engineering Physics-I	2	--	--	2	--	--	2
BSC103	Engineering Chemistry-I	2	--	--	2	--	--	2
ESC101	Engineering Mechanics	3	--	--	3	--	--	3
ESC102	Basic Electrical Engineering	2	--	--	2	--	--	2
BSL101	Engineering Physics-I Laboratory	--	1	--	--	0.5	--	0.5
BSL102	Engineering Chemistry-I Laboratory	--	1	--	--	0.5	--	0.5
ESL101	Engineering Mechanics Laboratory	--	2	--	--	1	--	1
ESL102	Basic Electrical Engineering Laboratory	--	2	--	--	1	--	1
ESL103	Programming Laboratory-I (C)	--	2*+2	--	--	2	--	2
SEC101	Basic Workshop Practice-I	--	2	--	--	1	--	1
VEC101	Universal Human Values	2	--	--	2	--	--	2
	Total	**14**	**12**	**1**	**14**	**6**	**1**	**21**

*** Instructions should be conducted for the entire class.**

NOTE 1: Compulsory Non-Credit Activities: Participation and/or coordination of co-curricular and extra-curricular events at the Institute or Department level is mandatory for all students from semesters 1 to 8 as part of non-credit liberal education. Please consult the department's Curriculum Book for more information. These activities do not yield credits. Upon successful participation or organization of activities, a certificate will be awarded at the conclusion of semester 8.

NOTE 2: Please note that during semesters 1 to 8 some of the non-technical courses such as Humanities Social Sciences and Management (HSSM), Open Electives (OE), Value Education Course (VEC), and Liberal Learning Course (LLC) may be conducted either online synchronously or asynchronously. For more information, please consult the curriculum book of your respective department.

Examination Scheme – FY Semester-I

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
BSC101	Engineering Mathematics-I	20+25@	30	50	1.5	2	125
BSC102	Engineering Physics-I	15	20	40	1.0	1.5	75
BSC103	Engineering Chemistry-I	15	20	40	1.0	1.5	75
ESC101	Engineering Mechanics	20	30	50	1.5	2	100
ESC102	Basic Electrical Engineering	15	20	40	1.0	1.5	75
BSL101	Engineering Physics-I Laboratory	25	--	--	--	--	25
BSL102	Engineering Chemistry-I Laboratory	25	--	--	--	--	25
ESL101	Engineering Mechanics Laboratory	25	--	--	--	--	25
ESL102	Basic Electrical Engineering Laboratory	25	--	25	--	--	50
ESL103	Programming Laboratory-I (C)	50	--	50	--	--	100
SEC101	Basic Workshop Practice-I	50	--	--	--	--	50
VEC101	Universal Human Values	50	--	--	--	--	50
	Total	**360**	**120**	**295**	**--**	**--**	**775**

$Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

@For continuous assessment of tutorials.

Curriculum Structure – FY Semester-II

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
BSC204	Engineering Mathematics-II	3	--	1	3	--	1	4
BSC205	Engineering Physics-II	2	--	--	2	--	--	2
BSC206	Engineering Chemistry-II	2	--	--	2	--	--	2
AEC201	Professional Communication and Ethics-I	2	2	--	2	1	--	3
ESC203	Basic Electronics Engineering	2	--	--	2	--	--	2
BSL203	Engineering Physics-II Laboratory	--	1	--	--	0.5	--	0.5
BSL204	Engineering Chemistry-II Laboratory	--	1	--	--	0.5	--	0.5
ESL204	Engineering Graphics Laboratory	--	2*+2	--	--	2	--	2
ESL205	Programming Laboratory-II (Java)	--	2*+2	--	--	2	--	2
ESL206	Basic Electronics Engineering Laboratory	--	2	--	--	1	--	1
SEC202	Basic Workshop Practice-II	--	2	--	--	1	--	1
IKS201	Indian Knowledge System	2	--	--	2	--	--	2
	Total	**13**	**16**	**1**	**13**	**8**	**1**	**22**

*** Instructions should be conducted for the entire class.**

Examination Scheme – FY Semester-II

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
BSC204	Engineering Mathematics-II	20+25@	30	50	1.5	2	125
BSC205	Engineering Physics-II	15	20	40	1.0	1.5	75
BSC206	Engineering Chemistry-II	15	20	40	1.0	1.5	75
AEC201	Professional Communication and Ethics-I	50	--	--	--	--	50
ESC203	Basic Electronics Engineering	15	20	40	1.0	1.5	75
BSL203	Engineering Physics-II Laboratory	25	--	--	--	--	25
BSL204	Engineering Chemistry-II Laboratory	25	--	--	--	--	25
ESL204	Engineering Graphics Laboratory	50	--	50	--	--	100
ESL205	Programming Laboratory-II (Java)	50	--	50	--	--	100
ESL206	Basic Electronics Engineering Laboratory	25	--	25	--	--	50
SEC202	Basic Workshop Practice-II	50	--	--	--	--	50
IKS201	Indian Knowledge System	50	--	--	--	--	50
	Total	**415**	**90**	**295**	--	--	**800**

$Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

@For continuous assessment of tutorials.

Curriculum Structure – SY Semester-III

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
CEPCC301	Engineering Mathematics-III	3	--	1	3	--	1	4
CEPCC302	Discrete Structures and Graph Theory	3	--	1	3	--	1	4
CEPCC303	Data Structures	3	--	--	3	--	--	3
CEPCC304	Database Management System	3	--	--	3	--	--	3
XXMDM301Y**	----	3	--	--	3	--	--	3
CELBC301	Data Structures Laboratory	--	2	--	--	1	--	1
CELBC302	Database Laboratory	--	2	--	--	1	--	1
CESBL301	Python Programming Laboratory	--	4	--	--	2	--	2
CEMNP301	Mini Project-1A	--	3	--	--	1	--	1
HSS301	Product Design	2	--	--	2	--	--	2
	Total	**17**	**11**	**2**	**17**	**5**	**2**	**24**

**Four Theory courses (Three 3-credit and one 4-credit) and one Laboratory course (1-credit) offered by other department has to be taken by Computer Engineering students, to complete the 14-credit requirement of MDM.

Examination Scheme – SY Semester-III

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
CEPCC301	Engineering Mathematics-III	20+25@	30	50	1.5	2	125
CEPCC302	Discrete Structures and Graph Theory	20+25@	30	50	1.5	2	125
CEPCC303	Data Structures	20	30	50	1.5	2	100
CEPCC304	Database Management System	20	30	50	1.5	2	100
XXMDM301Y	----	20	30	50	1.5	2	100
CELBC301	Data Structures Laboratory	25	--	25	--	--	50
CELBC302	Database Laboratory	25	--	25	--	--	50
CESBL301	Python Programming Laboratory	50	--	50	--	--	100
CEMNP301	Mini Project-1A	50	--	--	--	--	50
HSS301	Product Design	50	--	--	--	--	50
	Total	**350**	**150**	**350**	--	--	**850**

$ Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

@For continuous assessment of tutorials.

Curriculum Structure – SY Semester-IV

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
CEPCC405	Engineering Mathematics-IV	3	--	1	3	--	1	4
CEPCC406	Analysis of Algorithms	3		--	3	--	--	3
CEPCC407	Operating System	3	--	--	3	--	--	3
CEPCC408	Computer Network	3	--	--	3	--	--	3
XXMDM402Y	----	3	--	--	3	--	--	3
CELBC403	Analysis of Algorithms Laboratory	--	2	--	--	1	--	1
CELBC404	Operating System Laboratory	--	2	--	--	1	--	1
CELBC405	Computer Network Laboratory	--	2	--	--	1	--	1
CESBL402	Web Development Laboratory	--	4	--	--	2	--	2
CEMNP402	Mini Project-1B	--	3	--	--	1	--	1
VEC402	Environment and Sustainability	2	--	--	2	--	--	2
	Total	**17**	**13**	**1**	**17**	**6**	**1**	**24**

Examination Scheme – SY Semester-IV

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
CEPCC405	Engineering Mathematics-IV	20+25@	30	50	1.5	2	125
CEPCC406	Analysis of Algorithms	20	30	50	1.5	2	100
CEPCC407	Operating System	20	30	50	1.5	2	100
CEPCC408	Computer Network	20	30	50	1.5	2	100
XXMDM402Y	----	20	30	50	1.5	2	100
CELBC403	Analysis of Algorithms Laboratory	25	--	25	--	--	50
CELBC404	Operating System Laboratory	25	--	25	--	--	50
CELBC405	Computer Network Laboratory	25	--	25	--	--	50
CESBL402	Web Development Laboratory	50	--	50	--	--	100
CEMNP402	Mini Project-1B	50	--	50	--	--	100
VEC402	Environment and Sustainability	50	--	--	--	--	50
	Total	**350**	**150**	**425**	--	--	**925**

$ Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

@For continuous assessment of tutorials.

Curriculum Structure – TY Semester-V

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
CEPCC509	Theoretical Computer Science	3	--	--	3	--	--	3
CEPCC510	Software Engineering	3	--	--	3	--	--	3
XXMDM503Y	----	3	--	--	3	--	--	3
CEPEC501Y	Program Elective-I	3	--	--	3	--	--	3
CELBC506	Software Engineering Laboratory	--	2	--	--	1	--	1
CELBC507	Application Development Laboratory	--	2	--	--	1	--	1
XXMDL501Y	----	--	2	--	--	1	--	1
AEC502	Professional Communication and Ethics-II	1	2	--	1	1	--	2
CEMNP503	Mini Project-2A	--	3	--	--	1	--	1
HSS502	Entrepreneurship	2	--	--	2	--	--	2
	Total	**15**	**11**	**--**	**15**	**5**	**--**	**20**

NOTE: Students who choose not to pursue Honours or Minor are welcome to register for the initial two courses of the fifth and sixth semesters' Honours or Minor track in 'Audit' mode (AU). This allows them to explore the course material without the expectation of earning a letter grade. Upon fulfilling the requirements in 'Audit' mode, their participation will be acknowledged on the grade sheet. Audit courses are excluded from grade point averages and have no impact on SGPI/CGPI calculations. For more information on Honours and Minor track courses, please refer to the Institute Handbook for Honours/Minor/Honours in Research degree programs.

Program Elective-I:

Every student is required to take Program Elective for Semester V. Students can take this course from the following list of Program Elective-I.

Program Elective-I	
CEPEC5011	Data Mining & Business Intelligence
CEPEC5012	Advanced Database Systems
CEPEC5013	Computer Graphics and Animation
CEPEC5014	Distributed Computing

Examination Scheme – TY Semester-V

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid - Sem	End-Sem	
CEPCC509	Theoretical Computer Science	20	30	50	1.5	2	100
CEPCC510	Software Engineering	20	30	50	1.5	2	100
XXMDM503Y	----	20	30	50	1.5	2	100
CEPEC501Y	Program Elective-I	20	30	50	1.5	2	100
CELBC506	Software Engineering Laboratory	25	--	25	--	--	50
CELBC507	Application Development Laboratory	25	--	25	--	--	50
XXMDL501Y	----	25	--	25	--	--	50
AEC502	Professional Communication and Ethics-II	50	--	--	--	--	50
CEMNP503	Mini Project-2A	50	--	--	--	--	50
HSS502	Entrepreneurship	50	--	--	--	--	50
Total		**305**	**120**	**275**	--	--	**700**

$ Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

Curriculum Structure – TY Semester-VI

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
CEPCC611	Cryptography and System Security	3		--	3	--	--	3
XXMDM604Y	---	4		--	4	--	--	4
CEPEC602Y	Program Elective-II	3	--	--	3	--	--	3
CELBC608	System Security Laboratory	--	2	--	--	1	--	1
CELBC609	IOT & Cloud Computing Laboratory	--	2	--	--	1	--	1
CESBL603	Full Stack Development Laboratory	--	4	--	--	2	--	2
CEMNP604	Mini Project-2B	--	3	--	--	1	--	1
ELC601	Research Methodology	2	--	--	2	--	--	2
LLC601Y*	Liberal Learning Course	2	--	--	2	--	--	2
	Total	**14**	**11**	**--**	**14**	**5**	**--**	**19**

NOTE: Students who choose not to pursue Honours or Minor are welcome to register for the initial two courses of the fifth and sixth semesters' Honours or Minor track in 'Audit' mode (AU). This allows them to explore the course material without the expectation of earning a letter grade. Upon fulfilling the requirements in 'Audit' mode, their participation will be acknowledged on the grade sheet. Audit courses are excluded from grade point averages and have no impact on SGPI/CGPI calculations. For more information on Honours and Minor track courses, please refer to the Institute Handbook for Honours/Minor/Honours in Research degree prorams.

***Liberal Learning Course:**

Every student should take Liberal Learning Course for Semester VI. Students can take this course from the following list of Liberal Learning Courses.

Liberal Learning Courses	
Course Code	**Course Name**
LLC6011	**Art of Living**
LLC6012	**Yoga and Meditation**
LLC6013	**Health and Wellness**
LLC6014	**Diet and Nutrition**
LLC6015	**Personality Development**

Program Elective-II:

Every student is required to take Program Elective for Semester VI. Students can take this course from the following list of Program Elective-II.

Program Elective-II	
CEPEC6021	Machine Learning ##
CEPEC6022	Big Data Analytics
CEPEC6023	Advanced Networks
CEPEC6024	High Performance Computing

Students who opted the AIML as Honours/Minor vertical should not opt Machine Learning as Program Elective-II.

Examination Scheme – TY Semester-VI

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem. Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
CEPCC611	Cryptography and System Security	20	30	50	1.5	2	100
XXMDM604Y	---	20	30	50	1.5	2	100
CEPEC602Y	Program Elective-II	20	30	50	1.5	2	100
CELBC608	System Security Laboratory	25	--	25	--	--	50
CELBC609	IOT & Cloud Computing Laboratory	25	--	25	--	--	50
CESBL603	Full Stack Development Laboratory	50	--	50	--	--	100
CEMNP604	Mini Project-2B	50	--	50	--	--	100
ELC601	Research Methodology	50	--	--	--	--	50
LLC601Y	Liberal Learning Course	50	--	--	--	--	50
	Total	**310**	**90**	**300**	--	--	**700**

$ Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

Curriculum Structure – B. Tech Semester-VII

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
CEPCC712	Artificial Intelligence	3	--	--	3	--	--	3
CEPEC703Y	Program Elective-III	3	--	--	3	--	--	3
CEPEC704Y	Program Elective-IV	3	--	--	3	--	--	3
OEC701Y	Open Elective-I	3	--	--	3	--	--	3
CELBC710	Computational Intelligence Laboratory	--	2	--	--	1	--	1
CELBC711	Data Engineering Laboratory	--	2	--	--	1	--	1
CEMJP701	Major Project-A	--	6	--	--	2	--	2
HSS703	Financial Planning	2	--	--	2	--	--	2
Total		**14**	**10**	**--**	**14**	**4**	**--**	**18**

Program Elective-III & IV:

Every student is required to take two Program Electives for Semester VII. Students can take this course from the following list of Program Elective-III & Program Elective-IV.

Program Elective-III	
CEPEC7031	Deep Learning ##
CEPEC7032	Natural Language Processing
CEPEC7033	UI/UX
CEPEC7034	Advance Algorithms

Students who opted the AIML as Honours/Minor vertical should not opt Deep Learning as Program Elective-III.

Program Elective-IV	
CEPEC7041	Applied Data Science
CEPEC7042	Blockchain ##
CEPEC7043	Adhoc Networks
CEPEC7044	Compiler Design

Students who opted the Blockchain as Honours/Minor vertical should not opt Blockchain as Program Elective-IV.

Open Elective-I:
Every student is required to take one Open Elective for Semester VII. Students can take this course from the following list of Open Elective-I.

Course Code	Open Elective-I
OEC7011	Product Lifecycle Management
OEC7012	Reliability Engineering
OEC7013	Management Information System
OEC7014	Design of Experiments
OEC7015	Operation Research
OEC7016 @@	Cyber Security and Laws
OEC7017	Disaster Management and Mitigation Measures
OEC7018	Energy Audit and Management
OEC7019	Development Engineering

@@ Students who opted for Honours/Minor vertical as Cybersecurity should not opt Cybersecurity and Laws from the Open Elective-I list.

Examination Scheme – B. Tech Semester-VII

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
CEPCC712	Artificial Intelligence	20	30	50	1.5	2	100
CEPEC703Y	Program Elective-III	20	30	50	1.5	2	100
CEPEC704Y	Program Elective-IV	20	30	50	1.5	2	100
OEC701Y	Open Elective-I	20	30	50	1.5	2	100
CELBC710	Computational Intelligence Laboratory	25	--	25	--	--	50
CELBC711	Data Engineering Laboratory	25	--	25	--	--	50
CEMJP701	Major Project-A	50	--	--	--	--	50
HSS703	Financial Planning	50	--	--	--	--	50
	Total	**230**	**120**	**250**	--	--	**600**

$ Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

Curriculum Structure – B. Tech Semester-VIII

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
CEPEC805Y	Program Elective-V	3	--	--	3	--	--	3
OEC802Y	Open Elective-II	3	--	--	3	--	--	3
CEMJP802	Major Project-B	--	12	--	--	4	--	4
INT801	Internship~	--	--	--	--	8	--	8
	Total	**6**	**12**	**--**	**6**	**12**	**--**	**18**
~ Students have the opportunity to engage in a three-month internship within industry, research organizations, foreign universities, or internal internship for research and product development during the 8th semester, provided they meet the semester requirements and receive approval from the institute.								

Program Elective-V:

Every student is required to take Program Elective for Semester VIII. Students can choose Program Elective - V, from one of the domains listed below. The list of courses within the individual domain will be made available before the course registration.

Program Elective-V	
CEPEC8051	Social Network Data Analytics
CEPEC8052	Software Development
CEPEC8053	Network and Security
CEPEC8054	Machine Intelligence

Open Elective-II:

Every student is required to take one Open Elective for Semester VIII. Students can take this course from the following list of Open Elective-II.

Course Code	Open Elective-II
OEC8021	Project Management
OEC8022	Finance Management
OEC8023	Entrepreneurship Development and Management
OEC8024	Human Resource Management
OEC8025	Professional Ethics and CSR
OEC8026	Circular Economy
OEC8027	IPR and Patenting
OEC8028	Digital Business Management
OEC8029	Environmental Management

Examination Scheme – B. Tech Semester-VIII

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
CEPEC805Y	Program Elective-V	20	30	50	1.5	2	100
OEC802Y	Open Elective-II	20	30	50	1.5	2	100
CEMJP802	Major Project-B	50	--	50	--	--	100
INT801	Internship	50	--	50	--	--	100
Total		**140**	**60**	**200**	--	--	**400**

$ Please refer to the Syllabus for guidelines on in-semester assessments for both theory and laboratory courses.

NOTE: Please note that due to the internship requirement in the 8th semester, theory courses during this semester will be conducted either online synchronously or asynchronously. For more information, please consult the curriculum book of your respective department.

D. Multidisciplinary Minor Courses Offered by the Department for the Other Program Students

Curriculum Structure for MDM Courses

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		L	P	T	L	P	T	Total
CEMDM301	Data Structures and Algorithms	3	--	--	3	--	--	3
CEMDM402	Database Management Systems	3	--	--	3	--	--	3
CEMDM503	Cloud computing	3	--	--	3	--	--	3
CEMDM604	Soft Computing	4	--	--	4	--	--	4
CEMDL601	Machine Learning Laboratory	--	2	--	--	1	--	1
	Total	**13**	**2**	**--**	**13**	**1**	**--**	**14**

Examination Scheme for MDM Courses

Course Code	Course Name	Examination Scheme					Total
		In-Semester Assessment$		End Sem Exam (ESE)	Exam Duration for Theory (in Hrs)		
		Continuous Assessment	Mid-Sem Exam		Mid-Sem	End-Sem	
CEMDM301	Data Structures and Algorithms	20	30	50	1.5	2	100
CEMDM402	Database Management Systems	20	30	50	1.5	2	100
CEMDM503	Cloud computing	20	30	50	1.5	2	100
CEMDM604	Soft Computing	20	30	50	1.5	2	100
CEMDL601	Machine Learning Laboratory	25	--	25	--	--	50
	Total	**105**	**120**	**225**	**--**	**--**	**450**

E. Honours, Minor, and Honours in Research Degree Program

The Honours, Minor, and Honours in Research degree programs aim to empower students by offering specialized courses/research internships or projects in emerging fields of their interest, thus enhancing their proficiency in those areas. Students who achieve a CGPI of 7.5 or higher by the end of the fourth semester are eligible to pursue an additional 18 credits from the fifth to eighth semesters to qualify for a B. Tech degree with Honours, Minor, or Honours in Research designation. Students need to refer to the Institute level Handbook for Honours/Minor/Honours in Research Degree Programs for further details.

F. Second Year Syllabi

Course Type	Course Code	Course Name	Credits
PCC	CEPCC301	ENGINEERING MATHEMATICS-III	03+01*

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20 + 25*	30	50	1.5	2	125

**** Tutorial***

Pre-requisite:

1. BSC101- Engineering Mathematics-I
2. BSC204- Engineering Mathematics-II

Program Outcomes addressed :

1. PO1: Engineering knowledge
2. PO2: Problem analysis

Course Objectives:

1. To provide the basic knowledge on the concepts of Mathematics in the field of Engineering.
2. To build a foundation to the methodology necessary for solving problems by applying the knowledge of Mathematics to the field of Engineering.

Module	Details	Hrs.
	Course Introduction Engineering Mathematics III is often a foundational course designed to provide students with the mathematical tools and concepts essential for various engineering disciplines. Engineering Mathematics III has many applications in Information Technology such as 1) Application of Vector space in Deep Learning, dimension reduction and Neural Networks 2) Application of Linear Mapping in Deep learning and Machine Learning. 3) Application of Number theory in Cryptography.	**01**
01.	**Vector Space** ***Learning Objective/s:*** *To analyse the definition of basis and apply it to determine the basis of a Vector Space*	**7-9**
	Contents: Vector Spaces, Examples of vector spaces, Subspace, Linear combinations, Spanning Set, Linear Independence and Dependence. Linear spans, Row Space of a matrix , Basis and Dimensions	

	Self-Learning Topics: *Independence and Dependence of functions*	
	Learning Outcomes : *A learner will be able to* 1. *Apply the axioms of closure, addition and scalar multiplication and prove that the given set of vectors is a Vector Space (P.I. -1,1,1)* 2. *Identify the conditions of closure and prove that the given space is a Subspace. (P.I. -2.1.3)* 3. *Applying the condition for linear independency and the span of a Vector Space determine the basis of a vector space. (P.I. 1.1.2)* 4. *Identifying Vector space and its operations to determine the Basis of a Vector Space.(P.I. -2.1.2)*	
02.	**Linear Mappings** ***Learning Objective/s:*** *To apply the concepts of kernel and image of a linear map to compute and analyze rank and nullity.*	**6-8**
	Contents: Mappings, Function, Linear mapping (Linear transformations), Kernel and Image of a linear mapping, Rank and Nullity, Singular and non-singular mapping, Isomorphism.	
	Self-Learning Topics: *Operations with linear mapping.*	
	Learning Outcomes : *A learner will be able to* 1. *Apply the properties of vector addition and scalar multiplication to examine whether the mapping is a Linear mapping. (P.I. 1.1.1)* 2. *Determine the rank and nullity of the given linear mapping using the matrix method (P.I. 1.1.2)* 3. *Examine whether a mapping is a linear mapping and find its kernel and image. (P.I. 2.1.2)* 4. *Analyze the property of homomorphism and one-one mapping isomorphism between the vector space to prove that the mapping an isomorphism. (P. I. -2.1.3)*	
03.	**Linear Mappings and Matrices** ***Learning Objective/s:*** *To analyze and compute the change of basis matrix for linear map.*	**6-8**
	Contents: Introduction, Matrix Representations of a linear operator, Change of Basis, Similarity, Matrices and general linear mapping.	

