SISTER NIVEDITA UNIVERSITY

DG 1/2 New Town, Kolkata – 700156
www.snuniv.ac.in

SCHOOL OF ENGINEERING

Department of Computer Science & Engineering
Bachelor of Technology (B. Tech)

REGULATIONS (R23) [NEP]

Credit Definition

Type	Duration (in Hour)	Credit
Lecture (L)	1	1
Tutorial (T)	1	1
Practical (P)	2	1

Total Credit Distribution

Semester	Credits										Credits/Semester
	MC	ME	Project	NM	NV	MDC	AEC	SEC	VAC	INT	
1	12	0	0	4	2	0	2	0	2	0	22
2	16	0	0	0	1	4	2	0	2	0	25
3	15	0	0	0	2	3	2	3	0	0	25
4	10	4	0	0	1	2	2	3	2	0	25
5	14	4	0	4	3	0	0	0	0	0	24
6	5	8	0	4	3	0	0	3	0	0	23
7	0	4	4	4	0	0	0	0	0	4	16
8	0	0	8	4	0	0	0	0	0	0	12
Credits/Course	**72**	**20**	**12**	**20**	**12**	**9**	**8**	**9**	**6**	**4**	**172**

Category Definition

Definition of Category/Type	Abbreviation
Major Compulsory	**MC**
Major Elective	**ME**
Non-Major Specific Subject Course	**NM**
Non-major Vocational Education and Training	**NV**
Multidisciplinary Courses	**MDC**
Ability Enhancement Courses	**AEC**
Skill Enhancement Courses	**SEC**
Value Added Courses	**VAC**
Internship	**INT**

FIRST YEAR

SEMESTER-I

Sl No	Course Title	Code	Type	Credit	Type		
					L	T	P
1	Discrete Mathematics		MC	3	3	0	0
2	Fundamentals of Computer Science & Problem Solving		MC	4	4	0	0
3	Digital Electronics		MC	3	3	0	0
4	Probability and Statistics		NM	4	4	0	0
5	Soft-Skill Development-I		NV	1	1	0	0
6	Anyone (Sports/Yoga/NCC/NSS) EAA-I		NV	1	0	0	2
7	Communicative English-I		AEC	2	2	0	0
8	Environmental Science-I		VAC	2	2	0	0
9	Fundamentals of Computer Science & Problem-Solving Lab		MC	1	0	0	2
10	Digital Electronics Lab		MC	1	0	0	2
Total Credit				**22 Credit**			

DISCRETE MATHEMATICS

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Discrete Mathematics	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 1st

THEORY

Learning objectives: *On completion of the course, student will be able to: apply the knowledge of graph theory to solve complex engineering problem.*

Prerequisite: *Before learning the concepts of Discrete Mathematics, you should have a basic knowledge of set, relation, mapping, matrix etc.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Boolean Algebra	4	
Module-II: Abstract Algebra	6	
Module-III: Combinatories	6	
Module-IV: Fundamental concepts of Graph Theory	6	
Module-V: Tree and Network flow	6	

Module VI: Logic	6	

SYLLABUS OUTLINE:

Module I: Boolean algebra:[4L]

Introduction of Boolean algebra, truth table, basic logic gate, basic postulates of Boolean algebra, principle of duality, canonical form, Karnaugh map.

Module II: Abstract algebra:[6L]

Set, Functions, relation, Partially ordered sets, lattice, distributive and complete lattices, group, ring, field.

Module III: Combinatorics: [6L]

Pascal Triangle, Basic counting, balls and bins problems, generating functions, recurrence relations. Principle of mathematical induction, pigeonhole principle. Principle of inclusion and exclusion.

Module IV: Fundamental concepts of Graph Theory: [6L]

Graphs and digraphs, complement, isomorphism, connectedness and reachability, adjacency matrix, Eulerian paths and circuits in graphs and digraphs, Hamiltonian paths and circuits in graphs and tournaments, Planar graphs, Euler's formula, dual of a planer graph, independence number and clique number, chromatic number, statement of Four-color theorem.

Module V: Tree and Network flow: [6L]

Basics: equivalent characterizations of trees, forests , Spanning trees and 2-switches, Distance and center ,Optimization: Kruskal's Theorem and Dijkstra's Theorem

Network flow, Max-flow Min-cut theorem (statement only); Ford and Fulkerson algorithm.

Module VI: Logic: [6L]

Propositional calculus - propositions and connectives, syntax; Semantics - truth assignments and truth tables, validity and satisfiability, tautology; Adequate set of connectives; Equivalence and normal forms; Compactness and resolution; Formal reducibility - natural deduction system and axiom system; Soundness and completeness. Distributive and complete lattices.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. *Topics in Algebra,* I. N. Herstein, John Wiley and Sons.
2. *Digital Logic & Computer Design,* M. Morris Mano, Pearson.
3. *Elements of Discrete Mathematics,* (Second Edition) C. L. LiuMcGraw Hill, New Delhi.
4. *Graph Theory with Applications,* J. A. Bondy and U. S. R. Murty, Macmillan Press, London.
5. *Mathematical Logic for Computer Science,* L. Zhongwan, World Scientific, Singapore.

Reference Books:

1. *Introduction to linear algebra.* Gilbert Strang.
2. *Introductory Combinatorics,* R. A. Brualdi, North-Holland, New York.
3. *Graph Theory with Applications to Engineering and Computer Science,* N. Deo, Prentice Hall, Englewood Cliffs.
4. *Introduction to Mathematical Logic,* (Second Edition), E. Mendelsohn, Van-Nostrand, London.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
XX.CO1	3	2	-	-	-	-	-	-	-	-	-	1
XX.CO2	3	2	-	-	-	-	-	-	-	-	-	-
XX.CO3	2	1	-	-	-	-	-	-	-	-	-	-
XX.CO4	2	1	-	-	-	-	-	-	-	-	-	1
XX.CO5	2	1	-	-	-	-	-	-	-	-	-	-
XX.CO6	3	2	-	-	-	-	-	-	-	-	-	1
Avg	3	2	-	-	-	-	-	-	-	-	-	1

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

XX.CO1: Understand the fundamentals of Propositional Logic

XX.CO2: Identify truth tables and logical operators to analyse problems.

XX.CO3: Understand the fundamental theorems of Group theory.

XX.CO4: Understand the fundamental concepts in graph theory.

XX.CO5: Apply the knowledge of Boolean algebra in switching circuits.

XX.CO6: Use Max-flow Min-cut theorem, Ford and Fulkerson algorithm to design complex engineering problems.

Fundamentals of Computer Science & Problem Solving

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Fundamentals of Computer Science & Problem Solving	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 1st

THEORY

- ***Learning objectives:*** *On completion of the course student will be able to: Understand and use various constructs of the programming language such as conditionals, iteration, and recursion. Develop simple C programs to illustrate the applications of arrays, pointers, functions. This course is intended for students to implement algorithm to build C-programs.*

Prerequisite: Basic Mathematics and analytics

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: General problem Solving concepts	8	14%
Module-II: Operators & Expressions	8	15%
Module-III: Control and Iterative Flow	6	15%
Module-IV: Functions and Program Structure with discussion on standard library:	10	20%
Module-V: Pointers and Arrays:	8	18%
Module-VI: User defined data types	8	18%

SYLLABUS OUTLINE:

Module-I: General problem Solving concepts [8L]

Basic Anatomy of Computer System, Primary & Secondary Memory, Processing Unit, Input & Output device. Number Systems: Binary, Octal, Decimal, and Hexadecimal.

Problem Solving approach: Algorithm & Flow charts, formulate simple algorithm for arithmetic and logical problems. Creating and Running Programs.

Module-II: Operators & Expressions [**8L**]

Variable Names, Data Type and Sizes (Little Endian Big Endian), Constants, Declarations, Arithmetic Operators, Relational Operators, Logical Operators, Type Conversion, Increment Decrement Operators, Bitwise Operators, Assignment Operators and Conditional Operators. Expressions, Precedence and Order of Evaluation, proper variable naming and Hungarian Notation.

Module-III: Control and Iterative Flow [6L]

Statements and Blocks, if-else, switch-case, Loops – while, do-while, for, break and continue, structured and unstructured programming.

Module-IV: Functions and Program Structure with discussion on standard library:[**10L**]

Basic of functions, function prototypess, function definition, function returning values, functions not returning values, auto, external, static and register variables, scope rules, C pre-processor, command line arguments.

Module-V: Pointers and Arrays: [**8L**]

Pointers and address, Pointers and Function Arguments, Pointers and Arrays, Address Arithmetic, Pointer to an Arrays, Array of Pointers, Pointer to Pointer, Multi-dimensional array and Row/column major formats, Command line arguments, Pointer to functions, Dynamic memory allocation.

Module-VI: User defined data types [8L]

Basic Structures, Structures and Functions, Array of structures, Pointer to structures, Self-referral structures, typedef, unions, Bit-fields. Enumurated data types.

Module-VII: Input and Output (Extra) [4L]

Standard I/O, Formatted Output – printf, Formated Input – scanf, Variable length argument list, file access including FILE structure, fopen, stdin, sdtout and stderr, Debugging, Macro, User Defined Header, User Defined Library Function, makefile utility.

Pedagogy for Course Delivery: *Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA):*NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Yashwant Kanetkar Let us C, 6th Edition , BPB publication
2. Byron S Gottfried “Programming with C” Second edition, Tata McGrawhill, 2007 (Paper back)
3. E. Balagurusamy Programming in ANSI C, 5th Edition, Tata McGraw-Hill Publications

Reference Books:

1. Kerningham Dennis Ritchie The C programming language (ANSI C version), 2nd Edition, PHI India
2. Jeri R Hanly Elliot B Koffman Problem solving and program design in C Person Addison Wesley 2006

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	-	-	-	-	-	-	-		1
CO2	3	3	3	-	-	-	-	-	-	-	-	1
CO3	3	3	3	-	-	-	-	-	-	-	-	1
CO4	3	3	3	-	-	-	-	-	-	-	-	1
CO5	3	3	3	-	-	-	-	-	-	-	-	1
CO6	3	3	3	-	-	-	-	-	-	-	-	1
Avg												1

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

1XXXXX. CO1: To be able to develop an algorithm for solving a problem. [BT3]

1XXXXX. CO2: To be able to explain the utility of operators in C. [BT2]

1XXXXXX. CO3: To be able to make use of control statements for solving the related problems. [BT3]

1XXXXXX. CO4: To be able to utilize the concept of user defined functions for breaking a problem into sub problems. [BT3]

1XXXXXX. CO5: To be able to solve different problems using pointers and arrays. [BT3]

1XXXXX. CO6: To be able to make use of structures for constructing a complex data type which is more meaningful and relevant? [BT3]

Digital Electronics

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Digital Electronics	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 1st

THEORY

Learning objectives: *To develop the concept and understanding of various number systems, realization of boolean algebra using logic gates, solve different types of combinational and sequential circuits, knowledge of ADC DAC and logic families*

Prerequisite: *High school Mathematics and knowledge of basic electrical elements*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Number Systems and Codes	6	
Module-II: Boolean Algebra	6	
Module-III: Logic Families	4	
Module-IV: Combinational Logic	6	
Module-V: Flip Flop	6	
Module-VI: Registers & Counters	8	

SYLLABUS OUTLINE:

Module-I: Number system and codes: Binary, octal, hexadecimal and decimal Number systems and their inter conversion, BCD numbers (8421-2421), gray code, excess–3 code, code conversion, ASCII, EBCDIC codes. Binary addition and subtraction, signed and unsigned binary numbers, 1's and 2's complement representation.

Module-II: Boolean Algebra : Basic logic circuits: Logic gates (AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR and their truth tables, Universal Gates, Laws of Boolean algebra, De-Morgan's theorem, Min term, Max term, POS, SOP, K-Map, Simplification by Boolean theorems, don’t care condition, Q-M method of function realization

Module-III: Logic Families: Introduction to digital logic family such as RTL, DTL, TTL, ECL, CMOS, IIR,HTL etc., their comparative study, Basic circuit, performance characteristics, Wired logic, opencollector output etc.

Module-IV: Combinational Logic: The Half adder, the full adder, subtractor circuit, comparator, Multiplxer de-multiplexer, decoder, BCD to seven segment decoder, Encoders.

Module-V: Flip flop and Timing circuit: set-reset laches, D-flipflop, R-S flip-flop, J-K Flip-flop, Masterslave Flip flop, edge triggered flip-flop, T flip-flop.

Module-VI: Registers & Counters: Synchronous/Asynchronous counter operation, Up/down synchronous counter, application of counter, Serial in/Serial out shift register, Serial in/Serial out shift register, Serial in/parallel out shift register, parallel in/ parallel out shift register, parallel in/Serial out shift register, Bi-directional register

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Digital Fundamentals by Morris and Mano, PHI Publication
2. Fundamental of digital circuits by A. ANANDKUMAR, PHI Publication.
3. Digital Fundamentals by FLOYD & JAIN, Pearsons Pub
4. Fundamentals of Logic Design by Charles H. Roth Thomson

CO-PO Mapping

	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	2	-	-	2	2		3
CO3	3	3	3	3	3	2	-	-	2	2		3
CO4	3	3	3	3	3	2	-	-	2	2		3
CO5	3	3	3	3	3	2	-	-	2	2		3
CO6	3	3	3	3	3	2	-	-	2	2		3
Avg	3	3	3	3	3	2	-	-	2	2	-	3

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

XXXX. CO1: Explaining the number systems and Boolean function simplification methods
XXXX. CO2: Design and simulation of combinational logic circuits
XXXX. CO3: Design and simulation of sequential logic circuits

XXXX. CO4: Construct combinational circuits using memory and PLDs
XXXX. CO5: Demonstrate the working principles of ADC and DACs
XXXX. CO6: Discuss about the logic families

PROBABILITY AND STATISTICS

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Probability and Statistics	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 1st

THEORY

Learning objectives:

- Learning basic statistical tools, types of qualitative and quantitative data, diagrammatic and graphical representation and organize, manage and present data.
- Acquire the knowledge about different measures of central tendency, dispersion, moments, skewness and kurtosis, bivariate data.

