R 25

Indian in Character International in Excellence

SYLLABUS BOOK

Department of

COMPUTER SCIENCE & ENGINEERING (CSE)

Bachelor of Technology (B.Tech)
(R25 Regulations)

Applicable to Regular students admitted from A.Y. 2025–26 onwards and Lateral Entry students from A.Y. 2026–27 onwards.

Up to II B.Tech, II Semester

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

Medbowli, Meerpet, Hyderabad - 500097, Telangana, India

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Approved by AICTE, Affiliated to JNTUH, Accredited by NBA and NAAC

Medbowli, Meerpet, Balapur, Hyderabad, Telangana- 500097

Mob: 9393959597. Email: info@tkrec.ac.in, deanacademics@tkrec.ac.in

B.Tech. in COMPUTER SCIENCE AND ENGINEERING
Course Structure &Syllabus(R-25 Regulations)
Applicable from AY 2025-2026 Batch

I YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1.	25MA101BS	Matrices and Calculus	3	1	0	4
2.	25CH102BS	Engineering Chemistry	3	0	0	3
3.	25EN103HS	English for Skill Enhancement	3	0	0	3
4.	25EC104ES	Electronic Devices and Circuits	3	0	0	3
5.	25CS105ES	Programming for Problem Solving	3	0	0	3
6.	25CH106BS	Engineering Chemistry Lab	0	0	2	1
7.	25CS107ES	Programming for Problem Solving Lab	0	0	2	1
8.	25EN108HS	English Language and Communication Skills Lab	0	0	2	1
9.	25ME109ES	Engineering Workshop	0	0	2	1
10.		Induction Program				
		Total Credits	**15**	**1**	**8**	**20**

I YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1.	25MA201BS	Ordinary Differential Equations and Vector Calculus	3	0	0	3
2.	25PH202BS	Advanced Engineering Physics	3	0	0	3
3.	25ME203ES	Engineering Drawing and Computer Aided Drafting	2	0	2	3
4.	25EE204ES	Basic Electrical Engineering	3	0	0	3
5.	25CS205ES	Data Structures	3	0	0	3
6.	25PH206BS	Advanced Engineering Physics Lab	0	0	2	1
7.	25CS207ES	Data Structures Lab	0	0	2	1
8.	25CS208ES	Python Programming Lab	0	0	2	1
9.	25EE209ES	Basic Electrical Engineering Lab	0	0	2	1
10.	25CS210ES	IT Workshop	0	0	2	1
		Total Credits	**14**	**0**	**12**	**20**

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Approved by AICTE, Affiliated to JNTUH, Accredited by NBA and NAAC

Medbowli, Meerpet, Balapur, Hyderabad, Telangana- 500097

Mob: 9393959597. Email: info@tkrec.ac.in, deanacademics@tkrec.ac.in

II YEAR I SEMESTER

S.No.	Course Code	Course Title	L	T	P	Credits
1.	25CS301PC	Discrete Mathematics	3	0	0	3
2.	25CS302PC	Computer Organization and Architecture	3	0	0	3
3.	25CS303PC	Object Oriented Programming through java	3	0	0	3
4.	25CS304PC	Software Engineering	3	0	0	3
5.	25CS305PC	Database Management Systems	3	0	0	3
6.	25MS306HS	Innovation and Entrepreneurship	2	0	0	2
7.	25CS307PC	Object Oriented Programming through java Lab	0	0	2	1
8.	25CS308PC	Software Engineering Lab	0	0	2	1
9.	25CS309PC	Database Management Systems Lab	0	0	2	1
10.	25CS310SD	Node Js/React JS/Django	0	0	2	1
11.	25VA300ES	Environmental Science	1	0	0	1
		Total Credits	**18**	**0**	**08**	**22**

II YEAR II SEMESTER

S.No.	Course Code	Course Title	L	T	P	Credits
1.	25MA401PC	Computer oriented Statistical Methods	3	0	0	3
2.	25CS402PC	Operating Systems	3	0	0	3
3.	25CS403PC	Algorithm design and Analysis	3	0	0	3
4.	25CS404PC	Computer Networks	3	0	0	3
5.	25CS405PC	Machine Learning	3	0	0	3
6.	25MA406PC	Computational Mathematics Lab	0	0	2	1
7.	25CS407PC	Operating Systems Lab	0	0	2	1
8.	25CS408PC	Computer Networks Lab	0	0	2	1
9.	25CS409PC	Machine Learning Lab	0	0	2	1
10.	25CS410SD	Data Visualization-R/Python/Power BI	0	0	2	1
		Total Credits	**15**	**0**	**10**	**20**

COURSE CONTENT

MATRICES AND CALCULUS								
I Semester: EEE / ECE / CSE / IT / CSE (AI & ML) / CSE (DS)					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA101BS	Foundation	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Basic Principles of Algebra and Calculus								

I. COURSE OVERVIEW:

This course provides a comprehensive introduction to Matrices & Caluculus, covering fundamental concepts, techniques and applications in Mathematics, Physics, Engineering and other fields.

II. COURSE OBJECTIVES:

The students will try to learn:

I Applying basic operations on matrices and their properties.
II Concept of a rank of the matrix and applying this concept to know the consistency and solving the system of linear equations.
III Concept of eigen values and eigen vectors and to reduce the quadratic form to canonical form
IV Geometrical approach to the mean value theorems and their application to the mathematical problems
V Finding maxima and minima of functions of two and three variables.
VI Evaluation of multiple integrals and their applications.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Write the matrix representation of a set of linear equations and to analyse the solution of the system of equations
CO2:	Find the Eigen values and Eigen vectors and Reduce the quadratic form to canonical form using orthogonal transformations.
CO3:	Apply the mean value theorems and Studying curve tracing in Cartesian coordinates.
CO4:	Find the extreme values of functions of two variables with/ without constraints.
CO5:	Evaluate the multiple integrals and apply the concept to find areas, volumes

IV. COURSE ARTICULATION MATRIX

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11
CO1	3	3		1							
CO2	2	2		1							
CO3	2	2									
CO4	2	2		1	2						
CO5	1	1									
	2.0	**2.0**		**1.0**	**2.0**						

V. COURSE CONTENT:

UNIT- I: (10 Hrs)

Matrices

Rank of a matrix by Echelon form and Normal form — Inverse of Non-singular matrices by Gauss- Jordan method. System of linear equations: Solving system of Homogeneous and Non-Homogeneous equations. Gauss Seidel Iteration Method.

UNIT – II: (10 Hrs)

Eigen values and Eigen vectors

Linear Transformation and Orthogonal Transformation: Eigen values — Eigen vectors and their properties – Diagonalization of a matrix – Cayley-Hamilton Theorem (without proof) – Fnding inverse and power of a matrix by Cayley-Hamilton Theorem. Quadratic forms and Nature of the Quadratic Forms – Reduction of Quadratic form to canonical form by Orthogonal Transformation

UNIT - III: (10 Hrs)

Single Variable Calculus

Limit and Continuous of functions and its properties. Mean value theorems: Rolle's theorem — Lagrange's Mean value theorem with their Geometrical Interpretation and applications — Cauchy's Mean value Theorem — Taylor's Series (All the theorems without proof).

Curve Tracing: Curve tracing in cartesian coordinates.

UNIT - IV: (10 Hrs)

Multivariable Calculus (Partial Differentiation and applications)

Definitions of Limit and continuity — Partial Differentiation: Euler's Theorem — Total derivative — Jacobian – Functional dependence & independence. Applications: Maxima and minima of functions of two variables and three variables using method of Lagrange multipliers.

UNIT - V: (10 Hrs)

Multivariable Calculus (Integration)

Evaluation of Double Integrals (Cartesian and polar coordinates) – change of order of integration (only Cartesian form) — Change of variables for double integrals (Cartesian to polar). Evaluation of Triple Integrals — Change of variables for triple integrals (Cartesian to Spherical and Cylindrical polar coordinates). Applications: Areas by double integrals and volumes by triple integrals.

VI. TEXT BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 45th Edition, 2024-25.
2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publications, 10th Editon, 2020.

VII. REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9thEdition, Pearson, Reprint, 2002.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
4. H. K. Dass and Er. Rajnish Verma, Higher Engineering Mathematics, S Chand and Company Limited, New Delhi.

COURSE CONTENT

ENGINEERING CHEMISTRY								
I Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CH102BS	**Basic Sciences**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic understanding of level chemistry concepts								

I. COURSE OVERVIEW:

The Engineering Chemistry course provides the foundational chemical principles and applications essential for all branches of engineering. It bridges the gap between pure chemistry and engineering practices by explaining how chemical processes, materials, and technologies influence modern engineering systems.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To develop adaptability to new advances in Engineering Chemistry and acquire the essential skills to become a competent engineering professional.
2. To understand the industrial significance of water treatment, fundamental principles of battery chemistry, and the impact of corrosion along with its control methods for structural protection.
3. To impart foundational knowledge of various energy sources and their practical applications in engineering.
4. To equip students with an understanding of smart materials, biosensors, and analytical techniques applicable in engineering, industrial, environmental, and biomedical fields.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Basic properties of water and its treatment methods for both domestic and industrial use.
CO2:	Basic understanding of electrochemical processes, including corrosion and its control mechanisms.
CO3:	Comprehend principles and applications of various energy sources, such as batteries and fuels, relevant to engineering.
CO4:	Basic concepts and properties and applications of different types of polymers and engineering materials.
CO5:	Apply the principles of spectroscopic techniques (UV-Vis, IR, Raman) for material analysis and pollution detection.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2	3		1	2	-	-	-	-	-	1		
CO2	2				2	2	-	-	2	3			
CO3	2	2	-		3	-	-	-	-	2	1		
CO4	1	2	-		1	-	1	-	2	3			
CO5	2		2				1	-	-	2	3		

V. COURSE CONTENT:

UNIT- I: (12 Hrs)

Water and its treatment (14 Hrs)

Introduction- Hardness, types, degree of hardness and units. Estimation of hardness of water by complexometric method - Numerical problems. Potable water and its specifications (WHO) - Steps involved in the treatment of potable water - Disinfection of potable water by chlorination and break-point chlorination. Defluoridation - Nalgonda technique.

Boiler troubles: Scales, Sludges and Caustic embrittlement. Internal treatment of boiler feed water - Calgon conditioning, Phosphate conditioning, Colloidal conditioning. External treatment methods - Softening of water by ion- exchange processes. Desalination of brackish water – Reverse

UNIT – II: (12 Hrs)

Electrochemistry and Corrosin (12 Hrs)

Introduction- Electrode potential, standard electrode potential, Nernst equation (no derivation), electro-chemical cell - Galvanic cell, cell representation, EMF of cell - Numerical problems. Types of electrodes, reference electrodes - Primary reference electrode - Standard Hydrogen Electrode (SHE), Secondary reference electrode – Calomel electrode. Construction, working and determination of pH of unknown solution using SHE and Calomel electrode.

Corrosion:

Introduction- Definition, causes and effects of corrosion – Theories of corrosion, chemical and electrochemical theories of corrosion, Types of corrosion: galvanic, water-line and pitting corrosion. Factors affecting rate of corrosion - Nature of the metal, Nature of the corroding environment. Corrosion control methods - Cathodic protection Methods - Sacrificial anode and impressed current methods.

UNIT - III: (12 Hrs)

Energy sources:

Batteries: Introduction – Classification of batteries - Primary, secondary and reserve batteries with examples. Construction, working and applications of Zn-air and Lithium ion battery. Fuel Cells – Differences between a battery and a fuel cell, Construction and applications of Direct Methanol Fuel Cell (DMFC).

Fuels: Introduction and characteristics of a good fuel, Calorific value – Units - HCV, LCV- Dulongs **formula – Numerical problems.**

Fossil fuels: Introduction, Classification, Petroleum - Refining of Crude oil, Cracking - Types of cracking

Synthetic Fuels – Moving bed catalytic cracking. LPG and CNG composition and uses.

: Fischer-Tropsch process, Introduction and applications of Hythane and Green Hydrogen.

UNIT - IV: (12 Hrs)

Polymers

Definition - Classification of polymers: Based on origin and tacticity with examples –Types of polymerization-Addition (free radical addition mechanism) and condensation polymerization. **Plastics, Elastomers and Fibers:** Definition and applications (PVC, Buna-S, Nylon-6,6). Differences between thermoplastics and thermo setting plastics, Fiber reinforced plastics (FRP).

Conducting polymers: Definition and Classification with examples - Mechanism of conduction in trans-poly-acetylene and applications of conducting polymers.

Biodegradable polymers: Polylactic acid and its applications.

UNIT - V: (12 Hrs)

Advanced Functional Materials:

Smart materials: Introduction, Classification with examples - Shape Memory Alloys – Nitinol,- Piezo-electric materials – quartz and their engineering applications.

Biosensor - Definition, Amperometric Glucose monitor sensor. Interpretative spectroscopic applications of UV-Visible spectroscopy for Analysis of pollutants in dye industry, IR spectroscopy in night vision-security, Pollution Under Control- CO sensor (Passive Infrared detection), Raman spectroscopy (application) - Tumour detection in Medical applications.

VI. TEXT BOOKS:

1. Engineering Chemistry by P.C. Jain and M. Jain, Dhanpatrai Publishing Company, 2010.
2. Engineering Chemistry by Rama Devi, Dr. P. Aparna and Rath, Cengage learning, 2025.

VII. REFERENCE BOOKS:

1. Engineering Chemistry: by Thirumala Chary Laxminarayana & Shashikala, Pearson Publications (2020)
2. Engineering Chemistry by Shashi Chawla, Dhanpatrai and Company (P) Ltd. Delhi 2011.
3. Engineering Chemistry by Shikha Agarwal, Cambridge University Press, Delhi 2015.
4. Engineering Analysis of Smart Material Systems by Donald J. Leo, Wiley, 2007.
5. Challenges and Opportunities in Green Hydrogen by Editors: Paramvir Singh, Avinash Kumar Agarwal, Anupma Thakur, R.K Sinha.
6. Raman Spectroscopy in Human Health and Biomedicine, https://www.worldscientific.com/doi/epdf/10.1142/13094

VIII. ELECTRONICS RESOURCES:

E-Content- https://doi.org/10.1142/13094 | October 2023
E-books: https://archive.org/details/EngineeringChemistryByShashiChawla/page/n11/mode/2u

COURSE CONTENT

ENGLISH FOR SKILL ENHANCEMENT								
I Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EN103HS	**Humanities & Sciences**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic understanding of vocabulary, grammar, reading, and writing skills in English								

I. COURSE OVERVIEW:

English for Skill Enhancement focuses on skill development, fostering ideas and practice of language skills in various contexts and cultures in the areas of vocabulary, grammar, reading and writing.

II. COURSE OBJECTIVES:

This course will enable the students to:

I. Improve their vocabulary.
II. Use appropriate sentence structures in their oral and written communication.
III. Develop their reading and study skills.
IV. Equip students to write paragraphs, essays, précis and draft letters.
V. Acquire skills for Technical report writing

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Choose appropriate vocabulary in their oral and written communication.
CO2:	Demonstrate their understanding of the rules of functional grammar and sentence structures.
CO3:	Develop comprehension skills from known and unknown passages.
CO4:	Write paragraphs, essays, précis and draft letters.
CO5:	Write abstracts and reports in various contexts.

IV. COURSE CONTENT:

UNIT- I: (12 Hrs)

Theme:	Perspectives **Lesson on 'The Generation Gap' by Benjamin M. Spock from the prescribed textbook titled *English for the Young in the Digital World* published by Orient Black Swan Pvt. Ltd.**
Vocabulary:	The Concept of Word Formation-The Use of Prefixes and Suffixes-Words Often Misspelt - Synonyms and Antonyms
Grammar:	Identifying Common Errors in Writing with Reference to Parts of Speech particularly Articles and Prepositions – Degrees of Comparison
Reading:	Reading and Its Importance-Sub Skills of Reading–Skimming and Scanning.
Writing:	Sentence Structures and Types -Use of Phrases and Clauses in Sentences-Importance of Proper Punctuation-Techniques for Writing Precisely–Nature and Style of Formal Writing.

UNIT – II: (12 Hrs)

Theme:	Digital Transformation **Lesson on '*Emerging Technologies*' from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.**
Vocabulary:	Homophones, Homonyms and Homographs
Grammar:	Identifying Common Errors in Writing with Reference to Noun-pronoun Agreement and Subject-verb Agreement.
Reading:	Reading Strategies-Guessing Meaning from Context–Identifying Main Ideas–Exercises for Practice
Writing:	Paragraph Writing – Types, Structures and Features of a Paragraph - Creating Coherence – Linkers and Connectives - Organizing Principles in a Paragraph – Defining- Describing People, Objects, Places and Events – Classifying- Providing Examples or Evidence - Essay Writing - Writing Introduction and Conclusion.

UNIT - III: (12 Hrs)

Theme:	Attitude and Gratitude **Poems on '*Leisure*' by William Henry Davies and '*Be Thankful*' -Unknown Author from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.**
Vocabulary:	Words Often Confused-Words from Foreign Languages and their Use in English.
Grammar:	Identifying Common Errors in Writing with Reference to Misplaced Modifiers and Tenses.
Reading:	Sub-Skills of Reading – Identifying Topic Sentence and Providing Supporting Ideas – Exercises for Practice
Writing:	Format of a Formal Letter-Writing Formal Letters E.g.., Letter of Complaint, Letter of Requisition, Job Application with CV/Resume–Difference between Writing a Letter and an Email - Email Etiquette

UNIT - IV: (12 Hrs)

Theme:	Entrepreneurship **Lesson on '*Why a Start-Up Needs to Find its Customers First*' by Pranav Jain from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.**
Vocabulary:	Standard Abbreviations in English – Inferring Meanings of Words through Context – Phrasal Verbs – Idioms.
Grammar:	Redundancies and Clichés in Written Communication – Converting Passive to Active Voice and Vice-Versa.
Reading:	Prompt Engineering Techniques–Comprehending and Generating Appropriate Prompts - Exercises for Practice
Writing:	Writing Practices-Note Making – Précis Writing.

UNIT - V: (12 Hrs)

Theme:	Integrity and Professionalism **Lesson on '*Professional Ethics*' from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.**
Vocabulary:	Technical Vocabulary and their Usage–One Word Substitutes–Collocations.
Grammar:	Direct and Indirect Speech-Common Errors in English (Covering all the other aspects of grammar which were not covered in the previous units)
Reading:	Survey, Question, Read, Recite and Review (SQ3R Method) – Inferring the Meaning and Evaluating a Text- Exercises for Practice
Writing:	Report Writing- Technical Reports-Introduction–Characteristics of a Report – Categories of Reports Formats- Structure of Reports (Manuscript Format)-Types of Reports-Writing a Technical Report.

Note: ***Listening and Speaking skills which are given under Unit-6 in AICTE Model Curriculum are covered in the syllabus of ELCS Lab Course.***

- (**Note**: As the syllabus of English given in AICTE *Model Curriculum-2018 for B.Tech. First Year is* Open-ended, besides following the prescribed textbook; it is required to prepare teaching/learning materials by the teachers collectively in the form of handouts based on the needs of the students in their respective colleges for effective teaching/learning in the class.)