	Learning Outcomes : *A learner will be able to* 1. *Express the linear mapping in the matrix form.(P.I. 1.1.1)* 2. *Apply matrix method to prove that the two linear mappings are similar (P.I. 1.1.2)* 3. *Analyse the procedure and determine the change of Basis Matrix from one coordinate system to the alternate coordinate system (P.I. 2.1.3)* 4. *Apply the concept of linear mapping and the change of basis to solve mathematical models (P.I.- 2.4.1)*	
04.	**Inner Product spaces, Orthogonality** ***Learning Objective/s:*** *To analyse and apply Gram-Schmidt Technique to determine an Orthonormal Basis.*	**6-8**
	Contents: Introduction, Inner product spaces, Examples of Inner product spaces, Cauchy-Schwartz Inequality, Orthogonality, orthogonal sets and Basis Gram-Schmidt orthogonalizations process	
	Learning Outcomes : *A learner will be able to* 1 *Prove Cauchy's Schwartz inequality for set of all matrices (P.I. 1.1.1)* 2 *Identify the inner product space to prove that the set of vectors is an orthogonal basis. (P.I. 2.1.2)* 3 *Identify the inner product space to prove that the set of vectors is an orthonormal basis. (P.I. 2.1.3)* 4 *Apply Gram Schmidt Technique to determine the orthonormal basis. (P.I. 1.1.2)*	
05.	**Number Theory** ***Learning Objective/s:*** *To identify and apply the appropriate theorem of Number Theory to solve the simultaneous system of congruences.*	**6-8**
	Contents: Modular Arithmetic, Divisibility and Euclid Algorithm, Primes and Sieve of Eratosthenes, Euler's and Fermat Little Theorem, Congruences, Computing Inverse in Congruences, Legendre and Jacobi Symbol, Chinese Remainder Theorem	
	Self-Learning Topics: *Testing for Primes, Prime Number Theorem*	
	Learning Outcomes : *A learner will be able to* 1. *Apply Euler's or Fermat little theorem to solve congruent modulo equation. (P.I-. 1.1.1)* 2. *Formulate and solve the linear congruent equation for the given problem. (P.I.- 2.2.2)* 3. *Identify appropriate theorem and solve the linear congruent equation. (P.I.- 2.1.3)* 4. *Apply Chinese Remainder theorem to solve the given simultaneous linear congruence.(P. I. - 1.1.2)*	
06.	**Numerical Methods** ***Learning Objective/s:*** *To analyse and apply the appropriate numerical method to solve transcendental equation and system of simultaneous equations.*	**6-8**

	Contents: Solution of Transcendental Equations: Newton Raphson method, Regula – Falsi Method, Solution of system of linear algebraic equations, Gauss Jacobi Iteration Method, Gauss Seidel Iteration Method.	
	Self-Learning Topics: *Bisection Method, Gauss Elimination Method.*	
	Learning Outcomes : *A learner will be able to* 1. *Apply Newton Raphson method and Regula Falsi method to solve the transcendental equation. (P.I.- 1.1.1)* 2. *Apply Gauss Jordan or Gauss Siedel Iterative method to solve the system of equations. (P.I.- 1.1.2)* 3. *Identify the appropriate numerical method to solve the system of equation. (P.I.- 2.1.3)* 4. *Examine the limitation for the convergent solution of system of equation is using iterative method.(P.I.- 2.4.3)*	
	Course Conclusion Engineering Mathematics provides the language and framework through which engineers model, analyze, and optimize systems.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.1.1	Apply the knowledge of discrete structures, linear algebra, statistics and numerical techniques to solve problems.
1.1.2	Apply the concepts of probability, statistics and queuing theory in modelling of computer-based system, data and network protocols.
2.1.2	Identify processes/modules/algorithms of a computer-based system and parameters to solve a problem
2.1.3	Identify mathematical algorithmic knowledge that applies to a given problem
2.2.2	Identify functionalities and computing resources.
2.4.1	Applies engineering mathematics to implement the solution.
2.4.3	Identify the limitations of the solution and sources/causes.

Course Outcomes :

1. Apply the fundamentals of Vector Space to analyse the Basis of the Vector Space. *(LO 1.1, LO 1.2, LO 1.3, LO 1.4)*
2. Apply the fundamentals of Linear Mapping to identify the Kernel and Image of the Linear Mapping. *(LO 2.1, LO 2.2, LO 2.3, LO 2.4, LO 3.1, LO 3.2, LO 3.3, LO 3.4)*
3. Analyse the Inner Product Space and apply the properties to determine the orthogonal basis. *(LO 4.1, LO 4.2, LO 4.3, LO 4.4)*
4. Identify and apply the appropriate theorem of Number Theory to solve the congruent equation and the simultaneous congruent equations. *(LO 5.1, LO 5.2, LO 5.3, LO 5.4)*
5. Identify and apply appropriate numerical methods to solve System of equations. *(LO 6.1, LO 6.2, LO 6.3, LO 6.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC301.1	3	2	-	-	-	-	-	-	-	-	-
CEPCC301.2	3	2	-	-	-	-	-	-	-	-	-
CEPCC301.3	3	2	-	-	-	-	-	-	-	-	-
CEPCC301.4	3	2	-	-	-	-	-	-	-	-	-
CEPCC301.5	3	2	-	-	-	-	-	-	-	-	-
Average	**3**	**2**	-	-	-	-	-	-	-	-	-

Text Books :

1. C. L. Liu: Elements of Discrete Mathematics, 2nd edition, TMH.
2. Cryptography and Network Security by B. A. Forouzan & D. Mukhopadhyay, 11th edition, McGraw Hill Publication.

Reference Books :

1. Elementary Number Theory and its applications by Kenneth H. Rosen, 5th edition, Addison Wesley Publication 2005.
2. Advanced Engineering Mathematics by C. Ray Wylie & Louis Barrett. TMH International Edition
3. Seymour Lipschutz "Beginning Linear Algebra" Schaum's outline series, Mc-Graw Hill Publication.

Other Resources :

1. NPTEL Course: Applied Linear Algebra IIT Madras: Dr Andrew Thangaraj: :-Web link- https://nptel.ac.in/courses/108106171.
2. NPTEL Course: Computational Number Theory and Cryptography IIT Guwahati: Dr Pinaki Mishra:- Web link- https://nptel.ac.in/courses/106103015.
3. NPTEL Course::NOC: Numerical Methods, IIT Roorkee Prof. Ameeya Kumar Nayak, Prof. Sanjeev Kumar - Web link- https://nptel.ac.in/courses/111107105.

IN-SEMESTER ASSESSMENT (75 MARKS)

1. Continuous Assessment-Theory (20 Marks)

Suggested breakup of distribution

One MCQ test as per GATE exam pattern / level: 05 Marks.

One Class test: 05 Marks

One Team-Pair-Solo activity: 05 Marks

Regularity and active participation: 05 Marks

2. Continuous Assessment - Tutorial (25 Marks)

Suggested breakup of distribution

Tutorial Assignments and Class tests: 20 Marks

Students must be encouraged to write at least 6 class tutorials. At least 6 Class tests will be conducted based on class tutorials on entire syllabus. Each class tests carries 20 Marks. Average will be taken of all class tests**.** Regularity and active participation: 05 Marks.

3. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20%- 30% weightage, and the syllabus covered from MSE to ESE carrying 70%- 80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	CEPCC302	DISCRETE STRUCTURES AND GRAPH THEORY	03+01*

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20 + 25*	30	50	1.5	2	125

**** Tutorial***

Pre-requisite:

1. BSC101- Engineering Mathematics-I
2. BSC204-Engineering Mathematics-II

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions

Course Objectives:

1. To inculcate the notion of Logical Thinking.
2. To familiarize with the concepts of Sets, Relations, and Functions.
3. To familiarize with the concepts of Posets and Lattice.
4. To enable to apply the advanced concepts of Counting and Recurrence Relations.
5. To acquaint learners with Algebraic Structures and Coding Theory.

Module	Details	Hrs
	Course Introduction Discrete structures and graph theory enhances problem-solving abilities, vital for algorithm design, network analysis, cryptography, and software engineering, driving professional growth in computer engineering fields.	**01**
01.	**Fundamental of logic:** ***Learning Objective:*** *Students are expected to apply mathematical logic to solve logical problems and mathematical induction.*	**5-7**
	Contents: Basic connectives and truth tables, Logic equivalence- laws of logic Logical Implication-Rules of Inference, Fundamentals of Logic-Use of Quantifier, Mathematical Induction.	

	Self-Learning Topics: *Normal forms*	
	Learning Outcomes: *A learner will be able to* *LO1.1: Verify the correctness of an argument using propositional and predicate logic. (P.I.- 1.1.1)* *LO1.2: Prove that the given statement is factual using mathematical induction (P.I.- 1.3.1)* *LO1.3: Find the logical equivalent of the given statement. (P.I.- 1.4.1)*	
02.	**Relations and Functions:** ***Learning Objective:*** *Students are expected to identify and analyze the operations associated with set theory, functions and relations.*	**7-9**
	Contents: Basic concepts of Set Theory, Relations: Definition, Types of Relations, Representation of Relations, Closures of Relations, Wars hall's algorithm, Equivalence relations and Equivalence Classes, Functions: Definition, Types of functions, Composition of functions, Identity and Inverse function.	
	Self-Learning Topics: *Equivalence Classes.*	
	Learning Outcomes: *A learner will be able to* *LO2.1: Identify if the given relation is an equivalence relation. (P.I.- 1.1.1)* *LO2.2: Solve problems based on set theory, functions. (P.I.- 1.2.1)* *LO2.3: Find the transitive closure using Warshall's algorithm. (P.I.- 1.4.1)* *LO2.4: Find the composition of the function. (P.I.-1.3.1)*	
03.	**Posets and Lattice** ***Learning Objective:*** *Student are expected to construct hasse diagram using poset and analyse chain, anti-chain and Lattice using Hasse diagram*	**5-7**
	Contents: Partial Order Relations, Poset, Hasse Diagram, Chain and Anti chains, Lattice, Types of Lattices, Sub lattice, Application of lattice in real life	
	Self-Learning Topics: *Isomorphism in Lattice.*	
	Learning Outcomes: *A learner will be able to* *LO3.1: Construct Hasse Diagram of the Poset by applying the partial order relation (P.I.- 2.1.3)* *LO3.2:Identify and analyze chain, anti-chain and lattice in Hasse diagram (P.I.-2.4.1)*	
04.	**Fundamental Principle of Counting**	**6-8**
	Learning Objectives:	
	Students are expected to solve recurrence relations and apply advance concepts of counting to provide the solution to associated problems	

	Contents: Basic Counting Principle-Sum Rule, Product Rule, Inclusion Exclusion Principle, Pigeonhole Principal Recurrence relations, Solving recurrence relations	
	Self-Learning Topics: *Permutation and combination* ***Learning Outcomes:*** *A learner will be able to* *LO4.1: Use the sum and product rule to solve counting problems. (P.I.- 1.1.2)* *LO4.2: Use the inclusion exclusion principle to solve counting problem (P.I.- 1.2.1)* *LO4.3: Find the generating sequence of the recurrence relation. (P.I.- 1.4.1)* *LO4.4:Solve recurrence relation using characteristic roots (P.I.- 1.3.1)*	
05.	**Groups** ***Learning Objective/s:*** *To acquaint students with algebraic structure and able to design out lane for real life problems.*	**7-9**
	Contents: Semi group, Monoid, Groups, Subgroups, Abelian Group, Cyclic group, Isomorphism of group, Application of algebraic structure in real life	
	Self-Learning Topics: *Co-sets and Lagrange's theorem* ***Learning Outcomes :*** *A learner will be able to* *LO5.1: Identify if the given operation is algebraic structure (P.I.- 1.3.1)* *LO5.2: Identify Semi-group, monoid, groups, subgroups and abelian based on the binary operation. (P.I.- 1.1.1)* *LO5.3: Identify if the groups are isomorphic. (P.I.- 1.4.1)* *LO5.4: Able to model algebraic structure concept in real life applications (P.I.-2.4.1)* *LO5.5: Able to identify specified groups concept to find the solution real life applications (P.I-2.1.3)*	
06.	**Graph Theory** ***Learning Objective/s:*** *Student will expect to design and form a solution to real life problems through the concept of graph theory*	**7-9**
	Contents:	
	Types of graphs, Graph Representation, Sub graphs, Operations on Graphs, Walk, Path, Circuit, Connected Graphs, Disconnected Graph, Components, Homomorphism and Isomorphism of Graphs Euler and Hamiltonian Graphs, Planar Graph, Application of Graph Theory in Real life (Computer Network, Social media analytics, etc.)	

	Self-Learning Topics: *Cut vertex, cut set vertex*	
	Learning Outcomes: *A learner will be able to* *LO6.1: Identify walk, path circuit of the graph (P.I.- 1.1.1)* *LO6.2: Identify if the graph is homomorphic or isomorphic (P.I.- 1.4.1)* *LO6.3: Apply the theorems to identify if the graph is Euler or Hamiltonian graph (P.I.- 1.3.1)* *LO6.4: Design a graphical model solution to different application based on real life scenario (P.I.- 3.1.1)* *LO6.5: Design and validate community detection for real life application (P.I.- 3.4.3)*	
	Course Conclusion In conclusion, proficiency in discrete structures and graph theory is invaluable for problem-solving in computer engineering, facilitating career growth and impactful contributions across diverse domains.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	**P.I. Statement**
1.1.1	Apply the knowledge of discrete structures, linear algebra, statistics and numerical techniques to solve problems.
1.1.2	Apply the concepts of probability, statistics and queuing theory in modelling of computer-based system, data and network protocols.
1.2.1	Apply laws of natural science to an engineering problem.
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science and engineering to solve an engineering problem
2.1.3	Identify mathematical algorithmic knowledge that applies to a given problem
2.4.1	Applies engineering mathematics to implement the solution.
3.1.1	Able to define a precise problem statement with objectives and scope.
3.4.3	Able to verify the functionalities and validate the design.

Course Outcomes: A learner will be able to -

1. Solve the basic problems of Logical Thinking. (*LO1.1, LO1.2, LO1.3)*
2. Apply basic principles of Set theory, Relation, and Functions to solve the problem. (*LO2.1, LO2.2, LO2.3, LO2.4)*
3. Apply and analyse the basic concepts of Posets and Lattice to find the solution to computer engineering problem. (*LO3.1, LO3.2)*
4. Apply the concepts of counting and recurrence relations to solve the given problems. (*LO4.1, LO4.2, LO4.3, LO4.4, LO4.5)*
5. Design a solution by using the concepts of Groups and Graph Theory for the problems associated with various domains of Computer Engineering. (*LO5.1, LO5.2, LO5.3, LO5.4, LO5.5, LO6.1, LO6.2, LO6.3, LO6.4, LO6.5)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC303.1	3	-	-	-	-	-	-	-	-	-	-
CEPCC303.2	3	-	-	-	-	-	-	-	-	-	-
CEPCC303.3	-	3	-	-	-	-	-	-	-	-	-
CEPCC303.4	3	-	-	-	-	-	-	-	-	-	-
CEPCC303.5	3	3	3	-	-	-	-	-	-	-	-
Average	**3**	**3**	**3**	**-**	-	-	-	-	-	-	-

Text Books :

1. Discrete Mathematical Structures, Bernad Kolman, Robert Busby, Sharon Cutler Ross, Nadeem-ur-Rehman, Sixth edition , 2015 Pearson Education.
2. Discrete Mathematics and applications, K. H. Rosen, fifth edition 2003, Tata McGraw Hill Publishing Company.
3. Elements of Discrete Mathematics, C. L. Liu, second edition 1985, Reprinted 2000, McGraw-Hill Book Company.

Reference Books :

1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel, T. P. Baker, Second Edition 1986, Prentice Hall of India.
2. Discrete Mathematical Structures with Applications to Computer Science, J. P. Trembley, R. Manohar ,2002,Tata McGraw Hill Publishing Company.
3. Graph Theory with applications to engineering and computer science, Narsing Deo, 1st publication 2016, PHI Publications.
4. Discrete Mathematics, P. K. Bisht, H. S. Dhami, 1st publication 2015, Oxford press.
5. Discrete Mathematical Structures, Y N Singh, 2010,Wiley-India.

Other Resources :

1. NPTEL Course: Discrete Mathematics By Prof. Sudarshan Iyengar, Department of Computer Science Engineering, IIT Ropar :-Web link- https://onlinecourses.nptel.ac.in/noc19_cs49/
2. NPTEL Course: Discrete Mathematics By Prof. Sourav Chakraborty,Department of Computer Science Engineering, IIT Madras :-Web link- https:://nptel.ac.in/courses/111/106/111106086/
3. NPTEL Course: Discrete Mathematics By Prof. Ashish Choudhury, Department of Computer Engineering, IIIT Bangalore:- Web link- https://nptel.ac.in/courses/107/106/107106081/

IN-SEMESTER ASSESSMENT (75 MARKS)

1. Continuous Assessment - Theory-(20 Marks)

Suggested breakup of distribution

MCQ test strictly as per GATE exam pattern / level): 05 Marks

Class test: 05 Marks

Open book test/ Open notes test: 05 Marks

Regularity and active participation : 05

Marks

2. Continuous Assessment - Tutorial- (25 Marks)

Suggested breakup of distribution

Numerical Assignments: 10 Marks

Class test based on above numerical assignment: 10 Marks

Regularity and active participation : 05 Marks

3. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20%-30% weightage, and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	CEPCC303	DATA STRUCTURES	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite:

1. ESL103- Programming Laboratory-I (C)

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct investigations of complex problems

Course Objectives:

1. To impart the fundamental knowledge of data structures.
2. To instruct learners in applying the most suitable data structures to various applications.
3. To instruct learners in comparing various data structures.

Module	Details	Hrs
	Course Introduction Data Structure: This course provides a foundational exploration of data structures, covering concepts such as stacks, queues, and linked lists. Learners will gain proficiency in implementing these structures using programming languages and will develop the analytical skills to choose the appropriate data structure for various applications.	**01**
01.	**Introduction to Data Structures** ***Learning Objective/s:*** *Students are expected to apply engineering knowledge to grasp concepts of various data structure.*	**3-5**
	Contents: 1.1 Introduction to Data Structures – Basic Terminology, Importance of Data Structures 1.2 Types of Data Structures, Operations on Data Structures. 1.3 Abstract Data type (ADT), Advantages of Data structures 1.4 Applications of Data Structures	
	Self-Learning Topics: *Array Data Structure, Elementary Data Structure Organization*	

	Learning Outcomes: *A learner will be able to:* *LO1.1: Apply the concepts of data types and arrays to grasp the concept of data structures. (P.I. - 1.3.1)* *LO1.2: Apply the concept of data structures to write an Abstract Data Type. (P.I. - 1.4.1)* *LO1.3: Compare and contrast various data structures (P.I. - 2.2.4)*	
02.	**Stacks and Queues** ***Learning Objective/s:*** *Expected to write functions to perform operations like insertion, deletion, and traversal operations on-stack and queue data structures.*	**9-11**
	Contents: 2.1 Stacks- Introduction, ADT of Stack, Operations on Stack, Implementation of Stack using array, Applications of Stack- Well form-ness of Parenthesis, Infix to Postfix Conversion and Postfix Evaluation, Reverse String 2.2 Queues - Introduction, ADT of Queue, Operations on Queue, Array Implementation of Queue, Types of Queue-Circular Queue, Priority Queue, Double-Ended Queue, Applications of Queue.	
	Self-Learning Topics: *Use of stack in Recursion, Multiple Queue*	
	Learning Outcomes: *A learner will be able to* *LO2.1: Apply the concepts of data structures to perceive the abstract data types of stack and queue. (P.I. - 1.3.1)* *LO2.2: Compare and contrast array, stack, and queue data structures (P.I. - 2.2.4)* LO2.3: *Write separate functions for each operation of stack and queue data structure and integrate them. (P.I. - 3.4.2)* LO2.4: *Represent data in a predefined format to facilitate explanation of the data (P.I. – 4.3.3)*	
03.	**Linked Lists** ***Learning Objective/s:*** *Expected to write functions to perform operations like insertion, deletion, and traversal operations on linked list data structures.*	**9-11**
	Contents: 3.1 Linked Lists- Basic Terminologies, Representation on Linked List, Linked Lists versus Arrays, Memory Allocation and De-allocation for a Linked List, 3.2 Types of Linked List, Singly Linked List: Traversing a SLL, searching for a value in a SLL, inserting a New Node in a SLL, Deleting a Node from SLL, 3.3 Doubly Linked List: Traversing a DLL, searching for a Value in a DLL, inserting a New Node in a DLL, Deleting a Node from DLL 3.4 Applications of LL: Stack and Queue implementation using LL	
	Self-Learning Topics: *Circular Linked List, Polynomial Representation of equation using LL.*	

	Learning Outcomes: *A learner will be able to* *LO3.1: Use the procedure to perform various operations on linked list data structures. (P.I. - 2.1.2)* *LO3.2: Compare and contrast array, stack, queue, and linked list data structures (P.I. - 2.2.4)* *LO3.3: Write separate functions for each operation of linked list data structure and integrate them. (P.I. - 3.4.2)* *LO3.4: Represent data in predefined form to facilitate explanation of the data, (P.I. – 4.3.3)*	
04.	**Trees** ***Learning Objective/s:*** *Expected to write functions to perform operations like insertion, deletion, and traversal operation on trees data structures.*	**9-11**
	Contents: 4.1 Introduction, Tree Terminologies, Types of Binary trees 4.2 Binary Tree representation, Binary Tree traversals techniques, constructing a Binary Tree from Traversal 4.3 Applications of Binary Tree - Expression Tree, Huffman encoding 4.4 Efficient binary tree: Binary Search Tree and operations on Binary search tree. AVL Trees, Operations on AVL Trees, 4.5 Multi-way Tree - Introduction of multiway tree, Types of multiway tree, B Tree – Operations on B tree, Applications of B tree.	
	Self-Learning Topics: *Threaded Binary Tree, B+ Tree*	
	Learning Outcomes: *A learner will be able to* *LO 4.1: Use algorithms to represent various operations on tree data structures. (P.I. - 2.1.2)* *LO 4.2: Compare and contrast linear and tree data structures (P.I. - 2.2.4)* *LO4.3: Write separate functions for each operation of tree data structure and integrate them. (P.I. - 3.4.2)* *LO4.4: Represent data in predefined form to facilitate explanation of the data (P.I. – 4.3.3)*	
05.	**Graphs** ***Learning Objective/s:*** *Expected to write functions to perform operations like insertion, deletion, and traversal operation on graphs data structures.*	**4-6**
	Contents: 5.1 Introduction, Basic Graph Terminologies, 5.2 Representation of Graph, 5.3 Graph Traversals: Depth First Search (DFS) and Breadth First Search (BFS), 5.4 Topological Sorting. 5.5 Application of Graph	

	Learning Outcomes: *A learner will be able to* *LO 5.1: Use algorithms to represent various operations on graph data structures. (P.I. - 2.1.2)* *LO5.2: Compare and contrast linear and non-linear data structures (P.I. - 2.2.4)* *LO5.3: Write separate functions for each operation of graph data structure and integrate them. (P.I. - 3.4.2)* *LO5.4: Represent data in predefined form to facilitate explanation of the data (P.I. – 4.3.3)*	
06.	**Searching Techniques**	**3-5**
	Learning Objective/s: *To develop proficiency in identifying, comparing, listing collision techniques, solving collision-related problems, and implementing various searching techniques.*	
	Contents: 6.1 Introduction, Hash table, Hash Function: Different Hash Functions, 6.2 Collision resolution techniques 6.3 Pros and Cons of Hashing, Applications of Hashing 6.4 Searching Techniques: Introduction, Linear Search, Binary Search	
	Self-Learning Topics: *Real world application of hashing*	
	Learning Outcomes: *A learner will be able to* *LO 6.1: Select and apply hashing technique, collision resolution technique, searching technique for a given problem (P.I. – 2.1.2)* *LO 6.2: Apply engineering mathematics to solve the given problem (P.I.- 2.4.1)* *LO 6.3: Compare and contrast various searching techniques (P.I. - 2.2.4)* *LO 6.4: Represent data in predefined form to facilitate explanation of the data, (P.I. – 4.3.3)*	
	Course Conclusion Students will be able to write procedures for different data structures. They will be capable of choosing the appropriate data structure for a specific problem and recognizing various uses of data structures.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science and engineering to solve an engineering problem
2.1.2	Identify processes/modules/algorithms of a computer-based system and parameters to solve a problem.
2.2.4	Compare and contrast alternative solution/methods to select the best methods.
2.4.1	Applies engineering mathematics to implement the solution.
3.4.2	Able to write separate functions for each operation of linear data structure and integrate them. (Modified P.I.)
4.3.3	Represent data in predefined form so as to facilitate explanation of the data. (Modified P.I.)

Course Outcomes: A learner will be able to-

1. Represent various operations of data structure as a function and integrate them. *(LO1.1, LO1.2, LO1.3)*
2. Identify and apply a suitable linear data structure for a given problem.*(LO2.1, LO 2.2, LO 2.3, LO 3.1, LO 3.2, LO 3.3,)*
3. Identify and apply a suitable non-linear data structure for a given problem.*(LO 4.1, LO 4.2, LO 4.3, LO 5.1, LO 5.2, LO 5.3)*
4. Apply an appropriate searching technique for a given problem.*(LO 6.1, LO 6.2, LO 6.3)*
5. Analyse various data structures. *(LO 2.4, LO 3.4, LO 4.4, LO 5.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC303.1	3	2	-	-	-	-	-	-	-	-	-
CEPCC303.2	2	3	2	-	-	-	-	-	-	-	-
CEPCC303.3	-	3	2	-	-	-	-	-	-	-	-
CEPCC303.4	-	3	-	2	-	-	-	-	-	-	-
CEPCC303.5	-	-	-	2	-	-	-	-	-	-	-
Average	**3**	**3**	**2**	**2**	-	-	-	-	-	-	-

Text Books:

1. Data Structures Using C, Aaron M Tenenbaum, Yedidyah Langsam, Moshe J Augenstein, 1st Edition, 2019, Pearson Publication.
2. Data Structures using C, Reema Thareja, 2nd Edition, 2014, Oxford Press.
3. Data Structures: A Pseudocode Approach with C, Richard F. Gilberg and Behrouz A. Forouzan, 2nd Edition, 2007, CENGAGE Learning.

Reference Books:

1. Data Structures using C, E Balagurusamy, 1st Edition, 2013, McGraw-Hill Education India
2. Data Structures using C and C++, Rajesh K Shukla, 1st Edition, 2009, Wiley-India.

Other Resources:

1. NPTEL Course: Data Structures and Algorithms, by Prof. Naveen Garg, Department of Computer Science and Engineering Department, IIT Delhi:-Web link- https://nptel.ac.in/courses/106/102/106102064/

IN-SEMESTER ASSESSMENT (50 MARKS)

1.Continuous Assessment (20 Marks)

Suggested breakup of distribution

One MCQ test as per GATE exam pattern/ level: 05 marks.

One Class test: 05 marks

Open book test/ Open notes test: 05 Marks

Observation and active participation:05 Marks

2. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20%- 30% weightage, and the syllabus covered from MSE to ESE carrying 70% - 80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	CEPCC304	DATABASE MANAGEMENT SYSTEM	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite: *Nil*

Program Outcomes addressed:

1. PO1: Engineering knowledge.
2. PO2: Problem analysis.
3. PO3: Design/development of solutions.
4. PO11: Life-long learning.