Prerequisite: *Before learning the concepts of Probability for Computer Science, you should have a basic knowledge of basic mathematics.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Statistical Methods	4	15%
Module-II: Univariate Data Analysis	8	20%
Module-III Bivariate Data Analysis	4	15%
Module-IV: Introductory Probability	8	15%
Module-V: Conditional Probability	6	15%
Module-VI: Random Variables and Generating Functions	6	20%

SYLLABUS OUTLINE:

Module-I: Introduction to Statistical Methods [4L]

Definition and scope of Statistics, concepts of statistical Population and Sample. Data: Quantitative and Qualitative, Discrete and Continuous, Cross-sectional and Time-series, Primary and Secondary. Scales of measurement: Nominal, Ordinal, Interval and Ratio.

Presentation of data: textual, tabular and graphical. Frequency distributions, cumulative frequency distributions and their graphical representations.

Module-II: Univariate Data Analysis [**6L**]

Measures of Central Tendency: Mean, Median, Mode. Measures of Dispersion: Range, Mean deviation, Standard deviation, Quartile deviation, Coefficient of variation. Moments, Skewness and Kurtosis. Sheppard's corrections for Moments. Box Plot and Outliers detection.

Module-III: Bivariate Data Analysis: [4L]

Definition, Scatter diagram, simple Correlation, simple linear Regression, principle of least squares, fitting of Polynomial and Exponential curves, Rank correlation: Spearman's (untied and tied cases).

Module-IV: Introductory Probability: [**6L**]

Introduction, Random Experiments, Sample Space, concept of three types of Sample Spaces – finite, countably infinite and uncountably infinite, Events and Algebra of Events, Definitions of Probability – Classical, Statistical and Axiomatic, applications.

.

Module-V: Conditional Probability: [**6L**]

Conditional Probability, laws of Addition and Multiplication, theorem of Total Probability, Bayes' theorem and its applications, Independent events

Module-VI: Random Variables and Generating Functions: [10L]

Definition, probability distribution of Random Variables, Cumulative Distribution Function (C.D.F.) and its properties (with proof), Discrete and Continuous Random Variables, Probability Mass Function (P.M.F.) and Probability Density Function (P.D.F.), Expectation and Moments, Dispersion, Skewness, Kurtosis, Quantiles.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

1. Gun A.M., Gupta M.K. and Dasgupta B. (2002): Fundamentals of Statistics, Vol. I, 8th Edition. World Press, Kolkata.
2. Gun A.M., Gupta M.K. & Dasgupta, B. (1994): An Outline of Statistical Theory, Vol. I, World Press.
3. Gun A.M. and Roy D. (2006): Problems In Probability Theory, 2nd Edition, World Press.

4. Ross S. (2002): A First Course in Probability, Prentice Hall.
5. Feller W. (1968): An Introduction to Probability Theory & its Applications, John Wiley.
6. Uspensky J.V. (1937): Introduction to Mathematical Probability, McGraw Hill.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	1	-	-	-	-	-	-	-	1
CO2	3	3	2	2	-	-	-	-	-	-	-	1
CO3	3	3	2	1	-	-	-	-	-	-	-	1
CO4	3	3	2	-	-	-	-	-	-	-	-	1
CO5	3	3	2	2	-	-	-	-	-	-	-	1
CO6	3	3	2	2	-	-	-	-	-	-	-	1
Avg	3	3	2	1.3	-	-	-	-	-	-	-	1.2

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

XXXXX ***CO1*: Build** knowledge about basic statistical methods and representations of data
XXXXX ***CO2:* Explain** the concept of frequency distributions and their graphical presentations.
XXXXX ***CO3:* Make use of** the knowledge about the measures of central tendency, measures of absolute and relative dispersion, moments, measures of skewness and kurtosis, measures of moments.
XXXXX ***CO4:* Apply** the concepts of scatter diagram, simple correlation, rank correlation, simple linear regression and curve fitting

XXXXX ***CO5:* Apply** the concepts of basic probability, concepts of conditional probability, Bayes' theorem and independent events, the fundamental knowledge of one dimensional discrete random variables and their related properties.

XXXXX ***CO6:* Build** the fundamental knowledge of one dimensional continuous random variables and their related properties.

Soft-Skill Development-I

Provided by Respective Department / School

Anyone (Sports/Yoga/NCC/NSS) EAA-I

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Communicative English-I

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Environmental Science-I

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Fundamentals of Computer Science & Problem-Solving Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Fundamentals of Computer Science & Problem-Solving Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 1st

Practical:

Learning objectives: *On completion of the course students will be able to enhance their analysing and problem solving skills and use the same for writing programs in C.*

Introduction to C Programming Language:

Basic structure of a C program
Writing and executing a simple C program

Data Types and Variables:

Basic data types (int, float, char, etc.)
Constants and variables
Declaration and initialization of variables

Operators and Expressions:

Arithmetic operators
Relational and logical operators
Increment and decrement operators
Assignment operators

Control Structures:

Conditional statements (if, if-else, nested if)
Switch statement

Loops (while, do-while, for), Nested loops.

Arrays and Strings:
Declaration and initialization of arrays
Accessing array elements, Multi-dimensional arrays
String handling functions

Functions:
Declaration and definition of functions
Function prototypes, Call by value and call by reference
Recursion

Pointers:
Introduction to pointers
Pointer arithmetic
Dynamic memory allocation (malloc, calloc, realloc, free)

Structures and Unions:
Defining structures and unions
Accessing structure members
Nested structures, Array of structures

File Handling:
File operations (opening, reading, writing, closing)
Sequential file processing
Random file processing

Pre-processor Directives:
#define, #include, #ifdef, #ifndef, etc.
Macros

Digital Electronics Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Digital Electronics Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 1st

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

SEMESTER-II

Sl No	Course Title	Code	Type	Credit	Type		
					L	T	P
1	Linear Algebra		MC	3	3	0	0
2	Programming and Data Structures		MC	4	4	0	0
3	Computer Organization		MC	3	3	0	0
4	Signals and Systems		MC	3	3	0	0
5	Soft-Skill Development-II		NV	1	1	0	0
6	MDC1:Selected by candidate from Other Discipline		MDC	4	4	0	0
7	Communicative English-II		AEC	2	2	0	0
8	Environmental Science-II		VAC	2	2	0	0
9	Programming and Data Structures Lab		MC	1	0	0	2
10	Computer Organization Lab		MC	1	0	0	2
11	Signals and Systems Lab		MC	1	0	0	2
Total Credit				**25 Credit**			

Linear Algebra

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Linear Algebra	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 2nd

THEORY

Learning objectives: *On completion of the course, student will be able to: apply the knowledge of matrix algebra, system of equations, vector space and linear transform as a tool in the field of Image Processing, Machine Learning and artificial intelligence etc.*

Prerequisite: *Before learning the concepts of Discrete Mathematics, you should have a basic knowledge of set, relation, mapping, matrix etc.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module I:Matrix, Determinant and	6L	12%
Module II: System of Equations	6L	13%
Module III: Vector Space	8L	16%

Module IV: Linear Transform	6L	21%
Module V: Inner Product Space	6L	26%
Module VI: Application	4L	8%

SYLLABUS OUTLINE:

Module I: Matrix, Determinant [6L]

Introduction to Matrices and Determinants, Inverse of a Matrix, Elementary operations, Echelon form, Row-reduced echelon form, Rank of a matrix. Symmetric and Skew-symmetric matrix, Orthogonal matrix, Hermitian and unitary matrices.

Module II: System of Equations [6L]

Solution of System of Linear Equations; Cramer's rule, Gaussian elimination; LU Decomposition; Solving Systems of Linear Equations using the tools of Matrices.

Module III: Vector Space [8L]

Definition of Vector space, Examples of vector space, Subspaces, linear dependence, Linear independence, Linear Span, Basis, Dimension.

Module IV: Linear Transform [6L]

Linear transformations, Examples of Linear Transform (Rotation, Projection etc.), Matrix representation of Linear transform, Linear Operator, Eigenvalues and Eigenvectors, Positive definite matrices.

Module V: Inner Product Space [6L]

Inner Product Space, Orthogonality, Projections, Gram-Schmidt orthogonalization theorem and QR decomposition. Singular value decomposition.

Module VI:Application:[4L]

Introduction to the applications of Linear Transform and inner product space in Image Processing and Machine Learning.

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Linear Algebra: Stephen H. Freiedberg, Arnold J. Insel and Lorence E. Spence
2. Higher Algebra- S.K. Mapa

Reference Books:

3. Linear Algebra - Ghosh and Chakraborty
4. Linear Algebra – Hadley

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	1
CO2	3	2	-	-	-	-	-	-	-	-	-	-
CO3	3	2	-	-	-	-	-	-	-	-	-	-
CO4	3	3	-	-	-	-	-	-	-	-	-	1
CO5	2	3	-	-	-	-	-	-	-	-	-	-
CO6	3	3	3	-	-	-	-	-	-	-	-	1
Avg	3	3	1	-	-	-	-	-	-	-	-	1

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: *(CO)*

After attending this course, the students will be able to

XXXXX.**CO1:** Understand the fundamentals matrix algebra.

XXXXX.**CO2:** Describe properties of linear systems using vectors and solve systems of linear equations and interpret their results.

XXXXX.**CO3:** Identify vector spaces and subspaces.

XXXXX.**CO4:** Identify Linear Transform.

XXXXX.**CO5:** Construct the matrix representation of a linear transform

XXXXX.**CO6:** Apply the knowledge of Eigenvalue, Eigenvector, Singular value decomposition and Principal component analysis to solve problems in Image Processing and Machine Learning.

Programming and Data Structures

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Programming and data Structures	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 2nd

THEORY

Learning objectives: *On completion of the course, student will be able to: Understand basic data structures and their implementation. Develop skills to apply appropriate data structures in problem solving.*

Prerequisite: *Fundamentals of Computer Science & Problem Solving*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Data Structure	2	7%
Module-II: Array	6	12%
Module-III: Linked List	10	20%
Module-IV: Stack and Queue	8	18%
Module-V:Trees	14	25%
Module-VI: Searching & Sorting	8	18%

SYLLABUS OUTLINE:

Module-I: Introduction to Data Structure [2L]

Introduction: Requirement of data structure. Concepts of data structures: a) Data and data structure b) Abstract Data Type and Data Type. Algorithms and programs, basic idea of pseudo-code.

Module-II: Array [6L]

Different representations – row major, column major. Sparse matrix - its implementation and usage. Array representation of polynomials.

Module-III: Linked List [10L]

Singly linked list, circular linked list, doubly linked list, linked list representation of polynomials and applications.

Module-IV: Stack and Queue [**8L**]

Stack and its implementations (using array, using linked list), applications: Polish notation.

Queue, circular queue, dequeue. Implementation of queue- both linear and circular (using array, using linked list), applications: Topological sort.

Recursion:

Principles of recursion – use of stack, differences between recursion and iteration, tail recursion. Applications - The Tower of Hanoi.

Module-V: Trees [**14L**]

Binary trees - definition, binary tree traversal (pre-, in-, post- order), binary tree representation (using array, using linked list), threaded binary tree (left, right, full) - non-recursive traversal algorithms using threaded binary tree, expression tree.

Binary search tree- operations (creation, insertion, deletion, searching). Height balanced binary tree – AVL tree (insertion, deletion with examples only), Red-Black Tree. B Trees – operations (insertion, deletion with examples only). B+ Trees.

Module-VI: Searching & Sorting [8L]

Sorting Algorithms: Bubble sort, insertion sort, selection sort, merge sort, quick sort, heap sort (concept of max heap, application – priority queue).

Searching: linear search, binary search.

Hashing: Hashing functions, collision resolution techniques.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Fundamentals of Data Structures in C, E.Horowitz- S.Sahni, Galgotia-2006

2. Data Structures and Algorithm Analysis in C, M.A.Weiss, Pearson Education-Fourth Edition

Reference Books:

1. Data Structures, Algorithms and Applications in C, Sartaj Sahni, University Press
2. Data Structures using C by Yedidyah Langsam, Moshe J. Augenstein and Aron M. Tananbaum, PHI.2002

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	2	1	-	-	-	-	-	-	-	-	-
CO2	3	3	3	1	-	-	-	-	-	-	-	1
CO3	3	3	3	2	-	-	-	-	-	-	-	1
CO4	3	3	3	1	-	-	-	-	-	-	-	1
CO5	3	3	3	2	-	-	-	-	-	-	-	1
CO6	3	3	3	2	-	-	-	-	-	-	-	1
Avg												

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

1XXXXX. CO1: To be able to classify linear and non-linear data structure.

1XXXXX. CO2: To be able to solve different problems using Arrays.

1XXXXXX. CO3: To be able to make use of linked list for various operations on polynomials, sparse matrix etc.

1XXXXXX. CO4: To be able to utilize the knowledge of Stack, Queues in solving real life problem.

1XXXXXX. CO5: To be able to apply the knowledge of several binary trees in problem solving.

1XXXXX. CO6: To be able to identify of the most appropriate searching or sorting algorithm for enhancing the efficiency (i.e. reduce the run-time) or for better memory utilization.

Computer Organization

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Organization	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:**
CODE:	**SEMESTER:** 2nd

THEORY

Learning objectives: *On completion of the course, student will be able to: Demonstrate computer organization concepts related to design of modern processors, memories and I/Os. Analyse the performance of commercially available computers. This course is intended to teach the basics involved in data representation and digital logic circuits used in the computer system.*

Prerequisite: *Before learning the concepts of Computer Organization, you should have a basic knowledge prior to Computer System Architecture, basic functional units of a computer system, Binary numbers etc.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Fundamental of Computer Organization	5	14
Module-II: ALU Design	7	20
Module-III: Computer Arithmetic	7	20
Module-IV: Design of Control Unit	6	17
Module-V: Memory	6	15
Module-VI: Input-Output Organization	5	14

SYLLABUS OUTLINE:

Module-I: Fundamental of Computer Organization [5L]

Basic organization of the stored program computer and operation sequence for execution of a program. Role of operating systems and compiler/assembler.Fetch, decode and execute cycle, Concept of operator, operand, registers and storage, Instruction format. Instruction sets and addressing modes.