V. TEXT BOOKS:

1. Board of Editors.*2025. English for the Young in the Digital World.* Orient Black Swan Pvt. Ltd.

VI. REFERENCE BOOKS:

1. Swan,Michael.(2016).*Practical English Usage.*Oxford University Press. New Edition.
2. *Karal, Rajeevan.2023 . English Grammar Just for You.* Oxford University Press. New Delhi
3. 2024.*Empowering with Language: Communicative English for Undergraduates.* Cengage Learning India Pvt. Ltd. New Delhi
4. Sanjay Kumar & Pushp Lata. 2022. *Communication Skills – A Workbook.* Oxford University Press. New Delhi
5. Wood ,F.T.(2007). *Remedial English Grammar.* Macmillan.
6. Vishwamohan, Aysha.(2013).*English for Technical Communication for Engineering Students.* McGraw-Hill Education India Pvt. Ltd.

COURSE CONTENT

ELECTRONIC DEVICES AND CIRCUITS								
I Semester: C S E					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EC104ES	**Engineering& Sciences**	3	0	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes:**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic Electrical and Electronics knowledge								

I. COURSE OVERVIEW:

This course introduces fundamental semiconductor devices and their behavior, including diodes, BJTs, and FETs. It covers their characteristics, applications, and the analysis of basic electronic circuits. The course also explores rectifiers, voltage regulation, amplifier design, and advanced semiconductor technologies like FinFETs and CNTFETs. Emphasis is placed on developing a strong foundation for analog circuit design and understanding modern device technologies in electronics.

II. COURSE OBJECTIVES:

The students will try to learn:

I To understand the characteristics of semiconductor diodes and its applications.
II To understand the operations and Characteristics of BJT configurations.
III To analyze the Biasing techniques for BJT.
IV To analyze the transistor parameters in various configurations.
V To acquire the knowledge of special purpose electronic devices and their application

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Analyze the electrical characteristics and models of semiconductor diodes and apply them in rectifier and clipping circuits.
CO2:	Evaluate the operation and configurations of Bipolar Junction Transistors (BJTs) and analyze their input and output characteristics.
CO3:	Evaluate the various biasing networks for BJTs and determine the operating point for amplifier
CO4:	Analyze transistor amplifier circuits using h-parameter models and assess performance for various configurations.
CO5:	Analyze the structure, working, and characteristics of JFETs, MOSFETs, and advanced devices like Fin FETs and CNTFETs, and acquire the knowledge of special purpose devices.

IV. COURSE CONTENT:

UNIT- I: (10 Hrs)

Diode Characteristics and Applications: PN junction diode – I-V characteristics, Diode resistance and capacitance, Diode models (Ideal, Simplified, Piecewise Linear), Rectifiers – Half-wave, Full wave (Center-tap and bridge), Capacitor filter for rectifiers, Clippers and clampers, Zener diode – I-V characteristics and voltage regulation

UNIT – II: (14 Hrs)

Bipolar Junction Transistor (BJT): Structure and working principle of BJT, Current components and transistor action, Configurations: Common Base (CB), Common Emitter (CE), Common Collector (CC), Input and output characteristics, Determination of h-parameters from transistor characteristics

UNIT - III: (12 Hrs)

BJT Biasing: Need for biasing and stabilization, Load line and operating point, Biasing techniques: Fixed bias, Collector-to-base bias, Voltage divider bias, Stability factors and thermal runaway

UNIT - IV: (14 Hrs)

Transistor Amplifiers: Transistor as a small-signal amplifier, h-parameter equivalent circuit, CE, CB, CC amplifier analysis using h-parameters, Approximate CE model – with and without emitter bypass capacitor.

UNIT - V: (10)

Special Purpose Diodes: Principle of Operation of – SCR, Tunnel Diode, Varactor Diode, Photo Diode, Solar Cell, LED and Schottky Diode

Field Effect Transistors and Advanced Devices: JFET: Structure, operation, and characteristics, **MOSFET:** Enhancement and Depletion modes – Structure, operation, and characteristics, Introduction of Advanced Devices: FinFETs and CNTFETs

V. TEXT BOOKS:

1. Millman, Jacob, and Christos C. Halkias. Electronic Devices and Circuits. Tata McGraw-Hill, 1991.
2. Boylestad, Robert L., and Louis Nashelsky. Electronic Devices and Circuit Theory. Pearson, 11th ed., 2013.
3. Sedra, Adel S., and Kenneth C. Smith. Microelectronic Circuits. Oxford University Press, 7th ed., 2014.

VI. REFERENCE BOOKS:

1. Bell, David A. Electronic Devices and Circuits. Oxford University Press, 5th ed., 2008.
2. Neamen, Donald A. Electronic Circuit Analysis and Design. McGraw-Hill, 2nd ed., 2001.
3. Salivahanan, S., and N. Suresh Kumar. Electronic Devices and Circuits. McGraw-Hill Education, 4th ed., 2017.
4. Razavi, Behzad. Fundamentals of Microelectronics. Wiley, 2nd ed., 2013.
5. Taur, Yuan, and Tak H. Ning. Fundamentals of Modern VLSI Devices. Cambridge University Press, 2nd ed., 2009.

COURSE CONTENT

PROGRAMMING FOR PROBLEM SOLVING								
I Semester: EEE / ECE / CSE / IT / CSE (AI & ML) / CSE (DS)					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS105ES	Foundation	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic Mathematics Fundamentals and Computer Knowledge								

I. COURSE OVERVIEW:

This course is Programming for Problem Solving is a foundational, general-purpose language developed in the early 1972s, known for its efficiency, portability, and influence on modern languages like C++, Java, and Python. The course assumes no prior programming experience and focuses on building core skills through hands-on coding, debugging, and problem-solving. By the end, learners can write, compile, and run simple to moderately complex programs

I. COURSE OBJECTIVES:

The students will try to learn:

I To learn the fundamentals of computers.
II To understand the various steps in program development.
III To learn the syntax and semantics of the C programming language.
IV To learn the usage of structured programming approaches in solving problems.

II. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	To write algorithms, to draw flowcharts for solving problems and to convert the algorithms/flowcharts to C programs.
CO2:	To code and test a given logic in the C programming language.
CO3:	To decompose a problem into functions and to develop modular reusable code.
CO4:	To use arrays, pointers, strings and structures to write C programs.
CO5:	Searching and sorting problems.

III. COURSE CONTENT:

UNIT- I:

Overview of C:C Language Elements, Variable Declarations and Data Types, Creating, Compiling and Executing a program, General Form of a C Program, Operators, Operator Precedence, Formatting Numbers in Program Output, Type Conversions.

Selection Structures: Control Structures, Conditions, if Statement, if Statements with Compound Statements, Decision Steps in Algorithms.

Repetition and Loop Statements: Repetition in Programs, Counting Loops and the while Statement, Computing a Sum or Product in a Loop, for Statement, Conditional Loops, Loop Design, Nested Loops, do-while Statement.

UNIT – II:

Arrays: Declaring and Referencing Arrays, Array Subscripts, Using for Loops for Sequential Access, Using Array Elements as Function Arguments, Array Arguments, Searching and Sorting an Array, Multidimensional Arrays, Arrays of Pointers.

Strings: String Basics, String Library Functions: Assignment and Substrings, Longer Strings: Concatenation and Whole- Line Input, String Comparison.

UNIT - III:

Top-Down Design with Functions: Building Programs from Existing Information, Library Functions, Top-Down Design and Structure Charts, Functions without Arguments, Functions with Input Arguments.

Pointers and Modular Programming: Pointers and the Indirection Operator, Functions with Output Parameters, Multiple Calls to a Function with Input/ Output Parameters, Scope of Names, Formal Output Parameters as Actual Arguments, Dynamic Memory Allocation.

UNIT - IV:

Recursion: The Nature of Recursion, Tracing a Recursive Function, Recursive Mathematical Functions, Recursive Functions with Array and String Parameters

Structure and Union Types: User-Defined Structure Types, Structure Type Data as Input and Output Parameters, Functions with Structured Result Values, Union Types, Enumerated Types.

UNIT - V

Text and Binary File Pointers: Input/ Output Files - Review and Further Study, Binary Files.

Searching and Sorting: Basic searching in an array of elements (linear and binary search techniques), Basic algorithms to sort array of elements (Bubble, Insertion and Selection sort algorithms).

IV. TEXT BOOKS:

1. Jeri R. Hanly and Elliot B. Koffman, Problem solving and Program Design in C 7th Edition, Pearson.
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition).

V. REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.
2. E. Balagurusamy, Computer fundamentals and C, 2nd Edition, McGraw-Hill.
3. Yashavant Kanetkar, Let Us C, 18th Edition, BPB.
4. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression).
5. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
6. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4th Edition.
7. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

VI. ELECTRONICS RESOURCES:

1. https://www.geeksforgeeks.org/c-programming-language/
2. https://www.tutorialspoint.com/cprogramming/
3. https://www.programiz.com/c-programming
4. https://www.w3schools.com/c/c_intro.php

COURSE CONTENT

ENGINEERING CHEMISTRY LAB								
I Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CH106BS	**Basic Sciences**	-	-	2	1	40	60	100
Contact Classes: 0	**Tutorial Classes: 0**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic lab handling skills in introductory chemistry courses								

I. COURSE OVERVIEW:

Engineering Chemistry Lab is designed to give students hands-on experimental experience to reinforce and complement the theoretical concepts learned in this course. It helps bridge the gap between theory and practice, by enabling students to perform chemical analyses relevant to engineering applications.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Students will understand and perform experiments based on core chemical principles relevant to engineering applications.
2. Students will learn to estimate the hardness of water to assess its suitability for drinking purposes.
3. Students will acquire the ability to perform acid-base titrations using instrumental methods such as conductometry, potentiometry, and pH metry.
4. Students will gain hands-on experience in synthesizing polymers like Bakelite and Nylon 6, 6 in the laboratory.
5. Students will learn to determine the unknown concentration of potassium permanganate ($KMnO_4$) using a calibration curve.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Perform volumetric and instrumental analysis to estimate the concentration of chemical species in solutions.
CO2:	To determine important parameters such as water hardness and the corrosion rate of mild steel under various conditions.
CO3:	Applying techniques like conductometry, potentiometry, and pH metry to determine concentrations or equivalence points in acid-base reactions.
CO4:	To synthesizing polymers like Bakelite and Nylon – 6, 6 in the laboratory.
CO5:	To understand the working principle of colorimetry and the relationship between absorbance and concentration (Beer-Lambert Law).

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2	3		1	2	-	-	-	-	-	1		
CO2	2				2	2	-	-	2	3			
CO3	2	2	-		3	-	-	-	-	2	1		
CO4	1	2	-		1	-	1	-	2	3			
CO5	2		2				1	-	-	2	3		

V. COURSE CONTENT:

List of Experiments:

I. Volumetric Analysis: Estimation of Hardness of water by EDTA Complexometric method.

II. Conductometry:

1. Estimation of the concentration of strong acid by Conductometry.
2. Estimation of the concentration of strong and weak acid in an acid mixture by Conductometry.

II. Potentiometry:

1. Estimation of concentration of Fe^{+2}ion by Potentiometry using $KMnO_4$.
2. Estimation of concentration of strong acid with strong base by Potentiometry using quinhydrone

III. pH Metry: Determination of an acid concentration using pH meter.

IV. Colorimetry: Verification of Lambert-Beer's law using $KMnO_4$.

V. Preparations:

1. Preparation of Bakelite.
2. Preparation Nylon – 6, 6.

VI. Corrosion: Determination of rate of corrosion of mild steel in the presence and absence of inhibitor.

VII. Virtual lab experiments:

1. Construction of Fuel cell and it's working.
2. Smart materials for Biomedical applications
3. Batteries for electrical vehicles.
4. Functioning of solar cell and its applications.

VI. REFERENCE BOOKS:

1. Lab manual for Engineering chemistry by B. Ramadevi and P. Aparna, S Chand Publications, New Delhi (2022).
2. Vogel's text book of practical organic chemistry 5th edition.

COURSE CONTENT

PROGRAMMING FOR PROBLEM SOLVING LAB								
I Semester: EEE / ECE / CSE / IT / CSE (AI & ML) / CSE (DS)					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS107ES	Engineering and Science	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic Mathematics Fundamentals and Computer Knowledge								

I. COURSE OVERVIEW:

Programming for Problem Solving Lab is intended to provide hands-on exposure to problem solving using the C programming language. This course enables students to understand the program development process and to work effectively with an Integrated Development Environment (IDE) for writing, compiling, debugging, and executing programs.

II. COURSE OBJECTIVES:

The students will try to learn:

I To work with an IDE to create, edit, compile, run and debug programs
II To analyze the various steps in program development.
III To develop programs to solve basic problems by understanding basic concepts in C like operators, control statements etc.
IV To develop modular, reusable and readable C Programs using the concepts like functions, arrays etc.
V To Write programs using the Dynamic Memory Allocation concept.
VI To create, read from and write to text and binary files

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Formulate algorithms for simple problem-solving tasks and Translate algorithms into correct and executable C programs.
CO2:	Identify and correct syntax and logical errors in programs.
CO3:	Use arrays, strings, structures, and pointers to represent and manipulate data.
CO4:	Apply modular programming concepts using functions, dynamic memory allocation, and file handling.
CO5:	Develop efficient solutions using sorting, searching, and standard problem-solving techniques.

IV. COURSE CONTENT:

PRACTICE SESSIONS:

Simple numeric problems:

a) Write a program for finding the max and min from the three numbers.
b) Write the program for the simple, compound interest.
c) Write a program that prints a multiplication table for a given number and the number of rows in the table. For example, for a number 5 and rows = 3, the output should be:
   5 x 1 = 5
   5 x 2 = 10
   5 x 3 = 15
d) Write a program that shows the binary equivalent of a given positive number between 0 to 255.

Expression Evaluation:

a) Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement).
b) Write a program that finds if a given number is a prime number.
c) Write a C program to find the sum of individual digits of a positive integer and test given number is palindrome.
d) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.

Arrays, Pointers and Functions:

a) Write a C program to find the minimum, maximum and average in an array of integers.
b) Write a C program that uses functions to perform the following:
   I. Addition of Two Matrices
   II. Multiplication of Two Matrices
c) Write a program for reading elements using a pointer into an array and display the values using the array.
d) Write a program for display values reverse order from an array using a pointer.

Files:

a) Write a C program which copies one file to another, replacing all lowercase characters with their uppercase equivalents.
b) Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file).

Strings:

a) Write a C program that uses functions to perform the following operations:
   I. To insert a sub-string into a given main string from a given position.
   II. To delete n Characters from a given position in a given string
b) Write a C program to determine if the given string is a palindrome or not (Spelled same in both directions with or without a meaning like madam, civic, noon, abcba, etc.)
c) Write a C program that displays the position of a character ch in the string S or – 1 if S doesn't contain ch.
d) Write a C program to count the lines, words and characters in a given text.

Sorting and Searching:

a) Write a C program that uses non-recursive function to search for a Key value in a given list of integers using linear search method.
b) Write a C program that uses non-recursive function to search for a Key value in a given sorted list of integers using binary search method.
c) Write a C program that implements the Bubble sort method to sort a given list of integers in ascending order.
d) Write a C program that sorts the given array of integers using selection sort in descending order
e) Write a C program that sorts the given array of integers using insertion sort in ascending order

f) Write a C program that sorts a given array of names.

V. TEXT BOOKS:

1. Jeri R. Hanly and Elliot B. Koffman, Problem solving and Program Design in C 7th Edition, Pearson.
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition).

VI. REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.
2. E. Balagurusamy, Computer fundamentals and C, 2nd Edition, McGraw-Hill.
3. Yashavant Kanetkar, Let Us C, 18th Edition, BPB.
4. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression).
5. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
6. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4th Edition.
7. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

VII. ELECTRONICS RESOURCES:

1. https://www.geeksforgeeks.org/c-programming-language/
2. https://www.tutorialspoint.com/cprogramming/
3. https://www.programiz.com/c-programming
4. https://www.w3schools.com/c/c_intro.php

COURSE CONTENT

ENGLISH LANGUAGE AND COMMUNICATION SKILLS LAB								
I Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EN108HS	**Humanities & Social Sciences**	-	-	2	1	40	60	100
Contact Classes: 0	**Tutorial Classes: 0**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic listening and speaking skills in English								

I. COURSE OVERVIEW:

The **English Language and Communication Skills (ELCS) Lab** focuses on listening and speaking skills, particularly on the production and practice of sounds of language and familiarizes the students with the use of English in everyday situations both in formal and informal contexts.

II. COURSE OBJECTIVES:

To enable students, develop their active listening skills:

I	To enable students, develop their active listening skills
II	To equip students with necessary training in listening, so that they can comprehend the speech of people from different linguistic backgrounds
III	To improve their pronunciation and neutralize accent
IV	To enable students express themselves fluently and appropriately
V	To practice speaking in social and professional contexts

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Listen actively and identify important information in spoken texts
CO2:	Interpret the speech and infer the intention of the speaker
CO3:	Improve their accent for intelligibility
CO4:	Speak fluently with clarity and confidence
CO5:	Use the language in real life situations

IV. COURSE CONTENT:

The English Language and Communication Skills Lab (ELCS) shall have two parts:

a. Computer Assisted Language Learning (CALL) Lab, which focuses on listening skills

b. Interactive Communication Skills (ICS) Lab, which focuses on speaking skills

The following course content is prescribed for the English Language and Communication Skills Lab.

Exercise– I
CALL Lab:
Instruction: Speech Sounds-Listening Skill-Importance–Purpose-Types-Barriers-Active Listening
Practice: Listening to Distinguish Speech Sounds (Minimal Pairs)-Testing Exercises
ICS Lab:
❖ **Diagnostic Test – Activity titled '*Express Your View*'**
Instruction: Spoken and Written language - Formal and Informal English - Greetings - Introducing Oneself and Others
Practice: Any Ice-Breaking Activity

Exercise–II
CALL Lab:
Instruction: Listening vs. Hearing- Barriers to Listening
Practice: Listening for General Information - Multiple Choice Questions - Listening Comprehension Exercises (It is essential to identify a suitable passage with exercises for practice.)
ICS Lab:
Instruction: Features of Good Conversation–Strategies for Effective Communication
Practice: Role Play Activity - Situational Dialogues –Expressions used in Various Situations –Making Requests and Seeking Permissions – Taking Leave - Telephone Etiquette

Exercise-III
CALL Lab:
Instruction: Errors in Pronunciation–Tips for Neutralizing Mother Tongue Influence (MTI)
Practice: Differences between British and American Pronunciation –Listening Comprehension Exercises
ICS Lab:
Instruction: Describing Objects, Situations, Places, People and Events
Practice: Picture Description Activity–Looking at a Picture and Describing Objects, Situations, Places, People and Events (A wide range of Materials / Handouts are to be made available in the lab.)