Course Objectives:

1. To familiarize fundamental concepts of database management System.
2. To guide in designing ER/EER Models and Relational model for specific applications.
3. To guide in designing relational database model and formulate relational algebra queries.
4. To guide in formulating appropriate query.
5. To impart the concepts of transaction processing, concurrency control, and database recovery techniques by emphasizing their pivotal roles in operational efficiency.

Module	Details	Hrs.
	Course Introduction Database Management System course is vital for computer engineering professionals, equipping them with essential skills in database architecture, optimization, and resilience, crucial for data-centric projects and advancing research, thereby enhancing their professional competence in the field.	**01**
01.	**Introduction Database Concepts** ***Learning Objective:*** *Expected to apply database theory to elucidate core concepts and functionalities, and depict architecture with key components, structure, and dynamics.*	**2-4**

	Contents: 1.1 Basic Concept: -Purpose of Database Systems, Data models, File system v/s Database system. 1.2 Database Architecture: -Views of data, three-schema architecture of DBMS, Data abstraction and data Independence, Database Administrator, Database users, DBMS system architecture.	
	Self-Learning Topics: *Client/Server Architectures for DBMS, Database Storage Structures.*	
	Learning Outcomes: *A learner will be able to* *LO1.1: Use core principles of engineering to understand the importance of database system. (P.I.-1.3.1)* *LO1.2: To apply computer engineering concepts to illustrate the database architecture, providing comprehensive summary of roles and responsibilities of various database users along with operational details. (P.I.-1.4.1)*	
02.	**Entity–Relationship Data Model** ***Learning Objective:*** *Expected to design tailored Entity-Relationship Models for specific applications, and delve into advanced EER Model concepts.*	**5-7**
	Contents:	
	2.1 The Entity-Relationship (ER) Model: - Entity types: Weak and strong entity sets, Entity sets, Types of Attributes, Keys, roles, Relationship, Relationship set, Relationship constraints: Cardinality and Participation, ER diagrams. 2.2 Extended Entity Relationship (EER)Model: Generalization, Specialization and Aggregation, EER Diagrams.	
	Self-Learning Topics: *Constraints and Characteristics of Specialization and Generalization Hierarchies.*	
	Learning Outcomes : *A learner will be able to* *LO2.1: Use the knowledge of discrete structures to define cardinality ratio on ER. (P.I.-1.1.1)* *LO2.2: Apply theory and principles of ER/EER to provide solution to real world problem. (P.I.-1.4.1)* *LO2.3: Evaluate the problem statement and identify various components of ER/EER. (P.I.-2.1.1)* *LO2.4: Identify the participation constraints on ER. (P.I.-2.3.2)* *LO2.5: Design ER/EER data model for the real-world problem. (P.I.-3.2.2)* *LO2.6: Examine ER/EER data model and validate it. (P.I.-3.4.3)*	
03.	**Relational Model and Relational Algebra** ***Learning Objective:*** *Expected to design relational database schema and formulate relational algebra queries.*	**5-7**

	Contents: 3.1 Relational Model: Introduction to the Relational Model, relational database schemas, concept of keys, Mapping the ER and EER Model to the Relational Model, 3.2 Relational Algebra operators: Unary and Binary relational operations, additional relational operations: Aggregate, grouping Examples of Queries in relational algebra.	
	Self-Learning Topics: *The Tuple Relational Calculus, The Domain Relational Calculus*.	
	Learning Outcomes: *Learner should be able to* *LO3.1: Identify process/rules to appropriately map ER model to relational model. (P.I.-2.1.2)* *LO3.2: Identify appropriate mapping of relationships based on the cardinality to ensure performance is not hampered. (P.I.-2.3.2)* *LO3.3: Design Relational model from conceptual model. (P.I.-3.2.2)* *LO3.4: Examine the relational model and validate it. (P.I.-3.4.3)* *LO3.5: Formulate appropriate relational algebraic query statement to retrieve requires data. (P.I.-1.1.1)* *LO3.6: Identify suitable operator useful to retrieve required information. (P.I.-1.4.1)*	
04.	**Structured Query Language (SQL)** ***Learning Objective:*** *Expected to apply adeptly formulate query for retrieving data.*	**9-11**
	Contents: 4.1 SQL Introduction: -SQL Data Definition and Data Types, Integrity constraints: entity integrity constraint, key constraints, Domain Constraints, Referential integrity, check constraints. 4.2 Querying in SQL: -Data Manipulation commands, Basic Retrieval Queries in SQL, set operator, string operator, inner join, outer join, nested and complex queries, aggregate functions, group by and having clause, Views in SQL, triggers, Data Control commands.	
	Self-Learning Topics: *Database Stored Procedures and functions*.	
	Learning Outcomes: *A learner will be able to* LO4.1: *Use the knowledge of query language in formulating appropriate query. (P.I.-1.1.1)* LO4.2: *Apply theory and principles of SQL concepts and constraints to enable database correctness. (P.I.-1.4.1)* LO4.3: *Interpret the statement and identify suitable clauses useful to fetch data. (P.I.-2.1.1)* LO4.4: *Formulate suitable SQL query by ensuring ongoing skill development and adaptation to evolving SQL advancements. (P.I.-2.1.2 & P.I.-11.2.1)*	

05.	**Relational Database Design** ***Learning Objective:*** *Expected to apply normalization to enhance the performance relational database design.* **Contents:** 5.1 Introduction to Relational-Database Design Pitfalls in Relational Database designs, Functional Dependency (Basic concepts, F+, Closure of an Attribute set, Armstrong's axioms), Concept of Decomposition, Desirable Properties of Decomposition (Lossless join, Lossy join, Dependency Preservation) 5.2 Normalization: -Concept of normalization, Normal Forms: 1NF,2NF and 3NF, Boyce-Codd Normal Form. ***Self-Learning Topics:*** *Multivalued Dependency and Fourth Normal Form* ***Learning Outcomes :*** *A learner will be able to* *LO5.1: Apply the properties of Relational database design to evaluate the relational model. (P.I.-1.1.1)* *LO5.2: Apply properties of decomposition to normalize the relational model to enhance the performance. (P.I.-1.4.1)*	**7-9**
06.	**Transactions Management, Concurrency Control and Database Recovery** ***Learning Objective:*** *Expected to apply concepts of transaction processing, concurrency control and Database Recovery techniques and recognize the importance in real world application* **Contents:** 6.1 Transaction Management: Transaction concept, Transaction states, ACID properties, Transaction Control Commands, Concurrent Executions, characterizing schedules based on Serializability (Serializability-Conflict and View). 6.2 Concurrency Control: Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering. 6.3 Recovery System: Recovery Concepts, Recovery based on Deferred update, Recovery techniques based on immediate update, Shadow paging, Log based recovery. ***Self-Learning Topics:*** *Characterizing schedules based on recoverability, Database backup and recovery from catastrophic failures***.** ***Learning Outcomes:*** *A learner will be able to* *LO6.1: Apply concepts of serializability, recoverability to ensure integrity and consistency of database. (P.I.-1.3.1)* *LO6.2: 7Apply transaction, concurrency control, and recovery concepts to foster lifelong learning in database management. (P.I.-1.4.1, P.I.-11.2.2)*	**9-11**
	Course Conclusion A DBMS course equips professionals with crucial skills in database architecture, optimization, and resilience, essential for successful data projects, industry adaptability, and innovative research, enhancing overall competence in data management.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.1.1	Apply the knowledge of discrete structures, linear algebra, statistics and numerical techniques to solve problems
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science and engineering. (modified PI).
2.1.1	Evaluate problem statements and identifies objectives.
2.3.2	Identify design constraints for required performance criteria.
2.1.2	Identify processes/modules/algorithms of a computer-based system and parameters to solve a problem.
3.2.2	Able to produce potential design solutions suited to meet functional requirements. (modified PI).
3.4.3	Able to verify the functionalities and validate the design.
11.2.1	Identify historic points of technological advance in engineering that required practitioners to seek education in order to stay current.
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field.

Course Outcomes: A learner will be able to -

1. Identify the purpose of database management system and its operational details. (LO1.1, LO1.2)
2. Design ER and EER diagram for real life applications. (LO2.1, LO2.2, LO2.3, LO2.4, LO2.5, LO2.6)
3. Construct relational model and formulate relational algebra queries. (LO3.1, LO3.2, LO3.3, LO3.4, LO3.5, LO3.6)
4. Analyse the query statement and formulate SQL queries.(LO4.1, LO4.2, LO4.3, LO4.4)
5. Apply the concept of normalization to enhance relational database design. (LO5.1,LO5.2)
6. Apply the concept of transaction management, concurrency control and Database recovery. (LO6.1,LO6.2)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC304.1	3	-	-	-	-	-	-	-	-	-	-
CEPCC304.2	3	3	3	-	-	-	-	-	-	-	-
CEPCC304.3	3	3	3	-	-	-	-	-	-	-	-
CEPCC304.4	3	3	-	-	-	-	-	-	-	-	2
CEPCC304.5	3	-	-	-	-	-	-	-	-	-	-
CEPCC304.6	3	-	-	-	-	-	-	-	-	-	2
Average	**3**	**3**	**3**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**2**

Text Books:

1. Database System Concepts, Korth, Silberchatz, Sudarshan, 6th Edition, McGraw Hill.
2. Fundamentals of Database Systems, Elmasri and Navathe, 5th Edition, Pearson Education.
3. Database Management Systems, Raghu Ramkrishnan and Johannes Gehrke, TMH.

Reference Books:

1. Database Systems Design, Implementation and Management, Peter Rob and Carlos Coronel, 5th Edition, Thomson Learning
2. SQL and PL/SQL for Oracle 10g, Black Book, Dr. P. S. Deshpande, Dreamtech Press.
3. Database Management Systems, G. K. Gupta, 2012, McGraw Hill.

Other Resources:

1. NPTEL Course: Database Management System, IIT, Kharagpur by Prof. Partha Pratim Das, Prof. Samiran Chattopadhyay, Prof. Kaushi Dutta :-Web Link-https://nptel.ac.in/courses/106105175.
2. NPTEL Course: Database Management System, IIT, Kharagpur by Prof. Partha Pratim Das, Web link- https://onlinecourses.nptel.ac.in/noc19_cs46/preview.
3. MOOC Course: DBMS, Web link- https://www.mooc-list.com/tags/dbms.

IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

Assignment on live problems/ case studies, wherein problems are given prior. Students are expected to research and collect required resources. They can use the resources and solve the problem on assigned date and time in Institute premises in presence of faculty member: 10 Marks.

Think Pair share worksheet : 05 Marks.

Regularity and active participation: 05 Marks.

2. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20%-30% weightage, and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
LBC	CELBC301	DATA STRUCTURES LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Examination (ESE)**	**Total**
25	25	50

Pre-requisite:

1. ESL103- Programming Laboratory -I (C)

Program Outcomes addressed:

1. PO3: Design/development of solutions
2. PO4: Conduct investigations of complex problems
3. PO5: Engineering tool usage
4. PO8: Individual and Collaborative Team work
5. PO9: Communication

Course Objectives:

1. To guide learners to implement various data structures using suitable programming language.
2. To guide learners in distinguishing various data structures.

Module	Detailed Contents	Hrs.
	Course Introduction Introduction to the data structure laboratory encompasses explaining the significance of data structures, tools used and emphasizing hands-on learning.	
01.	***Learning Objective:*** *Expected to implement linear data structure using primitive data types.*	**06**
	Laboratory Exercises: Problem statements based on- 1. Stack implementation 2. Queue implementation 3. Singly-linked list implementation	
	Self-Learning Topics: *Doubly linked list implementation* ***Learning Outcomes:*** *A learner will be able to* *LO1.1: Select the appropriate procedure for the given problem (P.I.-4.1.2)* *LO1.2: Develop functions to carry out various operations of the data structure (P.I.-4.2.1)* *LO1.3: Integrate different functions to carry out various operations related to that data structure. (P.I.-3.4.2)* *LO1.4: Implement data structure using suitable software tools and present the results (P.I.-5.1.2, P.I.- 8.2.1, P.I.-9.1.2)*	

02.	***Learning Objective:*** *Expected to implement data structures using linked list.*	**06**
	Laboratory Exercises: Problem statements based on- 1. Stack using the Linked List Implementation 2. Queue using the Linked List Implementation	
	Self-Learning Topics: *Implement circular queue using circular Linked list.* ***Learning Outcomes:*** *A learner will be able to* *LO2.1: Select the appropriate procedure for the given problem (P.I.-4.1.2)* *LO2.2: Develop functions to carry out various operations of the data structure (P.I.-4.2.1)* *LO2.3: Integrate different functions to carry out various operations related to that data structure. (P.I.-3.4.2)* *LO2.4: Use suitable software tool to implement data structure using linked list and present the results (P.I.-5.1.2, P.I.-8.2.1, P.I.-9.1.2)*	
03.	***Learning Objective:*** *Expected to implement tree data structure*	**06**
	Laboratory Exercises: Problem statements based on- 1. Binary Tree Implementation 2. Binary Search Tree Implementation	
	Learning Outcomes: *A learner will be able to* *LO3.1: Select the appropriate procedure for the given problem (P.I.-4.1.2)* *LO3.2: Develop functions to carry out various operations of the data structure (4.2.1)* *LO3.3: Integrate different functions to carry out various operations related to that data structure. (P.I.-3.4.2)* *LO3.4: Implement data structure using suitable software tools and present the results (P.I.-5.1.2, P.I.-8.2.1, P.I.-9.1.2)*	
04.	***Learning Objective:*** *Expected to implement Graph Traversal Techniques.*	**06**
	Laboratory Exercises: Problem statements based on-	
	1. DFS traversal techniques Implementation 2. BFS traversal techniques Implementation	
	Learning Outcomes: *A learner will be able to* *LO4.1: Select the appropriate procedure for the given problem (P.I.-4.1.2)* *LO4.2: Develop functions to carry out various operations of the data structure (P.I.-4.2.1)* *LO4.3: Integrate different functions to carry out various operations related to that data structure. (P.I.-3.4.2)* *LO4.4: Implement data structure using suitable software tools and present the results (P.I.-5.1.2, P.I.-8.2.1, P.I.-9.1.2)*	
05.	***Learning Objective:*** *Expected to implement hash tables*	**06**
	Laboratory Exercises: Problem statements based on-	
	1. Linear probing Implementation 2. Quadratic probing Implementation	

	Learning Outcomes: *A learner will be able to* *LO5.1: Select the appropriate procedure to perform the operations of hashing (4.1.2)* *LO5.2: Develop functions to carry out various operations. (4.2.1)* *LO5.3: Integrate different functions to carry out various operations. (3.4.2)* *LO5.4: Implement the operations using suitable software tools and present the results (5.1.2, 8.2.1, 9.1.2)*	
		Total **30**

Performance Indicators:

P.I. No.	**P.I. Statement**
3.4.2	Able to implement and integrate the modules.
4.1.2	Able to choose appropriate procedure/algorithm (modified PI).
4.2.1	Design and develop appropriate procedures/functions based on the programming objectives (modified PI).
5.1.2	Create/adapt/modify/extend tools and techniques to solve engineering problems.
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents.

Course Outcomes: A learner will be able to-

1 Select an appropriate data structure for the given problem.*(LO1.1, LO2.1, LO3.1, LO4.1)*

2 Develop procedures to carry out various operations of the data structure. *(LO1.2, LO2.2, LO3.2, LO4.2)*

3 Implement procedures and data structures using software tools. *(LO1.3, LO1.4, LO2.3, LO2.4, LO3.3, LO3.4, LO4.3, LO4.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC303.1	-	-	-	2	-	-	-	-	-	-	-
CEPCC303.2	-	-	-	2	-	-	-	-	-	-	-
CEPCC303.3	-	-	-	-	2	-	-	2	2	-	-
Average	**-**	**-**	**-**	**2**	2	-	-	2	2	-	-

Text Books:

1. Data Structures using C, Reema Thareja, 2nd Edition, 2014, Oxford Press.

Reference Books:

1. Data Structures using C and C++, Rajesh K Shukla, 1st Edition, 2009, Wiley-India.

Other Resources:

1. NPTEL Course: Data Structures and Algorithms by Prof. Naveen Garg, Department of Computer Science and Engineering Department, IIT Delhi: -Web link- https://nptel.ac.in/courses/106/102/106102064/

CONTINUOUS ASSESSMENT (25 Marks)

Suggested breakup of distribution

Practical Exercises- 10 Marks

Internal Assessment-

Practical Test – 10 Marks

Evaluation of the conceptual, problem solving and programming skills of each student will be assessed based on their approach towards problem solving, implementation of concepts and execution of the task using a software tool.

Regularity and active participation - 5 Marks.

END SEMESTER EXAMINATION (Practical/Oral Examination) (25 Marks)

Students will be assessed based on three parameters:

- Concept/Algorithmic knowledge
- Practical programming knowledge
- Oral

- Students will be randomly allocated a program from the list of laboratory exercises and will be asked to write appropriate algorithm for the same. The algorithm is checked by the examiners (Internal and External) and weightage for this is 05 Marks.

Then the student will be allowed to start with the implementation of the program.

- Students will be allocated 1 hour to complete the execution. The program is then checked by both the examiners for its correctness. The weightage of the program implementation is 10 Marks.

- Students will then be appearing for Oral in front of both Internal and External examiners. The weightage of Oral will be of 10 Marks.

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
LBC	CELBC302	DATABASE LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Examination**	**Total**
25	25	50

Pre-requisite : *Nil*

Program Outcomes addressed :

1. PO3: Design/Development of Solutions
2. PO4: Conduct investigations of complex problems
3. PO5: Engineering Tool Usage
4. PO7: Ethics
5. PO8: Individual and Collaborative team work
6. PO9: Communication

Course Objectives :

1. To introduce conceptual design and development of relational model.
2. To introduce the basics of SQL and formulate queries.
3. To familiarize learners with the basic functions of transaction processing.

Module	Detailed Contents	Hrs
	Course Introduction Database Lab course aims to provide students with hands-on experience in designing, implementing, and managing databases. The course usually covers fundamental concepts of database management systems (DBMS) along with practical exercises. This lab enables the students to practice the concepts learnt in the subject DBMS by developing a database. The topics that will be covered throughout the semester are: Introduction to SQL, Relational Database Design, Normalization, Database Querying and Manipulation, Transaction Management, Controlled access to database.	
01.	***Learning Objective:*** *Expected to identify design tool for modeling an application's data requirements and designing database schemas as a team.*	**06**
	Laboratory Exercises: Design of ER /EER model	
	Use conceptual tool to sketch an application's data requirements and design database schema. Laboratory Exercises list 1. Identify the real-world problem, draft the problem statement and derive entities, attributes and their associations. 2. Design ER /EER diagrams for the identified problem statement.	

	3. Construct Relation model of the ER/EER.	
	Learning Outcomes : *A learner will be able to* *LO1.1:Formulate problem statements for real life applications. (P.I.- 3.1.1)* *LO1.2:Design conceptual model for the identified problem statement. (P.I.- 3.4.1)* *LO1.3:Select appropriate mapping rules. (P.I.- 4.1.2)* *LO1.4:Construct Relational model from ER and EER diagram. (P.I.- 4.3.3)* *LO1.5:Use a tool to sketch the designed schema. (P.I.- 5.1.1)* *LO1.6::Test correctness of designed schema (P.I.- 5.2.2)*	
02.	***Learning Objective:*** *Expected to develop database and tables with reference to the designed schema for the selected real life application and apply different constraints on database.*	**08**
	Laboratory Exercises : DDL & DML With reference to the database schema, use structured query language to create database & tables with different constraints and populate them. Laboratory Exercises list 1. Implementation of DDL commands of SQL • Create table • Alter table • Drop Table 2. Implementation of DML commands of SQL • Insert • Update • Delete 3. Implementation of different types of constraints	
	Learning Outcomes : *A learner will be able to* *LO2.1:Identify software tool/language to create database. (P.I.- 5.1.1)* *LO2.2Choose appropriate DDL and DML commands to create database and tables and populate them. (P.I.- 4.1.2)* *LO2.3: Represent database to check its correctness (P.I.- 4.3.3)* *LO2.4Demonstrate the use of selected tools to represent the output of DDL and DML(P.I.- 5.2.2)*	
03.	***Learning Objective:*** *Expected to formulate appropriate SQL statements for querying a database to retrieve useful information from the database*.	**08**

	Laboratory Exercises: Formulate queries for information retrieval Apply different filters (using where clause and nested queries), built-in functions to retrieve data from the database and use views on relational database, access control privileges for database correctness. Laboratory Exercises list 1. Implementation of different types of function ▪ Number function ▪ Aggregate Function ▪ Character Function ▪ Date Function 2. Implementation of different types of operators ▪ Arithmetic Operators ▪ Logical Operators ▪ Comparison Operator ▪ Special Operator (LIKE, IN, EXISTS, BETWEEN ▪ ROWNUM/ROW_NUMBER()) 3. Implementation of Set Operation 4. Implementation of different types of Joins ▪ Inner Join ▪ Outer Join ▪ Natural Join, etc. 5. Implementation of ▪ Group By & having clause ▪ Order by clause 6. Implementation of Sub queries. 7. Implementation of Views. 8. Apply DCL commands ▪ GRANT ▪ REVOKE 9. Apply TCL commands ▪ Rollback ▪ Commit ▪ Savepoint ***Learning Outcomes :*** *A learner will be able to* *LO3.1 Use a software tool to retrieve required information from the database (P.I.- 5.1.1)* *LO3.2Select and apply different filters to fetch appropriate data from the database (P.I.-4.1.2)* *LO3.3Represent the filtered output. (P.I-.4.3.3)* *LO3.4Check the correctness of fetched information. (P.I.- 5.2.2)*	
04.	***Learning Objective:*** *To identify the requisite SQL statements for implementing stored procedures and functions as well as triggers, to manage events within a relational database.*	**08**
	Laboratory Exercises: Procedure, Functions & Triggers Implement stored procedures and functions within a relational database, alongside triggers designed to automate the execution of predefined actions or tasks in response to specific events or conditions Laboratory Exercises list 1. Implementation of procedures 2. Implementation of functions 3. Implementation of triggers	

	Learning Outcomes: *A learner will be able to* *LO4.1Use a software tool to write procedural code (such as loops, conditions, and variables) within the SQL commands. (P.I.- 5.1.1)* *LO4.2Use and implement procedures and functions and triggers to perform database operations. (P.I.- 4.1.2)* *Represent the output of database operations in tabular format. (P.I-4.3.3)* *LO4.3Verify the correctness and functionality of procedures, functions, and triggers within the database system. (P.I.- 5.2.2)* *LO4.4Demonstrate foundational data analysis tasks, incorporating fundamental database design ethics, producing clear, well-constructed written documents, and applying acquired knowledge of database concepts for creating mini project. (P.I- 5.1.2,5.2.2, 7.1.1, 8.2.1, 8.3.1, 9.1.2, 9.1.3)*	
	Minimum 2 or 3 Laboratory Exercises from each module, and total at least 10 Laboratory Exercises and a mini project	
	Total	**30**

Performance Indicators:

P.I. No.	**P.I. Statement**
3.1.1	Able to define a precise problem statement with objectives and scope.
3.4.1	Able to refine architecture design into a detailed design within the existing constraints.
4.1.2	Able to choose appropriate procedure/algorithm, dataset and test cases.
4.3.3	Represent data (in tabular and/or graphical forms) so as to facilitate analysis and explanation of the data, and drawing of conclusions
5.1.1	Identify modern engineering tools such as computer aided drafting, modeling and analysis; techniques and resources for engineering activities.
5.2.2	Demonstrate proficiency in using discipline-specific tools.
7.1.1	Identify situations of unethical professional conduct and propose ethical alternatives.
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills.
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts.
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents.
9.1.3	Create flow in a document or presentation - a logical progression of ideas so that the main point is clear.

Course Outcomes : A learner will be able to -

1. Design conceptual schema for a given real world problem. (LO1.1,LO1.2, LO1.3,LO1.4, LO1.5,LO1.6)
2. Create conceptual database schema using software tool/language. (LO2.1,LO2.2, LO2.3,LO2.4)
3. Formulate and implement simple queries to demonstrate database operations using software tool/language(LO2.1,LO2.2, LO2.3,LO2.4)
4. Implement various procedures, functions and triggers for a specific task in a database. (LO2.1,LO2.2, LO2.3,LO2.4)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CELBC302.1	-	-	3	3	3	-	-	-	-	-	-
CELBC302.2	-	-	-	3	3	-	-	-	-	-	-
CELBC302.3	-	-	-	3	3	-	-	-	-	-	-
CELBC302.4	-	-	-	3	3	-	2	3	3	-	-
Average	**-**	**-**	**-**	**3**	**3**	**-**	**2**	**3**	**3**	**-**	**-**

Text Books :

1. Korth, Slberchatz, Sudarshan, Database System Concepts, 6th Edition, McGraw Hill.
2. Elmasri and Navathe, Fundamentals of Database Systems, 6th Edition, Pearson education.
3. Raghu Ramkrishnan and Johannes Gehrke, Database Management Systems, TMH.

Reference Books :

1. SQL & PL / SQL for Oracle 11g Black Book, Dreamtech Press.
2. G. K. Gupta : "Database Management Systems", McGraw – Hill.
3. Peter Rob and Carlos Coronel, ―Database Systems Design, Implementation and Management‖, Thomson Learning, 9th Edition.

Other Resources :

1. SQL : https://www.tutorialspoint.com/sql/index.html.
2. SQL : https://www.w3schools.com/sql/
3. Microsoft SQL Server Document: https://learn.microsoft.com/en-us/sql/?view=sql-server-ver16.