Module-II: ALU Design [7L]

The ALU: ALU organization, Integer Representation , Serial and parallel Adders, 1’S and 2’s Complement Arithmetic, Multiplication of Signed binary numbers, Overflow detection , Status flags. Floating point - IEEE 754 standard. Fixed and floating point representation of numbers. Floating point number arithmetic, Design of ALU.

Module-III: Computer Arithmetic [7L]

Overflow and underflow. Design of adders - ripple carry and carry look-ahead principles. Fixed point multiplication -Booth's algorithm. Fixed point division - Restoring and non-restoring algorithms.

Module-IV: Design of Control Unit [6L]

Hardwired and micro- programmed design approaches , Case study - design of simple hypothetical CPU.

Module-V: Memory [6L]

Memory unit design with special emphasis on implementation of CPU-memory interfacing. Memory organization, static and dynamic memory, memory hierarchy, associative memory. Cache memory, Virtual memory. Data path design for read/write access.

Module-VI: Input-Output Organization [5L]

Input-Output Subsystems, I/O transfers - program controlled , interrupt driven and DMA, privileged and non- privileged instructions , software interrupts and exceptions, Programs and processes - role of interrupts in process state transitions.

Pedagogy for Course Delivery: *Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA):*NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Computer Organization and Design: The Hardware/Software Interface, David A. Patterson and John L. Hennessy, Elsevier.

2. Computer Organization, Carl Hamachar, Zvonco Vranesic and Safwat Zaky, McGraw Hill.

3. Computer Architecture and Organization, John P. Hayes, McGraw Hill.

Reference Books:

4. Computer Organization and Architecture: Designing for Performance, William Stallings, Pearson Education.

5. Computer Systems Design and Architecture, Vincent P. Heuring and Harry F. Jordan, Pearson Education.

CO-PO Mapping

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

CO4	2	3	3	-	3	-	-	-	-	-	-	2
CO5	1	1	3	3	1	-	-	-	-	-	-	1
CO6	2	1	1	2	-	-	-	-	-	-	-	1
Avg	2	2.16	2	1.8	2	-	-	-	-	-	-	1.2

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXX. CO1: Understand the structure, function and characteristics of computer systems and understand the design of the various functional units and components of computers.

1XXXXX. CO2: Design the arithmetic and Logic unit and understand the floating and fixed point number representation

1XXXXXX. CO3: Analyze the performance of ripple carry adder and carry look ahead adder and understand the multiplication and division algorithm

1XXXXXX. CO4: Identify the elements of control unit and design of control unit

1XXXXXX. CO5: Explain the function of each element of a memory hierarchy.

1XXXXX. CO6: Understand the input output subsystem and analyze the role of interrupts in process state transition.

Signal and Systems

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Signal and Systems	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 2nd

THEORY

Prerequisite: *Higher Secondary Mathematics: indices, exponentials, logarithms, basic calculus*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I:	5	
Module-II:	5	
Module-III:	8	
Module-IV:	6	
Module-V:	6	
Module-VI:	6	

SYLLABUS OUTLINE:

Module-I:

Energy and power signals, continuous and discrete time signals, continuous and discrete amplitude signals. System properties: linearity: additively and homogeneity, shift-invariance, causality, stability, reliability.

Module-II:

Linear shift-invariant (LSI) systems, impulse response and step response, convolution, input output behaviour with aperiodic convergent inputs. Characterization of causality and stability of linear shift-invariant systems. System representation through differential equations and difference equations.

Module-III:

Periodic and semi-periodic inputs to an LSI system, the notion of a frequency response and its relation to the impulse response, Fourier series representation, the Fourier Transform, convolution/multiplication and their effect in the frequency domain, magnitude and phase response, Fourier domain duality. The Discrete-Time Fourier Transform (DTFT) and the Discrete Fourier Transform (DFT). Parseval's Theorem. The idea of signal space and orthogonal bases.

Module-IV:

The Laplace Transform, notion of Eigen functions of LSI systems, a basis of eigen functions, region of convergence, poles and zeros of system, Laplace domain analysis, solution to differential equations and system behaviour.

Module-V:

The z-Transform for discrete time signals and systems- Eigen functions, region of convergence, z-domain analysis.

Module-VI:

State-space analysis and multi-input, multi-output representation. The state-transition matrix and its role. The Sampling Theorem and its implications- Spectra of sampled signals. Reconstruction: ideal interpolator, zero-order hold, first-order hold, and so on. Aliasing and its effects. Relation between continuous and discrete time systems.

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. A.V. Oppenheim, A.S. Willsky and I.T. Young, "Signals and Systems", Prentice Hall, 1983.

2. R.F. Ziemer, W.H. Tranter and D.R. Fannin, "Signals and Systems - Continuous and Discrete", 4th edition, Prentice Hall, 1998.

3. Papoulis, "Circuits and Systems: A Modern Approach", HRW, 1980.

4. B.P. Lathi, "Signal Processing and Linear Systems", Oxford University Press, c1998.

5. Douglas K. Lindner, "Introduction to Signals and Systems", McGraw Hill International Edition: c1999.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	2	2	2	-	-	-	-	-	-	-	3
CO2	3	-	-	3	2	-	-	-	-	-	-	3
CO3	2	3	3	3	2	-	-	-	-	-	-	2
CO4	-	2	1	2	-	-	-	-	-	-	-	-
CO5	2	2	2	2	1	-	-	-	-	-	-	3
CO6	3	-	-	3	2	-	-	-	-	-	-	3
Avg	3	3	3	3	2	-	-	-	-	-	-	3

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

CO1: Describe the basic mathematical operations on signals and systems

CO2: Convert the Analog signal into discrete time signal using sampling theorem

CO3: Explain the properties of Fourier series and transformations

CO4: Discuss the properties of Laplace and Z transformation

CO5: Develop input output relationship for linear shift invariant system and understand the convolution operator for continuous and discrete time system.

CO6: Compute the response of the LTI system for random inputs

Soft-Skill Development-II
PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Selected by candidate from Other Discipline
PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Communicative English-II
PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Environmental Science-II
PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Programming and Data Structures Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Programming and Data Structure Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 2nd

PRACTICAL (**total contact hours,** hr/week), if applicable

Learning objectives: *On completion of the course, student will be able to: To develop programming skills with a systematic approach in organizing a program in C language with an understanding of basic data structures. Develop skills to apply appropriate data structures in problem solving in the context of specific engineering problems*

List of practical

1. Implement a stack by using array then do the PUSH & POP operation
2. Write a program to evaluate a postfix notation.
3. Write a program to convert infix to postfix.
4. Implement a Circular Queue by using array then do the enqueue and dqueue operation.
5. Implement Single Linked List and does insertion, deletion, display, reverse.
6. Implement Doubly Linked List and does insertion, deletion, display, reverse.
7. Implement a stack using linked lists.
8. Implement Circular Linked List, queue using linked lists.
9. Implement JOSEPHUS problem.
10. Write a program to add two polynomials.
11. Write a program to multiply two polynomials.
12. Write a program for addition of sparse matrix.
13. Write a program to multiplication of sparse matrix.
14. Create binary search tree and implement Preorder, Inorder, Postorder and delete an element from the tree
15. Implement a threaded binary tree and perform the inorder traversal operation.
16. Implement AVL tree.
17. Implement Splay tree.
18. Implement Priority Queue using Heap.
19. Implement BFS, DFS.
20. Implement Prim's and Kruskal's Algorithm.
21. Write a program to sort an array using Bubble sort.
22. Write a program to sort an array using Insertion sort
23. Write a program to sort an array using Selection sort.
24. Write a program to sort an array using Quick sort.
25. Write a program to sort an array using Merge sort.
26. Write a program to sort an array using Heap sort.
27. Write a program to sort an array using Radix sort.
28. Write a program to sort an array using Shell sort.
29. Implement Linear and Binary search.
30. Implement interpolation search.

COMPUTER ORGANIZATION LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Organization Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 2nd

PRACTICAL (**total contact hours,** hr/week), if applicable

Learning objectives:

The students will learn the processor design, interfacing with CPU, DAC, ADC, keyboard-display modules, etc.,

List of practical

Exp. No.	***Experiment Name***	***CO Mapping***
1.	To design the circuit of half adder.	**CO1**
2.	To design the circuit of full adder.	**CO1**
3.	To design the circuit of half subtractor.	**CO2**
4.	To design the circuit of full subtractor.	**CO2**
5.	To design an 8×1 Multiplexer.	**CO3**
6.	To design a 4 bit combinational shifter.	**CO4**
7.	To design a BCD adder	**CO5**
8.	To design a 4-bit adder subtractor.	**CO5**
9.	To design 2:4 Decoder	**CO5**
10.	To design an ALU.	**CO6**

***Course learning outcome**:*

XXXXXX. CO1: *To implement adder circuits using basic gates*

XXXXXX. CO2: *To understand the converter circuits using basic gates.*

XXXXXX. CO3: *To understand the working of Multiplexer by using IC 74153*

XXXXXX. CO4: *To understand combinational Shift Circuit.*

XXXXXX. CO5: *To understand Adder and Decoder Circuit.*

XXXXXX. CO6: *To understand the various circuits for ALU, data path and control units*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

SIGNALS AND SYSTEMS LAB
PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

SECOND YEAR

SEMESTER-III

Sl No	Course Title	Code	Type	Credit	Type		
					L	T	P
1	Algorithm-I		MC	3	3	0	0
2	Computer Architecture		MC	3	3	0	0
3	Formal Language and Automata Theory		MC	4	4	0	0
4	Object Oriented Programming		MC	1	1	0	0
5	Anyone (Sports/Yoga/NCC/NSS) EAA-II		NV	1	0	0	2
6	Soft-Skill Development-III		NV	1	1	0	0
7	MDC2:Selected by candidate from Other Discipline		MDC	3	3	0	0
8	SEC1:Entrepreneurship Skill Development		SEC	3	3	0	0
9	Foreign language-I		AEC	2	2	0	0
10	Algorithm-I Lab		MC	1	0	0	2
11	Computer Architecture Lab		MC	1	0	0	2
12	Object Oriented Programming Lab		MC	2	0	0	4
Total Credit					**25 Credit**		

Algorithm-I

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Algorithm-I	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 3rd

THEORY

Learning objectives: *To design the algorithms for solving different types of problems in Computer Science. It also helps to design and analyse the logic on how the program will work before developing the actual code for a program.*

Prerequisite: *Discrete Math, Programming and Data Structure.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Analysis of Algorithm	4	6%
Module-II: Foundations of Design and Analysis	8	25%
Module-III: Sorting	6	22%
Module-IV: Graph	10	25%
Module-V: Optimization Technique	4	12%
Module-VI: Selected Topics	4	10%

SYLLABUS OUTLINE:

Module-I: Introduction to Analysis of Algorithm [4L]

Characterizing features of an algorithm, Performance analysis, Time and Space Complexities – Worst case and Average case, Asymptotic Notations - Big O, Small O, Big Omega, Small Omega and Theta notations.

Module-II: Design and Analysis Technique [**8L**]

Introduction to different algorithmic paradigms with one example for each: Divide and Conquer - Binary Search, Greedy – Job Sequencing Problem, Dynamic Programming - Matrix Chain Multiplication, Backtracking- Eight Queen's Problem.

Module-III: Sorting [6L]

Lower Bound on the time complexity, Quicksort (including analysis of worst-case and average case complexities), Merge Sort and its complexity analysis, Counting sort, Radix sort, Bucket sort.

Module-IV: Graph [10L]

Graph representations/storage implementations – adjacency matrix, adjacency list, Graph traversal and connectivity, Depth-first search (DFS), Breadth-first search (BFS), Disjoint Set Manipulation: UNION - FIND Algorithms. MST- Prim's algorithm, Kruskal's Algorithm, Single-Source Shortest Paths - Bellman-Ford algorithm, Dijkstra's algorithm; All-Pairs Shortest Paths – Shortest paths and matrix multiplication, Floyd-Warshall algorithm.

Module-V: Optimization Technique [4L]

Huffman coding, Knapsack Problem, Bin-Packing Problem.

Module-VI: Selected Topics [4L]

Dynamic Programming: Binomial Coefficient, Longest common subsequence. Branch & Bound: 15-Puzzle Problem.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. *Introduction to Algorithms*, T. H. Cormen, C. E. Leiserson and R. L. Rivest.
2. *The Design and Analysis of Computer Algorithms*, A. Aho, J. Hopcroft and J. Ullman.

Reference Books:

1. *Fundamental of Computer Algorithms*, E. Horowitz and S. Sahni.
2. *The Art of Computer Programming, Vol. 1, Vol. 2 and Vol. 3*, .D. E. Knuth.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	3										1
CO2	3	3	3									1
CO3	3	3	3	1								1
CO4	3	3	3									1
CO5	3	3	3									1
CO6	3	3	3									1
Avg	3	3	3	1								1

Highly Correlated: **3**

Moderately Correlated**: 2**

Slightly Correlated: **1**

Course learning outcome: (CO)

CO1: To be able to **utilize** various asymptotic notations to compute the complexity of different algorithms.

CO2: To be able to **choose** the suitable standard algorithm design techniques such as divide & conquer, greedy, dynamic programming, backtracking in solving problems.

CO3: To be able to **compare** the complexity of various sorting algorithm.

CO4: To be able to **make use of** various graph algorithms for solving problems, i.e. finding shortest path, minimum spanning tree etc.

CO5: To be able to **select** the appropriate algorithm strategy for several optimization problems.

CO6: To be able to **utilize** various algorithm strategies like Branch & Bound, LCS for solving real life problems.

Computer Architecture

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Architecture	**COURSE CREDIT :** 03 [3-0-0]

DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 3rd

THEORY

Learning objectives: *Students will be able to conceptualize the basics of organizational and architectural issues of a digital computer, Classify and compute the performance of machines, Machine Instructions. They will be able to learn about various data transfer techniques in digital computer and the I/O interfaces. The students will be able to estimate the performance of various classes of Memories, build large memories using small memories for better performance and Relate to arithmetic for ALU implementation. They will be able to understand the basics of hardwired and micro-programmed control of the CPU, pipelined architectures, Hazards and Superscalar Operations.*

Prerequisite: *Computer Organization, Digital Logic, Machine Instructions, Dataflow.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	3	
Module-II: Pipelining	8	
Module-III: Memory Organization	6	
Module-IV: Instruction-Level Parallelism	7	
Module-V: Multiprocessor Architecture	8	
Module-VI: Non Von Neumann Architecture	4	

SYLLABUS OUTLINE:

Module-I: Introduction [3L]

Review of basic computer architecture (Revisited), Quantitative techniques in computer design, measuring and reporting performance.

