R23 (B. Tech CSE)

Department: Computer Science & Engineering

Curriculum Structure & Syllabus

(Effective from 2023-24 admission batch)

1st Year 1st Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A. THEORY									
1	ENGG	Major	CS101	Programming for Problem Solving	3	0	0	3	3
2	SCI	Multidisciplinary	CH(CS)101	Engineering Chemistry	2	0	0	2	2
3	SCI	Multidisciplinary	M(CS)101	Engineering Mathematics – I	3	0	0	3	3
4	HUM	Ability Enhancement Course	HU101	Professional Communication	2	0	0	2	2
5	HUM	Value added course	HU102	Values and Ethics	2	0	0	2	2
6	HUM	Value added course	HU103	Constitution of India	1	0	0	1	1
B. PRACTICAL									
1	ENGG	Major	CS191	Programming for Problem Solving Lab	0	0	3	3	1.5
2	HUM	Ability Enhancement Course	HU191	Professional Communication Lab	0	0	2	2	1
3	SCI	Skill enhancement course	CH(CS)191	Engineering Chemistry Lab	0	0	2	2	1
4	ENGG	Skill enhancement course	ME(CS)191	Workshop & Manufacturing Practices Lab	0	0	3	3	1.5
Total of Theory & Practical								23	18

**HUM: Humanities; ENGG: Engineering; SCI: Science; PRJ: Project;*

1st Year 2nd Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A. THEORY									
1	ENGG	Major	CS201	Data Structures	3	0	0	3	3
2	ENGG	Minor	EE(CS)201	Basic Electrical & Electronics Engineering	3	0	0	3	3
3	SCI	Multidisciplinary	PH(CS)201	Engineering Physics	3	0	0	3	3
4	SCI	Multidisciplinary	M(CS)201	Engineering Mathematics –II	3	0	0	3	3
5	HUM	Value added course	HU204	Environmental Science	2	0	0	2	2
6	HUM	Value added course	HU205	Indian Knowledge System	1	0	0	1	1
B. PRACTICAL									
1	ENGG	Major	CS291	Data Structures Lab	0	0	3	3	1.5
2	ENGG	Minor	EE(CS)291	Basic Electrical & Electronics Engineering Lab	0	0	3	3	1.5
3	HUM	Ability Enhancement Course	HU292	Design Thinking	0	0	2	2	1
4	SCI	Skill enhancement course	PH(CS)291	Engineering Physics Lab	0	0	3	3	1.5
5	ENGG	Skill enhancement course	ME(CS)291	Engineering Graphics & Design Lab	0	0	3	3	1.5
Total of Theory & Practical								29	22

2nd Year 3rd Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A. THEORY									
1	ENGG	Major	CS301	Computer Organization and Architecture	3	0	0	3	3
2	ENGG	Major	CS302	Design and Analysis of Algorithms	3	1	0	4	4
3	SCI	Minor	M(CS)301	Discrete Mathematics	3	0	0	3	3
4	ENGG	Minor	EC(CS)301	Digital Logic and Electronics	3	0	0	3	3
B. PRACTICAL									
1	ENGG	Major	CS391	Computer Organization and Architecture Lab	0	0	3	3	1.5
2	ENGG	Major	CS392	Design and Analysis of Algorithms Lab	0	0	3	3	1.5
3	ENGG	Minor	EC(CS)391	Digital Electronics Lab	0	0	3	3	1.5
4	ENGG	Skill Enhancement Course	CS393	IT Workshop Lab (SciLab/MATLAB/C++)	0	1	3	4	2.5
Total of Theory, Practical								26	20

2nd Year 4th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A. THEORY									
1	ENGG	Major	CS401	Operating Systems	3	0	0	3	3
2	ENGG	Major	CS402	Computer Networks	3	0	0	3	3
3	ENG	Major	CS403	Formal Language and Automata Theory	3	0	0	3	3
4	SCI	Minor	M(CS)401	Probability and Statistics	3	0	0	3	3
5	HUM	Ability Enhancement Course	HU(CS)401	Principles of Management	2	0	0	2	2
B. PRACTICAL									
1	ENGG	Major	CS491	Operating Systems Lab	0	0	3	3	1.5
2	ENGG	Major	CS492	Computer Networks Lab	0	0	3	3	1.5
3	ENGG	Major	CS493	Programming using Python	0	0	3	3	1.5
4	ENGG	Minor	M(CS)491	Numerical Methods Lab	0	0	3	3	1.5
5	HUM	Ability Enhancement Course	HU(CS)491	Soft Skill & Aptitude	2	0	0	2	1
Total of Theory, Practical								28	21

3rd Year 5th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A. THEORY									
1	ENGG	Major	CS 501	Artificial Intelligence	3	0	0	3	3
2	ENGG	Major	CS 502	Database Management Systems	3	0	0	3	3
3	ENGG	Major	CS 503	Object Oriented Programming using Java	3	0	0	3	3
4	ENGG	Major	CS 504 A CS 504 B CS 504 C	Compiler Design Cryptography and Network Security Computer Graphics	3	0	0	3	3
5	HUM	Minor	HU(CS)501	Economics for Engineers	2	0	0	2	2
B. PRACTICAL									
6	ENGG	Major	CS 591	Artificial Intelligence Lab	0	0	3	3	1.5
7	ENGG	Major	CS 592	Database Management Systems Lab	0	0	3	3	1.5
8	ENGG	Major	CS 593	Object Oriented Programming using Java Lab	0	0	3	3	1.5
9	PRJ	Internship	CS581	Internship	0	0	2	2	2
Total of Theory, Practical and Mandatory Activities/Courses								25	20.5

3rd Year 6th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A. THEORY									
1	ENGG	Major	CS 601	Web and Internet Technology	3	0	0	3	3
2	ENGG	Major	CS 602	Machine Learning	3	1	0	4	4
3	ENGG	Major	CS 603	Software Engineering	3	0	0	3	3
4	ENGG	Major	CS 604 A CS 604 B CS 604 C	Mobile Computing Natural Language Processing Cloud Computing	3	0	0	3	3
5	ENGG	Minor	CS 605	Cyber Law and Ethics	3	0	0	3	3
B. PRACTICAL									
6	ENGG	Major	CS 691	Web and Internet Technology Lab	0	0	3	3	1.5
7	ENGG	Major	CS 692	Machine Learning Lab	0	0	3	3	1.5
8	ENGG	Major	CS 693	Software Engineering Lab	0	0	3	3	1.5
Total of Theory, Practical and Mandatory Activities/Courses								25	20.5

4th Year 7th Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A. THEORY									
1	ENGG	Major	CS 701	Neural Networks and Deep Learning	3	0	0	3	3
2	ENGG	Major	CS 702 A CS 702 B CS 702 C	Advanced Algorithms Advanced Computer Architecture Advanced Operating Systems	3	0	0	3	3
3	ENGG	Minor	CS 703 A CS 703 B CS 703 C	Information Theory and Coding Ad-Hoc and Sensor Networks Data Mining and Data Warehouse	3	0	0	3	3
4	HUM	Minor	HU(CS)701	Human Resource Development and Organizational Behavior	2	0	0	2	2
B. PRACTICAL									
5	ENGG	Major	CS 791	Neural Networks and Deep Learning Lab	0	0	3	3	1.5
6	ENGG	Major	CS 792 A CS 792 B CS 792 C	Advanced Algorithms Lab Advanced Computer Architecture Lab Advanced Operating Systems Lab	0	0	3	3	1.5
7	PRJ	Project	CS 793	Major Project-I	0	0	12	12	6
Total of Theory, Practical and Mandatory Activities/Courses								29	20

4th Year 8th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A. THEORY									
1	ENGG	Major	CS 801 A CS 801 B CS 801 C	Real Time Systems Data Analytics Soft Computing	3	1	0	4	4
2	ENGG	Major	CS 802 A CS 802 B CS 802 C	VLSI Design & Application Bio-informatics Robotics	3	1	0	4	4
3	ENGG	Minor	CS 803 A CS 803 B CS 803 C	Introduction to IoT Image Processing Optimization Techniques	3	0	0	3	3
B. PRACTICAL									
4	ENGG	Minor	CS 893 A CS 893 B CS 893 C	Internet of Things Lab Image Processing Lab Optimization Techniques Lab	0	0	3	3	1.5
5	PRJ	Project	CS 881	Major Project-II	0	0	12	12	6
6	PRJ	Internship	CS882	Grand Viva	0	0	0	0	1.5
Total of Theory, Practical and Mandatory Activities/Courses								26	20

Curriculum & Syllabus for B. Tech under Autonomy (NEP 2020 Implemented)

Computer Science and Engineering

(Effective from 2023-24 admission batch)

Department: Computer Science & Engineering

Curriculum Structure & Syllabus

(Effective from 2023-24 admission batch)

1st Year 1st Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A. THEORY									
1	ENGG	Major	CS101	Programming for Problem Solving	3	0	0	3	3
2	SCI	Multidisciplinary	CH(CS)101	Engineering Chemistry	2	0	0	2	2
3	SCI	Multidisciplinary	M(CS)101	Engineering Mathematics – I	3	0	0	3	3
4	HUM	Ability Enhancement Course	HU101	Professional Communication	2	0	0	2	2
5	HUM	Value added course	HU102	Values and Ethics	2	0	0	2	2
6	HUM	Value added course	HU103	Constitution of India	1	0	0	1	1
B. PRACTICAL									
1	ENGG	Major	CS191	Programming for Problem Solving Lab	0	0	3	3	1.5
2	HUM	Ability Enhancement Course	HU191	Professional Communication Lab	0	0	2	2	1
3	SCI	Skill enhancement course	CH(CS)191	Engineering Chemistry Lab	0	0	2	2	1
4	ENGG	Skill enhancement course	ME(CS)191	Workshop & Manufacturing Practices Lab	0	0	3	3	1.5
Total of Theory & Practical								23	18

**HUM: Humanities; ENGG: Engineering; SCI: Science; PRJ: Project;*

Course Name: Programming for problem solving
Course Code: CS101
Contact (Periods/Week):3L/Week
Total Contact Hours: 36
Credits: 3

Course Outcome(s):

CO1: To identify the working principle of input and output devices of Computers memorize the basic terminology used in computer programming.

CO2: To express programs in C language and use different data types for writing the programs.

CO3: To implement programs using the dynamic behaviour of memory by the use of pointers.

CO4: To explain the difference between call by value and call by address.

CO5: To write programs using basic data files and developing applications for real world problems.

CO-PO-PSO Mapping:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO1 12	PSO 1	PSO 2	PSO 3
CO1	3	3	3	2	2						2	3	2	3	3
CO2	2	2	3	3	3							3	2	2	3
CO3	2	3	2	2	2							3	2	3	2
CO4	3	2	2	3	3							2	2	2	2
CO5	2	2	2	1	1						2	3	3	3	3

Course Content:

Module-1: Fundamentals of Computer (9L)

History of Computer, Generation of Computer, Classification of Computers, Basic structure of Computer System, Primary & Secondary Memory, Processing Unit, Input & Output devices.

Number System: basic of Binary, Octal, Decimal and Hexadecimal number systems; Representation and interchanging of number in different number systems. Introduction to complements system, Representation of signed and unsigned numbers in singed magnitude singed 1's complement system and signed 2's complement system.

Arithmetic– Addition and Subtraction (using1's complement and 2's complement). Representation of Characters-ASCII Code, Basics of Compiler, Interpreter and Assembler

Problem solving – Basic concept of Algorithm. Representation of algorithm using flowchart and pseudo code, Some basic examples.

Module-2: Introduction to C Programming (5L)

Overview of Procedural vs Structural language; History of C Programming Language. Variable and Data Types: The C characters identifiers and keywords, data type & sizes, variable names, declaration, statements. Operators & Expressions: Arithmetic operators, relational operators, Logical operators, increment and decrement operators, bitwise operators, Assignment operators, conditional operators, special operators-type Conversion, C expressions, precedence and associativity. Input and Output: Standard input and output, formatted output–printf, formatted input scanf.

Module-3: **Branch and Loop (5L)**

Branching: Concept of Statement and Blocks in C, Simple if, if -else, nested if-else and if-else ladder. Switch Case: break and continue; switch-case, concept of go to and labels Loops- while, for, do while.

Module-4: Program Structures (4L)

Function: Basics of Functions, function types, function prototypes, formal and actual parameter, function calling, functions returning values, functions not returning values. Recursion and Recursive Function.
Storage Class in C: Storage Class-auto, external, static and registers to rage class, scope rules and lifetime of variables
C pre-processor: Pre-processing directive and macro, parameterized macro.

Module-5: **Array and Pointer (7L)**

Arrays: One dimensional array, Two-dimensional arrays Passing an array to a function Pointers: Pointers, Pointer and Array, Pointer and functions.
Strings: Character array and string, array of strings, Passing a string to a function, String related functions, Pointer and String. Dynamic memory allocation: Malloc, calloc, realloc and free with example.

Module-6: Structures, Unions and Enum (3L)

Basic of structures, arrays of structures, structures and pointers, bit fields. Basics of union and enum, difference between structure and union.

Module-7: File in C (3L)

Files handling-opening and closing a file indifferent mode, formatted and unformatted files, Command line arguments, fopen, fclose, fgetc, fputc, fprintf, fscanf function.

Textbook:

1. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill
2. Kanetkar Y.- Let us C, BPB Publication, 15th Edition

Reference Books:

1. Brian W.Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India
2. K R Venugopal & S R Prasad– MASTERINGC, TMH, 2nd Edition

Course Name: Engineering Chemistry
Paper Code: CH(CS)101
Total Contact Hours: 24
Credit: 2
Prerequisites:

COURSE OBJECTIVE

To understand the basic principles of elements, organic reactions, drug synthesis and technological aspects of modern chemistry

To apply the knowledge of different engineering materials, advanced polymers, and nanomaterials to solve complex engineering problems

To analyse and evaluate quality parameters of water and its treatment

Apply the knowledge of free energy, energy storage device, semiconductors, fuels and corrosion to design environment friendly & sustainable devices

Apply the knowledge of different instrumental techniques to analyse unknown engineering materials.

COURSE OUTCOME

CO1. Able to understand the basic principles of elements, organic reactions drug synthesis and computational chemistry

CO2. Able to apply the knowledge of different engineering materials, advanced polymers, and nanomaterials to solve complex engineering problems

CO3. Able to analyse and evaluate water quality parameters and its treatment

CO4. Able to the knowledge of free energy, energy storage device, fuels and corrosion to design environment friendly & sustainable devices

CO5. Able to apply the knowledge of different instrumental techniques to analyse unknown engineering materials

CO v/s PO MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
1	3	3	2	2	2	-	-	-	-	-	2	2
2	3	3	3	3	-	-	2	-	-	-	2	2
3	3	3	-	-	-	-	3	-	-	-	3	2
4	3	3	3	2	-	-	3	-	-	-	3	2
5	3	3	3	3	2	-	-	-	-	-	2	2

COURSE CONTENT

Module 1 - Elements and their properties (6L)

1. **Elements and their properties (3L)**
   Bohr's theory for one electron system, Hydrogen spectrum, Quantum numbers, atomic orbitals, Pauli's exclusion principle, Hund's rule, exchange energy, Aufbau principle, Electronic configuration and Magnetic properties.
2. **Periodic Table for Engineers (3L)**
   Modern Periodic table, Periodic properties, study of advanced functional materials like Silicones, Silicates, Zeolite and alloys like steel, mischmetall, Neodymium alloy and their applications.

Module 2 - Energy devices and Semiconductors (6L)

1. **Use of free energy in chemical equilibria (3L)**
   Laws of Thermodynamics, Enthalpy, Entropy, Spontaneity, Electrochemical Cell, Dry Cell, Mercury Cell, Lead Storage batteries, Ni-Cd Cells, Fuel Cells, Solar Cells, Nernst equation and applications, Electrochemical sensors
2. **Crystals and Semiconductors (3L)**
   Crystals and their defects, Stoichiometric and Non-stoichiometric defects, Band theory and Doping, n- type and p-type semiconductors, Superconductors

Module 3 –Industrial Applications of Chemistry (8L)

1. **Advanced Polymeric materials (3L)**
   Classification, Engineering Plastics, conducting polymers, bio polymers, polymer composites
2. **Industrial corrosion (2L)**
   Classification, Effects of corrosion, Preventive measures
3. **Analysis of Water Quality (1L)**
   Physicochemical and Biological parameters
4. **Nano materials (1L)**
   Synthesis of Nano materials, Applications in modern devices
5. **Basic Computational Chemistry (1L)**
   Introduction of computational chemistry and their applications

Module 4 – Organic Reaction Products and their spectroscopic analysis (4L)

1. **Organic Reactions (2L)**
   Substitution, Elimination and Addition reactions
2. **Drug designing and synthesis (1L)**
   Paracetamol, Aspirin
3. **Spectroscopic Analysis (1L)** UV – Visible Spectra, IR spectra

Course Name: Engineering Mathematics - I
Paper Code: M(CS)101
Contact (L: T: P): 3:0:0
Total Contact Hours: 36
Credit: 3

Prerequisites:

The students to whom this course will be offered must have the concept of (10+2) standard matrix algebra, and calculus.

Course Objectives:

The objective of this course is to familiarize the prospective engineers with techniques in matrix algebra and calculus. It aims to equip the students with standard concepts and tools at an intermediate to advanced level that will serve them well towards tackling more advanced level of mathematics and applications that they would find useful in their disciplines.

Course Outcomes (COs):

On successful completion of the learning sessions of the course, the learner will be able to:

CO1: Recall the properties related to matrix algebra and calculus.

CO2: Determine the solutions of the problems related to matrix algebra and calculus.

CO3: Apply the appropriate mathematical tools of matrix algebra and calculus for the solutions of the problems.

CO4: Analyze different engineering problems linked with matrix algebra and calculus.

CO-PO/PSO Mapping:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	3	2	-	-	-	-	-	-	-	-	-	2
CO2	3	2	-	-	-	-	-	-	-	-	-	1
CO3	3	2	2	-	-	-	-	-	-	-	-	1
CO4	2	3	1	2	-	-	-	-	-	-	-	1
CO	**2.75**	**2.25**	**1.5**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1.25**

Weightage Values: Strongly mapped: ‘3’, Moderately mapped: ‘2’, Weakly mapped: ‘1’, Not mapped: ‘-‘

Course Content:

Module I: Liner Algebra (11L)
Echelon form and normal (canonical) form of a matrix; Inverse and rank of a matrix; Consistency and inconsistency of system of linear equations, Solution of system of linear equations; Eigen values and eigenvectors; Diagonalization of matrix, Cayley-Hamilton theorem.

Module II: Single Variable Calculus (5L)
Rolle's Theorem, Mean value theorems, Taylor's and Maclaurin theorems with remainders; Concept of sequence and series, Power series; Taylor's series.

Module III: Multivariable Calculus (Differentiation) (13L)
Function of several variables; Concept of limit, continuity and differentiability; Partial derivatives, Total derivative and its application; chain rules, Derivatives of implicit functions Euler's theorem on homogeneous function; Jacobian; Maxima and minima of functions of two variables.

Module IV: Multivariable Calculus (Integration) (7L)
Double Integral, Triple Integral; Change of order in multiple integrals; Line Integral, Surface Integral, Volume Integral. Change of variables in multiple integrals.

Text Books:

1. Grewal, B.S., Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.
2. Kreyszig, E., Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.

Reference Books:

1. Guruprasad, S. A text book of Engineering Mathematics-I, New age International Publishers.
2. Ramana, B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.
3. Veerarajan, T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.

Course Name: Professional Communication Paper
Code: HU101
Contact: 2:0:0
Total Contact Hours: 24
Credit: 2

Pre-requisites:	Basic (10+2) level of knowledge of English grammar, vocabulary reading and writing skills.
Course Objectives	The course aims to impart domain and industry-specific communications kill sin a globalized context and to promote the understanding of business communication ractices and cross-cultural dynamics.
Course Outcomes:	By pursuing this course the students shall be able to
	1. Define, describe and classify the modalities and nuances of communication in a workplace context.
	2. Review, appraise and understand the modes, contexts and appropriacy of communicating across cultures and societies.
	3. Identify, interpret and demonstrate the basic formats, templates of business and official communication.
	4. Identify, compare and illustrate reading strategies and basic writing strategies.
	5.Interpret, analyze and evaluate semantic-structural, interpersonal and multicultural dynamics in business communication.

Course Content:

Module1:
Verbal and Nonverbal communication **4L**
Definition, Relevance and Effective Usage
Components of Verbal Communication: Written and Oral Communication
Components of Non-verbal Communication: Kinesics, Proxemics, Chronemics, Haptics Paralanguage
Barriers to Effective Communication

Module2:
Workplace Communication Essentials and Cross-Cultural Communication **4L**
Communication at the Workplace—Formal and Informal Situations
Language in Use—Jargon, Speech Acts/Language Functions, Syntactical and Grammatical Appropriacy
Cultural Contexts in Global Business: High Context and Low Context Cultures Understanding Cultural

Nuances and Stereotyping Achieving Culturally Neutral Communication in Speech and Writing

Module3: **4L**

Reading Strategies and Basic Writing Skills Reading: Purposes and Nature of Reading
Reading Sub-Skills—Skimming, Scanning, Intensive Reading
Reading General and Business Texts (Reading for Comprehension and Detailed Understanding) Basic Writing Skills—Paragraph and Essay writing, writing technical documents
Writing Technicalities—Paragraphing, Sentence Structure and Punctuation

Module4: **4L**

Report Writing

Nature and Function of Reports Types of Reports
Researching for a Business Report Format, Language and Style Report Documentation

Module5:

Employment Communication

a. Writing Business Letters—(Enquiry, Order, Sales, Complaint, Adjustment, Job Application, Offer) **2L**

b. Creating an Employee Profile—Preparing a CV or Résumé.
Creating a Digital/Online Profile–LinkedIn (Résumé/Video Profile) **2L**

c. Writing Other Interoffice Correspondence--E-mails: types, convention, and etiquette, Memo, Notices and Circulars **2L**

d. Preparing Meeting Documentation—Drafting Notice and Agenda of Meetings, Preparing Minutes of Meetings. **2L**

References:-

1. Meenakshi Raman and Sangeetha Sharma. Technical Communication. 3rd edition. New Delhi: Oxford University Press, 2015.
2. Mark Ibbotson. Cambridge English for Engineering. Cambridge: Cambridge University Press, 2008.
3. Mark Ibbotson. Professional English in Use: Engineering. Cambridge: Cambridge UP, 2009.
4. Lesikar et al. Business Communication: Connecting in a Digital World. New Delhi: Tata McGraw-Hill,2014.
5. John Seeley. Writing Reports. Oxford: Oxford University Press, 2002.
6. Judith Leigh. CVs and Job Applications. Oxford: Oxford University Press, 2002.
7. Judith Leigh. Organizing and Participating in Meetings. Oxford: Oxford University Press, 2002.
8. Michael Swan. Practical English Usage. Oxford: OUP, 1980.
9. Pickett, Laster and Staples. Technical English: Writing, Reading & Speaking. 8 thed. London:

Longman, 2001.
10. Diana Booher. E-writing: 21st Century Tools for Effective Communication.

Links:-

1. Purdue University's Online Writing Lab (OWL)-https://owl.purdue.edu/
2. Business English Pod-https://www.businessenglishpod.com/

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	-	-	-	-	-	2	1	1	2	3	-	2
CO2	-	-	-	-	-	1	1	2	2	3	-	3
CO3	-	-	-	-	-	3	3	1	1	3	2	3
CO4	-	-	-	-	-	3	3	1	-	3	-	3
CO5						2	2	2	2	3	-	3

Course Name: Values and Ethics Course
Code: HU102
Contacts: 2:0:0
Total Contact Hours: 24 Credit: 2
Prerequisite: None

Module: 1 Introduction:(4L)
Definition of Ethics; Approaches to Ethics: Psychological, Philosophical, and Social
Types of values-Social, Psychological, Aesthetic, Spiritual, and Organizational
Natural acceptance of human values. Definitiveness of Ethical Human Conduct. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order.

Module 2: Universal Human Harmony. (4L)
Basic Human Aspirations, Happiness and Prosperity, Self-Exploration, Self and the Body
Understanding the harmony in the Nature.
Interconnectedness and mutual fulfilment among the four orders of nature- recyclability and self-regulation in nature.
Values Crisis in contemporary society Nature of values: Value Spectrum of a good life (Maslow's Pyramid)

Module: 3 Ethical Concerns: (6L)
Renewable Energy Resources, Environmental degradation and pollution. Eco-friendly Technologies. Environmental Regulations, Environmental Ethics. Rapid Technological growth and depletion of resources, Reports of the Club of Rome.

Problems of Technology transfer- Technology assessment impact analysis -Human Centered Technology.

Module: 4 Ethics of Profession: **(4L)**
Work Ethics and Work Values, Business Ethics, Human values in organizations: Social and ethical responsibilities of Technologists. Codes of professional ethics.

Types of Ethical issues-Internal Ethics of Business–
Whistle Blowing, Impact of Ethics on Business Policies and Strategies– Ethical Leadership – Characteristics
Module: 5 Self Development AND Gender Awareness (6L)
Definition of Gender, Basic Gender Concepts and Terminology, Exploring Attitudes towards Gender, Social Construction of Gender
Gender Roles and Relations, Types of Gender Roles, Gender Roles and Relationships Matrix, Gender-based Division and Valuation of Labour. Gender Development Issues, Identifying Gender Issues

Text Books:
1. Beneria, Lourdes. (2004). Gender, Development, and Globalization: Economics as if All People Mattered. Roultedge Press. (GDGE)
2. Molyneux and Razavi. (2002). Gender Justice, Development and Rights. Oxford University Press (GJDR or WGD)
3. Visvanathan, Duggan,Wiegersma and Nisonoff. (2011).
4. The Women, Gender and Development Reader. 2nd Edition. Zed Press (WGD)
5. Stephen H Unger, Controlling Technology: Ethics and the Responsible Engineers, John Wiley & Sons, New York 1994 (2nd Ed)
6. Deborah Johnson, Ethical Issues in Engineering, Prentice Hall, Englewood Cliffs, New Jersey 1991.

7. A N Tripathi, Human values in the Engineering Profession, Monograph published by IIM, Calcutta 1996.

Course Outcomes:

CO 1	Understand the significance of values, various approaches to ethics and its applications in life and profession.
CO2	Able to distinguish Self and the Body, to understand Harmony in the Self
CO3	To identify and eradicate environmental concerns through technology
CO4	Demonstrate work ethics and analyse business strategies
CO5	Ability to understand gender terminologies and to identify gender issues

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	**3**	**-**	**2**	**-**	**-**	**-**	**-**	**-**	**3**	**3**	**-**	**2**
CO2	**-**	**-**	**-**	**-**	**-**	**3**	**3**	**-**	**3**	**2**	**-**	**-**
CO3	**-**	**3**	**3**	**-**	**3**	**2**	**3**	**-**	**-**	**-**	**-**	**2**
CO4	**2**	**-**	**-**	**2**	**-**	**-**	**-**	**3**	**-**	**-**	**2**	**-**
CO5	**-**	**3**	**-**	**-**	**-**	**2**	**1**	**-**	**-**	**2**	**-**	**2**

Paper Name: Constitution of India Paper
Code: HU103
Credit: 01
No. of lectures: 12

Module 1: History of Making of the Indian Constitution: History. Drafting Committee, (Composition & Working) **3L**
Philosophy of the Indian Constitution: Preamble Salient Features
Module 2: Fundamental Rights, Fundamental Duties, Directive Principles of State Policy **6L**

The Right to Equality
The Right to Freedom: I (Article 19)
The Right to Freedom: II (Articles 20, 21 and 22) The
Right against Exploitation
The Right to freedom of Religion
Cultural and Educational rights The
Right to Property
The Right to Constitutional Remedies
Fundamental Duties

Module-3: Organs of Governance: **3L**
Parliament - Composition - Qualifications and Disqualifications -Powers and Functions – Executive- President - Governor - Council of Ministers - Judiciary, Appointment and Transfer of Judges, Qualifications - Powers and Functions

Text / Reference Books:

1) Indian Constitution by D. D. Basu, The Publisher, LexisNexis
2) Constitution of India by Subhas C Kasyap, Vitasta Publishing
3) The Constitution of India, P.M Bakshi, Universal Law Publishing Co.Ltd, New Delhi, 2003.
4) Indian Constitution Text Book - Avasthi, Avasthi,Publisher: LAKSHMI NARAIN AGARWAL
5) Introduction to the Constitution of India, Brij Kishore Sharma, PHI

Course Name: Programming for problem solving Lab Course
Code: CS191
Contact Hours: 3L/Week Total
Contact Hours: 36 Credits: 1.5

Course Outcomes	Name of Course Outcomes
CO1	To identify the working of different operating systems like DOS, Windows, Linux
CO2	To express programs in C language
CO3	To implement programs connecting decision structures, loops
CO4	To experiment with user defined functions to solve real time problems
CO5	To write C programs using Pointers to access arrays, strings, functions, structures and files

CO-PO-PSO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	3	3	3	2	2						2	3	2	3	3
CO2	2	2	3	3	3							3	2	2	3
CO3	2	3	2	2	2							3	2	3	2
CO4	3	2	2	3	3							2	2	2	2
CO5	2	2	2	1	1						2	3	3	3	3

Course Content:

Module- 1: Familiarization with some basic commands of DOS and Linux. File handling and Directory structures, file permissions, creating and editing simple C program in different editor and IDE, compilation and execution of C program. Introduction to Code block.

Module-2: Problem based on

a) Basic data types
b) Different arithmetic operators.
c) Printf() and scanf() functions.

Module-3: Problem based on conditional statements using

a) if-else statements
b) different relational operators
c) different logical operators

Module-4: Problem based on

a) **for** loop
b) **while** loop
c) **do-while** loop

Module-5: Problem based on

a) How to write a menu driven program using switch-case statement
b) How to write a function and passing values to a function
c) How to write a recursive function.

Module-6: Problem based on

a) How to use array (bothI-Dand2-D).
b) How to pass an array to a function.

Module-7: Problem based on manipulation of strings in different way.

Module-8: Problem based on

a) How to handle compound variable sin C
b) How to handle file in C
c) How to use command line argument in C

Textbook:

1. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill

2. Kanetkar Y.-Let us C, BPB Publication,15th Edition

Reference Books:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India
K R Venugopal & S R Prasad–MASTERINGC,TMH, 2nd Edition

Paper Name: Professional Communication Lab Paper
Code: HU191
Contact: (0:0:2)
Total Contact Hours: 26 Credit: 1

Pre requisites: Basic knowledge of LSRW skills.

Course Objectives: To train the students in acquiring interpersonal communication skills by focusing on language skill acquisition techniques and error feedback.

Course Outcome:
By pursuing this course the students will be able to:
CO1: Recognize, identify and express advanced skills of Technical Communication in English through Language Laboratory.
CO2: Understand, categorize, differentiate and infer listening, speaking, reading and writing skills in societal and professional life.
CO3: Articulate and present the skills necessary to be a competent Interpersonal communicator. CO4: Deconstruct, appraise and critique communication behaviours.
CO5: Adapt, negotiate and facilitate with multifarious socio-economical and professional arenas with effective communication and interpersonal skills.

Course Contents:
Module 1: Introduction to the Language Lab
a. The Need for a Language Laboratory
b. Tasks in the Lab
c. Writing a Laboratory Note Book

Module 2: Active Listening
a. What is Active Listening?
b. Listening Sub-Skills—Predicting, Clarifying, Inferencing, Evaluating, Note-taking
c. Listening in Business Telephony

Module 3: Speaking
a. Speaking—Accuracy and Fluency Parameters
b. Pronunciation Guide—Basics of Sound Scripting, Stress and Intonation
c. Fluency-focused activities—JAM, Conversational Role Plays, Speaking using Picture/Audio Visual inputs
d. Accuracy-focused activities—Identifying Minimal Pairs, Sound Mazes, Open and Closed Pair Drilling, Student Recordings (using software)
e. Group Discussion: Principles and Practice
f. Giving a Presentation—Learning Presentation Basics and Giving Micro Presentations

Module 4: Lab Project Work
a. Writing a Book Review
b. Writing a Film Review
c. Scripting a Short Presentation (2 minutes)
d. Making a short video CV (1-2 minutes)

References:

1. IT Mumbai**, Preparatory Course in English** syllabus
2. IIT Mumbai, **Introduction to Linguistics** syllabus
3. Sasikumar et al. *A Course in Listening and Speaking*. New Delhi: Foundation Books, 2005.
4. Tony Lynch, *Study Listening*. Cambridge: Cambridge UP, 2004.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	2	-	-	1	1	-	2	3	-	2
CO2	-	-	2	2	-	3	3	-	2	3	-	3
CO3	-	-	2	2	-	3	3	2	2	3	-	3
CO4	-	-	-	-	-	3	3	2	2	3	-	3
CO5	-	-	2	2	-	3	3	2	2	3	-	3

Course Name: Engineering chemistry lab Paper
Code: CH(CS)191
Total Contact Hours: 24
Credit: 1

Course Objective

- Study the basic principles of pH meter and conductivity meter for different applications
- Analysis of water for its various parameters & its significance in industries
- Learn to synthesis Polymeric materials and drugs
- Study the various reactions in homogeneous and heterogeneous medium

Course Outcome

CH191.1: Able to operate different types of instruments for estimation of small quantities chemicals used in industries and scientific and technical fields.
CH191.2: Able to analyse and determine the composition and physical property of liquid and solid samples when working as an individual and also as a team member
CH191.3: Able to analyse different parameters of water considering environmental issues
CH191.4: Able to synthesize drug and sustainable polymer materials.
CH191.5: Capable to design innovative experiments applying the fundamentals of modern chemistry

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
1	3	1	3	1	-	2	3	-	-	-	-	1
2	2	2	1	1	-	1	-	-	-	1	-	1
3	-	-	-	-	-	-	-	-	3	3	2	2
4	2	1	2	2	-	-	1	-	-	-	-	2
5	3	3	3	3	1	1	1	1	-	-	2	2

COURSE CONTENT

1. Determination of the concentration of the electrolyte through conductance measurement.
2. Determination of water quality measurement techniques.
3. Determination of the concentration of the electrolyte through pH measurement.
4. Estimation of Cu in brass
5. Estimation of Fe_2O_3 in Cement
6. Isolation of graphene from dead dry batteries and their use for temporary soldering.

7. Synthesis of Silver Nanoparticles doped organic thin film for organic transistors.
8. Estimation of corrosion in a given sample metal.
9. Preparation of Si-nano crystals for future memory devices.
10. Green Synthesis of ZnO based Polymer Nano composites.
11. Synthesis of polymers for electrical devices and PCBs.
12. Determination of Partition Coefficient of acetic acid between two immiscible liquids.
13. Drug design and synthesis
14. Rheological properties of the Newtonian fluids
15. Innovative Experiments

Course Name: Workshop and Manufacturing
Practices Lab Course Code: ME(CS)191
Contact: 0:0:3
Credits: 1.5

Prerequisite: Physics & Mathematics (10+2 Level)

CO1: Gain basic knowledge of Workshop Practice and Safety useful for our daily living.
CO2: Understand the use of Instruments of a pattern shop like Hand Saw, Jack Plain, Chisels etc.
CO3: Apply and performing operations like such as Marking, Cutting etc used in manufacturing processes.
CO4: Analyze the various operations in the Fitting Shop using Hack Saw, various files, Scriber, etc to understand the concept of tolerances applicable in all kind of manufacturing.
CO5: Get hands on practice of in Welding and apply various machining processes which give a lot of confidence to manufacture physical prototype sin project works.

Course Content:

3P

(i) Theoretical discussions:

1. Manufacturing Methods- casting, forming, machining, joining, advanced manufacturing methods
2. Fitting operations & power tools
3. Carpentry
4. Welding (arc welding & gas welding), brazing
5. Electrical & Electronics
6. Metal casting
7. CNC machining, Additive manufacturing, 3D Printing
8. Plastic moulding & Glass Cutting

(ii) Workshop Practice:

At least 6 modules should be covered

Module 1 - Machine shop 6P

Typical jobs that may be made in this practice module:

i. To make a pin from a mild steel rod in a lathe.

ii. To make rectangular and vee slot in a block of cast iron or mild steel in a shaping and / ormilling machine.