Exercise–IV CALL
Lab:
Instruction: Techniques for Effective Listening
Practice: Listening for Specific Details-Listening-Gap Fill Exercises-Listening Comprehension Exercises
(It is essential to identify a suitable passage with exercises for practice.)
ICS Lab:
Instruction: How to Tell a Good Story-Story Star-Sequencing-Creativity
Practice: Activity on Telling and Retelling Stories-Collage

Exercise–V
CALL Lab:
Instruction: Identifying the literal and implied meaning
Practice: Listening for Evaluation-Write the Summary–Listening Comprehension Exercises
(It is essential to identify a suitable passage with exercises for practice.)
ICS Lab:
Instruction: Understanding Non-Verbal Communication
Practice: Silent Speech- Dumb Charades Activity

Minimum Requirement of Infrastructural Facilities for ELCS Lab

1. Computer Assisted Language Learning (CALL) Lab

The Computer Assisted Language Learning Lab has to accommodate 40 students with 40 systems, with one master console, LAN facility, and English language learning software for self-study by students.

System Requirements (Hardware Components):

- Computer network with LAN facility (minimum 40 systems with multimedia) with the following specifications:
  i) Computers with suitable configuration
  ii) High-fidelity headphones

2. Interactive Communication Skills (ICS) Lab

The Interactive Communication Skills Lab should be a spacious room with movable chairs and audio-visual aids, a public address system, a TV or LCD, a digital stereo audio and video system, and a camcorder.

Note: <u>English language teachers are requested to prepare materials and handouts for each activity for the use of those materials in CALL and ICS Labs.</u>

Suggested Software

- Cambridge Advanced Learner's English Dictionary with CD
- Grammar Made Easy by Dorling Kindersley
- Punctuation Made Easy by Dorling Kindersley
- Oxford Advanced Learner's Compass, 10th Edition
- English in Mind (Series 1–4) by Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge

COURSE CONTENT

ENGINEERING WORKSHOP								
I Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25ME109ES	**Engineering and Science**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Practical skill								

I. COURSE OVERVIEW:

The Engineering Workshop course provides students with hands-on exposure to basic manufacturing processes, workshop practices, and fabrication techniques. It introduces fundamental skills in carpentry, fitting, welding, sheet metal work, machining, and house wiring, along with safety practices and proper handling of tools and equipment.

Through practical exercises, students gain the ability to read and interpret workshop drawings, use hand tools and measuring instruments accurately, and perform basic operations in various trades. The course also emphasizes teamwork, time management, and quality awareness, building a solid foundation for future engineering practice and industrial production environments.

II. COURSE OBJECTIVES:

The students will try to learn:

I To introduce students to basic manufacturing processes and workshop practices.
II To provide hands-on training in carpentry, fitting, welding, sheet metal, and machining
III To develop skills in using hand tools and measuring instruments.
IV To enhance safety awareness and proper handling of workshop equipment.
V To build a foundational understanding of industrial production and fabrication.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand the basic manufacturing processes and operations.
CO2:	Use hand tools and equipment safely and efficiently.
CO3:	Perform basic operations in carpentry, fitting, welding, sheet metal work, and machining
CO4:	Read and interpret workshop drawings
CO5:	Develop teamwork, time management, and quality awareness in a workshop environment.

IV. COURSE CONTENT:

PRACTICE SESSIONS:

1. TRADES FOR EXERCISES: At least two exercises from each trade:

i. **Carpentry:** T- Lap Joint, Dovetail Joint, Mortise and Tenon Joint
ii. **Fitting:** V- Fit, Dovetail Fit and Semi- circular fit
iii. **Tin Smithy**: Square Tin, Rectangular Tray and Conical Funnel
iv. **Foundry**: Preparation of Green Sand Mould using Single Piece and Split Pattern
v. **Welding Practice:** Arc Welding and Gas Welding
vi. **House wiring**: Parallel and Series, Two-way Switch and Tube Light
vii. **Black Smithy**: Round to Square, Fan Hook and S- Hook

2. TRADES FOR DEMONSTRATION AND EXPOSURE: Plumbing, Machine Shop, Metal Cutting (Water Plasma), Power tools in construction and Wood Working

V. TEXT BOOKS:

1. Workshop Practice, B. L. Juneja, Cengage Learning India, 1st edition, 2015.
2. Workshop Practice Manual, K. Venkata Reddy, BS Publication, 6th Edition, Rpt.2025.

VI. REFERENCE BOOKS:

1. Workshop Manual, K. Venugopal, Anuradha Publications, 2012th edition, 2012.

VII. ELECTRONICS RESOURCES:

1. NPTEL Online Courses – Workshop Practice:
   https://nptel.ac.in/courses/112/106/112106194/
2. MIT OpenCourseWare – Manufacturing Processes:
   https://ocw.mit.edu/courses/mechanical-engineering/2-008-design-and-manufacturing-ii-spring-2011/
3. YouTube Tutorials on Workshop Practices:
4. Carpentry, Fitting, Sheet Metal, Welding, and House Wiring
   https://www.youtube.com/results?search_query=engineering+workshop+practice
5. Online Resources for Machine Shop and Metal Cutting:
   https://www.engineeringtoolbox.com/machining-tools-d_1214.html
6. Plumbing and Electrical Basics:
   https://www.electricaltechnology.org/
   https://www.plumbingandhvac.com/
7. Industrial Safety and Workshop Guidelines:
   https://www.osha.gov/workplace-safety

COURSE CONTENT

ORDINARY DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS								
II Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA201BS	**Basic Sciences**	3	-	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Mathematical knowledge at pre-university level								

I. COURSE OVERVIEW:

This course provides a comprehensive introduction to Matrices & Caluculus, covering fundamental concepts, techniques and applications in Mathematics, Physics, Engineering and other fields.

II. COURSE OBJECTIVES:

The students will try to learn:

- **I** Methods of solving the differential equations of first and higher order.
- **II** Concept, properties of Laplace transforms.
- **III** Solving ordinary differential equations using Laplace transforms techniques.
- **IV** The physical quantities involved in engineering field related to vector valued functions
- **V** The basic properties of vector valued functions and their applications to line, surface and volume integrals

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Identify whether the given differential equation of first order is exact or not
CO2:	Solve higher differential equation and apply the concept of differential equation to real world problems.
CO3:	Use the Laplace Transforms techniques for solving Ordinary Differential Equations.
CO4:	Evaluate the Line, Surface and Volume integrals and converting them from one to another
CO5:	Evaluate line, surface, and volume integrals and apply vector calculus theorems such as Green's, Gauss's, and Stokes' theorems.

IV. COURSE CONTENT:

UNIT- I: (10 Hrs)

Matrices

Rank of a matrix by Echelon form and Normal form — Inverse of Non-singular matrices by Gauss- Jordan method. System of linear equations: Solving system of Homogeneous and Non-Homogeneous equations. Gauss Seidel Iteration Method.

UNIT – II: (10 Hrs)

First Order Ordinary Differential Equations

Exact differential equations – Equations reducible to exact differential equations – linear and Bernoulli's equations – Orthogonal Trajectories (only in Cartesian Coordinates). Applications: Newton's law of cooling – Law of natural growth and decay.

UNIT - III: (10 Hrs)

Ordinary Differential Equations of Higher Order

Higher-order linear differential equations with constant coefficients. Non-homogeneous terms of type e^{ax}, $\sin ax$, $\cos ax$,polynomials in x, $e^{ax}V(x)$, $xV(x)$. Method of variation of parameters.

UNIT - IV: (10 Hrs)

Laplace Transforms

Laplace Transforms: Laplace Transform of standard functions — First shifting theorem — Laplace transforms of functions multiplied by 't' and divided by 't' — Laplace transforms of derivatives and integrals of function — Evaluation of integrals by Laplace transforms — Laplace transform of periodic functions — Inverse Laplace transform by different methods, convolution theorem (without proof). Applications: solving Initial value problems by Laplace Transform method

UNIT - V: (10 Hrs)

Vector Integration

Line, Surface and Volume Integrals. Theorems of Green, Gauss and Stokes (without proofs) and their applications.

V. TEXT BOOKS:

1. B.S. Grewal, *Higher Engineering Mathematics*, Khanna Publishers, 36th Edition, 2010.
2. R.K. Jain and S.R.K. Iyengar, *Advanced Engineering Mathematics*, Narosa Publications, 5th Edition, 2016.

VI. REFERENCE BOOKS:

1. Erwin Kreyszig, *Advanced Engineering Mathematics*, 9th Edition, John Wiley & Sons, 2006.
2. G.B. Thomas and R.L. Finney, *Calculus and Analytic Geometry*, 9th Edition, Pearson, Reprint 2002.
3. N.P. Bali and Manish Goyal, *A Textbook of Engineering Mathematics*, Laxmi Publications, Reprint 2008.
4. H.K. Dass and Er. Rajnish Verma, *Higher Engineering Mathematics*, S Chand and Company Limited. New Delhi.

VII. ELECTRONICS RESOURCES:

1. NPTEL Online Courses – Differential Equations and Vector Calculus
   https://nptel.ac.in/courses/111/106/111106121/
2. MIT OpenCourseWare – Differential Equations
   https://ocw.mit.edu/courses/mathematics/18-03sc-differential-equations-fall-2011/
3. Khan Academy – Differential Equations
   https://www.khanacademy.org/math/differential-equations
4. Paul's Online Math Notes – Ordinary Differential Equations
   http://tutorial.math.lamar.edu/Classes/DE/DE.aspx
5. Vector Calculus Tutorials – Math Insight
   https://mathinsight.org/vector_calculus
6. YouTube Lecture Series – Engineering Mathematics
   https://www.youtube.com/results?search_query=engineering+mathematics+vector+calculus

COURSE CONTENT

ADVANCED ENGINEERING PHYSICS								
II Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25PH202BS	**Basic Sciences**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: 10+2 Physics								

I. COURSE OVERVIEW:

Engineering Physics is a multidisciplinary course blending physics principles with engineering applications to solve complex technological problems, covering core areas like Quantum Mechanics, Optics, Materials Science, Electromagnetism, and Semiconductors, preparing students for roles in research, innovation, and advanced tech fields like Nanotechnology and Photonics through a mix of theory, lab work, and practical projects.

II. COURSE OBJECTIVES:

The students will try to learn:

I. To study crystal structures, defects, and material characterization techniques like XRD and SEM.
II. To understand fundamental concepts of quantum mechanics and their applications in solids and nanomaterials.
III. To introduce quantum computing principles, quantum gates, and basic quantum algorithms.
IV. To learn the properties and applications of magnetic and dielectric materials.
V. To explore the working and applications of lasers and fibre optics in modern technology.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Analyze crystal structures, identify defects, and apply XRD and SEM techniques for material characterization.
CO2:	Apply quantum mechanical principles to explain particle behaviour and energy band formation in solids.
CO3:	Understand quantum computing concepts, use quantum gates, and explain basic quantum algorithms.
CO4:	Classify magnetic and dielectric materials and explain their properties, synthesis, and applications.
CO5:	Apply the principles of lasers and fibre optics and their applications in communication and sensing.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2	3		1	2	-	-	-	-	-	1		
CO2	2				2	2	-	-	2	3			
CO3	2	2	-		3	-	-	-	-	2	1		
CO4	1	2	-		1	-	1	-	2	3			
CO5	2		2				1	-	-	2	3		

V. COURSE CONTENT:

UNIT- I: (12 Hrs)

Crystallography & Materials Characterization

Introduction: Unit cell, space lattice, basis, lattice parameters; crystal structures, Bravais lattices, packing factor: SC, BCC, FCC; Miller indices, inter-planar distance; defects in crystals (Qualitative): point defects, line defects, surface defects and volume defects. Concept of nanomaterials: surface to volume ratio, X -ray diffraction: Bragg's law, powder method, calculation of average crystallite size using Debye Scherrer's formula, scanning electron microscopy (SEM): block diagram, working principle.

UNIT – II: (12 Hrs)

Quantum Mechanics

Introduction, de-Broglie hypothesis, Heisenberg uncertainty principle, physical significance of wave function, postulates of quantum mechanics: operators in quantum mechanics, eigen values and eigen functions, expectation value; Schrödinger's time independent wave equation, particle in a 1D box, Bloch's theorem (qualitative), Kronig-Penney model (qualitative): E-k diagram, effective mass of electron, formation of energy bands, origin of bandgap, classification of solids, concept of discrete energy levels and quantum confinement in nanomaterials.

UNIT - III: (12 Hrs)

Quantum Computing

Introduction, linear algebra for quantum computation, Dirac's Bra and Ket notation and their properties, Hilbert space, Bloch's sphere, concept of quantum computer, classical bits, Qubits, multiple Qubit system, quantum computing system for information processing, evolution of quantum systems, quantum measurements, entanglement, quantum gates, challenges and advantages of quantum computing over classical computation, quantum algorithms: Deutsch-Jozsa, Shor, Grover.

UNIT - IV: (12 Hrs)

Magnetic and Dielectric Materials

Introduction to magnetic materials, origin of magnetic moment-classification of magnetic materials, hysteresis, Weiss domain theory of ferromagnetism, soft and hard magnetic materials, synthesis of ferrimagnetic materials using sol-gel method, applications: magnetic hyperthermia for cancer treatment, magnets for EV, Giant Magneto Resistance (GMR) device.

Introduction to dielectric materials, types of polarization (qualitative): electronics, ionic & orientation; ferroelectric, piezoelectric, pyroelectric materials and their applications: Ferroelectric Random-Access Memory (Fe-RAM), load cell and fire sensor.

UNIT - V: (12 Hrs)

Laser and Fibre Optics

Introduction to laser, characteristics of laser, Einstein coefficients and their relations, metastable state, population inversion, pumping, lasing action, Ruby laser, He-Ne laser, CO_2 laser, semiconductor diode laser, applications: Bar code scanner, LIDAR for autonomous vehicle.

Introduction to fibre optics, total internal reflection, construction of optical fibre, acceptance angle, numerical aperture, classification of optical fibres, losses in optical fibre, applications: optical fibre for communication system, sensor for structural health monitoring.

VI. TEXT BOOKS:

1. Walter Borchardt-Ott, Crystallography: An Introduction, Springer.
2. Charles Kittel, Introduction to Solid State Physics, John Wiley & Sons, Inc.
3. Thomas G. Wong, Introduction to Classical and Quantum Computing, Rooted Grove

VII. REFERENCE BOOKS:

1. Jozef Gruska, *Quantum Computing*, McGraw Hill
2. Michael A. Nielsen & Isaac L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press.
3. John M. Senior, *Optical Fiber Communications Principles and Practice*, Pearson Education Limited.

VIII. ELECTRONICS RESOURCES:

1. https://shijuinpallotti.wordpress.com/wp-content/uploads/2019/07/optical-fiber-communications-principles-and-pr.pdf
2. https://www.geokniga.org/bookfiles/geokniga-crystallography_0.pdf
3. https://dpbck.ac.in/wp-content/uploads/2022/10/Introduction-to-Solid-State-PhysicsCharles-Kittel.pdf
4. https://www.thomaswong.net/introduction-to-classical-and-quantum-computing-1e4p.pdf
5. https://www.fi.muni.cz/usr/gruska/qbook1.pdf
6. https://profmcruz.wordpress.com/wp-content/uploads/2017/08/quantum-computation-and-quantum-information-nielsen-chuang.pdf

COURSE CONTENT

ENGINEERING DRAWING AND COMPUTER AIDED DRAFTING							
II Semester: CSE					**Regulations: R25**		
CourseCode	**Category**	**Hours/Week**			**Credits**	**MaximumMarks**	
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25ME203ES	**Foundation**	2	0	2	3	40	60	100
Contact Classes:60	**Tutorial Classes:0**	**Practical Classes:60**				**Total Classes:60**		
Prerequisite: Imagination Skills Required							

I. COURSE OVERVIEW:

An Engineering drawing and Computer Aided Drafting (ED&CAD) course overview covers the fundamentals of graphic representation of designs, including manual sketching techniques and their transition to using CAD software for creating 2D and 3D models. Key Topics includes Introduction to Engineering Graphics, Orthographic Projections (Projections of points, Straight lines, Planes, Auxiliary planes, Projections of Regular Solids and sectional Views, Development of Surfaces, Isometric Projections of planes, solids, Conversions.

II. COURSE OBJECTIVES:

The students will try to learn:.

1. To introduce the fundamentals of engineering drawing and projection systems.
2. To develop skills in constructing orthographic, isometric, and sectional views.
3. To train students in interpreting and creating technical drawings using CAD tools.
4. To familiarize students with dimensioning standards and drafting conventions.
5. To bridge manual drafting techniques with computer-aided drafting practices.

III.COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Apply the principles of Engineering Drawing to create drawings of various geometric constructions, conic sections, curves and scales as per BIS standards by conventional mode.
CO2:	Construct orthographic projections for points, lines and planes in different quadrants and Auxiliary views of planes both in Conventional and Computer Aided Drafting
CO3:	Draw the sectional views and true shape of sections of solids, by applying principles of projections both in conventional and Computer Aided Drafting
CO4:	Draw the development of surfaces of solids in real time situations by conventional mode.
CO5:	Develop isometric and orthographic views of the objects both in conventional and Computer Aided Drafting.

IV. COURSE MARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	2	2	1	2				1	1	
CO2	2	2	2	1	2				1	1	
CO3	2	2	2	1	2				1	1	
CO4	2	2	2	1	2				1	1	
CO5	2	2	2	1	2				1	1	

V. COURSE CONTENT:

UNIT- I:

Introduction to Engineering Graphics (Conventional)
Principles of Engineering Graphics and their Significance, Geometrical Constructions,Scales,Plain and Diagonal, Conic Sections including the Rectangular Hyperbola, General method only. Cycloid, Epicycloid and Hypocycloid

UNIT – II:

Orthographic Projections (Conventional and Computer Aided)
Principles of Ortho graphic Projections, Conventions, Projections of Points and Lines, Projections of Plane regular geometric figures. Auxiliary Planes. Computer aided orthographic projections, points, lines and planes. Introduction to Computer aided drafting, views, commands and conics.

UNIT - III:

Projections of Regular Solids (Conventional and Computer Aided) (14Hrs)
Auxiliary Views, Sections or Sectional views of Right Regular Solids, Prism, Cylinder, Pyramid, Cone, Auxiliary views, Computer aided projections of solids, sectional views

UNIT - IV:

Development of Surfaces (Conventional):
Prism, Cylinder, Pyramid and Cone.

UNIT – V

Isometric Projections (Conventional and Computer Aided) (10Hrs)
Principles of Isometric Projection, Isometric Scale, Isometric Views, Conventions, Isometric Views of Lines, Plane Figures, Simple and Compound Solids, Isometric Projection of objects having non, isometric lines. Isometric Projection of Spherical Parts. Conversion of Isometric Views to Orthographic Views and Vice- versa, Conventions. Conversion of orthographic projection into isometric view.