Module-II: Pipelining [8L]

Pipelining: Basic concepts, instruction and arithmetic pipeline, data hazards, control hazards and structural hazards, techniques for handling hazards. Exception handling. Pipeline optimization techniques; Compiler techniques for improving performance.

Module-III: Memory Organization [6L]

Revisiting Hierarchical memory technology: Inclusion, Coherence and locality properties; Cache coherence problem; Virtual memory organization, mapping and management techniques, memory replacement policies, interleaved memory organization, C access, S access, CS access

Module-IV: Instruction-Level Parallelism [7L]

Instruction-level parallelism: basic concepts, techniques for increasing ILP, superscalar, super-pipelined and VLIW processor architectures. Array and vector processors.

Module-V: Multiprocessor Architecture [8L]

Multiprocessor architecture: taxonomy of parallel architectures; Centralized shared- memory architecture: synchronization, memory consistency, interconnection networks. Distributed shared-memory architecture. Cluster computers.

Module-VI: Non Von Neumann Architecture **[4L]**

Non von Neumann architectures: data flow computers, reduction computer architectures, systolic architectures.

Pedagogy for Course Delivery: *Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA):*NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. J. L. Hennessy and D. A. Patterson, “Computer Architecture A Quantitative Approach”, Morgan Kauffman, 2011.
2. V. Carl, G. Zvonko and S. G. Zaky, “Computer organization”, McGraw Hill, 1978.
3. Computer Organization and Architecture: Designing for Performance, William Stallings, Pearson Education.
4. Computer Systems Design and Architecture, Vincent P. Heuring and Harry F. Jordan, Pearson Education.
5. B. Brey and C. R. Sarma, “The Intel microprocessors”, Pearson Education, 2000.

Reference Books:

1. Rajaraman – “Computer Organization & Architecture”, PHI
2. B.Ram – “Computer Organization & Architecture”, Newage Publications

<u>CO-PO Mapping</u>

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

CO4	3	1	2	1	-	-	-	-	-	-	-	1
CO5	2	-	-	2	-	-	-	-	-	-	-	-
CO6	2	-	2	2	-	1	-	-	-	-	-	1

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

1XXXXX. CO1: Understand the concepts of pipelining and parallel processing.

1XXXXX. CO2: Design arithmetic and instruction pipeline and be able to solve the problems of pipeline hazards.

1XXXXX. CO3: Understand the interleaved memory organization and concurrent and simultaneous memory access and analysis the cache coherence problem.

1XXXXX. CO4: Understand the techniques for designing superscalar and super-pipelined architecture.

1XXXXX. CO5: Understand the concepts of multiprocessor architectures.

1XXXXX. CO6: Understand the concepts of non-von Neumann architectures like dataflow computer, systolic architecture etc.

Formal Language and Automata Theory

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Formal Language and Automata Theory	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 3rd

THEORY

Learning objectives: *This course focuses on the basic theory of Computer Science and formal methods of computation like automata theory, formal languages, grammars and Turing Machines. The objective of this course is to explore the theoretical foundations of computer science from the perspective of formal languages and classify machines by their power to recognize languages.*

Prerequisite: *The primary prerequisite for this course is reasonable "mathematical sophistication." The basic mathematical notations are required to know. The logical functional principles of machine are also need to know. Sets & Types, Sequences, Tuples,* *Propositional and Predicate Logic, Mathematical Induction, Recursive Definitions, Big-O Notation, Relations and Functions*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Finite State Machines and Models	10	20
Module-II: Finite Automation	10	20
Module-III: Closure Properties of Regular Sets	4	15
Module-IV: Context Free Grammars	4	15
Module-V: Pushdown Automata	4	15
Module-VI: Turing machine and Linear Bounded Automata	4	15

SYLLABUS OUTLINE:

Module-I: Finite State Machines and Models [10L]

Introduction, definition, concept of sequential circuits, state table & state assignments, concept of synchronous, asynchronous and linear sequential machines.

Basic definition, mathematical representation, Moore versus Mealy m/c, capability & limitations of FSM, state equivalence & minimization, machine equivalence, incompletely specified machines, merger graph & compatibility graph, merger table, Finite memory, definite, information loss less & inverse machines: testing table & testing graph.

Module-II: Finite Automation [10L]

Finite Automata: Deterministic Finite Automata, Non-Deterministic Finite Automata, Finite Automata with Outputs(without conversions). Regular Expressions and Languages: Regular Expressions, Finite Automata and Regular Expressions, Algebraic Laws for Regular expressions (without proofs). Properties of regular Languages: Proving Languages not to be regular.

Module-III: Closure Properties of Regular Sets [4L]

Pumping lemma & its application, closure properties minimization of finite automata: minimization by distinguishable pair, Myhill-Nerode theorem.

Module-IV: Context Free Grammars [4L]

Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Tress, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages, Chomsky Normal Form, Griebech Normal Form.

Module-V: Pushdown Automata [4L]

Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state, Acceptance by empty stack, Deterministic Pushdown Automata. From CFG to PDA, From PDA to CFG.

Module-VI: Turing machine and Linear Bounded Automata **[4L]**

Introduction and basic concepts, Representation of Turing Machine, Design of Turing Machine, Linear bounded automata, and languages, Type 0 Grammars

Pedagogy for Course Delivery: *Hybrid Mode (Offline Class / Presentation / Video / MOODLE / NPTEL)*

List of Professional Skill Development Activities (PSDA):*NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Hopcroft and Ullman, “Introduction to Automata Theory, Languages and Computation”, 2nd edition, Pearson/Prentice Hall India, 2007.

Reference Books:

1. K.L.P. Mishra and N.Chandrasekaran, “Theory of Computer Science: Automata, Languages and Computation”, 2nd edition, Pearson/Prentice Hall India, 2004.
2. Martin J. C., “Introduction to Languages and Theory of Computations”, 2nd edition, Tata McGraw Hill, 2005.
3. Papadimitrou, C. and Lewis, C.L., “Elements of the Theory of Computation”, 2nd edition, Pearson/Prentice Hall India, 2009.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	2	2	-	-	-	-	-	-	-	1
CO2	3	-	-	1	-	-	-	-	-	-	-	1
CO3	3	2	1	-	-	1	-	-	-	-	-	-
CO4	-	2	2	1	-	-	-	-	-	-	-	1
CO5	2	-	-	1	2	1	-	-	-	-	-	1
CO6	2	2	2	-	-	1	-	-	-	-	-	-
Avg	2.16	1	1.16	0.83	0.33	0.5						0.66

Highly Correlated: **3**
Moderately Correlated**: 2**

Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXX. CO1: To be able to Understand the fundamental concepts of Finite State Machines and Models

1XXXXX. CO2: To be able to Understand the fundamental concepts of Formal Languages and Automata.

1XXXXX. CO3: To be able to apply the pumping lemma, closure properties to problems.

1XXXXX. CO4: To be able to Understand the fundamental concepts of Context free grammars.

1XXXXX. CO5: To be able to Understand the fundamental concepts of Pushdown Automata.

1XXXXX. CO6: To be able to Understand the fundamental concepts of Turing machine and Linear Bounded Automata.

OBJECT ORIENTED PROGRAMMING

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Object Oriented Programming through C++	**COURSE CREDIT :** 01 [1-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 3rd

THEORY

Learning objectives: *On completion of the course, students will be able to understand the basic object-oriented programming concepts and apply them in problem-solving, illustrate inheritance concepts for reusing the program, and demonstrate the concepts of classes and objects with reality.*

Prerequisite: *Fundamentals of Computer Science and Problem Solving*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: The Fundamentals of Object-Oriented Programming	2	6
Module-II: Difference between procedural and object-oriented programming	6	18
Module-III Class & Object Properties	6	18
Module-IV: Essentials of Object-Oriented Programming	8	23
Module-V: Inheritance	6	17

Module-VI: **More on C++**	6	18

SYLLABUS OUTLINE:

Module 1: The Fundamentals of Object-Oriented Programming [2L]

Necessity for OOP, Data Hiding, Data Abstraction, Encapsulation, Procedural Abstraction, Polymorphism, Inheritance

Module 2: Difference between procedural and object-oriented programming [6L]

Some differences between C and C++: Single line comments, Local variable declaration within function scope, function declaration, function overloading, stronger type checking, Reference variable, parameter passing – value vs reference, passing a pointer by value or reference, #define constant vs const, Operator new and delete, the type casting operator, Inline Functions in contrast to macro, default arguments

Module 3: Class & Object Properties [6L]

More extensions to C in C++ to provide OOP Facilities: Class and Object, Scope of Class and Scope Resolution Operator, Member Function of a Class, private, protected, and public Access Specifier, this Keyword, Constructors and Destructors, error handling (exception)

Module 4: Essentials of Object-Oriented Programming [6L]

Operator overloading, function Overloading, friend function, friend class.

Module 5: Inheritance [4L]

Single and Multiple, Class Hierarchy, Pointers to Objects, Assignment of an Object to another Object, Polymorphism through dynamic binding, Virtual Functions, Overloading, overriding and hiding, and Virtual base class.

Module 6: More on C++ [4L]

Error Handling, Generic Programming: Template concept, class template, function template, template specialization, Input and Output: Streams, Files, Library functions, formatted output

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. The Complete Reference C++, 4th Edition, Herbert Schildt, Tata McGraw Hill.

2. Problem solving with C++: The Object of Programming, 4th Edition, Walter Savitch, Pearson Education

Reference Books:

1. The C++ Programming Language, 3rd Edition, B. Stroutstrup, Pearson Education.

2. OOP in C++, 3rd Edition, T. Gaddis, J. Walters and G. Muganda, Wiley Dream Tech Press.

3. Object Oriented Programming in C++, 3rd Edition, R. Lafore, Galigotia Publications Pvt Ltd

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	3	-	-	-	-	-	-	-	-	-
CO2	2	1	3	-	-	-	-	-	-	-	-	-
CO3	3	-	1	2	-	-	-	-	-	-	-	-
CO4	3	2	-	1	1	-	-	-	-	-	-	2
CO5	3	2	2	2	-	-	-	-	-	-	-	2
CO6	3	-	3	3	2	-	-	-	-	-	-	3
Avg	2.8	2	2.2	2	1.5							2.3

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course Outcome (CO):

XXXXX. CO 1: To be able to **describe** the procedural and object-oriented paradigm with concepts of streams, classes, functions, data, and objects.
XXXXX. CO 2: To be able to **apply** dynamic memory management techniques using pointers, constructors, destructors, etc
XXXXX. CO 3: To be able to **apply** the concept of classes and objects with an idea of scope resolution operator and various access specifies.
XXXXX. CO 4: To be able to **describe** the concept of function overloading, operator overloading, virtual functions, and polymorphism.
XXXXX. CO 5: To be able to **apply** inheritance with an insight into an early and late binding, usage of exception handling, generic programming
XXXXX. CO 6: To be able to **apply** the knowledge C++ template in designing generic classes

Anyone (Sports/Yoga/NCC/NSS) EAA-II

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

SOFT-SKILL DEVELOPMENT-III

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

SELECTED BY CANDIDATE FROM OTHER DISCIPLINE

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

ENTREPRENEURSHIP SKILL DEVELOPMENT

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

FOREIGN LANGUAGE-I

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

ALGORITHM-I LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Algorithm-I Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 3rd

LIST OF ASSIGNMENTS:

1. Write a program to find the minimum and maximum elements from an array.
2. Write a program to perform a binary search algorithm using recursion.
3. Write a program to find minimum and maximum elements from an array using Divide and Conquer approach.
4. Write a program to display the Fibonacci series till n numbers using recursion.
5. Write a program to perform a bubble sort algorithm using a functional approach and print the time complexity.
6. Write a program to perform a selection sort algorithm using a functional approach and print the time complexity.

7. Write a program to calculate the shortest path using **prims algorithm**.
8. Write a program to calculate the shortest path using the **Kruskal algorithm**.
9. Write a program to implement the **DFS** algorithm.
10. Write a program to implement the **BFS** algorithm.
11. Write a program to implement **Matrix Chain Multiplication** using DP.
12. Write a program to perform the **Fractional knapsack** (greedy approach) algorithm using a functional approach
13. Write a program to perform **0 - 1 knapsack** (DP approach) algorithm using a functional approach
14. Write a program to calculate the shortest path using the **Dijkstra algorithm** (greedy approach).
15. Write a program to calculate the shortest path using **Bellman Ford algorithm** (DP approach).
16. Write a program to perform **Job-sequence-with deadline**.
17. Write a program to implement the N-Queen problem using backtracking.
18. Write a program to implement Traveling Salesman Problem.

COMPUTER ARCHITECTURE LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Architecture Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 3rd

Tentative List of Experiments:

- Simulation of simple fundamental units like half adder, full adder, multiplexer, de-multiplexer, Arithmetic logic Unit, Simple processor (CPU) etc using VHDL code. (Using Xilinx)

OBJECT ORIENTED PROGRAMMING LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Object Oriented Programming Lab Lab	**COURSE CREDIT :** 02 [0-0-4]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 3rd

List of practicals

Exp. No.	***Experiment Name***
1.	Write a C++ program to determine whether a number is a palindrome or not.
2.	Write a C++ program to design a class polar which describes a point in the plane using polar coordinates radius and angle. Use the overloaded + operator to add two objects of polar.
3.	Write a C++ program create a class FLOAT that contains one float data member. Overload all four arithmetic operators so that they operate on the objects of FLOAT.
4.	Write a C++ program to create a Class MAT of size M*N. Define all possible matrix operations for MAT-type objects.
5.	Write a C++ program having a class to represent a vector (a series of float values). Include member functions to perform the following tasks: a) To create a vector b) To modify the value of a given element c) To multiply by a scalar value d) To display the vector in the form (10, 20, 30, …) Write a C++ program to test your class.
6.	Write a C++ program considering two classes DM and DB which store the value of distances. DM stores distance in meters and centimetres, and DB in feet and Inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out addition operations.