CONTINUOUS ASSESSMENT (25 Marks)

Laboratory Exercises: 10 Marks

Internal Assessment: 10 marks

As a part of Internal Assessment, students will do course mini project. It is will be group activity. Students will be tasked to develop a small-scale database application with front end (optional) using database concepts, query language and programming language. This project-based assessment will require students to apply their knowledge and skills gained throughout the course to design and implement a functional system using database concepts, query language and programming language.

Course Project Rules:

1. **Group Size:** Groups of 3 to 4 members allowed.
2. **Project Proposal**: Detailed proposal with scope, objectives.
3. **Project Requirements:**
   - Develop using database concepts, query language and programming language.
   - Encouraged to use open-source database design tool, query language.
4. **Presentation:**
   - Present project features, challenges faced, and solutions.
   - Q&A session for evaluation.
5. **Evaluation Criteria:**
   - Adherence to requirements and objectives.
   - Design quality and organization.
   - Functionality, UI/UX (if applicable), and error handling.
   - Effective presentation and Q&A skills.

6. Documentation

Clear and comprehensive documentation is essential. This includes a project report detailing the design decisions, implementation details, challenges faced, and how they were addressed.

Regularity and Active Participation : 5 marks.

END SEMESTER EXAMINATION (Practical /Oral Exam) -25 Marks

Students will be assessed based on three parameters:

- Concept/SQL knowledge
- Practically design and implementation of queries
- Oral

1. Students will be allocated a set of questions or task with a problem statement. This involves scenarios where students need to design a database schema, implement queries to extract specific information, or perform data manipulation operations.

2. Students will be allocated 1 hour to complete the design part of question. The weightage for design part is 10 marks.

3. The design part will be checked by the examiners (Internal and External) following which student will be allowed to start with the implementation using database management systems (DBMS) such as MySQL or SQL Server or/and design tools.
4. Students will be allocated 1 hour to complete the implementation. Then output of queries will be checked by both the examiners for its correctness. The weightage of the queries implementation is 10 marks.
5. Students will then be appearing for Oral in front of both Internal and External examiners. The weightage of Oral will be of 5 Marks.

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
SBL	CESBL301	PYTHON PROGRAMMING LABORATORY	02

Examination Scheme		
Continuous Assessment	**End Semester Examination (ESE)**	**Total**
50	50	100

Pre-requisite:

1. ESL103 Programming Laboratory-I (C)
2. ESL205 Programming Laboratory-II (Java)

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO4: Conduct investigations of complex problems
4. PO5: Engineering tool usage
5. PO7: Ethics
6. PO8: Individual and Collaborative team work
7. PO9: Communication
8. PO11: Life Long Learning

Course Objectives:

1. To establish a strong foundation in Python programming.
2. To guide in achieving mastery of Object-Oriented Programming (OOP) concepts using Python programming.
3. To assist in applying advanced Python concepts effectively in real-world scenarios.
4. To encourage in exploring and engaging with practical applications of Python libraries and frameworks, fostering a dynamic learning environment.

Module	Detailed Contents	Hrs
	Course Introduction Python has emerged as one of the most popular programming languages globally, and its relevance continues to grow across various industries. Understanding and mastering Python not only enhances students' academic pursuits but also equips them with sought-after skills in the job market. It opens doors to diverse career paths and enables professionals to contribute meaningfully to innovative technological advancements across industries. Python Importance- Versatility and Simplicity, Rich Ecosystem, Industry Adoption and Community Support. History and evolution, Use cases and applications.	
01.	**Introduction to Python** *Learning Objective/s:* *Learners are expected to build on prior programming knowledge, adapt Python's unique features, and develop problem-solving skills.*	**10**

	Content: 1.1 Basic Syntax and Data Types - Variables and data types, Operators, Input and output, 1.2 Data Structures- list, tuple, set and dictionary 1.3 Understanding the Syntax Transition: From C and Java to Python	
	Laboratory Exercise/s 1. **Personalized Greeting Generator *-** Write a python code to generate Personalized Greeting. 2. **Calculating Areas of Geometric Figures *-** Write a python program to calculate areas of different geometric figures like circle, rectangle and triangle. 3. **Developing Conversion Utilities *:** Develop Converter such as Rupees to dollar, temperature convertor, inch to feet etc. 4. **Handling new Data Structure of Python *:** Write a python code for creating and manipulating data structures like list, tuple, set and dictionary	
	Learning Outcomes : *A learner will be able to* *LO1.1: Grasp Python programming concepts and develop program logic independently by leveraging their proficiency in languages such as C and Java from previous semesters in engineering. (P.I -1.4.1, 8.2.1)* *LO1.2: Learners will demonstrate a proficient understanding of basic Python concepts through practical code implementation to solve fundamental programming problems. They will also adhere to fundamental programming ethics, produce clear and well-constructed written documents of their results, and apply their knowledge of logic development to write Python program effectively (P.I- 5.1.2, 7.1.1, 9.1.2, 11.1.1.)*	
02.	**Control Flow and Functions** ***Learning Objective/s:*** *To reinforce understanding and application, learners will recall the syntax and usage of conditional statements (if, else, elif) and loops (for and while), adopt these structures in Python programming to control program execution based on conditions and iterations, and comprehend the concept and syntax of functions in Python to develop efficient and reusable code.*	**10**
	Content: 2.1 Conditional Statements: if, else, elif 2.2 Loops: for and while loop 2.3 Functions- Defining functions, Parameters and return values, Scope and lifetime of variables	

	Laboratory Exercise/s 1. **Generating multiplication table*:** Create a Python program that takes a numerical input from the user and generates its multiplication table. 2. **Generate a number analyzer (Menu driven)** *: Develop a Python program to analyze an input number, determining whether it is even or odd and checking for primality 3. **To-Do List Application (Lists and Functions):** Write a python code to create a to-do list application that allows users to add, delete, and view tasks. Utilize lists and functions for task management 4. **Number Guessing Game (Control Flow and Loops**) *: Develop a number guessing game where the program generates a random number, and the user has to guess it. Implement loops and conditional statements for user interaction. 5. **Interactive Calculator (Basic Arithmetic Operations using Functions) *:** Implement a simple calculator that takes user input and performs basic arithmetic operations (addition, subtraction, multiplication, division) 6. **Contact Book (Dictionaries and Functions):** Create a simple contact book application where users can add, edit, and delete contacts. Utilize dictionaries to store contact information and functions for management.	
	Learning Outcomes: *A learner will be able to* *LO2.1 Grasp Python programming concepts and develop program logic independently by leveraging their proficiency in languages such as C and Java from previous semesters in engineering. (P.I -1.4.1, 8.2.1)* *LO2.2 Identify and implement modular processes, modules, algorithms, and parameters in Python programming. Independently, they will comprehend the syntax and usage of conditional statements, loops, and functions to break down complex tasks into manageable modules, enhancing code organization and reusability. (P.I- 2.1.2, 8.2.1)* *LO2.3 Demonstrate a strong understanding of basic Python concepts through coding to solve fundamental programming problems, adhere to fundamental programming ethics during development, produce clear and well-constructed written documents of their results, and apply their knowledge of control structures and modular programming to write Python programs effectively. (P.I- 5.1.2,7.1.1,9.1.2,11.1.1)*	
03.	**File I/O and High Order Functions** ***Learning Objective/s:*** *Learners are expected to grasp fundamental concepts such as file handling in Python, understand the significance of proper file management encompassing error handling and resource cleanup, and comprehend high-order functions in Python, including lambda expressions, filter, map, and reduce functions.*	**10**
	Content: 3.1 File Handling- Reading and writing files, Exception handling 3.2 High Order Functions: lambda, filter, map reduce	

	Laboratory Exercise/s 1. **Extracting Words from Text File ***: Develop a Python program that reads a text file and prints words of specified lengths (e.g., three, four, five, etc.) found within the file. 2. **Finding Closest Points in 3D Coordinates from CSV**: Write a python code to take a csv file as input with coordinates of points in three dimensions. Find out the two closest points. 3. **Sorting City Names from File:** Write a python code to take a file which contains city names on each line. Alphabetically sort the city names and write it in another file. 4. **Fortune Forecaster (using lambda function)*:** Create a list of quotes. Using lambda function return a random string from a list of string. 5. **Co-ordinates Converter:** Write a python code using lambda function to perform the Cartesian to polar coordinates conversion. 6. **Neuron Simulator*:** Create a function Neuron that performs linear combination of two Vectors. Modify function Neuron using Map, Reduce and filter function. (For e.g. filters only positive values after the map operation before passing it to reduce operation). ***Learning Outcomes :*** *A learner will be able to* *LO3.1 Identify and implement modular processes, modules, algorithms, and parameters in Python programming. Independently, they will comprehend the syntax and usage of high ordered functions to break down complex tasks into manageable modules, enhancing code organization and reusability. (P.I- 2.1.2, 8.2.1)* *LO3.2 Learners will demonstrate proficiency in advanced Python concepts through coding to solve fundamental programming problems, adhere to fundamental programming ethics during development, produce clear and well-constructed written documents of their results, and apply acquired knowledge of File I/O and High-order Functions in advanced subjects like AI, ML, and NLP. (P.I- 5.1.2, 7.1.1, 9.1.2, 11.1.1)*	
04.	**Object-Oriented Programming (OOP) in Python** ***Learning Objective/s:*** *Learners are expected to recall foundational OOP concepts, understand class and object syntax, and apply this knowledge to create Python classes and objects, integrating attributes, methods, constructors, and destructors to depict real-world scenarios.*	**10**
	Content: 4.1 Introduction to OOP: Classes and objects, Encapsulation, inheritance, and polymorphism 4.2 Creating Classes and Objects: Class attributes and methods Constructor and destructor.	

	Laboratory Exercise/s 1. **College Festival Representation using OOPs**: Develop a python code to depict the ETAMAX/ FACES/or any other a College festival using OOPs Concept /or any real world scenario. 2. **Autonomous College Administrative Hierarchy *:** Develop a python code to depict an autonomous college administrative hierarchy using OOPs Concept or any other real world scenario. 3. **Quiz Game (Object-Oriented Programming):** Implement a quiz game where questions are objects of a class. Include features like scoring and multiple-choice questions. 4. **Text-based Adventure Game (Classes and Inheritance):** Design a text-based adventure game with different scenarios and outcomes. Use classes and inheritance for character types and game elements.	
	Learning Outcomes : *A learner will be able to* *LO 4.1: Define hierarchical systems and complex scenario for open ended problem using OOPs principles. (P.I 4.1.1)* *LO 4.2: Demonstrate the ability to select and apply suitable OOPs concepts, classes, and structures individually to effectively model and solve open-ended problems (P.I 4.2.1, 8.2.1)* *LO 4.3: Apply Python programming's OOPS concepts effectively to solve open-ended problems, while adhering to fundamental programming ethics, documenting results clearly, and preparing for future project development (P.I-5.1.2,7.1.1,9.1.2,11.1.1)*	
05	**Advanced Python Concepts** *Learning Objective/s:* *Learners are expected to master regular expressions for text manipulation and both frontend and backend development techniques in Python*	**10**
	Content 5.1 Regular Expressions: Pattern matching, Regex functions in Python 5.2 GUI Development using Tkinter 5.3 Database connectivity and networking **Laboratory Exercise/s** 1. **Data Extraction using Regular Expressions***: Build a program that will read text/csv file and extracts using expressions. 2. **Real-World Entity CRUD Operations***: Frontend-Backend Development- Developing a frontend and backend system to represent real-world entities (such as students, teachers, sportspersons, or scientists) and execute basic CRUD (Create, Read, Update, Delete) operations through the frontend interface.	

	Learning Outcomes: *A learner will be able to* *LO5.1: Specify the system's scope and requirements necessitating seamless connectivity between the frontend and backend components. (P.I -4.1.1)* *LO5.2: Select suitable GUI components and implement database operations tailored to a specific application's requirements, aiding, in application design and development. (P.I-4.1.2)* *LO5.3: Design and develop a frontend-backend systems using Tkinter for GUI development, enabling CRUD operations for real-world entities. The gain proficiency in advanced concept can be used for future Mini or Major project in higher classes. (P.I 4.2.1, 11.1.1)* *LO5.4: Apply advanced Python programming concepts to develop frontend, backend, and database connectivity for a small-scale system, while adhering to fundamental programming ethics and individually producing clear, well-constructed written documents. (P.I- 5.1.2, 5.2.2, 7.1.1,8.2.1,9.1,2)*	
06	**Python Libraries** *Learning Objective/s:* *Learners should explore various Python libraries to master Python programming, such as NumPy for efficient numerical computing and managing large datasets, Matplotlib for visualizing data with graphs, charts, and histograms, and Pandas for advanced data manipulation and analysis using Series and DataFrame structures.* **Content** 6.1 Introduction to Popular Libraries 6.2 NumPy for numerical computing, 6.3 Matplotlib for data manipulation 6.4 Pandas for data visualization **Laboratory Exercise/s** 1. **Performing Basic Data Exploration (Using NumPy, Pandas and Matplotlib) ***: Analyze a dataset (CSV file) using NumPy and Pandas. Calculate statistics, visualize data using Matplotlib, and draw insights from the analysis. 2. **Currency Converter (API Integration):** Build a currency converter that fetches the latest exchange rates from an API. Use requests library for API integration 3. **Password Generator (Random Module) *:** Develop a password generator that creates strong, random passwords. Use the `random` module to generate different combinations. ***Learning Outcomes:*** *A learner will be able to* *LO6.1 :Gain the ability to utilize appropriate libraries such as NumPy, Matplotlib, and Pandas for effective data collection, analysis, and visualization. (P.I-4.3.I)* *LO6.2: Develop skills to represent data in tabular and graphical forms using Python libraries such as Matplotlib and Pandas, aiding in data analysis, interpretation, and drawing conclusions. (P.I- 4.3.3)* *LO6.3: Identify modern engineering tools, techniques, and resources such as NumPy, Matplotlib, and Pandas for data analysis and visualization in Python programming. (P.I-5.1.1)* *LO6.4:The learner will demonstrate foundational data analysis tasks using Python programming, incorporating fundamental programming ethics, producing clear, well-*	**10**

	constructed written documents, and applying acquired knowledge of popular libraries for mini or major projects in subsequent classes. (P.I- 5.1.2,5.2.2, 7.1.1,8.2.1,9.1.2,11.1.1)	
	The Laboratory Exercises marked with an asterisk (*) are mandatory for each module and are designed to help students, to build a foundational understanding of Python programming. The Laboratory Exercises not marked with an asterisk are optional and intended to provide additional hands-on experience.	
	Total	**60**

Performance Indicators:

P.I. No.	**P.I. Statement**
1.4.1	Apply theory and principles of computer science and engineering to solve an engineering problem.
2.1.2	Identify processes/modules/algorithms of a computer-based system and parameters to solve a problem.
4.1.1	Define a problem for purposes of investigation, its scope and importance.
4.1.2	Able to choose appropriate procedure/algorithm, dataset and test cases.
4.2.1	Design and develop appropriate procedures/methodologies based on the study objectives.
4.3.1	Use appropriate procedures, tools and techniques to collect and analyze data.
4.3.3	Represent data (in tabular and/or graphical forms) so as to facilitate analysis and explanation of the data, and drawing of conclusions.
5.1.1	Identify modern engineering tools, techniques and resources for engineering activities.
5.1.2	Adapt the tools and techniques to solve engineering problems.
5.2.2	Demonstrate proficiency in using discipline-specific tools.
7.1.1	Identify situations of unethical professional conduct and propose ethical alternatives.
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills.
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents.
11.1.1	Describe the rationale for the requirement for continuing professional development.

Course Outcomes: A Learner will able to-

1. Demonstrate the proficiency in python programming (LO 1.1, LO1.2, LO2.1,LO2.2,LO2.3, LO3.1,LO3.2)
2. Demonstrate the ability to apply OOPs concepts in Python programming to develop solutions for real-world problems (LO4.1,LO4.2,LO4.3)
3. Design and develop GUI with backend connectivity for specific applications using advanced Python programming skills.(LO 5.1,LO5.2,LO5.3,LO5.4)
4. Investigate and apply popular Python libraries to conduct efficient data handling tasks. (LO 6.1, LO6.2,LO6.3,LO6.4)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CESBL301.1	2	2	-	-	2	-	2	2	2	-	2
CESBL301.2	-	-	-	2	2	-	2	2	2	-	2
CESBL301.3	-	-	-	3	3	-	2	2	2	-	2
CESBL301.4	-	-	-	3	3	-	2	2	2	-	2
Average	**3**	**2**	**-**	**3**	**3**	**-**	**2**	**2**	**2**	**-**	**2**

Books :

1. Core Python Programming, Dr. R. Nageswara Rao, Second Edition, Dreamtech Press.
2. Beginning Python: Using Python 2.6 and Python 3.1. James Payne, Wrox Publication.
3. Python Programming, Anurag Gupta and G. P. Biswas, First Edition,McGraw-Hill Education.
4. Introduction to computing and problem-solving using python, Balagurusamy, First Edition, McGraw Hill Education.

Reference Books :

1. Learn Python the Hard Way, , Zed Shaw, Third Edition, Addison-Wesley.
2. Python Projects, Laura Cassell,Alan Gauld, First Edition, Wrox Publication.

Other Resources :

1. Python Tutorial: http://docs.python.org/release/3.0.1/tutorial/
2. Python for everybody specialization: https://www.coursera.org/specializations/python.

CONTINUOUS ASSESSMENT (50 Marks)

Suggested breakup of distribution

A. Laboratory Exercises**: 10 Marks**

B. Internal Assessment**: 15 Marks**

As a part of Internal Assessment, students will do course mini project as a team to inculcate teamwork [P. I. – 8.3.1]. Students will be tasked with developing a small-scale system using Python. This project-based assessment will require students to apply their knowledge and skills gained throughout the course to design and implement a functional system using Python programming language.

Course Project Rules in Python:

1. Group Size: Groups of 2 to 4 members allowed.

2. Project Proposal: Detailed proposal with scope, objectives.

3. Project Requirements:

- Develop using Python.
- Encouraged to use relevant libraries and show core concepts understanding.

4. Presentation:

- Present project features, challenges faced, and solutions.
- Q&A session for evaluation.

5. Evaluation Criteria:

- Adherence to requirements and objectives.
- Code quality, readability, and organization.
- Functionality, UI/UX (if applicable), and error handling.
- Effective presentation and Q&A skills.

C. Practical Tests**: 20 Marks**

Two practical tests will be conducted based on laboratory exercises.

1. The allocation of laboratory exercises for testing programming and problem-solving skills, with each student receiving two or more laboratory exercises.

2. Students will have a designated 2-hour timeframe for code development. After the first hour, an internal examiner will review the progress and evaluate the above skills.

3. During the practical assessment or at its conclusion, students will be queried to evaluate their conceptual understanding, ensuring comprehension.

D. Regularity and active participation**: 05 Marks**

END SEMESTER EXAMINATION (Practical/Oral Exam) (50 Marks)

For the End semester exams, practical examination will be conducted. The details of the end-semester evaluation are as follows. It will consist of three sections:

Section 1: Practical Examination (20 Marks)

This section will have practical exam based on the laboratory exercises conducted during the term. The assessment criteria will be similar to Internal Practical Test.

Section 2: Debugging and Output Prediction Exercise (20 Marks)

This section involves questions problems such as providing partial code segments with bugs and asking students to identify and correct the errors, predict the output of the corrected code, complete the code, identify the appropriate library etc. This option is designed to prepare students for placements or industry roles by testing their ability to debug and understand code in real-world scenarios.

Section 3: Oral (10 Marks)

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
MNP	CEMNP301	MINI PROJECT-1A	01

Program Outcomes addressed:

1. PO1 : Engineering knowledge
2. PO2 : Problem Analysis
3. PO3 : Design/Development of Solutions
4. PO4 : Conduct investigations of complex problems
5. PO5 : Engineering Tool Usage
6. PO6 : The Engineer and The World
7. PO7 : Ethics
8. PO8 : Individual & Collaborative team work
9. PO9 : Communication
10. PO10: Project Management & Finance
11. PO11: Life-long learning

Course Objectives

1. To familiarize students about available infrastructure at Department/Institute level, online resources, plagiarism, expectations from MP 1A and 1B, etc.
2. To guide students in identifying societal or research needs and formulating them into problem statements.
3. To facilitate problem-solving in group settings.
4. To apply basic engineering principles to address identified problems.
5. To foster self-learning and research skills.

Course Outcomes

At the end of the course, learner will be able to:

1. Identify problems based on societal or research needs and methodology for solving them.
2. Apply knowledge and skills to solve societal problems collaboratively.
3. Develop interpersonal skills necessary for teamwork.
4. Analyze, verify, and validate results effectively through various methodologies, including, test cases/benchmark data/theoretical/inferences/experiments/simulations, etc.
5. Evaluate the societal and environmental impacts of proposed solutions.
6. Adhere to standard engineering practices.
7. Excel in written and oral communication by technical report writing, oral presentation, and publishing results in
   - Research/white paper/article/blog writing/publication, etc.
   - Business plan for entrepreneurship product creation
   - Patent filing/copyright.
8. Gain technical competencies by participating in competitions, hackathons, etc.
9. Demonstrate lifelong learning capabilities through self-directed group projects.
10. Apply project management principles effectively.

Guidelines for the Mini Project

- At the beginning of semester-III, project guides are required to conduct around 4 hours' orientation sessions including following topics:
  - Familiarizing students about infrastructure available at Department/Institute level and how to use it.
  - How to identify societal problems and formulate project problem statement.
  - How to carry out literature survey.
  - What is plagiarism and what care needs to be taken while writing a report.
  - What is project report template and how it should be used.
  - What are expectations from mini-projects 1A and 1B.
- Mini project may be carried out in one or more form of following:
  Product preparations, prototype development model, fabrication of set-ups, laboratory experiment development, process modification/development, simulation, software development, integration of software (frontend-backend) and hardware, statistical data analysis, creating awareness in society/environment etc.
- Students must form groups of 3 to 4 members either from the same or from different departments.
- Groups should conduct surveys to identify needs and develop problem statements in consultation with faculty.
- An implementation plan in Gantt/PERT/CPM chart format covering weekly activities must be submitted.
- Each group must maintain a logbook to record weekly progress, to be verified by the faculty supervisor.
- Faculty input should emphasize guiding by faculty and self-learning by group members.
- Groups should propose multiple solutions, select the best one in consultation with the supervisor, and develop a working model.
- The solution to be validated with proper justification and report to be compiled in standard format of the Institute. Research papers, competition certificates may be submitted as part of annexure to the report.
- With the focus on self-learning, innovation, addressing societal/research/innovation problems and entrepreneurship quality development within the students through the Mini Projects, it is preferable that a single project of appropriate level and quality be carried out in two semesters by all the groups of the students.
- 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 a case-by-case basis.

In-Semester Continuous Assessment and End-Semester Examination Guidelines

- The Head of the Departments will assign a guide to each of the mini-projects and shall form a progress monitoring committee. The guide will carry out weekly monitoring of the project's progress. The committee shall carry out in-semester project evaluation based on presentations with a minimum of two evaluations per semester.
- Assessment will be based on individual contributions, understanding, and responses to questions asked.

- Continuous Assessment marks distribution in semester III (50 marks):
    - 0 5 marks for the Topic Approval Presentation in front of the progress monitoring committee
    - 15 marks for the Mid-Semester Progress Presentation in front of the progress monitoring committee
    - 25 marks for the Final Report & Presentation
    - 05 marks for Regularity and active participation

Semester III:

- Theoretical solution completion, including component/system selection/design of software solution and cost analysis.
- Two reviews will occur:
    - The first review will focus on finalizing the problem statement (topic approval).
    - The second review will focus on finalizing the proposed solution.

Course Type	Course Code	Course Name	Credits
HSSM	HSS301	PRODUCT DESIGN	02

Program Outcomes addressed:

1. PO2 : Problem Analysis
2. PO3 : Design/Development of Solutions
3. PO5 : Engineering Tool Usage
4. PO6 : The Engineer and The World
5. PO7 : Ethics
6. PO10: Project Management & Finance
7. PO11: Life-long learning

Course Objectives:

1. Understand the product design process and its user-centered principles.
2. Apply fundamental design principles to create innovative product designs.
3. Demonstrate proficiency in generating and evaluating design concepts through ideation techniques.
4. Evaluate and synthesize sustainable and user-centric design practices in product development.

Module	Details
01.	**Introduction to Product Design** Overview of product design process, Importance of user-centered design, Design thinking methodologies, Case studies of successful product designs, Introduction to design tools and software (e.g., Sketch, Adobe XD)
02.	**Design Principles and Fundamentals** Understanding design principles (e.g., balance, hierarchy, contrast), Human factors in design (ergonomics, anthropometrics), Material selection and properties, Basics of aesthetics and styling, Hands-on exercises in sketching and prototyping
03.	**Concept Generation and Ideation** Techniques for brainstorming and idea generation, Sketching andvisualization techniques, Developing design briefs and specifications, Evaluating and selecting design concepts, Rapid prototyping methods(e.g., 3D printing, CNC machining)

04.	**Renewable energy & Energy efficiency** Detailed overview of the product development lifecycle, Design for manufacturability (DFM) considerations, Cost estimation and budgeting, Collaborative design tools and project management Regulatory and compliance requirements (e.g., safety standards)
05.	**User Experience (UX) Design** Understanding user needs and behaviour, Usability testing and feedback gathering, Wireframing and prototyping for digital products Iterative design process, Accessibility and inclusive design principles.
06.	**Sustainability in Product Design** Environmental impact assessment in product design, Sustainable materials and manufacturing processes, Design for disassembly and recycling, Circular economy principles Case studies of eco-friendly product designs
Total No. of Hours: 30	

Course Outcomes: A learner will be able to-

1. Apply design thinking methodologies effectively to solve design problems.
2. Demonstrate proficiency in utilizing design tools and techniques for product development.
3. Communicate and collaborate effectively for interdisciplinary teamwork.
4. Create functional and aesthetically pleasing product designs.
5. Integrate sustainable and user-centric design principles into product development processes.