Module 2 - Fitting shop 6P

Typical jobs that may be made in this practice module: To make a Gauge from MS plate.

Module 3 – Carpentry Shop 6P

Typical jobs that may be made in this practice module: To make wooden joints and/or a pattern or like.

Module 4 - Welding & Soldering shop 6P

Typical jobs that may be made in this practice module:

i. Arc Welding: To join two thick (approx 5mm) MS plates by manual metal arc welding.
ii. Gas Welding: To join two thin mild steel plates or sheets by gas welding.
iii. House wiring, soft Soldering

Module 5 – Smithy & Casting 6P

Typical jobs that may be made in this practice module:

i. A simple job of making a square rod from a round bar or similar.
ii. One/ two green sand moduls to prepare, and a casting be demonstrated.

Module 6 – CNC Machining & Laser Cutting 6P

Typical jobs that may be made in this practice module:

i. At least one sample shape on mild steel plate should be made using CNC Milling / CNC Lathe Machine
ii. At least one sample shape on glass should be made using laser cutting machine.

Module 7 – 3D Printing 6P

i) Exposure to a 3D printing machine,
ii) 3D printing of at least one sample model using available materials.

Examinations could involve the actual fabrication of simple components, utilizing one or more ofthe techniques covered above.

Text Books:

1. Hajra Choudhury S.K., Hajra Choudhury A.K. and NirjharRoy S.K., –Elements of Workshop Technology‖, Vol. I 2008 and Vol. II 2010, Media promoters and publishers private limited, Mumbai.
2. Rao P.N., –Manufacturing Technology‖, Vol. I and Vol. II, Tata McGrawHill House, 2017.

Reference Books:

1. Gowri P., Hariharan and A. Suresh Babu, ManufacturingTechnology – I, Pearson Education, 2008.
2. Roy A. Lindberg, –Processes and Materials of Manufacture‖, 4th edition, Prentice Hall India, 1998.
3. Kalpakjian S. and Steven S. Schmid, ManufacturingEngineering and Technology, 4th edition, Pearson Education India Edition, 2002.
4. Manufacturing Science by A. Ghosh and A.K. Mallick,Wiley Eastern.
5. Principles of Metal Cutting/Principles of Machine Tools by G.C. Sen and A. Bhattacharya, New Central Book Agency, Kolkata.

CO-PO/PSO Mapping:

CO Codes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3						2		2	2					
CO2	3						2		2	2					
CO3	3						2		2	2			2		2
CO4	3						2		2	2			2		2
CO5	3	2	2				2		2	2					

1st Year 2nd Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A. THEORY									
1	ENGG	Major	CS201	Data Structures	3	0	0	3	3
2	ENGG	Minor	EE(CS)201	Basic Electrical & Electronics Engineering	3	0	0	3	3
3	SCI	Multidisciplinary	PH(CS)201	Engineering Physics	3	0	0	3	3
4	SCI	Multidisciplinary	M(CS)201	Engineering Mathematics –II	3	0	0	3	3
5	HUM	Value added course	HU204	Environmental Science	2	0	0	2	2
6	HUM	Value added course	HU205	Indian Knowledge System	1	0	0	1	1
B. PRACTICAL									
1	ENGG	Major	CS291	Data Structures Lab	0	0	3	3	1.5
2	ENGG	Minor	EE(CS)291	Basic Electrical & Electronics Engineering Lab	0	0	3	3	1.5
3	HUM	Ability Enhancement Course	HU292	Design Thinking	0	0	2	2	1
4	SCI	Skill enhancement course	PH(CS)291	Engineering Physics Lab	0	0	3	3	1.5
5	ENGG	Skill enhancement course	ME(CS)291	Engineering Graphics & Design Lab	0	0	3	3	1.5
Total of Theory & Practical								29	22

Course Name: Data Structures
Course Code: CS201
Contact (Periods/Week):=3L/Week Total
Contact Hours: 36
Credits: 3

Course Objectives:

1. To learn the basics of abstract data types.
2. To learn the principles of linear and nonlinear data structures.
3. To build an application using sorting and searching.

Course Outcomes	Name of Course Outcomes
CO1	To identify how the choices of data structure & algorithm methods impact the performance of program.
CO2	To express problems based upon different data structure for writing programs.
CO3	To implement programs using appropriate data structure & algorithmic methods for solving problems.
CO4	To explain the computational efficiency of the principal algorithms for sorting, searching, and hashing.
CO5	To write programs using dynamic and static data structures and building applications for real world problems.

CO-PO-PSO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO 1	3	2		2	3						1	3	1	1	1
CO 2	3	2	2	2	2							2	3	2	2
CO 3	2	3	3	2	3						1	2	3	3	3
CO 4	2	2	2	3	1							1	2	1	2
CO 5	2	3	3	3	2						1	2	3	3	3
	2.40	**2.40**	**2.50**	**2.40**	**2.20**							**2.00**	**2.40**	**2.00**	**2.20**

Course Content:

Module 1: Introduction [4L]

Concepts of data and information; Concept of Abstract Data Type, Data Structure and Data Type. Classification of Data Structures- Primitive and Non-Primitive Data Structure, Linear and Non-Linear Data Structure. Need of Data Structures. (1L)

Concept of algorithms and programs, Different methods of representing algorithm; Algorithm analysis, time and space analysis of algorithms – Asymptotic notations like Big Oh (O), Small Oh(o), Big Omega(Ω), Small Omega(ω) and Theta(Θ) notation (definition and significance). (3L)

Module 2: Non-Restricted Linear Data Structure[9L]

List or Linear List: Definition and Example, List as ADT. Representation of Linear List- Sequential Representation and Linked Representation.

Array: Introduction to sequential representation, Linearization of multidimensional array. Application of array-representation of polynomial using array, Representation of Sparse matrix using array.

Linked List: Introduction to linked representation, Implementation of different types of linked list- Singly linked list, Doubly linked list, Circular linked list, Circular Doubly Linked List. Application of Linked list-Representation of polynomial.

Module 3: Restricted Linear Data Structure [6L]

Stack: Definition of Stack, implementations of stack using array and linked list, Applications of stack- infix to postfix conversion, Postfix Evaluation

Recursion: Principles of recursion - use of stack, tail recursion. Tower of Hanoi using recursion. Queue: Definition of Queue; Implementation of queue using array-physical, linear and circular model; Implementation of queue using linked list.

Dequeue - Definition and different types of dequeue.

Module 4: Nonlinear Data structures [9L]

Trees and Binary Tree:

Basic terminologies; Definition of tree and binary tree. Difference between tree and binary tree, Representation of binary tree (using array and linked list)

Binary tree traversal (pre-, in-, post- order); Threaded binary tree- definition, insertion and deletion algorithm; Binary search tree- Definition, insertion, deletion, searching algorithm;

Height balanced binary tree: AVL tree- definition, insertion and deletion with examples only.

m –Way Search Tree: B Tree – Definition, insertion and deletion with examples only; B+ Tree – Definition, insertion and deletion with examples only.

Heap: Definition (min heap and max heap), creation, insertion and deletion algorithm. Application of heap (priority queue and sorting).

Graphs: Definition and representation (adjacency matrix, incidence matrix and adjacency list).

Graph traversal– Depth-first search (DFS), Breadth-first search (BFS) - concepts of edges used in DFS and BFS (tree-edge, back-edge, cross-edge, and forward-edge).

Module 5: Sorting and Searching [8L]

Sorting Algorithms: Definition and need of sorting, different types of sorting algorithm (internal, external, stable, in-place, comparison based); Factors affecting sorting Methods, Bubble sort, Insertion sort, Selection sort, Quick sort, Merge sort, Radix sort – algorithm with analysis (time complexity) Searching: Factors affecting searching Methods; Sequential search –algorithm with analysis (time complexity); improvement using sentinel.

Binary search and Interpolation Search algorithm with analysis (time complexity)

Hashing: Introduction and purpose of Hashing and Hash functions (division, folding and mid-square), Collision resolution techniques.

Text book:

1. Data Structures Through 'C' Language by Samiran Chattopadhyay, Debabrata Ghosh Dastidar, Matangini Chattopadhyay, Edition: 2001, BPB Publications

2. Fundamentals of Data Structures of C by Ellis Horowitz, Sartaj Sahni, Susan Anderson-freed 2nd Edition, Universities Press

Reference Books:

1. Data Structures, Algorithms, and Software Principles in C by Thomas A. Standish, 1 Edition, Pearson.
2. Data Structures by S. Lipschutz, Special Indian Edition, Tata McGraw Hill Education (India) Private Limited
3. Data Structures and Program Design in C by Robert L. Kruse, Bruce P. Leung 2nd Edition, Pearson
4. Data Structures in C by Aaron M. Tenenbaum, 1st Edition, Pearson.

Course Name: Basic Electrical & Electronics Engineering Course
Code: EE(CS)201
Contact: 3:0:0
Total Contact Hours: 36 Credit: 3

Prerequisite: Basic 12th standard Physics and Mathematics, Concept of components of electric circuit.

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

CO	Statement
CO1	Apply fundamental concepts and circuit laws to solve simple DC electric circuits
CO2	To solve simple ac circuits in steady state
CO3	Impart the knowledge of Basic Electronics Devices and ICs.
CO4	Analyze the simple electronics circuits

MODULE 1: Elementary Concepts of Electric Circuits **6L**

DC Circuits: Circuit Components: Conductor, Resistor, Inductor, Capacitor – Ohm's Law - Kirchhoff's Laws –Independent and Dependent Sources – Simple problems- Nodal Analysis, Mesh analysis with independent sources only (Steady state)
Introduction to AC Circuits and Parameters: Waveforms, Average value, RMS Value, Instantaneous power, real power, reactive power and apparent power, power factor – Steady state analysis of RLC circuits (Simple problems only)

MODULE 2: Electrical machine **8L**

Transformer: Magnetic materials, BH characteristics, ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency.
DC Machines: Brief idea on constructional features, classifications, working principle of both motor and generator. Simple problems on Voltage equation.

MODULE 3: Fundamentals of Semiconductor Devices: **6L**

Introduction to Semiconductor: Concept of energy band diagram; Comparison among metal, insulator, semiconductor; Semiconductors-classifications and Fermi energy level; Charge neutrality and Mass-Action law in semiconductor; Current flow in semiconductor due to drift & diffusion process; Einstein relation.

MODULE 4: PN Junction Diode: **4L**

Principle of operation; V-I characteristics; principle of avalanche & Zener breakdown; Junction resistances and capacitances; V-I characteristics of Zener diode.

MODULE 5: Bipolar Junction Transistors: **4L**

PNP and NPN structures; Principle of operation; Current gains in CE, CB and CC mode; input and output characteristics; Biasing & Stability Analysis-Concept of Fixed Bias, Collector to base Bias & voltage divider bias.

MODULE 6: Introduction to IC: 8L

Integrated circuit-Basic idea, classifications, advantages, disadvantages; OPAMP(IC741)-Pin configuration and equivalent circuit; Characteristics of OPAMP(IC741); Inverting & Non- Inverting Amplifier; Adder, Subtractor, Differentiator & Integrator Circuit.

Textbooks:

1. A Textbook of Electrical Technology - Volume I (Basic Electrical Engineering) & Volume II (Ac & DC Machines)-B. L Theraja & A.K. Teraja, S. Chad,23rd Edition, 1959
2. D. Chattopadhyay, P.C Rakshit, "Electronics Fundamentals and Applications", New Age International (P) Limited Publishers, Seneth Edition,2006
3. Basic Electrical & Electronics Engineering by J.B. Gupta, S.K. Kataria & Sons,2013
4. Basic Electrical and Electronics Engineering-I by Abhijit Chakrabarti and Sudip Debnath, McGraw Hill, 2015
5. M.S. Sukhija and T.K. Nagsarkar, Basic Electrical and Electronics Engineering, Oxford University Press,2012.
6. DP Kothari and IJ Nagrath, "Basic Electrical & Electronics Engineering", Tata McGraw Hill,2020.

Reference Books

1. D C Kulshreshtha, "Basic Electrical Engineering", TataMcGrawHill,2010.
2. T.K. Nagsarkar, M.S. Sukhija, "Basic Electrical Engineering", Oxford Higher Education.
3. Hughes, "Electrical and Electronic Technology", Pearson Education".
4. Parker and Smith, "Problems in Electrical Engineering", CBS Publishers and Distributors.
5. Anant Agarwal, Jeffrey Lang, Foundations of Analog and Digital Electronic Circuits, Morgan Kaufmann Publishers, 2005.
6. Bernard Grob, Basic Electronics, McGrawHill.
7. Chinmoy Saha, Arindham Halder and Debarati Ganguly, Basic Electronics-Principles and Applications, Cambridge University Press, 2018.

CO-PO Course Articulation Matrix Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	3	3	2	1	-	-	-	-	-	-	2	2
CO2	3	3	2	1	-	-	-	-	-	-	2	2
CO3	3	2	2	1	-	-	-	-	-	-	1	2
CO4	2	3	2	1	-	-	-	-	-	-	2	1

Course Name: Engineering Physics Course
Code: PH(CS)201
Contact: (3:0:0)
Total Contact Hours: 36
Credits: 3
Prerequisites: Knowledge of Physics up to 12th standard.

Course Objectives:
The aim of courses in Physic-I is to provide adequate exposure and develop insight about the basic principles of physical sciences and its practical aspects which would help engineers to learn underlying principles of various tools and techniques they use in core engineering and related industrial applications. The course would also inculcate innovative mindsets of the students and can create awareness of the vital role played by science and engineering in the development of new technologies.

Course Outcomes (COs):
After attending the course students' should be able to

CO	Description
CO1	explain basic principles of laser and optical fibers.
CO2	understand the properties of Nano material.
CO3	analyze different crystallographic structures according to their co-ordination number and packing factors.
CO4	analyze the structure, function and characteristics of different storage devices.
CO5	justify the need of a quantum mechanics as remedy to overcome limitations imposed by classical physics.

CO-PO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	--	--	--	--	--	--	--	2
CO2	3	3	2	2	--	--	--	--	--	--	--	2
CO3	3	3	2	2	--	--	--	--	--	--	--	1
CO4	3	2	2	2	--	--	--	--	--	--	--	2
CO5	3	3	3	2	2	--	--	--	--	--	--	1

Course Content:

Module 1 (12L) Modern Optics

1.01- Laser: Concepts of various emission and absorption processes, Einstein A and B coefficients and equations, working principle of laser, meta stable state, population inversion, condition necessary for active laser action, optical resonator, illustrations of Ruby laser, He-Ne laser, Semiconductor laser, applications of laser, related numerical problems. 6L

1.02-Fibre optics-Principle and propagation of light in optical fibers (Step index, Graded index, single and multiple modes) - Numerical aperture and Acceptance angle, Basic concept of losses in optical fiber, related numerical problems. 3L

1.03-Holography-Theory of holography, viewing of holography, applications 3L

Module 2 (6L) Solid State Physics

2.01 Crystal Structure: Structure of solids, amorphous and crystalline solids (definition and examples), lattice, basis, unit cell, Fundamental types of lattices –Bravais lattice, simple cubic, fcc and bcc lattices, Miller indices and miller planes, co-ordination number and atomic packing factor, Bragg's equation, applications, numerical problems. 3L

2.02 Semiconductor: Physics of semiconductors, electrons and holes, metal, insulator and semiconductor, intrinsic and extrinsic semiconductor, p-n junction. 3L

Module 3 (8L) Quantum Mechanics

3.01 Quantum Theory: Inadequacy of classical physics-concept of quantization of energy, particle concept of electromagnetic wave (example: photoelectric and Compton Effect; no derivation required, origin of modified and unmodified lines), wave particle duality; phase velocity and group velocity; de Broglie hypothesis; Davisson and Germer experiment, related numerical problems. 4L

3.02 Quantum Mechanics 1: Concept of wave function, physical significance of wave function, probability interpretation; normalization of wave functions-Qualitative discussion; uncertainty principle, relevant numerical problems, Introduction of Schrödinger wave equation (only statement). 4L

Module 4 (4L)

Physics of Nano materials

Reduction of dimensionality, properties of nanomaterials, Quantum wells (two dimensional), Quantum wires (one dimensional), Quantum dots (zero dimensional); Quantum size effect and Quantum confinement. Carbon allotropes. Application of nanomaterials (CNT, graphene, electronic, environment, medical).

Module 5 (6L)

Storage and display devices

Different storage and display devices-Magnetic storage materials, Hard disc (examples related to computers compared with semiconductor storage viz. Pendrive), Operation and application of CRT, CRO, Liquid crystal display (LCD), LED, OLED, Plasma display, Thin film transistor display).

Recommended Text Books for Physics I:

Text Books:

1. Refresher courses in physics (Vol. 1, Vol. 2 & Vol. 3)-C. L. Arora (S. Chand Publishers)
2. Basic Engineering Physics-Amal Chakraborty (Chaya Prakashani Pvt. Ltd.)
3. Perspective & Concept of Modern Physics -Arthur Baiser
4. Principles of engineering physics – Md. N Khan and S Panigrahi.

Course Name: Engineering Mathematics - II Paper
Code: M(CS)201
Contact (L: T: P): 3:0:0
Total Contact Hours: 36
Credit: 3

Prerequisites:

The students to whom this course will be offered must have the concept of (10+2) standard calculus.

Course Objectives:

The objective of this course is to familiarize the prospective engineers with techniques in ordinary differential equations, Laplace transform and numerical methods. It aims to equip the students with standard concepts and tools at an intermediate to advanced level that will serve them well towards tackling more advanced level of mathematics and applications that they would find useful in their disciplines.

Course Outcomes (COs):

On successful completion of the learning sessions of the course, the learner will be able to:

CO1: Recall the properties related to ordinary differential equations, Laplace transform and numerical techniques.

CO2: Determine the solutions of the problems related to ordinary differential equations, Laplace transform and numerical techniques.

CO3: Apply appropriate mathematical tools of ordinary differential equations, Laplace transform and numerical techniques for the solutions of the problems.

CO4: Analyze engineering problems by using ordinary differential equation, Laplace transform and numerical Methods.

CO-PO/PSO Mapping:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	3	2	-	-	-	-	-	-	-	-	-	2
CO2	3	2	-	-	-	-	-	-	-	-	-	1
CO3	3	2	2	-	-	-	-	-	-	-	-	1
CO4	2	3	1	2	-	-	-	-	-	-	-	1
M(CS) 201	**2.75**	**2.25**	**1.5**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1.25**

Weightage Values: Strongly mapped: ‘3’, Moderately mapped: ‘2’, Weakly mapped: ‘1’, Not mapped: ’-‘.

Course Content:

Module I: First Order Ordinary Differential Equations (ODE) (9L)

Solution of first order and first degree ODE: Exact ODE, Rules for finding Integrating factors, Linear ODE, Bernoulli's equation.

Solution of first order and higher degree ODE: solvable forp, solvable for y and solvable for x and Clairaut's equation.

Module II: Second Order Ordinary Differential Equations (ODE) (8L)

Solution of second order ODE with constant coefficients: C.F. &P.I., Method of variation of parameters, Cauchy-Euler equations.

Module III: Laplace Transform (LT) (12L)

Concept of improper integrals; Definition and existence of LT, LT of elementary functions, First and second shifting properties, Change of scale property, LT of tf(t), LT of f(t)t, LT of derivatives of f(t), LT of integral of f(t), Evaluation of improper integrals using LT, LT of periodic and step functions, Inverse LT: Definition and its properties, Convolution theorem (statement only) and its application to the evaluation of inverse LT, Solution of linear ODE with constant coefficients (initial value problem) using LT.

Module IV: Numerical Methods (7L)

Introduction to error analysis, Calculus of finite difference. **Interpolation:** Newton forward and backward interpolation, Lagrange's interpolation. **Numerical integration:** Trapezoidal rule, Simpson's 1/3 rule. **Numerical solution of ordinary differential equation:** Euler method, Fourth order Runge - Kutta method.

Text Books:

1. Grewal, B.S., Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.
2. Kreyszig, E., Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.

Reference Books:

1. Guruprasad, S. A text book of Engineering Mathematics-I, New age International Publishers.
2. Ramana, B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.
3. Veerarajan, T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.
4. Bali, N.P. and Goyal, M., A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
5. Thomas, G.B. and Finney, R.L., Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
6. Apostol, M., Calculus, Volumes 1 and 2 (2nd Edition), Wiley Eastern, 1980.
7. Kumaresan, S., Linear Algebra - A Geometric approach, Prentice Hall of India, 2000.
8. Poole, D., Linear Algebra: A Modern Introduction, 2nd Edition, Brooks/Cole, 2005.
9. Bronson, R., Schaum's Outline of Matrix Operations. 1988.
10. Piskunov, N., Differential and Integral Calculus, Vol. I & Vol. II, Mir Publishers, 1969.

Course Name: Environmental Science Paper
Code: HU204
Contact (L: T: P): 2 : 0 : 0 Total
Contact Hours: 24 Credit: 3

Course Objective(s)

This course will enable the students to,

- Realize the importance of environment and its resources.
- Apply the fundamental knowledge of science and engineering to assess environmental and health risk.
- Know about environmental laws and regulations to develop guidelines and procedures for health and safety issues.
- Solve scientific problem-solving related to air, water, land and noise pollution.

Course Outcome

CO	Statement
CO1	Able to understand the natural environment and its relationships with human activities
CO2	The ability to apply the fundamental knowledge of science and engineering to assess environmental and health risk
CO3	Ability to understand environmental laws and regulations to develop guidelines and procedures for health and safety issues
CO4	Acquire skills for scientific problem-solving related to air, water, noise & land pollution.

CO – PO Mapping

CO	Statement	PO1	PO 2	PO 3	PO 4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
1	Able to understand the natural environment and its relationships with human activities	2	2	3	-	-	2	3	3	-	-	1	2
2	The ability to apply the fundamental knowledge of science and engineering to assess environmental and health risk	3	3	3	1	1	2	3	3	-	-	1	2

3	Ability to understand environmental laws and regulations to develop guidelines and procedures for health and safety issues	3	3	3	2	1	2	3	3	-	-	1	2
4	Acquire skills for scientific problem-solving related to air, water, noise & land pollution.	1	1	1	1	2	2	3	3	-	-	1	2
AVERAGE		2	2	2	1	1	2	3	3	-	-	1	2

Module 1 - Resources and Ecosystem (6L)

1. **Resources (2L)**
   Types of resources, resistance to resources, Human resource, Population Growth models: Exponential Growth, logistic growth
2. **Ecosystem (3L)**
   Components of ecosystem, types of ecosystem, Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems, Food chain, Food web.
3. **Energy and Environment(1L)**
   Conventional energy sources, coal and petroleum, Green energy sources, solar energy, tidal energy, geothermal energy, biomass

Module 2 – Environmental Degradation (9L)

1. **Air Pollution and its impact on Environment (3L)**
   Air Pollutants, primary & secondary pollutants, Criteria pollutants, Smog, Photochemical smog and London smog, Greenhouse effect, Global Warming, Acid rain, Ozone Layer Depletion.
2. **Water Pollution and its impact on Environment (3L)**
   Water Pollutants, Oxygen demanding wastes, heavy metals, BOD, COD, Eutrophication, Hardness, Alkalinity, TDS and Chloride, Heavy metal poisoning and toxicity.
3. **Land Pollution and its impact on Environment (2L)**
   Solid wastes, types of Solid Waste, Municipal Solid wastes, hazardous wastes, bio-medical wastes, E-wastes
4. **Noise Pollution and its impact on Environment (1L)**
   Types of noise, Noise frequency, Noise pressure, Noise intensity, Noise Threshold limit, Effect of noise pollution on human health.

Module 3 – Environmental Management (6L)

1. **Environmental Impact Assessment (1L)**
   Objectives of Environmental management, Components of Environmental Management, Environmental Auditing, Environmental laws and Protection Acts of India
2. **Pollution Control and Treatment (2L)**
   Air Pollution controlling devices, Catalytic Converter, Electrostatic Precipitator, etc., Waste Water Treatment, Noise pollution control.

3. **Waste Management (3L)**
   Solid waste management, Open dumping, Land filling, incineration, composting, E-waste management, Biomedical Waste management.

Module 4 – Disaster Management (3L)

1. **Study of some important disasters (2L)**
   Natural and Man-made disasters, earthquakes, floods drought, landside, cyclones, volcanic eruptions, tsunami, Global climate change. Terrorism, gas and radiations leaks, toxic waste disposal, oil spills, forest fires.
2. **Disaster management Techniques (1L)**
   Basic principles of disasters management, Disaster Management cycle, Disaster management policy, Awareness generation program

Text Books:

1. Basic Environmental Engineering and Elementary Biology (For MAKAUT), Gourkrishna Dasmohapatra, Vikas Publishing.
2. Basic Environmental Engineering and Elementary Biology, Dr. Monindra Nath Patra & Rahul Kumar Singha, Aryan Publishing House.
3. Textbook of Environmental Studies for Undergraduate Courses, Erach Barucha for UGC, Universities Press

Reference Books:

1. A Text Book of Environmental Studies, Dr. D.K. Asthana & Dr. Meera Asthana, S.Chand Publications.
2. Environmental Science (As per NEP 2020), Subrat Roy, Khanna Publisher

Paper Name: Indian knowledge System Paper
Code: HU205
Credit: 01
No. of lectures: 12

Module-1 (3L)

An overview of Indian Knowledge System (IKS): Importance of Ancient Knowledge - Definition of IKS - Classification framework of IKS - Unique aspects of IKS.
The Vedic corpus: Vedas and Vedangas - Distinctive features of Vedic life. Indian philosophical systems: Different schools of philosophy.

Module-2 (3L)

Salient features of the Indian numeral system - Importance of decimal representation - The discovery of zero and its importance - Unique approaches to represent numbers.

Highlights of Indian Astronomy: Historical development of astronomy in India

Module-3 (3L)

Indian science and technology heritage - Metals and metalworking - Mining and ore extraction – Physical structures in India - Irrigation and water management - Dyes and painting technology - Surgical Techniques - Shipbuilding

Module-4 (3L)

Traditional Knowledge in Different Sectors: Traditional knowledge and engineering, Traditional medicine system, Traditional Knowledge in agriculture, Traditional societies depend on it for their food and healthcare needs.

References:

1. Introduction to Indian knowledge system: concepts and applications-Mahadevan B.Bhat, Vinayak Rajat, Nagendra Pavana R.N.,PHI
2. Traditional Knowledge system in India, Amit Jha, Atlantic Publishers
3. S. N. Sen and K. S. Shukla, History of Astronomy in India, Indian National Science Academy, 2nd edition, New Delhi, 2000

Course Name: Data Structures Lab Course
Code: CS291
Contact (Periods/Week): 3L/Week Total
Contact Hours: 36
Credits: 1.5

Course Outcomes	Name of Course Outcomes
CO1	To identify the appropriate data structure as applied to specified problem definition.
CO2	To summarize operations like searching, insertion, deletion, traversing mechanism used on various data structures.
CO3	To implement practical knowledge of data structures on the applications.
CO4	To illustrate how to store, manipulate and arrange data in an efficient manner.
CO5	To write programs to access queue and stack using arrays and linked list, binary tree and binary search tree.

CO-PO-PSO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	3	3	3	2	3	-	-	-	-	-	-	2	1	1	1
CO2	3	2	2	3	3	-	-	-	-	-	3	2	3	2	2
CO3	2	3	3	-	2	-	-	-	-	-	-	2	3	3	3
CO4	2	2	1	3	2	-	-	-	-	-	2	3	2	1	2
CO5	2	2	3	1	2	-	-	-	-	-	-	3	3	3	3
	2.4	2.4	2.4	2.3	2.4	-	-	-	-	-	2.5	2.4	2.4	2	2.2

Course Content:

Module 1: Implementing Non-Restricted Linear Data Structure [2 Lab]
Problem based on Implementation of Non-Restricted Linear Data Structure like- Implementation of list as data structure using array. Implementation of list as data structure using linked list of different types. Implementation of polynomial as data structure using array and linked list. Implementation of sparse matrix as data stricture using array.

Module 2: Implementing Restricted Linear Data Structure [3 Lab] Problem based on Implementation of Restricted Linear Data Structure like- Implementation of stack as data structure using array. Implementation of stack as data structure using linked list. Implementation of queue as data structure using array (physic al, linear and circular model). Implementation of queue as data structure using linked list. Converting infix to post-fix and evaluating post-fix expression using stack. Implementing Tower-of-Hanoi problem.

Module 3: Implementing Non-Linear Data Structure [2 Lab]
Problem based on Implementation of Non-Linear Data Structure like Implementation of Binary Tree as data structure using array and linked list. Implementation of Binary Search Tree (BST) as data structure using linked list. Implementation of Heap as data structure using array. Implementation of Priority Queue as data structure using Heap.

Module 4: Implementing Sorting and Searching algorithm [5 Lab] Problem based on Implementation of Sorting and Searching algorithm
Implementation of Bubble sort using appropriate data structure. Implementation of Selection sort using appropriate data structure.
Implementation of Insertion sort using appropriate data structure. Implementation of Quick sort using appropriate data structure.
Implementation of Merge sort using appropriate data structure. Implementation of Heap sort using appropriate data structure. Implementation of Radix sort using appropriate data structure. Implementation of Sequential Search using appropriate data structure. Implementation of Binary Search using appropriate data structure. Implementation of hashing with collision resolution using linear and quadratic probing.

Text books:

1. Data Structures Through 'C' Language by Samiran Chattopadhyay, Debabrata Ghosh Dastidar, Matangini Chattopadhyay, Edition: 2001, BPB Publications.
2. Fundamentals of Data Structures of C by Ellis Horowitz, Sartaj Sahni, Susan Anderson- freed 2nd Edition, Universities Press.

Reference books:

1. Data Structures, Algorithms, and Software Principles in C by Thomas A. Standish, 1 Edition, Pearson.
2. Data Structures by S. Lipschutz, Special Indian Edition, Tata McGraw Hill Education (India) Private.
3. Limited Data Structures and Program Design In C by Robert L. Kruse, Bruce P. Leung 2nd Edition, Pearson.
4. Data Structures in C by Aaron M. Tenenbaum, 1St Edition, Pearson

Course Name: Basic Electrical & Electronics Engineering Lab Course
Code: EE(CS)291
L-T-P: 0-0-3
Total Lecture: 36
Credit: 1.5

CO	Statement
CO1	To Analyze a given network by applying KVL and KCL.
CO2	To Examine the Operation of DC Motor.
CO3	To Examine the Operation of Basic Electronics Devices and ICs.
CO4	To design simple electronics circuits.

List of Experiments: -

1. Familiarization with different passive and active electrical & electronic components.
2. Familiarization with different Electrical & Electronics Instruments.
3. Verification of KVL and KCL.
4. Forward and reversal of DC shunt motor.
5. Speed control of DC shunt motor.
6. Study of the P-N junction diode V-I characteristics (Forward & Reverse Bias).
7. Study of the Characteristics of Zener diode (Forward & Reverse Bias).
8. Study of the Input and Output characteristics of BJT in CE mode.
9. Determination of offset voltage, offset current & bias current of OPAMP (IC741).
10. Determination of CMRR and slew rate of OPAMP(IC741).
11. Determination of inverting and non-inverting gain of OPAMP(IC741).
12. Extramural Experiment.

Textbooks:

1. Handbook of Laboratory Experiments in Electronics Engineering Vol. 1, Author Name: A.M. Zungeru, J.M. Chuma, H.U. Ezea, and M. Mangwala, Publisher -Notion Press Electronic Devices and Circuit Theory by Robert Boylestad Louis Nashelsky,7th Edition, Prentice Hall
2. Experiments Manual for use with Grob's Basic Electronics 12th Edition by Wes Ponick, Publisher- McGraw Hill,2015
3. Laboratory Manual for 'Fundamentals of Electrical & Electronics Engineering': A handbook for Electrical & Electronics Engineering Students by Manoj Patil (Author), Jyoti Kharade (Author), 2020
4. The Art of Electronics, Paul Horowitz, Winfield Hill,Cambridge University Press, 2015.
5. A Handbook of Circuit Math for Technical Engineers, Robert L. Libbey CRC Press, 05- Jun-1991

Reference Books

1. Basic Electrical and Electronics Engineering, Author: S. K. Bhattacharya, Publisher: Pearson Education India,2011
2. Practical Electrical Engineering
3. By Sergey N. Makarov, Reinhold Ludwig, Stephen J. Bitar, Publisher: Springer International Publishing, 2016
4. Electronics Lab Manual (Volume 2) By Navas, K. A. Publisher: PHI Learning Pvt. Ltd. 2018
5. Practical Electronics Handbook, Ian R. Sinclair and John Dunton, Sixth edition 2007, Published by Elsevier Ltd.

CO-PO Course Articulation Matrix Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO 7	PO8	PO9	PO 10	PO 11	PO 12
CO1	3	2	3	2	-	2	-	-	2	-	2	3
CO2	3	3	2	3	-	2	-	-	3	-	2	2
CO3	3	2	2	3	-	2	-	-	2	-	3	3
CO4	3	3	2	2	-	2	-	-	3	-	2	3

Course Name: Engineering Physics
Course Code: PH(CS)291
Contact Hours: 0:0:3
Credit: 1.5
Prerequisites: Knowledge of Physics up to 12th standard.

Course Objectives:
The aim of course is to provide adequate exposure and develop insight about the basic principles of physical sciences and its practical aspects which would help engineers to learn underlying principles of various tools and techniques they use in core engineering and related industrial applications. The course would also inculcate innovative mindsets of the students and can create awareness of the vital role played by science and engineering in the development of new technologies.

Course Outcomes (COs):
After attending the course students' will be able to
CO1 : demonstrate experiments allied to their theoretical concepts
CO2 : conduct experiments using LASER, Optical fiber.
CO3 : participate as an individual, and as a member or leader in groups in laboratory sessions actively
CO4 : analyze experimental data from graphical representations, and to communicate effectively them in Laboratory reports including innovative experiment.
CO5: Design solutions for real life challenges.

CO-PO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	3	2	3	3	--	--	--	--	--	--	--
CO2	2	3	2	3	3	--	--	--	--	--	--	--
CO3	2	3	2	3	3	--	--	--	--	--	--	--
CO4	2	2	3	2	3	--	--	--	--	--	--	--
CO5	2	2	3	2	3	--	--	--	--	--	--	--

Course Content:

General idea about Measurements and Errors (One Mandatory):
Error estimation using Slide calipers/ Screw-gauge/travelling microscope for one experiment.
Experiments on Classical Physics (Any 4 to be performed from the following experiments):
1. Study of Torsional oscillation of Torsional pendulum & determination of time using various load of the oscillator.
2. Determination of Young's moduli of different materials.
3. Determination of Rigidity moduli of different materials.
4. Determination of wavelength of light by Newton's ring method.
5. Determination of wavelength of light by Laser diffraction method.
6. Optical Fibre-numerical aperture, power loss.

Experiments on Quantum Physics (Any 2 to be performed from the following experiments):
7. Determination of Planck's constant using photoelectric cell.
8. Verification of Bohr's atomic orbital theory through Frank-Hertz experiment.
9. Determination of Stefan's Constant.
10. Study of characteristics of solar cell.

Perform at least one of the following experiments:

11. Calibration of an oscillator using Lissajous Figure.
12. Determination of specific charge of an electron (e/m) by J. J Thompson Method.

**In addition it is recommended that each student should carry out at least one experiment beyond the syllabus/one experiment as Innovative experiment.

Probable experiments beyond the syllabus:

1. Study of dispersive power of material of a prism.
2. Study of viscosity using Poiseuille's capillary flow method/using Stoke's law.
3. Determination of thermal conductivity of a bad/good conductor using Lees-Charlton / Searle apparatus.
4. Determination of the angle of optical rotation of a polar solution using polarimeter.
5. Any other experiment related to the theory.