VI. TEXT BOOKS:

1. Engineering Drawing, N.D.Bhatt, Charotar, 54th Edition, 2023.
2. Engineering Drawing and graphics Using AutoCAD, T. Jeyapoovan and Vikas, S. Chand and company Ltd., 3rdEdition, 2010.

VII. REFERENCE BOOKS:

1.Engineering Drawing, Basant Agrawal and C.M.Agrawal, Mc Graw Hill, 3rdEdition, 2019.
2. Engineering Graphics and Design, WILEY, John Wiley and Sons Inc, 3rd Edition, 2020.
3. Engineering Drawing, M.B.Shah and B.C.Rane, Pearson, 2ndEdition, 2009.
4. Engineering Drawing, N.S.Parthasarathy and Vela Murali, Oxford,1st Edition,2015.
5.Computer Aided Engineering Drawing, K.Balaveera Reddy, CBS Publishers, 2ndEdition, 2015.

VIII. ELECTRONIC RESOURCES:

1. https://soaneemrana.com/onewebmedia/ENGINEERING%20DRAWING%20BY%20N.D%20BHATT.pdf
2. https://ia800105.us.archive.org/0/items/TextbookOfEngineeringDrawing_201802/Textbook%20of%20Engineering%20Drawing.pdf

COURSE CONTENT

BASIC ELECTRICAL ENGINEERING								
II Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EE204ES	**Foundation**	3	0	0	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Mathematics								

I. COURSE OVERVIEW:

This course Introduction to Electrical Engineer is a foundation course of electrical for all engineering branches. The concepts of DC & AC circuits, Power & Power factor, R-L-C elements, machine, transformer and electrical installations are very useful in real life. This course is applicable for industries.

II. COURSE OBJECTIVES:

The students will try to learn:

I To understand DC and Single & Three phase AC circuits
II To study and understand the different types of DC, AC machines and Transformers
III To import the knowledge of various electrical installations and the concept of power, power factor and its improvement

III. COURSE OUTCOMES:

At the end of the course, students should be able to:	
CO1:	Understand and analyze basic Electrical circuits
CO2:	Understand and analyze basic Electrical circuits Laws and Theorems
CO3:	Study the working principles of Transformers
CO4:	Study the working principles of Electrical Machines and Transformers
CO5:	Introduce components of Low Voltage Electrical Installations

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	3			2	2	1						1	1
CO2	2	3	2		2	2				1		2	
CO3	1		3	1		3					2	3	1
CO4	1	2	3		1					1		2	
CO5	1	1	3	2	2	1						1	

V. COURSE CONTENT:

UNIT- I: (5 Hrs)

D.C. Circuits: Electrical circuit elements (R, L and C), voltage and current sources, KVL&KCL, analysis of simple circuits with dc excitation. Superposition, Thevenin and Norton Theorems. Time-domain analysis of first-order RL and RC circuits. .

UNIT – II: (7 Hrs)

A.C. Circuits: Representation of sinusoidal waveforms, peak and rms values, phasor representation, real power, reactive power, apparent power, power factor, Analysis of single-phase ac circuits consisting of R, L, C, RL, RC, RLC combinations (series and parallel), resonance in series R-L-C circuit. Three-phase balanced circuits, voltage and current relations in star and delta connections**.**

UNIT - III: (6 Hrs)

Transformers: Ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency. Auto-transformer and three-phase transformer connections.

UNIT - IV: (7 Hrs)

Electrical Machines: Construction and working principle of dc machine, performance characteristics of dc shunt machine. Generation of rotating magnetic field, Construction and working of a three-phase induction motor, Significance of torque-slip characteristics. Single-phase induction motor, Construction and working. Construction and working of synchronous generator.

UNIT - V: (5 Hrs)

Electrical Installations: Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, ELCB, MCCB, Types of Wires and Cables, Earthing. Types of Batteries, Important Characteristics for Batteries. Elementary calculations for energy consumption, power factor improvement and battery backup**.**

VI.TEXT BOOKS:

1. D.P. Kothari and I. J. Nagrath, “Basic Electrical Engineering”, Tata McGraw Hill, 4th Edition, 2019.
2. MS Naidu and S Kamakshaiah, “Basic Electrical Engineering”, Tata McGraw Hill, 2nd Edition, 2008.

VII.REFERENCE BOOKS:

1. P. Ramana, M. Suryakalavathi, G.T. Chandrasheker, “Basic Electrical Engineering”, S. Chand, 2nd Edition, 2019. 2. D. C. Kulshreshtha, “Basic Electrical Engineering”, McGraw Hill, 2009
3. M. S. Sukhija, T. K. Nagsarkar, “Basic Electrical and Electronics Engineering”, Oxford, 1st Edition, 2012.
4. Abhijit Chakrabarthi, Sudipta Debnath, Chandan Kumar Chanda, “Basic Electrical Engineering”, 2nd Edition, McGraw Hill, 2021.
5. L. S. Bobrow, “Fundamentals of Electrical Engineering”, Oxford University Press, 2011.
6. E. Hughes, “Electrical and Electronics Technology”, Pearson, 2010. 7. V. D. Toro, “Electrical Engineering Fundamentals”, Prentice Hall India, 1989

VIII.ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/108/104/108104139/
2. https://onlinecourses.nptel.ac.in/noc25_ee91/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

DATA STRUCTURES								
II Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS205ES	Engineering and Sciences	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: A course on Programming for Problem Solving								

I. COURSE OVERVIEW:

This course provides a comprehensive introduction to data structures and algorithms used in computer programming. It emphasizes the selection of appropriate data structures, analysis of their efficiency, and implementation of algorithms for searching, sorting, hashing, and file organization.

I. COURSE OBJECTIVES:

The students will try to learn:

I. Exploring basic data structures such as stacks and queues.
II. Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
III. Introduces sorting and pattern matching algorithms.

II. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand and apply fundamental concepts of data structures and abstract data types to implement linear data structures such as linked lists, stacks, and queues.
CO2:	Analyze and implement hashing techniques using appropriate hash functions and collision resolution methods for efficient data storage and retrieval.
CO3:	Design and implement tree-based data structures including binary trees, heaps, and binary search trees, and perform traversal, insertion, deletion, and searching operations.
CO4:	Evaluate advanced tree structures such as AVL trees, Red-Black trees, Splay trees, threaded binary trees, and B-trees to achieve balanced and efficient searching.
CO5:	Apply graph representations, traversal algorithms, and sorting techniques to solve problems related to networks, relationships, and data organization.

III. COURSE CONTENT:

UNIT- I:

Introduction to Data Structures: Basic Terminology, Classification of Data Structures, Operation on Data Structures, abstract data types, selecting a Data Structure, Linear list — Introduction, singly linked list, Circular Linked Lists, Doubly Linked List, Stacks- Operations, Stack algorithm, Stack ADT, Stack applications, Queues- operations, Queue Algorithm, Queue ADT, Queue Applications.

UNIT – II:

Hashing and Collision Introduction, Hash Tables, Hash Functions, Different Hash Functions: Division Method, Multiplication Method, Mid-square Method, Folding Method, collisions: Collision Resolution by Open Addressing, Collision Resolution by Chaining.

UNIT - III:

Trees Introduction, Types of Trees, creating a Binary Tree from a General Tree, traversing a Binary Tree, Heaps, Comparison of various Heaps, Applications Searching, Binary Search Trees (BST), BST Operations Searching, Insertion and Deletion, BST ADT, BST Applications

UNIT - IV:

Threaded Binary Trees, AVL Trees, Red Black Trees, Splay Trees Multi way Search Trees, Introduction, B Trees

UNIT – V

Graphs: Introduction, Directed Graphs, Bi connected Components, Representation of Graphs. Graph Traversal Algorithms, Graph ADT, Applications of Graphs, **Sorting:** Radix Sort, Heap sort, Shell Sort

IV. TEXT BOOKS:

1. Data Structures: A Pseudocode Approach with C, 2 nd Edition, R. F. Gilberg and B.A.Forouzan, Cengage Learning
2. Data Structure using C– Reema Thareja, 3rd Edition, Oxford University Press.

V. REFERENCE BOOKS:

1. Data Structures using C – A. S.Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson Education.

VI. ELECTRONICS RESOURCES:

1. NPTEL – Data Structures and Algorithms (Video Lectures & Course) https://nptel.ac.in/courses/106/102/064/
2. GeeksforGeeks – Data Structures & Algorithms Tutorials and Practice https://www.geeksforgeeks.org/data-structures-and-algorithms-online-courses-free-and-paid/
3. Coursera – Data Structures Course (University of California San Diego) https://www.coursera.org/learn/data-structures
4. W3Schools (Data Structures Tutorial Index) https://www.w3schools.in/data-structures/tutorials/
5. Google Tech Dev Guide – Data Structures Resources & Practice https://techdevguide.withgoogle.com/resources/topics/data-structures/
6. GitHub – DSA Resources (Community Compiled Lists) https://github.com/GDSC-KIIT/DSA-Resources

COURSE CONTENT

ADVANCED ENGINEERING PHYSICS LAB								
II Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25PH206BS	**Basic Sciences**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nill**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic lab handling skills in introductory courses								

I. COURSE OVERVIEW:

An Engineering Physics Lab course provides hands-on experience to bridge theory and practice, focusing on experiments in optics, electronics, mechanics, and modern physics (lasers, fibers) to develop practical skills, precise measurement techniques, data analysis, and technical report writing, enabling students to apply physics principles to real-world engineering problems and design solutions.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To provide practical exposure to advanced concepts in solid-state and modern physics.
2. To synthesize and study the physical properties of materials like semiconductors, ferromagnetic, and ferroelectric substances.
3. To perform semiconductor characterization using Hall effect and band gap experiments.
4. To explore the working principles of lasers and optical fibers through hands-on experiments.
5. To develop skills in data analysis, interpretation, and scientific reporting.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Synthesize and analyze nanomaterials such as magnetite (Fe_3O_4) using chemical methods.
CO2:	Determine key electrical, magnetic, and optical properties of semiconductors and other functional materials
CO3:	Characterize semiconductors using Hall effect and energy gap measurement techniques
CO4:	Demonstrate working knowledge of laser systems and optical fiber parameters through experimental study
CO5:	Apply scientific methods for accurate data collection, analysis, and technical report writing

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11
CO1	1			3		2			1		2
CO2			3		2	1			1		2
CO3	2		2		2	3				2	
CO4	2				2	3			1	2	
CO5		2	3		2				1		1

COURSE CONTENT:

List of Experiments:

1. Synthesis of magnetite (Fe_3O_4) powder using sol-gel method.
2. Determination of energy gap of a semiconductor.
3. Determination of Hall coefficient and carrier concertation of a given semiconductor.
4. Determination of magnetic field along the axis of a current carrying coil.
5. Study of B-H curve of a ferro magnetic material.
6. Determination of work function and Planck's constant using photoelectric effect.
7. Determination of dielectric constant of a given material.
8. Determination of Curie's temperature of a given ferroelectric material.
9. Determination of wavelength of a laser using diffraction grating.
10. Study of V-I & L-I characteristics of a given laser diode.
11. Determination of numerical aperture of a given optical fibre.
    Determination of bending losses of a given optical fibre.

Note: Any 8 experiments are to be performed.

I. REFERENCE BOOKS:

1. Practical physics by Vijay Kumar and T Radha Krishna (JNTUH).

COURSE CONTENT

DATA STRUCTURES LAB								
II Semester: C S E					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS207ES	**Engineering and Science**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: A course on Programming for Problem Solving								

I. COURSE OVERVIEW:

The **Data Structures Lab** provides hands-on experience in implementing fundamental and advanced data structures using the C programming language. It reinforces theoretical concepts learned in the *Data Structures* theory course by enabling students to:

- Develop programs using arrays, pointers, functions, and control structures.
- Implement linear and non-linear data structures such as stacks, queues, linked lists, trees, heaps, graphs, and hash tables.
- Apply sorting and searching algorithms efficiently to solve real-world computational problems.
- Gain practical skills in modular program design, debugging, and algorithm optimization.

This lab is designed to strengthen logical thinking, programming proficiency, and problem-solving skills, bridging the gap between theory and practical applications of data structures in computer science.

II. COURSE OBJECTIVES:

The students will try to learn:

I It covers various concepts of C programming language
II It introduces searching and sorting algorithms
III It provides an understanding of data structures such as stacks and queues.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Develop C programs using arrays, functions, pointers, and control statements for solving computational problems.
CO2:	Implement and manipulate linear data structures such as stacks, queues, and linked lists efficiently.
CO3:	Construct and traverse non-linear data structures including binary trees, AVL trees, B-trees, and graphs.
CO4:	Apply sorting and searching algorithms such as radix sort, heap sort, shell sort, interpolation search, and binary search in practical scenarios.
CO5:	Implement and analyze hash functions and collision resolution techniques to store and retrieve data effectively.

IV. COURSE CONTENT:

PRACTICE SESSIONS:

List of Experiments

1. Write a program that uses functions to perform the following operations on singly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program that uses functions to perform the following operations on doubly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Write a program that uses functions to perform the following operations on circular linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that implement stack (its operations) using
   i) Arrays ii) ADT
5. Write a program that implement Queue (its operations) using
   i) Arrays ii) ADT
6. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
   i) Radix Sort, ii) Heap sort, iii) Shell Sort
7. Write a program to implement the tree traversal methods (Recursive and Non-Recursive).
8. Write a program to implement
   i) Binary Search tree ii) AVL trees iii) Red - Black trees
9. Write a program to implement the graph traversal methods.
10. Write a program to implement the following Hash Functions: i) Division Method, ii) Multiplication Method, iii) Mid-square Method, iv) Folding Method.

V. TEXT BOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

VI. REFERENCE BOOKS:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage Learning.

VII. ELECTRONICS RESOURCES:

1. NPTEL – Data Structures and Algorithms (Video Lectures & Course) https://nptel.ac.in/courses/106/102/064/
2. GeeksforGeeks – Data Structures & Algorithms Tutorials and Practice https://www.geeksforgeeks.org/data-structures-and-algorithms-online-courses-free-and-paid/
3. Coursera – Data Structures Course (University of California San Diego) https://www.coursera.org/learn/data-structures
4. edX – Learn Data Structures & Algorithms Courses https://www.edx.org/learn/data-structures
5. W3Schools (Data Structures Tutorial Index) https://www.w3schools.in/data-structures/tutorials/
6. Google Tech Dev Guide – Data Structures Resources & Practice https://techdevguide.withgoogle.com/resources/topics/data-structures/
7. GitHub – DSA Resources (Community Compiled Lists) https://github.com/GDSC-KIIT/DSA-Resources

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

PYTHON PROGRAMMING LAB								
II Semester: C S E					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS208ES	**Engineering and Science**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: A course on Programming for Problem Solving								

I. COURSE OVERVIEW:

The **Python Programming Lab** aims to provide hands-on experience in programming using Python. The lab emphasizes practical implementation of programming constructs, control structures, data types, and advanced features like file handling, object-oriented programming, and GUI design. Students will develop problem-solving skills and gain experience in creating applications using Python libraries and modules, including digital logic simulations, data processing, and visualization.

II. COURSE OBJECTIVES:

The students will try to learn:

I To install and run the Python interpreter
II To learn control structures.
III To Understand Lists, Dictionaries in python
IV To Handle Strings and Files in Python

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Develop the application specific codes using python.
CO2:	Understand Strings, Lists, Tuples and Dictionaries in Python
CO3:	Verify programs using modular approach, file I/O, Python standard library
CO4:	Implement Digital Systems using Python
CO5:	Design GUI-based applications using Python libraries such as Tkinter.

IV. COURSE CONTENT:

PRACTICE SESSIONS:

List of Experiments:

1.

i) Use a web browser to go to the Python website http://python.org. This page contains information about Python and links to Python-related pages, and it gives you the ability to search the Python documentation.

ii) Start the Python interpreter and type help() to start the online help utility.

2. Start a Python interpreter and use it as a Calculator.
3. Write a program to calculate compound interest when principal, rate and number of periods are given.
4. Read the name, address, email and phone number of a person through the keyboard and print the details.
5. Print the below triangle using for loop. 5

   4 4
   3 3 3
   2 2 2 2
   1 1 1 1 1
6. Write a program to check whether the given input is digit or lowercase character or uppercase character or a special character(use 'if-else-if' ladder)
7. Python program to print all prime numbers in a given interval (use break)
8. Write a program to convert a list and tuple into arrays.
9. Write a program to find common values between two arrays.
10. Write a function called palindrome that takes a string argument and returnsTrue if it is a palindrome and False otherwise. Remember that you can use the built-in function len to check the length of a string.
11. Write a function called is_sorted that takes a list as a parameter and returns True if the list is sorted in ascending order and False otherwise.
12. Write a function called has_duplicates that takes a list and returns True if there is any element that appears more than once. It should not modify the original list.
13. Write a function called remove_duplicates that takes a list and returns a new list with only the unique elements from the original. Hint: they don't have to be in the same order.
14. The wordlist I provided, words.txt, doesn't contain single letter words. So you might want to add "I", "a", and the empty string.
15. Write a python code to read dictionary values from the user. Construct a function to invert its content. i.e., keys should be values and values should be keys.
16. Add a comma between the characters. If the given word is 'Apple', it should become 'A,p,p,l,e'
17. Remove the given word in all the places in a string?
18. Write a function that takes a sentence as an input parameter and replaces the first letter of every word with the corresponding upper case letter and the rest of the letters in the word by corresponding letters in lower case without using a built-in function?
19. Writes a recursive function that generates all binary strings of n-bit length
20. Write a python program that defines a matrix and prints
21. Write a python program to perform multiplication of two square matrices
22. How do you make a module? Give an example of construction of a module using different geometrical shapes and operations on them as its functions.
23. Use the structure of exception handling all general-purpose exceptions.
24. Write a function called draw_rectangle that takes a Canvas and a Rectangle as arguments and draws a representation of the Rectangle on the Canvas.
25. Add an attribute named color to your Rectangle objects and modify draw_rectangle so that it uses the color attribute as the fill color.
26. Write a function called draw_point that takes a Canvas and a Point as arguments and draws a representation of the Point on the Canvas.
27. Define a new class called Circle with appropriate attributes and instantiate a few Circle objects. Write a function called draw_circle that draws circles on the canvas.
28. Write a python code to read a phone number and email-id from the user and validate it for correctness.
29. Write a Python code to merge two given file contents into a third file.
30. Write a Python code to open a given file and construct a function to check for given words present in it and display on found.
31. Write a Python code to Read text from a text file, find the word with most number of occurrences
32. Write a function that reads a file *file1*and displays the number of words, number of vowels, blank spaces, lower case letters and uppercase letters.
33. Import numpy, Plotpy and Scipy and explore their functionalities.
34. Install NumPypackage with pip and explore it.
35. Write a program to implement Digital Logic Gates – AND, OR, NOT, EX-OR
36. Write a GUI program to create a window wizard having two text labels, two text fields and two buttons as

Submit and Reset.