7.	Write a C++ program having a string that could work as a user-defined string type. Include constructors that will enable us to create an uninitialized string:
	String s1; // string with length 0
	and also initialize an object with a string constant at the time of creation like
	String s2 (“Well done!”);
	Include a function that adds two strings to make a third string. Note that the statement
	s2 = s1;
	Will be a perfectly reasonable expression to copy one string to another.
	Write a complete program to test your class to see that it does the following tasks:
	(a) Creates uninitialized string objects.
	(b) Creates objects with string constants.
	(c) Concatenates two strings properly.
	(d) Displays the desired string object.

8. Create a base class Shape. Use this Class to store two double-type values that could be used to compute areas. Add two derived Class Triangle and Rectangle from the base class Shape. Add to the base class, a member function get_data () to initialize the data members in the base class and add another member function display_area () to compute the area. Declare this member function as virtual. Write a C++ program to implement the class that accepts dimensions and calculate area. (RUN TIME POLYMORPHISM)

9. Write a simple C++ program for accessing files.

10. Write a simple C++ program to sort a set of data values using templates. It may be integer data or float data or character data.

SECOND YEAR

SEMESTER-IV

Sl No	Course Title	Code		Credit	Type		
					L	T	P
1	Operating Systems		MC	4	4	0	0
2	Database Management System		MC	4	4	0	0
3	Artificial Intelligence		MC	4	4	0	0
4	Algorithm-II / Compiler Design / Optimization Techniques / Computer Graphics		ME	3	3	0	0
5	Soft-Skill Development-IV		NV	1	1	0	0
6	MDC3:Selected by candidate from Other Discipline		MDC	2	2	0	0
7	Foreign language-II		AEC	2	2	0	0
8	Human Values and Ethics		VAC	2	2	0	0
9	Operating Systems Lab		MC	1	0	0	2
10	Database Management System Lab		MC	1	0	0	2
11	Artificial Intelligence Lab		MC	1	0	0	2
Total Credit				**25 Credit**			

OPERATING SYSTEMS

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Operating Systems	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 4th

THEORY

Learning objectives: *This course OPERATING SYSTEMS is an essential part of any Computer-Science education. The purpose of this course is to understand the mechanisms of the Operating Systems like Process Management, Process Synchronization, Memory Management, File System Implementation, Storage Structures used in OS and Protection Principles. How effectively the OS is utilizing the CPU resources with the help of these mechanisms.*

Prerequisite: *Good knowledge of C, Computer Organization and Architecture, x86 Assembly level programming.*

Course content/Syllabus:

Module no.	No of	Weightage (%)

	lecture/Contact hour	
Module-I: Introduction	10	21
Module-II: Process Management	10	21
Module-III: Process Synchronization and Deadlocks	10	21
Module-IV: Memory management and Virtual Memory	10	21
Module-V: File and I/O Systems Management	4	8
Module-VI: Disk Management	4	8

SYLLABUS OUTLINE:

Module-I: Introduction: [10L]

Introduction to OS, operating system functions, evaluation of OS, Different types of OS: batch, multi-programmed, time-sharing, real-time, distributed, parallel.

System Structure, Computer system operation, I/O structure, storage structure, storage hierarchy, different types of protections, operating system structure (simple, layered, virtual machine), O/S services, system calls, Process Management.

Module-II: Process Management: [**10L**]

Concept of processes, process scheduling, operations on processes, co-operating processes, interposes communication.

CPU scheduling: scheduling criteria, preemptive & non-preemptive scheduling, scheduling algorithms (FCFS, SJF, RR, and priority), and algorithm evaluation, multi-processor scheduling.

Module-III: Process Synchronization and Deadlocks: [10L]

Background, critical section problem, critical region, synchronization hardware, classical problems of synchronization, semaphores.

System model, deadlock characterization, methods for handling deadlocks, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock, Storage Management. Threads overview, benefits of threads, user and kernel threads.

Module-IV: Memory management and Virtual Memory: [**10L**]

Background, logical vs. physical address space, swapping, contiguous memory allocation, paging, segmentation, segmentation with paging. Virtual Memory background, demand paging, performance, page replacement, page replacement algorithms (FCFS, LRU), allocation of frames, thrashing.

Module-V: File and I/O Systems Management: [**4L**]

File concept, access methods, directory structure, file system structure, allocation methods (contiguous, linked, and indexed), and free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency & performance. I/O hardware, polling, interrupts, DMA, application I/O interface (block and character devices, network devices, clocks and timers, blocking and nonblocking I/O), kernel I/O subsystem (scheduling, buffering, caching, spooling and device reservation, error handling), performance.

Module-VI: Disk Management: [4L]

"Disk structure, disk scheduling (FCFS, SSTF, SCAN,C-SCAN) , disk reliability, disk formatting, boot block, bad blocks.

Protection & Security Goals of protection, domain of protection, security problem, authentication, one-time password, program threats, system threats, threat monitoring, encryption".

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Operating System Principles, Abraham Silberchatz, Peter B.Galvin,Greg Gagne,8th Edition, Wiley
Student Edition
2. Operating System-Internals and Design Principles. W. Stallings, 6th Edition, Pearson.

Reference Books:

1. Modern Operating System, Andre w s Tanenbaum, 3rd Edition, PHI
2. Operating System A concept-based Approach, 2nd Edition, D.M.Dhamdhere, TMH.
3. Principle Of Operating Systems, B.LStuart, Cengage Learning, India Edition
4. Operating system, A.s.Godbole, 2nd Edition, TMH.
5. An Introduction to Operating System, P.C.P.bhatt, PHI.

CO-PO Mapping

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

CO.5	3	2	1	2	-	-	-	-	-	-	-	1
CO.6	3	2	1	2	-	-	-	-	-	-	-	2

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXX. CO1: To be able to understand the design of an operating system and its types. I/O structures and storage structures.

1XXXXX. CO2: To be able to apply process scheduling algorithm in various batch process scheduling scenarios.

1XXXXX. CO3: To be able to solve process synchronization, and deadlock avoidance problems.

1XXXXX. CO4: To be able to compare different memory and I/O management approaches and use system calls for managing processes, memory and the file system.

1XXXXX. CO5: To be able to understand the structure and organization of the file system.

1XXXXX. CO6: To be able to compare and use different Disk scheduling techniques.

DATABASE MANAGEMENT SYSTEM

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Database Management System	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 4th

THEORY

Learning objectives:

- *To understand the basic concepts and the applications of database systems*
- *To be master the basics of SQL and construct queries using SQL*
- *To understand the relational database design principles*
- *To become familiar with the basic issues of transaction processing and concurrency control*
- *To become familiar with database storage structures and access techniques*

Prerequisite: *Basic computer knowledge and knowledge about Data Structure and Algorithm*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Database System Architecture	4	
Module-II: Data Models	6	
Module-III: Database Design, ER-Diagram and Database Language	10	
Module-IV: Relational Algebra and Relational Calculus	10	
Module-V: Constraints, Views and SQL	6	
Module-VI: Indexing and Transactions	12	

SYLLABUS OUTLINE:

Module-I: Database System Architecture **[4L]**

Introduction, Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML).

Module-II: Data Models [**6L**]

Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Module-III: Database Design, ER-Diagram and Database Language: **[10L]**

Database design and ER Model: overview, ER-Model, Constraints, ER-Diagrams, ERD Issues, weak entity sets, Codd's rules, Relational Schemas, Introduction to UML, Relational database model: Logical view of data, keys, integrity rules. Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF, 4NF).

Module-IV: Relational Algebra and Calculus: [**10L**]

Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison. Calculus: Tuple relational calculus, Domain relational Calculus, calculus vs algebra, computational capabilities.

Module-V: Constraints, Views and SQL: [**6L**]

What is constraints, types of constrains, Integrity constraints, Views: Introduction to views, data independence, security, updates on views, comparison between tables and views SQL: data definition, aggregate function, Null Values, nested sub queries, Joined relations. Triggers.

Module-VI: Indexing and Transactions: [12L]

Indices, B-trees, B+ trees, hashing, Concept of transaction, Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Concurrency Control schemes, Database recovery.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Data base System Concepts, Silberschatz, Korth, McGraw hill, Sixth Edition.

2. Fundamentals of Database Systems, Elmasri Navathe Pearson Education.

Reference Books:

1. An Introduction to Database systems, C.J. Date, A.Kannan, S.Swami Nadhan, Pearson, Eight Edition.

2. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGrawHill 3rd Edition.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	1	1							1
CO2	3		2			1						2
CO3	2	2		2	1							1
CO4			2	1		1						
CO5	2	2			1							
CO6		2	2	1								2
Avg	1.66	1.33	1.33	0.83	0.5	0.33						1

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXX. CO1: To be able to discuss basic concepts, data models, types of users and appreciate the applications of database systems.

1XXXXX. CO2: To be able to understand the logical design of the database including E-R models and the concept of generalization, specialization and aggregation.

1XXXXX. CO3: To be able to apply with a relational database system and Normalization.

1XXXXX. CO4: To be able to explain the basic concepts of relational database design, relational algebra and SQL.

1XXXXX. CO5: To be able to analyze relational database and formulate SQL queries on data.

1XXXXX. CO6: To be able to describe transaction processing and concurrency control concepts.

ARTIFICIAL INTELLIGENCE

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Artificial Intelligence	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 4th

THEORY

Learning objectives: *On completion of the course, student will be able to: Develop problem-solving ability, incorporate knowledge representation, allow continuous learning, encourage social Intelligence, Achieve general intelligence, Promote synergy between humans and AI*

Prerequisite: *Basic computer knowledge and Data Structure and Algorithm*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	6	12
Module-II: Search Techniques	8	18
Module-III: Knowledge & Reasoning	6	12
Module-IV: Probabilistic Reasoning	8	18
Module-V: Natural Language Processing	10	20
Module-VI: Expert Systems	10	20

SYLLABUS OUTLINE:

Module-I: Introduction [8L]

Problems of AI, AI technique, Tic- Tac - Toe problem, games and game playing approaches. Agents & environment, nature of environment, structure of learning agents. Problem space, state space search, problem characteristics, issues in the design of search programs.

Module-II: Search Techniques [8L]

Solving problems by searching: problem solving agents, searching for solutions; uniform search strategies: breadth first search, depth first search, depth limited search, bidirectional search, comparing uniform search strategies. Greedy best-first search, A * search, memory bounded heuristic search: local search algorithms & optimization problems: Hill climbing search, simulated annealing search, local beam search, local search for constraint satisfaction problems. Adversarial Search: Games, optimal decisions & strategies in games, the minimax search procedure, alpha-beta pruning, additional refinements iterative deepening.

Module-III: Knowledge & Reasoning: [4L]

Knowledge representation issues, representation & mapping, approaches to knowledge representation, issues in knowledge representation. Using Predicate Logic: Representing simple fact in logic, representing instant & ISA relationship, computable functions & predicates, resolution, natural deduction.

Module-IV: Probabilistic Reasoning : [6L]

Representing knowledge in an uncertain domain, the semantics of Bayesian networks, Dempster-Shafer theory, Fuzzy sets & fuzzy logics.

Module-V: Natural Language Processing : [4L]

Introduction, syntactic processing, semantic analysis, discourse & pragmatic processing.

Module-VI: Expert Systems : [6L]

Representing and using domain knowledge, expert system shells, knowledge acquisition. Learning: Forms of learning, inductive learning, learning decision trees, explanation based learning, learning using relevance information, neural net learning & genetic learning.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books

1. Russell, Stuart, and Peter Norvig. Artificial Intelligence: A Modern Approach. 3rd ed., Pearson, 2016.

2. Bishop, Christopher M. Pattern Recognition and Machine Learning. Springer, 2006.

3. Goodfellow, Ian, et al. Deep Learning. MIT Press, 2016.

4. Sutton, Richard S., and Andrew G. Barto. Reinforcement Learning: An Introduction. 2nd ed., MIT Press, 2018.

5. Poole, David, and Alan Mackworth. Artificial Intelligence: Foundations of Computational Agents. Cambridge University Press, 2017.

6. Murphy, Kevin P. Machine Learning: A Probabilistic Perspective. MIT Press, 2012.

7. Heaton, Jeff. Artificial Intelligence for Humans. CreateSpace Independent Publishing Platform, 2015.

Reference Books

1. Russell, Stuart, and Peter Norvig. Artificial Intelligence: A Modern Approach. 3rd ed., Pearson, 2016.

2. Bishop, Christopher M. Pattern Recognition and Machine Learning. Springer, 2006.

3. Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016.

4. Sutton, Richard S., and Andrew G. Barto. Reinforcement Learning: An Introduction. 2nd ed., MIT Press, 2018.

5. Poole, David, and Alan Mackworth. Artificial Intelligence: Foundations of Computational Agents. Cambridge University Press, 2017.

6. Murphy, Kevin P. Machine Learning: A Probabilistic Perspective. MIT Press, 2012.

7. Heaton, Jeff. Artificial Intelligence for Humans. CreateSpace Independent Publishing Platform, 2015.

CO-PO Mapping

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

CO2	1	2	2	2	-	-	-	-	-	-	-	-
CO3	-	-	-	2	2	2	-	-	-	-	-	2
CO4	1	3	2	1	-	-	-	-	-	-	-	1
CO5	1	2	-	2	1	-	-	-	-	-	-	1
CO6	-	2	2	2	2	-	-	-	-	-	-	1
Avg	**1**	**1.33**	**1.83**	**1.5**	**0.83**	**0.33**	**-**	**-**	**-**	**-**	**-**	**1**

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

1XXXXX. CO1: To be able to understand the informed and uninformed problem types and apply search strategies to solve them.

1XXXXX. CO2: To be able to apply difficult real life problems in a state space representation so as to solve them using AI techniques like searching and game playing.

1XXXXX. CO3: To be able to design and evaluate intelligent expert models for perception and prediction from intelligent environment.