Text Books:

1. Practical Physics by Chatterjee & Rakshit (Book & Allied Publisher)
2. Practical Physics by K.G. Mazumder (New Central Publishing)
3. Practical Physics by R. K. Kar (Book & Allied Publisher)

Paper Name: Design thinking Paper
Code: HU292
Credit: 01
L:T:P:: 0:0:2

Module 1: Understanding the Learning Process, Kolb's Learning Styles, Assessing and Interpreting. **2**

Module 2: Memory: process, Sensory memory, STM and LTM, Problems in retention, Memory enhancement techniques. **4**

Module 3: Emotions: Experience & Expression Understanding Emotions, Empathy, And Concept of Emotional Intelligence. **2**

Module 4: Basics of Design Thinking Definition of Design Thinking, Need for Design Thinking, Objective of Design Thinking, Concepts & Brainstorming, Stages of Design Thinking Process (explain with examples) – Empathize, Define, Ideate, Prototype, Test. **6**

Module 5: Understanding Creative thinking process, Understanding Problem Solving, Testing Creative Problem Solving **4**

Unit 6: Prototyping & Testing -Rapid Prototype Development process, Testing, Sample Example, Test Group Marketing **2**

Module-7: Design thinking for strategic innovations Growth –Change- Examples of Customer Challenges, Use of Design Thinking to Enhance Customer Experience -Value redefinition - Extreme Competition – Standardization – –Strategy– Business Model design. **4**

References:

1. John.R.Karsnitz, Stephen O'Brien and John P. Hutchinson, "Engineering Design", Cengage learning (International edition) second Edition, 2013.

2. Roger Martin, "The Design of Business: Why Design Thinking is the Next Competitive Advantage", Harvard Business Press, 2009.

3. Idris Mootee, "Design Thinking for Strategic Innovation: What They Can't Teach You at Business or Design School", John Wiley & Sons 2013

4. George, E, Dieter, Linda, C, Schmidt. (2017). Engineering Design, McGraw Hill publisher, 4th edition

Course Name: Engineering Graphics & Design Lab Course
Code: ME(CS)291
Contact: 0:0:3
Credits: 1.5

Prerequisites: Basic knowledge of geometry

Course Outcomes: Upon successful completion of this course, the student will be able to: CO1: Learn the basics of drafting

CO2: Understand the use of drafting tools which develops the fundamental skills of industrial drawings.
CO3: Apply the concept of engineering scales, dimensioning and various geometric curves necessary to understand design of machine elements.
CO4: Analyse the concept of projection of line, surface and solids to create the knowledge base of orthographic and isometric view of structures and machine parts.
CO5: Evaluate the design model to different sections of industries as well as for research & development.

Course Contents:

Basic Engineering Graphics: **(3P)**

Principles of Engineering Graphics; Orthographic Projection; Descriptive Geometry; Drawing Principles; Isometric Projection; Surface Development; Perspective; Reading a Drawing; Sectional Views; Dimensioning & Tolerances; True Length, Angle; intersection, Shortest Distance.

Module 1: Introduction to Engineering Drawing (6P)

Principles of Engineering Graphics and their significance, Usage of Drawing instruments, lettering, Conic sections including Rectangular Hyperbola (General method only); Cycloid, Epicycloid and Involute; Scales – Plain, Diagonal and Vernier Scales.

Module 2: Orthographic & Isometric Projections (6P)

Principles of Orthographic Projections-Conventions - Projections of Points and lines inclined to both planes; Projections of planes on inclined Planes - Auxiliary Planes; Projection of Solids inclined to both the Planes- Auxiliary Views; Isometric Scale, Isometric Views of lines, Planes, Simple and compound Solids; Conversion of Isometric Views to Orthographic Views and Vice- versa.

Module 3: Sections and Sectional Views of Right Angular Solids (6P)

Drawing sectional views of solids for Prism, Cylinder, Pyramid, Cone and project the true shape of the sectioned surface, Auxiliary Views; Development of surfaces of Right Regular Solids - Prism, Pyramid, Cylinder and Cone; Draw sectional orthographic views of objects from industry and dwellings (foundation to slab only).

Computer Graphics: (**3P)**

Engineering Graphics Software; -Spatial Transformations; Orthographic Projections; Model Viewing; Co-ordinate Systems; Multi-view Projection; Exploded Assembly; Model Viewing; Animation; Spatial Manipulation; Surface Modeling; Solid Modeling.

Module 4: Overview of Computer Graphics: (3P)

Demonstration of CAD software [The Menu System, Toolbars (Standard, Properties, Draw, Modify and Dimension), Drawing Area (Background, Crosshairs, Coordinate System), Dialog boxes and windows, Shortcut menus (Button Bars), Zooming methods, Select and erase objects].

Module 5: CAD Drawing, Customization, Annotations, layering (6P)

Set up of drawing page including scale settings, ISO and ANSI standards for dimensioning and tolerance; Using various methods to draw straight lines, circles, applying dimensions and annotations to drawings; Setting up and use of Layers, changing line lengths (extend/lengthen); Drawing sectional views of solids; Drawing annotation, CAD modeling of parts and assemblies with animation, Parametric and nonparametric solid, surface and wireframe modeling, Part editing and printing documents.

Module 6: Demonstration of a simple team design project (3P)

Illustrating Geometry and topology of engineered components: creation of engineering models and their presentation in standard 2D blueprint form and as 3D wire-frame and shaded solids; Meshed topologies for engineering analysis and tool-path generation for component manufacture, use of solid-modeling software for creating associative models at the component and assembly levels.

Text Books:

1. Bhatt N.D., Panchal V.M. & Ingle P.R, (2014), Engineering Drawing, Charotar Publishing House
2. K. Venugopal, Engineering Drawing + AutoCAD, New Age International publishers

Reference Books:

1. Pradeep Jain, Ankita Maheswari, A.P. Gautam, Engineering Graphics & Design, Khanna Publishing House
2. Agrawal B. & Agrawal C. M. (2012), Engineering Graphics, TMH Publication.
3. Shah, M.B. & Rana B.C. (2008), Engineering Drawing and Computer Graphics, Pearson Education
4. Narayana, K.L. & P Kannaiah (2008), Text book on Engineering Drawing, Scitech Publishers.

CO-PO/PSO Mapping:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	2			2									2	2	2
CO2	2			2									2	2	2
CO3	3			2									2	2	2
CO4	3			3									3	3	2
CO5	3	2		3	2								3	3	2

2nd Year 3rd Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A. THEORY									
1	ENGG	Major	CS301	mputer Organization and Architecture	3	0	0	3	3
2	ENGG	Major	CS302	gn and Analysis of Algorithms	3	1	0	4	4
3	SCI	Minor	M(CS)301	Discrete Mathematics	3	0	0	3	3
4	ENGG	Minor	EC(CS)301	igital Logic and Electronics	3	0	0	3	3
B. PRACTICAL									
1	ENGG	Major	CS391	omputer Organization and Architecture Lab	0	0	3	3	1.5
2	ENGG	Major	CS392	ign and Analysis of Algorithms Lab	0	0	3	3	1.5
3	ENGG	Minor	EC(CS)391	Digital Electronics Lab	0	0	3	3	1.5
4	ENGG	Skill Enhancement Course	CS393	IT Workshop Lab (SciLab/MATLAB/C++)	0	1	3	4	2.5
Total of Theory, Practical								26	20

Course Name: Computer Organization and Architecture
Course Code:CS301
Contact: 3:0:0
Total Contact Hours: 36L
Credits: 3

Prerequisite: Digital Electronics
Course Outcomes (COs):
After attending the course students should be able to

CO1	Illustrate the basic concept of computer architecture and its performance measurement, parallel processing, Flynn's classification and Amdahl's law and apply this knowledge in designing solutions for real life engineering problems.
CO2	Summarize the basic concept of pipeline, instruction pipeline, arithmetic pipeline hazards detection and prevention and use this knowledge for designing and implementing mathematical and engineering problems leading to lifelong learning.
CO3	Identify the concept of Instruction-Level Parallelism to solve engineering problems.
CO4	Illustrate and compare the concept of Multiprocessor architecture and parallel architecture and apply this knowledge for developing an approach by means of existing and new methods as a team work.
CO5	Understand the concept of message passing architecture and interconnection network and design an optimized model for building a new solution as a professional engineering practice as a team.

Course Contents:

Module 1[8L]:
Introduction to CPU and concepts of ALU [2L], Instruction format and Instruction Cycle [1L], Addressing Modes [1L] Fixed- point multiplication -Booth's algorithm. [2L], Fixed-point division - Restoring and non-restoring algorithms [1L], Floating-point number representation-IEEE754 format and Floating-point arithmetic operation [1L].

Module 2 [7L]:
Introduction to basic computer architecture [1L], Stored Program Concepts: Von Neumann & Harvard Architecture [1L], RISC VS CISC [1L],Amdahl law [1L], Performance measurement parameters – MIPS, MFLOPS, SPEC ratings, CPI etc. [2L] Micro programmed and hardwired control unit [1L].

Module 3[8L]:
Introduction to memory-RAM and ROM [1L], Register transfer, memory transfer, Tri-state bus buffer, Memory Hierarchy: Secondary memory [1L], Main Memory [1L], Cache Memory [1L], Mapping Technique in cache memory: Direct, Full Associative and Set Associative [2L], Performance Implementation in Cache Memory [1L], Virtual memory Concepts [1L], page replacement policies [1L].

Module 4[9L]:
Pipelining: Basic concepts, instruction and arithmetic pipeline[2L], data hazards, control hazards and structural hazards, techniques for handling hazards[2L]Pipeline vs. Parallelism, Levels of parallelism [1L], Instruction- Level Parallelism: Basic Concepts, Techniques for Increasing ILP, Superscalar, Super Pipelined and VLIW Processor Architectures [2L], Array and Vector Processors[1L]

Module 5[4L]:
Multiprocessor architecture: taxonomy of parallel architectures; Flynn Classification [1L], Centralized and Shared-

memory architecture: synchronization [1L], Interconnection Network (Omega, Baseline, Butterfly, Crossbar) [2L].

Text Books:

1. Hwang - Advanced Computer Architecture Parallelism Scalability Programmability, Tata McGraw- Hill Education Private Limited ISBN-13: 978-0-07-053070-6 ISBN-10:0-07-053070-X
2. Hwang & Briggs—Computer Architecture & Parallel Processing, TMH
3.

Reference Books:

1. Patterson D.A. and Hennessy, J.L.―Computerarchitectureaquantitativeapproach‖,2nd ed. Morgan Kaufman,1996
2. Hayes J. P., ―Computer Architecture & Organization, McGrawHill
3. Siegel, H.J., ―Interconnection Network for Large Scale parallel Processing‖, 2nd Ed. McGrawHill,1990
4. Design and Analysis of Parallel Algorithm-Schism G. Akl

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	
CO1	3	3	3	2	-	-	-	-	-	-	-	-
CO2	3	3	3	3	-	-	-	-	-	-	-	3
CO3	2	2	2	3	-	-	-	-	-	-	-	-
CO4	3	3	3	3	-	-	-	-	-	-	-	-
CO5	3	3	3	3	-	-	-	-	-	-	-	-

CO-PSO Mapping

Cos	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

Course Name: Design & Analysis of Algorithm
Course Code:CS302
Contact: 3:1:0
Total Contact Hours: 36L
Credits: 4

Prerequisites: To know data-structure and basic programming ability

Course Outcomes (COs):
After attending the course students should be able to

CO1	To understand and illustrate the concepts of time and space complexity, worst case, average case and best-case complexities and the asymptotic notation.
CO2	To analyze and apply the design principles and concepts to various basic algorithm design viz. dynamic programming, greedy methods etc.
CO3	To understand and analyze various string matching and graph algorithms.
CO4	To understand, illustrate and analyze the different complexity classes
CO5	To discuss, implement and analyze, verify the efficiency of the randomized and approximation algorithms.

Course Content:

Module-1 [4L]

Algorithm Development & Complexity Analysis: [4L] Stages of algorithm development for solving a problem: Describing the problem, identifying a suitable technique, Design of an algorithm, Proof of Correctness of the algorithm. Time and Space Complexity, Different Asymptotic notations – their mathematical significance. Solving Recurrences: Substitution Method, Recurrence Tree Method, Master Theorem (Statement Only).

Module-2 [14L]

Algorithm Design Techniques Brute force techniques – Traveling Salesman Problem, Divide and Conquer - Matrix multiplication: Strassen algorithm, Greedy techniques - Fractional Knapsack problem, Job Sequencing with Deadline, Graph Coloring, Finding Minimum Cost Spanning Tree using Prim's and Kruskal's algorithm, Dynamic programming - O/1 Knapsack problem, Matrix chain multiplication, Travelling Salesman Problem, Backtracking-N-Queens Problem, Knights Tour on Chess Board.

Module-3 [3L]

String matching problem: Different techniques – Naive algorithm, string matching using finite automata, and Knuth, Morris, Pratt (KMP) algorithm with their complexities.

Module-4 [5L]

Graph Algorithms Single Source Shortest Path -Dijkstra Algorithm, All pair shortest path – Floyd-Warshall Algorithm. Network Flows, Maximum Flows – Ford-Fulkerson Algorithm, Push Re-label Algorithm, Minimum Cost Flows – Cycle Cancelling Algorithm.

Module-5 [5L]

Complexity Classes: The Class P, The Class NP, Reducibility and NP-completeness – SAT (without proof), 3-SAT, Vertex Cover, Independent Set, Maximum Clique.

Module-6 [5L]

Approximation and Randomized Algorithms [3L], Approximation Algorithms - The set-covering problem – Vertex cover, K-center clustering. Randomized Algorithms - The hiring problem, Finding the global Minimum. Recent Trends [2L]

Text book:

1. "Introduction to Algorithms" by Cormen, Leiserson, Rivest, Stein.
2. "The Design and Analysis of Computer Algorithms" by Aho, Hopcroft, Ullman. 3. "Algorithm Design" by Kleinberg and Tardos.

Reference Books:

1. Design & Analysis of Algorithms, Gajendra Sharma, Khanna Publishing House, New Delhi
2. Design Analysis and Algorithms| by Hari Mohan Pandey.

CO-PO Mapping

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	2	-	-	-	-	-	-	-	2
CO2	3	3	3	3	-	-	-	-	-	-	-	2
CO3	3	3	3	3	-	-	-	-	-	-	-	2
CO4	3	3	3	3	-	-	-	-	-	-	-	2
CO5	3	3	3	3	-	-	-	-	-	-	-	2

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

COURSE NAME: DISCRETE MATHEMATICS

COURSE CODE: M(CS) 301

CONTACT: 3:0:0

TOTAL CONTACT HOURS: 36

CREDITS: 3

Prerequisites:

The students to whom this course will be offered must have the concept of (10+2) standard Mathematics.

Course Objectives:

The objective of this course is to disseminate the prospective engineers with the knowledge of Abstract Algebra, Combinatorics, Recurrence Relation, Propositional Logic, Number Theory and Graph Theory.

Course Outcome(s):

On successful completion of the learning sessions of the course, the learner will be able to:

CO1: Recall the properties of abstract algebra, recurrence relation, propositional logic, number theory and graph theory.

CO2: Explain the theoretical working of the concepts of abstract algebra, recurrence relation, propositional logic, number theory and graph theory.

CO3: Apply the concepts of abstract algebra, recurrence relation, number theory, propositional logic and algorithms of graph theory in Computer Science.

CO4: Examine the Poset, Lattices, algebraic structures and graphs using the underlying concepts.

Course Content:

Module-I: Set Theory **[11L]**

Posets& Lattices: [6L]

Relation: Types of Relations, Properties of Binary Relation, Equivalence Relation, Partial Ordering Relation and Posets, Lattices.

Combinatorics: [2L]

Principle of Inclusion Exclusion, Pigeon Hole Principle.

Generating Functions and Recurrence Relations: [3L]

Generating functions, Recurrence relations: Formulation of different counting problems in terms of recurrence relations, Solution of recurrence relations with constant coefficients by Generating functions method.

Module-II: Propositional Logic**[5L]**

Basics of Boolean Logic, Idea of Propositional Logic, well-formed formula, Logical Connectives, Truth tables, Tautology, Contradiction, Algebra of proposition, Logical Equivalence, Normal Forms: Disjunctive Normal Forms (DNF) and Conjunctive Normal Forms (CNF).

Module-III: Number Theory **[4L]**

Well-Ordering Principle, Divisibility theory and properties of Divisibility, Fundamental theorem of Arithmetic, Prime and Composite Numbers, Greatest Common Divisor and Euclidean Algorithm, Congruence, Residue Classes.

Module-IV: Algebraic Structures **[8L]**

Concepts of Groups, Subgroups and Order, Cyclic Groups, Cosets, Lagrange's theorem, Normal Subgroups, Permutation Groups and Symmetric Groups, Definition of Ring and Field.

Module-V: Graph Theory **[8L]**

Graph theory, Theorems (statement only), Digraphs, Weighted Graph, Walk, Path, Circuit, Connected and Disconnected Graph, Bipartite Graph, Complement of a Graph, Regular Graph, Complete Graph, Adjacency and Incidence matrices of a graph (digraph), Dijkstra's algorithm.

Tree, Binary Tree, Theorems on Tree (statement only), Spanning Tree, Minimal Spanning Tree, Kruskal's Algorithm, Prim's Algorithm.

Project Domains:

1. Study of physical processes through Graph theory.
2. Application of Propositional Logic in real world engineering problems.
3. Application of Abstract Algebra in engineering problems.

Text Books:

1. Deo, N.,*Graph Theory with Applications to Engineering and Computer Science*, Prentice Hall.
2. Mapa, S. K.,*Higher Algebra: Abstract and Linear,* Levant, 2011.
3. Chakraborty, S. K. and Sarkar,B. K.,*Discrete Mathematics*, OXFORD University Press.
4. Rosen, K. H.,*Discrete Mathematics and its Applications*, Tata McGraw – Hill.

Reference Books:

1. Grewal, B. S.,*Higher Engineering Mathematics,* Khanna Pub.
2. Kreyzig, E.,*Advanced Engineering Mathematics,* John Wiley and Sons.
3. Sharma, J.K.,*Discrete Mathematics*, Macmillan.
4. Liu, C. L. and Mohapatra, D. P.,*Elements of Discrete Mathematics A Computer Oriented Approach*, 3rd Edition, Tata McGraw – Hill.
5. Tremblay, J. P. and Manohar, R.,*Discrete Mathematical Structure and It's Application to Computer Science*, TMG Edition, TataMcgraw-Hill.

CO-PO Mapping:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	3	2	-	-	-	-	-	-	-	-	-	1
CO2	3	2	1	-	-	-	-	-	-	-	-	1
CO3	3	3	2	1	-	-	-	-	-	-	-	2
CO4	3	3	3	2	-	-	-	-	-	-	-	2
M(CS) 301	**3**	**2.5**	**2**	**1.5**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1.5**

Weightage Values: Strongly mapped: '3', Moderately mapped: '2', Weakly mapped: '1', Not mapped: '-'.

Course Name: Digital Logic and Electronics
Course Code: EC(CS)301
Contact: 3:0:0
Credit: 3
Total Contact Hours: 36L

Pre-requisite:

Basic concepts of Logic gates, Truth Tables, Concept of basic components of a digital computer.

Course Outcome(s):

CO1: To realize basic gate operations and laws Boolean algebra.
CO2: To understand basic mechanism of digital computers and digital logic behind different arithmetic and control unit operations.
CO3: To design combinational circuits and combinational functions for larger more complex circuits.
CO4: To perform different operations with sequential circuits.
CO5: To understand fundamental concepts and techniques used in Logic families and PLDs

Course Content: –

Module – 1[8L]
Binary Number System [1L]; BCD, ASCII, EBDIC, Gray codes and their conversions [1L], Introduction and laws of Boolean algebra [1L], Boolean functions, Minterm and maxterm, Prime implicants, Representation in SOP and POS forms[2L], Minimization of logic expressions by Karnaugh Map and algebraic method [3L]

Module – 2[8L]
Combinational circuits:
Adder and Subtractor (half-full adder & subtractor) [2L], Serial & Parallel Adder, Carry look ahead adder and Parity Generator[2L], Encoder, Decoder, Multiplexer [2L], Demultiplexer, Comparator, Code Converters [2L]

Module – 3[12L]
Sequential Circuits:
Flip-Flops, SR, JK, Master slave JK, D, T, characteristic Tables, Excitation tables [5L]. Basic concept of Synchronous and Asynchronous counters, Up/Down Counters, Ring counter, Johnson counter, Design of Modulo-N Counter, Counter applications [5L]. Registers (SISO, SIPO, PIPO, PISO) [2L].

Module – 4[8L]
A/D and D/A conversion techniques – Basic concepts (D/A:R-2-R only [2L], A/D: successive approximation [2L])Logic families- TTL, ECL, MOS and CMOS - basic concepts [2L],Programmable logic Array, programmable Array logic, Sequential Programmable Devices [2L].

Text Book:
1. Saliva Hanan S, Digital Circuits and Design, Oxford
2. Morries Mano- Digital Logic Design- PHI

Reference Book:
1. R.P.Jain—Modern Digital Electronics, 2/e, Mc Graw Hill
2. Digital Fundamentals – A Systems Approach – Thomas L. Floyd, Pearson

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	1	-	-	1	-
CO2	3	3	-	2	-	-	2	-	-	1	-	-
CO3	3	3	-	-	-	-	-	2	-	2	-	2
CO4	-	-	3	3	3	-	-	-	2	-	-	-
CO5	3	2	-	-	-	2	-	-	2		2	-

Course Name: Computer Organization and Architecture Lab Course
Code: CS391
Contact: 0:0:3
Credits: 1.5

Prerequisites:
Knowledge of designing different circuits in Computer Organization Lab

Course Outcomes (COs):
After attending the course students should be able to

CO1	Illustrate and use proper syntax in appropriate platforms for developing programs to solve problems related to Mathematics and Engineering fields leading to lifelong learning.
CO2	Apply the knowledge of algorithms in the computational area to efficient programming codes to design the problem using modern tools for solving complex engineering problems.
CO3	Outline different types of digital electronic circuits such as adder, subtract or, encoder decoder, multiplexer, demultiplexer, flip-flops, register, counter using various mapping and modern tools to prepare the most simplified circuit and optimize using various mapping and mathematical methods for solving the problem as a professional engineering practice as a team.
CO4	Apply the knowledge of digital electronic circuits to design memory and ALU and analyze the same to solve engineering-related computational problems as a team.
CO5	Interpret the result of the experiments, prepare laboratory reports based on observed output and analyze it to validate professional ethics and responsibilities and norms of the engineering practice.

List of Experiment:
1. Implement different types of Basic gates and simulate for truth table verification.
2. Implement half adder circuit and simulate for truth table verification.
3. Implement full adder circuit and simulate for truth table verification.
4. Implement half subtractor circuit and simulate for truth table verification.
5. Implement a full subtractor circuit and simulate for truth table verification.
6. Implement Multiplexer, De-Multiplexer circuit and simulate for truth table verification.
7. Implement Encoder, Decoder circuit and simulate for truth table verification.
8. Implement different types of flip flop and simulate for truth table verification.
9. Implement different types of parallel circuits (SISO, SIPO, PISO, PIPO) and simulate the result.
10. Implement ALU and simulate the result.
11. Implement a RAM chip and simulate the result.
12. Innovative Experiments.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO1 2
CO1	3	3	3	2	3	-	-	-	3	-	-	-
CO2	3	3	3	3	3	-	-	-	3	-	-	-
CO3	3	3	3	3	3	-	-	-	3	-	-	-
CO4	3	3	3	3	3	-	-	-	3	-	-	-
CO5	3	3	3	3	3	-	-	-	3	-	-	-

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

Course Name: Design & Analysis of Algorithm Lab
Course Code: CS392
Contact: 0:0:3
Credit: 1.5

Prerequisite:
Programming knowledge

Course Outcomes (COs):
After attending the course students should be able to

CO1	To identify and prove the correctness and analyze the running time of the basic algorithms for those classic problems in various domains.
CO2	To understand and illustrate methods for analyzing the efficiency and correctness of algorithms (such as exchange arguments, recurrence, induction, and average case analysis)
CO3	To analyze and design algorithms using the dynamic programming, greedy method, Backtracking, Branch and Bound strategy, and recite algorithms that employ this strategy.
CO4	To understand, compare, contrast, and choose appropriate implementation of the algorithmic design techniques to present an algorithm that solves a given problem.
CO5	To Identify and analyze criteria and specifications appropriate to new problems.

Course Content:

A. Implementation of various Divide & Conquer Methods; viz. Matrix Multiplication.
B. Implementations of various Dynamic Programming Methods, viz. Matrix Chain Multiplication Method, Travelling Salesman Problem etc.
C. Implementations of various Branch & Bound Techniques, viz.
D. Implementations of various Backtracking Methods, viz. n-Queen Problem.
E. Implementations of Greedy Method, viz. Fractional Knapsack Problem, Job Sequencing Problem etc.
F. Implementations of String-matching Algorithm viz. Naïve Algorithm, String Matching with Finite Automata etc.
G. Implementations of Various Graph Algorithms, viz. Dijkstra ‘s Algorithm, Floyd Algorithm etc.
H. Implementation of some Real-Life Trendy Problems

CO-PO Mapping

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	3	-	-	-	3	-	-	3
CO2	3	3	3	2	3	-	-	-	3	-	-	3
CO3	3	3	2	3	3	-	-	-	3	-	-	3
CO4	3	3	2	2	3	-	-	-	3	-	-	3
CO5	3	3	3	2	3	-	-	-	3	-	-	3

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

Course Name: Digital Electronics Lab
Course Code: EC(CS)391
Contact: 0:0:3
Credit: 1.5

Prerequisite:
Basic concepts of Logic gates, Truth Tables, function realization –minimization of Logic expressions by K-map, Concept of basic components of a digital computer, Binary Arithmetic

Course Outcomes (COs):
After attending the course students should be able to

CO1	Knowledge of Electronic components such as Resistors, Capacitors, Diodes, Transistors measuring equipment like DC power supply, Multimeter, CRO, Signal generator, DC power supply.
CO2	Analyze the characteristics of Junction Diode, Zener Diode, BJT & FET and different types of Rectifier Circuits.
CO3	Determination of input-offset voltage, input bias current and Slew rate, Common- mode Rejection ratio, Bandwidth and Off-set null of OPAMPs.
CO4	Able to know the application of Diode, BJT &OPAMP.
CO5	Familiarization and basic knowledge of Integrated Circuits

Course Content:

1. A) Realization of basic gates and universal gates.
B) Realization of basic gates using universal gates.
2. Design a Half adder and Full Adder circuit using basic gates and verify its output.
3. Design a Half subtractor and Full Subtractor circuit using basic gates and verify its output
4. Design an Adder/Subtractor composite unit.
5. Design of a _Carry-Look-Ahead Adder circuit.
6. Realization of a) Encoder, b) Decoder c) Multiplexer, d) De-mux, e) Comparator and their Truth Table verification.
7. Realization of RS / JK / D flip flops using logic gates.
8. Design of Shift Register using J-K / D Flip Flop.
9. Realization of Synchronous Up/Down counters.
10. Design of MOD- N Counter
11. Study of DAC
12. Study of logic families and PLDs

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3	3	-	-	-	-	-	-	-
CO2	3	3	3	3	3	-	-	-	-	-	-	-
CO3	3	3	3	3	3	-	-	-	-	-	-	-
CO4	3	3	3	3	3	-	-	-	-	-	-	-
CO5	3	3	3	3	3	-	-	-	-	-	-	-

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	2	2	2
CO2	2	2	2
CO3	2		2
CO4	2	2	2
CO5	2	2	2

Course Name: IT Workshop Lab (SciLab/MATLAB/C++)
Course Code: CS393
Contact: 0:1:3
Credits: 2.5

Prerequisite
Computer Fundamentals and principles of computer programming

Course Outcomes (COs):
After attending the course students should be able to

CO1	Demonstrate a thorough understanding of modular programming by designing programs that require the use of programmer-defined functions.
CO2	Demonstrate a thorough understanding of arrays by designing and implementing programs that search and sort arrays.
CO3	Demonstrate a thorough understanding of the object-oriented programming concepts of encapsulation, data abstraction and composition by designing and implementing classes including the use of overloaded functions and constructors.
CO4	Demonstrate a thorough understanding of the concept of pointers and dynamic memory allocation the implementation of programmer-defined functions and classes by writing code, performing unit testing and debugging of multiple complex programs.
CO5	Demonstrate an understanding of the differences between C and C++ in the areas of strings, pass by reference/passing pointers, and structs by designing and implementing programs that use C strings, C++

Course Content:

1. Introduction of UNIX/Linux Operating System which includes preliminary commands, start-up & shutdown methodology, file.
2. Handling as well as introduction to editors like Vi editor, introduction to GNU C & C++ compiler, as well as introduction to GNU & GDB script.
3. Introduction to C++, basic loop control, executing programs.
4. Writing functions, selection statements, review of functions and parameters, command line arguments, recursion, I/O streams, arrays and string manipulation, pointers, structures & unions.
5. Object-Oriented Programming in C++, fundamentals of classes, constructors-destructors.
6. Dealing with member functions, operator overloading and polymorphism (both static & dynamic).
7. Dealing with inheritance, derived class handling.
8. Abstract class, virtual class, overriding, template class, name-space & exception handling.
9. Dynamic memory allocation, implementation of Linked Lists, using C++.
10. Reading and Writing to file, Numerical simulation.
11. Innovative experiments/Projects

Text Books

1. The C++ Programming Language by Bjarne Stroustrup Addison-Wesley publisher
2. Object-Oriented Programming in C++ b by Robert Lafore Publisher: Sams

Reference Books

1. Object Oriented Programming with C++ by Balaguruswamy McGraw Hill Education; Sixth edition Addison-Wesley publisher
2. C++ Programming Language (4th Edition) by Bjarne Stroustrup- Pearson publisher

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	2	-	-	-	3	-	-	-
CO2	3	2	2	2	2	-	-	-	3	-	-	-
CO3	3	3	3	2	2	-	-	-	3	-	-	-
CO4	3	3	3	2	2	-	-	-	3	-	-	-
CO5	3	3	3	2	2	-	-	-	3	-	-	-

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

2nd Year 4th Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A. THEORY									
1	ENGG	Major	CS401	Operating Systems	3	0	0	3	3
2	ENGG	Major	CS402	Computer Networks	3	0	0	3	3
3	ENG	Major	CS403	Formal Language and Automata Theory	3	0	0	3	3
4	SCI	Minor	M(CS)401	Probability and Statistics	3	0	0	3	3
5	HUM	Ability Enhancement Course	HU(CS)401	Principles of Management	2	0	0	2	2
B. PRACTICAL									
1	ENGG	Major	CS491	Operating Systems Lab	0	0	3	3	1.5
2	ENGG	Major	CS492	Computer Networks Lab	0	0	3	3	1.5
3	ENGG	Major	CS493	gramming using Python	0	0	3	3	1.5
4	ENGG	Minor	M(CS)491	merical Methods Lab	0	0	3	3	1.5
5	HUM	Ability Enhancement Course	HU(CS)491	Soft Skill & Aptitude	2	0	0	2	1
Total of Theory, Practical								28	21

Paper Name: Operating System
Paper Code: CS401
Contact Hours/Week: 3
Credit: 3
Total Contact Hours: 36L

Prerequisites:

1. Computer organization
2. Computer Architecture
3. Data Structures
4. Algorithms & Programming Concept

Course Outcomes (COs):

After attending the course students should be able to

CO1	Understand the fundamental concepts of Operating System, Protection & Security and differentiate different types of Operating System.
CO2	Understand and implement process & thread; understand, apply, compare different process synchronization algorithm and inter process communication to solve engineering problems
CO3	Understand/explain/analyze different synchronization techniques, critical section problems and deadlock and apply them to solve engineering problems.
CO4	Understand/explain different memory management techniques including virtual memory management; also able to apply, compare, and implement different page replacement algorithms to solve engineering problems.
CO5	Understand/explain different I/O mechanisms, File structures and disk management techniques and solving engineering problems applying different disk scheduling algorithms.

Course Content:

Module – 1:[3L]

Functionalities of Operating System, Evolution of Operating System.
Types of Operating System: batch, multi-programmed, time-sharing, real-time, distributed, parallel, Structural overview, Protection & Security.[3L]

Module – 2: [9L]

Processes: Concept of processes, process states, PCB, process scheduling, co-operating processes, independent process, suspended process, Interaction between processes and OS, [2L]
Threads: overview, benefits of threads, user and kernel level threads. [1L]
CPU scheduling: Scheduling criteria, preemptive & non-preemptive scheduling, scheduling algorithms (FCFS, SJF, SRTF, RR, priority, multilevel queue, multilevel feedback queue scheduling). [6L]

Module – 3: [10L]

Process Synchronization: background, critical section problem, synchronization hardware, classical problems of synchronization (producer-consumer, readers-writer, dining philosophers, etc), semaphores.[5L] Deadlocks: deadlock characterization, methods for handling deadlocks, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock.[5L]

Module 4: [6L]

Background, logical vs. physical address space, swapping, contiguous memory allocation, paging, Segmentation, TLB. [3L]

Virtual Memory: background, demand paging, page replacement algorithms (FCFS, LRU, Optimal), thrashing, Working set model. [3L]

Module 5: [8L]

Disk structure, disk scheduling (FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK etc), disk reliability, disk formatting, boot block, bad blocks. [2L]

File: File concept, access methods, directory structure, file system structure, UNIX file structure, allocation methods (contiguous, linked, indexed), free-space management (bit vector). [2L]

I/O: I/O hardware, polling, interrupts, DMA, caching, buffering, blocking-non blocking I/O. [1L]

Case Study: Inter-process communication: Message passing. Thread models, monitors [3L]

Text Book:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts.
2. Operating systems: concepts and design by Milan Milenkovič- Mc. Graw-Hill Publication

Reference Book:

1. Dietel H. N., ―An Introduction to Operating Systems‖, Addison Wesley.
2. Andrew Tanenbaum, Modern Operating Systems, Prentice Hall.
3. William Stallings, Operating Systems, Prentice Hall.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3								3
CO2	3	3	3	3								3
CO3	3	3	3	3								3
CO4	3	3	3	3								3
CO5	3	3	3	3								3

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

Course Name: Computer Networks
Course Code: CS402
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite:

1. Familiarity and knowledge of Operating Systems and Computer Architecture.
2. Also require a little bit of programming languages concepts like C, Java**.**

Course Outcomes (COs):

After attending the course students should be able to

CO1	Understand basics of computer network and different architecture and topologies of computer network and analyze the requirements for a given organizational structure and select the most appropriate networking architecture and technologies.
CO2	Understand/analyze different protocols of the data link layer and apply them to solve engineering problems.
CO3	Understand/analyze different protocols of Network and Transport Layer and apply them to solve engineering problems.
CO4	Understand/analyze different protocols of session and application layer and apply them to solve engineering problems.
CO5	Develop, Analyze, specify and design the topological and routing strategies using socket programming.

Course Contents:

Module 1: Introduction [6L]

Introduction (3L):

Introduction: Computer Network, data communication, topology, OSI & TCP/IP Reference Models, layers and characteristics, Wireless Network, comparison to wired and wireless network.