Text Books:

1. "Product Design and Development" by Karl T. Ulrich and Steven D. Eppinger, published by McGraw-Hill Education; 7th edition (January 25, 2021).
2. "Engineering Design: A Project-Based Introduction" by Clive L. Dym and Patrick Little, published by Wiley; 4th edition (August 26, 2015).
3. "Universal Principles of Design" by William Lidwell, Kritina Holden, and Jill Butler, published by Rockport Publishers; Revised and updated edition (January 1, 2010).

Reference Books:

1. "Sketching: Drawing Techniques for Product Designers" by Koos Eissen and Roselien Steur, published by BIS Publishers; 2nd edition (March 1, 2011).
2. "Materials and Design: The Art and Science of Material Selection in Product Design" by MikeAshby and Kara Johnson, published by Butterworth-Heinemann; 3rd edition (October 10, 2014).
3. "The Design of Everyday Things" by Don Norman, published by Basic Books; Revised and expanded edition (November 5, 2013).

Other Resources:

1. NPTEL Course: Product Design and Development, Prof. Inderdeep Singh, IIT Roorkee
   Weblink:- https://onlinecourses.nptel.ac.in/noc21_me83/preview
2. NPTEL Course: Product Design and Innovation, By Prof. Supradip Das, Prof. Swati Pal,Prof. Debayan Dhar, IIT Guwahati, IIT Guwahati,
   Web link- https://onlinecourses.nptel.ac.in/noc21_de01/preview

Course Type	Course Code	Course Name	Credits
PCC	CEPCC405	ENGINEERING MATHEMATICS -IV	03+01*

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20 +25*	30	50	1.5	2	125

**** Tutorial***

Pre-requisite : Nil

Program Outcomes addressed :

1. PO1: Engineering knowledge
2. PO2: Problem analysis

Course Objectives :

1. To provide the basic knowledge on the concepts of Mathematics in the field of Engineering.
2. To build a foundation to the methodology necessary for solving problems by applying the knowledge of Mathematics to the field of Engineering.

Module	Details	Hrs.
	Course Introduction Engineering Mathematics IV aims to equip students with the foundational knowledge and skills necessary for analyzing uncertainty and making informed decisions in engineering contexts. This course has many applications in Information Technology For example 1) Application of in Probability in Machine Learning, NLP part of speech Hidden Markov Model. 2) Application of in Correlation and Regression in Deep learning. 3) Application of Statistics in hypothetical or inferential learning. 4) Application of Algebraic Structure in Cyber Security and Cryptography.	**01**
01.	**Probability Theory and Random Variable** ***Learning Objective/s:*** *The learner will be able to analyze random variables using the basic theory of probability and will be able to apply various mathematical techniques in determining probability functions.*	**6-8**
	Contents: Conditional Probability , Bayes Theorem, Total Probability Theorem, Definition of Random Variable. Types of Random Variable: Discrete and Continuous, Probability Mass and Density Function.	

	Self-Learning Topics: *Cumulative Distribution and Density Function*	
	Learning Outcomes : *A learner will be able to* 1. *Identify independent sets and disjoint sets and use its knowledge in the context of conditional probability. (P.I.-2.1.3)* 2. *Apply mathematical techniques of union, intersection and addition of sets, numbers for finding probabilities of events using Bayes' Theorem and Total Probability Theorem. (P.I.-1.1.1)* 3. *Identify if a given Random variable is Discrete or continuous in nature using existing definitions and formulas from Probability. (P.I.-2.1.2)* 4. *Apply mathematical techniques of integration and summation for finding Expectation, Variance, Probability density function and Probability distribution function. (P.I.-1.1.2)*	
02.	**Probability Distribution** ***Learning Objective/s:*** *Learner will be able to analyse and identify standard probability distribution functions and apply the knowledge of distribution for finding probabilities of various events.*	**6-8**
	Contents: Measures of Central Tendency and Dispersion, Binomial distribution, Poisson Distribution , Fitting of a Poisson Curve, Gaussian Distribution, Normal Distribution (Standard Normal distribution , Reverse problem of Normal distribution)	
	Self-Learning Topics: *Joint Probability Distribution*	
	Learning Outcomes : *A learner will be able to* 1. *Apply mathematical techniques of exponents, algebra and basic probability for finding the probabilities of various events using Binomial, Poisson and Normal Distribution. (P.I.-1.1.1)* 2. *Apply the advance mathematical techniques of statistics to find the probabilities the random variable (P.I. -1.1.2)* 3. *Identify the area under a Standard Normal Curve (bounded or unbounded) and use its knowledge in the context of Normal Distribution. (P.I.-2.1.3)* 4. *Identify whether Poisson distribution or Normal Distribution is applicable to a given problem using basic definitions of distribution and the data inferred from the problem. (P.I.-2.1.1)*	
03.	**Sampling Theory-I** ***Learning Objective/s:*** *Learner will be able to formulate the null hypothesis and apply parametric testing to test the hypothesis.*	**5-7**
	Contents: Introduction to Sampling Theory, Testing of Hypothesis, level of significance, Critical region, One tailed and two tailed test, Students' t-distribution. Test significance of large samples test: single mean, difference between the two means,	

	Self-Learning Topics: *sampling distribution of proportions*	
	Learning Outcomes: *A learner will be able to* 1. *Identify and test the hypothesis of significance difference between the parameter and the statistics (P.I.-2.2.2)* 2. *Identify and test the hypothesis of significance difference between the two means (P.I.-2.2.4)* 3. *Identify and apply appropriate test to be used to test the given hypothesis.(P.I.-2.1.1)* 4. *Determine the test statistics using the appropriate formula (P.I.-1.1.1)* 5. *Determine frequencies fitting a particular probability distribution(P.I.-1.1.2)*	
04.	**Sampling Theory-II** ***Learning Objective/s:*** *Learner will be able to formulate the hypothesis and apply non-parametric testing to test the it.*	**5-7**
	Contents: Chi-square test: Test of goodness of fit , Independence of attributes (Contingency table), distribution of sample variance F-test , significant difference between variances of two Samples.	
	Self-Learning Topics: *Yate's Correction, ANOVA*	
	Learning Outcomes : *A learner will be able to* 1. *Identify and test the hypothesis of test the independence of attributes (P.I.-2.2.2)* 2. *Identify and test the hypothesis of significance difference between the two variances (P.I.-2.2.4)* 3. *Determine the expected frequencies of the assumption. (P.I.-1.1.1)* 4. *Determine the expected frequencies of the contingency table(P.I.-1.1.2)*	
05.	**Correlation and Regression** ***Learning Objective/s:*** *Learner will be able to analyze the mathematical dataset given and apply techniques of correlation and regression to identify the relationships between variables from the dataset.*	**7-9**
	Contents: Correlation, Karl Pearson's coefficients of correlation(r), Spearman's Rank correlation coefficient (R): Repeated Rank, Non-repeated rank, Regression, Line of regression, Curve fitting: Linear and Second-Degree Curves.	
	Self-Learning Topics: *Fitting of an exponential Curve*	
	Learning Outcomes : *A learner will be able to* 1. *Identify whether Karl Pearson's or Spearman's coefficient of correlation is to be used in establishing relationship between two variables depending on the dataset given. (P.I.- 2.1.3)* 2. *Apply basic mathematical techniques from algebra in finding the lines of regression and regression coefficients. (P.I.-1.1.1)* 3. *Apply Least Square Method to fit a particular to the given data (P.I.-1.1.2)* 4. *Identify whether a linear degree curve or a quadratic degree curve is to be fit for the given data set based on the knowledge of Curve Fitting (P.I.-2.2.2)*	

06.	**Algebraic Structure** ***Learning Objective/s:*** *The learner will be able analyze the Algebraic Structure using the basic properties.*	**7-9**
	Contents: Rings, Integral domain, Fields, Ring Homomorphism, Ring Isomorphism	
	Self-Learning Topics: *Orthonormal basis, Basis and Dimension.*	
	Learning Outcomes : *A learner will be able to* 1. *Apply mathematical operations defined on algebraic structures like Rings, Integral domain and Field and demonstrating closure under these operations. (P.I.-1.1.1)* 2. *Identify substructures within algebraic systems and the concept of homomorphism between them. (P.I.-2.1.2)* 3. *Apply the properties of homomorphism and one-one proves that the homomorphism is an isomorphism(P.I. 1.1.2)* 4. *Identify and characterize various algebraic structures based on their properties.(P.I.-2.2.2)*	
	Course Conclusion Engineering Mathematics plays an important role in providing the analytical tools necessary for designing, analyzing, and optimizing various electronic systems and communication networks.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	**P.I. Statement**
1.1.1	Apply mathematical techniques such as calculus, linear algebra, and statistics to solve problems.
1.1.3	Apply advanced mathematical techniques such as integral and differential equations to describe/solve/construct a mathematical model of a system.
2.1.1	Articulate problem statements and identify primary objectives and key constraints.
2.1.2	Identify engineering systems, variables, and parameters to solve the problems.
2.1.3	Identify the mathematical, engineering and other relevant knowledge that applies to a given problem.
2.2.2	Identify, assemble and evaluate information and resources.
2.2.4	Compare and contrast alternative solution processes to select the best process.

Course Outcomes :

1. Analyse random variables and apply the concepts of probability for getting the spread of data.*(LO 1.1, LO 1.2, LO 1.3 , LO 1.4)*
2. Analyse the mathematical problem given and apply the concepts of distribution in finding probabilities. *(LO 2.1, LO 2.2, LO 2.3 , LO 2..4)*
3. Apply sampling theory principles and techniques to real-world research problems in various fields and interpret the result. *(LO 3.1, LO 3.2, LO 3.3 , LO 3.4, LO 3.5, LO 4.1, LO 4.2, LO 4.3)*
4. Analyse and interpret the data using Correlation and Regression.(*LO 5.1, LO 5.2, LO 5.3 , LO 5.4)*
5. Apply the properties and Identify the Algebraic Structure. *(LO 6.1, LO 6.2, LO 6.3, LO 6.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC405.1	3	2									
CEPCC405.2	3	2									
CEPCC405.3	3	2									
CEPCC405.4	3	2									
CEPCC405.5	3	2									
Average	**3**	**2**									

Text Books :

1. Higher Engineering Mathematics by Grewal B. S. 38th edition, Khanna Publication 2005.
2. Advanced Engineering Mathematics by Kreyszig E. 9th edition, John Wiley.
3. A Text Book of Applied Mathematics Vol. I & II by P.N.Wartikar & J.N.Wartikar, Pune, Vidyarthi Griha Prakashan, Pune.

Reference Books :

1. Topics in Algebra by Herstein.
2. Fundamentals of Probability and Statistics V. K Kapoor and S. C. Gupta,. S . Chand, Publications.
3. Advanced engineering mathematics, H.K. Das, S . Chand, Publications.

Other Resources :

1. NPTEL Course: NOC: Probability for Computer Science IIT Kanpur: Prof Nitin Saxena:-Web link- https://nptel.ac.in/courses/106104233/
2. NPTEL Course: Sampling Theory, IIT Kanpur: Prof Shalabh :-Web link- https://nptel.ac.in/courses/111104073.
3. NPTEL Course::- NOC: Algebra -I IMSc, Prof S Vishwanath, Prof Amritanshu Prasad Web link- https://nptel.ac.in/courses/111106137.

IN-SEMESTER ASSESSMENT (75 MARKS)

1. Continuous Assessment-Theory (20 Marks)

Suggested breakup of distribution

One MCQ test as per GATE exam pattern / level: 05 Marks.

One Class test: 05 Marks.

One Team-Pair-Solo activity: 05 Marks.

Regularity and active participation: 05 Marks.

2. Continuous Assessment - Tutorial (25 Marks)

Suggested breakup of distribution

Tutorial Assignments and Class tests 20 Marks.

Students must be encouraged to write at least 6 class tutorials. At least 6 Class tests will be conducted based on class tutorials on entire syllabus. Each class tests carries 20 Marks. Average will be taken of all class tests**.**

Regularity and active participation: 05 Marks.

3. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50 % syllabus.

END SEMERSTER EXAMINATION (50 MARKS)

End semester examination will be based on syllabus coverage up to the Mid semester examination (MSE) carrying 20%-30% weightage and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	CEPCC406	ANALYSIS OF ALGORITHMS	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite:

1. ESL103- Programming Laboratory- I (C)
2. BSC101- Engineering Mathematics - I

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO4: Conduct investigations of complex problems
4. PO11: Life-long learning:

Course Objectives:

1. To familiarize mathematical approaches to analyse the algorithms.
2. To familiarize the applications of algorithmic strategies to solve various problems.
3. To introduce the concept of time and space complexity for analysis.
4. To facilitate to compare different algorithmic approaches.

Module	Details	Hrs.
	Course Introduction Analysis of Algorithm: This course provides the foundation for analyzing time and space complexity for various algorithmic approaches. The course focuses on developing skills to assess the efficiency and performance of algorithms in problem-solving.	**01**
01.	**Introduction to Analysis of Algorithm** ***Learning Objective/s:*** *To familiarize the concept of time and space complexity of the algorithm.*	**4-6**
	Contents: 1.1 Performance Analysis, Asymptotic notations, Time and space complexity, Mathematical background for algorithm analysis, Definitions of P, NP, NP-Hard, NP-Complete 1.2 Analyzing time and space complexity of Iterative Algorithm- Insertion Sort, Selection Sort 1.3 Analyzing time complexity of Recursive Algorithm	
	Self-Learning Topics: *Randomized Algorithms*	

	Learning Outcomes: *A learner will be able to* *LO1.1 Apply algorithmic fundamentals and get familiarized with asymptotic notations to find the time and space complexity of an algorithm (P.I. - 1.3.1)* *LO1.2 Apply the various methods to find the complexity of iterative and recursive algorithm approaches (1.4.1)* *LO1.3 Compare iterative and recursive algorithmic approaches. (P.I. - 2.2.4)* *LO1.4 Analyze the time and space complexity of iterative and recursive algorithms (P.I. - 2.4.2)*	
02.	**Divide and Conquer Approach** ***Learning Objective/s:*** *To familiarize the time and space complexity of divide & conquer*	**6-8**
	Contents: 2.1 Divide and Conquer: General Method 2.2 Merge Sort, Quick sort, Performance analysis of sorting using iterative approach and divide and conquer approach. 2.3 Finding Minimum and Maximum, Performance analysis of iterative approach and divide and conquer approach to find minimum and maximum. 2.4 Binary Search, Performance analysis of Linear search and binary search.	
	Self-Learning Topics: *Strassen's Matrix Multiplication*	
	Learning Outcomes: *A learner will be able to* *LO2.1 Apply algorithmic fundamentals to perceive divide & conquer approach (P.I. -1.3.1)* *LO2.2Apply divide & conquer approach to solve various problems (P.I. -1.4.1)* *LO2.3Compare the divide & conquer and iterative algorithm approaches. (P.I. -2.2.4)* *LO2.4Analyze the time and space complexity divide & conquer approach (P.I. -2.4.2)* *LO2.5Choose appropriate procedure/algorithms with respect to the divide & conquer algorithmic approach in the current field of Computer Engineering. (P.I. -4.1.2, P.I. -11.2.2)*	
03.	**Greedy Approach** ***Learning Objective/s:*** *Students are expected to apply greedy approaches to solve various problems and analyze the time and space complexity of these approaches*	**8-10**
	Contents: 3.1 Greedy Algorithms – General Method 3.2 Single Source shortest path: Dijkstra's Algorithm 3.3 Knapsack Problem 3.4 Job sequencing with deadline 3.5 Minimum cost spanning tree algorithms: Prims and Kruskal's algorithm	
	Self-Learning Topics: *Optimal Randomized Algorithm*	

	Learning Outcomes: *A learner will be able to* *LO3.1Apply algorithmic fundamentals to perceive greedy approach (P.I. -1.3.1)* *LO3.2Apply the greedy approach to solve various problems (P.I. -1.4.1)* *LO3.3Compare greedy strategies for spanning tree (P.I. -2.2.4)* *LO3.4Analyze the time and space complexity of Greedy approach (P.I. -2.4.2)* *LO3.5Choose appropriate procedure/algorithms with respect to the greedy algorithmic approach in the current field of Computer Engineering. (P.I. -4.1.2, P.I. -11.2.2)*	
04.	**Dynamic Programming Approach** ***Learning Objective/s:*** *Students are expected to analyze the time and space complexity of dynamic programming approaches.*	**8-10**
	Contents: 3.1 Dynamic Programming- General Method 3.2 Multistage graphs 3.3 Single source shortest path: Bellman Ford Algorithm 3.4 All pair shortest path: Floyd Warshall Algorithm 3.5 Longest common subsequence 3.6 Travelling Salesman Problem 3.7 Knapsack problem: Comparison between greedy knapsack and dynamic programming knapsack 3.8 Comparison of Greedy approach and Dynamic approach	
	Self-Learning Topics: *Travelling salesperson problem, Assembly line scheduling*	
	Learning Outcomes: *LO4.1Apply algorithmic fundamentals to perceive dynamic programming approach (P.I. -1.3.1)* *LO4.2Apply dynamic programming approach to solve various problems (P.I. -1.4.1)* *LO4.3Compare the greedy, and dynamic programming strategies. (P.I. -2.2.4)* *LO4.4Analyze the time and space complexity dynamic programming approach (P.I. -2.4.2)* *LO4.5Choose appropriate procedure/algorithms with respect to the dynamic algorithmic approach in the current field of Computer Engineering. (P.I. -4.1.2, P.I. -11.2.2)*	
05.	**Backtracking approach** ***Learning Objective/s:*** *Students are expected to apply the backtracking to solve various problems and analyze the time and space complexity of these approach.*	**6-8**
	Contents: 5.1 Backtracking- General Method 5.2 N-queen problem 5.3 Sum of Subset 5.4 Graph Coloring 5.5 Knapsack Problem: Comparison between greedy, dynamic programming and backtracking approach	

	Self-Learning Topics: *Hamilton Cycle*	
	Learning Outcomes: *LO5.1Apply algorithmic fundamentals to perceive backtracking approach (P.I. -1.3.1)* *LO5.2Apply the backtracking approach to solve various problems (P.I. -1.4.1)* *LO5.3Compare the dynamic programming, greedy, and backtracking approaches (P.I. -2.2.4)* *LO5.4Analyze the time and space complexity backtracking approach (P.I. -2.4.2)* *LO5.5Choose appropriate procedure/algorithms with respect to the backtracking algorithmic approach in the current field of Computer Engineering. (P.I. -4.1.2, P.I. -11.2.2)*	
06.	**Branch & Bound Approach** ***Learning Objective/s:*** *Apply the branch & bound to solve various problems and analyze the time and space complexity of these approaches*	**5-7**
	Contents: 6.1 Branch & Bound: LC and FIFO branch & bound 6.2 15- puzzle Problem 6.3 Knapsack Problem: Comparison between greedy, dynamic programming, backtracking and branch and bound approach 6.4 Comparison of backtracking, and branch & bound	
	Self-Learning Topics: *Travelling Salesperson Problem* ***Learning Outcomes:*** *LO6.1Apply algorithmic fundamentals to perceive branch & bound approach (P.I. -1.3.1)* *LO6.2Apply the branch & bound approach to solve various problems (P.I. -1.4.1)* *LO6.3Compare the, greedy, dynamic programming, backtracking, and branch and bound approaches of algorithm. (P.I. -2.2.4)* *LO6.4Analyze the time and space complexity branch & bound approach (P.I. -2.4.2)* *LO6.5Choose appropriate procedure/algorithms with respect to the branch and bound algorithmic approach in the current field of Computer Engineering. (P.I. -4.1.2, P.I. -11.2.2)*	
	Course Conclusion It emphasizes the importance of applying efficient algorithms to solve problems and analyzing their performance in terms of time and space.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science and engineering to solve an engineering problem.
2.2.4	Compare and contrast alternative solutions/methods to select the best methods.
2.4.2	Analyze and interpret the results. (modified PI)
4.1.2	Able to choose appropriate procedure/algorithm.(modified PI)
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field.

Course Outcomes: A learner will be able to-

1. Apply algorithmic fundamentals and asymptotic notations to analyze and compare the time and space complexity of iterative and recursive algorithms. (LO1.1, LO1.2, LO1.3, LO1.4)
2. Apply divide and conquer algorithms to solve sorting and searching, analyze their complexities, and compare them with iterative approaches. (LO1.1, LO1.2, LO1.3, LO1.4)
3. Apply greedy and dynamic programming approaches to solve classical problems, analyze their complexities, and compare them. (LO2.1, LO2.2, LO2.3, LO2.4, LO2.5)
4. Apply backtracking and branch and bound strategies to solve complex problems, analyze their performance, and compare them with other algorithmic methods. (LO3.1, LO3.2, LO3.3, L31.4, LO3.5) (LO4.1, LO4.2, LO4.3, LO4.4, LO4.5)
5. Compare and contrast various algorithmic approaches to select the most appropriate method for a given problem. (LO1.1, LO1.2, LO1.3, LO1.4) (LO2.1, LO2.2, LO2.3, LO2.4, LO2.5) (LO3.1, LO3.2, LO3.3, L31.4, LO3.5) (LO4.1, LO4.2, LO4.3, LO4.4, LO4.5) . (LO5.1, LO5.2, LO5.3, LO5.4, LO5.5) (LO6.1, LO6.2, LO6.3, LO6.4, LO6.5)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC406.1	3	3	-	-	-	-	-	-	-	-	-
CEPCC406.2	3	3	-	2	-	-	-	-	-	-	2
CEPCC406.3	3	3	-	2	-	-	-	-	-	-	2
CEPCC406.4	3	3	-	2	-	-	-	-	-	-	2
CEPCC406.5	3	3	-	2	-	-	-	-	-	-	2
Average	**3**	**3**	**-**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**2**

Text Books:

1. Introduction to algorithms, T. H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, 3rd Edition, 2009, The MIT Press.
2. Fundamentals of computer algorithms, Ellis Horowitz, Sartaj Sahni, S. Rajsekaran, 2nd Edition, 2008, Universities Press.

Reference Books:

1. Algorithms, Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, 1st Edition, 2006, McGraw- Hill India.
2. Design Methods and Analysis of Algorithm, S. K. Basu, 2nd Edition, 2005, PHI.

Other Resources:

1. NPTEL Course: Data Structures and Algorithms by Prof. Naveen Garg, Department of Computer Science and Engineering Department, IIT Delhi :-Web link- https://nptel.ac.in/courses/106/102/106102064/
2. NPTEL Course: Design and Analysis of Algorithms By Prof. Madhavan Mukund, Department of computer science at Chennai Mathematical Institute, Web link- https://nptel.ac.in/courses/106106131.

IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

Numerical Assignment/s (min 20 problems): 05 Marks.

Class test based on above numerical assignment: 05 Marks.

Open book test/ Open notes test: 05 Marks

Regularity & Active Participation: 05 Marks

2. Mid Semester Examination (30 MARKS)

Mid semester examination will be based on 40% to 50% syllabus.

END SEMESTER EXAMINATION (50 MARKS)

Mid semester examination will be based on syllabus coverage up to Mid Semester Examination(MSE) carrying 20%-30% weightage, and syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	CEPCC407	OPERATING SYSTEM	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite :

Nil

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis

Course Objectives:

1. To impart comprehensive understanding of fundamental concepts and functions of operating systems.
2. To instruct how to use the concept of process, thread and resource management.
3. To impart learners to use the concepts of process synchronization and deadlock.
4. To familiarise various Memory, I/O and File management techniques.