1XXXXX. CO4: To be able to Identify valid solutions for problems involving uncertain inputs or outcomes by using decision making techniques

1XXXXX. CO5: To be able to demonstrate and enrich knowledge to select and apply AI tools to synthesize information and develop models within constraints of application area.

1XXXXX. CO6: To be able to analyse the issues involved in knowledge bases, reasoning systems and planning

Algorithm-II / Compiler Design / Optimization Techniques / Computer Graphics

Algorithm-II

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Algorithm-II	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 4th

THEORY

Learning objectives: *On completion of the course, student will be able to: analyse Amortized cost of an algorithm, understand Linear time sorting, and Approximation algorithm. Understand Computational Geometry, NP Completeness, and advanced topics like DFT & FFT algorithm; integer multiplication schemes, etc.*

Prerequisite: *Discrete Maths, Programming and Data Structure, Algorithm-I.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	4	15%
Module-II: Sorting	6	20%
Module-III: Graph	6	15%
Module-IV: Selected topics	4	15%
Module-V: Approximation Algorithm	6	15%
Module-VI: NP Completeness	10	20%

SYLLABUS OUTLINE:

Module-I: Introduction [4L]

Amortized complexity: Aggregate Method; Advanced data structures: forward and backward traversal of single linked list, link inversion traversal of binary trees; Binomial heap.

Module-II: Sorting [**6L**]

Topological sort; Sorting networks:0-1 principle, Batcher's odd–even merge sort, Batcher’s bitonic sort

Module-III: Graph [6L]

Max-Flow problem : Ford-Fulkerson algorithm

Module-IV: Selected topics [4L]

Integer exponentiation; Euclid’s algorithm for GCD; FFT algorithm; Polynomial evaluation and multiplication of polynomials; String matching : KMP algorithm; Computational Geometry: line segment properties, convex hull.

Module-V: Approximation Algorithm [**6L**]

Introduction; Travelling Salesman Problem; Vertex Cover Problem, Randomization and Linear Programming.

Module-VI: NP Completeness [10L]

P, NP, NP-hard, NP-complete, 3-SAT problem, NP-completeness and reducibility.

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Ellis Horowitz, Satraj Sahni and S Rajasekharam, Fundamentals of Computer Algorithms, Galgotia publishers

2. Parag Himanshu Dave, Himanshu BhalchandraDave, Design and Analysis algorithms

Pearson Publication

3. M.T. Goodrich, Robert Tamassia, Algorithm design: Foundations, Analysis and Internet examples, Wiley student Edn, John Wiley & sons

Reference Books:

4. M.T. Goodrich, Robert Tamassia, Algorithm design: Foundations, Analysis and Internet examples, Wiley student Edn, John Wiley & sons

5. R C T Lee, Hang and TT Sai, Introduction to Design and Analysis of Algorithms, A strategic approach, TMH

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3											
CO2	1	1	1									
CO3	3	3	3									
CO4	2	2										
CO5	2	2		2	2							
CO6	3	3		3								
Avg	2.33	1.83	0.66	0.83	0.33							

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

1XXXXX. CO1: To be able to apply the Amortized analysis to find the complexity/performance of different algorithms.

1XXXXX. CO2: To be able to understand the concept of linear time sorting.

1XXXXX. CO3: To be able to understand verity of approximation algorithms, such as Vertex cover problem, travelling salesman problem, set covering problem, randomization and linear programming, subset sum problem.

1XXXXX. CO4: To be able to understand the concept of Computational Geometry.

1XXXXX.CO5: To be able to analyse advanced issues related to design and analysis techniques of algorithms and their relation to NP-complete problems.

1XXXXX. CO6: To be able to apply the most suitable algorithm for any given task.

Compiler Design

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Compiler Design	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 4th

THEORY

Learning objectives: *On completion of the course, student will be able to: understand the structure of a compiler, and how the source and target languages influence various choices in its design, understand A new appreciation for programming language features and the implementation challenges they pose, as well as for the actual hardware architecture and the run-time system in which your generated code executes. Students will also understand some specific components of compiler technology, such as lexical analysis, grammars and parsing, type-checking, intermediate representations, static analysis, common optimizations, instruction selection, register allocation, code generation, and run-time organization.*

Prerequisite: *Before learning the concepts of Compiler Design, you should have a basic knowledge Programming for problem solving and Formal Languages and Automata Theory etc.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Compiling	8	
Module-II: Lexical Analysis	5	
Module-III: Syntax Analysis	7	

Module-IV: Syntax directed translation and Type Checking	6	
Module-V: Run time environments and Intermediate Code Generation	5	
Module-VI: Code optimization and Code generations	5	

SYLLABUS OUTLINE:

Module-I: Introduction to Compiling: [8L]

Compilers, Analysis of the source program, the phases of the compiler, Cousins of the compiler.

Module-II: Lexical Analysis: [5L]

The role of the lexical analyzer, Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of a tokens, Finite automata, from a regular expression to an NFA, From a regular expression to NFA, From a regular expression to DFA, Design of a lexical analyzer generator (Lex).

Module-III: Syntax Analysis : [7L]

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), Parser generators (YACC). Error Recovery strategies for different parsing techniques.

Module-IV: Syntax directed translation and Type Checking : [**6L**]

Syntax director definitions, Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes.Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions

Module-V: Run time environments and Intermediate Code Generation : [**5L**]

Source language issues (Activation trees, Control stack, scope of declaration, Binding of names), Storage organization (Subdivision of run-time memory, Activation records), Storage allocation strategies, Parameter passing (call by value, call by reference, copy restore, call by name), Symbol tables, dynamic storage allocation techniques. Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).

Module-VI: Code optimization and Code generations : [5L]

Introduction, Basic blocks & flow graphs, Transformation of basic blocks, Dag representation of basic blocks, the principle sources of optimization, Loops in flow graph, Peephole optimization.Issues in the design of code generator, a simple code generator, Register allocation & assignment.

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Aho, Sethi, Ulman - “Compiler Principles”, Techniques and Tools” - Pearson Education.

2. Computer Organization, Carl Hamachar, Zvonco Vranesic and Safwat Zaky, McGraw Hill.

Reference Books:

1. Holub - “Compiler Design in C” - PHI

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	1	-	-	-	-	-	-	-	-	-	-
CO2	2	2	-	2	1	-	-	-	-	-	-	-
CO3	2	3	-	1	1	-	-	-	-	-	-	-
CO4	2	2	-	1	-	-	-	-	-	-	-	-
CO5	1	1	-	1	-	-	-	-	-	-	-	-
CO6	-	-	-	1	-	-	-	-	-	-	-	1
Avg	1.33	1.5		1	0.33							0.16

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

1XXXXX. CO1: To identify different phases and passes of the compiler and also able to use the compiler tools.

1XXXXX. CO2: To able to analyze and compare different types of compiler tools to meet the requirements of the realistic constraints of compilers

1XXXXX. CO3: To understand the parser and its types i.e. Top-Down and Bottom-up parsers and construction of LL, SLR, CLR, and LALR parsing table and evaluate the issues

1XXXXX. CO4: To Construct the compiler using syntax-directed translation method and get knowledge about the synthesized and inherited attributes.

1XXXXX.CO5: To collect knowledge about run time data structure like symbol table organization and different techniques used in that.

1XXXXX. CO6: To understand the target machine's run time environment, its instruction set for code generation and techniques used for code optimization.

Optimization Techniques

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Optimization Techniques	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 4th

THEORY

Learning objectives: *On completion of the course, student will be able to: apply the knowledge of linear programming problem, queuing theory, inventory control to solve complex engineering problems.*

Prerequisite: *Before learning the concepts of Optimization Techniques, you should have a basic knowledge of set, vector space, probability theory..*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module I: Introduction to OR	2	
Module II: Linear Programming	8	
Module III: Transportation and Assignment problems	6	
Module IV: PERT – CPM	6	
Module V: Inventory Control	6	
Module VI: Queuing Theory	8	

SYLLABUS OUTLINE:

Module I: Introduction to OR

Origin of OR and its definition. Types of OR problems, Deterministic vs. Stochastic optimization, Phases of OR problem approach – problem formulation, building mathematical model, deriving solutions, validating model, controlling and implementing solution.

Module II: Linear Programming

Linear programming – Examples from industrial cases, formulation & definitions, Matrix form. Implicit assumptions of LPP. Some basic concepts and results of linear algebra – Vectors, Matrices, Linear Independence /Dependence of vectors, Rank, Basis, System of linear eqns., Hyper plane, Convex set, Convex polyhedron, Extreme points, Basic feasible solutions. Geometric method: 2-variable case, Special cases – infeasibility, unboundedness, redundancy &degeneracy, Sensitivity analysis. Simplex Algorithm – slack, surplus & artificial variables, computational details, big-M method, identification and resolution of special cases through simplex iterations. Duality – formulation, results, fundamental theorem of duality, dual-simplex and primal-dual algorithms.

Module III: Transportation and Assignment problems

TP - Examples, Definitions – decision variables, supply & demand constraints, formulation, Balanced & unbalanced situations, Solution methods – NWCR, minimum cost and VAM, test for optimality (MODI method), degeneracy and its resolution. AP - Examples, Definitions – decision variables, constraints, formulation, Balanced & unbalanced situations, Solution method – Hungarian, test for optimality (MODI method), degeneracy & its resolution.

Module IV: PERT – CPM

Project definition, Project scheduling techniques – Gantt chart, PERT & CPM, Determination of critical paths, Estimation of Project time and its variance in PERT using statistical principles, Concept of project crashing/time-cost trade-off.

Module V: Inventory Control

Functions of inventory and its disadvantages, Concept of inventory costs, Basics of inventory policy (order, lead time, types), Fixed order-quantity models – EOQ, POQ & Quantity discount models.

Module VI: Queuing Theory

Definitions – queue (waiting line), waiting costs, characteristics (arrival, queue, service discipline) of queuing system, queue types (channel vs. phase).Kendall’s notation, Little’s law, steady state behavior, Poisson’s Process & queue, Models with examples - M/M/1 and its performance measures; M/M/m and its performance measures; brief description about some special models.

Text Books:

1. Operations Research: An Introduction. H.A. Taha.

Reference Books:

1. Linear Programming. K.G. Murthy.
2. Linear Programming. G. Hadley.
3. Principles of OR with Application to Managerial Decisions. H.M. Wagner.

4. Introduction to Operations Research. F.S. Hiller and G.J. Lieberman.
5. Elements of Queuing Theory. Thomas L. Saaty.
6. Operations Research and Management Science, Hand Book: Edited By A. Ravi Ravindran.
7. Management Guide to PERT/CPM. Wiest & Levy.
8. Modern Inventory Management. J.W. Prichard and R.H. Eagle.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	-	-	-	-	-	-	-	-	-	-	1
CO2	3	3	-	-	-	-	-	-	-	-	-	1
CO3	3	3	-	-	-	-	-	-	-	-	-	-
CO4	3	-	-	-	-	-	-	-	-	-	-	-
CO5	3	-	-	-	-	-	-	-	-	-	-	1
CO6	2	3	-	-	-	-	-	-	-	-	-	1
Avg	3	2	-	-	-	-	-	-	-	-	-	1

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

After attending this course the students will be able to

XX.CO1: Understand the concept of Operations Research and the basic concepts linear algebra.

XX.CO2: Formulate Mathematical Model of various optimization problems and solve linear programming problems using appropriate techniques.

XX.CO3: Determine optimal strategy for Transportation and Assignment problems.

XX.CO4: Determine the critical path, project time and its variance using the project scheduling techniques – Gantt chart, PERT & CPM.

XX.CO5: Understand the concept of inventory costs, Basics of inventory policy and fixed order-quantity models like EOQ, POQ.

XX.CO6: Understand the concept of queuing theory and identify the queuing models like M/M/1 and M/M/m.

Computer Graphics

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Graphics	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 4th

MODULE-I: BASICS OF COMPUTER GRAPHICS

Introduction, what is computer Graphics? Area of Computer Graphics, Design and Drawing, Animation Multimedia applications, Simulation, How are pictures actually stored and displayed, Difficulties for displaying pictures.

MODULE-II: GRAPHIC DEVICES

Cathode Ray Tube, Quality of Phosphors, CRTs for Color Display, Beam Penetration CRT, The Shadow - Mask CRT, Direct View Storage Tube, Tablets, The light Pen, Three Dimensional Devices C Graphics Basics, Graphics programming, initializing the graphics, C Graphical functions, simple programs

Point Plotting Techniques, Qualities of good line drawing algorithms, The Digital Differential Analyzer (DDA), Bresenham's Algorithm, and Generation of Circles.

MODULE-III: TWO DIMENSIONAL TRANSFORMATIONS and CLIPPING AND WINDOWING

What is transformation?, Matrix representation of points, Basic transformation, Need for Clipping and Windowing, Line Clipping Algorithms, The midpoint subdivision Method, Other Clipping Methods, Sutherland – Hodgeman Algorithm, Viewing Transformations

MODULE-IV: GRAPHICAL INPUT TECHNIQUES

Graphical Input Techniques, Positioning Techniques, Positional Constraints, Rubber band Techniques, Need for 3-Dimensional Imaging, Techniques for 3-Dimesional displaying, Parallel Projections, Perspective projection, Intensity cues, Stereoscope effect,
Kinetic depth effect, Shading

MODULE-V: SOLID AREA SCAN CONVERSION AND THREE DIMENSIONAL TRANSFORMATIONS

Solid Area Scan Conversion, Scan Conversion of Polygons, Algorithm Singularity, Three Dimensional transformation, Translations, Scaling, Rotation, Viewing Transformation, The Perspective, Algorithms, Three Dimensional Clipping, Perspective view of Cube

MODULE-VI: HIDDEN SURFACE REMOVAL

Need for hidden surface removal, The Depth - Buffer Algorithm, Properties that help in reducing efforts, Scan Line coherence algorithm, Span – Coherence algorithm, Area-Coherence Algorithms, Warnock's Algorithm, Priority Algorithms

Soft-Skill Development-IV

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

MDC3: Selected by candidate from Other Discipline

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Foreign language-II

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Human Values and Ethics

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Operating Systems Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Operating Systems Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 4th

PRACTICAL (**total contact hours,** hr/week), if applicable

Learning objectives:

To familiarize the students with the Operating System.