Physical Layer: [3L]

Overview of data (analog & digital), signal (analog & digital), transmission (analog & digital) & transmission media (guided & unguided); Circuit switching: time division & space division switch, TDM bus; Telephone Network;

Module 2: Data Link Layer [9L]

Framing, Error Control, Error Detection and Correction, Flow Control, Data Link Protocols, Simple Stop- and-Wait Protocol, ARQ mechanism, Sliding Window Protocols, One-Bit Sliding Window Protocol, Go- Back-N and Selective Repeat, HDLC, PPP Medium Access Control Sub-layer, The Channel Allocation.[5L] Multiple Access Protocols**:** ALOHA, Carrier Sense Multiple Access Protocols, IEEE 802.x Ethernet, Switched Ethernet, Fast Ethernet, Gigabit Ethernet, 10 Gigabit Ethernet, Wireless LANs - IEEE 802.xx, Bluetooth, RFID, Bridges, Virtual LANs, Switching**.**[4L]

Module 3: Network Layer [10L]
IP Addressing, IPv4and IPv6. Difference IPv4and IPv6, Conversion ofIPv4and IPv6, Subnetting, Supernetting, Design Issues, Store-and-Forward Packet Switching, Virtual-Circuit and Datagram Networks, ARP, IP, ICMP, IPV6, BOOTP and DHCP–Delivery protocols Other Protocols such as mobileIP in wireless Network. [5L]
Routing: Shortest Path Algorithms, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing, Broadcast Routing, Multicast Routing, Anycast Routing: RIP, OSPF, BGP; Routing for Mobile Hosts. [5L]

Module 4: Transport layer: [5L]

Process to Process delivery; UDP; TCP, SCTP, TCP RENO, TCP/IP in Wireless environment, Congestion control in TCP: Congestion Control: Open Loop, Closed Loop packets; Quality of service: techniques to improve QoS: Leaky bucket algorithm, Token bucket algorithm. [4L]
Advanced topic such as Remote Procedure Call, Delay Tolerant Networks. [1L]

Module 5: Application Layer [5L]

Introduction to DNS, SMTP, SNMP, FTP, HTTP & WWW: Cryptography (Public, Private Key based), Digital Signature, Firewalls

Text books:

1. B. A. Forouzan – ―Data Communications and Networking (3rd Ed.) ―– TMH
2. S. Tanenbaum – ―Computer Networks (4th Ed.)‖ – Pearson Education/PHI

Reference books:

1. W. Stallings – ―Data and Computer Communications (5th Ed.)‖ – PHI/ Pearson Education
2. Zheng & Akhtar, Network for Computer Scientists & Engineers, OUP
3. Comer – ―Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.)‖ – Pearson Education/PHI

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	2				2	2		3
CO2	3	3	3	3	3				2	2		3
CO3	3	3	3	3	3				2	2		3
CO4	3	3	3	3	3				2	2		3
CO5	2	3	3	3	3				2	2		3

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

Course Name: Formal Language and Automata Theory Course
Code: CS403
Contacts: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisites:

1. Digital Logic
2. Computer organization
3. Computer Fundamentals

Course Outcomes (COs):

After attending the course students should be able to

CO1	Understand the fundamental concepts of Finite State Automata to Explain or Illustrate and Identify problems where students can Apply the concept appropriately to Solve them.
CO2	Understand the fundamental concepts of Regular Expressions and its relation with DFA so that they can Develop regular expression for a specified language and Validate it.
CO3	Understand the fundamental concepts of Context Free Grammar so that they can Design grammar for a specified language and Validate it.
CO4	Explain or Illustrate the fundamental operating principles of Push Down Automata and Use it appropriately to Solve problems.
CO5	Understand the operating principles of Turing Machine and Design Turing Machines to Proposesolutionstotherelatedproblemsappropriatelyandvalidatetheeffectivenessaswell as limitations of computations making the students aware of its utilitarian importance for further explorations leading towards lifelong learning.

Course Contents:

Module-1: [9L]

Fundamentals: Basic definition of sequential circuit, block diagram, mathematical representation, concept of transition table and transition diagram, **[1L]**
Introduction to Finite State Model (FSM), Design of sequence detector, Finite State Machine, Finite Automata, Deterministic Finite Automation (DFA) and Non-deterministic Finite Automation (NFA), Transition diagrams, Transition tables and Language recognizers. **[3L]**
NFA with empty transitions, Equivalence between NFA with and without empty transitions. NFA to DFA conversion. **[2L]**
Minimization of FSM: Minimization Algorithm for DFA, Introduction to Myhill-Nerode Theorem **[2L]**
Limitations of FSM, Application of Finite Automata**[1L]**

Module-2: [7L]

Finite Automata with output − Moore & Mealy machine. Representation of Moore & Mealy Machine, Processing of the String through Moore & Mealy Machine, Equivalence of Moore & Mealy Machine – Inter- conversion. **[2L]**
Equivalent states and Distinguishable States, Equivalence and k-equivalence, Minimization of Mealy Machine **[1L]**
Minimization of incompletely specified machine−Merger Graph, Merger Table, Compatibility Graph **[2L]**
Lossless and Lossy Machine – Testing Table, Testing Graph **[2L]**

Module-3: [5L]
Regular Languages, Regular Sets, Regular Expressions, Algebraic Rules for Regular Expressions, Arden's Theorem statement and proof **[1L]**
Constructing Finite Automata (FA) for given regular expressions, Regular string accepted by FA **[2L]**
Constructing Regular Expression for a given Finite Automata **[1L]**
Pumping Lemma of Regular Sets. Closure properties of regular sets **[1L]**

Module-4: [10L]

Grammar Formalism-Context Free Grammars, Derivation trees, sentential forms. Rightmost and leftmost derivation of strings, Parse Tree, Ambiguity in context free grammars. **[1L]**
Minimization of Context Free Grammars. **[1L]**
Removal of null and unit production **[1L]**
Chomsky normal form and Greibach normal form. **[1L]**
Pumping Lemma for Context Free Languages. **[1L]**

Enumeration of properties of CFL, Closure property of CFL, Ogden's lemma & its applications **[1L]**
Regular grammars−right linear and left linear grammars **[1L]**
Pushdown Automata: Pushdown automata, definition. Introduction to DCFL, DPDA, NCFL, NPDA **[1L]**
Acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. **[1L]**
Equivalence of CFL and PDA, inter-conversion. [**1L]**

Module-5: [5L]

Turing Machine: Definition, model **[1L]**
Design of TM, Computable functions **[1L],** Church 's hypothesis, counter machine **[1L]** Types of Turing machines **[1L]**
Universal Turing Machine, Halting problem **[1L]**

Textbook:

1. Introduction to Automata Theory Languages and Computation‖, Hopcroft. E. and Ullman J.D., Pearson Education.

Reference Books:

1. Formal Languages and Automata Theory‖, C. K. Nagpal, Oxford
2. Switching and Finite Automata Theory‖, Zvi Kohavi, 2nd Edition, Tata McGraw Hill

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3								3
CO2	2	2	2	2								3
CO3	3	3	3	3								3
CO4	3	3	3	3								3
CO5	3	3	3	3					2	2		3

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

COURSE NAME: PROBABILITY AND STATISTICS

COURSE CODE: M(CS) 401

CONTACT: 3:0:0

TOTAL CONTACT HOURS: 36

CREDITS: 3

Prerequisites:

The students to whom this course will be offered must have the concept of (10+2) standard Mathematics.

Course Objectives:

The objective of this course is to disseminate the prospective engineers with the knowledge of probabilistic approaches and inferential statistics.

Course Outcome(s):

On successful completion of the learning sessions of the course, the learner will be able to:

CO1: Recall the properties related probability distribution and inferential statistics.

CO2: Explain the theoretical working of the concepts of probability distribution and inferential statistics.

CO3: Apply the appropriate mathematical tools using the concepts of probability distribution and inferential statistics in Computer Science.

CO4: Analyze the real-world problems using the underlying principles of both probabilistic and statistical approaches.

Course Content:

Module-I: Probability and Random Variables **[10L]**

Discrete and continuous random variables, probability mass, probability density and cumulative distribution functions, mathematical expectation, Moments, Moment generating functions, Binomial, Poisson, Uniform, Exponential and Normal distributions

Module-II: Two Dimensional Random Variables **[9L]**

Joint distributions, Marginal and conditional distributions, Covariance, Correlation and linear regression, T Transformation of random variables, Central limit theorem (for independent and identically distributed random variables).

Module-III: Sampling Theory & Estimation of Parameters **[10L]**

Sampling Theory: Random sampling, Parameter & Statistic, Standard error of statistic, Distributions of the sample mean and the sample variance for a normal population, Chi-Square distributions, t distributions and F distributions.

Estimation of Parameters: Unbiased and consistent estimators, Point estimation, Interval estimation, Maximum likelihood estimation of parameters (Binomial, Poisson and Normal), Confidence intervals and related problems.

Module-IV: Testing of Hypothesis **[7L]**

Simple and Composite hypothesis, Critical region, Level of significance, Type I and Type II errors, one sample and two sample tests for means and proportions, $\chi 2$ - test for goodness of fit.

Project Domains:

1. Study of uncertainty in real world phenomena using Probability Distribution.
2. Application of Sampling Theory and Estimation of Parameters in engineering problems.
3. Application of Testing of Hypothesis in engineering problems.

Text Books:

1. Das, N.G., *Probability and Statistics*, The McGraw Hill Companies.
2. Gupta S. C. and Kapoor V. K., *Fundamentals of Mathematical Statistics*, Sultan Chand & Sons.
3. Goon A.M., Gupta M. K. and Dasgupta, B., *Fundamental of Statistics*, The World Press Pvt. Ltd.
4. Kreyszig, E., *Advanced Engineering Mathematics*, 9thEdition; John Wiley & Sons, 2006.

Reference Books:

1. Lipschutz, S. and Lipson, M., *Schaum's Outline in Probability* (2ndEd.); McGraw Hill Education.
2. Soong, T. T., *Fundamentals of Probability and Statistics for Engineers*; Wiley Publications.
3. Spiegel, M. R., *Theory and Problems of Probability and Statistics (Schaum's Outline Series)*; McGraw Hill Book Co.
4. Montgomery, D.C. and Runger, G.C., *Applied Statistics and Probability for Engineers*, Wiley Publications.

CO-PO Mapping:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	3	2	-	-	-	-	-	-	-	-	-	1
CO2	3	2	1	-	-	-	-	-	-	-	-	1
CO3	3	3	2	1	-	-	-	-	-	-	-	2
CO4	3	3	3	2	-	-	-	-	-	-	-	2
M(CS) 401	**3**	**2.5**	**2**	**1.5**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1.5**

Weightage Values: Strongly mapped: '3', Moderately mapped: '2', Weakly mapped: '1', Not mapped: '-'.

Course Name: Principles of Management
Course Code: HU(CS)401
Contacts: 2:0:0
Total Contact Hours: 24
Credits: 2

Course Outcomes (COs):

After attending the course students should be able to

CO1	Understand the basic concepts and technologies used in the field of management information Systems
CO2	Have the knowledge of the different types of management information systems
CO3	Understand the processes of developing and implementing information systems.
CO4	Be aware of the ethical, social, and security issues of information systems.
CO5	An ability to effectively integrate IT-based solutions into the user environment

Course Content:

Module-1: Management Concepts: Definition, roles, functions and importance of Management, Evolution of Management thought-contribution made by Taylor, Fayol, Gilbreth, Elton Mayo, McGregor, Maslow **(4L)**

Module - 2: Planning and Control: Planning: Nature and importance of planning, -types of planning, Levels of planning - The Planning Process. –MBO, SWOT analysis, McKinsey‘s7S Approach.
Organizing for decision making: Nature of organizing, span of control, Organizational structure –line and staff authority.
Basic control process -control as a feedback system – Feed Forward Control –Requirements for effective control – control **(4L)**

Module - 3: Group dynamics: Types of groups, characteristics, objectives of Group Dynamics. Leadership: Definition, styles & functions of leadership, qualities for good leadership, Theories of leadership **(4L)**

Module – 4: Work Study and work measurement: Definition of work study**,** Method Study Steps, Tools and Techniques used in the Method Study and Work Measurement Time Study: Aim & Objectives, Use of stopwatch procedure in making Time Study. Performance rating, allowances and its types. Calculation of Standard Time. Work sampling **(4L)**

Module - 5: Marketing Management: Functions of Marketing, Product Planning and development, Promotional Strategy (**2L)**

Module - 6: Quality management: Quality definition, Statistical quality control, acceptance sampling, Control Charts –Mean chart, range chart,c chart,p chart,np chart, Zero Defects, Quality circles,, Kaizen & Six Sigma, ISO -9000 Implementation steps, Total quality management **(6L)**

Text Books:

1. Essentials of Management, by Harold Koontz & Heinz Weihrich Tata McGraw
2. Production and Operations Management-K. Aswathapa, K. Shridhara Bhat, Himalayan Publishing House

References:

1. Organizational Behavior, by Stephen Robbins Pearson Education, New Delhi
2. New era Management, Daft, 11th Edition, Cengage Learning
3. Principles of Marketing, Kotlar Philip and Armstrong Gary, Pearson publication

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	2							3	
CO2	3	2	2	2							3	
CO3	3	3	2	2							3	
CO4	3	2	2	2							3	
CO5	3	2	2	2							3	

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	2	2	2
CO2	2	2	2
CO3	2	2	2
CO4	2	2	2
CO5	2	2	2

Course Name: Operating Systems Lab
Course Code: CS491
Allotted Hours: 36L

Prerequisites:

1. Computer organization
2. Computer Architecture
3. Data Structures
4. Algorithms & Programming Concept

Course Outcomes (COs):

After attending the course students should be able to

CO1	Analyze and simulate CPU Scheduling Algorithms like FCFS, Round Robin, SJF,and Priority.
CO2	Understand the concepts of deadlock in operating systems.
CO3	Implement them in Multiprogramming system.
CO4	Create process creation and implement inter process communication
CO5	Analyze the performance of the various page replacement schemes

Course Content:

1. **Essential Linux Commands[9P]:** Commands for files and directories cd, cp, mv, rm, mkdir, more, less, creating and viewing files, using cat, file comparisons, View files, kill, ps, who, sleep, grep, fgrep, find, sort, cal, banner, touch, file related commands – ws, sat, cut, grep etc. Mathematical commands –expr, factor, units, Pipes (use functions pipe, popen, pclose), named Pipes (FIFOs, accessing FIFO)

2. **Shell Programming [6P]:** Creating a script, making a script executable, shell syntax (variables, conditions, control structures, functions, and commands).

3. **Process [3P]**: Starting new process, replacing a process image, duplicating a process image.

4. **Semaphore [3P]:** Programming with semaphores (use functions semget, semop, semaphore_p, semaphore_v).

5. **POSIX Threads[6P]:** Programming with pthread functions (viz. pthread_create, pthread_join, pthread_exit, pthread_attr_init, pthread_cancel).

6. **Shared Memory [9P]:** Create the shared memory, Attach the shared memory segment to the address space of the calling process, Read information from the standard input and write to the shared memory, Read the content of the shared memory and write on to the standard output, Delete the shared memory

Books:

1. Yashavant P. Kanetkar, UNIX Shell Programming, 1st edition, BPB Publications
2. W. Richard Stevens, UNIX Network Programming, 2nd edition, Prentice Hall

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3	3				3			
CO2	3	3	3	3	3				3			
CO3	3	3	3	3	3				3			
CO4	3	3	3	3	3				3			
CO5	3	3	3	3	3				3			

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

Course Name: Computer Networks Lab
Course Code: CS492
Allotted Hours: 36L

Prerequisites:

1. Familiarity and knowledge of Computer Networkand Computer Architecture
2. Also require strong knowledge of programming languages like C, Java and UNIX or Linux environment.

Course Outcomes (COs):

After attending the course students should be able to

CO1	To design and implement small size network and to understand various networking commands.
CO2	To provide the knowledge of various networking tools and their related concepts.
CO3	To understand various application layer protocols for its implementation inclient/server environment
CO4	Understand the TCP/IP configuration for Windows and Linux
CO5	Learn the major software and hardware technologies used on computer networks

Course Contents:

1. Familiarization of UNIX or Linux environment, UNIX or Linux general Commands specially Network Commands. Familiarization of Internetworking - Network Cables - Color coding - Crimping. Internetworking Operating Systems - Configurations. **[6L]**
2. Socket Programming using TCP and UDP **[18L]**
3. Implementing routing protocols such as RIP, OSPF. **[2L]**
4. Familiarization of advanced simulators like Packet Tracer, NS2/NS3, OMNET++, TinyOS **[4L]**
5. Server Configuration: only web server (Iftime permit, Instructor can do more than that) **[6L]**

Textbooks:

1. TCP sockets in C Programs-Practical guide for Programmers By Micheal, J Donahoo and Kenneth L calvert.
2. 2.Socket Programming by Raj Kumar Buyaa.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	3							2
CO2	3	3	3	3	3							2
CO3	3	3	3	3	3							2
CO4	3	3	3	3	3							2
CO5	2	3	2	2	3							2

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	3	3	3
CO2	3	3	3
CO3	3	3	3
CO4	3	3	3
CO5	3	3	3

Course Name: Programming using Python
Course Code: CS493
Allotted Hours: 36L

Prerequisites:
Knowledge of Mathematics and basic concepts of Programming

Course Outcome(s):
After completion of the course students will be able to CO1:
Understand basic of Python Programming Language
CO2: Understand the use of Conditional statement and Loops
CO3: Learn functions in python and represent collection type data using List and Dictionary
CO4: Read and write data from & to files in Python
CO5: Understand Numpy array and numerical operations on Numpy array.

Course Contents:

1. **Basics of Python:** Python Installation, python variables, data types and Operator.
2. **Loops:** While and For loops, Python Syntax, Colon & Indentation, Conditional Statements: if, elif and else.
3. **Functions:** Defining Functions in python; passing arguments.
4. **String:** Python Programming to explore string functions
5. **Lists**: Python programs using Lists; understand the use of List methods, Slicing on List.
6. **Sets**: Working with Sets; Write programs to show different set operations.
7. **Dictionary**: Demonstrate the use of Dictionaries
8. **File handling:** Reading & Writing data from a file, Redirecting output streams to files.
9. **Basics of Module & Package:** Creation, Updation, Deletion.
10. **Numpy**: Numerical operations using Numpy array; slicing numpy array; stacking numpy arrays; Write programs to show different numerical operations on numpy array;
11. **Pandas**: Numerical operations using Pandas array; slicing Pandas array; stacking Pandas arrays; Write programs to show different numerical operations on Pandas.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	-	-	2	-	-	-	-	-	-	-	1
CO2	2	3	-	2	-	-	-	-	-	-	-	-
CO3	2	3	3	2	-	-	-	-	-	-	-	1
CO4	2	3	2	2	-	-	-	-	-	-	-	-
CO5	3	3	3	3	-	-	-	-	-	-	-	2

Course Name: Numerical Methods Lab
Course Code: M(CS)491
Allotted Hours: 30L

Prerequisite: Any introductory course on programming language (example. C/ Matlab).

Course Outcomes (COs):

After attending the course students should be able to

CO1	Describe and explain the theoretical workings of numerical techniques with the help of C
CO2	Compute basic command and scripts in a mathematical programming language
CO3	Apply the programming skills to solve the problems using multiple numerical approaches.
CO4	Analyze if the results are reasonable, and then interpret and clearly communicate the
CO5	Apply the distinctive principles of numerical analysis and the associated error measures.

Course Content:

1. Assignments on Newton forward /backward, Lagrange‘s interpolation.

2. Assignments on numerical integration using Trapezoidal rule, Simpson‘s 1/3 rule, Weddle‘s rule.

3. Assignments on numerical solution of a system of linear equations using Gauss elimination, Tridiagonal matrix algorithm, Gauss-Seidel iterations. LU Factorization method.

4. Assignments on numerical solution of Algebraic Equation by Bisection method, Regula-Falsi method, Secant Method, Newton-Raphson method

5. Assignments on ordinary differential equation: Euler‘s method, Euler‘s modified method, Runge-Kutta methods, Taylor series method and Predictor-Corrector method.

   Implementation of numerical methods on computer through C/C++ and commercial Software Packages: Matlab/Scilab/Labview / Mathematica/NAG (Numerical Algorithms Group) / Python.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	3	2			3			
CO2	3	3	2	2	3	2			3			
CO3	3	3	2	2	3	2			3			
CO4	3	3	2	2	3	2			3			
CO5	3	3	2	2	3	2			3			

CO-PSO Mapping

COs	PSO1	PSO2	PSO3
CO1	2	2	2
CO2	2	2	2
CO3	2	2	2
CO4	2	2	2
CO5	2	2	2

Paper Name: Soft Skills and Aptitude
Paper Code: HU (CS) 491
Credits: 1
Contact Hours: 2
Lecture: 24

Course Outcomes (COs)

CO1: Develop effective communication skills (verbal, written, and non-verbal) for professional environments.
CO2: Enhance problem-solving and analytical thinking through logical reasoning and aptitude training.
CO3: Demonstrate strong teamwork, leadership, and interpersonal skills in a corporate setting.
CO4: Manage time efficiently and handle workplace stress through practical strategies.
CO5: Build a professional profile (resume, LinkedIn, portfolio) to enhance employability.

Module 1: Communication & Corporate Etiquette (5L)

Objective: To develop professional communication skills and workplace etiquette.
Topics Covered:

- Fundamentals of Corporate Communication (Email, Meetings, Reports)
- Business Writing: Resume, Cover Letter, and Statement of Purpose (SOP)
- Public Speaking & Presentation Skills (With AI/Tech-based Tools)
- Group Discussion & Personal Interview Techniques
- Corporate Etiquette: Dress Code, Body Language, Networking
- Persuasion & Negotiation Skills in Interviews
- Handling Workplace Conflicts and Professional Conduct

Practical Exercises:
Mock GD & PI Sessions
Elevator Pitch Challenge
Drafting Business Emails and Reports
Role-playing Interview Scenarios

Module 2: Verbal & Analytical Ability for Competitive Exams (4L)

Objective: To strengthen verbal reasoning, numerical ability, and logical thinking for aptitude tests.
Topics Covered:

- Grammar & Sentence Correction
- Vocabulary Building & Contextual Usage
- Reading Comprehension Techniques
- Logical Reasoning: Syllogisms, Blood Relations, Seating Arrangements
- Quantitative Aptitude: Percentage, Ratio-Proportion, Time & Work

- Data Interpretation & Graphical Analysis
- Puzzles and Critical Thinking Exercises
- Decision-Making and Problem-Solving Strategies

Practical Exercises:
Weekly Verbal & Quantitative Aptitude Tests
Logical Puzzles and Reasoning Games
Speed Reading & Summary Writing
Mock Corporate Aptitude Tests

Module 3: Teamwork, Leadership & Conflict Resolution (5L)

Objective: To cultivate leadership, teamwork, and interpersonal effectiveness.
Topics Covered:

- Teamwork vs. Individual Contribution
- Leadership Styles & Decision Making
- Conflict Management & Negotiation Skills
- Emotional Intelligence (EQ) in Workplace Relationships
- Cross-Cultural Communication in Global Tech Companies
- Adapting to Change & Handling Difficult Conversations
- Empathy & Relationship Management in Professional Spaces

Practical Exercises:
Team-Based Problem-Solving Activities
Leadership Role-Play Scenarios
Conflict Resolution Case Studies
Workplace Adaptability Drills

Module 4: Time & Stress Management for Engineers (3L)

Objective: To develop productivity habits for balancing work and personal growth.
Topics Covered:

- Prioritization Techniques (Eisenhower Matrix, Pomodoro Technique)
- Handling Work Pressure & Burnout in IT Jobs
- Effective Goal-Setting & Productivity Hacks
- Mindfulness & Stress Reduction Techniques
- Work-Life Balance for Engineers
- Managing Deadlines & Overcoming Procrastination
- Stress-Handling Strategies in High-Stakes Environments

Practical Exercises:
Time-Blocking & Task Prioritization Drills

Stress-Management Workshops
Industry Case Studies on Work-Life Balance
Productivity Habit Challenges

Module 5: Profile Building & Career Readiness (7L)

Objective: To prepare students for industry placements and professional networking.
Topics Covered:

- Resume Optimization (ATS-Friendly Formats)
- LinkedIn & GitHub Profile Enhancement
- Personal Branding & Online Portfolio Development
- Industry Certifications & Upskilling (Coursera, Udemy, AWS, etc.)
- Networking Strategies & Building Corporate Connections
- Strategies for Effective Salary Negotiation
- Leveraging Internship & Freelance Experience for Career Growth

Practical Exercises:
Resume Review & LinkedIn Profile Audit
Creating a Personal Website/Portfolio
Mock HR & Technical Interviews
Industry Networking & Mentorship Sessions

Evaluation & Assessment

- **Class Participation & Practical Assignments** – 20%
- **Aptitude & Verbal Ability Tests** – 20%
- **Mock GD & PI Performance** – 20%
- **Corporate Project & Leadership Activity** – 20%
- **Final Profile Submission (Resume, LinkedIn, Portfolio)** – 20%

Industry Integration & Certifications

- Guest Lectures from Corporate Trainers, HR Leaders & Entrepreneurs
- Collaboration with Placement Cells & Corporate Partners (e.g., TCS, Infosys, Wipro)
- Soft Skill Certifications (TCS iON, NPTEL, LinkedIn Learning)

Program Outcomes (POs) Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓		✓		✓			✓		✓	✓	✓
CO2	✓	✓		✓	✓					✓		✓
CO3			✓			✓	✓	✓	✓	✓	✓	✓

CO4					✓	✓	✓	✓	✓	✓	✓	✓
CO5			✓		✓			✓	✓	✓	✓	✓
CO6	✓	✓		✓	✓				✓	✓		✓
CO7	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		

Program Specific Outcomes (PSOs) Mapping

COs	PSO1 (Software & System Development Skills)	PSO2 (Employability & Entrepreneurship Skills)	PSO3 (Professional & Ethical Responsibilities)
CO1		✓	✓
CO2	✓	✓	
CO3		✓	✓
CO4		✓	✓
CO5	✓	✓	✓

Program Educational Objectives (PEOs) Mapping

PEO1: To equip students with the ability to apply engineering knowledge, analyze problems, and develop solutions.

PEO2: To instill strong communication, leadership, and teamwork skills for corporate success.

PEO3: To develop ethical and socially responsible engineers who can contribute to sustainable growth.

PEO4: To encourage lifelong learning and adaptation to technological advancements.

COs	PEO1 (Engineering Knowledge & Problem Solving)	PEO2 (Communication & Teamwork)	PEO3 (Ethics & Sustainability)	PEO4 (Lifelong Learning & Career Growth)
CO1	✓	✓	✓	✓
CO2	✓			✓
CO3		✓	✓	✓
CO4		✓	✓	✓
CO5	✓	✓	✓	✓

3rd Year 5th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
1. THEORY									
1	ENGG	Major	CS 501	Artificial Intelligence	3	0	0	3	3
2	ENGG	Major	CS 502	Database Management Systems	3	0	0	3	3
3	ENGG	Major	CS 503	Object Oriented Programming using Java	3	0	0	3	3
4	ENGG	Major	CS 504 A CS 504 B CS 504 C	Compiler Design Cryptography and Network Security Computer Graphics	3	0	0	3	3
5	HUM	Minor	HU(CS)501	Economics for Engineers	2	0	0	2	2
2. PRACTICAL									
6	ENGG	Major	CS 591	Artificial Intelligence Lab	0	0	3	3	1.5
7	ENGG	Major	CS 592	Database Management Systems Lab	0	0	3	3	1.5
8	ENGG	Major	CS 593	Object Oriented Programming using Java Lab	0	0	3	3	1.5
9	PRJ	Internship	CS581	Internship	0	0	2	2	2
Total of Theory, Practical and Mandatory Activities/Courses								25	20.5

Course Name: Artificial Intelligence
Course Code: CS501
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite:
1. Data Structure
2. Design and Analysis of Algorithms
3. Statistics

Course Objective(s):

The objective of the course is to enable students to

- Comprehend the fundamental concepts of Knowledge Representation and Inferencing in Artificial Intelligence and its utilitarian importance in current technological context
- Formulate a problem as State-Space Exploration Framework or an Inferencing Framework of Artificial Intelligence.
- Use the strategies of AI-Heuristics to find acceptable solutions avoiding brute-force techniques.
- Design AI-Frameworks for Inferencing based on knowledge base.
- Analyze the effectiveness of an AI-Inferencing Model in offering solutions to the respective problem.

Course Outcomes(s):

CO1 To **Understand and explain** the fundamental concepts of Knowledge Representation and Inferencing in Artificial Intelligence and its utilitarian importance in current technological context for further exploration leading towards lifelong learning.

CO2 To **Identify** and **formulate** an engineering problem primarily to fit a State-Space Exploration Framework or an Inferencing Agent Design Framework within the scope of Artificial Intelligence paradigm.

CO3 To **Explore** relevant literature and **apply** the concept of Heuristic Techniques or Inferencing Models of Artificial Intelligence to **solve** problems.
CO4 To **Develop** Inferencing Models for **proposing** solutions to the problems of Artificial Intelligence.
CO5 To **Implement** Inferencing Models of Artificial Intelligence through **developing** feasible algorithms and **investigate** their effectiveness by **analyzing** their performances in solving the relevant problems.

Course Content:
Module-1: Introduction to Artificial Intelligence [2L]
Basic Concepts, History of Artificial Intelligence, Architecture of an Artificial Intelligent Agent, Applications of Artificial Intelligence

Module-2: Artificial Intelligence Problem Formulation as State-Space Exploration Problem for Goal Searching [8L]

Basic Concepts, State-Space Exploration Formulation for Water Jug Problem, Missionaries and Cannibals Problems, Farmer-Wolf-Goat-Cabbage Problem, 8-Puzzle Problem, Constraint Satisfaction Problem and Production System for Goal Searching.

Blind Search Techniques for Goal Searching: Breadth First Search, Depth First Search, Depth Limited Search, Iterative Deepening Search, Uniform Cost Search, Bi-directional Search.

Module-3: Heuristic Techniques for Goal Searching [8L]

Basic Concepts of Heuristic Techniques and Properties of Heuristic Functions, Hill Climbing Search. Best First Search, A* Search, AO* Search

Genetic Algorithm Based Evolutionary Search, Ant Colony Optimization, Particle Swarm Optimization.

Module-4: Adversarial Search for Game Playing [3L]

Basic Concepts, Minimax Search, Alpha-Beta Pruning.

Module-5: Knowledge Representation and Inference using Propositional Logic and Predicate Logic [5L]

Propositional Logic: Knowledge Representation and Inference using Propositional Logic

Predicate Logic: Knowledge Representation, Inference and Answer Extraction using First Order Predicate Logic

Module-6: Reasoning under Uncertainty [5L]

Bayesian Inferencing and Bayesian Belief Network, Dempster-Shafer Theory, Overview of Fuzzy Logic and Inferencing,

Module-7: Introduction to Natural Language Processing [2L]

Basic Concepts, Steps of Natural Language Processing, Morphological, Syntactic and Semantic Analysis, Discourse Integration and Pragmatic Analysis, Applications of Natural Language Processing.

Module-8: Introduction to Machine Learning [3L]

Basic concepts of Machine Learning Model, Supervised Learning, Unsupervised Learning, and Reinforced Learning, Overview of Artificial Neural Network

Textbook:

1. Russell, S. and Norvig, P. 2015. Artificial Intelligence - A Modern Approach, 3rd edition, Prentice Hall.
2. Rich, E., Knight, K and Shankar, B. 2009. Artificial Intelligence, 3rd edition, Tata McGraw Hill.

Reference Books:

1. Padhy, N.P. 2009. Artificial Intelligence and Intelligent Systems, Oxford University Press.
2. Deepak Khemani, “A First Course in Artificial Intelligence”, McGraw Hill.

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2										3			
CO2	2	3													
CO3	2	2	3	2											
CO4	2	2	2	3								2			
CO5	2	2	3	3								2			

Name of the Paper: Database Management System
Paper Code: CS502
Contact (Periods/Week): 3:0:0
Credit Point: 3
No. of Lectures: 36

Prerequisite:
1. Logic of programming language
2. Basic concepts of data structure and algorithms

Course Outcomes:
On successful completion of the learning sessions of the course, the learner will be able:

CS502.1: To Express the knowledge of data models.
CS502.2: To Implement the concept of designing an efficient relational database system.
CS502.3: To Correlate real world queries with database system.
CS502.4: To Illustrate transaction processing, concurrency control and recovery management of a database.
CS502.5: To Assess the internal storage structure to implement a proper database for an organization.

Module 1:
Introduction [3L]
Concept & Overview of DBMS, Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS.

Module 2:
Entity-Relationship and Relational Database Model [11L]
Basic concepts, Design Issues, Mapping Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended E-R features, case study on E-R Model. Structure of relational Databases, Relational Algebra, Relational Calculus, Extended Relational Algebra Operations, Views, Modifications of the Database.

Module 3:
SQL and Integrity Constraints [6L]
Concept of DDL, DML, DCL. Basic Structure, Set operations, Aggregate Functions, Null Values, Domain Constraints, Referential Integrity Constraints, assertions, views, Nested Subqueries, Database security application development using SQL, Stored procedures and triggers.

Module 4:
Relational Database Design [8L]
Functional Dependency, Different anomalies in designing a Database., Normalization using functional dependencies, Decomposition, Boyce-Codd Normal Form, 3NF, Normalization using multi-valued dependencies, 4NF, 5NF, Case Study

Module 5:
Internals of RDBMS [9L]
Physical data structures, Query optimization: join algorithm, statistics and cost bas optimization. Transaction processing, Concurrency control and Recovery Management: transaction model properties, state serializability, lock base protocols; two phase locking, Dead Lock handling.

Module 6:
File Organization & Index Structures [6L]
File & Record Concept, Placing file records on Disk, Fixed and Variable sized Records, Types of Single-Level Index (primary, secondary, clustering), Multilevel Indexes

Text Books:
1. Henry F. Korth and Silberschatz Abraham, "Database System Concepts", McGraw Hill.
2. Elmasri Ramez and Novathe Shamkant, "Fundamentals of Database Systems", Benjamin Cummings Publishing. Company.

Reference:
1. "Fundamentals of Database Systems", Ramez Elmasri, Shamkant B. Navathe, Addison Wesley Publishing.
2. Ramakrishnan: Database Management System, McGraw-Hill

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS502.1	3	3	3	3	3	2	-	-	3	-	-	2	3	2	2
CS502.2	3	3	3	2	3	2	-	-	2	-	-	1	3	2	2
CS502.3	3	3	3	3	3	-	-	-	2	-	-	2	3	1	1
CS502.4	2	2	2	2	3	1	-	-	2	-	-	1	3	1	2
CS502.5	3	2	2	2	3	1	-	-	1	-	-	1	3	2	2

Course Name: Object Oriented Programming using Java
Course Code: CS503
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite: Partial Object-Oriented Programming using C++

Course Outcomes:

CS503.1: To define the process of interaction between Objects and System w.r.t. Object Oriented Paradigm.
CS503.2: To summarize basic concepts of Object Orientation in Java Programming along with different properties and features.
CS503.3: To implement various string handling functions as well as basic I/O operations in object-oriented environment.
CS503.4: To explain basic code reusability concept w.r.t. Inheritance, Package and Interface.
CS503.5: To construct Java programs utilizing core object-oriented concepts, with a focus on Exception Handling, Multithreading, and Applet-based Web Programming.

Course Contents:

Module 1: Introduction [2L]
Object Oriented Analysis (OOA) & Object-Oriented Design (OOD) - Concepts of object-oriented programming language, Relationships among objects and classes-Generalization, Specialization, Aggregation, Association, Composition, links, Meta-class, Object Oriented Programming concepts - Difference between Java and C++; Different features of Java.

Module 2: Java Basics [10L]
Basic concepts of java programming - Advantages of java, Byte-code & JVM, Data types, Different types of Variables, Java Operators & Control statements, Java loops, Array, Creation of class, object, method, Constructor- Definition, Usage of Constructor, Different types of Constructors, finalize method and garbage collection, Method & Constructor overloading, this keyword, use of objects as parameter & methods returning objects, Call by value & call by reference, Static variables & methods, Nested & inner classes.

Module 3: Basic String handling & I/O [5L]
Basic string handling concepts- Concept of mutable and immutable string, Methods of String class, Methods of String buffer class, Command line arguments, basics of I/O operations – keyboard input using Buffered Reader, Scanner class in Java I/O operation.

Module 4: Inheritance and Java Packages [8L]
Inheritance - Definition, Advantages, Different types of inheritance and their implementation, Super and final keywords, super() method, Method overriding, Dynamic method dispatch, Abstract classes & methods, Interface - Definition, Use of Interface, Multiple inheritance by using Interface, Java Packages - Definition, Creation of packages, Java Access Modifiers - public, private, default and protected, Importing packages, member access for packages.

Module 5: Exception handling, Multithreading and Applet Programming [11L]
Exception handling - Basics, different types of exception classes. Difference between Checked & Unchecked Exception, Try & catch related case studies, Throw, throws & finally, Creation of user defined exception, Multithreading - Basics, main thread, Thread life cycle, Creation of multiple threads - yield(), suspend(), sleep(n), resume(), wait(), notify(), join(), is Alive(), Thread priorities, thread synchronization, Interthread communication, deadlocks for threads, Applet Programming - Basics, applet life cycle, difference between application & applet programming, Parameter passing in applets.

Textbooks:
1. Herbert Schildt – "Java: The Complete Reference" – 9th Ed. – TMH
2. E. Balagurusamy – "Programming with Java: A Primer" – 3rd Ed. – TMH.