Module	Details	Hrs
	Course Introduction This is the basic course of computer engineering which explore various functions of Operating Systems and how it used as an interface between the user and the computer hardware, and controls the execution of all kinds of programs. This course will provide basic to advanced concepts like process scheduling, concurrency control, Memory, File and IO management	**01**
01.	**Introduction of operating system** ***Learning Objective/s:*** *Learners are expected to use computer engineering knowledge /principles for learning the Objectives, Functions, architectures of Operating System and recognize it's need in the field of computer engineering.*	**4-6**
	Contents: Concept of Operating Systems (OS), Generations of OS, Types of OS, OS Services, Interrupt handling and System Calls, Basic architectural concepts of an OS, Concept of Virtual Machine.	
	Self-Learning Topics: *Resource Manager view, process view and hierarchical view of an OS*	

	Learning Outcomes: *A learner will be able to* *LO1.1: Use principles of computer organization to identify the importance of Operating system. (P.I.-1.3.1)* *LO1.2: Apply principles of operating system to interpret its objectives and functions. (P.I.-1.4.1)*	
02.	**Process Management** ***Learning Objective/s:*** *Learners are expected to Apply computer engineering principles and mathematical knowledge to learn about process and how they are scheduled.*	**8-10**
	Contents: 2.1 Process Definition, Different states of a Process, Process State transitions, Process Control block (PCB), Context switching. 2.2 Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads. 2.3 Process Scheduling: Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time. Scheduling algorithms: Pre-emptive and non-pre-emptive, FCFS, SJF, RR, Real time scheduling	
	Self-Learning Topics: *rate monotonic scheduling, earliest deadline first scheduling, and deadline monotonic scheduling.*	
	Learning Outcomes: *A learner will be able to* *LO2.1: Use principles of Operating System to illustrate the importance of Process Scheduling. (P.I.-1.3.1)* *LO2.2: Interpret the concepts of a Process, Process States, Process Description and Process Control Block, importance of threads(P.I.-1.4.1)* *LO2.3: Compare the scheduling algorithm and justify which is more effective with respect to CPU utilization and response time. (P.I.-2.2.4)* *LO2.4: Identify scheduling algorithm that applies to a given problem. (P.I.-2.1.3)*	
03.	**Process Concurrency control and Deadlock** ***Learning Objective/s:*** *Learners are expected to Conceptualize how OS handles concurrency control and deadlock mechanism, and conclude on suitability of solution.*	**8-10**
	Contents: 3.1 Concurrent processes, precedence graphs, Critical Section, Critical region Race Conditions, Mutual Exclusion, Hardware Solution, Semaphores: binary and counting, Peterson's Solution, The Producer / Consumer Problem, Monitors, Inter Process Communication: Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem 3.2 Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Dead lock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.	
	Self-Learning Topics: *Barber's shop problem*	

	Learning Outcomes: *A learner will be able to* *LO3.1: Use Operating system fundamentals to perceive concurrency control mechanisms. (P.I.-1.3.1)* *LO3.2: Apply OS concepts to solve Concurrency control and Deadlock problems. (P.I.-1.4.1)* *LO3.3: Select the appropriate concurrency control mechanism to solve the classical synchronization problems with approximations and assumptions. (P.I.-2.2.3)* *LO3.4: Select a suitable semaphore solution for a given concurrency control problem and conclude. (P.I.-2.4.4)*	
04.	**Memory Management**	**9-11**
	Learning Objective/s: *Learners are expected to apply principles of mathematics to comprehend the optimal memory management solution and use it to analyse the system behaviour.*	
	Contents: 4.1 Basic concept of memory, Logical and Physical address maps, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction. 4.2 Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page allocation, Partitioning, Paging, Page fault, Working Set, Segmentation, Demand paging, Page Replacement algorithms: Optimal, first in first out (FIFO), Least Recently used (LRU), Second Chance (SC)	
	Self-Learning Topics: *Not recently used (NRU)*	
	Learning Outcomes: *A learner will be able to* *LO 4.1: Apply computer architecture fundamentals to perceive the need of memory management. (P.I.-1.3.1)* *LO 4.2: Apply memory management techniques in solving problems related to various partitioning and virtual memory management methods. (P.I.- 1.4.1)* *LO 4.3: Compare and contrast memory partitioning techniques to select best one.. (P.I.-2.2.4)* *LO 4.4: Analyse and Conclude on the best memory allocation strategies. (P.I.-2.4.4)*	
05.	**I/O management** ***Learning Objective/s:*** *Learners are expected to analyse the disk organisation techniques*	**4-6**
	Contents: Principles of I/O and Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting.	
	Self-Learning Topics: *Boot-block, Bad blocks.*	

	Learning Outcomes: *A learner will be able to* *LO 5.1: Apply concepts of I/O functions and Disk Organization using engineering knowledge. (P.I.-1.4.1)* *LO 5.2: Illustrate the working of I/O management by applying OS concepts. (P.I.-1.3.1)* *LO 5.3: Identify suitable page replacement algorithms to solve the problem of demand paging. (P.I.-2.2.3)* *LO 5.4: Use and analyze the performance of various disk scheduling techniques and conclude. (P.I.-2.4.4)*	
06.	**File management** ***Learning Objective/s:*** *Learners are expected to Apply computer engineering principles and choose suitable file allocation method and justify suitability of file allocation method to optimize system performance.* **Contents:** Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed). ***Self-Learning Topics:*** *Free-space management (bit vector, linked list, grouping).* ***Learning Outcomes:*** *A learner will be able to* *LO 6.1: Apply OS fundamentals to illustrate the working of file management. (P.I.-1.3.1)* *LO 6.2: Use OS principles to summarize Various File organization and Access methods. (P.I.-1.4.1)* *LO 6.3: Compare various file allocation methods and choose optimal solution on file management. (P.I.-2.2.4)* *LO 6.4: Conclude on suitability of allocation based on given system design constraints. (P.I.-2.4.4)*	**4-6**
	Course Conclusion Operating System course provides fundamental understanding of system software and its role in managing computer resources efficiently, crucial for computer science and engineering students. Through theoretical concepts and practical implementations, students grasp essential skills in designing, implementing, and optimizing operating systems.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science and engineering to solve an engineering problem.
2.1.3	Identify mathematical algorithmic knowledge that applies to a given problem.
2.2.4	Compare and contrast alternative solution/methods to select the best methods.
2.2.3	Identify existing solution/mechanism to solve the problem, including forming justified approximations and assumptions.
2.4.4	Arrive at conclusions with respect to the objectives.

Course Outcomes: A learner will be able to-

1. Apply the concepts of Operating System to interpret its objectives and functions. *(LO1.1, LO1.2)*
2. Apply the concepts of process scheduling on a given scheduling scenario and justify its applicability to increase CPU utilization. *(LO2.1, LO 2.2, LO 2.3, LO 2.4)*
3. Apply concurrency control mechanism and select suitable solution for problems of deadlock and concurrency control. *(LO 3.1, LO 3.2, LO 3.3, LO 3.4)*
4. Analyse and select optimal memory management solutions. .*(LO 4.1, LO 4.2, LO 4.3, LO 4.4)*
5. Analyse various disk organization, I/O methods.(*LO 5.1, LO 5.2, LO 5.3, LO 5.4)*
6. Analyse different file access and file allocation methods in terms of efficiency.(LO 6.1, LO 6.2, LO 6.3, LO 6.4)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC407.1	3	-	-	-	-	-	-	-	-	-	-
CEPCC407.2	3	3	-	-	-	-	-	-	-	-	-
CEPCC407.3	3	3	-	-	-	-	-	-	-	-	-
CEPCC407.4	3	3	-	-	-	-	-	-	-	-	-
CEPCC407.5	3	3	-	-	-	-	-	-	-	-	-
Average	**3**	**3**	-	-	-	-	-	-	-	-	-

Text Books:

1. Operating Systems: Internals and Design Principles by William Stallings, 9th edition (Global edition), Pearson, 2018.
2. Operating System Concepts by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, 10th edition (Global edition), Wiley, 2018 (2023).

Reference Books :

1. Operating Systems by Achyut Godbole and Atul Kahate, 3rd Edition, McGraw Hill Education.
2. Modern Operating Systems by Tanenbaum and Herbert Bos, 5th edition, Pearson, 2023.
3. UNIX: Concepts and Applications by Sumitabha Das, 4th Edition, McGraw Hill.2020

Other Resources :

1. NPTEL Course: OS fundamentals by Prof. Santanu Chattopadhyay IIT Kharagpur Web link- https://nptel.ac.in/courses/106105214

IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

One MCQ test as per GATE exam pattern/ level: 05 marks.

One Class test: 05 marks

Think Pair share worksheet: 05 Marks

Regularity and active participation: 05 Marks

2. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination(MSE) carrying 20%-30% weightage, and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	CEPCC408	COMPUTER NETWORK	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite:

1. CEPCC303 - Data Structures

Program Outcomes addressed:

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

Course Objectives:

1. To impart the fundamental knowledge of computer networks.
2. To introduce the concept of IP addresses and various protocols used at network layer.
3. To familiarize the strengths and weaknesses of various routing algorithms.
4. To introduce the protocols of various transport layer and application layer.

Module	Details	Hrs.
	Course Introduction A computer network theory course offers foundational understanding in data transmission, protocols, and security, empowering students for research, project development, higher education pursuits, and industry roles like network engineering and cybersecurity analysis.	**01**
01.	**Introduction to Networking** ***Learning Objective/s:*** *To apply fundamentals of computer engineering to learn the concepts of computer networks and network requirements for real life applications.*	**6-8**

	Contents: 1.1 Basic concepts and fundamentals of data communication and computer network, Basic Networking Devices: Repeater, Hub, Switch, Router, NIC, Modem, Network Topologies, Type of networks (LAN, WAN, MAN) 1.2 Layers of OSI and TCP/IP Design Issues of Layers. Guided and Unguided media. Switching– Circuit-switched Networks – Packet Switching, Message switching	
	Self-Learning Topics: *Study of Network Tools (NS2 and Cisco Packet tracer)*	
	Learning Outcomes: *A learner will be able to:* *LO1.1: Use the computer engineering concept to identify required software and hardware comp*onents for each layer of network.*(PI-1.4.1)* *LO1.2: Compare and contrast different switching techniques and Transmission Media. (P.I.-1.4.2)* *LO1.3: Identify the functionalities and computing resources of all layers of OSI and TCP/IP models.(PI-2.2.2)* *LO1.4: Compare different network topologies and conclude which topology performs better(PI-2.4.4)* *LO1.5: Use the principles of engineering to understand the basic concepts and fundamentals of computer network useful in the future(PI-1.3.1,11.2.2)*	
02.	**Data Link Layer** ***Learning Objective/s:*** *To use fundamentals of computer engineering to learn the various issues and the available solution of Data Link Layer and Medium access control layer.*	**7-9**
	Contents: 2.1 Overview of DLL, Issues of DLL: Framing, Error Detection and Correction: Parity, CRC, Checksum, Hamming Code 2.2 Elementary Data Link protocols, Stop and Wait, Sliding Window (Go Back N, Selective Repeat) 2.3 Medium Access Control Sublayer Channel Allocation problem, Multiple access Protocol (Aloha, Carrier Sense Multiple Access (CSMA/CD)	
	Self-Learning Topics: *HDLC protocol, CSMA/CA*	
	Learning Outcomes : *A learner will be able to* *LO2.1: Apply engineering fundamentals to solve channel allocation problem in Medium Access Control sublayer (P.I-1.3.1)* *LO2.2: Apply computer engineering fundamentals to find various issues of DLL. (P.I.-1.4.1)* *LO2.3: Compare various Elementary Data Link protocols based on functionalities. (P.I.-2.2.2)* *LO2.4: Apply engineering mathematics to solve numerical problems based on Error Detection and Correction techniques. (P.I.-2.4.1)*	
03.	**Network Layer** ***Learning Objective/s:*** *To apply fundamentals of IP addressing to design network using the concepts of subnet / supernet.*	**7-9**

	Contents: 3.1 Network layer issues, Communication Primitives, IPV4 datagram format, IPv4 address, classful address, Design network using the concept of subnetting and supernetting, classless addressing, IPV6 datagram format, Transition from IPV4 to IPV6 3.2 Protocols - ARP, RARP, ICMP, IGMP	
	Self-Learning Topics: ***NAT***	
	Learning Outcomes: *A learner will be able to:* *LO3.1: Use the principles of engineering to restate the concepts of Communication primitives used in computer network. (P.I.-1.3.1)* *LO3.2: Use the concepts of* computer engineering *to summarize various issues of Network layer. (P.I.-1.4.1)* *LO3.3: Summarize the functionalities and computing resources of various network layer protocols. (P.I.-2.2.2)* *LO3.4: Compare IPv4 and IPv6 protocols based on functionalities. (P.I.-2.4.4)* *LO3.5: Apply Classful and classless addressing to explore design alternatives. (P.I.-3.2.1)* *LO3.6: Apply the concepts of IP addressing to develop networks and verify the functionalities. (P.I.-3.4.3)*	
04.	**Routing in Network Layer** ***Learning Objective/s:*** *To use the fundamentals of various routing algorithms and protocols at network layer to find the optimal path.*	**5-7**
	Contents: Introduction to Routing in computer network, Routing algorithms- Shortest Path (Dijkastra's), Link state routing, Distance Vector Routing, RIP, BGP.	
	Self-Learning Topics: *Routing Protocol (OSPF)*	
	Learning Outcomes: *A learner will be able to* *LO4.1: Apply the concepts of queueing theory to summarize the different routing protocols. (P.I.-1.1.2)* *LO4.2: To use computer engineering concepts to restate the purpose and use of intra/internetwork layer protocols. (P.I.-1.4.1)* *LO4.3: Apply various routing algorithms to find the optimal path between the source and the destination nodes to the given situation/problem. (P.I.-4.1.2).* *LO4.4: Identify the appropriate routing protocols for the given problem/data/situation. (P.I.-4.3.1).*	
05.	**Transport Layer** ***Learning Objective/s:*** *To know the functionalities of transport layer and analyze various protocol works at transport layer.*	**7-9**

	Contents: 5.1 Introduction to Transport layer services: User Datagram Protocol (UDP), Transmission Control Protocols (TCP) 5.2 Congestion control algorithms: Open loop congestion control, Closed loop congestion control, QoS parameters, Token & Leaky bucket algorithms 5.3 TCP Flow control (sliding Window), TCP Congestion Control: Slow Start	
	Self-Learning Topics: *Fast Transmit /Fast Recovery protocols*	
	Learning Outcomes: *A learner will be able to:* *LO5.1: Use the engineering fundamentals to summarize the main objectives, service primitives of transport layer. (P.I.-1.3.1)* *LO5.2: Apply the concepts of Transmission control protocol and summarize flow control and congestion control. (P.I.-1.4.1)* *LO5.3: Identify the functionalities and computing resources to compare the transport layer protocols: TCP and UDP. (P.I.-2.2.2)* *LO5.4: Compare Token & Leaky bucket algorithms and conclude with respect to the given objectives. (P.I.- 2.4.4)*	
06.	**Application Layer** ***Learning Objective/s:*** *To recognize the different protocols their functionalities used at application layer which are useful in real life applications.*	**6-8**
	Contents: DNS, Telnet, HTTP, FTP, SMTP, Streaming audio and video RTSP, SRTP.	
	Self-Learning Topics: *SSH*	
	Learning Outcomes: *A learner will be able to:* *LO 6.1: Use the basic concepts of computer engineering to summarize the DNS and various types of name server. (P.I.-1.3.1)* *LO 6.2: Analyze the structure of HTTP request and response. (P.I.-.2.2.2)* *LO 6.3: Analyze different intermediaries used in mail delivery. (P.I.-.2.2.3)* *LO 6.4: Identify the purpose of various protocols used in the application layer for different real- life applications (P.I- 1.4.1,11.2.2)* *LO6.5: Explore advanced protocols of Application Layer used in recent trend (P.I-1.3.1,11.3.1)*	
	Course Conclusion: In this course, students will be able to apply theoretical knowledge of computer networks to practical scenarios.	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.1.2	Apply the concepts of probability, statistics and queuing theory in modelling of computer-based system, data and network protocols.
1.3.1	Apply engineering fundamentals
1.4.1	Apply fundamental engineering concepts to solve computer engineering problems
2.2.2	Identifies functionalities and computing resources
2.4.1	Applies engineering mathematics to implement the solution.
2.4.4	Arrive at conclusions with respect to the objectives (as per network requirements)
3.2.1.	Able to explore design alternatives.
3.4.3	Able to verify the functionalities and validate the design.
4.1.2	Ability to choose appropriate procedure/algorithm/protocol depends on the situation (M)
4.3.1	Use appropriate procedures, tools, and techniques to collect and analyse data
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field
11.3.1	Source and comprehend technical literature and other credible sources of information.

Course Outcomes: A learner will be able to-

1. Use the basic concepts of computer networks to compare different network topologies and network models.(LO1.1, LO1.2, LO1.3, LO1.4, LO1.5)
2. Apply the functionalities (Framing, Error detection, flow control, error control and collision detection) of data link layer to solve the given problems.(LO2.1, LO2.2, LO2.3,LO2.4)
3. Design a network using subnetting / supernetting schemes.(LO3.1, LO3.2, LO3.3, LO3.4,LO3.5, LO3.6)
4. Apply the appropriate routing algorithm / protocol to find the optimal path.(LO4.1, LO4.2, LO4.3, LO4.4)
5. Compare the transport layer protocols with reference to their functionalities.(LO5.1, LO5.2, LO5.3, LO5.4)
6. Identify the features and operations of various application layer protocols such as DNS, HTTP, SMTP, Telnet, FTP and DHCP.(LO6.1, LO6.2, LO6.3,LO6.4, LO6.5)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC408.1	3	3	-	-	-	-	-	-	-	-	2
CEPCC408.2	3	3	-	-	-	-	-	-	-	-	-
CEPCC408.3	3	3	3	-	-	-	-	-	-	-	-
CEPCC408.4	3	-	-	3	-	-	-	-	-	-	-
CEPCC408.5	3	3	-	-	-	-	-	-	-	-	-
CEPCC408.6	3	2	-	-	-	-	-	-	-	-	3
Average	**3**	**3**	**3**	**3**	-	-	-	-	-	-	**3**

Text Books:

1. Computer Networks, A.S. Tanenbaum,4th edition, 2018 Pearson Education.
2. Data Communications and Networking, B.A. Forouzan, 5th edition, 2017, TMH.
3. A Top-Down Approach Featuring the Internet, James F. Kurose, Keith W. Ross, Computer .Networking, 6th edition, 2017, Addison Wesley.

Reference Books:

1. An Engineering Approach to Computer Networking, S. Keshav,1 Edition, 2007, Pearson.
2. Computer Networks: Principles, Technologies & Protocols for Network Design, Natalia Olifer & Victor Olifer, 1 Edition, 2006 Wiley India.
3. Computer Networks: A Systems Approach, Larry L.Peterson, Bruce S.Davie, , Second Edition, 2011, The Morgan Kaufmann Series in Networking.

Other Resources:

1. NPTEL Course: Computer Networks by Prof. Sujoy Ghosh, Department of Computer Science and Engineering IIT Kharagpur: -Web link- https://nptel.ac.in/courses/106105081
2. CISCO Network academy course: https://www.netacad.com/courses/networking/networking-essentials
3. The Bits and Bytes of Computer Networking : https://www.coursera.org/learn/computer-networking

IN-SEMERSTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

One MCQ test as per GATE exam pattern / level: 05 Marks

One Class test: 05Marks

Think Pair Share (TPS) activity: 05 Marks

Regularity and Active Participation: 05 Marks

2. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50 % syllabus.

END SEMERSTER EXAMINATION (50 MARKS)

End semester examination will be based on syllabus coverage up to the Mid semester examination (MSE) carrying 20%-30% weightage and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
LBC	CELBC403	ANALYSIS OF ALGORITHMS LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Examination (ESE)**	**Total**
25	25	50

Pre-requisite:

1. ESL103- Programming Laboratory- I (C)

Program Outcomes addressed:

1. PO2: Problem analysis
2. PO3: Design/development of solutions
3. PO4: Conduct investigations of complex problems
4. PO5: Engineering tool usage
5. PO8: Individual and Collaborative team work
6. PO9: Communication

Course Objectives:

1. To guide students in implementing various algorithms.
2. To guide students in analysing the performance of algorithms.

Module	Detailed Contents	Hrs
	Course Introduction This course emphasizes the practical implementation of algorithms, as well as delving into their performance analysis.	**01**
01.	***Learning Objective/s:*** *Expected to implement iterative sorting approach.*	**04**
	Laboratory Exercises: Implement iterative approach for the given problems and analyze their performance (Time and Space Complexity).	
	Learning Outcomes: *A learner will be able to* *LO1.1Develop appropriate algorithms based on the programming objectives (P.I.-4.2.1)* *LO1.2 Implement various iterative sorting methods. (P.I.-3.4.2)* *LO1.3 Use software tool to implement iterative approach (P.I.-5.1.2)* *LO1.4 Analyze iterative approach for the given problems based on their time and space complexity (P.I.-2.4.2)* *LO1.5 Facilitate the analysis by drawing conclusions and present in report form (P.I.-2.4.4, P.I.-8.2.1, P.I.-9.1.2)*	
02.	***Learning Objective/s:*** *Expected to implement divide & conquer algorithmic approach.*	**04**

	Laboratory Exercises: Implement divide & conquer approach for the given problems and analyze their performance (Time and Space Complexity).	
	Learning Outcomes: *A learner will be able to* *LO2.1 Develop appropriate procedures/algorithms based on the programming objectives (P.I.-4.2.1)* *LO2.2 Implement divide & conquer approach for given problems. (P.I.-3.4.2)* *LO2.3 Use software tool to implement divide & conquer approach. (P.I.-5.1.2)* *LO2.4 Analyze divide & conquer approach for the given problems based on their time and space complexity (P.I.-2.4.2)* *LO2.5 Facilitate the analysis by drawing conclusions and present in report form (P.I.-2.4.4, P.I.-8.2.1, P.I.-9.1.2)*	
03.	***Learning Objective/s:*** *Expected to implement greedy algorithmic approach.*	**06**
	Laboratory Exercises: Implement greedy approach for the given problems and analyze their performance (Time and Space Complexity).	
	Learning Outcomes: *A learner will be able to* *LO3.1 Design and develop appropriate procedures/algorithms based on the programming objectives (P.I.-4.2.1)* *LO3.2 Implement various greedy algorithms. (P.I.-3.4.2)* *LO3.3 Use software tool to implement various greedy approach. (P.I.-5.1.2)* *LO3.4 Analyse greedy approach for the given problems based on their time and space complexity (P.I.-2.4.2)* *LO3.5 Facilitate the analysis by drawing conclusions and present in report form (P.I.-2.4.4, P.I.-8.2.1, P.I.-9.1.2)*	
04	***Learning Objective/s:*** *Expected to implement dynamic programming algorithmic approach*	**06**
	Laboratory Exercises: Implement dynamic programming approach for the given problems and analyze their performance (Time and Space Complexity).	
	Learning Outcomes: *A learner will be able to* *LO4.1 Develop appropriate procedures/algorithms based on the programming objectives (P.I.-4.2.1)* *LO4.2 Implement various dynamic programming algorithms. (P.I.-3.4.2)* *LO4.3 Use software tool to implement various dynamic programming approach(P.I.-5.1.2)* *LO4.4 Analyse dynamic programming approach for the given problems based on their time and space complexity (P.I.-2.4.2)* *LO4.5 Facilitate the analysis by drawing conclusions and present in report form (P.I.-2.4.4, P.I.-8.2.1, P.I.-9.1.2)*	
05.	***Learning Objective/s:*** *Expected to implement backtracking algorithmic approach*	**05**

	Laboratory Exercises: Implement backtracking approach for the given problems and analyze their performance (Time and Space Complexity).	
	Learning Outcomes: *A learner will be able to* *LO5.1 Develop appropriate procedures/algorithms based on the programming objectives (P.I.-4.2.1)* *LO5.2 Implement various backtracking algorithms. (P.I.-3.4.2)* *LO5.3 Use software tool to implement various backtracking approach. (P.I.-5.1.2)* *LO5.4 Analyze backtracking approach for the given problems based on their time and space complexity (P.I.-2.4.2)* *LO5.5 Facilitate the analysis by drawing conclusions and present in report form (P.I.-2.4.4, P.I.-8.2.1, P.I.-9.1.2)*	
06.	***Learning Objective/s:*** *Expected to implement branch & bound algorithmic approach*	**04**
	Laboratory Exercises: Implement branch and bound approach for the given problems and analyze their performance (Time and Space Complexity).	
	Learning Outcomes: *A learner will be able to* *LO6.1 Develop appropriate procedures/algorithms based on the programming objectives (P.I.-4.2.1)* *LO6.2 Implement various branch & bound algorithms. (P.I.-3.4.2)* *LO6.3 Use software tool to implement various branch & bound approach. (P.I.-5.1.2)* *LO6.4 Analyze branch and bound approach for the given problems based on their time and space complexity (P.I.-2.4.2)* *LO6.5 Facilitate the analysis by drawing conclusions and present in report form (P.I.-2.4.4, P.I.-8.2.1, P.I.-9.1.2)*	
	Total	**30**

Performance Indicators:

P.I. No.	**P.I. Statement**
2.2.4	Compare and contrast alternative solutions/methods to select the best methods
2.4.2	Analyze and interpret the results using mathematical principles (modified PI).
3.4.2	Able to implement functions and integrate the modules
4.2.1	Design and develop appropriate procedures/ algorithms based on the programming objectives (modified PI).
5.1.2	Create/adapt/modify/extend tools and techniques to solve engineering problems.
8.2.1	Demonstrate effective communication, problem-solving, and conflict resolution.
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents

Course Outcomes: A learner will be able to-

1 Develop an algorithm or procedure for the given problem using specified approach. (LO1.1, LO1.2, LO1.3, LO1.4,LO1.5) (LO2.1, LO2.2, LO2.3, LO2.4, LO2.5) (LO3.1, LO3.2, LO3.3, L31.4, LO3.5) (LO4.1, LO4.2, LO4.3, LO4.4, LO4.5) . (LO5.1, LO5.2, LO5.3, LO5.4, LO5.5) (LO6.1, LO6.2, LO6.3, LO6.4, LO6.5)

2 Implement an algorithm for the given problem using appropriate software tool. . (LO1.1, LO1.2, LO1.3, LO1.4, LO1.5) (LO2.1, LO2.2, LO2.3, LO2.4, LO2.5) (LO3.1, LO3.2, LO3.3, L31.4, LO3.5) (LO4.1, LO4.2, LO4.3, LO4.4, LO4.5) . (LO5.1, LO5.2, LO5.3, LO5.4, LO5.5) (LO6.1, LO6.2, LO6.3, LO6.4, LO6.5)

3 Analyze and draw conclusions based on complexity of various algorithms. . (LO1.1, LO1.2, LO1.3, LO1.4, LO1.5) (LO2.1, LO2.2, LO2.3, LO2.4, LO2.5) (LO3.1, LO3.2, LO3.3, L31.4, LO3.5) (LO4.1, LO4.2, LO4.3, LO4.4, LO4.5) . (LO5.1, LO5.2, LO5.3, LO5.4, LO5.5) (LO6.1, LO6.2, LO6.3, LO6.4, LO6.5)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC403.1	-	3	2	2	2	-	-	2	2	-	-
CEPCC403.2		3	2	2	2	-	-	2	2	-	-
CEPCC403.3	-	3	2	2	2	-	-	2	2	-	-
Average	**-**	**3**	**2**	**2**	**2**	**-**	**-**	**2**	**2**	**-**	**-**

Books:

1. Introduction to algorithms, T. H. Cormen, C.E. Leiserson,R.L. Rivest, and C. Stein, 3rd Edition, 2009, The MIT Press
2. Fundamentals of computer algorithms, Ellis Horowitz, Sartaj Sahni, S. Rajsekaran, 2nd Edition, 2008, Universities Press

Reference Books:

1. Algorithms, Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, 1st Edition, 2006, McGraw- Hill India
2. Design Methods and Analysis of Algorithm, S. K. Basu, 2nd Edition, 2005, PHI

Other Resources:

1. NPTEL Course: Data Structures and Algorithms by Prof. Naveen Garg, Department of Computer Science and Engineering Department, IIT Delhi: -Web link- https://nptel.ac.in/courses/106/102/106102064/

2. NPTEL Course: Design and Analysis of Algorithms by Prof. Madhavan Mukund, Department of computer science at Chennai Mathematical Institute, Web link- https://nptel.ac.in/courses/106106131

CONTINUOUS ASSESSMENT (25 Marks)

Suggested breakup of distribution

Laboratory Exercises- 10 Marks

Internal Assessment-

Practical Test – 10 Marks

Evaluation of the conceptual, problem solving and programming skills of each student will be assessed based on their approach towards problem solving, implementation of concepts and execution of the task using a software tool.