V. TEXT BOOKS:

1. Supercharged Python: Take your code to the next level, Overland
2. Learning Python, Mark Lutz, O'reilly

VI. REFERENCE BOOKS:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Python Programming A Modular Approach with Graphics, Database, Mobile, and Web Applications, Sheetal Taneja, Naveen Kumar, Pearson
3. Introduction to Python Programming, Gowrishakar S, Veena A, CRC Press
4. Programming with Python, A User's Book, Michael Dawson, Cengage Learning, India Edition
5. Python for Data Science, Dr. Mohd Abdul Hameed, Wiley publications
6. Core Python Programming, Dr. R. Nageswara Rao, Dreamtech press
7. Introduction to Python, Gowrishankar S, Veena A., CRC Press

VII. ELECTRONICS RESOURCES:

1. Python Official Website: https://www.python.org
2. NPTEL Python Programming: https://nptel.ac.in/courses/106/106/106106144/
3. GeeksforGeeks Python Tutorials: https://www.geeksforgeeks.org/python-programming-language/
4. W3Schools Python Tutorial: https://www.w3schools.com/python/
5. TutorialsPoint Python Tutorial: https://www.tutorialspoint.com/python/index.htm

COURSECONTENT

BASIC ELECTRICAL ENGINEERING LAB								
II Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EE209ES	**Foundation**	0	0	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic Electrical Engineering								

I. COURSEOVERVIEW:

Focuses on hands-on verification of circuit laws and theorems (Ohm's, Kirchhoff's), resonance, series/parallel RLC circuits, Thevenin's/Norton's theorems,practical knowledge on the fundamentals of DC and AC circuits.

II. COURSEOBJECTIVES:

Thestudentswilltrytolearn:
To measure the electrical parameters for different types of DC and AC circuits using conventional and theorems approach.
To study the transient response of various R, L and C circuits using different excitations.
To determine the performance of different types of DC, AC machines and Transformers.

III. COURSEOUTCOMES:

Attheendofthecoursestudentsshouldbeableto:	
CO1:	Verify the basic Electrical circuits through different experiments.
CO2:	Evaluate the performance calculations of Electrical Machines and Transformers through various testing methods.
CO3:	Analyze the transient responses of R, L and C circuits for different input conditions.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	3	2	1	0	1	0	0	2	0	2	2	3	3
CO2	3	2	1	0	3	1	0	2	1	2	-	-	-
CO3	3	2	1	1	3	2	0	0	1	-	-	-	3

V. COURSECONTENT:

List of experiments/demonstrations:

PART- A (compulsory)

1. Verification of KVL and KCL
2. Verification of Thevenin's and Norton's theorem
3. Transient Response of Series RL and RC circuits for DC excitation
4. Resonance in series RLC circuit
5. Calculations and Verification of Impedance and Current of RL, RC and RLC series circuits
6. Measurement of Voltage, Current and Real Power in primary and Secondary Circuits of a Single-Phase Transformer
7. Performance Characteristics of a DC Shunt Motor
8. Torque-Speed Characteristics of a Three-phase Induction Motor.

PART-B (any two experiments from the given list)

1. Verification of Superposition theorem.
2. Three Phase Transformer: Verification of Relationship between Voltages and Currents (Star-Delta, Delta-Delta, Delta-star, Star-Star)
3. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
4. Measurement of Active and Reactive Power in a balanced Three-phase circuit
5. No-Load Characteristics of a Three-phase Alternator

VI TEXTBOOKS:

1. D.P. Kothari and I. J. Nagrath, "Basic Electrical Engineering", Tata McGraw Hill, 4th Edition, 2019.
2. MS Naidu and S Kamakshaiah, "Basic Electrical Engineering", Tata McGraw Hill, 2nd Edition, 2008.

VII REFERENCEBOOKS:

1. P. Ramana, M. Suryakalavathi, G.T.Chandrasheker,"Basic Electrical Engineering", S. Chand, 2nd Edition, 2019.
2. D. C. Kulshreshtha, "Basic Electrical Engineering", McGraw Hill, 2009
3. M. S. Sukhija, T. K. Nagsarkar, "Basic Electrical and Electronics Engineering", Oxford, 1st Edition, 2012.
4. AbhijitChakrabarthi, SudiptaDebnath, Chandan Kumar Chanda, "Basic Electrical Engineering", 2nd Edition, McGraw Hill, 2021.
5. L. S. Bobrow, "Fundamentals of Electrical Engineering", Oxford University Press, 2011
6. E. Hughes, "Electrical and Electronics Technology", Pearson, 2010
7. V. D. Toro, "Electrical Engineering Fundamentals", Prentice Hall India, 1989.

VIII.ELECTRONICSRESOURCES:

1. https://nptel.ac.in/courses/108/104/108104139/
2. https://nptel.ac.in/courses/108/106/108106172/
3. https://ocw.mit.edu/search/ocwsearch.htm?q=laboratory

COURSE CONTENT

IT WORKSHOP								
II Semester: C S E					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS210ES	**Engineering and Science**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: A course on Programming for Problem Solving								

I. COURSE OVERVIEW:

The **IT Workshop** course is a hands-on laboratory-oriented program designed to provide first-year engineering students with practical exposure to essential computer hardware, operating systems, networking, and productivity tools. The course focuses on developing foundational IT skills required for effective use, maintenance, and troubleshooting of computer systems.

II. COURSE OBJECTIVES:

The students will try to learn:

The IT Workshop for engineers is a training lab course spread over 60 hours. The modules include training on PC Hardware, Internet & World Wide Web and Productivity tools including Word, Excel, PowerPoint and Publisher.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Perform Hardware troubleshooting
CO2:	Understand Hardware components and inter dependencies
CO3:	Safeguard computer systems from viruses/worms
CO4:	Document/ Presentation preparation
CO5:	Perform calculations using spreadsheets

IV. COURSE CONTENT:

PRACTICE SESSIONS:

PC Hardware

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Task 4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Internet & World Wide Web

Task1: **Orientation & Connectivity Boot Camp:** Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Task 3: **Search Engines & Netiquette:** Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

LaTeX and WORD

Task 1 – Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office or equivalent (FOSS) tool word: Importance of LaTeX and MS office or equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using LaTeX and word — Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

Task 2: Using LaTeX and Word to create a project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Task 3: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 4: Creating a Newsletter: Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

Excel

Excel Orientation: The mentor needs to tell the importance of MS office or equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the four tasks and features that would be covered in each. Using Excel — Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 1: Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text

Task 2: Calculating GPA - Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP

Task 3: Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting

PowerPoint

Task 1: Students will be working on basic power point utilities and tools which help them create basic PowerPoint presentations. PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.

Task 2: Interactive presentations - Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides.

V. REFERENCE BOOKS:

1. Comdex Information Technology course tool kit Vikas Gupta, *WILEY Dreamtech*
2. The Complete Computer upgrade and repair book, 3rd edition Cheryl A Schmidt, *WILEY Dreamtech*
3. Introduction to Information Technology, ITL Education Solutions limited, *Pearson Education.*
4. PC Hardware - A Handbook – Kate J. Chase *PHI* (Microsoft)
5. LaTeX Companion – Leslie Lamport, *PHI/Pearson.*
6. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. — *CISCO Press, Pearson Education.*
7. IT Essentials PC Hardware and Software Labs and Study Guide Third Edition by Patrick Regan — CISCO Press, *Pearson Education.*

VI. ELECTRONICS RESOURCES:

1. PC Hardware & Troubleshooting

- How Computers Work – Tutorials
- PC Hardware Repair & Upgrade Guide
- YouTube – PC Assembly & Disassembly

---

2. Operating Systems

- **Windows**
  - Microsoft Windows Installation Guide
  - Dual Boot Windows & Linux Tutorial
- **Linux**
  - Ubuntu Official Documentation
  - Linux Commands Cheat Sheet
  - Linux Dual Boot Guide

---

3. Networking & Internet

- Cisco Networking Academy
- IP Configuration & TCP/IP Settings Tutorial
- Cyber Hygiene & Internet Safety
- Search Engine Tips & Netiquette

---

4. Productivity Tools

- **Microsoft Word**
  - Word Tutorials – Microsoft
  - LaTeX Official Documentation
  - Overleaf Online LaTeX Editor
- **Microsoft Excel**
  - Excel Formulas & Functions
  - Excel Tutorial – W3Schools
- **Microsoft PowerPoint**
  - PowerPoint Training & Tutorials
  - Creating Effective Presentations

---

5. Free/Open Source Alternatives

- LibreOffice Suite (Word, Excel, PowerPoint alternative)
- GIMP – Free Image Editor
- Inkscape – Free Vector Graphics

COURSE CONTENT

DISCRETE MATHEMATICS								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS301PC	**Professional Core**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Mathematic- II and Programming for Problem Solving.								

I. COURSE OVERVIEW:

This course is a fundamental subject for computer science and engineering students. It provides the mathematical foundations required for reasoning about algorithms, data structures, computer logic, software development, and networking. This course focuses on logic, set theory, algebraic structures, combinatorics, and graph theory, which are essential for understanding problem-solving, formal proofs, and discrete structures in computing.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Introduces elementary discrete mathematics for computer science and engineering.
2. Topics include formal logic notation, methods of proof, induction, sets, relations, algebraic structures, elementary graph theory, permutations and combinations, counting principles; recurrence relations and generating functions.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand and construct precise mathematical proofs.
CO2:	Apply logic and set theory to formulate precise statements.
CO3:	Analyze and solve counting problems on finite and discrete structures.
CO4:	Describe and manipulate sequences.
CO5:	Apply graph theory in solving computing problems.

IV: COURSE CONTENT:

UNIT- I:

Mathematical logic: Introduction, Statements and Notation, Connectives, Normal Forms, Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus.

UNIT – II:

Set theory: Introduction, Basic Concepts of Set Theory, Representation of Discrete Structures, Relations and Ordering, Functions.

UNIT - III:

Algebraic Structures: Introduction, Algebraic Systems, Semi groups and Monoids, Lattices as Partially Ordered Sets, Boolean Algebra.

UNIT - IV:

Elementary Combinatorics: Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutation with Constrained Repetitions, Binomial Coefficient, The Binomial and Multinomial Theorems, The Principle of Exclusion.

UNIT - V:

Graph Theory: Basic Concepts, Isomorphism and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multi-graphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

V: TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science: J.P. Tremblay, R. Manohar, McGraw-Hill, 1st ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe l. Mott, Abraham Kandel, Teodore P. Baker, Prentis Hall of India, 2nd ed.

VI: REFERENCE BOOKS:

1. Discrete and Combinatorial Mathematics - an applied introduction: Ralph. P. Grimald, Pearson education, 5th edition.
2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill Publishing co.

ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/111/106/111106103/
2. https://www.geeksforgeeks.org/propositional-logic-and-predicate-logic/
3. https://www.tutorialspoint.com/discrete_mathematics/set_theory.htm

COURSE CONTENT

COMPUTER ORGANIZATION AND ARCHITECTURE								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS302PC	Professional Core	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: No prerequisites.								

I. COURSE OVERVIEW:

Computer Organization and Architecture (COA) provides a comprehensive understanding of the internal working of digital computers. The course introduces the structure and function of computer systems, focusing on the organization of processors, memory, input/output mechanisms, and control units. It covers instruction set design, arithmetic operations, pipelining, microprogramming, and memory hierarchy, enabling students to bridge the gap between hardware and software

I. COURSE OBJECTIVES:

The students will try to learn:

I. To introduce the principles of computer organization and basic architectural concepts
II. To understand the organization, design, and programming of a simple digital computer
III. To familiarize students with register transfer language and micro-operations
IV. To explain computer arithmetic, instruction set design, and control unit organization
V. To introduce memory organization, I/O systems, pipelining, and multiprocessors

II. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand the basics of instruction sets and their impact on processor design.
CO2:	Demonstrate an understanding of the design of the functional units of a digital computer system.
CO3:	Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
CO4:	Design a pipeline for consistent execution of instructions with minimum hazards.
CO5:	Recognize and manipulate representations of numbers stored in digital computers

III. COURSE CONTENT:

UNIT- I:

Boolean Algebra and Logic Gates: Binary codes, Binary Storage and Registers, Binary logic. **Digital logic gates. Data Representation**: Data types, Complements, Fixed Point Representation, Floating Point Representation

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

UNIT – II:

Combinational Logic: Combinational Circuits, Analysis procedure Design procedure, Binary Adder-Subtractor Decimal Adder, Binary multiplier, magnitude comparator, Decoders, Encoders, Multiplexers, HDL for combinational circuits.

Sequential Logic: Sequential circuits, latches, Flip-Flops Analysis of clocked sequential circuits, state Reduction and Assignment, Design Procedure**.** Registers, shift Registers, Ripple counters, synchronous counters, other counters.

UNIT - III:

.**Register Transfer Language and Micro operations**: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt

UNIT - IV:

Microprogrammed Control: Control memory, Address sequencing, micro program example, design of control unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

UNIT – V

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

IV. TEXT BOOKS:

1. . **M. Morris Mano**, *Digital Design*, Third Edition, Pearson / PHI
2. **M. Morris Mano**, *Computer System Architecture*, Third Edition, Pearson / PHI

V. REFERENCE BOOKS:

1. Switching and Finite Automata Theory, ZVI. Kohavi, Tata Mc Graw Hill.
2. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, 5th Edition, McGraw Hill.
3. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
4. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson.

VI. ELECTRONICS RESOURCES:

- NPTEL – Computer Organization and Architecture
- MIT OpenCourseWare – Computer Architecture
- IEEE Xplore Digital Library
- Pearson and McGraw Hill e-resources

COURSE CONTENT

OBJECT ORIENTED PROGRAMMING THROUGH JAVA								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS303PC	Professional Core	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic Programming Knowledge								

I. COURSE OVERVIEW:

Object Oriented Programming through Java introduces the principles of object-oriented programming using Java language. The course emphasizes concepts such as classes, objects, inheritance, polymorphism, exception handling, multithreading, GUI programming, and database connectivity. It enables students to develop robust, reusable, and platform-independent applications using Java.

I. COURSE OBJECTIVES:

The students will try to learn:

I. To Understand the basic object-oriented programming concepts and apply them in problem solving.
II. To Illustrate inheritance concepts for reusing the program.
III. To Demonstrate multitasking by using multiple threads and event handling
IV. To Develop data-centric applications using JDBC.
V. To Understand the basics of java console and GUI based programming

II. C

OURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Demonstrate the behavior of programs involving the basic programming constructs like control structures, constructors, string handling and garbage collection.
CO2:	Demonstrate the implementation of inheritance (multilevel, hierarchical and multiple) by using extend and implement keywords
CO3:	Use multithreading concepts to develop inter process communication.
CO4:	Understand the process of graphical user interface design and implementation using AWT or swings.
CO5:	Develop applets that interact abundantly with the client environment and deploy on the server.

III. COURSE CONTENT:

UNIT- I:

Object oriented thinking and Java Basics- Need for oop paradigm, summary of oop concepts, coping with complexity, abstraction mechanisms. History of Java, Java buzzwords, data types, variables, scope and lifetime of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, nested and inner classes, exploring String class.

UNIT – II:

Inheritance, Packages and Interfaces — Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super keyword uses, using final keyword with inheritance, polymorphism- method overriding, abstract classes, the Object class. Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces.

UNIT - III:

Exception handling and Multithreading-- Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception subclasses. Differences between multithreading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication, thread groups, daemon threads.

UNIT - IV:

Exploring String class, Object class, Exploring java.util package, Exploring java.io package.
Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes. graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

UNIT – V

Swing — Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JFrame and JComponent, JLabel, ImageIcon, JTextField, JButton, JCheckBox, JRadioButton, JList, JComboBox, Tabbed Panes, Scroll Panes, Trees, and Tables. Menu Basics, Menu related classes - JMenuBar, JMenu, JMenuItem, JCheckBoxMenuItem, JRadioButtonMenuItem, JSeperator. creating a popup menu.

IV. TEXT BOOKS:

1. Java the complete reference, 13th edition, Herbert schildt, Dr. Denny Coward, Mc Graw Hill.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson education.

V. REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java, J.Nino and F.A. Hosch, John Wiley & sons.
2. An Introduction to OOP, third edition, T. Budd, Pearson education.
3. Introduction to Java programming, Y. Daniel Liang, Pearson education.
4. An introduction to Java programming and object-oriented application development, R.A. Johnson- Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education.
6. Core Java 2, Vol 2, Advanced Features, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education
7. Object Oriented Programming with Java, R.Buyya, S.T.Selvi, X.Chu, TMH.
8. Java and Object Orientation, an introduction, John Hunt, second edition, Springer.
9. Maurach's Beginning Java2 JDK 5, SPD.

VI. ELECTRONICS RESOURCES:

1. NPTEL – Programming in Java
   https://nptel.ac.in/courses/106/105/106105191
2. NPTEL – Object Oriented Programming
   https://nptel.ac.in/courses/106/105/106105153
3. Oracle Official Java Documentation
   https://docs.oracle.com/en/java/
4. Oracle Java Tutorials
   https://docs.oracle.com/javase/tutorial/
5. OpenJDK Official Site
   https://openjdk.org/
6. GeeksforGeeks – Java Programming
   https://www.geeksforgeeks.org/java/
7. TutorialsPoint – Java Tutorial
   https://www.tutorialspoint.com/java/
8. W3Schools – Java Tutorial
   https://www.w3schools.com/java/
9. MIT OpenCourseWare – Object-Oriented Programming
   https://ocw.mit.edu
10. IEEE Xplore Digital Library
    https://ieeexplore.ieee.org

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

SOFTWARE ENGINEERING								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS304PC	**Professional Core**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic knowledge of programming concepts, data structures.								

I. COURSE OVERVIEW:

Software Engineering focuses on systematic, disciplined, and quantifiable approaches to the development, operation, and maintenance of software. This course provides knowledge of software process models, requirements engineering, software design, testing strategies, metrics, risk management, quality assurance, and UML modeling techniques essential for managing large-scale software projects.

I. COURSE OBJECTIVES:

The students will try to learn:

I. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.

II. Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

II. C

OURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Ability to translate end-user requirements into system and software requirements, using e.g.
CO2:	UML, and structure the requirements in a Software Requirements Document (SRD).
CO3:	Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
CO4:	Will have experience and/or awareness of testing problems and will be able to develop a simple testing report
CO5:	Identify, analyze, and manage software risks and quality assurance activities

III. COURSE CONTENT:

UNIT- I:

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths. A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI). Process models: The waterfall model, Spiral model, Incremental Process Models, Concurrent Models, Component based development and Agile Development.

UNIT – II:

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.
Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT - III:

Design Engineering: Design process and design quality, design concepts, the design model.
Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, use case diagrams, class diagrams, sequence diagrams, collaboration diagrams, activity diagrams and component diagrams.

UNIT - IV:

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.
Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT – V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM. Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

IV. TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

V. REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
3. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.
4. Fundamentals of Software Engineering-Rajib Mall, PHI.