Reference Books:
1. R.K. Das – "Core Java for Beginners" – VIKAS PUBLISHING.
2. Rambaugh, James Michael, Blaha – "Object Oriented Modelling and Design" – Prentice Hall, India.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS503.1	3	3	2	3	3	-	-	-	-	-	3	1	2	2	3
CS503.2	2	3	3	2	2	-	-	-	-	-	2	2	2	2	3
CS503.3	3	3	2	2	1	-	-	-	-	-	1	3	2	2	3
CS503.4	2	2	3	2	3	-	-	-	-	-	2	2	2	2	3
CS503.5	3	3	2	3	1	-	-	-	-	-	1	3	3	3	3

Course Name: Compiler Design
Course Code: CS504A
Contact: 3:0:0
Total Contact Hours: 36
Credit: 3

Prerequisites:
1. Mathematics
2. Concept of programming languages
3. Data structures
4. Computer architecture
5. Formal languages and automata theory
6. Some advanced math might be required if you adventure in code optimization

Course Objective(s):
- To make the student understand the process involved in a compiler
- To create an overall view of various types of translators, linkers, loaders, and phases of a compiler
- To understand the concepts of syntax analysis, various types of parsers especially the top-down approach
- To create awareness among students about various types of bottom-up parsers
- To understand the syntax analysis and, intermediate code generation, type checking, the role of symbol table and its organization, Code generation, machine in dependent code optimization and instruction scheduling

Course Outcomes:

On completion of the course students will be able

CS504A.1: To define compilers and various components of a compiler.
CS504A.2: To understand the roles of different phases of a compiler.
CS504A.3: To implement different algorithms for designing different phases of compilers.
CS504A.4: To build simple compilers using C programming languages.

Course Contents:

Module-1: Introduction to Compiler and Lexical Analyzer [7L]
Compilers, Cousins of the Compiler, Analysis-synthesis model, Phases of the compiler, Role of the lexical analyzer, Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of tokens, Finite automata, From a regular expression to an NFA, From a regular expression to DFA, Design of a lexical analyser generator (Lex).

Module-2: Syntax Analyzer and Semantic Analyzer [10L]
The role of a parser, Context free grammars, Writing a grammar, Top down Parsing, Non-recursive Predictive parsing (LL),Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR, Canonical LR), Parser generators (YACC),

Error Recovery strategies for different parsing techniques, Syntax directed translation: Syntax directed definitions, Construction of syntax trees, Bottom-up evaluation of S-attributed definitions, L-attributed definitions, Bottom-up evaluation of inherited attributes.

Module-3: Type Checker and Run-time Environment [7L]
Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions, Source language issues (Activation trees, Control stack, scope of declaration, Binding of names), Symbol tables, dynamic storage allocation techniques.

Module-4: Intermediate Language Generator [4L]
Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).

Module-5: Code Optimizer and Code Generator [8L]
Consideration for Optimization, scope of optimization, local optimization, loop optimization, folding, DAG representation, Flow Graph, Data flow equation, global optimization, redundant sub expression elimination, induction variable elimination, copy propagation, basic blocks & flow graphs, transformation of basic blocks, DAG representation of basic blocks, peep hole optimization, Object code forms, machine dependent code optimization, register allocation and assignment, generic code generation algorithms, DAG for register allocation.

Text Books:
1. Aho, A. V., Sethi, R., & Ullman, J. D. Addison - Wesley, 2007. Compilers-Principles, Techniques, and Tools.
2. Holub, A. I. (1990). Compiler design in C (Vol.5). Englewood Cliffs, N J: Prentice Hall.

Reference Books:
1. Chattopadhyay, S. (2005). Compiler Design. PHI Learning Pvt. Ltd.
2. Tremblay, J. P., & Sorenson, P.G. (1985). Theory and Practice of Compiler Writing. McGraw-Hill, Inc.
3. Appel, A.W. (2004). Modern compiler implementation in C. Cambridge university press.
4. Barrett, W.A., Bates, R.M., Gustafson, D.A., & Couch, J.D. (1986). Compiler construction: theory and practice. SRA School Group.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS504A.1	3	1	3	3	2	-	-	-	-	-	3	3	3	3	3
CS504A.2	3	3	3	2	1	-	-	-	-	-	2	1	2	3	2
CS504A.3	-	2	2	2	2	-	-	-	-	-	1	3	3	3	3
CS504A.4	2	3	1	3	1	-	-	-	-	-	1	3	3	3	3

Course Name: Cryptography and Network Security
Course Code: CS504B
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisites
1. Knowledge of Computer Networks and Operating Systems fundamentals
2. Understanding of Discreet Mathematics concepts

Course Objective(s):
- To impart concepts on cryptography and Network security
- To gain knowledge of the standard algorithms used to provide confidentiality, integrity, and authenticity
- To recognize the various key distribution and management systems for security of a cryptosystem

Course Outcomes:

After completion of course, students would be able

CS504B.1: To understand the basic concepts in cryptography
CS504B.2: To apply the deployment of different encryption techniques to secure messages in transit across data networks
CS504B.3: To discuss various techniques used to assure Integrity and Authentication
CS504B.4: To analyze diverse security measures and issues in practice.

Course Contents
Module-1: Introduction [7L]
Introduction - Services, Mechanisms, and Attacks, OSI security architecture, Network security model
Classical Encryption techniques (Symmetric cipher model, substitution techniques, transposition techniques, steganography)
Finite Fields and Number Theory: Groups, Rings, Fields, Modular arithmetic, Euclid's algorithm
Polynomial Arithmetic, Prime numbers, Fermat's and Euler's theorem
Testing for primality -The Chinese remainder theorem - Discrete logarithms

Module-2: Symmetric and Asymmetric Encryption Techniques [9L]
Data Encryption Standard- Block cipher principles, block cipher modes of operation
Advanced Encryption Standard (AES), Triple DES, Blowfish, RC5 algorithm
Public key cryptography: Principles of public key cryptosystems, The RSA algorithm
Key management - Diffie Hellman Key exchange, Elliptic curve arithmetic, Elliptic curve cryptography

Module-3: Authentication, Hashing, and Digital Signatures [6L]
Authentication requirement, Authentication function, MAC, Hash function

Security of hash function and MAC, MD5, SHA, HMAC, CMAC
Digital signature and authentication protocols, DSS, ElGamal, Schnorr

Module-4: Applied Network Security and Trusted Systems [7L]
Authentication applications, Kerberos, X.509
Internet Firewalls for Trusted System: Roles of Firewalls, Firewall related terminology- Types of Firewalls, Firewall designs principles
SET for E-Commerce Transactions
Intruder, Intrusion detection system
Virus and related threats, Countermeasures
Trusted systems, Practical implementation of cryptography and security

Module-5: Email, IP, and Web Security Protocols [7L]
E-mail Security: Security Services for E-mail-attacks possible through E-mail, Establishing keys privacy, authentication of the source
Message Integrity, Non-repudiation, Pretty Good Privacy, S/MIME
IP Security: Overview of IPSec, IPv4 and IPv6-Authentication Header, Encapsulation Security Payload (ESP)
Internet Key Exchange (Phases of IKE, ISAKMP/IKE Encoding)
Web Security: SSL/TLS Basic Protocol, computing the keys, client authentication
PKI as deployed by SSL Attacks fixed in v3, Exportability, Encoding, Secure Electronic Transaction

Text Books:
1. Kahate, A. (2013). Cryptography and network security. Tata McGraw-Hill Education.
2. Forouzan, B. A., & Mukhopadhyay, D. (2015). Cryptography and network security. New York, NY: Mc Graw Hill Education (India) Private Limited.

Reference Books:
1. Stallings, W. (2006). Cryptography and network security, 4/E. Pearson Education India.
2. Daras, N. J., & Rassias, M. T. (Eds.). (2015). Computation, cryptography, and network security (pp. 253-287). Springer.
3. Kumar, A., & Bose, S. (2017). Cryptography and network security. Pearson Education India.

CO-PO-PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS504B.1	3	1	2	1	1	-	-	-	3	2	1	1	3	1	1
CS504B.2	3	2	3	2	3	-	-	-	2	1	1	1	3	1	1
CS504B.3	1	3	2	3	2	-	-	-	2	3	1	1	3	1	3
CS504B.4	2	3	1	3	1	-	-	-	1	1	1	1	3	1	3

Course Name: Computer Graphics
Course Code: CS504C
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisites:
Mathematics, Computer Fundamentals &Principle of Computer Programming.

Course Outcomes
On successful completion of the learning sessions of the course, the learner will be able
CS504C.1: To remember the foundations of computer graphics and different display technology and devices.
CS504C.2: To analyze the concept of geometric, mathematical and algorithmic approach necessary for programming computer graphics.
CS504C.3: To explain clipping with the comprehension of windows, view-ports in relation to images display on screen.
CS504C.4: To experiment and compare different hidden surface illumination methods

Course Content:

Module - 1: Introduction to Computer Graphics [5L]
Overview of computer graphics, Basic Terminologies in Graphics, lookup table,3D viewing devices, Plotters, printers, digitizers, light pens etc., Active & Passive graphics, Computer graphics software. Light &Color models, Raster Scan and Random scan displays.

Module - 2: Scan Conversion [10L]
Points & lines, Line drawing algorithms; DDA algorithm, Bresenham's line algorithm, Midpoint Circle generation algorithm, Ellipse generating algorithm. Scan line polygon fill algorithm, boundary fill algorithm, flood fill algorithm.

Module - 3: 2D and 3D Transformation [8L]
Basic transformations: translation, rotation, scaling, Matrix representations & homogeneous coordinates, transformations between coordinate systems, reflection shear, Transformation of points, lines, parallel lines, intersecting lines.
3D transformations: translation, rotation, scaling.

Module - 4: 2D-Viewing, Clipping and Projection [8L]
Viewing pipeline, Window to viewport co-ordinate transformation. Clipping operations: Point clipping, Cohen Sutherland line clipping algorithm, Polygon clipping algorithm, Viewport clipping, Basic concepts of different type of projections.

Module - 5: Curve Generation and Hidden Surface Removal [5L]
Bezier curves, B-spline curves, Hidden Surface Removal: Z-buffer algorithm, Back face detection, BSP tree method, Painter's algorithm.

Text Books:

1. Hearn and P. M. Baker: Computer Graphics, 2nd ed. Prentice Hall of India, New Delhi, 1997.
2. W. M. Newman and R. F. Sproull: Principles of Interactive Computer Graphics, McGraw Hill, New Delhi, 1979.

Reference Books:

1. F. S. Hill: Computer Graphics, McMillan, New York, 1990.
2. D. P. Mukherjee: Fundamentals of Computer Graphics and Multimedia, Prentice Hall of India, New Delhi, 1999.
3. J. D. Foley et al.: Computer Graphics, 2nd ed., Addison-Wesley, Reading, Mass., 1993.
4. W. K. Giloi: Interactive Computer Graphics: Data Structure, Algorithms, Languages, Prentice Hall, Englewood Cliffs, 1978.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS504C.1	3	1	3	3	2	-	-	-	-	-	3	3	3	3	3
CS504C.2	3	3	3	2	1	-	-	-	-	-	2	1	2	3	2
CS504C.3	-	2	2	2	2	-	-	-	-	-	1	3	3	3	3
CS504C.4	2	3	1	3	1	-	-	-	-	-	1	3	3	3	3

Course Name: Economics for Engineers
Course Code: HU(CS)501
Contact: 2:0:0
Total Contact Hours: 24
Credits: 2

Pre-requisites:
MATH–College Algebra, Pre-Calculus Algebra and Trigonometry.

Course Outcomes:
On successful completion of the learning sessions of the course, the learner will be able to:
HU(CS)501: Apply the appropriate engineering economics analysis method(s) for problem solving: present worth, annual cost, rate-of-return, payback, break-even, benefit-cost ratio.
HU(CS)501: Evaluate the cost effectiveness of individual engineering projects using the methods learned and draw inferences for the investment decisions.
HU(CS)501: Compare the life cycle cost of multiple projects using the methods learned, and make a quantitative decision between alternate facilities and/or systems.
HU(CS)501: Evaluate the profit of a firm, carry out the break-even analysis and employ this tool to make production decision.
HU(CS)501: Discuss and solve advanced economic engineering analysis problems including taxation and inflation.

Course Contents:

Module 1: Introduction [3L]
Managerial Economics-Relationship with other disciplines-Firms: Types, Objectives and goals-Managerial Decisions-Decision Analysis.

Module 2: Demand and Supply Analysis [5L]
Demand - Types of demand – determinants of demand – Demand function – Demand Elasticity – Demand forecasting – Supply – Determinants of Supply - Supply Function - Supply Elasticity.

Module 3: Cost Analysis [5L]
Element of costs, Marginal cost, Marginal Revenue, Sunk cost, Opportunity cost, Break-even analysis –PV ratio.

Module 4: Elementary economic Analysis [4L]
Inflation-Meaning of inflation, types, causes, measures to control inflation.
National Income-Definition, Concepts of national income, Method of measuring national income.

Module 5: Financial Accounting [5L]
Concepts and Definition of Accounting, Journal, Ledger, Trial Balance. Trading A/C, Profit& Loss A/C and Balance Sheet.

Module 6: Investment Decision [2L]
Time value of money-Interest- Simple and compound, nominal and effective rate of interest, Cashflow diagrams, Principles of economic equivalence. Evaluation of engineering projects-Present worth method, Future worth method, Annual worth method, Internal rate of return method, Cost benefit analysis for public projects.

Textbooks:
1. Riggs, Bedworth, and, Randhwa, "Engineering Economics", McGraw Hill Education, India
2. Principles of Economics, Deviga Vengedasalam; Karunagaran Madhavan, Oxford University Press

Reference Books:
1. Engineering Economy by William G. Sullivan, Elin M. Wicks, C. Patric Koelling, Pearson
2. R. Paneer Seelvan, "Engineering Economics", PHI
3. Ahuja, H. L., "Principles of Micro Economics", S. Chand & Company Ltd
4. Jhingan, M. L., "Macro Economic Theory"
5. Macro Economics by S. P. Gupta, TMH
6. Haniff and Mukherjee, Modern Accounting, Vol-1, TMG
7. Modern Economic Theory –K. K. Dewett (S. Chand)

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
HU(CS)501.1	-	-	-	-	-	-	-	-	-	-	-	3	-	-	-
HU(CS)501.2	-	-	-	3	-	3	-	-	-	-	-	-	-	-	-
HU(CS)501.3	-	-	2	-	-	-	-	-	-	-	-	2	-	-	-
HU(CS)501.4	-	-	-	-	-	-	-	-	2	-	-	-	-	-	-
HU(CS)501.5	-	-	-	-	-	-	-	-	-	-	-	2	-	-	-

Course Name: Artificial Intelligence Lab
Course Code: CS591
Contact: 0:0:3
Total Contact Hours: 36
Credits: 1.5

Prerequisite: Data Structure, Design and Analysis of Algorithms, Statistics

Course Objective(s):
The objective of the course is to enable students to -

- Gain foundational knowledge of PROLOG to implement an Artificial Intelligent Agent as an executable computer program for Knowledge Representation and Inferencing.
- Formulate a problem by analyzing its characteristics to fit a State-Space Exploration Frame work or an Inferencing Frame work of Artificial Intelligence.
- Apply the concepts of Artificial Intelligence to solve a problem by implementing well-known Artificial Intelligence strategies using proper techniques and tools of PROLOG.
- Build expert systems offering solutions to the challenging problems of Artificial Intelligence.
- Implement Artificial Intelligence based ideas as executable PROLOG programs through developing intelligent heuristic strategies.

Course Outcomes:
On completion of the course students will be able

CS591.1: To understand the concept of simple programming using PROLOG/ LISP.
CS591.2: To understand the concept of AI based programs using PROLOG/ LISP.
CS591.3: To apply logic-based techniques in various real life problem domains.
CS591.4: To analyze logic-based techniques in various real life problem domains.

Course Content:
WEEK-1: Introduction to PROLOG Programming along with the IDE and its Basic Components
Assignments for understanding the Basic Components of Knowledge Representation and Inferencing in Artificial Intelligence using PROLOG Programming and its working strategy.

WEEK -2: Arithmetic, Boolean Expression, Decision Making Strategies
Assignments for understanding implementation of Arithmetic Expression, Boolean Expression, and Decision-Making Strategies.

WEEK -3: Recursion and Looping through Recursion
Assignments for understanding implementation of Recursion and Looping through Recursion.

WEEK -4: List of Data Items in PROLOG
Assignments for understanding the utility of List in solving various problems.

WEEK -5: Blind Search Techniques – BFS, DFS

Implementation of BFS and DFS Algorithms for Goal Searching to solve Puzzles (8-Puzzle, Water Jug Puzzle)

WEEK -6: Heuristic Search Techniques – A* Search
Implementation of A* Search Algorithm for Goal Searching to solve Puzzles (8-Puzzle, Route Finding Puzzle)

WEEK-7: Constraint Satisfaction Problem Solving
Implementation of Backtracking Strategies to solve Constraint Satisfaction Problems (Graph Colouring Problem, 8-Queens Problem)

WEEK -8: Game Playing
Implementation of Adversarial Search Algorithm with alpha-beta pruning strategy for Game Playing (Tic-Tac-Toe)

WEEK -9: Discussion on Project Problems and Allocation (Problem Description Report Submission)

WEEK -10: Designing Solution Model and Proposal Report Submission

WEEK -11: Project Implementation, Verification and Documentation

WEEK -12: Project Demonstration and Project Report Review

Textbook:
1. Ivan Bratko, Prolog Programming for Artificial Intelligence, 4th Edition, Addison-Wesley
2. Russell, S. and Norvig, P. 2015. Artificial Intelligence - A Modern Approach, 3rd edition, Prentice Hall.
3. Rich, E., Knight, K and Shankar, B. 2009. Artificial Intelligence, 3rd edition, Tata McGraw-Hill.

Reference Books:
1. Padhy, N.P. 2009. Artificial Intelligence and Intelligent Systems, Oxford University Press.
2. Deepak Khemani, "A First Course in Artificial Intelligence", McGraw Hill.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS591.1	3	1	3	3	2	-	-	-	-	-	3	3	2	1	2
CS591.2	3	2	3	2	2	-	-	-	-	-	2	3	2	2	3
CS591.3	-	3	-	2	2	-	-	-	-	-	1	3	2	3	-
CS591.4	2	2	3	2	3	-	-	-	-	-	2	2	3	-	1

Course Name: Database Management System Lab
Course Code: CS592
Contact: 0:0:3
Total Contact Hours: 36
Credits: 1.5

Prerequisite:
1. Logic of programming language
2. Basic concepts of data structure and algorithms

Course Objective(s):
- To learn the data models, conceptualize and depict a data base system.
- To learn the fundamental concepts of SQL queries.
- To understand the concept of designing a database with the necessary attributes.
- To know the methodology of Accessing, Modifying and Updating data & information from the relational databases.
- To learn database design as well as to design user interface and how to connect with database.

Course Outcomes:

On completion of the course students will be able to

CS592.1: Understand the basic concepts regarding database, know about query processing and techniques involved in query optimization and understand the concepts of database transaction and related database facilities including concurrency control, backup and recovery.
CS592.2: Understand the introductory concepts of some advanced topics in data management like distributed databases, data warehousing, deductive databases and be aware of some advanced databases like partial multimedia and mobile databases.
CS592.3: Differentiate between DBMS and advanced DBMS and use of advanced database concepts and become proficient in creating database queries.
CS592.4: Analyze database system concepts and apply normalization to the database.
CS592.5: Apply and create different transaction processing and concurrency control applications.

Module 1:
Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)

Module 2:
Converting ER Model to Relational Model (Represent entities and relationships in Tabular form, represent attributes as columns, identifying keys) and apply the normalization techniques.

Module 3:
Creation of Tables using SQL- Overview of using SQL tool, Data types in SQL, Creating Tables (along with Primary and Foreign keys), Altering Tables and Dropping Tables

Module 4:
Practicing DML commands- Insert, Select, Update, Delete

Module 5:
Practicing Queries using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSECT, CONSTRAINTS etc., Practicing Sub queries (Nested, Correlated) and Joins (Inner, Outer and Equi).

Module 6:
Practice Queries using COUNT, SUM, AVG, MAX, MIN, GROUP BY, HAVING, VIEWS Creation and Dropping, Practicing on Triggers - creation of trigger, Insertion using trigger, Deletion using trigger, Updating using trigger

Module 7:
Procedures- Creation of Stored Procedures, Execution of Procedure, and Modification of Procedure, PL/SQL, Cursors- Declaring Cursor, Opening Cursor, Fetching the data, closing the cursor.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS592.1	3	3	3	3	3	2	-	-	2	-	-	2	3	2	2
CS592.2	3	3	3	2	3	2	-	-	2	-	-	1	3	2	2
CS592.3	3	3	3	3	3	-	-	-	2	-	-	2	3	1	1
CS592.4	2	2	2	2	3	1	-	-	2	-	-	1	3	1	2
CS592.5	3	2	2	2	3	1	-	-	1	-	-	1	3	2	2

Course Name: Object Oriented Programming using Java Lab
Course Code: CS593
Contact: 0:0:3
Total Contact Hours: 36
Credits: 1.5

Prerequisites:
1. Computer Fundamentals
2. Basic understanding of Computer Programming and related Programming Paradigms
3. Problem Solving Techniques with proper logic Implementation.

Course Objective(s):
- It demonstrates that how can you change the implementation of an object without affecting any other code by increasing data security and protecting unwanted data access. (Encapsulation).
- It allows you to have many different functions, all with the same name, all doing the same job, but depending upon different data. (Polymorphism).
- It guides you to write generic code: which will work with a range of data, so you don't have to write basic stuff over, and over again. (Generics).
- It lets you write a set of functions, then expand them in different direction without changing or copying them in any way. (Inheritance)

Course Outcomes:

On completion of the course students will be able

CS593.1: To describe the object-oriented approach in Java by outlining the relationship between classes, objects, and constructors.
CS593.2: To apply the concept of code reusability in Java through the use of inheritance and class hierarchies.
CS593.3: To implement Java programs using encapsulation, polymorphism, and relevant object-oriented keywords.
CS593.4: To analyze object-oriented features such as data abstraction, packages, and interfaces to determine their roles in Java programming.
CS593.5: To construct Java applications using exception handling, multithreading, and applet-based web programming techniques.

Course Contents:
Week – 1:
Basic Java programs such as
i. printing “Hello, GNIT”.
ii. checking whether a number is even or odd.
iii. finding out the roots of a quadratic equation.
iv. finding out the factorial of a given number.
v. printing fibonacci series upto n terms.

vi. Creating a class calculator that has 4 methods like add, sub, mul & div. Then doing the addition, subtraction, multiplication, and division of 2 integer numbers using these 4 methods.

Week – 2:
Java programs to implement default constructor, parameterized constructor using command line argument, 'this' keyword.

Week – 3:
Java programs to implement method overloading, constructor overloading, call by value, call by reference, recursion.

Week – 4:
Java programs to implement the difference between public and private access specifier, 'static' keyword, inner class.

Week – 5:
Java programs to implement simple inheritance, hierarchical inheritance, and multilevel inheritance.

Week – 6:
Java programs to implement 'super' keyword to access a superclass member, 'super' keyword to access a superclass constructor, method overriding.

Week – 7:
Java programs to implement run-time polymorphism, abstract class and method.

Week – 8:
Java programs to implement interface, multiple inheritance.

Week – 9:
Java programs to

i. create two user-defined packages pkg1 and pkg2 and import both to another program which is outside the packages.
ii. create multiple packages containing classes with identical names.
iii. show how a protected variable of one package can be accessed in a subclass in another package.
iv. show how to add multiple public classes to a single package.

Week – 10:
Java programs to implement ArithmeticException, ArrayIndexOutOfBoundsException, 'throw' and 'throws' keywords, finally block.

Week – 11:
Java programs to

i. create 3 threads - the 1st thread to display GOOD MORNING for every 1 second, the 2nd thread to display HELLO for every 2 seconds and the 3rd thread to display WELCOME for every 3 seconds.

ii. implement the above program by assigning priorities to the created threads such that the 1st thread executes first followed by the 2nd thread and lastly the 3rd thread.

Week – 12:

Java programs to

i. develop an applet that display simple message.
ii. develop an applet that will add two integer numbers.
iii. develop an applet that will draw lines, rectangle and oval.

Textbooks:

1. Herbert Schildt – "Java: The Complete Reference " – 9th Ed. – TMH
2. E. Balagurusamy – " Programming with Java: A Primer " – 3rd Ed. – TMH.

Reference Books:

1. R. K. Das – " Core Java for Beginners " – VIKAS PUBLISHING.
2. Rambaugh, James Michael, Blaha – " Object Oriented Modelling and Design " – Prentice Hall, India

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS593.1	2	2	3	3	2	-	-	-	-	-	3	1	2	2	3
CS593.2	1	3	2	2	3	-	-	-	-	-	1	2	2	2	3
CS593.3	2	-	1	2	2	-	-	-	-	-	2	3	2	2	3
CS593.4	3	2	3	2	3	-	-	-	-	-	2	2	2	2	3
CS593.5	2	3	1	3	2	-	-	-	-	-	1	3	3	3	3

3rd Year 6th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
C. THEORY									
1	ENGG	Major	CS 601	Web and Internet Technology	3	0	0	3	3
2	ENGG	Major	CS 602	Machine Learning	3	1	0	4	4
3	ENGG	Major	CS 603	Software Engineering	3	0	0	3	3
4	ENGG	Major	CS 604 A CS 604 B CS 604 C	Mobile Computing Natural Language Processing Cloud Computing	3	0	0	3	3
5	ENGG	Minor	CS 605	Cyber Law and Ethics	3	0	0	3	3
D. PRACTICAL									
6	ENGG	Major	CS 691	Web and Internet Technology Lab	0	0	3	3	1.5
7	ENGG	Major	CS 692	Machine Learning Lab	0	0	3	3	1.5
8	ENGG	Major	CS 693	Software Engineering Lab	0	0	3	3	1.5
Total of Theory, Practical and Mandatory Activities/Courses								25	20.5

Course Name: Web and Internet Technology
Course Code: CS601
Contact (Periods/Week): 3:0:0
Credit Point: 3
No. of Lectures: 36

Prerequisite: Computer Networks

Course Objective(s):
The objective of the course is to enable students

- To impart the design, development and implementation of Static and Dynamic Web Pages.
- To develop programs for Web using Scripting Languages.
- To give an overview of Server-Side Programming in Web.

Course Outcomes(s):
After completion of the course students will be able to

CO1: To define the concepts of World Wide Web (www), Internet, HTTP Protocol, Web Browsers, Client-Server etc.
CO2: To summarize interactive web pages using HTML, DHTML and CSS.
CO3: To implement the knowledge of different information interchange formats like XML.
CO4: To explain web applications using scripting languages like JavaScript, CGI, PHP.
CO5: To write different server-side programming like Servlet, JSP.

Course Content:

Module 1-Introduction to Web and Internet Technology [5L]:
Concept of World Wide Web (www), Internet and the relation with www [1L]; The Internet - Basic Internet Protocols, HTTP Protocol - Request and Response, Web browser [1L]; Web clients and Web servers, Dynamic IP [1L]; Clients, Servers, and Communication, Web site design principles, Planning the site and navigation [2L].

Module -2: Web Designing [12L]
HTML, DHTML & CSS [8L]: Introduction, Elements, Attributes, Heading, Paragraph. Formatting [2L]; Link, Table, List, Block, Layout, Html Forms and input [1L]; IFrame, Colors, Image Maps and attributes of image area [2L]; Introduction to CSS, basic syntax and structure of CSS, different types- internal, external and inline CSS [1L]; Basic Introduction of DHTML, Difference between HTML and DHTML, Documentary Object Model (DOM) [2L].
Extended Markup Language (XML) [4L]: Introduction, Difference between HTML & XML, XML Tree [2L]; Syntax, Elements, Attributes, Validation and parsing, DTD [2L].

Module 3- Web Scripting [9L]
Java Scripts [4L]: Basic Introduction, Statements, comments, variable, operators, data types[1L]; condition, switch, loop, break [1L]; Java script functions, objects, and events[2L].

CGI Scripts [1L]: Introduction, Environment Variable, GET and POST Methods.

PHP Scripting [4L]: Introduction, Syntax, Variables, Output, Data types, String, Constants [1L]; Operator, Decision Control statements [1L]; switch-case, Loop, PHP function[1L]; array, Form Handling[1L].

Module 4 - JSP and Servlet [11L]:
Java Server Page (JSP) [8L]: JSP Architecture [1L]; JSP Servers, JSP Life Cycle [1L]; Understanding the layout of JSP, JSP Script-let Tag [1L]; JSP implicit object (request and response) [1L]; Variable declaration, methods in JSP [1L]; JSP directive (Tag-lib and include), JavaBean- inserting JavaBean in JSP [1L]; JSP Action tags (Forward & Include) [1L]; Creating ODBC data source name, Introduction to JDBC, prepared statement and callable statement [1L].
Java Servlet [3L]: Servlet environment and role, Servlet life cycle [1L]; Servlet methods-Request, Response, Get and post [1L]; Cookies and Session [1L].

Text book:
1. "Web Technology: A Developer's Perspective", N.P. Gopalan and J. Akilandeswari, PHI Learning, Delhi, 2013. (Topics covered: html, CSS, imagemap, xml)
2. "Learning PHP, MySQL & JavaScript", Robin Nixon, O'Reilly Publication. (Topics covered: PHP, Java Script)
3. "Head First Servlet's & JSP", Bryan Basham, Kathy Sterra, Bert Bates, O'Reilly Publication. (Topics covered: Servlet, JSP)

Reference
1. "Programming the World Wide Web", Robert. W. Sebesta, Fourth Edition, Pearson Education, 2007.
2. "Core Web Programming"- Second Edition-Volume I and II, Marty Hall and Larry Brown, Pearson Education, 2001.

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	3	3	2	-	-	-	-	-	3	3	2	1	2
CO2	3	2	3	2	2	-	-	-	-	-	2	2	2	2	3
CO3	2	3	2	2	2	-	-	-	-	-	1	1	2	3	2
CO4	2	2	3	2	3	-	-	-	-	-	2	-	3	3	1
CO5	2	3	1	3	1	-	-	-	-	-	1	1	2	2	2

Course Name: Machine Learning
Course Code: CS602
Contact (Periods/Week): 3:1:0
Credit Point: 4
No. of Lectures: 48

Prerequisite:

1. Basic programming skills (preferably Python or Java)
2. Algorithm design and analysis
3. Probability: axioms, conditional probability, common distributions (Bernoulli, Binomial, Gaussian, etc.), expectation, variance, Bayes' rule
4. Linear algebra: vectors, matrices, eigenvalues/eigenvectors
5. Basic calculus and statistics
6. Data structures (arrays, lists, trees, graphs)

Course Objectives

1. To introduce the fundamental concepts of supervised and unsupervised learning, including real-world applications, model training workflows, data preprocessing, and evaluation metrics.

2. To provide practical understanding of clustering techniques, dimensionality reduction, and anomaly detection for unsupervised learning and pattern discovery.

3. To familiarize students with model evaluation strategies, cross-validation, and the theoretical concepts of bias-variance trade-off for robust model development.

4. To develop foundational knowledge in neural networks and deep learning architectures, including sequence modeling, transfer learning, and representation learning.

5. To explore scalable and modern machine learning paradigms such as online, semi-supervised, and reinforcement learning, along with probabilistic graphical models and distributed ML.

6. To expose students to recent trends in machine learning including AutoML, federated learning, interpretability, and the ethical aspects of AI systems.

Course Outcomes (COs)

CO1. Understand the fundamental concepts and methods of machine learning
CO2. Learn to design and analyze various machine learning algorithms
CO3. Explore supervised and unsupervised learning paradigms
CO4. Gain practical experience with machine learning tools and libraries
CO5. Develop the ability to apply machine learning techniques to solve real-world problems

Course Content

Module 1: [10L]
Supervised learning basics, Types of machine learning, Real-world applications, ML workflow

and pipeline, Data preprocessing and feature engineering, Train-test-validation split, Overfitting and regularization, Evaluation metrics

Module 2: [8L]
Clustering fundamentals, K-means algorithm, Hierarchical clustering, Dimensionality reduction (PCA), Anomaly detection

Module 3: [6L]
Model evaluation strategies, Cross-validation techniques, Confusion matrix and ROC, Bias-variance trade-off

Module 4: [10L]
Introduction to neural networks, Perceptron and activation functions, Feed-forward architectures, Deep learning overview, Feature representation learning, Sequence and time-series modeling, Transfer learning basics

Module 5: [8L]
Scalable machine learning, Online learning concepts, Distributed computing in ML, Semi-supervised learning, Active learning, Reinforcement learning introduction, Probabilistic graphical models

Module 6: [6L]
Recent trends in ML, Federated and privacy-preserving learning, AutoML and hyperparameter tuning, Interpretability and ethical AI

Textbook:
1. Kevin P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2012
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer
3. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow* (2 e), O'Reilly – added for its practical, beginner-friendly approach.

Reference
1. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2007
2. Andriy Burkov, *The Hundred-Page Machine Learning Book*, 2019 – concise overview for quick revision.

CO–PO-PSO Mapping:

CO/PO	PO1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 10	PO1 11	PO1 2	PSO 1	PSO 2	PS O3
CS602.1	3	2	3	2	2	–	–	–	–	–	–	2	1	1	–
CS602.2	3	3	2	–	–	–	–	–	–	–	1	1	–	1	–
CS602.3	2	3	3	2	–	–	–	–	–	–	–	–	1	–	–
CS602.4	2	2	3	3	2	–	–	–	–	–	2	3	–	–	–

CS602.5	2	2	2	3	1	1	–	–	–	2	–	1	–	–	–

Course Name: Software Engineering
Course Code: CS603
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisites: Programming for Problem Solving

Course Objectives

- **Understand Software Development Life Cycles (SDLCs):** To equip students with knowledge of different SDLC models (e.g., Waterfall, Agile, Spiral) and their appropriate applications in software project planning and management.
- **Master Requirements Engineering:** To develop skills in gathering, analyzing, documenting, and validating software requirements using structured and systematic approaches.
- **Learn Software Design Principles:** To introduce students to architectural and design principles, including modularity, abstraction, design patterns, and modeling with UML.
- **Apply Software Testing and Quality Assurance Techniques:** To train students in verification, validation, and various software testing methodologies (unit, integration, system, acceptance testing) to ensure software reliability and quality.
- **Understand Project Management and Maintenance:** To provide knowledge of software project management practices, including estimation, scheduling, risk management, and maintenance activities post-deployment.

Course Outcomes:

On completion of the course students will be able

CO1: To identify software engineering problems, including specification, design, implementation, and testing of software systems that meet performance and quality assurance

CO2: To gather software requirements through a productive working relationship with various stakeholders of the project

CO3: To prepare solutions in one or more application domains using software engineering approaches that integrates ethical, social, legal and economic concerns.

CO4: To validate the code from the design and effectively apply relevant standards and perform testing, and quality management and practice.

CO5: To build modern engineering tools necessary for project management, software reuse and maintenance.

Course Content:

Module-1: [6L]

Introduction: Software Engineering, Characteristics, Components, Application, Definitions. Software Project Planning-Feasibility Analysis, Technical Feasibility, Cost-Benefit Analysis, Basics of estimation: COCOMO (Basic, intermediate, Complete) model.

Module- 2: [6L]

Evolution and impact of Software engineering, software life cycle models: Waterfall, prototyping, Evolutionary, and Spiral models. Feasibility study, Functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification.

Module -3: [8L]

Basic issues in software design, modularity, cohesion, coupling and layering, function-oriented software design: DFD and Structure chart, object modeling using UML, Object-oriented software development, user interface design. Coding standards and Code review techniques.

Module -4: [7L]

Fundamentals of testing, White-box, and black-box testing, Test coverage analysis and test case design techniques, mutation testing, Static and dynamic analysis, Software reliability metrics, reliability growth modeling.

Module -5: [9L]

Software project management, Project planning and control, cost estimation, project scheduling using PERT and GANTT charts, cost-time relations: Rayleigh-Norden results, quality management, ISO and SEI CMMI, PSP and Six Sigma. Computer aided software engineering, software maintenance, software reuse, Component-based software development.