Regularity and Active Participation - 5 Marks

END SEMESTER EXAMINATION (Practical/Oral Examination) (25 Marks)

Students will be assessed based on three parameters:

- Concept/Algorithmic and analysis knowledge
- Practical programming knowledge
- Oral

1. Students will be randomly allocated a program from the list of laboratory exercises and will be asked to write appropriate algorithm for the same. The algorithm and analysis are checked by the examiners (Internal and External) and weightage for this is 10 Marks.

Then the student will be allowed to start with the implementation of the program.

2. Students will be allocated 1 hour to complete the execution. The program is then checked by both the examiners for its correctness. The weightage of the program implementation is 5 Marks

3. Students will then be appearing for Oral in front of both Internal and External examiners. The weightage of Oral will be of 10 Marks.

Two examiners, one Internal and one External will do the evaluation.

ourse Type	Course Code	Course Name	Credits
LBC	CELBC404	OPERATING SYSTEM LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Examination (ESE)**	**Total**
25	25	50

Pre-requisite :

ESL103 - Programming Laboratory-I (C)

Program Outcomes addressed:

1. PO2: Problem analysis
2. PO3: Design/Development of Solutions
3. PO4: Conduct investigations of complex problems
4. PO5: Engineering tool usage
5. PO8: Individual and Collaborative team work
6. PO9: Communication

Course Objectives:

1. To familiarize various system calls, shell scripting and basic Linux commands.
2. To guide in implementing different process scheduling algorithms.
3. To facilitate in exploring the practical concepts of memory and deadlock avoidance.

Module	Detailed Contents	Hrs
	Course Introduction The course aims to explore the importance of the operating system, its function and different techniques used by the operating system to achieve its goals as resource manager. This course focuses on the practical experience of designing and implementing the important concept of Operating System.	
01.	**Linux Introduction** ***Learning Objective:*** *To familiarize the functioning of various Linux Tools and application.*	**06**
	Laboratory Exercise Implement basic Linux Commands.	
	Learning Outcomes: *A learner will be able to* *LO1.1: Identify and implement various operations related to File and directory structure using Linux based tool (P.I -2.1.2)* *LO1.2: Use Linux environment as a platform for implementing basic commands and demonstrate results. (P.I - 5.1.2, 8.2.1)*	

02.	**Shell Scripting and System Calls** ***Learning Objective:*** *To investigate various file manipulation and I/O operations using shell programming and system calls*	**06**
	Laboratory Exercises 1. Implement Shell scripting programs. 2. Implement system calls for process management and I/O management	
	Learning Outcomes : *A learner will be able to* *LO2.1: Identify the appropriate command to perform the given File, I/O and Process Management functions. (P.I - 2.1.2)* *LO2.2: Use Linux environment for implement system calls, shell scripting programs and demonstrate results. (P.I - 5.1.2, 8.2.1, 9.1.2)* *LO2.3: Implement the system call in both Linux as well as C environment and identify the strength and limitation of both. (P.I -5.2.1)*	
03.	**Process Scheduling** ***Learning Objective:*** *Learners are expected to select and implement efficient scheduling technique on the given problem.*	**06**
	Laboratory Exercises 1. Implement and justify which CPU Scheduling algorithm discriminates in favor of performance parameter. 2. Implement and show the following pairs of scheduling criteria with respect to: a. CPU utilization and response time b. Average turnaround time and maximum waiting time	
	Learning Outcomes : *A learner will be able to* *LO3.1: Implement and integrate different functions to carry out various operations related to the given scheduling algorithm. (P.I -3.4.2)* *LO3.2: Compare the scheduling algorithms and justify which is more effective with respect to CPU utilization and response time. (P.I -2.2.4)* *LO3.3: Facilitate the explanation and analysis for the calculation of turnaround time and waiting time. (P.I -4.3.3)* *LO3.4: Synthesize the information about the results, draw conclusion and demonstrate results. (P.I -4.3.4, 8.2.1, 9.1.2)*	
04.	**Process Synchronization and Deadlock** ***Learning Objective:*** *Learners are expected to implement suitable solution for the given classic Inter-process Communication problem and deadlock avoidance algorithm.*	**06**
	Laboratory Exercises: • **Synchronization problem** 1. Bounded-buffer (or Producer-Consumer) Problem 2. Dining-Philosophers Problem 3. Readers and Writers Problem	

	4. Sleeping Barber Problem • **Deadlock avoidance** 1. Banker's Algorithm	
	Learning Outcomes : *A learner will be able to* *LO4.1: Select the appropriate algorithm to develop functions carrying out various operations of the process synchronization problem and deadlock avoidance algorithm (P.I -4.1.2)* *LO4.2: Design and develop appropriate algorithm based on classic inter-process communication problems. (P.I -4.2.1)* *LO4.3: Implement solutions for different Synchronization problems and demonstrate results. (P.I. -3.2.2, 8.2.1, 9.1.2)*	
05.	**Virtual Memory** ***Learning Objective:*** *Learners are expected to select and implement efficient scheduling technique to implement the given real-world application*	**06**
	Laboratory Exercises: 1. For the given set of string with some finite page frames implement and analyse which algorithm produces minimum number of page faults. a. First Come First Serve b. Least Recently Used c. Optimal Page replacement	
	Learning Outcomes : *A learner will be able to* *LO5.1: Implement and integrate different functions to calculate page faults related to page replacement. (P.I.-3.4.2).* *LO5.2: Use appropriate programming tool for given parameter and demonstrate results in report format. (P.I - 4.3.1, 8.2.1, 9.1.2)* *LO5.3: Analyze the page replacement algorithm for their performance and limitations. (P.I - 4.3.2)*	
	Total	**30**

Performance Indicators:

P.I. No.	P.I. Statement
2.1.2	Identify processes/modules/algorithms of a computer-based system and parameters to solve a problem.
2.2.4	Compare and contrast alternative solution/methods to select the best methods.
3.2.2	Able to produce a variety of potential design solutions suited to meet functional requirements.
3.4.2	Able to implement functions and integrate the modules.
4.1.2	Able to choose appropriate procedure/algorithm(M).
4.2.1	Design and develop appropriate procedures/functions based on the programming objectives. (M)
4.3.1	Use appropriate procedures, tools and techniques to collect and analyze data.

4.3.2 Analyze data for trends, correlations and limitations.

4.3.3 Represent data (in tabular and/or graphical forms) so as to facilitate analysis and explanation of the data, and drawing of conclusions.

4.3.4 Synthesize information and knowledge about the problem from the raw data to reach appropriate conclusions.

5.1.2 Create/adapt/modify/extend tools and techniques to solve engineering problems.

5.2.1 Identify the strengths and limitations of tools for (i) acquiring information, (ii) modeling and simulating, (iii) monitoring system performance, and (iv) creating engineering designs.

8.2.1 Demonstrate effective communication, problem-solving, and conflict resolution.

9.1.2 Produce clear, well-constructed, and well-supported written engineering documents.

Course Outcomes: A learner will be able to-

1. Investigate and demonstrate the ability to select commands and perform the given operations using system calls and shell scripting in Linux Environment. (LO1.1, LO1.2, LO1.3, LO2.1, LO2.2, LO2.3)
2. Select appropriate process scheduling algorithm and integrating its function for evaluating performance parameter using suitable software tool.(LO3.1, LO3.2, LO3.3, LO3.4)
3. Develop appropriate procedure to propose solution of classic Inter-process Communication problems and effectively implement deadlock avoidance algorithm. (LO4.1, LO4.2, LO4.3)
4. Select appropriate page replacement algorithm and analyse their performance based on a given parameter.(LO5.1, LO5.2, LO5.3)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CEPCC303.1	-	2	-	-	3	-	-	2	2	-	-
CEPCC303.2		2	2	3	-	-	-	2	2	-	-
CEPCC303.3	-	3	2	-	-	-	-	-	-	-	-
CEPCC303.4	-	-	2	2	-	-	-	2	2	-	-
CEPCC303.5	-	-	2	3	-	-	-	2	2	-	-
Average	**-**	**2**	**2**	**3**	**3**	**-**	**-**	**2**	**2**	**-**	**-**

Books :

1. Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, Operating System Concepts, 10th edition (Global edition), Wiley, 2018 (2023).
2. Tanenbaum and Herbert Bos, Modern Operating Systems, 5th edition, Pearson, 2023.
3. Willings, Operating Systems: Internals and Design Principles, 9th edition (Global edition), Pearson, 2018.

Reference Books :

1. Practicing Hand Book for Operating System Laboratory by Sathish Kumar Ravichandran, Archana Sasi.
2. Operating System Lab Programs: Guide to Shell and OS lab programs by S.Sydhani Begum.

Other Resources :

1. Operating System Fundamentals: https://archive.nptel.ac.in/courses/106/105/106105214/
2. Brian "Beej Jorgensen" Hall, Beej's Guide to Interprocess Communication, online tutorial.

CONTINUOUS ASSESSMENT (25 Marks)

Suggested breakup of distribution

Laboratory Exercises- 10 Marks

Internal Assessment-

Practical Test – 10 Marks

Evaluation of the conceptual and programming skills of each student will be assessed based on their skills to implement the concepts and execution of the task assigned using a software tool, during regular laboratory sessions.

Regularity and Active Participation- 5 Marks

END SEMESTER EXAMINATION (Practical/Oral Examination) -25 Marks

Students will be assessed based on three parameters:

- Concepts/Algorithmic knowledge
- Practical programming knowledge
- Oral

1. Students will be randomly allocated a program from the list of laboratory exercises and will be asked to write appropriate algorithm for the same. The algorithm is checked by the examiners (Internal and External) and weightage for this is 5 Marks.

Then the student will be allowed to start with the implementation of the program.

2. Students will be allocated 1 hour to complete the execution. The program is then checked by both the examiners for its correctness. The weightage of the program implementation is 10 Marks.

3. Students will then be appearing for Oral in front of both Internal and External examiners. The weightage of Oral will be of 10 Marks.

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
LBC	CELBC405	COMPUTER NETWORK LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Examination (ESE)**	**Total**
25	25	50

Pre-requisite:

ESL205- Programming Laboratory -II (Java)

Program Outcomes addressed:

1. PO3: Design/Development of solution
2. PO4: Conduct investigations of complex problems
3. PO5: Engineering tool usage
4. PO11: Life Long Learning

Course Objectives:

1. To guide in using network simulation tools.
2. To facilitate in developing the topological and routing strategies for an IP based networking infrastructure.
3. To guide in exploring various issues of a packet transfer from source to destination and to resolve them.

Module	Detailed Contents	Hrs.
	Course Introduction A network lab is a dedicated space equipped with hardware and software tools essential for experimenting with and studying computer networks. It provides hands-on learning opportunities for students and professionals to understand concepts such as network configuration, troubleshooting, and security measures. By simulating real-world scenarios, network labs foster practical skills development and facilitate deeper comprehension of networking principles. They serve as invaluable resources for honing expertise in designing, implementing, and maintaining robust network infrastructures.	**01**
01.	***Learning Objectives:*** *1. Impart the knowledge of Ethernet cabling and connection to setup computer network.* *2. Expected to use different Linux commands for computer networks* *3. Expected to know how to analyze network traffic*	**05**
	Ethernet Networking: Cabling, Linux Commands, Traffic Analysis **Laboratory Exercises:** 1. To demonstrate Ethernet cabling and connection (RJ45 connector and CAT6 cable). 2. Use basic networking commands in Linux/Windows (ping, tracert, nslookup, netstat, ARP, RARP, ip, ifconfig, dig, route, nmap) 3. Use simulation tool Wireshark to analyse network traffic.	

	Learning Outcomes: *A learner will be able to* *LO1.1 Identify various components required to setup computer network. (P.I.- 3.4.1)* *LO1.2 Basic tools of computer engineering required for analysing network traffic. (P.I.- 4.1.3)* *LO1.3 Crimp Ethernet network cable using CAT6 cable and RJ45 connector and check the cable using Line Tester. (P.I.- 4.3.1)* *LO1.4 Use basic networking commands in Linux/Windows (ping, tracert, nslookup, netstat, ARP, RARP, ip, ifconfig, dig, route) to analyze the network.(P.I.- 5.1.1)* *LO1.5 Use Wireshark tool for analysing network traffic. (P.I.- 5.2.1)* *LO1.6 Use network mapping tool Nmap to analyze the network. (P.I.- 5.2.2)*	
02.	***Learning Objective:*** *Expected to use network tools and simulators like NS2, Cisco Packet Tracer for building a simple network topology.*	**08**
	Network Topology Construction: NS2, Cisco Packet Tracer, and Tools **Laboratory Exercises:** 1. To use simulator (NS2) to understand functioning of ALOHA. 2. To build a simple network topology, configure it for static routing protocol and validate using Cisco Packet Tracer. 3. To configure network using static routing protocol and demonstrate the ability in identifying the advance protocols.	
	Learning Outcomes: *A learner will be able to* *LO2.1 Build a simple network topology using Cisco Packet Tracer. (P.I.- 4.1.3)* *LO2.2 Validate the created network using appropriate commands/tools. (P.I.- 4.2.1)* *LO2.3 Use network simulator to demonstrate ALOHA. (P.I.- 5.1.1)* *LO2.4 Apply static routing protocol to configure the network. (P.I.- 5.2.2, P.I.- 11.2.2)*	
03.	***Learning Objectives:*** *1. Expected to setup multiple IP addresses on a single LAN and configure routing information.* *2. Expected to create Virtual Private Network and Configure RIP/OSPF using Packet tracer.*	**08**
	Configuring Multi-IP LAN & VPN with Packet Tracer **Laboratory Exercises:** 1. Set up multiple IP addresses on a single LAN. 2. Using nestat and route commands of Linux, do the following: • View current routing table • Add and delete routes • Change default gateway 3. Design VPN and Configure RIP/OSPF using Packet tracer.	
	Learning Outcomes: *A learner will be able to* *LO3.1 Use linux commands to setup multiple IP addresses on a single LAN (IP Aliasing). (P.I.- 4.1.2)* *LO3.2 Use the appropriate tools for simulating computer networks. (P.I.- 4.1.3).* *LO3.3 Configure and update routing information using netstat and route command. (P.I.- 5.2.2)* *LO3.4 Create, configure and recognize virtual private network used in real life applications using Cisco Packet Tracer simulation tool. (P.I.- 5.1.1,11.2.2)*	

04.	***Learning Objectives:*** *1. Expected to implement core programming APIs for networking concepts like Socket programming.* *2. Expected to perform remote login using Telnet Server.*	**08**
	Networking APIs: Socket Programming & Telnet Remote Access **Laboratory Exercises:** 1. Write a program to implement socket programming using TCP 2. Write a program to implement socket programming using UDP. 3. To perform remote login using Telnet Server in Linux /windows	
	Learning Outcomes: *A learner will be able to* *LO4.1 Use JAVA/python language to write program for socket programming based on TCP or UDP. (P.I.- 4.1.3)* *LO4.2 Setup and configure telnet server and client for remote login and apply these strategies in future for professional development. (P.I.- 4.1.2, P.I.- 11.1.1)*	
	Minimum 2 or 3 laboratory exercises from each module, and total at least 10 exercises	
	Total	**30**

P.I. No. P.I. Statement

3.4.1	Able to refine architecture design into a detailed design within the existing constraints
4.1.2	Able to choose appropriate procedure/algorithm, dataset and filters.
4.1.3	Able to choose appropriate hardware/software tools to conduct the experiment.
4.2.1	Design and develop appropriate procedures/methodologies based on the study objectives.
4.3.1	Use appropriate procedures, tools and techniques to collect and analyze data.
5.1.1	Identify modern engineering tools such as computer aided drafting, modeling and analysis; techniques and resources for engineering activities.
5.2.1	Identify the strengths and limitations of tools for (i) acquiring information, (ii) modeling and simulating, (iii) monitoring system performance, and (iv) creating engineering designs.
5.2.2	Demonstrate proficiency in using discipline-specific tools.
11.1.1	Describe the rationale for the requirement for continuing professional development
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field.

Course Outcomes: A learner will be able to-

1. Develop and setup network environment in Linux.(LO1.4, LO1.6, LO3.1)
2. Use Network tools and simulators such as NS2, Wireshark etc. to explore network algorithms and protocols. .(LO1.2, LO1.5, LO2.1,LO2.3)
3. Setup and configure network connections. .(LO1.1, LO1.2, LO1.3)
4. Implement programs using core programming APIs for understanding network concepts. .(LO2.1, LO2.4, LO3.1,LO3.2,LO3.3,LO3.4,LO4.1,LO4.2)

CO-PO Mapping Table with Correlation Level

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CELBC405.1	-	-	2	3	3	-	-	-	-	-	-
CELBC405.2	-	-	-	3	3	-	-	-	-	-	2
CELBC405.3	-	-	-	3	3	-	-	-	-	-	2
CELBC405.4	-	-	-	2	-	-	-	-	-	-	2
Average	**-**	**-**	**2**	**3**	**3**	**-**	**-**	**-**	**-**	**-**	**2**

Text Books:

1. Computer Network Simulation in NS2 Basic Concepts and Protocol implementation, Prof Neeraj Bhargava, Pramod Singh Rathore, Dr.Ritu Bhargava Dr.Abhishek Kumar, First Edition, 2020, BPB Publication.
2. Packet analysis with Wire shark, Anish Nath, 2015, PACKT publishing NS2.34 Manual.
3. Practical Packet Analysis Using Wireshark to Solve Real-World Network Problems, Chris Sanders, Second Edition,2007, No Starch Press.

Reference Books:

1. An Engineering Approach to Computer Networking, S. Keshav,1 Edition, 2007, Pearson.
2. Computer Networks: Principles, Technologies & Protocols for Network Design, Natalia Olifer & Victor Olifer, 1 Edition, 2006 Wiley India.
3. Computer Networks: A Systems Approach, Larry L.Peterson, Bruce S.Davie, , Second Edition, 2011, The Morgan Kaufmann Series in Networking.

Other Resources:

1. CISCO Network Academy: https://www.netacad.com/courses/packet-tracer/introduction-packet-tracer
2. A coursera Course- https://www.coursera.org/projects/data-forwarding-computer-networks
3. TUMx: iLabX: https://www.edx.org/course/ilabx-the-internet-masterclass

CONTINUOUS ASSESSMENT (25 Marks)

Suggested breakup of distribution
Laboratory Exercises- 10 Marks
Internal Assessment-
Practical Test– 10 Marks

Evaluation of the conceptual, problem solving and programming skills of each student will be assessed based on their approach towards problem solving, implementation of concepts and execution of the task using a software tool.

Regularity and Active Participation- 5 Marks

END SEMESTER EXAMINATION (Practical /Oral Examination) (25 Marks)

Students will be assessed based on three parameters:

- Concept/Tool knowledge
- Practical programming knowledge
- Oral

1. Students will be randomly allocated a practical from the list of laboratory exercises and will be asked to write appropriate concept/methodology/program for the same. This is checked by the examiners (Internal and External) and weightage for this is 5 Marks.

   Then the student will be allowed to start with the implementation.

2. Students will be allocated 1 hour to complete the execution. The execution is then checked by both the examiners for its correctness. The weightage of the program implementation is 10 Marks.
3. Students will then be appearing for Oral in front of both Internal and External examiners. The weightage of Oral will be of 10 Marks

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
SBL	CESBL402	WEB DEVELOPMENT LABORATORY	02

Examination Scheme		
Continuous Assessment	**End Semester Examination (ESE)**	**Total**
50	50	100

Pre-requisite:

1. ESL103- Programming Laboratory -I (C)
2. ESL205- Java Programming Laboratory-II (Java)
3. CESBL301- Python Programming Laboratory

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct investigations of complex problems
5. PO5: Engineering tool usage
6. PO7: Ethics
7. PO8: Individual and Collaborative team work
8. PO9: Communication
9. PO11: Life Long Learning

Course Objectives:

1. To familiarize various components of HTML and CSS.
2. To facilitate in developing a responsive web application using JavaScript for enhanced user experience across devices.
3. To guide in implementing database connectivity using PHP and JSP to create dynamic web applications with data-driven content.
4. To guide in creating an interactive web application using AJAX, JavaScript, and jQuery to fetch and display dynamic content without page reloads.

Module	Detailed Contents	Hrs
00	**Course Introduction** Incorporating web development in computer science engineering curricula helps students apply theoretical knowledge to real-world projects, enhancing problem-solving skills. Exposure to diverse technologies cultivates versatility, aiding adaptation to industry trends. Collaborative web projects also foster teamwork, communication, and project management skills crucial for professional success. This integration enriches technical proficiency and develops essential soft skills, ensuring comprehensive readiness for future careers.	**01**
01.	**Foundations of Web Communication and Markup Languages** ***Learning Objective:*** *Learners are expected to identify and apply appropriate HTML and HTML5 elements to improve webpage interactivity, structure, and media integration.*	**09**

	Content: 1.1 Clients, Servers, and Communication. The Internet-Basic- Internet Protocols, The World Wide Web-HTTP request, Message-response Message-Web Clients Web Servers. 1.2 Basics of HTML- HTML Tags and attributes, meta tags, character, entities, hyperlink, lists, tables, images, forms, divs, XHTML 1.3 HTML5 control elements, Semantic elements, Audio – Video controls **Laboratory Exercise/s** **1. Designing Static Web Pages for any open-ended problem.** **(Case Study- Institution Web Page)** **a) Designing the Home Page:** - Create a static home page with three frames layout, typically including a navigation frame, content frame, and footer frame. - The navigation frame should contain links to different sections of the website such as About Us, Courses, Events, Contact Us, etc. - The content frame should display introductory information about the institution, highlights, and upcoming events. - The footer frame can include contact information, social media links, and copyright details. **b) Creating the Login Page:** - Design a login page with input fields for username and password, along with a login button. - Optionally, include a Forgot Password link for password recovery functionality. **c) Developing the Catalog Page:** - Design a catalog page to display details of available books or resources in a tabular format. - Include columns such as book title, author, ISBN, publication date, and availability status. - Populate the table with sample book data for demonstration purposes ***Self-Learning Topics:*** *Drag and Drop* ***Learning Outcomes :*** *A learner will be able to* *LO 1.1: Apply acquired programming knowledge to establish logical flow and structure when creating static web pages (P.I 1.4.1)* *LO 1.2: Identify HTML tags used to design static web pages independently (P.I.-2.1.1, 8.1.1)* *LO 1.3: Create a webpage using a variety of HTML5 elements, focusing on semantic elements, and apply this knowledge in subsequent mini-projects for advanced web development in higher-level classes. (P.I.-2.3.2,11.1.1)* *LO 1.4: Integrate audio and video elements into webpages using HTML5 technique to create a more engaging multimedia experience for users. They will also adhere to fundamental programming ethics, produce clear and well-constructed written documents of their results. (P.I- 5.1.2, 7.1.1, 11.1.2.)*	
02.	**Mastering CSS for Modern Web Design** ***Learning Objective:*** *Learners expected to design and style static websites proficiently using CSS, showcasing diverse designs by applying a range of CSS properties.*	**10**

Content:

2.1 Introduction to CSS, Types of CSS, Syntax, selectors, Manipulating texts, using fonts, background images, colors, borders and boxes, margins, padding lists, positioning using CSS.

2.2 Gradients, Shadow Effects, Transformations, transitions and animations, etc., CSS box modal and CSS Flex, Positioning systems of CSS, CSS media Queries.

Laboratory Exercise/s

2. Enhancing Web Page Design with Advanced CSS Techniques

Use advanced CSS techniques to enhance the web design and user experience of the web pages created as first Laboratory Exercise. This Laboratory Exercise focuses on incorporating CSS Animations, Transformations, Transitions, and styling elements to achieve dynamic and visually appealing results.

a) Implement CSS Animations:

- Identify specific elements on the web pages where animations will be applied, such as buttons, images, or text.
- Use CSS key frames to define animation sequences, such as fading effects, sliding animations, or rotating elements.

b) Incorporate CSS Transformations:

- Laboratory Exercise with CSS transformations like scale, rotate, skew, and translate to modify the appearance and layout of elements.
- Apply transformations to images, text blocks, or navigation menus to create dynamic visual effects.

c) Integrate CSS Transitions:

- Add transitions to elements such as hover effects on buttons or menu items to improve user experience.
- Define transition properties like duration, timing functions, and specific style changes during transitions.

d) Design Stylish Elements:

- Add borders, margins, padding to elements to create well-defined layouts and spacing.
- Style lists with customized bullet points, indentation, or hover effects for interactive lists.