To demonstrate the process, memory, file and directory management issues under the UNIX/ LINUX operating system

To introduce LINUX shell script programming.

List of practical

Tentative Experiment Name

Section 1:

a) Write a shell script to take the name of the user as input and print it.
b) Write a shell script to multiply two numbers and display the output.
c) Write a shell script program to emulate the calculator function.
d) Write a shell script that will find the maximum from the given three no.
e) Write a shell script that will find the GCD of two given numbers.
f).Write a shell script to generate a Fibonacci series of length with the first two no. of the series is 3 and 5 respectively.\
g) Write a Shell script to take 'n' number of elements in an array and print the third largest number. Value of 'n' must be taken from the terminal.
h)Store ‘n’ number of elements in an array and find out the sum of the array elements. Value of 'n' must be taken from the terminal.
i) Write a shell program that will accept 10 numbers from the
terminals and will search the position of a given no in the supplied nos.
j) Write a program in C under Linux to create a file.
k) Write a shell script program to search an integer in an array using linear search.

Section 2:

a) Write a C Program that will create a child process. Then print the process id & parent process id both from the child as well as from the parent.

b) Write a C program that will create a child process. Then modify the value of a globally defined variable from the child process and print the value of the variable from the parent process.
c) Write a c program that will take the name of person as command line argument and then it will print hallo name. Then write another program that will create a process using fork(). Then execute the previously created c program (hallo program) by the child process.
d) Write a program in C under Linux to copy the content of one file to another from command line.
e) Write a program in C to implement LRU page replacement algorithm
f) Write a program in C to implement CPU scheduling using Round Robin Scheduling algorithm
g) Write a program in C to implement CPU scheduling using FCFS Scheduling algorithm
h) Write a program in C to implement CPU scheduling using SJF Scheduling algorithm.

Section 3

Write a C program for implementing the Producer Consumer problem using Thread Synchronization.
Write a C program to count a number from 1 to 20 using two threads (Thread 1 and Thread 2) where the prime numbers are printed by Thread 1 and non prime numbers are printed by Thread 2.

Section 4

a)Write a program in C that demonstrates how two processes can share a variable using semaphore.
b) Write a C program to implement Semaphore to print from a parent as well as a child process, where both parent and child will print two consecutive words from a sentence.
c) Write a program in C to solve the Producer Consumer problem using POSIX semaphore.

Section 5

a) Write Unix Commands to do the following directory manipulation.

i. Display the absolute path of your home directory.
ii. Create a new subdirectory called 'Sister Nivedita University' in your home directory.
iii. Create a new subdirectory called 'Student' in Sister Nivedita University .
iv. Create a new subdirectory called 'Teacher' in Sister Nivedita University.
v. Display the contents of the directory 'Sister Nivedita University' .
vi. Delete the directory 'Teacher'.
vii. Display the contents of the directory 'Sister Nivedita University' in detail .

b). Write a program to create a pipe between parent and child and to send data down the pipe.

c) Write a program to convert lower case to upper case using FIFO pipe where the client sends a string in lower case to the server and the server responds with the string in Upper case back to the client.
d). Write a program to implement IPC using shared memory between two processes.
e).Write a program to implement IPC using message queue between two processes.
f) Write a C program to calculate the seek time by applying FCFS, SSTF, SCAN,C-SCAN algorithms

Database Management System Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Database Management System Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 4th

PRACTICAL (**total contact hours,** hr/week), if applicable

Learning objectives:

The students will Working on existing database systems, designing of database, creating relational database, analysis of table design. The lab course also provide practical knowledge to understand advanced database concepts.

List of practical

Introduction to SQL:
Basic concepts of databases and DBMS
Introduction to Structured Query Language (SQL)
Creating, querying, updating, and deleting tables using SQL

Data Definition Language (DDL):
Creating and modifying database schema using DDL commands (CREATE, ALTER, DROP)
Constraints (PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK)

Data Manipulation Language (DML):
Inserting, updating, and deleting data using DML commands (INSERT, UPDATE, DELETE)
Retrieving data using SELECT statement
Filtering data using WHERE clause
Sorting data using ORDER BY clause

Aggregating data using GROUP BY and aggregate functions (SUM, AVG, COUNT, MAX, MIN)

Joins and Subqueries:

Performing joins (INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN)

Writing subqueries to retrieve data

Understanding correlated subqueries

Indexes and Views:

Creating indexes for efficient data retrieval

Creating and managing views

Understanding materialized views

Transactions and Concurrency Control:

Introduction to transactions

ACID properties of transactions

Isolation levels (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, SERIALIZABLE)

Locking mechanisms for concurrency control

Database Connectivity:

Connecting to databases using programming languages (e.g., Java, Python) and APIs (e.g., JDBC, SQLAlchemy)

Performing CRUD operations through programming languages

Database Administration:

Managing users and permissions

Backup and recovery strategies

Monitoring database performance

Tuning SQL queries for better performance

Normalization:

Understanding normalization forms (1NF, 2NF, 3NF, BCNF)

Applying normalization techniques to improve database design

Stored Procedures and Triggers:

Creating and executing stored procedures

Defining and executing triggers

Database Design Project:

Students may be assigned a database design project where they have to design a database schema, implement it using SQL, and develop a simple application to interact with the database.

Artificial Intelligence Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Artificial Intelligence Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 4th

PRACTICAL (**total contact hours,** hr/week), if applicable

Learning objectives:

The students will learn the different applications and Programs Using SWI Prolog and Python Programming

List of practical

Exp. No.	*Experiment Name*	*CO Mapping*
1.	Execute the Basic Operations of SWI Prolog and Python along with the installation process of Python Jupyter Notebook and SWI Prolog	**CO1**
2.	Implementation of relational tree structure in SWI Prolog	**CO1**
3.	Implementation of Circuit Design Logic Using SWI Prolog	**CO2**
4.	Implementation of Predecessors and Successors in SWI Prolog	**CO2**
5.	Implementation of Graph Colouring(Vertices, Edges, Regions) in SWI Prolog	**CO3**
6.	Implementation of Greedy Algorithm using Python	**CO4**
7.	Hill Climbing and A* Algorithm using Python	**CO5**
8.	Implement BFS and DFS using Python	**CO5**
9.	Implement the Tower of Hanoi using SWI Prolog and Python	**CO5**
10.	Implement BFS and DFS using Python	**CO6**
11.	4 Queens Problem using Python	**CO5**
12.	Basic implementations and innovative algorithm design using of 2 Fuzzy Sets like Union, Intersection, Negation etc. using Python	**CO6**
13.	Case Study with Analysis	**CO6**

THIRD YEAR

SEMESTER-V

Sl No	Course Title	Code		Credit	Type			
					L	T	P	S
1	Computer Networks		MC	4	4	0	0	0
2	Software Engineering		MC	4	4	0	0	0
3	Digital Image Processing/Machine Learning		ME	3	3	0	0	0
4	NM Elective-I		NM	4	4	0	0	0
5	Soft-Skill Development-V		NV	1	1	0	0	0
6	Mentored Seminar-I		NV	2	0	0	0	2
7	SEC2:Current Programming Techniques		SEC	3	3	0	0	0
8	Computer Networks Lab		MC	1	0	0	2	0
9	Software Engineering Lab		MC	1	0	0	2	0
10	Digital Image Processing Lab /Machine Learning Lab		ME	1	0	0	2	0
Total Credit				**24 Credit**				

COMPUTER NETWORKS

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Networks	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 5th

THEORY

Learning objectives: *On completion of the course, student will be able to:*

- *Describe the general principles of data communication.*
- *Describe how computer networks are organized with the concept of layered approach.*
- *Describe how signals are used to transfer data between nodes.*
- *Implement a simple LAN with hubs, bridges and switches.*
- *Describe how packets in the Internet are delivered.*
- *Analyze the contents in a given data link layer packet, based on the layer concept.*
- *Design logical sub-address blocks with a given address block.*
- *Decide routing entries given a simple example of network topology*
- *Describe what classless addressing scheme is.*
- *Describe how routing protocols work.*
- *Use C programming language to implement network programs.*

Design and implement a network protocol.

Prerequisite: *Analog and Data Communication, Algorithm, and Programming logic .*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Data Communication	4	8
Module-II: Physical layer and Media	10	21
Module-III: Data Link Layer and Medium Access Sub Layer	12	25
Module-IV: Network Layer	10	21
Module-V: Transport Layer	6	13
Module-VI: Application Layer	6	13

SYLLABUS OUTLINE:

Module-I: Introduction to Data Communication [4L]

Components, Representation of data and its flow networks, Physical structures, Connection Topology, Protocols and Standards, OSI model, TCP/IP Protocol suite, Addressing.

Module-II: Physical layer and Media [10L]

Analog and Digital data, Signals, Transmission impairment, Data rate limit and Performance, Digital to Digital conversion, Analog to Digital conversion, Digital to Analog conversion, Analog to Analog conversion, Multiplexing and Spectrum Spreading: Multiplexing, Spread Spectrum, Transmission media: Guided Media, Unguided Media, Switching: Introduction, circuit switched networks, packet switched network, switching, structure of a switch.

Module-III: Data Link Layer and Medium Access Sub Layer: [12L]

Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA,CSMA/CD,CDMA/CA. HDLC, Ethernet, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN.

Module-IV: Network Layer : [10L]

Logical addressing – IPV4, IPV6; Address mapping – and DHCP–Delivery, Forwarding Unicast Routing protocols: RIP, OSPF, BGP Multicast Routing Protocol:

Module-V: Transport Layer : [6L]

Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP)s; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

Module-VI: Application Layer: **[6L]**

Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP.

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Computer Networks, Andrew. S. Tanenbaum, 4/e, Prentice Hall of India Private Ltd, 2003.

2. Data Communications and Networking, Behrouz A Forouzan, 4/e, Tata McGraw Hill Education Private Limited.

Reference Books:

1. Data Communications & Networks, Achyut S. Godbole, Tata McGraw Hill Education Private Limited, 2002.

2. Data and Computer Communication, William Stalling, 7/e, Prentice Hall of India Private Ltd, 2007.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	2	2	1	2	-	-	-	-	-	-	-	1
CO2	3	3	2	3	-	-	-	-	-	-	-	1
CO3	3	3	2	2	2	-	-	-	-	-	-	-
CO4	2	3	3	3	2	2	-	-	-	-	-	-
CO5	3	2	1	2	2	2	-	-	-	-	-	-
CO6	2	2	1	1	2	-	-	-	-	-	-	1

Highly Correlated: **3**
Moderately Correlated**: 2**

Slightly Correlated: **1**

***Course learning outcome**: (CO)*

1XXXXX. CO1: To be able to understand data communication components, representation of data, physical topologies and protocols.

1XXXXX. CO2: To be able to understand Analog and Digital transmission, multiplexing and working of transmission media.

1XXXXXX. CO3: To be able to solve problems related to error correction/detection and protocols of media access control layer.

1XXXXXX. CO4: To be able to solve IP subnetting problems and routing problems.

1XXXXXX. CO5: To analyze basic operations of transport layer and congestion control mechanisms.

1XXXXX. CO6: To be able to understand about various application layer functionalities.

Software Engineering

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Software Engineering	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 5th

THEORY

Learning objectives: *On completion of the course, student will be able to:*

1. Students will be able to decompose the given project in various phases of a lifecycle.
2. Students will be able to choose appropriate process model deProvided by Respective Department / School on the user requirements.
3. Students will be able perform various life cycle activities like Analysis, Design, Implementation, Testing and Maintenance.
4. Students will be able to know various processes used in all the phases of the product.
5. Students can apply the knowledge, techniques, and skills in the development of a software product.

Prerequisite: *Basic computer knowledge and Data Structure and Algorithm*

Course content/Syllabus:

Module no.	No of	Weightage (%)

	lecture/Contact hour	
Module-I: SDLC Models	9	
Module-II: System Design	9	
Module-III: Coding & Documentation	6	
Module-IV: Testing	8	
Module-V: Software Project Management	8	
Module-VI: Modelling Techniques	8	

SYLLABUS OUTLINE:

Module-I: SDLC Models. [9L]

System Concept, System Development Life Cycle, Waterfall Model ,Spiral Model, Feasibility Analysis, Technical Feasibility, Cost- Benefit Analysis, COCOMO model.

Module-II: System Design [9L]

Context diagram and DFD, Problem Partitioning, Top-Down and Bottom-Up design; Decision tree, decision table and structured English; Functional vs. Object- Oriented approach.

Module-III: Coding & Documentation: [6L]

Structured Programming, OO Programming, Information Hiding, Reuse, System Documentation.

Module-IV: Testing [8L]

Levels of Testing, Integration Testing, Test case Specification, Reliability Assessment, Validation & Verification Metrics, Monitoring & Control.

Module-V: Software Project Management : [8L]

Project Scheduling, Staffing, Software Configuration Management, Quality Assurance, Project Monitoring.

Module-VI: Modelling Techniques : [8L]

Static and dynamic models, why modelling, UML diagrams: Class diagram, interaction diagram: collaboration diagram, sequence diagram, state chart diagram, activity diagram, and implementation diagram.

Pedagogy for Course Delivery: *Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA):*NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Sommerville, Ian. Software Engineering. 10th ed., Addison-Wesley, 2015.
2. Pressman, Roger S. Software Engineering: A Practitioner's Approach. 9th ed., McGraw-Hill Education, 2021.
3. Pfleeger, Shari Lawrence, Joanne M. Atlee, and Robert L. Glass. Software Engineering: Theory and Practice. 4th ed., Pearson, 2014.
4. Ian, Mauro Pezzè, and Michal Young. Software Testing and Analysis: Process, Principles, and Techniques. Wiley, 2007.

Reference Books:

1. Ghezzi, Carlo, Mehdi Jazayeri, and Dino Mandrioli. Fundamentals of Software Engineering. 2nd ed., Prentice Hall, 2010.
2. Bass, Len, Paul Clements, and Rick Kazman. Software Architecture in Practice. 3rd ed., Addison-Wesley, 2012.
3. Shaw, Mary, and David Garlan. Software Architecture: Perspectives on an Emerging Discipline. Prentice Hall, 1996.
4. Wazlawick, Raul Sidnei. Object-Oriented Analysis and Design for Information Systems: Modeling with UML, OCL, and IFML. Elsevier, 2014.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		2		1						1		
CO2	2	2	3									
CO3	3	2	3			2						
CO4		2							1	1		
CO5	1	2		1					2			1
CO6		1	3		3					1		1
Avg	1	1.83	1.5	.33	0.5	0.33			0.5	0.5		0.33

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

1XXXXX. CO1: Ability to apply software engineering principles and techniques and understand the SDLC, SRS.