Text Books:

1. Fundamentals of Software Engineering by Rajib Mall, –PHI-3rd Edition, 2009.
2.Software Engineering-Pankaj Jalote (Wiley-India)

Reference

1.Software Engineering–Agarwal and Agarwal (PHI)
2.Software Engineering, by Ian Sommerville, Pearson Education Inc., New Delhi, (2009).
3. Software Engineering: A Practitioner’s Approach”, by Roger S. Pressman, McGraw-Hill. (2005)

CO–PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	2	1	1	-	3	2	2	2	3	3	3
CO2	3	3	2	3	3	1	1	-	3	3	2	3	2	3	2
CO3	2	3	2	3	3	-	2	3	3	2	2	1	3	3	3
CO4	2	3	2	3	2	-	1	-	3	2	2	2	3	3	3
CO5	3	2	3	3	3	1	1	-	3	1	3	-	2	3	2

Course Name: Mobile Computing
Course Code: CS604A
Contact: 3:0:0
Total Contact Hours: 36
Credit: 3

Prerequisite: Basic understanding of computer networks and digital communication.

Course Objective(s):

- Introduce fundamental concepts and principles of mobile computing.
- Explain wireless communication and networking principles supporting cellular, wireless LANs, and ad hoc networks.
- Understand localization, routing, energy management, and security in mobile systems.

Course Outcome(s):
On completion of the course, students will be able to:

- **CO1:** Analyze wireless communication systems and mobile technologies.
- **CO2:** Apply routing algorithms in infrastructure-based and ad hoc networks.
- **CO3:** Model bandwidth and mobility management in cellular networks.
- **CO4:** Design energy-efficient and secure communication protocols in mobile environments.
- **CO5:** Evaluate recent developments in mobile computing and wireless communication.

Course Content:
Module I: Introduction [6L]
Evolution of mobile communication systems; wireless vs. wired communication. Architecture of cellular networks. Key features of GSM, CDMA, LTE, and 5G. Cellular architecture, Mobility management: handoff types (hard, soft). Bandwidth and energy considerations. Security challenges in mobile environments, Generational evolution from 1G to 5G.

Module II: Wireless Data Communication [8L]
IEEE 802.11: architecture, protocol layers, QoS, roaming support. Bluetooth: piconets, scatternets, protocol stack (L2CAP, RFCOMM). Initialization and self-organization in ad hoc networks.

Leader election and routing basics. Energy-efficient MAC protocols. Basic wireless security models, Power consumption in mobile devices. Physical layer: transmission power control. MAC layer: sleep scheduling, contention resolution. Network layer: energy-aware routing. Application layer strategies: data compression and adaptive transmission. Energy-QoS trade-offs.

Module III: Mobility Management [8L]
GSM, Location management techniques: location update, paging. Call establishment in mobile networks. Mobile IP: agent discovery, registration, tunneling. GPRS components and working. Handoff decision strategies. Mobility models: Random Walk, Random Waypoint, Group mobility and their use in simulations.

Module IV: Bandwidth Management [4L]
Frequency spectrum utilization. Channel Assignment Problem (CAP): fixed and dynamic assignment. Graph-based modeling: node and edge coloring techniques. Benchmark instances and theoretical lower bounds. CAP optimization goals and trade-offs.

Module V: Node Localization [4L]
Fundamentals of localization in wireless networks. Outdoor vs. indoor localization challenges. Techniques: Time of Arrival (TOA), Angle of Arrival (AOA), triangulation. Effects of LOS/NLOS signals on accuracy. Beacon-based positioning. Error modeling and mitigation.

Module VI: Ad Hoc Network Communication [6L]
Issues in decentralized communication. MAC protocols for collision avoidance: slotted ALOHA, CSMA/CA. Deterministic scheduling methods. Time slot allocation techniques. Routing protocol classifications: proactive (DSDV), reactive (DSR, AODV), hybrid (ZRP, TORA). Metrics for routing performance.

Text books:
- K. Sinha, S. Ghosh and B. P. Sinha, Wireless Networks and Mobile Computing. CRC Press: New York, 2015.
- J. Schiller, Mobile Communication, Pearson
- Yi-Bing Lin & Imrich Chlamtac, Wireless and Mobile Networks Architectures, John Wiley & Sons, 2001
- Raj Pandya, Mobile and Personal Communication systems and services, Prentice Hall of India, 2001
- Xiang Yang Li, Wireless Adhoc and Sensor Networks, Cambridge University Press.

Reference
- Research articles published on secure wireless communication (authentication, mitigation of DoS, DDoS, eavesdropping) published in leading journals.
- Mark Ciampa, Guide to Designing and Implementing wireless LANs, Thomson learning, Vikas Publishing House, 2001.

- P. Stavronlakis, Third Generation Mobile Telecommunication systems, Springer Publishers.

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	1	1	1	--	--	--	--	1	3	3	2
CO2	3	3	2	2	1	2	1	--	--	--	--	1	3	2	2
CO3	3	2	3	3	2	2	1	1	--	--	--	--	3	3	2
CO4	3	3	3	2	3	2	1	--	--	--	--	1	3	3	3
CO5	2	3	2	3	1	1	--	1	--	--	--	--	3	2	3

Course Name: Natural Language Processing
Course Code: CS604B
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite:

1. Solid background in Linear algebra, Probability and Statistics, Artificial Intelligence and Neural Networks.
2. Good Exposure of Python packages.

Course Objective(s)

1. To explore Text Data using various industry standard tools.
2. To explore the Feature Engineering for Text Representation
3. To build the model for Clustering and Classifying Text
4. To implement Machine Learning and Deep Learning techniques.

Course Outcome(s)

On completion of the course, students will be able to

CO1. Understand the basic concepts of text data using various industry standard tools.
CO2. Understand the techniques to do the Feature Engineering for Text Representation.
CO3. Understand the approaches to build models, Clustering and Classifying Text.
CO4. Understand the different techniques in Machine learning and Deep Learning.

Course Content:

Module 1: Natural Language Processing Basics [9L]
What is Natural Language Processing? Different Phases of Natural Language Processing; Linguistics: Language Syntax and Structure, Words, Phrases, Clauses, Grammar, Dependency Grammar, Constituency Grammar, Word-Order Typology; Lemmas and Word forms, Homonyms, Homographs and Homophones, Heteronyms and Heterographs, Polysemes, Capitonyms, Synonyms and Antonyms, Hyponyms and Hypernyms, Stemming and Lemmatization; Representation of Semantics; Text Corpora: Corpora Annotation and Utilities, Accessing Popular Corpora; Parts of Speech Tagging: Training and Building POS Taggers; HMM Part-of-Speech Tagging; NER-Tagging; Relationship Extraction, Temporal Information Extraction, Event Extraction, Template Filling; Conditional Random Fields (CRFs); Shallow Parsing, Chunking; Building Dependency and Constituency Parsers, Application of NLP.

Module 2: Feature Engineering for Text Representation [9L]
Pre-processing the Text Corpus;N-gram Language Models, Smoothing; Traditional Feature Engineering Models; Extracting Features for New Documents; Topic Models in Gensim, LDA, LSI, Hierarchical Dirichlet process; Advanced Feature Engineering Models, Word Embedding, Word2Vec Model, The Continuous Bag of Words (CBOW) Model, The Skip-Gram Model; Semantic Analysis: Exploring WordNet, Understanding Synsets, Analyzing Lexical Semantic Relationships, Semantic Relationships and Similarity, Word Sense Disambiguation.

Module 3: Clustering and Classifying Text [9L]
Clustering text: Text Similarity, Analyzing Term Similarity, Analyzing Document Similarity; Classifying text: Classification Models, Evaluating Classification Models, Building and Evaluating of the Text Classifier; Sentiment Analysis: Text Pre-processing and Normalization, Unsupervised Lexicon-Based Models, Classifying Sentiment with Supervised Learning, Text Summarization, Question & Answering

Module 4: Deep Learning Architectures for Sequence Processing [9L]
Language Models Revisited; Getting words in order with convolutional neural networks (CNNs), Recurrent Neural Networks, Stacked and Bidirectional RNNs; LSTMs and GRUs; Attention, Transformers; Encoder-Decoder Model, Machine Translation; Beam Search; Text Classification using CNNs and LSTM; Chatbots

Text Books:
1. Bhargav Srinivasa-Desikan, “Natural Language Processing and Computational Linguistics", Packt Publishing
2. Dipanjan Sarkar, “Text Analytics with Python", Apress, ISBN-13 (pbk): 978-1-4842-4353-4
3. Daniel Jurafsky, James H. Martin, "Speech and Language Processing", Pearson Education India, Third Edition.
4. Sumit Raj, "Building Chatbots with Python", Apress, ISBN-13 (pbk): 978-1-4842-4095-3

Reference
1. Francois Chollet, “Deep Learning with Python”, Manning Publications; 1st edition
2. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, and Harshit Surana, "Practical Natural

Language Processing", Oreily
3. Hobson Lane, Cole Howard, Hannes Max Hapke, "Natural Language Processing in Action",Manning Publications

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	2	-	-	1	-	-	1	-	-	1	1	3	2
CO2	3	3	2	2	-	-	-	-	-	-	-	-	-	2	3
CO3	3	3	3	3	-	-	2	-	1	-	2	2	3	3	3
CO4	2	3	-	3	1	-	2	-	-	-	-	-	3	2	3
CO5	2	3	1	2	-	-	-	2	-	1	1	-	3	3	3

Course Name: Cloud Computing
Course Code: CS604C
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite

1. Should have the basic knowledge of Operating Systems.
2. Should be aware of the fundamental concepts of Networking.
3. Should have knowledge of heterogeneous systems and resource management.

Course Objective(s):

- To learn the workflow of cloud business model and optimized resource allocation.
- To gain knowledge of cloud service and delivery models and develop an understanding of the risk and compliance responsibilities and Challenges for each Cloud type and service delivery model.
- To learn virtualization techniques, load balancing, and work strategy of different cloud infrastructure.
- To know the security and privacy issues in cloud infrastructure

Course Outcomes:
After completion of course, students would be able:

CS604C.1: To identify the business model concepts, architecture and infrastructure of cloud computing, including cloud service models and deployment models.
CS604C.2: To journaling some important cloud computing driven commercial systems such as Google Apps, Microsoft Azure and Amazon Web Services and other business applications

CS604C.3: To articulate and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms.
CS604C.4: To categorize the core issues of cloud computing such as security, privacy, interoperability, and its impact on cloud application.

Course Contents:
Module 1: Definition of Cloud Computing and its Basics [8L]
Definition of Cloud Computing: Defining a Cloud, Cloud Types – NIST model, Cloud Cube model, Deployment models (Public, Private, Hybrid and Community Clouds), Service models – Infrastructure as a Service, Platform as a Service, Software as a Service with examples of services/ service providers, Cloud Reference model, Characteristics of Cloud Computing – a shift in paradigm Benefits and advantages of Cloud Computing [3]
Cloud Architecture: Cloud Infrastructure, Architecture of each components, Virtualization versus Traditional Approach, Virtualization Model for Cloud Computing. [2]
Services and Applications by Type [3]
IaaS – Basic concept, Workload, partitioning of virtual private server instances, Pods, aggregations, silos.
PaaS – Basic concept, tools and development environment with examples
SaaS - Basic concept and characteristics, Open SaaS and SOA, examples of SaaS platform
Identity as a Service (IDaaS) Compliance as a Service (CaaS)

Module 2: Use of Platforms in Cloud Computing [6L]
Concepts of Abstraction and Virtualization [2L]
Virtualization technologies: Types of virtualization, Load Balancing and Virtualization: Basic Concepts, Network resources for load balancing; Classification of Virtualization Environment: Scheduling-based Environment, Load-Distribution-Based Environment, Energy Aware-Based Environment, Operational-Based Environment, Distributed Pattern-Based Environment, Transactional-Based Environment
Mention of The Google Cloud as an example of use of load balancing Hypervisors: Virtual machine technology and types, VMware vSphere Machine imaging (including mention of Open Virtualization Format – OVF) [2L]
Porting of applications in the Cloud: The simple Cloud API and AppZero Virtual Application appliance
Concepts of Platform as a Service [2L]
Definition of services, Distinction between SaaS and PaaS (knowledge of Salesforce.com and Force.com), Application development. Use of PaaS Application frameworks.

Module 3: Cloud Service Models [6L]
Use of Google Web Services [2L]
Discussion of Google Applications Portfolio – Indexed search, Dark Web, Aggregation and disintermediation, Productivity applications and service, Adwords, Google Analytics, Google Translate, a brief discussion on Google Toolkit (including introduction of Google APIs in brief), major features of Google App Engine service.
Use of Amazon Web Services [2L]
Amazon Web Service components and services: Amazon Elastic Cloud, Amazon Simple Storage system, Amazon Elastic Block Store, Amazon SimpleDB and Relational Database Service

Use of Microsoft Cloud Services [2L]
Windows Azure platform: Microsoft's approach, architecture, and main elements, overview of Windows Azure AppFabric, Content Delivery Network, SQL Azure, and Windows Live services

Module 4: Cloud Infrastructure [10L]
Types of services required in implementation – Consulting, Configuration, Customization and Support
Cloud Management [3L]
An overview of the features of network management systems and a brief introduction of related products from large cloud vendors, Monitoring of an entire cloud computing deployment stack – an overview with mention of some products, Lifecycle management of cloud services (six stages of lifecycle)
Live Migration of Virtual Machines: [2L]
Need of Live Migration of Virtual Machine, A Designing Process of Live Migration, and Security Issues during live migration.
Concepts of Cloud Security [3L]
Infrastructure Security, Infrastructure Security: The Network Level, The Host Level, The Application Level, Data Security and Storage, Aspects of Data Security, Data Security Mitigation Provider Data and Its Security, Identity and Access Management.
Auditing and Compliance in Cloud Environment: [2L]
Data Security in Cloud Computing Environment, Need for Auditing in Cloud Computing Environment, Third Party Service Provider, Cloud Auditing Outsourcing Lifecycle Phases, Auditing Classification.

Module 5: Concepts of Services and Applications [6L]
Service Oriented Architecture: Basic concepts of message-based transactions, Protocol stack for an SOA architecture, Event-driven SOA, Enterprise Service Bus, Service catalogs [6]
Applications in the Cloud: Concepts of cloud transactions, functionality mapping, Application attributes, Cloud service attributes, System abstraction and Cloud Bursting, Applications and Cloud APIs [2]
Cloud-based Storage: Cloud storage definition – Manned and Unmanned. [1]
Webmail Services: Cloud mail services including Google Gmail, Mail2Web, Windows Live Hotmail, Yahoo mail, concepts of Syndication services [1]

Textbooks:

1. Cloud Computing Bible by Barrie Sosinsky, Wiley India Pvt. Ltd, 2013
2. Cloud Computing: A Practical Approach, Anthony T. Velte, Tata Mcgraw-Hill

Reference

1. Mastering Cloud Computing by Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, McGraw Hill Education (India)PrivateLimited,2013
2. Fundamentals of Cloud Computing by P. K. Pattnaik, S. Pal, M. R. Kabat, Vikas Publications, 2014.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS604C.1	3	3	3	3	2	-	-	-	-	-	-	-	3	1	1
CS604C.2	3	3	3	3	3	-	-	-	2	-	-	1	3	1	1
CS604C.3	3	3	3	3	3	-	-	-	1	-	-	-	3	1	1
CS604C.4	3	2	2	3	3	-	-	-	1	-	-	1	3	1	2

Course Name: Cyber Law and Ethics
Course Code: CS605
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite:

1. Familiarity in computer Networking.
2. Basic concepts about network security.

Course Objectives

- To understand, explore, and acquire a critical understanding of **Cyber Law**
- To learn the fundamentals of **Cybersecurity**
- To develop competencies in identifying and dealing with **frauds and deceptions** (e.g., confidence tricks, scams)

Course Outcomes (COs)

Upon successful completion of this course, students will be able to:
CS605.1: Understand the social and intellectual property issues arising from cyberspace
CS605.2: Gain knowledge of the Information Technology Act, legal frameworks for privacy, data security, and data protection
CS605.3: Analyze the interplay between commerce and cyberspace
CS605.4: Identify and review various network security threats and countermeasures
CS605.5: Understand and adapt to advanced cybersecurity technologies and issues

Course Contents

Module 1: Introduction to Cybercrime (5 Hours)

- Cybercrime, Forgery, Hacking, Software Piracy
- Computer Network Intrusion, Criminal planning, Passive vs. Active Attacks
- Cyber Stalking

Module 2: Cybercrime in Mobile and Wireless Devices (8 Hours)

- Security challenges in mobile environments
- Cryptographic security for mobile devices
- Attacks on mobile/cell phones: Theft, Viruses, Hacking
- Bluetooth security issues, Laptop malware

Module 3: Tools and Methods Used in Cybercrime (7 Hours)

- Proxy Servers, Password and Random Checking
- Trojan Horses, Backdoors, DoS and DDoS Attacks
- SQL Injection, Buffer Overflow, Script Kiddies, Packaged Defense Tools

Module 4: Cybercrime and Cybersecurity (6 Hours)

- Phishing, Identity Theft, Online Identity Management
- Legal aspects: Indian laws, IT Act, Public Key Infrastructure
- Designing Cybersecurity Policies
- UNCITRAL Model Law, Cyber Jurisdiction (civil, criminal, international)

Module 5: Cyber Ethics (5 Hours)

- Importance of Cyber Law and Cyber Ethics
- Necessity for Cyber Regulations
- Ethics in the Information Society
- Introduction to AI Ethics: Core principles and ethical concerns
- Introduction to Blockchain Ethics

Text Books:

1. Cybersecurity by Nina Gobole & Sunit Belapune; Pub: Wiley India.
2. Chris Reed & John Angel, Computer Law, OUP, New York, (2007).
3. Justice Yatindra Singh, Cyber Laws, Universal Law Publishing Co., New Delhi, (2012).
4. Verma S.K., Mittal Raman, Legal Dimensions of Cyber Space, Indian Law Institute, New Delhi, (2004).

Reference

1. Kenneth J. Knapp, “Cyber Security and Global Information Assurance: Threat Analysis and

Response Solutions", IGI Global, 2009.
2. Jonathan Rosenoer, *Cyberlaw: The Law of the Internet*, Springer-Verlag, 1997.
3. Sudhir Naib, *The Information Technology Act, 2005: A Handbook*, OUP, New York.
4. Vasu Deva, *Cyber Crimes and Law Enforcement*, Commonwealth Publishers, New Delhi, (2003).

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS605.1	1	3	1	1	1	3	-	3	-	-	-	2	-	1	-
CS605.2	3	3	1	2	3	-	-	3	-	-	-	1	1	2	-
CS605.3	2	3	3	3	1	1	-	1	-	1	-	1	2	2	1
CS605.4	2	2	3	3	2	-	-	-	-	-	-	-	2	3	2
CS605.5	3	2	3	3	3	1	-	2	-	-	-	-	3	3	2

Course Name: Web and Internet Technology Lab
Course Code: CS691
Contact: 0:0:3
Total Contact Hours: 36
Credits: 1.5

Prerequisites:
1. Computer Networks Lab
2. C Programming

Course Objective(s):
- To impart the design, development and implementation of Static and Dynamic Web Pages.
- To develop programs for Web using Scripting Languages
- To give an overview of Server-Side Programming in Web.

Course Outcome(s):
After completion of the course students will be able to

CO1: To define interactive web pages using HTML, DHTML, CSS and image map.
CO2: To summarize the knowledge of information interchange formats like XML.
CO3: To implement JavaScript – a client-side scripting languages in web programming.
CO4: To explain PHP applications
CO5: To write the server-side programming concepts using Servlet, JSP.

List of Experiments:

1. Write a single html program through which you can explain a) anchor tag, b)'imp' tag with 'src' attribute, c) paragraph d) heading.
2. Write a single html program through which you can draw a table which consists of 3 row and 4 columns where 1st row contains 4 different column fields of a student's information with red text color and Calibri font style with font 12. Rest cells of whole table contain values with blue text colors and Times new roman font style with font 10.
3. Write a single html program where 1st paragraph can collect its specified style from internal stylesheet describes inside that html program and 2nd paragraph can collect its specified style from another file (external stylesheet).
4. Write a single html program which implements image map concept using 'usemap' and <map>.
5. Write a html program to find out Celsius temperature of a given Fahrenheit temperature using JavaScript.
6. Write a html program to find out m to the power n (m, n valid integer no) using a function using JavaScript.
7. Write a xml parsing technique through which parse a text string into an XML DOM object, and extracts the info from it with JavaScript.
8. Write a simple PHP program through which you can find out maximum and minimum among three nos specified by the user.
9. Write a simple PHP program through which you can implement the concept of GET & POST method w.r.t PHP Form handling.
10. Write a simple program in PHP through which you can create a login page of your own website.
11. Write a simple JSP program through which you can print even and odd no separately within a given range.
12. Create an Online Registration form for individual user of a website using Servlet.

Text book:

1. "Web Technology: A Developer's Perspective", N.P. Gopalan and J. Akilandeswari, PHI Learning, Delhi, 2013. (Topics covered: html, CSS, imagemap, xml)
2. "Learning PHP, MySQL & JavaScript", Robin Nixon, O'Reilly Publication. (Topics covered: PHP, Java Script)
3. "Head First Servlet's & JSP", Bryan Basham, Kathy Sterra, Bert Bates, O'Reilly Publication. (Topics covered: Servlet, JSP)

Reference

1. "Programming the World Wide Web", Robert. W. Sebesta, Fourth Edition, Pearson Education.
2. "Core Web Programming"- Second Edition-Volume I and II, Marty Hall and Larry Brown, Pearson Education,

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	3	2	-	-	-	-	1	3	2	2	1	2
CO2	2	2	3	2	3	-	-	-	-	2	2	2	2	2	3
CO3	1	2	1	2	2	-	-	-	-	-	2	1	2	3	-
CO4	3	2	3	2	3	-	-	-	-	1	2	1	3	-	1
CO5	1	3	2	3	2	-	-	-	-	-	1	-	2	2	2

Course Name: Machine Learning Lab
Course Code: CS692
Contact: 0:0:3
Total Contact Hours: 36
Credits: 1.5

Prerequisites:

1. Linear Algebra and Probability
2. Data Structures and Algorithms
3. Programming in Python
4. Introduction to Artificial Intelligence

Course Objective(s):

This course will enable students to

1. Make use of data sets in implementing machine-learning algorithms.
2. Implement machine-learning concepts and algorithms in any suitable programming language.

Course Outcome(s):

After completion of the course students will be able to

CO1: Understand the implementation procedures for machine-learning algorithms.
CO2: Design Java/Python programs for various learning algorithms.
CO3: Apply appropriate data sets to machine-learning algorithms
CO4: Identify and apply machine-learning algorithms to solve real-world problems.

List of Lab Experiments

1. Implement and demonstrate the **FIND-S** algorithm for finding the most specific hypothesis based on a given set of training data samples (read data from a CSV file).

2. For a given set of training examples stored in a CSV file, implement and demonstrate the **Candidate-Elimination** algorithm to output all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision-tree-based **ID3** algorithm. Use an appropriate data set to build the tree and classify a new sample.
4. Build an **Artificial Neural Network** by implementing the back-propagation algorithm and test it with an appropriate data set.
5. Write a program to implement the **naïve-Bayes classifier** for a sample training data set stored as a CSV file. Compute classifier accuracy on a test set.
6. Assuming a set of documents that need classification, use a naïve-Bayes model to perform this task (built-in Java/Python libraries may be used). Calculate accuracy, precision and recall.
7. Construct a **Bayesian network** for medical data and demonstrate diagnosis of heart patients using the standard Heart-Disease data set (use Java/Python ML libraries).
8. Apply the **EM algorithm** to cluster data stored in a CSV file. Use the same data set for clustering with **k-means**; compare results and comment on clustering quality (libraries may be used).
9. Implement the **k-Nearest-Neighbour** algorithm to classify the Iris data set. Print both correct and wrong predictions (libraries permitted).
10. Implement **Locally Weighted Regression** (non-parametric) to fit data points. Select an appropriate data set and draw graphs of the fit.
11. **Build and evaluate a logistic-regression classifier** on a binary data set (e.g., Breast-Cancer or Spam). Report accuracy and ROC curve.
12. **Perform principal-component analysis (PCA)** for dimensionality reduction on a high-dimensional data set, then apply k-means to the reduced data and visualise the clusters.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS692.1	2	3	3	–	–	–	–	–	–	–	3	2	3	3	3
CS692.2	–	3	3	2	3	2	–	–	–	–	–	–	3	2	3
CS692.3	–	2	–	2	2	2	–	–	–	–	–	–	1	2	1
CS692.4	2	3	–	3	–	2	–	–	–	–	–	–	–	2	1

Course Name: Software Engineering Lab
Course Code: CS693
Contact: 0:0:3
Total Contact Hours: 36
Credits: 1.5

Prerequisites: Programming for Problem Solving

Course Objectives:

- **Apply Software Development Life Cycle (SDLC) in Practice:** To provide hands-on experience in implementing the various phases of the SDLC—requirements gathering, design, development, testing, and maintenance—using real-world scenarios or case studies.

- **Develop and Document Software Requirements:** To train students in creating Software Requirements Specifications (SRS) and related documentation through practical tools and techniques.
- **Design and Model Software Systems:** To enable students to apply design principles and model software systems using UML diagrams such as use case, class, activity, and sequence diagrams.
- **Implement and Test Software Projects:** To build working software systems based on predefined requirements and apply testing techniques to validate the correctness and quality of the developed systems.
- **Use Software Engineering Tools:** To familiarize students with essential software engineering tools such as version control (e.g., Git), project management (e.g., Trello or JIRA), CASE tools, and testing frameworks.

Course Outcomes:
On completion of the course students will be able

CS693.1: To identify software development models.
CS693.2: To prepare SRS document, design document, project management related document.
CS693.3: To validate function oriented and object-oriented software design using tools.
CS693.4: To adapt various testing techniques through test cases

Course Content:
Module-1: [6L]
Preparation of requirement document for standard application problems in standard format. (e.g., Library Management System, Railway Reservation system, Hospital management System, University Admission system). DFD of standard application problems.

Module-2: [6L]
Software Requirement Analysis: Describe the individual Phases/ modules of the project, Identify deliverables. Compute Process and Product Metrics (e.g. Defect Density, Defect Age, Productivity, Cost etc.) Estimation of project size using Function Point (FP) for calculation. Cost Estimation models. L

Module-3: [6L]
Use Case diagram, Class Diagram, Sequence Diagram, Activity Diagram and prepare Software Design Document using tools like Rational Rose. (For standard application problems)

Module-4: [9L]
Software Development, Coding Practice and Debugging, Design Test Script/Test Plan (both Black box and White Box approach)

Module-5: [9L]

Software project management, Project planning and control, configuration control, cost estimation, project scheduling using PERT and GANTT charts, cost-time relations using standard tools.

Text Books:

1. Fundamentals of Software Engineering by Rajib Mall, –PHI-3rd Edition, 2009.
2. Software Engineering-Pankaj Jalote (Wiley-India)

Reference

1. Software Engineering–Agarwal and Agarwal (PHI)
2. Software Engineering, by Ian Sommerville, Pearson Education Inc., New Delhi, (2009).
3. Software Engineering: A Practitioner's Approach, by Roger S. Pressman, McGraw-Hill. (2005)

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	2	1	1	-	3	2	2	2	2	3	2
CO2	3	3	2	3	3	1	1	-	3	3	2	2	2	3	2
CO3	2	3	2	3	3	-	2	-	3	2	2	1	3	3	3
CO4	2	3	2	3	2	-	1	-	3	2	2	-	3	3	3

4th Year 7th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
C. THEORY									
1	ENGG	Major	CS 701	Neural Networks and Deep Learning	3	0	0	3	3
2	ENGG	Major	CS 702 A CS 702 B CS 702 C	Advanced Algorithms Advanced Computer Architecture Advanced Operating Systems	3	0	0	3	3
3	ENGG	Minor	CS 703 A CS 703 B CS 703 C	Information Theory and Coding Ad-Hoc and Sensor Networks Data Mining and Data Warehouse	3	0	0	3	3
4	HUM	Minor	HU(CS)701	Human Resource Development and Organizational Behavior	2	0	0	2	2
D. PRACTICAL									
5	ENGG	Major	CS 791	Neural Networks and Deep Learning Lab	0	0	3	3	1.5
6	ENGG	Major	CS 792 A CS 792 B CS 792 C	Advanced Algorithms Lab Advanced Computer Architecture Lab Advanced Operating Systems Lab	0	0	3	3	1.5
7	PRJ	Project	CS 793	Major Project-I	0	0	12	12	6
Total of Theory, Practical and Mandatory Activities/Courses								29	20

Course Name: Neural Networks and Deep Learning
Course Code: CS 701
Contact: 3:0:0
Total Contact Hours: 36L
Credits: 3

Prerequisites:

- Statistics, Calculus, Linear Algebra, Probability
- Python libraries: NumPy, Pandas, Matplotlib, Scikit-learn

Course Objective(s):

- To introduce the fundamental techniques and core principles of Neural Networks
- To study various ANN models along with their real-world applications
- To familiarize learners with deep learning concepts, including CNNs and RNNs

Course Outcome(s):
After completion of the course students will be able to

CO1: Explain the fundamental concepts and principles of Neural Networks and Deep Learning.
CO2: Apply shallow and deep neural network models to solve basic machine learning problems.
CO3: Analyze Convolutional Neural Networks (CNNs) for image-related tasks.
CO4: Evaluate Recurrent Neural Networks (RNNs) for sequence and time-series data.
CO5: Design deep learning models for real-world applications using appropriate tools and techniques.

Course Content:

Module I [8L]
Introduction to supervised learning, Classifications, Logistic regression, Cost function, Gradient descent optimization, Computation graphs, Vectorization techniques, Basics of model evaluation.

Module II [8L]
Neural network representation, Forward propagation, Activation functions and their derivatives, Need for non-linearity, Loss functions, Gradient descent for neural networks, Backpropagation algorithm, Parameter initialization.

Module III [7L]
Deep neural network architecture, Forward and backward propagation in deep networks, Matrix dimension handling, Hyperparameters vs parameters, Regularization techniques, Optimization strategies, Building blocks of deep models.

Module IV [7L]
Convolution operations, Padding and stride, Pooling layers, Classic CNN architectures, Residual Networks (ResNet), Transfer learning, Applications in image classification, Object detection (YOLO), Image segmentation (U-Net), Introduction to OpenCV.

Module V [6L]
Recurrent Neural Networks (RNN), LSTM and GRU architectures, Vanishing and exploding gradients, Word embeddings (Word2Vec, GloVe), Sequence applications (sentiment analysis, machine translation), Attention mechanisms, Introduction to Transformers.

Text book:

1. Charu C Aggarwal "Neural Networks and Deep Learning", Springer, 2018.
2. Ian Goodfellow, Yoshua Bengio and Aaron Courville "Deep Learning", MIT Press, 2016.
3. Francois Chollet "Deep Learning with Python", Manning Publications, 2018.

Reference book:

1. Francois Chollet "Deep Learning with Python", Manning Publications, 2018.
2. Simon Haykin "Neural Networks and Learning Machines", Pearson Education, 2009.
3. Martin T Hagan, Howard B Demuth and Mark H Beale "Neural Network Design", PWS Publishing Company, 1996.
4. AurélienGéron "Hands-On Machine Learning WithScikit-Learn, Keras and TensorFlow", O'Reilly Media, 2019.

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	1	1	–	–	–	–	–	–	1	2	1
CO2	3	3	2	2	2	–	–	–	1	1	–	1	3	2
CO3	3	3	3	3	3	–	–	–	1	1	–	1	3	3
CO4	3	3	2	3	2	–	–	–	1	1	–	1	3	3
CO5	3	3	3	3	3	1	–	–	2	2	1	2	–	–

Course Name: Advanced Algorithm
Course Code: CS702A
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisites: Design & Analysis of Algorithm

Course Objective(s):

- The aim is to learn how to develop efficient algorithms for simple computational tasks and reasoning about the correctness of them.
- Through the complexity measures, different range of behaviors of algorithms and the notion of tractable and intractable problems will be understood.

Course Outcome(s):
After completion of the course students will be able to

CO1: Analyze the complexity/performance of different algorithms.

CO2: Determine the appropriate data structure for solving a particular set of problems.
CO3: Categorize the different problems in various classes according to their complexity.
CO4: Achieve an insight of recent activities in the field of the advanced data structure.
CO5: Design solutions for a real-world problem by applying relevant distributions.

Course Content:
Module-1 [4L]
Sorting:
Review of various sorting algorithms, topological sorting
Graph:
Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkstra's), DFS and computation of strongly connected components, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis.

Module-2 [6L]
Matroids:
Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST.
Graph Matching:
Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path.

Module-3 [16L]
Flow-Networks:
Maxflow-Mincut Theorem, Ford Fulkerson Method to compute Maximum Flow, Edmond-Karp maximum-flow algorithm.
Matrix Computations:
Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition
Shortest Path in Graphs:
Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming.
Modulo Representation of integers/ polynomials:
Chinese Remainder Theorem, Conversion between base-representation and modulo-representation. Extension to polynomials. Application: Interpolation problem.
Discrete Fourier Transform (DFT):
In complex field, DFT in modulo ring. Fast Fourier Transform algorithm. Schönhage-Strassen Integer Multiplication algorithm
Amortized Analysis:
Aggregate, Accounting, and Potential Method

Module-4 [10L]
Linear Programming:
Geometry of the feasibility region and Simplex algorithm
NP-completeness:
Examples, proof of NP-hardness and NP-completeness. One or more of the following topics based

on time and interest Approximation algorithms, Randomized Algorithms, Interior Point Method, Advanced Number Theoretic Algorithm

Problem Solving Application

Recent Trends in problem solving paradigms using recent searching and sorting techniques by applying recently proposed data structures.

Textbook:

1. "Introduction to Algorithms" by Cormen, Leiserson, Rivest, Stein.
2. "The Design and Analysis of Computer Algorithms" by Aho, Hopcroft, Ullman.

Reference Books:

1. "Algorithm Design" by Kleinberg and Tardos.
2. Design & Analysis of Algorithms, Gajendra Sharma, Khanna Publishing House, New Delhi

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	-	-	-	-	-	-	-	2	2	2	3
CO2	3	3	3	3	-	-	-	-	-	-	-	2	2	3	3
CO3	3	3	3	3	-	-	-	-	-	-	-	3	3	3	3
CO4	3	3	3	3	-	-	-	-	-	-	-	3-	3	2	3
CO5	3	3	3	3	-	-	-	-	-	-	-	3	3	3	3

Course Name: Advanced Computer Architecture
Course Code: CS702B
Contact: 3:1:0
Total Contact Hours: 36L
Credits: 4

Prerequisites:

- Familiarity with the functionalities of basic digital computer system.
- Fundamentals of Computer Architecture.

Course Objective(s):

- To understand the Concept of Parallel Processing and its applications
- To implement the Hardware for Arithmetic Operations
- To analyse the performance of different scalar Computers
- To understand the Pipelining Concept for a given set of Instructions
- To learn the performance of pipelining and non-pipelining environment in a processor

Course Outcome(s):

After completion of the course students will be able to

CO1: Acquire the knowledge of parallelism and pipelining
CO2: Develop knowledge of parallel processing
CO3: Combine the concept and design techniques of interconnection network
CO4: Acquire the knowledge of shared memory architecture
CO5: Describe the fundamentals of embedded system architecture

Course Content:
Module I: Introduction to Advanced Computer Architectures [5L]

Different types of architectural classifications – instruction vs. data (SISD, SIMD, MISD, MIMD), serial vs. parallel, pipelining vs. parallelism; Pipelining: Definition, different types of pipelining, hazards in pipelining. Concept of reservation tables, issue of multiple instructions with minimum average latency (MAL).

Module II: Parallel Processing & ILP [8L]
RISC architecture, characteristics of RISC instruction set & RISC pipeline, its comparisons with CISC, necessity of using optimizing compilers with RISC architecture, Review of instruction level parallelism-Super pipelining, Superscalar architecture, Diversified pipelines and out of order execution, VLIW architecture, Dataflow and Control Flow Architectures, Loop Parallelization

Module III: Interconnection Networks [13L]
Desirable properties of interconnection networks, static interconnection networks – path, cycle, double-loop, star, wheel, 2D mesh and its variants, multi-mesh, tree, shuffle-exchange, cube, cube connected cycles. Dynamic interconnection networks: concepts of blocking, rearrangeable and blocking but rearrangeable networks, various types of multistage interconnection networks (MIN)-crossbar, clos, baseline, omega, Benes.