VI. ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/106/105/106105193
2. https://www.tutorialspoint.com/software_engineering
3. https://www.geeksforgeeks.org/software-engineering/
4. https://ocw.mit.edu
5. https://ieeexplore.ieee.or

COURSE CONTENT

DATABASE MANAGEMENT SYSYTEMS								
III Semester: C S E					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS305PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Data Structures								

I. COURSE OVERVIEW:

The Database Management Systems (DBMS) course introduces students to data modeling, database design, relational databases, SQL, transactions, and indexing techniques. Students learn how to design, implement, and maintain databases effectively, focusing on data integrity, concurrency, and performance.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To understand the basic concepts and the applications of database systems.
2. To master the basics of SQL and construct queries using SQL.
3. Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Gain knowledge of fundamentals of DBMS.
CO2:	Apply relational model concepts, enforce integrity constraints, use relational algebra and calculus, and design logical database schemas with views.
CO3:	Master the basics of SQL for retrieval and management of data.
CO4:	Be acquainted with the basics of transaction processing and concurrency control.
CO5:	Familiarity with database storage structures and access techniques

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

Database System Applications: A Historical Perspective, File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model

UNIT – II: (14 Hrs)

Introduction to the Relational Model: Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical database design, introduction to views, destroying/altering tables and views. Relational Algebra, Tuple relational Calculus, Domain relational calculus.

UNIT - III: (10 Hrs)

SQL: QUERIES, CONSTRAINTS, TRIGGERS: form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases.

Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multivalued dependencies, FOURTH normal form, FIFTH normal form.

UNIT - IV: (12 Hrs)

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions.

UNIT - V: (10 Hrs)

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree based Indexing, Comparison of File Organizations, Indexes- Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

V:TEXT BOOKS:

1. Database System Concepts, Silberschatz, Korth, McGraw hill, V edition.3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill.

VI: REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

ELECTRONICS RESOURCES

1. https://nptel.ac.in/courses/106/105/106105064/
2. https://www.geeksforgeeks.org/relational-algebra/
3. https://www.tutorialspoint.com/sql/index.htm
4. https://www.w3schools.in/dbms/indexing/

COURSE CONTENT

INNOVATION AND ENTREPRENEURSHIP								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MS306HS	**Humanities & Social Sciences**	2	0	-	2	40	60	100
Contact Classes: 30	**Tutorial Classes: 00**	**Practical Classes: Nil**				**Total Classes: 30**		
Prerequisite: Programming for problem solving								

I. COURSE OVERVIEW:

The Innovation and Entrepreneurship course introduces students to the fundamentals of entrepreneurship, innovation processes, and startup development. It emphasizes problem identification, opportunity assessment, prototype development, business planning, and intellectual property.

II. COURSE OBJECTIVES:

1. To familiarize on the basic concepts of innovation, entrepreneurship and its importance.
2. To Identify and analyze the process of problem-opportunity identification, market segmentation, and idea generation techniques.
3. To initiate prototype development and understand minimum viable product.
4. To develop initial Business and financial planning and Go-to-Market strategies
5. To impart knowledge on establishing startups, venture pitching and IPR

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand the entrepreneurship and the entrepreneurial process and its significance in economic development.
CO2:	Assess the problem from an industry perspective and generate solutions using the design thinking principles.
CO3:	Assess market competition, estimate market size, and develop a prototype.
CO4:	Analyze Business and financial planning models and Go-to-Market strategies.
CO5:	Able to build a start-up, register IP and identify funding opportunities.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

Fundamentals of Innovation and Entrepreneurship

Innovation: Introduction, need for innovation, Features, Types of innovations, innovations in manufacturing and service sectors, fostering a culture of innovation, planning for innovation.

Entrepreneurship: Introduction, types of entrepreneurship attributes, mindset of entrepreneurial and intrapreneurial leadership, Role of entrepreneurs in economic development. Woman Entrepreneurship, Importance of on-campus startups. Understanding to build entrepreneurial mindset, attributes and networks individuals while on campus.

Core Teaching Tool: Simulation, Game, Industry Case Studies (Personalized for students — 16 industries to choose from), Venture Activity.

UNIT – II: (14 Hrs)

Intr Problem and Customer Identification

Identification of gap, problem, analyzing the problem from a industry perspective, real-world problems**,** market and customer segmentation, validation of customer problem fit, Iterating problem-customer fit, Competition and Industry trends mapping and assessing initial opportunity, Porter's Five Force Model. Idea generation, Ideation techniques: Brainstorming, Brain writing, Round robin, and SCAMPER, Design thinking principles, Mapping of solution to problem.

Core Teaching Tool: Several types of activities including: Class, game, Gen AI, 'Get out of the Building' and Venture Activity.

UNIT - III: (10 Hrs)

Opportunity assessment and Prototype development

Identify and map global competitors, review industry trends, and understand market sizing: TAM, SAM, and SOM. Assessing scope and potential scale for the opportunity.

Understanding prototyping and Minimum Viable Product (MVP). Developing a prototype: Testing, and validation.

Core Teaching Tool: Venture Activity, no-code Innovation tools, Class activity

UNIT - IV: (12 Hrs)

Business & Financial Models

Introduction to Business Model and types, Lean Canvas Approach: 9-block lean canvas model, building lean canvas for your startup. Business planning: components of Business plan- Sales plan, People plan and financial plan, Financial Planning: Types of costs, preparing a financial plan for profitability using a financial template, understanding the basics of Unit economics, Economies of Scale and analyzing financial performance. Go-To-Market (GTM) approach — Selecting the Right Channel, creating digital presence, and building customer acquisition strategy.

Core Teaching Tool: Founder Case Studies – Sama and Securely Share; Class activity and discussions; Venture Activities.

UNIT - V: (10 Hrs)

Startups and IPR

Startup requirements, building founding team members and mentors, pitch preparation, start-up registration process, funding opportunities and schemes, institutional support to entrepreneurs, startup lifecycle, documentation, legal aspects in startup, venture pitching readiness, National Innovation Startup Policy (NISP) and its features.

Patents, Designs, Patentability, Procedure for grants of patents. Indian Scenario of Patenting, International Scenario: International cooperation on Intellectual Property. Patent Rights: Scope of Patent Rights. Copyright, trademark, and GI. Licensing and transfer of technology.

Core Teaching Tool: Expert talks; Cases; Class activity and discussions; Venture Activities.

V:TEXT BOOKS:

1. John R Bessant, Joe Tidd, Innovation and Entrepreneurship, 4E, Wiley, Latest Edition.
2. Ajay Batra, The Stratup Launch Book- A Practical Guide for Launching Customer Centric Ventures, Wiley, 2020. (For Core Teaching Tool).
3. Entrepreneurship Development and Small Business Enterprises, Poornima M Charantimath, 3E, Pearson, 2018.

VI: REFERENCE BOOKS:

1. D.F. Kuratko and T.V. Rao, Entrepreneurship: A South-Asian Perspective, Cengage Learning, 2013.
2. Robert D. Hisrich, Michael P. Peters, Dean A. Shepherd, Sabyasachi Sinha (2020). Entrepreneurship, McGrawHill, 11th Edition.

ELECTRONICS RESOURCES

1. NISP -Brochure inside pages - startup_policy_2019.pdf
2. https://nptel.ac.in/courses/110/104/110104083/
3. https://www.ideo.com/post/design-kit
4. https://theleanstartup.com/principles
5. https://blog.hubspot.com/sales/go-to-market-strategy
6. https://www.startupindia.gov.in/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS307PC	Professional Core	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Programming for problem solving								

I. COURSE OVERVIEW:

This laboratory course is designed to provide hands-on experience in Java programming, reinforcing object-oriented concepts through practical implementation. The course enables students to design, implement, and test Java programs covering OOP principles, strings, exception handling, multithreading, file handling, collections framework, GUI programming, event handling, and applet-based applications.

II. COURSE OBJECTIVES:

1. Understand and apply fundamental object-oriented programming concepts such as classes, objects, inheritance, polymorphism, encapsulation, and abstraction using Java.
2. Develop Java programs using constructors, method and constructor overloading, overriding, dynamic binding, keywords, and access specifiers.
3. Implement string processing, file handling, and exception handling techniques to build robust Java applications.
4. Design and implement multithreaded programs including inter-thread communication and synchronization concepts.
5. Use Java Collections Framework and apply appropriate data structures for efficient data storage and manipulation.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Analyze a problem, identify and define the computing requirements appropriate to its solution using object-oriented programming concepts.
CO2:	Design the applications using Inheritance, Polymorphism and Synchronization concepts.
CO3:	Handle exceptions at Compile time and Run time.
CO4:	Solve the real-world problems using Java Collection framework.
CO5:	Develop GUI applications using Applets, AWT and Swings.

IV: COURSE CONTENT:

Note:

1. Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of the Eclipse platform.

2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

List of Experiments:

1. Use Eclipse or Net bean platform and acquaint yourself with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.

2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.

3.
   A) Develop an applet in Java that displays a simple message.
   B) Develop an applet in Java that receives an integer in one text field, and computes its factorial

4. Value and returns it in another text field, when the button named "Compute" is clicked.

5. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.

6. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer every 1 second and if the value is even, the second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of the cube of the number.

7. Write a Java program for the following: Create a doubly linked list of elements. Delete a given element from the above list.
   Display the contents of the list after deletion.

8. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in the selected color. Initially, there is no message shown.

9. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

10. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas.
11. Write a java program to display the table using Labels in Grid Layout.

12. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

13. Write a Java program that loads names and phone numbers from a text file where the data is
14. organized as one line per record and each field in a record are separated by a tab (\t). It takes a
15. name or phone number as input and prints the corresponding other value from the hash table (hint:
16. use hash tables).

17. Write a Java program that correctly implements the producer – consumer problem using the
18. concept of inter thread communication.

19. Write a Java program to list all the files in a directory including the files present in all its
20. subdirectories.

V. TEXT BOOKS:

1. *Java for Programmers*, P. J. Deitel & H. M. Deitel, 10th Edition.
2. *Thinking in Java*, Bruce Eckel.

VI. REFERENCE BOOKS:

1. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
2. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

VII. ELECTRONICS RESOURCES:

1. https://docs.oracle.com/javase/tutorial/java/concepts/
2. https://docs.oracle.com/javase/8/docs/api/java/util/StringTokenizer.html
3. https://www.geeksforgeeks.org/method-overriding-in-java/

COURSE CONTENT

SOFTWARE ENGINEERING LAB								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS308PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Programming for problem solving & Computer Organization and Architecture.								

I. COURSE OVERVIEW:

The Software Engineering Laboratory aims to provide hands-on experience in applying software engineering principles to real-world projects. Students work in teams or individually to develop software project documentation and artifacts by following a systematic software development life cycle (SDLC) approach.

II. COURSE OBJECTIVES:

The students will try to learn:

To have hands-on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Ability to translate end-user requirements into system and software requirements.
CO2:	Ability to generate a high-level design of the system from the software requirements.
CO3:	Develop software configuration management and risk management documents for a software project.
CO4:	Use CASE tools to model and design software systems using UML diagrams.
CO5:	Design and develop test cases for unit testing, integration testing, and apply white-box and black-box testing techniques.

IV: COURSE CONTENT:

List of Experiments:

Do the following seven exercises for any two projects given in the list of sample projects or any other Projects:

1. Development of problem statements.
2. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3. Preparation of Software Configuration Management and Risk Management related documents.
4. Study and usage of any Design phase CASE tool
5. Performing the Design by using any Design phase CASE tools.
6. Develop test cases for unit testing and integration testing
7. Develop test cases for various white box and black box testing techniques.

Sample Project:

1. Passport automation System
2. Book Bank
3. Online Exam Registration
4. Stock Maintenance System
5. Online course reservation system
6. E-ticketing
7. Software Personnel Management System
8. Credit Card Processing
9. E-book management System.
10. Recruitment system

IV: TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.

V: REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley. Operating System - A Design Approach-Crowley, TMH.
2. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-HilL.

IV. ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/106/106/106106182/
2. https://www.tutorialspoint.com/software_engineering/
3. https://www.geeksforgeeks.org/software-engineering/
4. https://staruml.io/
5. https://www.visual-paradigm.com/
6. http://argouml.tigris.org/

COURSE CONTENT

DATABASE MANAGEMENT SYSYTEMS LAB								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS309PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Data Structures								

I. COURSE OVERVIEW:

The Database Management Systems Lab provides hands-on practical exposure to the concepts of database design and implementation. This laboratory course complements the DBMS theory course by enabling students to apply conceptual, logical, and physical database design principles through structured experiments.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Design conceptual and logical database schemas using ER modeling and relational modeling techniques for real-world applications.
CO2:	Apply normalization techniques to database schemas to eliminate redundancy and ensure data integrity.
CO3:	Acquire skills in using SQL commands for data definition and data manipulation.
CO4:	Write advanced SQL queries using joins, subqueries, aggregate functions, views, and constraints.
CO5:	Develop solutions for database applications using procedures, cursors and triggers

IV: COURSE CONTENT:

list of Experiments:

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A) Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.) B) Nested, Correlated subqueries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

V:TEXT BOOKS:

1. Database System Concepts, Silberschatz, Korth, McGraw hill, V edition.3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill.

VI: REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

ELECTRONICS RESOURCES

1. https://nptel.ac.in/courses/106/105/106105064/
2. https://www.geeksforgeeks.org/relational-algebra/
3. https://www.tutorialspoint.com/sql/index.htm
4. https://www.w3schools.in/dbms/indexing/

COURSE CONTENT

NODE JS/ REACT JS/ DJANGO								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
25CS310SD	Skill Development Course	**L**	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
		0	0	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisites: Object Oriented Programming through Java, HTML Basics								

I. COURSE OVERVIEW:

This course is designed to provide students with hands-on experience in modern full stack web application development. It introduces both client-side and server-side technologies and guides learners through building complete, scalable, and interactive web applications.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To implement the static web pages using HTML and do client-side validation using JavaScript.
2. To design and work with databases using Java
3. To develop an end to end application using java full stack.
4. To introduce Node JS implementation for server-side programming.
5. To experiment with single page application development using React.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Build a custom website with HTML, CSS, and Bootstrap and little JavaScript.
CO2:	Demonstrate Advanced features of JavaScript and learn about JDBC
CO3:	Develop Server – side implementation using Java technologies like
CO4:	Develop the server – side implementation using Node JS.
CO5:	Design a Single Page Application using React.

IV: COURSE CONTENT:

Exercises:

1. Build a responsive web application for shopping cart with registration, login, catalog and cart pages using CSS3 features, flex and grid.
2. Make the above web application responsive web application using Bootstrap framework.
3. Use JavaScript for doing client – side validation of the pages implemented in experiment 1 and experiment 2.
4. Explore the features of ES6 like arrow functions, callbacks, promises, async/await. Implement an application for reading the weather information from openweathermap.org and display the information in the form of a graph on the web page.
5. Develop a java stand alone application that connects with the database (Oracle / mySql) and perform the

CRUD operation on the database tables.

6. Create an xml for the bookstore. Validate the same using both DTD and XSD.
7. Design a controller with servlet that provides the interaction with application developed in experiment 1 and the database created in experiment 5.
8. Maintaining the transactional history of any user is very important. Explore the various session tracking mechanism (Cookies, HTTP Session)
9. Create a custom server using http module and explore the other modules of Node JS like OS, path, event.
10. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)
11. For the above application create authorized end points using JWT (JSON Web Token).
12. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages.
13. Create a service in react that fetches the weather information from openweathermap.org and the display the current and historical weather information using graphical representation using chart.js
14. Create a TODO application in react with necessary components and deploy it into GitHub.

V: REFERENCE BOOKS:

1. Jon Duckett, Beginning HTML, XHTML, CSS, and JavaScript, Wrox Publications, 2010.
2. Bryan Basham, Kathy Sierra and Bert Bates, Head First Servlets and JSP, O'Reilly Media, 2nd Edition, 2008.
3. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node ,2nd Edition, APress.

VI: ELECTRONICS RESOURCES:

1. **W3Schools** – Tutorials and references for HTML, CSS, JavaScript, Bootstrap, Node.js, and React https://www.w3schools.com
2. **MDN Web Docs (Mozilla Developer Network)** – Comprehensive documentation on web standards, JavaScript, and web APIs https://developer.mozilla.org
3. **Node.js Official Documentation** – Guides and API references for Node.js https://nodejs.org/en/docs
4. **Express.js Documentation** – Official documentation for Express framework https://expressjs.com
5. **React Official Documentation** – Concepts, hooks, routing, and best practices for React https://react.dev
6. **OpenWeatherMap API** – Weather data APIs used for real-time and historical weather applications https://openweathermap.org/api
7. **Chart.js Documentation** – JavaScript library for data visualization and charts https://www.chartjs.org/docs
8. **Oracle Java Documentation** – JDBC, Servlets, and core Java concepts https://docs.oracle.com/en/java
9. **Postman Learning Center** – API testing and RESTful service interaction https://learning.postman.com
10. **GitHub** – Source code hosting and deployment for React applications https://github.com

COURSE CONTENT

ENVIRONMENTAL SCIENCE								
III Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
25VA300ES	**Engineering & Sciences**	L	T	P	C	CIE	SEE	Total
		1	-	-	1	40	60	100
Contact Classes: 15	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 15**		
Prerequisite: No prerequisites.								

I. COURSE OVERVIEW:

Environmental Science is a foundational course aimed at developing awareness and understanding of the environment, natural resources, biodiversity, pollution, and sustainability challenges faced by modern society. The course emphasizes the interrelationship between human activities and natural ecosystems, highlighting the need for responsible environmental stewardship.

I. COURSE OBJECTIVES:

The students will try to learn:

I. Understand the components, structure, and functions of ecosystems and their relevance to human society.
II. Comprehend classification, sustainable management, and challenges of natural resources including water, minerals, land, forests, and energy.
III. Grasp the significance, value, and conservation approaches for biodiversity, including threats and legislative frameworks.
IV. Analyze types, sources, and impacts of environmental pollution, and learn technological and policy measures for pollution prevention and control.
V. Develop awareness about global environmental challenges, international agreements, and the role of policy, law, and Environmental Impact Assessment (EIA) in sustainable development.

II. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand the structure, function, and significance of ecosystems, including energy flow, biogeochemical cycles, and biodiversity conservation through field experiences.
CO2:	Analyze the classification, utilization, and sustainable management of natural resources, along with alternative energy options.
CO3:	Evaluate biodiversity at genetic, species, and ecosystem levels, its values, threats, and conservation methods under national and international frameworks.
CO4:	Identify types, sources, and impacts of environmental pollution, and apply suitable control technologies while assessing global environmental challenges and protocols.
CO5:	Interpret environmental policies, legislation, and the EIA process to propose management plans addressing contemporary environmental and sustainability issues.

III. COURSE CONTENT:

UNIT- I:

Ecosystems: Definition, Scope, and Importance of ecosystem. Classification, structure, and function of an ecosystem, Food chains, food webs, and ecological pyramids. Flow of energy, Biogeochemical cycles, Bioaccumulation, Bio magnification, ecosystem value, services and carrying capacity, Field visits.