Self-Learning Topics:
Basics of Bootstrap

Learning Outcomes :
A learner will be able to

LO 2.1: Demonstrate the ability to create a diverse range of web page designs employing advanced CSS techniques and leverage this expertise in developing sophisticated web-based applications. (P.I.-3.2.2,5.1.1,11.1.1)

LO 2.2: Explore the design alternatives of web page in HTML document using various CSS properties independently (P.I.-3.2.1, 8.1.1)

LO 2.3: Enhance a static web pages using CSS techniques to apply borders, margins, padding, and lists for enhanced visual layout and styling. They will also adhere to fundamental programming ethics, produce clear and well-constructed written documents of their results. (P.I - 5.1.2, 7.1.1,9.1.2)

03.	**JavaScript Fundamentals** ***Learning Objective:*** *Design and develop web pages with appropriate JavaScript functions while analyzing and recognizing the concepts of exception and event handling mechanisms*	**10**
	Content: 3.1 Client side scripting, Introduction to JavaScript, writing simple JavaScript, Introduction to ES6, Difference between ES5 and ES6. Variables, Condition, Loops, Functions, Events, Arrow functions 3.2 JavaScript alert, prompt and confirm. Objects in JavaScript, Access /Manipulate web browser elements using DOM Structure, forms and validations, JavaScript events 3.3 Setting CSS Styles using JavaScript, DOM manipulation, Classes and Inheritance. 3.4 Iterators and Generators, Promise, Client-server communication, Fetch Self Learning Topics: Topological Sorting. **Laboratory Exercise/s** **3. Enhancing Web Page Design with JavaScript Interactivity** Building upon the foundation of designing static web pages and enhancing them with advanced CSS techniques, this Laboratory Exercise focuses on adding interactivity and functionality to web pages using JavaScript. **a) Incorporate JavaScript for user interactions:** - Implement event listeners for buttons, forms, and other elements to handle user actions. - Use conditional statements for dynamic content display based on user input or events. **b) Access and manipulate DOM elements:** - Use document methods to select and modify HTML elements dynamically. - Update text content, styles, and attributes of elements based on user interactions or data changes. **c) Implement client-side form validations:** - Validate user input for login credentials, form submissions, or other interactive elements. - Display error messages or feedback based on validation results. **d) Use JavaScript dialog boxes:** - Implement alerts, prompts, and confirmations for user notifications and interactions. - Customize dialog messages and responses based on application logic. **Note:** Please ensure to add dedicated pages or functionalities to accommodate and perform the above Laboratory Exercises within an existing web page ***Self-Learning Topics:*** *JSON introduction*	
	Learning Outcomes : *A learner will be able to* *LO 3.1: Design and develop web pages using appropriate JavaScript Functions independently (P.I.-4.1.2, 5.1.2, 8.1.1)* *LO 3.2: Demonstrate proficiency in integrating interactivity and functionality into web pages using JavaScript, recognizing the importance of designing interactive web pages in the digital age. Additionally, uphold fundamental programming ethics and produce well-constructed written documents detailing their results (P.I 4.2.1, 7.1.1,9.1.1, 11.1.1,11.2.2)*	
04.	**JQuery Essentials** ***Learning Objective:*** *To explore jQuery's capabilities in enhancing web page interactivity through animations, AJAX requests, and dynamic content updates, apply form submission to a server using jQuery's $.post() method, and utilize key JavaScript and jQuery concepts to create interactive and dynamic web content.*	**10**

	Content: 4.1 jQuery – Selectors :CSS Element Class Selector and Universal Selector,CSS Multiple Elements E, F, G Selector, jQuery Callback Functions 4.2 Query – Effects, JQuery Effect Methods, jQuery Load, jQuery Get , jQuery Post Hide and Show , jQuery Toggle , jQuery Slide – slideDown, slideUp, slideToggle, jQuery transition effects. **Laboratory Exercise/s** **4 Mastering jQuery Essentials for Web Development** Each Laboratory Exercise builds upon the previous one, gradually expanding your knowledge and skills in leveraging jQuery for dynamic and interactive web applications. **a) jQuery syntax and library inclusion in web pages.** - Use jQuery selectors to target HTML elements based on classes, IDs, and element types. - Create simple jQuery functions to manipulate DOM elements such as changing text, styles, and attributes dynamically. **b) Advanced jQuery Selectors and Events** - Utilize jQuery event handling methods (e.g., click, hover, submit) to create interactive user experiences. - Implement event delegation to handle events for dynamically added elements efficiently. **c) jQuery Effects and Animations** - Apply jQuery effect methods such as show, hide, toggle to control element visibility. - Implement slide animations (slideDown, slideUp, slideToggle) and fade animations (fadeIn, fadeOut, fadeTo) using jQuery. - Create custom animations using jQuery animate() method to animate CSS properties like height, width, opacity, etc. - Combine effects and animations to create engaging UI elements such as dropdown menus, accordions, and sliders. **Note:** Please ensure to add dedicated pages or functionalities to accommodate and perform the above Laboratory Exercises within an existing web page ***Self-Learning Topics:*** *React JS* ***Learning Outcomes :*** *A learner will be able to* *LO 4.1: Design and develop dynamic and interactive web pages using appropriate jquery independently (P.I.-4.1.2, 5.1.2, 8.1.1)* *LO 4.2: Demonstrate proficiency in web page development utilizing jQuery to integrate interactivity and dynamic elements, acknowledging the significance of designing dynamic web pages in the contemporary digital landscape. Additionally, uphold fundamental programming ethics and produce well-constructed written documents detailing their results (P.I 4.2.1, 7.1.1,9.1.1, 11.1.1,11.2.2)*	
05.	**Server-side Programming with Database Integration** ***Learning Objective:*** *To demonstrate proficiency in using PHP and JSP to create forms that validate user input, process and insert data into a database, and update or delete data, with verification using PHP and JSP.*	**10**

	Content: 5.1 Introduction to Server-side Programming: Fundamentals of PHP and JSP. Datatypes, Operators, Control Statements, Arrays and Functions of PHP and JSP. 5.2 Database interaction through PHP and JSP. **Laboratory Exercise/s** **5. Mastering PHP and MySQL Integration for Web Development** Each Laboratory Exercise builds upon the previous one, gradually expanding your knowledge and skills in PHP scripting, MySQL database connectivity, and handling data dynamically on the server side. **a) Building Dynamic Web Applications with PHP and MySQL** -Integrate PHP and MySQL functionalities to develop dynamic web applications such as content management systems, e-commerce platforms, or user management systems. -Implement user registration, login/logout functionality using PHP sessions and MySQL database storage. -Create interactive web pages that fetch, display, and update data from MySQL databases dynamically based on user interactions. **b) Debugging and Troubleshooting PHP and MySQL Interactions** -Learn debugging techniques for PHP code using logging, error reporting, and debugging tools like Xdebug. -Debug and troubleshoot common issues in PHP and MySQL interactions such as connection errors, query failures, and data retrieval problems. -Optimize PHP and MySQL configurations for performance, memory management, and scalability in production environments. -Implement error handling strategies and logging mechanisms for robust PHP and MySQL applications. **Note:** Please ensure to add dedicated pages or functionalities to accommodate and perform the above Laboratory Exercises within an existing web page ***Self-Learning Topics:*** *React Native* ***Learning Outcomes:*** *A learner will be able to* *LO 5.1: Demonstrate proficiency in creating and handling form submissions and database interactions using PHP independently. (P.I.-5.1.2, 8.1.1)* *LO 5.2: Handle server-side logic, interact with databases, and generate dynamic content using PHP and mysql. Recognize the fundamental skills in database management and interaction, providing a solid foundation for building more complex web applications and understanding how data is manipulated in real-world scenarios. They will also adhere to fundamental programming ethics, produce clear and well-constructed written documents of their results. (P.I-5.2.2,7.1.1,9.1.2 11.1.1, 11.2.2,)*	
06	**AJAX** ***Learning Objective:*** *Utilize AJAX (Asynchronous JavaScript and XML) to create dynamic web applications that fetch and display data asynchronously, enhancing user experience and reducing page reloads.*	**10**

Content:

6.1 Introduction to JavaScript and jQuery for AJAX, Handling JavaScript Events for AJAX, Asynchronous Programming in JavaScript for AJAX

6.2 Creating Visual Effects with jQuery Animations for AJAX, Building an AJAX-powered web application with JavaScript and jQuery, Debugging and Troubleshooting AJAX Code in JavaScript and jQuery.

Laboratory Exercise/s

6. Mastering AJAX Techniques with JavaScript and jQuery

Each Laboratory Exercise builds upon the previous one, gradually expanding your knowledge and skills in handling asynchronous data requests and integrating AJAX functionality into web projects.

a) AJAX with JavaScript

- Implement AJAX requests using vanilla JavaScript XML HttpRequest object to fetch data from a server asynchronously.

-Display fetched data dynamically on the web page using DOM manipulation techniques.

b) Handling JavaScript Events for AJAX

- Implement event-driven AJAX requests using JavaScript event handlers such as onclick, onchange, and onsubmit.

- Handle form submissions asynchronously using AJAX to prevent page reloads and improve user experience.

- Validate form data using JavaScript before sending AJAX requests to the server.

- Implement error handling and feedback mechanisms for AJAX responses (success and error scenarios).

c) Creating Visual Effects with jQuery Animations for AJAX

- Integrate jQuery library into the web project for enhanced DOM manipulation and animation capabilities.

- Implement jQuery animations (fadeIn, fadeOut, slideDown, slideUp, etc.) to create visual effects during AJAX interactions.

- Enhance user experience by adding loading spinners or progress bars during AJAX requests using jQuery animations.

- Combine jQuery animations with AJAX callbacks for seamless data updates and UI transitions

Note: Please ensure to add dedicated pages or functionalities to accommodate and perform the above Laboratory Exercises within an existing web page

Self-Learning Topics: --- *Deployment of website*

Learning Outcomes:

A learner will be able to

LO 6.1: Explore the design alternatives to create dynamic and interactive web applications using AJAX independently. (P.I.- 3.2.1, 8.1.1, 11.2.1)

LO 6.2: Verify the functionalities and validate the web site design through AJAX. (P.I.- 3.4.3)

LO 6.3: Analyze challenges in web development and employ AJAX techniques to address them effectively, facilitating the development of precise and responsive websites. (P.I.-4.3.1,11,2,2)

LO 6.4: Represent the live data appropriately by AJAX techniques. (P.I.-4.3.3)

	LO 6.5: Identify the proper technique or method to debug the AJAX code using Javascript and jquery. Analyze challenges in web development and employ AJAX techniques to address them effectively, facilitating the development of precise and responsive websites. They will also adhere to fundamental programming ethics, produce clear and well-constructed written documents of their results ((P.I.-5.1.1, 5.2.2, 7.1.1,9.1.1)	
	Course Conclusion: Upon completion of the Web Development Lab, students will acquire the proficiency to develop dynamic and interactive web applications.	
	Total	**60**

Performance Indicators:

P.I. No.	P.I. Statement
1.4.1	Apply theory and principles of computer science and engineering to solve an engineering problem
2.1.2	Identify processes/modules/algorithms of a computer-based system and parameters to solve a problem
3.2.1	Able to explore design alternatives.
3.2.2	Able to produce a variety of potential design solutions suited to meet functional requirements.
3.3.1	Able to perform systematic evaluation of the degree to which several design concepts meet the criteria.
3.4.3	Able to verify the functionalities and validate the design.
4.1.2	Able to choose appropriate procedure/algorithm, dataset and test cases.
4.2.1	Design and develop appropriate procedures/methodologies based on the study objectives.
4.3.1	Use appropriate procedures, tools and techniques to collect and analyze data.
5.1.1	Identify modern engineering tools, techniques and resources for engineering activities.
5.1.2	Adapt the tools and techniques to solve engineering problems.
5.2.2	Demonstrate proficiency in using discipline-specific tools.
7.1.1	Identify situations of unethical professional conduct and propose ethical alternatives.
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills.
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts.
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents.
9.1.3	Create flow in a document or presentation - a logical progression of ideas so that the main point is clear.
11.1.1	Describe the rationale for the requirement for continuing professional development.
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field.

Course Outcomes: A learner will be able to-

1. Develop interactive web pages using HTML and CSS to engage users dynamically. *(LO 1.1, LO 1.2, LO 1.3, LO 1.4, LO 2.1, LO 2.2, LO 2.3)*
2. Design and create a responsive web application using JavaScript to ensure compatibility across devices and screen sizes. *(LO 3.1, LO 3.2)*
3. Construct an interactive web application powered by AJAX using JavaScript and jQuery for seamless data retrieval and dynamic updates. *(LO 4.1, LO 4.2, LO 4.3, LO 6.1, LO 6.2, LO 6.3, LO 6.4, LO 6.5)*
4. Demonstrate the integration of PHP and MySQL to facilitate data storage, retrieval, and manipulation in web applications. *(LO 5.1, LO 5.2)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CESBL402.1	3	3	3	-	3	-	3	3	2	-	3
CESBL402.2			-	3	2	-	-	2	2	-	3
CESBL402.3			3	3	2	-	-	2	2	-	3
CESBL402.4			-	-	3	-	2	2	2	-	3
Average	**3**	**3**	3	**3**	3	-	2	2	2	-	3

Books :

1. Developing Web Applications Ralph Moseley, Second Edition, 2007 Willy.
2. Web Technology Black Book, Kogent Learning Sol., First Edition, Dreamtech Press, 2009
3. Learning PHP, MySQL, JavaScript, CSS & HTML5, Robin Nixon, Third Edition, O'REILLY, 2014.
4. Professional Rich Internet Applications: AJAX and Beyond Dana Moore, Raymond Budd, Edward Benson, 2007 Wiley publications.
5. Learning React Functional Web Development with React and Redux, Alex Banks and Eve Porcello, First Edition, 2017, OREILLY.

Reference Books :

1. Responsive Web Design with HTML5 and CSS3 by Ben Frain , Second Edition, 2015, Packt.
2. jQuery in Action, Bear Bibeault, Yehuda Katz, 2nd Edition, 2010, Manning.

CONTINUOUS ASSESSMENT (50 Marks)

Suggested breakup of distribution

Laboratory Exercises: 15 Marks

Internal Assessment: 10 Marks

As a part of Internal Assessment, students will do course mini project. It is will be group activity [P. I. - 9.3.1]. Students will be tasked with developing a website using web programming. This project-based assessment will require students to apply their knowledge and skills gained throughout the course to design and implement a functional system using Web programming language.

Course Project Rules in Web Development Lab

1. Group Size: Groups of 2 to 4 members allowed.
2. Project Proposal: Detailed proposal with scope, objectives.
3. Project Requirements:
   - Develop a project having functional backend.
   - Deploy using basic web technologies.
   - Developed web sites should be optimized for all devices.
4. Presentation:
   - Present web site features, challenges faced, and deliverables.
   - Q&A session for evaluation.
5. Evaluation Criteria:
   - Adherence to requirements and objectives.
   - Look and feel of web site, extend of completion and organization.
   - Functionality, UI/UX and error handling.
   - Effective presentation and Q&A skills.

Regularity and active participation: 05 Marks

Practical Test: 20 Marks

Two practical tests will be conducted based on laboratory exercises.

1. Students will be randomly assigned two or more web development tasks to evaluate their web development skills.

2. Students will have a designated 2-hour time frame for code development/task execution. After the first hour, an internal examiner will assess the progress of each student's project and provide feedback for enhancements, focusing on evaluating web development skills. Alongside web development skills, problem-solving abilities will also be evaluated.

3. Towards the end of the practical or during assessment, students will be asked questions to gauge their conceptual understanding of web development principles and techniques.

END SEMESTER EXAMINATION (Practical /Oral Exam) (50 Marks)

For the End semester exams, practical examination will be conducted. The details of the end-semester evaluation are as follows. It will consist of three sections:

Section 1: Practical Examination (35 Marks)

This section will have practical exam based on the laboratory exercises conducted during the term. The assessment criteria will be similar to Internal Practical Test. Each laboratory exercise can focus on different aspects of web development, such as HTML/CSS layout, JavaScript interactivity, JQuery, AJAX, Server-side scripting or integrating APIs.

Section 2: Documentation and Presentation (05 Marks)

During the practical exam, students should incorporate comprehensive code comments for project organization and readability. Additionally, they should demonstrate visually engaging and responsive user interfaces, prioritizing accessibility considerations.

Section 3: Oral (10 Marks)

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
MNP	CEMNP402	MINI PROJECT-1B	01

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct investigations of complex problems
5. PO5: Engineering Tool Usage
6. PO6: The Engineer & the world
7. PO7: Ethics
8. PO8: Individual & Collaborative team work
9. PO9: Communication
10. PO10: Project Management & Finance
11. PO11: Life-long learning

Course Objectives

1. To familiarize students about available infrastructure at Department/Institute level, online resources, plagiarism, expectations from MP 1A and 1B, etc.
2. To guide students in identifying societal or research needs and formulating them into problem statements.
3. To facilitate problem-solving in group settings.
4. To apply basic engineering principles to address identified problems.
5. To foster self-learning and research skills.

Course Outcomes

At the end of the course, learner will be able to:

1. Identify problems based on societal or research needs and methodology for solving them.
2. Apply knowledge and skills to solve societal problems collaboratively.
3. Develop interpersonal skills necessary for teamwork.
4. Analyze, verify, and validate results effectively through various methodologies, including, test cases/benchmark data/theoretical/inferences/experiments/simulations, etc.
5. Evaluate the societal and environmental impacts of proposed solutions.
6. Adhere to standard engineering practices.
7. Excel in written and oral communication by technical report writing, oral presentation, and publishing results in
   - Research/white paper/article/blog writing/publication, etc.
   - Business plan for entrepreneurship product creation
   - Patent filing/copyright.
8. Gain technical competencies by participating in competitions, hackathons, etc.
9. Demonstrate lifelong learning capabilities through self-directed group projects.
10. Apply project management principles effectively.

Guidelines for the Mini Project

- At the beginning of semester-III, project guides are required to conduct around 4 hours' orientation sessions including following topics:
  - Familiarizing students about infrastructure available at Department/Institute level and how to use it.
  - How to identify societal problems and formulate project problem statement.
  - How to carry out literature survey.
  - What is plagiarism and what care needs to be taken while writing a report.
  - What is project report template and how it should be used.
  - What are expectations from mini-projects 1A and 1B.
- Mini project may be carried out in one or more form of following:
  Product preparations, prototype development model, fabrication of set-ups, laboratory experiment development, process modification/development, simulation, software development, integration of software (frontend-backend) and hardware, statistical data analysis, creating awareness in society/environment etc.
- Students must form groups of 3 to 4 members either from the same or from different departments.
- Groups should conduct surveys to identify needs and develop problem statements in consultation with faculty.
- An implementation plan in Gantt/PERT/CPM chart format covering weekly activities must be submitted.
- Each group must maintain a logbook to record weekly progress, to be verified by the faculty supervisor.
- Faculty input should emphasize guiding by faculty and self-learning by group members.
- Groups should propose multiple solutions, select the best one in consultation with the supervisor, and develop a working model.
- The solution to be validated with proper justification and report to be compiled in standard format of the Institute. Research papers, competition certificates may be submitted as part of annexure to the report.
- With the focus on self-learning, innovation, addressing societal/research/innovation problems and entrepreneurship quality development within the students through the Mini Projects, it is preferable that a single project of appropriate level and quality be carried out in two semesters by all the groups of the students.
- 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 a case-by-case basis.

In-Semester Continuous Assessment and End-Semester Examination Guidelines

- The Head of the Departments will assign a guide to each of the mini-projects and shall form a progress monitoring committee. The guide will carry out weekly monitoring of the project's progress. The committee shall carry out in-semester project evaluation based on presentations with a minimum of two evaluations per semester.
- Assessment will be based on individual contributions, understanding, and responses to questions asked.

- Continuous Assessment marks distribution in semester IV (50 marks):
    - 15 marks for the In-Semester Two Presentations
    - 05 marks for the Participation in Project Competitions, TPP, etc.
    - 25 marks for the Final Report & Presentation
    - 05 marks for Regularity and active participation

The review/progress monitoring committee will assess projects based on the following criteria.

Semester IV:

- Expected tasks include procuring components/systems, constructing a working prototype, and validating results based on prior semester work.
- Reviews will be conducted as follows:
    - The first review will assess the readiness to build a working prototype.
    - The second review will involve a poster presentation and demonstration of the working model in the last month of the semester.

In addition to the above-mentioned points, the following performance criteria shall be included during the in-semester continuous assessment:

1. Quality of survey and need identification.
2. Clarity and innovativeness in problem definition and solutions.
3. Requirement gathering feasibility study, cost-effectiveness, and societal impact of proposed solutions.
4. Completeness and full functioning of the working model.
5. Effective use of skill sets and engineering norms.
6. Verification & validation of the solutions/test cases.
7. Individual contributions to the group.
8. Clarity in written and oral communication.
9. Participation in technical paper presentations/project competitions/hackathon competitions, etc.

End-Semester Examination in Semester IV (50 marks):

1. Presentation and demonstration to internal and external examiners: 20 marks.
2. Emphasis on problem clarity, innovativeness, societal impact, functioning of the model, skill utilization, and communication clarity: 30 marks.

Course Type	Course Code	Course Name	Credits
VEC	VEC402	ENVIRONMENT AND SUSTAINABILITY	02

Program Outcomes addressed:

1. PO2 : Problem Analysis
2. PO6 : The Engineer & the World
3. PO7 : Ethics
4. PO11: Life-long learning

Course Objectives:

1. To provide students with foundational knowledge and understanding of environmental science principles and concepts.
2. To explore the principles of sustainability and their applications in various domains of engineering and technology.
3. To familiarize students with the legal and ethical considerations associated with environmental management and sustainability practices.
4. To equip students with practical skills and strategies for promoting renewable energy, energy efficiency, waste management, and environmental impact assessment.

Module	Details
01.	**Foundations of Environmental Sciences** Introduction to Environmental Science, Earth's Systems: Atmosphere, Hydrosphere, Lithosphere, Biosphere, Ecological Principles: Energy flow, Nutrient cycling, Biodiversity, Environmental Degradation: Pollution, Deforestation, Habitat loss, Environmental Monitoring and Data Analysis.
02.	**Sustainability Basics** Concepts of Sustainability and Sustainable Development, Sustainable Resource Management: Water, Air, Land, Sustainable Agriculture and Food Systems, Sustainable Transportation and Urban Planning, Sustainable Business Practices and Corporate Social Responsibility.
03.	**Legal & Ethical Considerations** Environmental Laws and Regulations: National and International Perspectives, Environmental Policies and Governance Frameworks, Ethical Issues in Environmental Decision Making, Environmental Justice and Equity, Corporate Ethics and Environmental Responsibility.
04.	**Renewable energy & Energy efficiency** Introduction to Renewable Energy Sources: Solar, Wind, Hydro, Biomass, Geothermal, Energy Conversion Technologies and Systems. Energy Efficiency Measures and Strategies, Policy Support for Renewable Energy Deployment, Economic and Environmental Impacts of Renewable Energy.

05	**Waste management & recycling** Solid Waste Management: Collection, Treatment, Disposal, Recycling Processes and Technologies, E-waste Management and Hazardous Waste Handling, Circular Economy Principles, Waste Reduction Strategies: Source Reduction, Reuse, Repair
06.	**Environmental Impact Assessment** Introduction to Environmental Impact Assessment (EIA), EIA Process: Screening, Scoping, Impact Assessment, Mitigation, Monitoring, Methods and Tools for Impact Assessment: GIS, LCA, Risk Assessment, Case Studies of EIA in Various Sectors: Infrastructure, Energy, Mining, Construction, Role of Stakeholders in EIA Process
Total no. of hours: **30**	

Course Outcomes: A learner will be able to-

1. Gain a comprehensive understanding of key environmental science principles and their relevance to engineering disciplines.
2. Apply principles of sustainability to analyze and address environmental challenges in engineering projects and processes.
3. Demonstrate awareness of legal and ethical considerations in environmental decision-making and management practices.
4. Develop proficiency in implementing renewable energy technologies and energy-efficient practices in engineering designs and operations.
5. Acquire knowledge and skills in waste management, recycling, and circular economy principles for sustainable resource utilization.
6. Apply environmental impact assessment methods to evaluate and mitigate the environmental impacts of engineering projects and activities.

Text Books:

1. Environmental Science: Toward a Sustainable Future by Richard T. Wright and Dorothy F. Boorse (Publisher: Pearson Education)
2. Introduction to Environmental Engineering and Science by Gilbert M. Masters and Wendell P. Ela (Publisher: Pearson Education)
3. Renewable and Efficient Electric Power Systems by Gilbert M. Masters (Publisher: Wiley)

Reference Books:

1. Environmental Law Handbook by Thomas F. P. Sullivan, David R. Buente Jr., and Sally Fairfax, Bernan Press
2. Sustainability Science by Bert J. M. de Vries, Springer
3. Environmental Impact Assessment: Theory and Practice by Peter Wathern, Routledge

Other Resources:

1. NPTEL Course: Introduction to Environmetal Engineering & Science- Fundamental & Sustainability Concepts, Prof.Brajesh Kumar Dubey, Department of Multidisciplinary IIT Kharagpur :-Web link https://archive.nptel.ac.in/courses/127/105/127105018/
2. NPTEL Course: Environment And Development, By Prof. Ngamjahao Kipgen, IIT Guwahati, Web link- https://onlinecourses.nptel.ac.in/noc23_hs133/preview