Module IV: Shared Memory Architecture [5L]

Fundamentals of UMA, NUMA, NORMA, COMA architectures, Performance measurement for parallel architectures –Amadahl's law, Gustafson's law

Module V: Embedded System Architecture [5L]

Definition, Example, Classification of Embedded system, Embedded System Design Issues: Hardware issues (Processor, Memory, Peripherals), Software issues (Programming Languages, Time Criticality, RTOS).

Text Books:

1. J. L. Hennessey and D. A. Patterson: Computer Architecture: A Quantitative Approach, 5th edition, Morgan Kaufmann, 2012.
2. K. Hwang and F. A. Briggs: Computer Architecture and Parallel Processing, Tata McGraw Hill, New Delhi.

Reference Books:

1. Tse-yun Feng, A Survey of Interconnection Networks, IEEE, 1981.

2. Selim G. Akl, The Design and Analysis of Parallel Algorithms, Prentice-Hall, 1989.
3. Raj Kamal, Embedded Systems Architectures Programming and Design, Second Edition, The MacGraw-Hill (for Embedded System).

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	-	-	-	-	3	2	-	-	-	2	1	2	2	3
CO2	3	3	1	-	-	3	-	-	2	2	-	-	2	3	3
CO3	3	3	3	2	3	3	-	2	-	-	-	-	3	3	3
CO4	3	3	3	2	3	2	2	2	-	2	-	1	3	2	3
CO5	3	2	2	2	2	2	-	-	3	2	-	2	3	3	3

Course Name: Advanced Operating Systems
Course Code: CS702C
Contact: 3:0:0
Total Contact Hours: 36L
Credits: 3

Prerequisites: Fundamentals of Operating Systems, basic knowledge of computer architecture and data structures.

Course Objective(s):

- To study the principles and architectures of distributed and multiprocessor operating systems.
- To analyze synchronization, communication, and scheduling mechanisms in distributed environments.
- To explore modern techniques such as distributed shared memory, load balancing, and fault tolerance.

Course Outcome(s):

After completion of the course students will be able to

CO1: Demonstrate a comprehensive understanding of advanced operating system concepts and compare various OS architectures.
CO2: Analyze the design issues, challenges, and performance aspects of distributed operating systems.
CO3: Evaluate different architectural models used in distributed systems.
CO4: Analyze the design and functioning of multiprocessor operating systems.
CO5: Design efficient scheduling and resource management strategies for real-time and distributed applications.

Course Contents:

Module I [6L]

Architectures of Distributed Systems: Classification and comparison of system architecture types (client-server, peer-to-peer, hybrid). **Distributed Operating Systems:** Goals, features, and design issues; communication primitives including message passing and RPC. **Theoretical Foundations:** Inherent limitations of distributed systems such as lack of global clock and partial failures. **Logical Time and Event Ordering:** Lamport's logical clocks, vector clocks, causal ordering of messages, and distributed termination detection algorithms.

Module II [7L]

Distributed Mutual Exclusion: Classification, performance metrics, and design trade-offs of mutual exclusion algorithms. **Non-token-based algorithms:** Lamport's algorithm, Ricart–Agarwala algorithm, Maekawa's algorithm with analysis of message complexity and synchronization delay. **Token-based algorithms:** Suzuki–Kasami broadcast algorithm, Singhal's heuristic algorithm, Raymond's tree-based algorithm and their comparative evaluation.

Module III [7L]

Distributed Deadlock Detection: System model, resource allocation graph in distributed systems. **Deadlock Handling Strategies:** Prevention, avoidance, detection, and recovery in distributed environments. **Issues in Deadlock Handling**: Phantom deadlocks, communication delays, and consistency problems. **Deadlock Detection Algorithms:** Centralized, distributed, and hierarchical approaches with performance considerations and limitations.

Module IV [8L]

Multiprocessor System Architectures: Motivation, tightly coupled vs loosely coupled systems, shared memory and distributed memory architectures. **Multiprocessor Operating Systems:** Structures and design approaches; issues in scalability and reliability. Threads, concurrency, and process synchronization mechanisms. Processor scheduling techniques for multiprocessor environments. **Distributed File Systems:** Architecture, file sharing semantics, naming, caching, replication techniques, and design challenges.

Module V [8L]

Distributed Scheduling and Load Balancing: Issues in load distribution, components and policies of load balancing algorithms, stability considerations. Task migration techniques, process transfer mechanisms, and associated overheads. **Distributed Shared Memory (DSM):** Architecture, design goals, and implementation strategies. Algorithms for DSM management, memory consistency models. Memory coherence, coherence protocols (write-invalidate/write-update), and related design issues.

Text Books:

1. Mukesh Singhal and Niranjan Shivaratri, Advanced Concepts in Operating Systems, McGraw-Hill.
2. Andrew S. Tanenbaum, Distributed Operating Systems, ACM Press.

Reference Books:

1. Nancy Lynch, Distributed Algorithms, Morgan Kaufmann.
2. Jie Wu, Distributed Systems, CRC Press.

3. Hagit Attiya, Jennifer Welch, Distributed Computing: Fundamentals, Simulations and Advanced Topics, McGraw-Hill.
4. Sape Mullender (ed.), Distributed Systems, Addison-Wesley.

CO–PO–PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	-	-	-	-	-	-	-	-	2	1	-
CO2	3	3	3	3	2	-	-	-	-	-	-	-	2	2	1
CO3	3	3	3	3	-	-	-	-	-	-	-	-	3	2	1
CO4	3	3	3	3	-	-	-	-	-	-	-	-	2	2	-
CO5	3	3	3	3	2	-	-	-	3	2	-	-	2	2	2

Course Name: Information Theory and Coding
Course Code: CS703A
Contact: 3:1:0
Total Contact Hours: 36L
Credits: 4

Prerequisites:
Probability Theory, Discrete Mathematics, and Basic Communication Systems

Course Objective(s):
- To introduce fundamental concepts of information theory.
- To provide knowledge of source coding techniques.
- To understand channel coding techniques.
- To design reliable communication coding schemes.

Course Outcome(s):
CO1: Understand entropy and mutual information.
CO2: Analyze source coding techniques.
CO3: Apply channel coding techniques.
CO4: Evaluate communication systems.
CO5: Design coding schemes.

Course Content:
Module I [6L]
Introduction to Information Theory: Information, Uncertainty, Entropy, Joint and Conditional Entropy, Relative Entropy, Mutual Information, Chain Rules, Properties of Entropy and Mutual Information.

Module II [6L]
Source Coding: Shannon's Source Coding Theorem, Prefix Codes, Kraft-McMillan Inequality, Huffman Coding, Shannon-Fano Coding, Arithmetic Coding, Run Length Encoding, Lempel-Ziv Coding.

Module III [6L]

Discrete Memoryless Channels: Channel Models, Channel Capacity, Binary Symmetric Channel (BSC), Binary Erasure Channel (BEC), Mutual Information for Channels, Shannon's Channel Coding Theorem.

Module IV [6L]
Error Detection and Correction: Linear Block Codes, Generator and Parity Check Matrices, Syndrome Decoding, Hamming Codes, Cyclic Codes, CRC (Cyclic Redundancy Check).

Module V [6L]
Convolutional Codes: Encoding and Decoding, State Diagram, Trellis Diagram, Viterbi Algorithm, Performance Analysis of Convolutional Codes.

Module VI [6L]
Advanced Coding Techniques: BCH Codes, Reed-Solomon Codes, Low-Density Parity Check (LDPC) Codes, Turbo Codes, Introduction to Network Coding and Applications in Modern Communication Systems.

Text book:

1. Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory*, Wiley.
2. Simon Haykin, *Communication Systems*, Wiley.

Reference Books:

1. Shu Lin and Daniel J. Costello, *Error Control Coding*, Pearson.
2. Ranjan Bose, *Information Theory, Coding and Cryptography*, McGraw-Hill.
3. David J.C. MacKay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press.

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	-	-	-	-	-	-	-	1	2	2	3
CO2	3	3	3	2	-	-	-	-	-	-	-	-	2	3	3
CO3	3	3	3	3	-	-	-	-	-	-	-	-	3	3	3
CO4	3	3	3	3	-	-	-	-	-	-	-	1	3	2	3
CO5	3	3	3	3	-	-	-	-	-	-	-	-	3	3	3

Course Name: Ad-Hoc and Sensor Networks
Course code: CS703B
Contact: 3:0:0
Total Contact Hours: 36L
Credits: 3

Course Objective(s):
The student should be made to:

- Learn Ad hoc network and Sensor Network fundamentals
- Understand the different routing protocols
- Have an in-depth knowledge on sensor network architecture and design issues

- Understand the transport layer and security issues possible in Ad hoc and Sensor networks
- Have an exposure to mote programming platforms and tools

Course Outcome(s):
At the end of the course, the student would be able to:

CO1: Know the basics of Ad hoc networks and Wireless Sensor Networks.
CO2: Apply this knowledge to identify the suitable routing algorithm based on the network and user requirement.
CO3: Apply the knowledge to identify appropriate physical and MAC layer protocols.
CO4: Understand the transport layer and security issues possible in Ad hoc and sensor networks.
CO5: Be familiar with the OS used in Wireless Sensor Networks and build basic modules.

Course Content:
Module 1: AD HOC NETWORKS – INTRODUCTION AND ROUTING PROTOCOLS [8L]
Elements of Ad hoc Wireless Networks, Issues in Ad hoc wireless networks, Example commercial applications of Ad hoc networking, ad hoc wireless Internet, Issues in Designing a Routing Protocol for Ad Hoc Wireless Networks, Classifications of Routing Protocols, Table Driven Routing Protocols – Destination Sequenced Distance Vector (DSDV), On–Demand Routing protocols –Ad hoc On–Demand Distance Vector Routing (AODV).

Module 2: SENSOR NETWORKS – INTRODUCTION & ARCHITECTURES [7L]
Challenges for Wireless Sensor Networks, Enabling Technologies for Wireless Sensor Networks, WSN application examples, Single-Node Architecture – Hardware Components, Energy Consumption of Sensor Nodes, Network Architecture – Sensor Network Scenarios, Transceiver Design Considerations, Optimization Goals and Figures of Merit.

Module 3: WSN NETWORKING CONCEPTS AND PROTOCOLS [7L]
MAC Protocols for Wireless Sensor Networks, Low Duty Cycle Protocols and Wakeup Concepts – S-MAC, The Mediation Device Protocol, Contention based protocols – PAMAS, Schedule based protocols – LEACH, IEEE 802.15.4 MAC protocol, Routing Protocols, Energy Efficient Routing, Challenges and Issues in Transport layer protocol.

Module 4: SENSOR NETWORK SECURITY [7L]
Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Layer wise attacks in wireless sensor networks, possible solutions for jamming, tampering, black hole attack, flooding attack. Key Distribution and Management, Secure Routing – SPINS, reliability requirements in sensor networks.

Module 5: SENSOR NETWORK PLATFORMS AND TOOLS [7L]
Sensor Node Hardware – Berkeley Motes, Programming Challenges, Node-level software platforms – TinyOS, nesC, CONTIKIOS, Node-level Simulators – NS2 and its extension to sensor networks, COOJA, TOSSIM, Programming beyond individual nodes – State centric programming.

Text Book:

1. Siva Ram Murthy, and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols", Prentice Hall Professional Technical Reference, 2008.

Reference Book:

1. Carlos De Morais Cordeiro, Dharma Prakash Agrawal "Ad Hoc & Sensor Networks: Theory and Applications", World Scientific Publishing Company, 2006.
2. Feng Zhao and Leonides Guibas, "Wireless Sensor Networks", Elsevier Publication – 2002.
3. Holger Karl and Andreas Willig "Protocols and Architectures for Wireless Sensor Networks", Wiley, 2005

Online Resources:

1. www.wirelessnetworksonline.com
2. www.securityinwireless.com
3. www.ida.liu.se/~petel71/SN/lecture-notes/sn.pdf

Practice Aspects:

1. NS2 Simulator tool

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	-	3	-	2	-	-	-	3	-	-	-	-	2	2	1
CO2	3	2	2	2	-	-	1	-	-	-	-	-	3	3	2
CO3	3	2	-	-	3	-	-	-	1	-	-	-	3	2	2
CO4	-	1	-	3	-	-	-	-	-	1	-	-	2	3	1
CO5	3	1	2	2		1							3	2	3

Course Name: Data Mining and Data Warehousing
Course Code: CS703C
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisites: Data Structure, Design and Analysis of Algorithms, Database Management Systems, Statistics, Artificial Intelligence

Course Objectives

- To introduce the fundamental concepts and applications of Data Mining and Data Warehousing.
- To develop understanding of data preprocessing, data warehouse modeling, and OLAP techniques for efficient data analysis.

- To impart knowledge of pattern mining, classification, regression, clustering, and outlier analysis techniques.
- To familiarize students with advanced data mining topics including data stream, graph, text, social network, and web mining.
- To enable students to design and implement data mining solutions for real-world data analysis and decision-making problems.

Course Outcome(s):

After completion of the course students will be able to

CO1: Understand the fundamental concepts of the evolving technologies in Data Mining and Data Warehousing recognizing their utilitarian importance in current technological context for further exploration leading towards lifelong learning.
CO2: Identify an engineering problem within the scope of Data Mining and Data Warehousing paradigm.
CO3: Explore relevant literature and apply the concepts of Data Mining and Data Warehousing to solve problems of making automated decisions dealing with huge amount of data.
CO4: Develop ideas for proposing solutions to the challenging problems of Data Mining and Data Warehousing.
CO5: Implement ideas of Data Mining and Data Warehousing through developing feasible algorithms or frameworks and investigate their effectiveness by analyzing the performances in solving the relevant problems.

Course Content:

Module-1: Introduction to Data Mining [5L]

Basic Concepts - Data Exploration: Data Types, Data Attributes, Statistical Description of Data, Data Visualization, Data Similarity Measure
Data Preprocessing: Data Cleaning, Data Integration, Data Reduction, Data Transformation & Discretization

Module-2: Introduction to Data Warehousing [6L]

Basic Concepts - Data Warehouse Modeling: Data Cube and OLAP (OnLine Analytical Processing), Data Warehouse Design, Usage, Implementation Data Generalization by Attribute-Oriented Induction

Module-3: Mining Frequent Patterns, Associations and Correlation Analysis [5L]

Basic Concepts, Frequent Itemset Mining Methods: The Apriori Algorithm, Mining Frequent Item Sets without Candidate Generation, Mining Frequent Itemset Using Vertical Data Format, Correlation Analysis, Pattern Mining in Multilevel and Multidimensional Space

Module-4: Classification and Regression [6L]

Basic Concepts, k-Nearest-Neighbor Classifier, Decision Tree Classifier, Naïve Bayes Classifier, ANN-Backpropagation Based Classifier, Support Vector Machine Based Classifier, Linear and Nonlinear Regression Methods

Module-5: Clustering and Outlier Analysis [5L]
Basic Concepts, Partitioning Methods: k-Means and k-Medoids, Hierarchical Methods: Agglomerative and Divisive Hierarchical Clustering, Density-Based Methods: DBSCAN: Density-Based Clustering Based on Connected Regions with High Density, Frequent Pattern–Based Clustering Method, and Outlier Analysis

Module-6: Mining Data Stream, Time-Series, and Sequence Data [3L]
Basic Concepts of Data Stream Mining, Mining Time Series Data, Mining Sequence Patterns in Biological Data

Module-7: Introduction to Graph Mining, Social Network Analysis, Multi-relational Data Mining, Text Mining and World Wide Web (WWW) Mining [6L]
Graph Mining: Methods for Mining Frequent Sub graphs (Apriori-based Approach& Pattern Growth Approach), Basic Concepts of Social Network Analysis and Multi-relational Data Mining, Basic Concepts of Text Mining, Basic Concepts of World Wide Web (WWW) Mining.

Textbook:

1. Han J &Kamber M, "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, Third Edition.
2. Parteek Bhatia, "Data Mining and Data Warehousing: Principles and Practical Techniques", Cambridge University Press.

Reference Books:

1. Pang-Ning Tan, Vipin Kumar, Michael Steinbach, "Introduction to Data Mining", Pearson Education.
2. Robert Layton, "Learning Data Mining with Python", Packet Publishing

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	3	2	2	1
CO2	2	3	-	-	-	-	-	-	-	-	-	-	3	3	2
CO3	2	2	3	2	-	-	-	-	-	-	-	-	3	3	3
CO4	2	2	2	3	-	-	-	-	-	-	-	2	3	2	2
CO5	2	2	3	3	2	2	2	-	-	-	-	2	3	3	3

Course Name: Human Resource Development and Organizational Behavior
Course Code: HU(CS)701
Contact: 2:0:0
Total Contact Hours: 24L
Credits: 2

Course Objective(s):

- To develop an understanding of the nature, functioning and design of organization as social collectivizes.
- The basic concepts and theories underlying individual behavior besides developing better insights into one's own self.
- To gain insight into the organizational learning processes, how they can be fostered and enhanced.
- Individual behavior in groups, dynamics of groups and team building besides developing a better awareness of how they can be better facilitators for building effective teams as leaders themselves.

Course Outcome(s):

After completion of the course students will be able to

CO1: Understand the key functions and principles of management as applied in organizations.
CO2: Identify major Human Resource Development (HRD) practices and performance management processes in modern organizations.
CO3: Analyse the behaviour of individuals and groups and the factors influencing organisational behavior.
CO4: Assess the impact of organisational factors such as structure, culture, and change on organisational effectiveness and behavior.
CO5: Evaluate the effects of external environmental factors, including globalisation and technological advancements, on organisational behavior and management practices.

Course Content:

Module I [3L]

HRD-Macro Perspective: HRD Concept, Origin and Need, HRD as a Total System; Approaches to HRD; Human Development and HRD; HRD at Macro and Micro Climate. Areas of HRD; HRD Interventions Performance Appraisal, Potential Appraisal, Feedback and Performance Coaching, Training, Career Planning.

Module II [3L]

OD or Systems Development, Rewards, Employee Welfare and Quality of Work; Life and Human Resource Information; Staffing for HRD: Roles of HR Developer; Physical and Financial Resources for HRD; HR Accounting; HRD Audit, Strategic HRD.

Module III [3L]

Instructional Technology for HRD: Learning and HRD; Models and Curriculum; Principles of Learning; Group and Individual Learning; Transactional Analysis; Assessment Centre; Behaviour Modeling and Self-Directed Learning; Evaluating the HRD.

Module IV [2L]

Human Resource Training and Development: Concept and Importance; Assessing Training Needs; Designing and Evaluating T&D Programmes; Role, Responsibilities and challenges to Training Managers

Module V [7L]

Organisational Effectiveness (OE): Concept; Approaches to O E; Adoptive Coping Cycle for Effectiveness; Achieving OE; Organisational Climate: Concept, Determinants of Organisational Climate; Organization Theory: Classical Theory; Neo-Classical Theory, Modern Behavioural Theories, contingency theory, system theory, modern structural models; Organizational Culture;

Creating and Sustaining Culture; Work Culture

Module VI [6L]

Motivation: Types of Motives; Theories of Maslow; Herzberg, McGregor, Alderfers, Porter and Lawler's Model; Job Enlargement, Job Enrichment, Behavior Modification.

(a) Group & Group Dynamics - concept, importance, classification of groups, reason for group, formation, group cohesiveness. (b) Team work: meaning, concept, types, creating, and an effective team. (c) Leadership: Concept, Leader vs. Manager; Classical Studies on Leadership; Trait Theories; Behavioral Theories; Group and Exchange Theories; Contingency Theory of Leadership; Leadership Styles.

Text book:

1. Rao, T.V and Pareek, Udai: Designing and Managing Human Resource Systems, Oxford IBH Pub. Pvt. Ltd., New Delhi, 2005.
2. Viramani, B.R and Seth, Parmila: Evaluating Management Development, Vision Books, New Delhi.

Reference books:

1. Luthans, Fred: Organisational Behaviour, Tata McGraw-Hill Co. New Delhi, 2004.
2. John, W. Mewstrom & Davis, Keith: Organisational Behavior (Human Behavior at Work), Tata McGraw-Hill, New Delhi, 2002.

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	-	-	-	2	3	3	3	2	3	2	2	1	2	1
CO2	2	-	-	2	-	-	2	-	3	-	3	-	1	3	1
CO3	2	-	3	2	-	3	-	-	2	-	-	-	1	3	1
CO4	-	2	-	3	2	3	-	2	-	-	-	-	1	3	1
CO5	2	-	-	-	3	2	-	-	2	-	3	-	1	2	1

Course Name: Neural Networks and Deep Learning Lab
Course Code: CS791
Contact: 0:0:3
Credits: 1.5

Prerequisites:

- Mathematics: Linear Algebra, Calculus, Probability, Statistics
- Programming: Python
- Libraries: NumPy, Pandas, Matplotlib, Scikit-learn
- Basic understanding of Machine Learning concepts

Course Objectives

- To provide hands-on experience with deep learning frameworks (TensorFlow / Keras / PyTorch)

- To implement neural network models aligned with theoretical foundations
- To develop skills for solving real-world problems using deep learning
- To explore modern architectures like CNNs and RNNs
- To understand model evaluation, optimization, and deployment basics

Course Outcomes (COs)
After completion of the course students will be able to:

CO1: Implement supervised learning models using deep learning frameworks.
CO2: Design artificial neural networks using forward and backward propagation.
CO3: Build deep neural networks using tuning, regularization, and optimization techniques.
CO4: Develop CNN-based models for image processing and computer vision tasks.
CO5: Implement RNN/LSTM-based models for sequence and text-based applications.

Experiments:
1. Setup of deep learning environment (TensorFlow/Keras/PyTorch) and dataset handling
2. Implementation of linear regression using neural networks
3. Implementation of classification model (binary/multiclass) using ANN
4. Implementation of forward propagation and activation functions
5. Implementation of backpropagation and loss function visualization
6. Building a deep neural network (multi-layer ANN)
7. Hyperparameter tuning (learning rate, batch size, epochs)
8. Implementation of regularization techniques (Dropout, L2)
9. Comparison of optimizers (SGD, Adam, RMSProp)
10. Image classification using CNN (MNIST/CIFAR dataset)
11. Transfer learning using pre-trained models (VGG/ResNet – basic implementation)
12. Text/sequence modeling using RNN/LSTM (e.g., sentiment analysis or text generation)

CO-PO-PSO Mapping:

CO \ PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3	1	3	–	–	–	1	–	–	2	3	3	2
CO2	3	3	2	2	2	–	–	–	1	1	–	2	3	3	2
CO3	3	3	3	2	3	–	–	–	2	1	1	2	3	3	3
CO4	3	3	3	2	3	1	1	–	2	2	1	2	3	3	3
CO5	2	3	3	2	3	1	–	–	2	2	1	3	2	3	3

Course Name: Advanced Algorithms Lab
Course Code: CS792A
Contact: 0:0:3
Total Contact Hours: 36L
Credits: 1.5

Prerequisites:
- Programming knowledge

- Knowledge of Design and Analysis of Algorithm

Course Objective(s):

- Design and implement efficient algorithms for a specified application.
- Strengthen the ability to identify and apply the suitable algorithm for the given real-world problem.

Course Outcome(s):

After completion of the course students will be able to

CO1: Implement divide-and-conquer algorithms and analyse their performance for different problem scenarios.
CO2: Apply dynamic programming and greedy techniques to solve optimization and shortest path problems.
CO3: Develop graph traversal and network flow algorithms for solving complex graph-based problems.
CO4: Implement efficient string matching and modular arithmetic algorithms for computational problem solving.
CO5: Apply linear programming and advanced algorithmic techniques to solve real-world optimization problems.

Course Content:

1. Write the following problems in any programming language. Programming Language used: C
2. **Divide and Conquer:** Implementation of finding Maximum and Minimum element from an array of integer, Quick Sort, Check the running time for different positions of pivot elements. Randomized version of quick sort using Divide and Conquer Method.
3. **Dynamic Programming:** Calculation of the minimum number of scalar multiplications needed for chain of Matrices Multiplication Technique, Implementation of Single Source shortest Path for a graph (Dijkstra and Bellman Ford Algorithm), Implement all pair shortest path for a graph (Floyd-Warshall Algorithm)
4. **Greedy method:** Implementation of fractional Knapsack Problem, MST by Prim's algorithm, Implement MST by Kruskal's algorithm
5. **Graph Traversal Algorithm:** Implement Depth First Search (DFS), application of DFS (do topological sorting, identify strongly connected components)
6. **String Matching:** Implement KMP algorithm
7. **Network Flow:** Implement Ford-Fulkerson algorithm to get maximum flow of a given flow network.
8. **Modulo Representation of integers/ polynomials:** Chinese Remainder Theorem
9. **Linear Programming:** Simplex Algorithm

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	1	2	-	-	-	-	-	-	-	3	2	2	3
CO2	3	3	3	1	-	-	-	-	-	-	-	3	2	3	3
CO3	2	2	2	3	-	-	-	-	-	-	-	3	3	3	3
CO4	2	1	3	3	-	-	-	-	-	-	-	3	3	2	3
CO5	3	3	2	3	-	-	-	-	-	-	-	3	3	3	3

Course Name: Advanced Computer Architecture Lab
Course Code:CS792B
Contact: 0:0:3
Credits: 1.5

Prerequisites:
Knowledge of designing different circuits in Computer Organization and Architecture Lab

Course Objective(s):

- To develop proficiency in Hardware Description Language (HDL)
- To design and implement fundamental digital building blocks
- To analyze and design memory systems and data transfer mechanisms
- To design and simulate processor components and subsystems
- To apply architectural concepts to build integrated systems and innovative solutions

Course Outcome(s):
After completion of the course students will be able to

CO1: Understand the fundamentals of Hardware Description Language (HDL) and digital logic design concepts.
CO2: Design basic combinational and sequential circuits using HDL.
CO3: Implement arithmetic circuits, registers, memory units, and ALU using HDL programming.
CO4: Design RAM chips, shift registers, and interfacing mechanisms between CPU and memory.
CO5: Design simple processor architectures and innovative digital system designs using HDL tools.

List of Experiments:

1. HDL introduction
2. Basic digital logic base programming with HDL
3. 8-bit Addition, Multiplication, Division
4. 8-bit Register design

5. Memory unit design and perform memory operators.
6. Implement Encoder, Decoder circuit and simulate for truth table verification.
7. Implementdifferenttypesofflipflopandsimulatefortruthtableverification.
8. Implement different types of parallel circuits (SISO, SIPO, PISO, PIPO) and simulate the result
9. Implement ALU and simulate the result.
10. Implement RAM chip and simulate the result.
11. 8-bit simple CPU design
12. 8. Interfacing of CPU and Memory
13. Innovative Experiments.

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	-	-	-	-	3	2	-	-	-	2	1	2	2	3
CO2	3	3	1	-	-	3	-	-	2	2	-	-	2	3	3
CO3	3	3	3	2	3	3	-	2	-	-	-	-	3	3	3
CO4	3	3	3	2	3	2	2	2	-	2	-	1	3	2	3
CO5	3	2	2	2	2	2	-	-	3	2	-	2	3	3	3

Course Name: Advanced Operating Systems Lab
Course Code: CS792C
Contact: 0:0:3
Total Contact Hours: 36L
Credits: 1.5

Prerequisites:

- Programming knowledge (C / Shell scripting)
- Basic knowledge of Operating Systems

Course Objective(s):

- To design and implement core operating system functionalities using system-level programming.
- To strengthen the ability to analyze and apply OS concepts like process management, synchronization, and memory handling.
- To provide hands-on exposure to real-world operating system mechanisms.

Course Outcome(s):

After completion of the course students will be able to

CO1: Understand Linux commands, shell scripting, and process management concepts in operating systems.
CO2: Develop programs using process creation, process replacement, and inter-process communication techniques.

CO3: Implement multithreading and synchronization mechanisms using mutexes and semaphores.
CO4: Analyse CPU scheduling, memory allocation, and page replacement algorithms.
CO5: Design solutions for deadlock avoidance and resource management in operating systems.

Course Contents:

1. Write a shell script to demonstrate basic Linux commands for file and directory operations (ls, cp, mv, rm, mkdir).
2. Write a shell script using conditional statements and loops (if, case, for, while) to perform basic computations and file handling.
3. Write a C program to create a new process using fork() and demonstrate parent and child process execution.
4. Write a C program to replace a process image using exec() system call.
5. Write a program to demonstrate zombie and orphan processes and analyze their behavior.
6. Write a C program to implement inter-process communication using pipes (unidirectional).
7. Write a C program to demonstrate bidirectional communication using pipes.
8. Write a program to implement multithreading using POSIX threads (pthreads) and perform parallel computation.
9. Write a program to implement process synchronization using mutex and semaphores.
10. Implement the Producer–Consumer problem using semaphores.
11. Implement CPU Scheduling Algorithms:
    - **i.** FCFS
    - **ii.** Shortest Job First (SJF)
12. Implement Round Robin and Priority Scheduling algorithms and compare their performance.
13. Write programs to implement Page Replacement Algorithms:
    - **i.** FIFO
    - **ii.** LRU
    - **iii.** Optimal
14. Write a program to simulate memory allocation techniques:
    - **i.** First Fit
    - **ii.** Best Fit
    - **iii.** Worst Fit
15. Write a program to implement Banker's Algorithm for deadlock avoidance and analyze safe/unsafe states.

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	-	-	-	-	-	-	-	-	2	2	1
CO2	3	3	3	3	2	-	-	-	-	-	-	-	2	3	2
CO3	3	3	3	3	-	-	-	-	-	3	-	-	3	2	2
CO4	3	3	3	3	-	-	-	-	-	-	-	-	2	2	1
CO5	3	3	3	3	-	-	-	-	3	3	-	-	2	2	3

4th Year 8th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
C. THEORY									
1	ENGG	Major	CS 801 A CS 801 B CS 801 C	Real Time Systems Data Analytics Soft Computing	3	1	0	4	4
2	ENGG	Major	CS 802 A CS 802 B CS 802 C	VLSI Design & Application Bio-informatics Robotics	3	1	0	4	4
3	ENGG	Minor	CS 803 A CS 803 B CS 803 C	Introduction to IoT Image Processing Optimization Techniques	3	0	0	3	3
D. PRACTICAL									
4	ENGG	Minor	CS 893 A CS 893 B CS 893 C	Internet of Things Lab Image Processing Lab Optimization Techniques Lab	0	0	3	3	1.5
5	PRJ	Project	CS 881	Major Project-II	0	0	12	12	6
6	PRJ	Internship	CS882	Grand Viva	0	0	0	0	1.5
Total of Theory, Practical and Mandatory Activities/Courses								26	20

Course Name: Real Time Systems
Course Code: CS801A
Contact: 3:1:0
Total Contact Hours: 48L
Credits: 4

Prerequisites:

- Concepts of Operating systems and Algorithm.
- Knowledge of Distributed System basics.

Course Objective(s):

- To understand the real-time systems.
- Obtain a broad understanding of the technologies and applications for emerging and exciting domain of real-time systems.
- Get in-depth hands-on experience in designing and developing a real time system.

Course Outcome(s):
After completion of the course students will be able to

CO1: Understand the concepts of Real-Time systems.
CO2: Recognize the characteristics of a real-time system.
CO3: Understand and develop document on an architectural design of a real-time system.
CO4: Develop and document Task scheduling, resource management, real-time operating systems and fault tolerance applications of real-time systems.

Course Content:

Module-I: Introduction [12L]
Definition, Typical Real Time Applications: Digital control, High Level Controls, Signal processing etc., Release Times, Deadline period and time constraints, Hard and soft real time systems, Reference models for RTOS: Processors and Resources, Temporal parameters of Real-time workload, Periodic Task Model, Precedence Constraints and Data Dependency.

Module-II: Real Time Scheduling [12L]
Common Approaches to Real Time Scheduling: Clock Driven Approach, Weighted Round Robin Approach, Priority Driven Approach, Dynamic Versus Static Systems, Optimality of Effective-Deadline-First (EDF) and Least-Stack-Time-First (LST) algorithms, Rate Monotonic algorithm, Offline versus Online Scheduling.

Module-III: Resources Sharing [10L]
Effect of Resource Contention and Resource Access Control (RAC), Non-pre-emptive Critical Sections, Basic Priority- Inheritance and Priority-Ceiling Protocols, Stack based Priority Ceiling Protocol, Use of Priority Ceiling Protocol in Dynamic priority systems, Pre-emption Ceiling Protocol, Access control in Multiple Module Resources, Controlling Concurrent Accesses to Data Objects.

Module-IV: Real Time Communication [8L]
Basic Concepts of Real time Communication, Soft and Hard real-time Communication systems, Model of Real-time Communication, Priority based service and Weighted Round Robin Service disciplines for switched Networks, Medium Access control protocols for broadcast networks,

Internet and resource reservation protocols

Module-V: Real Time Operating Systems and Databases [6L]

Features of RTOS, Time Services, UNIX as RTOS, POSIX Issues, Characteristic of temporal data, temporal consistency, on-currency Control, and Overview of Commercial Real Time databases.

Textbook:

1. Real Time Systems – Jane W. S. Liu, Pearson Education Publication

Reference Books:

1. Real Time Systems – Mall Rajiv, Pearson Education
2. Real-Time Systems: Scheduling, Analysis, and Verification – Albert M. K. Cheng, Wiley.

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	3	1	2	-	-	-	-	-	-	-	2	2	3
CO2	3	2	3	3	-	-	-	-	-	-	-	-	2	3	2
CO3	3	3	3	3	2	-	-	-	-	-	-	-	3	3	3
CO4	3	2	3	3	2	-	-	-	-	-	-	-	3	3	3

Course Name: Data Analytics
Course Code:CS 801 B
Contact: 3:1:0
Total Contact Hours: 48L
Credits: 4

Prerequisites: Data Structure, Design and Analysis of Algorithms, Database Management Systems, Statistics, Artificial Intelligence, Programming skills of Python.

Course Objective(s):

- To comprehend the fundamental concepts of Data Analytics, exploring machine learning strategies such as supervised and unsupervised learning for analyzing large-scale structured and unstructured data distributed across multiple locations using MapReduce, Hadoop, and NoSQL frameworks.
- To formulate an engineering problem of analyzing large-scale data distributed across multiple locations to make automated meaningful decisions.
- To apply the concepts of Data Analytics to solve problems of making automated decisions dealing with large-scale structured and unstructured data distributed across multiple locations.
- To implement ideas to address the challenging issues of Data Analytics.
- To analyze the effectiveness of various Data Analytics frameworks.

Course Outcome(s):

After completion of the course students will be able to

CO1: Understand the fundamental concepts of Big Data Analytics for making automated decisions using machine learning strategies on large-scale structured and unstructured data distributed across multiple locations using MapReduce, Hadoop, and NoSQL frameworks, emphasizing their importance in the current technological context.
CO2: Formulate an engineering problem of analysing large-scale data distributed across multiple locations to make automated meaningful decisions within the scope of Big Data Analytics frameworks.
CO3: Apply the concepts of Big Data Analytics to solve problems of automated decision-making using large-scale structured and unstructured data with MapReduce, Hadoop, and advanced SQL frameworks.
CO4: Create ideas for proposing solutions to challenging problems in Big Data Analytics.
CO5: Analyse Big Data Analytics solutions by developing feasible algorithms or frameworks and evaluating their effectiveness using appropriate performance analysis techniques.

Course Content:

Module – 1: Introduction to Basic Analytics [14L]

Introduction: Big data overview, Analyst's perspective on data repositories, Current analytical architecture, Drivers of big data, Examples of big data analytics.
Life Cycle of Data Analytics: Phase 1: Discovery, Phase 2: Data preparation, Phase 3: Model planning, Phase 4: Model building, Phase 5: Communication of results, Phase 6: Making operational.
Basic Analytic Methods: Visualization, Dirty data, Data exploration versus presentation, Statistical methods for evaluation – hypothesis testing, difference of means, rank sum test, type I and type II errors, ANOVA.