UNIT – II:

Natural Resources: Classification of Resources: Living and Non-Living resources, **water resources:** use and over utilization of surface and ground water, floods and droughts, Dams: benefits and problems. **Mineral resources:** use and exploitation, environmental effects of extracting and using mineral resources, **Land resources:** Forest resources, **Energy resources:** growing energy needs, renewable and non-renewable energy sources, use of alternate energy source, case studies.

UNIT - III:

. **Biodiversity and Biotic Resources:** Introduction, Definition, genetic, species and ecosystem diversity. Value of biodiversity; consumptive use, productive use, social, ethical, aesthetic and optional values. India as a mega diversity nation, Hot spots of biodiversity. Field visit. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts; conservation of biodiversity: In-Situ and Ex-situ conservation. National Biodiversity act.

UNIT - IV:

Environmental Pollution and Control Technologies: Environmental Pollution: Classification of pollution, **Air Pollution:** Primary and secondary pollutants, Automobile and Industrial pollution, Ambient air quality standards. **Water pollution:** Sources and types of pollution, drinking water quality standards.

Soil Pollution: Sources and types, Impacts of modern agriculture, degradation of soil. **Noise Pollution:** Sources and Health hazards, standards, **Solid waste:** Municipal Solid Waste management, composition and characteristics of e-Waste and its management. **Pollution control technologies:** Wastewater Treatment methods: Primary, secondary and Tertiary.
Overview of air pollution control technologies, Concepts of bioremediation. **Global Environmental Issues and Global Efforts: C**limate change and impacts on human environment. Ozone depletion and Ozone depleting substances (ODS). Deforestation and desertification. International conventions / Protocols: Earth summit, Kyoto protocol, and Montréal Protocol, NAPCC-GoI Initiatives.

UNIT – V

Environmental Policy, Legislation & EIA: Environmental Protection act, Legal aspects Air Act-1981, Water Act, Forest Act, Wild life Act, Municipal solid waste management and handling rules, biomedical waste management and handling rules, hazardous waste management and handling rules. EIA: EIA structure, methods of baseline data acquisition. Overview on Impacts of air, water, biological and Socio- economical aspects. Strategies for risk assessment, Concepts of Environmental Management Plan(EMP). Contemporary Environmental Issues Climate change; Sustainable development goals (SDGs); Global environmental challenges; Environmental policies and international agreements.

IV. TEXT BOOKS:

1. Introduction to Environmental Science by Y. Anjaneyulu, BS. Publications.
2. Textbook of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission.
3. Environmental Studies by R. Rajagopalan, Oxford University Press.

V. REFERENCE BOOKS:

1. Environmental Science: towards a sustainable future by Richard T. Wright. 2008 PHL Learning Private Ltd. New Delhi.
2. Environmental Engineering and science by Gilbert M. Masters and Wendell P. Ela. 2008 PHI Learning Pvt. Ltd.
3. Environmental Science by Daniel B. Botkin & Edward A. Keller, Wiley INDIA edition.
4. Environmental Studies by Anubha Kaushik, 4th Edition, New age international publishers.
5. Text book of Environmental Science and Technology - Dr. M. Anji Reddy 2007, BS Publications.

VI. ELECTRONICS RESOURCES:

1. Ministry of Environment, Forest and Climate Change (MoEFCC), Government of India
   Policies, acts, rules, environmental programs, and NAPCC initiatives
   https://moef.gov.in
2. United Nations Environment Programme (UNEP)
   Global environmental issues, reports, conventions, and sustainability initiatives
   https://www.unep.org
3. World Health Organization (WHO) – Environment and Health
   Information on air, water, and noise pollution and health impacts
   https://www.who.int/health-topics/environmental-health
4. Intergovernmental Panel on Climate Change (IPCC)
   Scientific reports on climate change, impacts, mitigation, and adaptation
   https://www.ipcc.ch
5. National Biodiversity Authority (NBA), India
   Biodiversity conservation, National Biodiversity Act, and related resources
   https://nbaindia.org
6. Central Pollution Control Board (CPCB), India
   Pollution standards, air and water quality, waste management guidelines
   https://cpcb.nic.in
7. World Wide Fund for Nature (WWF)
   Biodiversity conservation, ecosystems, and sustainable development resources
   https://www.worldwildlife.org
8. Food and Agriculture Organization (FAO), United Nations
   Resources on forests, land, water, agriculture, and sustainable resource management
   https://www.fao.org
9. NPTEL – Environmental Science / Environmental Studies Courses
   Video lectures by IIT professors aligned with Indian curricula
   https://nptel.ac.in
10. Environmental Impact Assessment (EIA) Online Portal – India
    Information on EIA processes, notifications, and project clearances
    https://environmentclearance.nic.in

COURSE CONTENT

COMPUTER ORIENTED STATISTICAL METHODS								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA401PC	**Foundation**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic knowledge of probability and statistics								

I. COURSE OVERVIEW:

This course focuses on applying statistical techniques and methods using computers. They will also be able to apply these ideas to a wide range of problems that include the engineering applications.

II. COURSE OBJECTIVES:

The students will try to learn:

1. The theory of Probability, Probability distributions of single and multiple random variable
2. The Statistical Methods of studying data samples
3. The sampling theory, testing of hypothesis and making statistical inferences.
4. Stochastic process and Markov chains

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Apply the concept of probability and Random variables in solving given Real time problems.
CO2:	Formulate and solve problems involving random variables and apply statistical methods for analyzing experimental data.
CO3:	Apply various concepts of distribution to find out continuous distribution values
CO4:	Apply concepts of estimation and testing of hypothesis to case studies.
CO5:	Describe Stochastic Principles to simplify processes that satisfy Markov property.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12	
CO1	2	1		2								2	2
CO2	2	3			1							1	2
CO3	2	2			2							1	2
CO4	1	2			1								1
CO5	2	2			1							1	2
	1.8	2.0		2.0	1.3							1.3	1.8

V. COURSE CONTENT:

UNIT- I: (8 Hrs)

Random Variables and Probability Distributions

Concept of a Random Variable, DiscreteProbability Distributions, Continuous Probability Distributions. Mean of a Random Variable, Variance of a Random Variable.

Discrete Probability Distributions: Binomial Distribution, Poisson distribution.

UNIT – II: (10 Hrs)

Continuous and Sampling Distributions

Uniform Distribution, Normal Distribution, Areas under the Normal Curve, Applications of theNormal Distribution, Normal Approximation to the Binomial Distributions.

Fundamental Sampling Distributions: Random Sampling, Some Important Statistics, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem.

UNIT - III: (10 Hrs)

Estimation

Introduction, Statistical Inference, Classical Methods of Estimation, Single Sample: Estimation themean, standard error of a point estimate, prediction interval. Two samples: Estimating the differencebetween two means, Single sample: Estimating a proportion. Two samples: Estimating the difference between two proportions, two sample: Estimating the ratio of two variances.

UNIT - IV: (10 Hrs)

Tests of Hypotheses

Statistical Hypotheses: General Concepts, Testing a Statistical Hypothesis, Single sample: Testsconcerning a single mean, two samples: tests on two means. One sample: Test on a single proportion. Two samples: Tests on two proportions. Two sample tests concerning variances: F- distribution.

UNIT - V: (10 Hrs)

Stochastic Processes and Markov Chains

Introduction to Stochastic processes- Markov process. Transition Probability, Transition ProbabilityMatrix, First order and Higher order Markov process, n-step transition probabilities, Markov chain,Steady state condition, Markov analysis.

VI. TEXT BOOKS:

1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & Statistics for Engineers & Scientists, 9th Ed. Pearson Publishers.
2. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications.
3. S.D.Sharma, Operations Research, Kedarnath and Ramnath Publishers, Meerut, Delhi.

VII. REFERENCE BOOKS:

1. T.T. Soong, Fundamentals of Probability and Statistics for Engineers, John Wiley & Sons, 2004.
2. Sheldon M Ross, Probability and statistics for Engineers and scientists, academic press
3. Miller and Freund's, Probability and Statistics for Engineers, 8th Edition, Pearson Educations.

VIII. ELECTRONIC RESOURCES:

1. **NPTEL – Probability and Statistics Courses**
   Video lectures and course materials by IIT faculty aligned with engineering curricula
   https://nptel.ac.in
2. **MIT OpenCourseWare – Probability & Statistics**
   Free lecture notes, assignments, and exams for probability, statistics, and stochastic processes
   https://ocw.mit.edu
3. **Khan Academy – Statistics & Probability**
   Conceptual explanations and practice problems for probability distributions, sampling, and hypothesis testing
   https://www.khanacademy.org/math/statistics-probability
4. **StatTrek**
   Online tutorials, examples, and calculators for probability distributions, hypothesis testing, and estimation
   https://stattrek.com
5. **HyperStat Online Textbook**
   Interactive textbook covering probability theory, distributions, sampling, and inferential statistics
   https://www.itl.nist.gov/div898/handbook
6. **NIST/SEMATECH Engineering Statistics Handbook**
   Authoritative reference for statistical methods used in engineering and data analysis
   https://www.itl.nist.gov/div898/handbook
7. **Wolfram MathWorld – Probability & Statistics**
   Mathematical definitions, formulas, and explanations for random variables, distributions, and Markov chains
   https://mathworld.wolfram.com
8. **Towards Data Science (Medium)**
   Applied articles and case studies on probability, statistics, and stochastic modeling
   https://towardsdatascience.com
9. **Penn State Online – Probability and Statistics Notes**
   Well-structured lecture notes and examples
   https://online.stat.psu.edu

COURSE CONTENT

OPERATING SYSTEMS								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
25CS402PC	**PC**	**L**	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
		3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: A course on "Computer Programming and Data Structures". A course on "Computer Organization and Architecture".								

I. COURSE OVERVIEW:

This course explains how a computer manages hardware and software resources. Students learn about processes, memory management, file systems, scheduling, and security. It helps them understand how software runs efficiently on a computer.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Introduce operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection)
II. Introduce the issues to be considered in the design and development of operating system
III. Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Will be able to control access to a computer and the files that may be shared
CO2:	Demonstrate the knowledge of the components of computers and their respective roles in computing.
CO3:	Ability to recognize and resolve user problems with standard operating environments
CO4:	Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.
CO5:	Knowing the file system interfaces and operations on files.

IV. COURSE ARTICULATION MATRIX:

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	3	3	3	3	3	3	1	-	1	3	3	3	3
CO2	3	3	2	3	3	2	2	-	0	3	3	3	3
CO3	3	2	3	1	2	3	2	-	0	1	2	3	2
CO4	3	2	3	3	3	3	2	-	0	3	3	3	2
CO5	3	3	3	3	1	3	2	-	0	1	2	3	3

V. COURSE CONTENT:

UNIT- I: (10 Hrs.)

Operating System - Introduction, Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls
Process - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads.

UNIT – II: (10 Hrs.)

CPU Scheduling - Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, waitpid, exec
Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

UNIT - III: (9 Hrs.)

Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors **Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

UNIT - IV: (8 Hrs.)

Memory Management and Virtual Memory - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.

UNIT - V: (8 Hrs.)

File System Interface and Operations -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl system calls.

VI. TEXTBOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley
2. Advanced programming in the UNIX environment, W.R. Stevens, Pearson education.

VII. REFERENCE BOOKS:

1. Operating Systems- Internals and Design Principles, William Stallings, Fifth Edition– 2005, Pearson Education/PHI
2. Operating System A Design Approach- Crowley, TMH.
3. Modern Operating Systems, Andrew S. Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/ Pearson Education
5. UNIX Internals -The New Frontiers, U. Vahalia, Pearson Education.

VIII. ELECTRONICS RESOURCES:

1. NPTEL – Operating Systems https://nptel.ac.in/courses/106/106/106106183/
2. TutorialsPoint – OS https://www.tutorialspoint.com/operating_system/index.htm/
3. GeeksforGeeks – OS,https://www.geeksforgeeks.org/operating-systems/

COURSE CONTENT

ALGORITHMS DESIGN AND ANALYSIS								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
25CS403PC	**PC**	**L**	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
		3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Programming for problem solving and Data Structures								

I. COURSE OVERVIEW:

This subject focuses on designing efficient algorithms to solve problems. Students learn different algorithm techniques such as divide and conquer, greedy, and dynamic programming. They also study how to analyze the performance and efficiency of algorithms.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Develop proficiency in evaluating algorithms using asymptotic notations, including best-, average-, and worst-case time/space complexities, and solving related recurrence relations.
II. Master various algorithmic strategies—divide-and-conquer, greedy, dynamic programming, backtracking, and branch-and-bound—identifying suitable use cases and demonstrating their application.
III. Critically assess and contrast different algorithms in terms of efficiency, scalability, and correctness through rigorous analytical reasoning and empirical evaluation.
IV. Differentiate between tractable (polynomial-time) and intractable (super-polynomial or exponential-time) problems;
V. **Identify and classify** problems as P, NP, NP-hard, or NP-complete, and **assess** their relationships through polynomial-time reductions and Cook's theorem.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Able to Apply space and time complexity analysis using asymptotic notations
CO2:	Able to Design divide-and-conquer algorithms and critically assess their runtime and space trade-offs.
CO3:	Able to Device backtracking and dynamic programming solutions.
CO4:	Able to Apply greedy methods and graph traversal algorithms
CO5:	Able to Analyze and Design branch-and-bound algorithms for NP-hard problems.

IV. COURSE ARTICULATION MATRIX:

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2	3	3	3		2						2	2
CO2	3	3	3	3		2						2	2
CO3	3	3	3	3		2						2	2
CO4	3	3	3	3		2						2	2
CO5	3	3	3	3		2						2	2

V. COURSE CONTENT:

UNIT- I: (10 hrs.)

Introduction: Algorithm, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations- Big oh notation, Omega notation, Theta notation, and Little oh notation.
Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication.

UNIT – II: (8 hrs.)

Disjoint Sets: Disjoint set operations, union and find algorithms, Priority Queue- Heaps, Heap sort
Backtracking: General method, applications, n-queens problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

UNIT - III: (10 hrs.)

Dynamic Programming: General method, applications- Optimal binary search tree, 0/1 knapsack problem, All pairs shortest path problem, Traveling salesperson problem, Reliability design

UNIT - IV: (8 hrs.)

Greedy method: General method, applications- Job sequencing with deadlines, knapsack problem, Minimum cost spanning trees, Single source shortest path problem.
Basic Traversal and Search Techniques: Techniques for Binary Trees, Techniques for Graphs, Connected components, Bi-connected components.

UNIT - V: (9 hrs.)

Branch and Bound: General method, applications - Travelling salesperson problem, 0/1 knapsack problem - LC Branch and Bound solution, FIFO Branch and Bound solution.
NP-Hard and NP-Complete problems: Basic concepts, non-deterministic algorithms, NP - Hard and NP-Complete classes, Cook's theorem.

VI. TEXTBOOKS:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni, and Rajasekaran, University Press.

VII. REFERENCE BOOKS:

1. Design and Analysis of algorithms, Aho, Ullman, and Hopcroft, Pearson education.
2. Introduction to Algorithms, second edition, T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C.Stein, PHI Pvt. Ltd./ Pearson Education.
3. Algorithm Design: Foundations, Analysis and Internet Examples, M.T. Goodrich and R. Tamassia, John Wiley and Sons.

VIII. ELECTRONICS RESOURCES:

1. Coursera – Algorithms (Part I & II) https://www.coursera.org/specializations/algorithms/
2. MIT OpenCourseWare – Algorithms https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/
3. GeeksforGeeks – Algorithms https://www.geeksforgeeks.org/fundamentals-of-algorithms/

COURSE CONTENT

COMPUTER NETWORKS								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS404PC	**PC**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: A course on "Programming for problem solving".								

I. COURSE OVERVIEW:

This course introduces how computers communicate with each other. Students learn about network models, protocols, data transmission, and network security. It helps them understand the working of the Internet and communication systems.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Equip the students with a general overview of the concepts and fundamentals of computer networks.
II. Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.
III. Elucidate the students about working and implementation of protocols at various layers in protocols stack.
IV. Appreciating the protocol working by observing and analysing outputs of the packet sniffer,

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Gain the knowledge of the basic computer network technology.
CO2:	Gain the knowledge of the functions of each layer in the ISO-OSI and TCP/IP reference model.
CO3:	Obtain the skills of subnetting and routing mechanisms.
CO4:	Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation
CO5:	Understanding working of the protocols through traces captured by a packet sniffer

IV. COURSE ARTICULATION MATRIX:

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2				2	1						3	2
CO2	3	3	2	2	1							1	2
CO3	2	3	2	3	1							2	3
CO4	1			1								1	3
CO5	3	3	3	3	2	2	2					3	2

V. COURSE CONTENT:

UNIT- I: (10 Hrs.)

Introduction: The Internet, Protocol, Network Edge, Access Networks, Network Core, Packet Switching, Circuit Switching, Delay, Loss, and Throughput in Packet-Switched Networks, Protocol reference models: ISO-OSI, TCP/IP, Types of Network attacks, History of Computer Networking and the Internet.

UNIT – II: (14 Hrs.)

Application Layer: Principles of Network Applications, Network Application Architectures, Processes Communicating, Transport Services Available to Applications, Transport Services Provided by the Internet, Application-Layer Protocols, The Web and HTTP, File Transfer: FTP, Electronic Mail in the Internet, SMTP, DNS, Peer-to-Peer Applications, Socket Programming: Creating Network Applications.

UNIT - III: (12 Hrs.)

Transport Layer: Transport-Layer Services, Multiplexing and Demultiplexing, Connectionless Transport: UDP, Principles of Reliable Data Transfer, Building a Reliable Data Transfer Protocol, Pipelined Reliable Data Transfer Protocols, Go-Back-N (GBN), Selective Repeat (SR), Connection-Oriented Transport: TCP, The TCP Connection, Segment Structure, Round-Trip Time Estimation and Timeout, Reliable Data Transfer, Flow Control, TCP Connection Management, Principles of Congestion Control, TCP Congestion Control, Fairness.

UNIT - IV: (14 Hrs.)

Network Layer: Data and Control plane, Forwarding and Routing 308, Network Service Models, Virtual Circuit and Datagram Networks, Router working, The Internet Protocol (IP): Forwarding and Addressing in the Internet, Datagram Format, IPv4 Addressing, Internet Control Message Protocol (ICMP), IPv6, IP Security, Routing Algorithms- The Link-State (LS) Routing Algorithm, The Distance-Vector (DV) Routing Algorithm, Hierarchical Routing, Routing in the Internet-Intra-AS Routing in the Internet: RIP, Intra-AS Routing in the Internet: OSPF, Inter-AS Routing: BGP, Broadcast and Multicast Routing, Broadcast Routing Algorithms, Multicasting.

UNIT - V: (10 Hrs.)