Module - 2: Advanced Analytic Methods I [12L]

Clustering: Overview, K-means, Determining the number of clusters, Diagnostics.
Association Rules: Overview, Apriori algorithm, Evaluation of candidate rules, Application of association rules, Validation and testing, Diagnostics.
Regression: Linear regression - model description, Logistic regression – model description, other regression models.
Classification: Decision trees – overview, General algorithm, Decision tree algorithms, Evaluating a decision tree, Naïve Bayes – Bayes theorem, Naïve Bayes classifier, Diagnostics of classifiers.

Module – 3: Advanced Analytic Methods II [12L]

Time Series Analysis: Overview, Box-Jenkins methodology, Autocorrelation function (ACF), Autoregressive model, Moving average model, ARMA and ARIMA model, Building and evaluating an ARIMA model.
Text Analysis: Steps in text analysis, collecting raw text, representing text, Term Frequency-Inverse Document Frequency (TFIDF), Categorizing documents by types, Determining sentiments.
Map Reduce and Hadoop: Analytics for unstructured data – map reduce, Apache Hadoop, Hadoop Ecosystem – Pig, Hive, Hbase, Mahout.

Module – 4: Advanced Analytic Methods III [10L]

Technology and Tools: SQL essentials - Join, Set, grouping extensions, Advanced SQL – Window functions, User-defined functions, Ordered aggregates, MADlib, NoSQL.
Integration of Techniques: Communicating and operationalizing an analytic project.
Creating final deliverables – Developing core materials, project goals, Main findings, Approach,

Model description and model details, Recommendations, Providing technical specifications and code.
Data visualization basics - Key points, evolution of a graph, common representation methods, how to clean up a graphic.

Text book:

1. EMC Education Services (Editor), Data Science and Big Data Analytics. John Wiley & Sons, 2015.
2. Mike Barlow, Real-Time Big Data Analytics: Emerging Architecture. O'Reilly, 2013.

Reference book:

1. Nathan Marz and James Warren, Big Data: Principles and Best Practices for Scalable Realtime Data Systems. Manning Publications, 2015.
2. Venkat Ankam, Big Data Analytics. Packt Publishing Ltd., UK, 2016.

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	3	1	-	3
CO2	2	3	-	-	-	-	-	-	-	-	-	-	1	2	3
CO3	2	2	3	2	-	-	-	-	-	-	-	-	2	2	2
CO4	2	2	2	3	-	-	-	-	-	-	-	2	3	-	-
CO5	2	2	3	3	2	2	2	-	-	-	-	2	-	-	-

Course Name: Soft computing
Course Code: CS801C
Contact: 3:1:0
Total Contact Hours: 48
Credits: 4

Prerequisites: Discrete Mathematics, Probability and Statistics
Course Objective(s):

- To understand the fundamental concepts and principles of soft computing techniques.
- To develop the ability to handle uncertainty, imprecision, and incomplete data in real-world problems.
- To apply soft computing techniques for solving complex engineering and optimization problems.

Course Outcome(s):
After completion of the course students will be able to
CO1: Understand the basic concept of soft computing and hard computing and apply them in designing solution to engineering problem.

CO2: Understand appropriate learning rules for each of the architectures and learn several neural network paradigms and its applications to solving engineering and other problems.
CO3: Apply fuzzy logic and reasoning to handle uncertainty and solving interdisciplinary engineering problems
CO4: Use genetic algorithms to combinatorial optimization problems and recognize the feasibility of applying a soft computing methodology for a particular problem.
CO5: Understand the concept and techniques of designing and implementing of soft computing methods in real world problem.

Course Content:

Module-1: Introduction to Soft Computing [12L]

An Overview of Artificial Intelligence, Evolution of Computing - Soft Computing Constituents – From Conventional Artificial Intelligence to Computational Intelligence - Machine Learning Basics. Soft Computing: Introduction of soft computing, soft computing vs. hard computing, various types of soft computing techniques, applications of soft computing

Module-2: Fuzzy sets and Fuzzy logic [10L]

Introduction, Fuzzy sets versus crisp sets, operations on fuzzy sets, Extension principle, Fuzzy relations and relation equations, Fuzzy numbers, Linguistic variables, Fuzzy logic, Linguistic hedges, Applications, fuzzy controllers, fuzzy pattern recognition, fuzzy image processing, fuzzy database.

Module -3: Artificial Neural Networks [12L]

Artificial Neural Network: Introduction, basic models, Hebb's learning, Adeline, Perception, Multilayer feed forward network. Back propagation, Different issues regarding convergence of Multilayer Perceptron, Competitive learning, Self-Organizing Feature Maps, Adaptive Resonance Theory, Associative Memories, Applications.

Module -4: Genetic Algorithms [8L]

Evolutionary and Stochastic techniques: Genetic Algorithm (GA), different operators of Genetic Algorithm, Analysis of selection operations, Hypothesis of building Blocks, Schema theorem and convergence of Genetic Algorithm, Simulated annealing and Stochastic models, Boltzmann Machine, Applications. Rough Set: Introduction, Imprecise Categories Approximations and Rough Sets, Reduction of Knowledge, Decision Tables, and Applications.

Module -5: Hybrid Systems [6L]

Neural-Network-Based Fuzzy Systems, Fuzzy Logic-Based Neural Networks, Genetic Algorithm for Neural Network Design and Learning, Fuzzy Logic controlled Genetic Algorithm. Fuzzy Logic and Genetic Algorithm for Optimization, Applications.

Text book:

1. "Neural Networks, Fuzzy logic, and Genetic Algorithms", S. Rajasekaran& G. A. V. Pai, PHI.
2. "Principles of Soft Computing", S. N. Sivanandam, S.N Deepa, Wiley publications.
3. "Neural Networks", S. Haykin, Pearson Education, 2ed, 2001.
4. "An Introduction to Genetic Algorithm", Mitchell Melanie, Prentice Hall, 1998.

Reference Books:

1. "Genetic Algorithms in Search, Optimization and Machine Learning", David E. Goldberg, Addison Wesley, 1997.

2. "Intelligent Hybrid Systems", D. Ruan, Kluwer Academic Publisher, 1997.

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	-	-	-	-	-	-	-	2	2	3	1
CO2	3	3	3	3	2	2	-	-	-	-	-	2	3	2	2
CO3	3	3	3	3	2	2	-	-	-	-	-	2	2	3	1
CO4	3	3	3	3	2	-	-	-	-	-	-	2	3	3	2
CO5	3	3	3	3	-	2	-	-	2	2	-	2	3	3	2

Course Name: VLSI Design & Application
Course Code: CS802A
Contact: 3:0:0
Total Contact Hours: 36L
Credits: 3

Prerequisite: Concept of Solid-State Devices; Analog and Digital Electronic Circuit

Course Objective(s):

- To understand the basic concepts of designing combinational and sequential circuits.
- To motivate students for designing VLSI integrated circuits in the area of digital and analog electronics.
- To encourage the design of low power but high-speed ICs.
- To study various programmable logic devices like PLDs and FPGA.

Course Outcome(s):

CO1: Explain the concepts of scale of integration, VLSI design flow, and fundamental design methodologies.
CO2: Analyse MOS device characteristics and compute relevant parameters for designing digital circuits.
CO3: Apply knowledge of IC fabrication processes to develop stick diagrams and CMOS layouts following design rules.
CO4: Utilize Hardware Description Languages (VHDL/Verilog) to model and design combinational and sequential circuits.
CO5: Interpret and apply VLSI design principles in real-world applications across modern electronic systems.

Course Content:

Module–1: Introduction to VLSI Design [8L]

Basics of VLSI and ICs, MSI, LSI**;** various types of VLSI ICs: General purpose, ASIC, PLA, FPGA**;** Basic CMOS fabrication techniques and n-well CMOS process**;** VLSI Design Concepts: Moore's Law, scale of integration (SSI, MSI, LSI, VLSI, ULSI – basic idea)**;** Design Principles of digital VLSI, Concept of regularity, modularity, and granularity. Behavioral, Structural, Physical Design Domains, Y-chart. Steps of designing digital VLSI.

Module-2: Structure of MOS [7L]

E-MOS & D-MOS structure, Charge inversion in E-MOS, Threshold voltage, Flat band voltage, Potential balance & Charge balance, Inversion, MOS capacitances**;** Three Terminal MOS

Structure: Body effect. Four Terminal MOS Transistor: Drain current, I-V characteristics. Current-voltage equations (simple derivation)**;** General, Constant Voltage & Field scaling in MOSFET; CMOS inverter, Simple Combinational Gates - NAND gate and NOR Gate using CMOS.

Module-3: Micro-electronic Processes for VLSI Fabrication [8L]
Silicon Semiconductor Technology - An Overview, Wafer processing, Oxidation, Epitaxial deposition, Ion-implantation & Diffusion, Cleaning, Etching**;** Photo-lithography – Positive & Negative, p-well CMOS process**;** Twin tub process, Silicon on insulator Layout Design Rule: Stick diagram with examples, Layout rules.

Module-4: Hardware Description Language [6L]
Basics of Hardware Description Languages, need and advantages of HDL, comparison of VHDL and Verilog**;** Design units, entity–architecture (VHDL) / module (Verilog), data types, operators**;** Behavioural, Dataflow, and Structural modelling techniques**;** Implementation of basic combinational circuits such as adders, multiplexers, encoders, and decoders**;** Design of flip-flops, registers, counters, and finite state machines (FSM)**;** Test benches, simulation process, and basics of synthesis for digital circuit implementation.

Module-5: Applications of VLSI [7L]
Introduction to VLSI applications in modern electronics, Application areas: Consumer electronics, communication systems, automotive electronics, biomedical devices, and embedded systems**;** Role of VLSI in microprocessors, microcontrollers, and System-on-Chip (SoC) design**;** Low-power and high-performance VLSI applications**;** Emerging trends: AI hardware accelerators, IoT devices, and nanotechnology in VLSI.

Text book:

1. Digital Integrated Circuit, J. M. Rabaey, Chandrakasan, Nicolic, Pearson Education
2. CMOS Digital Integrated Circuits Analysis and Design, S. M. Kang & Y. Leblebici, TMH.

Reference book:

1. Microelectronic Circuits, Sedra & Smith, Oxford
2. Introduction to VLSI Circuits and System, Uyemura, Wiley

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3	1	1	-	-	1	2	1	1	1	2	2	3
CO2	3	2	1	2	1	-	-	2	2	2	1	2	2	3	3
CO3	3	3	3	2	1	-	-	1	1	1	2	2	3	3	3
CO4	2	2	1	1	1	-	-	1	2	2	1	1	3	2	3
CO5	2	3	1	3	1	-	-	1	2	2	1	2	3	3	3

Course Name: Bio-informatics
Course Code: CS802B

Contact: 3:1:0
Total Contact Hours: 48L
Credits: 4

Prerequisites: Bio-informatics at UG level

- Basic knowledge of biology and computer science fundamentals, including genetics, molecular biology, and introductory programming concepts, is required.
- Familiarity with mathematics and statistics, such as probability, data analysis, and basic algorithms, is essential for understanding bioinformatics techniques.

Course Objective(s):

- To develop foundational knowledge of biological systems and computational techniques, enabling students to understand and analyze biological data using algorithms and software tools.
- To equip students with practical skills in bioinformatics tools, databases, and programming (such as sequence alignment, genome analysis, and data mining) for solving real-world biological problems.
- To foster interdisciplinary problem-solving abilities by integrating concepts from biology, computer science, and statistics to support research in areas like genomics, proteomics, and drug discovery.

Course Outcome(s):

After completion of the course students will be able to

CO1: Acquire the knowledge of Bioinformatics technologies with the related concept of DNA, RNA and their implications
CO2: Develop idea in MOLECULAR BIOLOGY
CO3: Understand the concept and techniques of different types of Data Organization and Sequence Databases with different types of Analysis Tools for Sequence Data Banks
CO4: Acquire the knowledge of the DNA SEQUENCE ANALYSIS
CO5: Analyse the performance of different types of Probabilistic models used in Computational Biology.

Course Content:

Module 1: [12L]

Introduction to Molecular Biology: Concepts of Cell, tissue, types of cells, components of cell, organelle. Functions of different organelles. Concepts of DNA: Basic Structure of DNA; Double Helix structure; Watson and crick model. Exons and Introns and Gene Concept.
Concepts of RNA: Basic structure, Difference between RNA and DNA. Types of RNA. Concept of Protein: Basic components and structure.
Introduction to Central Dogma: Transcription and Translation, Introduction to Metabolic Pathways. Introduction to Bioinformatics. Recent challenges in Bioinformatics.

Module 2: [14L]

Introduction to Genomic data, Data Organization and Sequence Databases: Sequence Data Banks

- Introduction to sequence data banks - protein sequence data bank. Signal peptide data bank, Nucleic acid sequence data bank - GenBank, AIDS virus sequence data bank. RRNA data bank, structural data banks - protein Data Bank (PDB), The Cambridge Structural Database (CSD): Genome data bank - Metabolic pathway data: Microbial and Cellular Data Banks.
Introduction to MSDN (Microbial Strain Data Network): Numerical Coding Systems of Microbes, Hibridoma Data Bank Structure, Virus Information System Cell line information system; Protein Sequence Databases, DNA sequence databases. sequence database search programs like BLAST and FASTA. NCBI different modules: GenBank; OMIM, Taxonomy browser, PubMed.

Module 3: [12L]

DNA SEQUENCE ANALYSIS DNA Mapping and Assembly: Size of Human DNA, Copying DNA: Polymerase Chain Reaction (PCR), Hybridization and Microarrays, Cutting DNA into Fragments, Sequencing Secondary Structure predictions; prediction algorithms; Chao-Fasman algorithm, Hidden-Markov model, Neural Networking. Tertiary Structure predictions; prediction algorithms; Chao-Fasman algorithm, Hidden-Markov model, Neural Networking.

Module 4: [10L]

Introduction Probabilistic models used in Computational Biology: Probabilistic Models.
Gene Regulatory Method Application of HMM in Bioinformatics: Gene finding, profile searches, multiple sequence alignment and regulatory site identification.
Applications in Biotechnology: Protein classifications, Fold libraries, Protein structure prediction: Fold recognition (threading),
Protein structure predictions: Comparative modeling (Homology), Advanced topics: Protein folding, Protein-ligand interactions, Molecular Modeling & Dynamics, Drug Designing.

Text Book:

1. Yi-Ping Phoebe Chen (Ed), "BioInformatics Technologies", First Indian Reprint, Springer Verlag, 2007.

References Book:

1. Bryan Bergeron, "Bio Informatics Computing", Second Edition, Pearson Education, 2003.
2. Arthur M Lesk, "Introduction to Bioinformatics", Second Edition, Oxford University Press, 2005.

CO–PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	2	2	2	-	-	-	-	-	-	1	2	2	3
CO2	3	2	2	3	2	-	-	-	-	-	-	1	2	3	3
CO3	2	2	2	3	2	-	-	-	-	-	-	1	2	2	3
CO4	2	3	3	2	2	-	-	-	-	-	-	1	2	3	3
CO5	3	3	3	3	3	-	-	-	-	-	-	2	3	3	3

Course Name: Robotics
Course Code: CS802C

Contact: 3:1:0
Total Contact Hours: 36L
Credits: 4

Prerequisites: Microprocessor & Microcontroller Computer Organization & Architecture.

Course Objective(s):

- To study microcontroller operations for robotics.
- To study how different interfaces are implemented in a microcontroller.
- To learn how Microchip PIC micro PIC16F627 can be erased and reprogrammed.
- To learn how different sensors, outputs, and peripherals can be wired to a microcontroller to work cooperatively and create a high-level control program.
- To design robots in a real time environment.

Course Outcome(s):
After completion of the course students will be able to

CO1: Describe and explain the microcontrollers used the in robots.
CO2: Design the software and build the prototype of robots.
CO3: Apply localization and mapping aspects of mobile robotics.
CO4: Demonstrate self-learning capability.

Course Content:
Module I [5L]
Brief history, types, classification and usage, Science and Technology of robots, Some useful websites, textbooks and research journals. Position and orientation of a rigid body, Homogeneous transformations, Representation of joints, link representation using D-H parameters, Examples of D-H parameters and link transforms, different kinds of actuators – stepper, DC servo and brushless motors, model of a DC servo motor, Types of transmissions, Purpose of sensors, internal and external sensors, common sensors – encoders, tachometers, strain gauge based force-torque sensors, proximity and distance measuring sensors, and vision.
Module II [8L]
Introduction, Direct and inverse kinematics problems, Examples of kinematics of common serial manipulators, workspace of a serial robot, Inverse kinematics of constrained and redundant robots, Tractrix based approach for fixed and free robots and multi-body systems, simulations and experiments, Solution procedures using theory of elimination, Inverse kinematics solution for the general 6R serial manipulator. Degrees-of-freedom of parallel mechanisms and manipulators, Active and passive joints, Constraint and loop-closure equations, Direct kinematics problem, Mobility of parallel manipulators, Closed-from and numerical solution, Inverse kinematics of parallel manipulators and mechanisms, Direct kinematics of Gough-Stewart platform.
Module III [8L]
Linear and angular velocity of links, Velocity propagation, Manipulator Jacobians for serial and parallel manipulators, Velocity ellipse and ellipsoids, Singularity analysis for serial and parallel manipulators, Loss and gain of degree of freedom, Statics of serial and parallel manipulators, Statics and force transformation matrix of a Gough-Stewart platform, Singularity analysis and statics. Mass and inertia of links, Lagrangian formulation for equations of motion for serial and

parallel manipulators.

Module IV [9L]

Joint and Cartesian space trajectory planning and generation, Classical control concepts using the example of control of a single link, Independent joint PID control, Control of a multi-link manipulator, non-linear model-based control schemes, Simulation and experimental case studies on serial and parallel manipulators, Control of constrained manipulators, Cartesian control, Force control and hybrid position/force control, Advanced topics in non-linear control of manipulators.

Module V [6L]

Introduction and some well-known wheeled mobile robots (WMR), two and three-wheeled WMR on flat surfaces, Slip and its modelling, WMR on uneven terrain, Design of slip-free motion on uneven terrain, Kinematics, dynamics, and static stability of a three-wheeled WMR 's on uneven terrain, Simulations using MATLAB and ADAMS. Introduction to chaos, non-linear dynamics and chaos in robot equations, Simulations of planar 2 DOF manipulators, Analytical criterion for unforced motion. Gough-Stewart platform and its singularities, use of near singularity for fine motion for sensing, design of GoughStewart platform-based sensors. Over-constrained mechanisms and deployable structures, Algorithm to obtain redundant links and joints, Kinematics and statics of deployable structures with pantographs or scissor-like elements (SLE's).

Text book:

1. Myke Predko, ―Programming Robot Controllers‖ – McGrawHill, 1st edition, 2003.
2. J. Craig**,** "Introduction to Robotics: Mechanics and Control", Pearson Education.

Reference book:

1. Michael slater, ―Microprocessor – based design: A comprehensive Guide to Effective Hardware Design, Prentice Hall, 1989.
2. Myke Predko, ―Programming and customizing the 8051- micro-controller‖, Tata McGraw-Hill, New Delhi, 2000.

CO–PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	2	-	-	-	-	-	-	1	2	2	3
CO2	3	3	3	3	3	-	-	-	-	-	-	-	2	3	3
CO3	3	3	3	3	3	-	-	-	-	-	-	-	3	3	3
CO4	3	3	3	3	2	-	-	-	-	-	-	1	3	2	3
CO5	3	3	3	3	3	-	-	-	-	-	-	-	3	3	3

Course Name: Introduction to IoT
Course Code: CS803A
Contact: 3:0:0
Total Contact Hours: 36L
Credits: 3

Prerequisites:

- Basic knowledge of computer networks (TCP/IP, protocols)

- Fundamentals of embedded systems / microcontrollers
- Basic programming (C/Python)
- Introductory understanding of data communication

Course Objectives:

- To introduce the fundamental concepts and architecture of IoT systems.
- To understand sensors, actuators, and embedded platforms used in IoT.
- To explore communication protocols and networking in IoT environments.
- To develop basic IoT applications using hardware and software integration.
- To understand security, privacy, and real-world IoT applications.

Course Outcomes (COs):

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

CO1: Explain IoT architecture, components, and enabling technologies.
CO2: Analyze sensors, actuators, and embedded platforms used in IoT systems.
CO3: Apply IoT communication protocols and networking concepts.
CO4: Design and develop basic IoT-based applications.
CO5: Evaluate IoT systems considering security, scalability, and real-world constraints.

Course Content:

Module I: Introduction to IoT [6L]

Definition and characteristics of IoT. IoT ecosystem and architecture (3-layer, 5-layer models). Enabling technologies: RFID, WSN, cloud computing. IoT vs M2M. Applications: Smart homes, smart cities, healthcare

Module II: Sensors, Actuators & Embedded Systems [6L]

Types of sensors (temperature, humidity, motion, light). Actuators and control systems. Microcontrollers and platforms (Arduino, Raspberry Pi). Interfacing sensors with microcontrollers

Module III: IoT Communication & Networking [8L]

IoT communication models. Protocols: MQTT, CoAP, HTTP/REST. Wireless technologies: Wi-Fi, Bluetooth, Zigbee, LoRa WAN. IPv6 and 6LoWPAN

Module IV: IoT Data Processing & Cloud Integration [6L]

Data acquisition and preprocessing. Cloud platforms for IoT (AWS IoT, Azure IoT basics). Edge vs Cloud computing. Data analytics basics in IoT

Module V: IoT Security & Privacy [4L]

Security challenges in IoT. Authentication and encryption. Secure communication protocols. Privacy issues and data protection

Module VI: IoT Applications & Case Studies [6L]

Smart home systems. Smart agriculture. Industrial IoT (IIoT). Healthcare IoT. Real-world case studies and future trends

Text book:

1. Internet of Things: A Hands-on Approach by Arshdeep Bahga and Vijay Madisetti
2. Internet of Things Principles and Paradigms, Editors: Rajkumar Buyya, Amir Vahid Dastjerdi.

Reference book:

1. Al-Fuqaha, Ala, et al. "Internet of things: A survey on enabling technologies, protocols, and applications." IEEE communications surveys & tutorials 17.4 (2015): 2347-2376.
2. Atzori, Luigi, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey." Computer networks 54.15 (2010): 2787-2805.

CO–PO-PSO Mapping:

CO \ PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	–	–	–	–	1	–	2	3	1	1
CO2	3	3	2	2	2	–	–	–	1	1	–	2	3	2	2
CO3	3	3	2	2	3	–	–	–	1	1	–	2	3	2	2
CO4	3	3	3	2	3	1	1	–	2	2	1	2	3	3	3
CO5	2	3	2	3	3	2	2	1	1	2	1	3	2	2	3

Course Name: Image Processing
Course Code: CS803B
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite:

- Fourier analysis
- Linear algebra
- Probability

Course Objective(s):

- To learn discrete Fourier, transform and its properties
- To study the monochrome and color image fundamentals
- To learn the analytical tools and methods which are currently used in digital image processing as applied to image information for human viewing
- To learn image compression and segmentation techniques.

Course Outcomes:
After completion of course, students would be able to

CO1: Acquire the knowledge of basic pre-processing techniques in monochrome and color images.
CO2: Develop skill in concepts of image enhancement like linear and nonlinear spatial filters using MATLAB.
CO3: Understand the concept and techniques of simple image processing projects using different methods of restoration.
CO4: Acquire the knowledge of the various segmentation algorithms for practical applications.
CO5: Analyze the performance of Lossless and Lossy compression techniques in images.

Course Contents:

Module 1: Introduction [5L]

Digital Image Fundamentals: Overview, Computer imaging systems, Digital Image Representation, Fundamental steps in Image Processing**,** Elements of Digital Image Processing - Image Acquisition, Storage, Processing, Communication, Display**.** Digital Image Formation: A Simple Image Model, Use and Analysis of Color Models in Image Processing**,** Sampling & Quantization - Uniform & Non-uniform**.**

Module 2: Mathematical Preliminaries [5L]

Neighbor of pixels, Connectivity, Relations, Equivalence & Transitive Closure**;** Distance Measures, Arithmetic/Logic Operations, Discrete Signals and Systems - A Review – Fourier Transformation, Properties of The Two-Dimensional Fourier Transform**,** Discrete Fourier Transform, Discrete Cosine & Sine Transform**.**

Module 3: Image Enhancement [6L]

Spatial Domain: Gray level transformations – Histogram processing, Basics of Spatial Filtering– Smoothing and Sharpening Spatial Filtering – Frequency Domain **–** Smoothing and Sharpening frequency domain filters – Ideal, Butterworth and Gaussian filters**.**

Module 4: Image Restoration, Segmentation and Filtering [7L]

Image Restoration and Segmentation: Image restoration: noise removal: mean & adaptive filters, degradation model, inverse filter**.** Discrete Formulation, Algebraic Approach to Restoration Unconstrained& Constrained**;** Constrained Least Square Restoration, Restoration by Homomorphic Filtering**,** Geometric Transformation - Spatial Transformation, Gray Level Interpolation**.** Image Segmentation: Point Detection, Line Detection, Edge detection, Combined detection.

Module 5: Edge Linking, Boundary Detection and Image compression [8L]

Edge Linking & Boundary Detection- Local Processing, Global Processing via The Hough Transform; Thresholding - Foundation, Simple Global Thresholding, Optimal Thresholding**;** Region Oriented Segmentation - Basic Formulation, Region Growing by Pixel Aggregation, Region Splitting & Merging**,** Image compression: system model, lossless methods, lossy methods.

Module 6: Image Representation and Recognition [5L]

Image Representation and Recognition: Boundary representation – Chain Code – Polygonal approximation, signature, boundary segments – Boundary description – Shape number- Fourier Descriptor**,** moments- Regional Descriptors –Topological feature**,** Texture – Patterns and Pattern classes – Recognition based on matching**.**

Text Books:

1. Chanda & Majumder, Digital Image Processing & Analysis, PHI

Reference books:

1. Malay K. Pakhira, Digital Image Processing and Pattern Recognition, First Edition, PHI Learning Pvt. Ltd., 2011.
2. Rafael C. Gonzales and Richard E. Woods, Digital Image Processing, Third Edition, Pearson Education, 2010.

CO-PO-PSO Mapping:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	-	-	2	-	1	1	-	-	-	-	-	3	3	1
CO2	-	1	2	1	-	-	-	-	-	-	-	1	3	3	1
CO3	1	2	-	3	2	-	-	-	1	-	-	-	3	3	3
CO4	2	-	-	-	-	1	-	-	-	1	1	-	3	3	3
CO5	-	3	-	2	-	1	-	1	-	-	-	-	3	3	3

Course Name: Optimization Techniques
Course Code: CS803C
Contact: 3:0:0
Total Contact Hours: 36
Credits: 3

Prerequisite:
Basic Knowledge of Function, plotting of Equation and inequations, Formulation of Mathematical Problem. Finding maximum and minimum from row or column or from Matrix.

Course Objective:
Purpose of this course is to develop models and then analyze the model using the techniques of Operations Research, Decision making under uncertainty and risk.

Course Outcomes(s):
On successful completion of the learning sessions of the course, the learner will be able to:

CO1: Recall the distinctive characteristics of different types of decision-making problem to formulate and solve a real-world problem a prototype of mathematical problem.
CO2: Understand the theoretical workings of appropriate decision-making approaches and tools to identify the optimal strategy in competitive world.
CO3: Apply the principles of different Methods/Model of Operations Research to solve practical problems.
CO4: Analyze different engineering problems linked with Optimization Technique.
CO5: Design optimal decision-making models for real-life engineering and management problems.

Course Content:
Module 1 [10L]
Linear Programming Problem (LPP): Basics of Linear Programming Problem (LPP) and its Applications. General Mathematical Formulation of LPP; Definitions: Convex set, Solution, Feasible Solution, Basic and Non-Basic Variables, Basic Feasible Solution, Degenerate and Non-Degenerate solution, Optimum/Optimal Solution; Solution of LPP by Graphical Analysis/Method, Simplex Method, Charnes' Big M-Method; Duality Theory.

Module 2 [4L]
Transportation Problem, Assignment Problem.
Module 3 [6L]
Game Theory: Introduction**;** Two person Zero Sum game, Saddle Point; Mini-Max and Maxi-Min Theorems (statement only) and problems; Games without Saddle Point; Graphical Method; Principle of Dominance.
Module 4 [5L]:
Network Optimisation Models: CPM PERT (Arrow network), Time estimates, earliest expected time, latest allowable occurrence time, latest allowable occurrence time and stack. Critical path, Probability of meeting scheduled date of completion of project. Calculation of CPM network.Various floats for activities.
Module 5 [3L]
Sequencing: Johnson's Algorithm (1957) For n Jobs and two machines, n Jobs and three machines.

Module 6 [5L]
Queuing Theory: Introduction and Basic Structure of Queuing Theory; Basic Definations and Notations; Birth-and-Death Model (Poisson / Exponential distribution); Poisson Queue Models: (M/M/1) : (∞/FIFO) and (M/M/1) : (N/FIFO) and Problems.
Module 7 [3L]
Inventory Control: Determination of EOQ, Components, Deterministic Continuous & Deterministic Periodic Review Models, Stochastic Continuous & Stochastic Periodic Review Models.

Project Domain:

1. Study on Real life Problem Solve by Simplex Method.
2. Study on Real life Distribution Problem Solve by Transportation Problem.
3. Study on Real life Game / Marketing Problem Solve by Game Theory.
4. Study on Real life Network / Queueing Problem Solve by PERT/CPM and Queueing Theory.
5. Study on Real life Inventory Control Problem Solve by Inventory Control.

Text Books:

1. Operations Research by Kanti Swaroop and P.K. Man Mohan, Sultan Chand and Sons
2. Linear Programming and Theory of Games by Ghosh and Chakraborty, Central Book Agency

Reference Books:

1. Operations Research Theory and Applications by J.K.Sharma, Macmillan India Limited.
2. Operations Research, Vijayakumar, Scitech

CO-PO-PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	-	-	-	-	-	-	-	1	2	3	1
CO2	3	2	2	1	2	-	-	-	-	-	-	1	2	3	1
CO3	3	2	2	1	-	-	-	-	-	-	-	1	3	3	2
CO4	2	3	1	1	-	-	-	-	-	-	-	1	3	3	2

CO5	3	3	2	1	-	-	-	-	-	-	-	2	3	3	2

Course Name: Internet of Things Lab
Course Code: CS893A
Contact: 0:0:3
Credits: 1.5

Prerequisite:

- Basic programming knowledge (C/C++/Python)
- Fundamentals of embedded systems and computer networks
- Basic understanding of sensors and microcontrollers

Course Objective(s):

- To understand practical implementation of IoT systems using sensors and microcontrollers
- To develop skills in interfacing hardware with software platforms
- To learn IoT communication protocols and cloud integration
- To design and implement real-time IoT applications
- To understand security and data handling in IoT systems

Course Outcomes:
On completion of the course students will be able to

CO1: Understand and implement basic IoT system architecture including sensors, actuators, and microcontrollers.
CO2: Interface sensors with embedded platforms and acquire real-time data.
CO3: Apply IoT communication protocols (MQTT/HTTP) for data transmission.
CO4: Design and develop IoT applications with cloud/edge integration.
CO5: Analyze IoT systems for performance, security, and real-world deployment.

Experiments:

1. Introduction to IoT hardware: Arduino/Raspberry Pi setup and basic programming
2. Interfacing temperature and humidity sensors (DHT11/DHT22)
3. Interfacing motion/light sensors and actuators (LED, buzzer)
4. Serial communication and data acquisition from sensors
5. IoT data transmission using Wi-Fi module (ESP8266/ESP32)
6. Implementation of MQTT protocol for IoT communication
7. Sending sensor data to cloud platforms (ThingSpeak/AWS IoT)
8. Real-time IoT dashboard for monitoring sensor data
9. Smart home automation (control appliances via mobile/web)
10. Edge vs Cloud processing experiment (local vs cloud computation)
11. Basic IoT security implementation (authentication/encryption demo)
12. Mini-project: Design and implement a complete IoT application (e.g., smart agriculture / smart parking)

CO–PO–PSO Mapping:

CO \ PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	–	–	2	1	1	2	–	–	–	–	–	3	2	2
CO2	2	2	3	1	2	–	–	–	1	–	–	–	3	2	2
CO3	1	2	2	1	3	–	–	–	1	1	–	–	3	2	3
CO4	2	3	3	2	3	1	1	–	2	2	1	2	3	3	3
CO5	2	3	2	3	3	2	2	1	1	2	1	3	2	3	3

Course Name: Image Processing Lab
Course Code: CS893B
Contact: 0:0:3
Credits: 1.5

Prerequisite: Should have prior knowledge on syntaxes of programming like C++, JAVA.

Course Objective(s):

- To learn discrete Fourier, transform and its properties
- To study the monochrome and color image fundamentals
- To learn the analytical tools and methods which are currently used in digital image processing as applied to image information for human viewing.
- To learn image compression and segmentation techniques.

Course Outcomes:
On completion of the course students will be able to

CO1: Acquire the fundamental concepts of a digital image processing system such as image acquisition, enhancement, segmentation, transforms, compression, morphology, representation and description.
CO2: Analyze images in the spatial domain.
CO3: Analyze images in the frequency domain through the Fourier transform. R21 B.Tech.
CO4: Design and implement with MATLAB algorithms for digital image processing operations such as point processing, histogram processing,
CO5: Spatial and frequency domain filtering, denoising, transforms, compression, and morphological processing.

Experiments:

1. W.A.P in MATLAB to extract different attributes of an Image.
2. W.A.P in MATLAB program for Image Negation.
3. W.A.P in MATLAB for Power Law Transformation.
4. W.A.P in MATLAB for Histogram Mapping and Equalization.
5. W.A.P in MATLAB for Image Smoothening and Sharpening.
6. W.A.P in MATLAB for Edge Detection using Sobel, Prewitt and Roberts Operators.
7. W.A.P in MATLAB for Morphological Operations on Binary Images.
8. W.A.P in MATLAB for Pseudo Coloring of images.
9. W.A.P in MATLAB for Chain Coding applied on images.

10. W.A.P in MATLAB for DCT/IDCT Computation.

CO-PO-PSO Mapping:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	-	-	2	-	1	3	-	-	-	-	-	3	3	3
CO2	-	1	3	1	-	-	-	-	-	1	-	-	3	3	3
CO3	1	2	-	-	-	3	-	-	1	-	2	-	3	3	3
CO4	2	-	2	-	-	1	-	-	-	1	1	-	3	3	3
CO5	-	2	-	3	-	1	-	1	-	1	-	-	3	3	3

Course Name: Optimization Techniques Lab
Course Code: CS893C
Contact: 0:0:3
Credits: 1.5

Prerequisite:

- Basic knowledge of Mathematics (Linear Algebra and Calculus)
- Basic programming knowledge (preferably Python)

Course Objective(s):

- To introduce the optimization techniques using both linear and non-linear programming.
- To focus on the convex and non-convex function.
- To frame engineering minima-maxima problems in the framework of optimization problems.

Course Outcome(s):
After the completion of the course, the students will be able to

CO1: Demonstrate the basic principles and concepts of Python
CO2: Explore the applicability of programming skills in Python.
CO3: Summarize various optimization techniques like LPP models.
CO4: Analyse the transportation, inventory and assignment problems.
CO5: Explain the concepts of sequencing, game theory and dynamic programming.

List of Experiments (Includes but Not Limited to)

1. Matrix Operations
2. Minimum Cost Path
3. Finding Maximum Number in An Array
4. Array Sorting
5. Linear Programming Problem
6. Queuing Problem

7. Sequencing Problem
8. Game Theory
9. Assignment Problem
10. Dynamic Programming Problem
11. Inventory Problem

Text Books:

1. Foulds, L. R. (2012). Optimization techniques: an introduction. Springer Science & Business Media.
2. Onwubolu, G. C., &Babu, B. V. (2013). New optimization techniques in engineering (Vol. 141). Springer.

Reference Books:

16. Lopez, C. (2014). MATLAB optimization techniques. Apress.

CO-PO-PSO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	-	-	-	-	-	-	-	-	-	-	2	1	2
CO2			-	2	3	-	-	-	-	-	-	-	3	2	3
CO3	3	3	-	-	-	-	-	-	-	-	-	-	2	3	1
CO4	1	3	-	3	3	-	-	-	-	-	-	-	2	3	1
CO5	1	3	-	2	2	-	-	-	-	-	-	-	2	3	1