The Link Layer: The Services Provided by the Link Layer, Error-Detection and -Correction Techniques-Parity Checks, Checksum Methods, Cyclic Redundancy Check (CRC), Hamming code, Multiple Access Links and Protocols, Channel Partitioning Protocols, Random Access Protocols, Taking-Turns Protocols, DOCSIS: The Link-Layer Protocol for Cable Internet Access, Switched Local Area Networks, Link-Layer Addressing and ARP, Ethernet, Link-Layer Switches, Virtual Local Area Networks (VLANs), Link Virtualization-Multiprotocol Label Switching (MPLS), Data Center Networking, A Day in the Life of a Web Page Request. Wireless network characteristics, Wireless LAN.

VI. TEXTBOOKS:

1. Computer Networking: A Top-Down Approach – James F.Kurose, Keith W. Ross, Pearson
2. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson/PHI

VII. REFERENCE BOOKS:

1. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH.

VIII. ELECTRONICS RESOURCES:

1. NPTEL – Computer Networks-- https://nptel.ac.in/courses/106/106/106106181/
2. GeeksforGeeks – Computer Networks-- https://www.geeksforgeeks.org/computer-network-tutorials/
3. TutorialsPoint – CN-- https://www.tutorialspoint.com/computer_networking/index.htm/

COURSE CONTENT

MACHINE LEARNING								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS405PC	**PC**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**			**Total Classes: 45**			
Prerequisite: Database Management System, Probability and Statistics								

I. COURSE OVERVIEW:

This course introduces the fundamental concepts, techniques, and applications of **Machine Learning (ML)**, focusing on how systems can learn from data and improve performance without explicit programming. It provides a balanced treatment of **theoretical foundations**, **algorithmic approaches**, and **practical considerations** involved in building intelligent data-driven models.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To introduce students to the basic concepts and techniques of Machine Learning.
2. To have a thorough understanding of the Supervised and Unsupervised learning techniques
3. To study the various probability-based learning techniques

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Distinguish between, supervised, unsupervised and semi-supervised learning.
CO2:	Understand algorithms for building classifiers applied on datasets of non-linearly separable classes
CO3:	Design an ensemble model to increase the classification accuracy
CO4:	Understand the principles of RL evolutionary computing algorithms
CO5:	Explain the working principles of artificial neural networks, deep learning, reinforcement learning, and evolutionary learning, and analyze their applications in real-world problems.

IV. COURSE CONTENT:

UNIT- I: (10 Hrs.)

Introduction to Machine Learning: Types of Human learning, machine learning process, Well-posed learning problem, Types of machine learning and comparison, applications of machine learning.

Model Preparation, Evaluation and feature engineering: Machine learning activities, Types of data in machine learning, dataset understanding, plotting and exploration, checking data quality, remediation, data pre-processing, selecting a model, predictive and descriptive models, supervised learning model training, cross-validation and boot strapping, lazy vs eager learner, interpreting the model- underfitting, overfitting, bias-variance trade-off. Parameter for evaluating performance of classification, regression, and clustering model. Improving performance of a model.

UNIT – II: (10 Hrs.)

Feature Engineering: Feature transformation - feature construction, feature extraction by PCA, SVD, LDA. Feature subset selection — feature relevancy and redundancy measures. Feature selection process and approaches.
Review of Probability concepts: joint probability, conditional probability, bayes rule, Common discrete and continuous distributions, dealing with multiple random variables, central limit theorem. Bayes classifier, Multi-class Classification, Naïve Bayes classifier, Bayesian belief network.

UNIT - III: (8 Hrs.)

Supervised Learning - Introduction to supervised learning,
Regression: Introduction of regression, Regression algorithms: Simple linear regression, Multiple linear regression, Polynomial regression model, Logistic regression, Maximum likelihood estimation.
Classification: Classification model and learning steps, Classification algorithms: Naïve Bayes classifier, Distance measures, k-Nearest Neighbor (kNN), Decision tree, Support vector machines, Kernel trick, Random Forest.

UNIT - IV: (8 Hrs.)

Unsupervised Learning: Introduction to unsupervised learning, Unsupervised vs supervised learning, Application of unsupervised learning, Clustering and its types, Partitioning method: k-Means and K- Medoids, Hierarchical clustering, Density-based methods – DBSCAN.

UNIT - V: (9 Hrs.)

Artificial Neural Network: Biological neuron, Artificial neuron, Activation functions, neural network architecture, perceptron, learning process in ANN, Back propagation.
Introduction to deep learning, overview of reinforcement learning, Representation learning, Evolutionary learning. Case-study of ML applications: Image recognition, Email spam filtering, Online fraud detection

V. TEXTBOOKS:

1. Saikat Dutt, S. Chjandramouli, Das – Machine Learning, Frist Edition, Pearson
2. M N Murty, Anathanarayana V S – Machine Learning, First Edition, University Press
3. Tom M Mitchell, ―Machine Learning, First Edition, McGraw Hill Education, 2013.

VI. REFERENCE BOOKS:

1. Stephen Marsland, ―Machine Learning – An Algorithmic Perspective, Second Edition,
2. Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.

VII. ELECTRONICS RESOURCES:

1. NPTEL – Machine Learning
2. https://nptel.ac.in/courses/106/106/106106139/
3. MIT OpenCourseWare – Introduction to Machine Learning
4. https://ocw.mit.edu/courses/6-036-introduction-to-machine-learning-fall-2020/
5. Stanford CS229: Machine Learning
6. https://cs229.stanford.edu/
7. Coursera – Machine Learning by Andrew Ng
8. https://www.coursera.org/learn/machine-learning
9. https://developers.google.com/machine-learning/crash-course
10. https://ieeexplore.ieee.org/

COURSE CONTENT

COMPUTATIONAL MATHEMATICS LAB (Using Python/MATLAB software)								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA406PC	**PC**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Matrices, Iterative methods and ordinary differential equations								

I. COURSE OVERVIEW:

This lab provides practical experience in solving mathematical problems using computers. Students apply numerical methods and computational techniques. It strengthens analytical thinking and technical skills.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Solve problems of Eigen values and Eigen Vectors using Python/MATLAB.
II. Solution of Algebraic and Transcendental Equations using Python/MATLAB
III. Solve problems of Linear system of equations
IV. Solve problems of **First-Order ODEs Higher order linear differential equations with** constant coefficients.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Develop the code to find the Eigen values and Eigen Vectors using Python/MATLAB.
CO2:	Develop the code find solution of Algebraic and Transcendental Equations.
CO3:	Develop the code find solution of Linear system of equations using Python/MATLAB
CO4:	Write the code to solve problems of **First-Order ODEs**
CO5:	Develop the **Higher order linear differential equations with constant coefficients**

*** Visualize all solutions Graphically through programmers**

IV. COURSE CONTENT:

UNIT- I: (6 Hrs.)

Eigen values and Eigenvectors:
Programs:

- Finding real and complex Eigen values.
- Finding Eigen vectors.

UNIT – II: (6 Hrs.)

Solution of Algebraic and Transcendental Equations

Bisection method, Newton Raphson Method

Programs:

- Root of a given equation using Bisection method.
- Root of a given equation Newton Raphson Method.

UNIT - III: (6 Hrs.)

Linear system of equations:

Jacobi's iteration method and Gauss-Seidal iteration method

Programs:

- Solution of given system of linear equations using Jacobi's method
- Solution of given system of linear equations using Gauss-Seidal method

UNIT - IV: (6 Hrs.)

First-Order ODEs

Exact and non-exact equations, Applications: exponential growth/decay, Newton's law of cooling.

Programs:

- Solving exact and non-exact equations
- Solving exponential growth/decay and Newton's law of cooling problems.

UNIT - V: (6 Hrs.)

Higher order linear differential equations with constant coefficients

Programs:

- Solving homogeneous ODEs
- Solving non-homogeneous ODEs

V. TEXTBOOKS:

1. MATLAB and its Applications in Engineering, Rajkumar Basal, Ashok Kumar Geo, Manoj Kumar Sharma, Pearson publication.
2. Kenneth A. Lambert, The fundamentals of Python: First Programs, 2011, Cengage Learnings.
3. Think Python First Edition, by Allen B. Downey, Orielly publishing.
4. Introduction to Python Programming, William Mitchell, Povel Solin, Martin Novak et al., NCLab Public Computing, 2012.
5. Introduction to Python Programming, ©Jacob Fredslund, 2007.

VI. REFERENCE BOOKS:

1. An Introduction to Python, John C. Lusth, The University of Alabama, 2011.
2. Introduction to Python, ©Dave Kuhlman, 2008.

VII. ELECTRONICS RESOURCES:

1. Numerical Methods – NPTEL — https://nptel.ac.in/courses/111/105/111105068/
2. MATLAB Onramp (Free course) — https://matlabacademy.mathworks.com/
3. Python for Scientific Computing — https://docs.scipy.org/doc/scipy/

COURSE CONTENT

OPERATING SYSTEMS LAB								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS407PC	**PC**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: A course on "Programming for Problem Solving". A course on "Computer Organization and Architecture".								

I. COURSE OVERVIEW:

This lab provides hands-on experience with operating system concepts. Students perform practical tasks on processes, memory management, file handling, and scheduling. It strengthens their understanding of how an OS works internally.

II. COURSE OBJECTIVES:

The students will try to learn:

I. To provide an understanding of the design aspects of operating system concepts through simulation.

II. Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Implement and evaluate CPU scheduling algorithms and page replacement policies to analyze system performance.
CO2:	Apply UNIX/Linux system calls to perform file operations, process communication, and synchronization.
CO3:	Design and implement deadlock avoidance and synchronization solutions using Banker's Algorithm and semaphores
CO4:	Implement and compare inter-process communication mechanisms such as Pipes, FIFOs, Message Queues, and Shared Memory.
CO5:	Simulate and analyse memory management techniques including Paging and Segmentation

IV. COURSE ARTICULATION MATRIX:

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	3	2	2	2	1							2	1
CO2	3	2	2	2	1							2	1
CO3	2	2	2	2	1							2	1
CO4	3	2	2	2	1							3	1
CO5	3	2	2	2	1							2	2

V. COURSE CONTENT:

List of Experiments:

1. Write C programs to simulate the following CPU Scheduling algorithms a) FCFS b) SJF c) Round Robin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, lseek, stat, fork, exit)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms a) Pipes b) FIFOs c) Message Queues d) Shared Memory
6. Write C programs to simulate the following memory management techniques a) Paging b) Segmentation
7. Write C programs to simulate Page replacement policies a) FCFS b) LRU c) Optimal.

VI. TEXTBOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley.
2. Advanced programming in the Unix environment, W. R. Stevens, Pearson education.

VII. REFERENCE BOOKS:

1. Operating Systems – Internals and Design Principles, William Stallings, Fifth Edition–2005, Pearson Education/PHI.
2. Operating System - A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI.
4. UNIX Programming Environment, Kernighan and Pike, PHI/Pearson Education.
5. UNIX Internals: The New Frontiers, U. Vahalia, Pearson Education.

VIII. ELECTRONICS RESOURCES:

1. OS Lab Exercises – GeeksforGeeks https://www.geeksforgeeks.org/os-lab-experiments/
2. GitHub OS Lab Codes https://github.com/search?q=operating+systems+lab
3. YouTube – OS Lab Tutorials Search “Operating Systems lab experiments”

COURSE CONTENT

COMPUTER NETWORKS LAB								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS408PC	**PC**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basics of C programming fundamentals								

I. COURSE OVERVIEW:

This course focuses on practical networking skills. Students work with network configurations, protocols, and simulations. It helps them understand real-time communication between computers.

II. COURSE OBJECTIVES:

The students will try to learn:

I. To understand the working principle of various communication protocols.
II. To understand the network simulator environment and visualize a network topology and observe its performance
III. To analyze the traffic flow and the contents of protocol frames.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Implement data link layer farming methods
CO2:	Analyze error detection and error correction codes.
CO3:	Implement and analyze routing and congestion issues in network design.
CO4:	Implement Encoding and Decoding techniques used in presentation layer
CO5:	To be able to work with different network tools

I. COURSE ARTICULATION MATRIX:

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	1	1	2									1	
CO2	2	2	2	2								1	2
CO3	2	1	3	3								2	2
CO4	1	2	2						2	1	2	1	
CO5	2	2	1	2	3	2	1	1				2	2

IV. COURSE CONTENT:

List of Experiments:

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijsktra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting techniques used in buffers.
10. Wireshark
    i. Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.
1. How to run Nmap scan
2. Operating System Detection using Nmap
3. Do the following using NS2 Simulator
    I. NS2 Simulator-Introduction
    II. Simulate to Find the Number of Packets Dropped
    III. Simulate to Find the Number of Packets Dropped by TCP/UDP
    IV. Simulate to Find the Number of Packets Dropped due to Congestion
    V. Simulate to Compare Data Rate & Throughput.
    VI. Simulate to Plot Congestion for Different Source/Destination
    VII. Simulate to Determine the Performance with respect to Transmission of Packets

V. TEXTBOOKS:

1. Computer Networks, Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI

VI. REFERENCE BOOKS:

1. An Engineering Approach to Computer Networks, S.Keshav, 2nd Edition, Pearson Education
2. Data Communications and Networking – Behrouz A. Forouzan. 3rd Edition, TMH.

VII. ELECTRONICS RESOURCES:

1. TutorialsPoint https://www.tutorialspoint.com/computer_networking/computer_networking_lab_manual.htm
2. Cisco Packet Tracer Tutorials https://www.netacad.com/courses/packet-tracer
3. YouTube – CN Lab Search "Computer Networks lab experiments"

COURSE CONTENT

MACHINE LEARNING LAB								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS409PC	Professional Core	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Data Structures								

I. COURSE OVERVIEW:

The **Machine Learning Lab** is designed to provide hands-on experience with fundamental and advanced machine learning techniques using **Python programming**. This laboratory course complements the Machine Learning theory course by enabling students to **implement, experiment with, and analyze** various machine learning algorithms on real-world datasets.

II. COURSE OBJECTIVES:

The students will try to learn:

1. The objective of this lab is to get an overview of the various machine learning techniques and can demonstrate them using python.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand modern notions in predictive data analysis
CO2:	Select data, model selection, model complexity and identify the trends
CO3:	Understand a range of machine learning algorithms along with their strengths and weaknesses
CO4:	Build predictive models from data and analyze their performance
CO5:	Analyze, compare, and evaluate the performance of machine learning models and develop a mini-project to solve a real-world predictive analytics problem.

IV: COURSE CONTENT:

list of Experiments:

1. Write a python program to compute Central Tendency Measures: Mean, Median, Mode, Measure of Dispersion: Variance, Standard Deviation
2. Study of Python Basic Libraries such as Statistics, Math, Numpy and Scipy
3. Study of Python Libraries for ML application such as Pandas and Matplotlib
4. Write a Python program to implement Simple Linear Regression
5. Implementation of Multiple Linear Regression for House Price Prediction using sklearn
6. Implementation of Decision tree using sklearn and its parameter tuning
7. Implementation of KNN using sklearn
8. Implementation of Logistic Regression using sklearn
9. Implementation of K-Means Clustering
10. Performance analysis of Classification Algorithms on a specific dataset (Mini Project)

V:TEXT BOOKS:

1. Machine Learning – Tom M. Mitchell, - MGH.

VI: REFERENCE BOOKS:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis.

ELECTRONICS RESOURCES:

- Kaggle Learn
  - Hands-on tutorials for Python, Pandas, Matplotlib, scikit-learn, and machine learning experiments https://www.kaggle.com/learn
- Scikit-learn Official Documentation
  - Implementation guides, example datasets, and API reference for ML algorithms https://scikit-learn.org/stable/
- Google Colab
  - Online Python environment to run ML experiments without installation https://colab.research.google.com/
- Python Libraries Documentation
  - NumPy: https://numpy.org/doc/
  - Pandas: https://pandas.pydata.org/docs/
  - Matplotlib: https://matplotlib.org/stable/contents.html
  - SciPy: https://docs.scipy.org/doc/scipy/
- MIT OpenCourseWare – Python for Data Science
  - https://ocw.mit.edu/courses/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/

COURSE CONTENT

DATA VISUALIZATION - R PROGRAMMING/PYTHON/ POWER BI								
IV Semester: CSE					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS410SD	**Foundation**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**			**Total Classes: 30**			
Prerequisite: Basic knowledge of programming								

I. COURSE OVERVIEW:

This course teaches how to represent data visually for better understanding and decision making. Students learn tools like R and Power BI to create charts, dashboards, and reports. It helps in data analysis and business intelligence.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Effective use of Business Intelligence (BI) technology (Tableau) to apply data visualization
II. To discern patterns and relationships in the data.
III. To build Dashboard applications.
IV. To communicate the results clearly and concisely.
V. To be able to work with different formats of data sets.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand How to import data into Tableau
CO2:	Understand Tableau concepts of Dimensions and Measures.
CO3:	Develop Programs and understand how to map Visual Layouts and Graphical Properties.
CO4:	Create a Dashboard that links multiple visualizations.
CO5:	Use graphical user interfaces to create Frames for providing solutions to real world problems.

I. COURSE ARTICULATION MATRIX:

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	1		1		2	1					1	1	1
CO2	2	1			1	1			1	1	2	1	1
CO3	1	2	1		1	1				1	1	1	1
CO4		1	1		2				1			1	1
CO5	1	1	2	1	2	1					1	1	1

IV. COURSE CONTENT:

1. Understanding Data, What is data, where to find data, Foundations for building Data Visualizations, Creating Your First visualization?
2. Getting started with Tableau Software using Data file formats, connecting your Data to Tableau, creating basic charts(line, bar charts, Tree maps),Using the Show me panel.
3. Tableau Calculations, Overview of SUM, AVR, and Aggregate features, Creating custom calculations and fields.
4. Applying new data calculations to your visualizations, Formatting Visualizations, Formatting Tools and Menus, Formatting specific parts of the view.
5. Editing and Formatting Axes, Manipulating Data in Tableau data, Pivoting Tableau data.
6. Structuring your data, Sorting and filtering Tableau data, Pivoting Tableau data.
7. Advanced Visualization Tools: Using Filters, Using the Detail panel, using the Size panels, customizing filters, Using and Customizing tooltips, Formatting your data with colors.
8. Creating Dashboards & Storytelling, creating your first dashboard and Story, Design for different displays, adding interactivity to your Dashboard, Distributing & Publishing your Visualization.
9. Tableau file types, publishing to Tableau Online, Sharing your visualizations, printing, and Exporting.
10. Creating custom charts, cyclical data and circular area charts, Dual Axis charts.

V. REFERENCE BOOKS:

1. Microsoft Power BI cookbook, Brett Powell, 2nd edition.
2. R Programming for Data Science by Roger D. Peng (References)
3. The Art of R Programming by Norman Matloff Cengage Learning India.

VI. ELECTRONICS RESOURCES

1. RStudio – Learning https://rstudio.cloud/learn/primers/
2. DataCamp R Courses (free intro) https://www.datacamp.com/courses/free-introduction-to-r/
3. Microsoft Power BI Learning https://learn.microsoft.com/en-us/power-bi/
4. YouTube – Power BI Tutorials