1XXXXX. CO2: Ability to develop, maintain and evaluate software design.

1XXXXX. CO3: Analyze the coding standard and justify the code with different testing techniques.

1XXXXX. CO4: Apply the knowledge of system design for testing software in various environment

1XXXXX. CO5: Estimate the scheduling and budgeting for maintaining the project management, and Illustrate the quality control and maintenance of software.

1XXXXX. CO6: To be able to analyze the interaction among various model in a software design using Class diagram, interaction diagram: collaboration diagram, sequence diagram, state chart diagram, activity diagram, and implementation diagram.

Digital Image Processing / Machine Learning / Internet of Things

Digital Image Processing

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Digital Image Processing	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 5th

THEORY

Learning objectives: *On completion of the course, student will be able to: Demonstrate basic concept of image processing concepts related to different types of application like satellite imaging, geostationary images etc.*

Prerequisite: *A strong mathematical background. Programming skills in Python, MATLAB, etc.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Image processing	6	

Module-II: Image Enhancement Techniques	6	
Module-III: Image Segmentation	6	
Module-IV: Morphological Operations	6	
Module-V: Image Registration	4	
Module-VI: Color Image Processing	8	

SYLLABUS OUTLINE:

Module-I: Introduction to Image processing: [6L]

Introduction to Image processing: Fundamental steps in image processing; Components of image processing system; Pixels; coordinate conventions; Imaging Geometry; Spatial Domain; Frequency Domain; sampling and quantization; Basic relationship between pixels; Applications of Image Processing.

Module-II: Image Enhancement Techniques: [**6L**]

Enhancement, Contrast Stretching; Gray Level Slicing; Bit Plane Slicing; Histogram Processing–Equalization; Specification. Basics of Spatial Filtering – Smoothing: Smoothing Linear Filters; local contrast enhancement; sharpening, spatial convolution, Gaussian smoothing, DoG, LoG.

Module-III: Image Segmentation: [6L]

Image Segmentation: Pixel-Based Approach- Multi-Level Thresholding, Local Thresholding, Threshold Detection Method; Region-Based Approach- Region Growing Based Segmentation, Region Splitting, Region Merging, Split and Merge, Edge Detection - Edge Operators; Line Detection, Corner Detection.

Module-IV: Morphological Operations: [**6L**]

Morphological Operations Basics of Set Theory; Dilation and Erosion - Dilation, Erosion; Structuring Element; Opening and Closing; Hit or Miss Transformation. Representation and Description Representation - Boundary, Chain codes, Polygonal approximation approaches, Boundary segments

Module-V: Image Registration: [**4L**]

Mono-modal/multimodal image registration; Global/local registration; Transform and similarity measures for registration; Intensity/pixel interpolation.

Module-VI: Color Image Processing: [8L]

Fundamentals of different colour models - RGB, CMY, HSI, YCbCr, Lab; False colour; Pseudo colour, Case studies

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*: NA*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Gonzalez, Rafael C., and Richard E. Woods. Digital Image Processing. 4th ed., Pearson, 2018.
2. Burger, Wilhelm, and Mark J. Burge. Digital Image Processing: An Algorithmic Introduction Using Java. Springer, 2016.
3. Sonka, Milan, Vaclav Hlavac, and Roger Boyle. Image Processing, Analysis, and Machine Vision. 4th ed., Cengage Learning, 2018.
4. Jain, Anil K. Fundamentals of Digital Image Processing. Prentice Hall, 1989.

Reference Books:

1. Woods, Richard E., and Steven L. Eddins. Digital Image Processing Using MATLAB. 2nd ed., Gatesmark Publishing, 2009.
2. Gonzalez, Rafael C., Richard E. Woods, and Steven L. Eddins. Digital Image Processing Using MATLAB. 3rd ed., Gatesmark Publishing, 2020.

CO-PO Mapping

CO Code	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO.1	3	3	2	-	2	2	-	-	-	-	-	1
CO.2	3	-	2	-	-	-	-	-	-	-	-	-
CO.3	-	3	-	2	-	-	-	-	-	-	-	1
CO.4	3	2	-	-	1	2	-	-	-	-	-	-
CO.5	2	-	2	-	-	2	-	-	-	-	-	-
CO.6	2	3	2	2	-	2	-	-	-	-	-	2
Avg	2.16	1.83	1.33	0.66	0.5	1.33	0	0	0	0	0	0.66

Highly Correlated: **3**

Moderately Correlated**: 2**

Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXX. CO1: To be able to understand basic fundamental concepts of image processing.

1XXXXX. CO2: To be able to implement various image enhancement techniques.

1XXXXX. CO3: To be able to apply different segmentation techniques based on the input image property.

1XXXXX. CO4: To be able to apply various morphological operations on various image.

1XXXXX. CO5: To be able to compare among image registration operations.

1XXXXX. CO6: To be able to establish new image processing techniques for preserving images.

MACHINE LEARNING

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Machine Learning	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 5th

THEORY

Learning objectives:

The objective of the course is

- *To understand the basic theory underlying machine learning.*
- *To be able to formulate machine learning problems corresponding to different applications.*
- *To understand a range of machine learning algorithms along with their strengths and weaknesses.*
- *To be able to apply machine learning algorithms to solve problems of moderate complexity.*
- *To apply the algorithms to a real-world problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.*

Prerequisite: *Knowledge of Artificial Intelligence, Linear algebra, Calculus, Mathematical logic and differential equation*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Machine Learning	4	
Module-II: Feature Engineering	7	
Module-III: Classification	8	
Module-IV: Clustering	7	
Module-V: Machine Learning System Design	6	

Module-VI: Case studies	4	

SYLLABUS OUTLINE:

Module-I: Introduction to Machine Learning [4L]

Basic Concepts of Machine Learning, Types of Machine Learning, Supervised Learning Versus Unsupervised Learning Versus Reinforcement Learning, Discriminative Algorithms.

Module-II: Feature Engineering [**7L**]

Introduction to Data Processing, ETL, Measurement of Purity, Entropy and Gini Index, Normalization and Standardization, Dimension Reduction, ICA (Independent Components Analysis), EM. Mixture of Gaussians, Factor Analysis, Normal Distribution and Gaussian Distribution.

Module-III: Classification [8L]

Introduction to Supervised Learning, Concepts of Linear Algebra, Linear Regression and Logistic Regression, Concepts Bias/ Variance Trade off, Prediction Versus Classification Problem, Naive Bayes, Maximum Entropy, Perceptron, Basic Concept of Neural Network, Generative Learning Algorithms, Gradient Descent, Regularization, Feed Forward Neural Network, Back Propagation Neural Network, Gaussian Discriminant Analysis, Concepts of vectorization , Support Vector Machines, Introduction of Deep Learning, Hidden Markov Model, Genetic Algorithms,

Module-IV: Clustering [7L]

Introduction to Unsupervised learning: Introduction to Clustering, K-means and Hierarchical Clustering, Comparison among classification and clustering, Dimension reduction: PCA (Principal Components Analysis), Factor analysis.

Module-V: Machine Learning System design [6L]

Underfitting and Overfitting Problem, Bias-Variance as Function of Lambda, Cross Validation, Learning Curves, Error Analysis, Confusion Matrix, Trading off Precision and Recall, ROC Curve, F1-Score and Accuracy Analysis

Module-VI: Case studies [4L]

Applications of ML in Case Studies.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Bishop, Christopher M. Pattern Recognition and Machine Learning. Springer, 2006.
2. Alpaydin, Ethem. Introduction to Machine Learning. 3rd ed., The MIT Press, 2014.
3. Murphy, Kevin P. Machine Learning: A Probabilistic Perspective. The MIT Press, 2012.
4. Marsland, Stephen. Machine Learning: An Algorithmic Perspective. 2nd ed., CRC Press, 2014.

Reference Books

1. Shalev-Shwartz, Shai, and Shai Ben-David. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, 2014.
2. Géron, Aurélien. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd ed., O'Reilly Media, 2019.
3. Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer, 2009.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	1								1
CO2	3	2		2	1							2
CO3	2	2	3	2	2							2
CO4	1	2	3	3	2							3
CO5	2	2	3	3	3							3
CO6		2	3	3	2							
Avg	**2.20**	**2.17**	**2.80**	**2.33**	**2.00**							**2.20**

Highly Correlated: **3**

Moderately Correlated**: 2**

Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXX. CO1: To be able to discuss the basics of learning problems with hypothesis

1XXXXX. CO2: To be able to understand the features of machine learning to deal with real world problems

1XXXXXX. CO3: To be able to differentiate the machine learning algorithms as supervised learning and unsupervised learning

1XXXXXX. CO4: To be able to design and analyze various classification and clustering algorithms

1XXXXXX. CO5: To be able to develop and tune the machine learning models with datasets

1XXXXX. CO6: To be able to evaluate the models for optimization engineering problems

NM Elective-I

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Soft-Skill Development-V

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Mentored Seminar-I

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

SEC2: Current Programming Techniques

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Computer Networks Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Network Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 5th

S.no	Experiment	CO
1.	a)Write a program to calculate hamming distance between two bytes mechanisms	CO3
2.	a) Write a C Program to implement Echo server using TCP/IP protocol. b) Write a C Program to implement Echo server using UDP protocol.	CO5
3.	a) Write a C Program to implement Chat server using TCP/IP protocol. b) Write a C Program to implement Chat server using UDP protocol.	CO5
4.	a) Write a C Program to implement Concurrent server using TCP/IP protocol.	CO5
5.	a) Write a C Program to implement Time server using TCP/IP protocol.	CO5
6.	a) Write a C Program to implement File server using UDP protocol.	CO5
7.	a) Write a C Program to implement Calculator server using TCP protocol.	CO5
8.	a) Write a C Program to implement Multicasting using UDP protocol.	CO4

Software Engineering Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Database Management System Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 5th

Course Outcomes:

1 To understand the software engineering methodologies involved in the phases for project development.
2 To gain the knowledge about the Software Project Management.

3 In Software Engineering Lab, design a project proposal which will be used throughout the lab for performing different experiments using CASE Tools.
Project Schedule preparation using tools like MS Project. ➢ Arrange a project in Work Break down Structure (WBS) and ➢ Design Gantt from schedule.
➢ Design PERT chart and estimate the duration from schedule.
4. SRS Design
5. DFD Design
6. UML Design

Laboratory Experiments:

- Problem Analysis and Project Planning -Thorough study of the problem – Identify Project scope, Objectives and Infrastructure.
- Software Requirement Analysis – Describe the individual Phases/modules of the project and Identify deliverables. Identify functional and non-functional requirements.
- Data Modeling – SRS Design - Use work products – data dictionary.
- Software Designing - Develop use case diagrams and activity diagrams, build and test class diagrams, sequence diagrams and add interface to class diagrams.
- Prototype model – Develop the prototype of the product.
- The SRS and prototype model should be submitted for end semester examination.
- Any experiment specially designed by the college

(Detailed instructions for Laboratory Manual to be followed for further guidance)

Digital Image Processing Lab /Machine Learning Lab /Internet of Things Lab

Digital Image Processing Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Digital Image Processing Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 5th

PRACTICAL (**total contact hours,** hr/week), if applicable

Learning objectives:

The students will learn the different image processing techniques and able to apply these using Matlab / Python Programming

List of practical

Introduction to Image Processing Tools:

Overview of digital image processing
Introduction to commonly used software tools such as MATLAB, OpenCV, or similar platforms

Image Enhancement:
Basic image enhancement techniques (e.g., histogram equalization, contrast stretching)
Spatial domain methods (e.g., spatial filtering, neighborhood processing)
Frequency domain methods (e.g., Fourier transform, filtering in the frequency domain)

Image Restoration:
Image degradation and models
Image restoration techniques (e.g., inverse filtering, Wiener filtering)
Noise reduction methods (e.g., mean filtering, median filtering)

Color Image Processing:
Color models (RGB, HSI, CMYK, etc.)
Color image enhancement and restoration techniques
Color image segmentation and feature extraction

Image Compression:
Introduction to image compression concepts
Lossless and lossy compression techniques
Image compression standards (e.g., JPEG, JPEG2000)

Image Segmentation:
Image segmentation fundamentals
Thresholding techniques
Region-based segmentation methods (e.g., region growing, split and merge)

Feature Extraction and Object Recognition:
Feature extraction techniques (e.g., edge detection, corner detection)
Object recognition methods (e.g., template matching, Hough transform)

Morphological Image Processing:
Basic morphological operations (e.g., dilation, erosion)
Structuring elements and their properties
Applications of morphological operations (e.g., image filtering, segmentation)

Image Analysis and Pattern Recognition:
Image analysis fundamentals
Pattern recognition techniques (e.g., classification, clustering)
Applications of image analysis and pattern recognition in various fields

Machine Learning Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Machine Learning Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 5th

PRACTICAL (**total contact hours,** hr/week), if applicable

Learning objectives:

The students will learn about different machine learning algorithms and able to apply those using Python Programming.

List of practical

1. Write a Program to perform the following operations on matrices

a) Matrix addition
b) Matrix Subtraction
c) Matrix Multiplication
d) Matrix Inversion
e) Transpose of a Matrix
2. Write a Program to perform the following operations

a) Find the minimum and maximum element of the matrix
b) Find the minimum and maximum element of each row in the matrix
c) Find the minimum and maximum element of each column in the matrix
d) Find trace of the given matrix
e) Find rank of the given matrix
f) Find eigenvalues and eigenvectors of the given matrix

3. Write a Program to find the mean, median, standard deviation and mode using user defined functions.

4. Create a data frame with columns at least 5 observations

a) Retrieve a particular column from the DataFrame
b) Summarize the data frame and observe the statistics of the DataFrame created
c) Observe the mean and standard deviation of the data frame and print the values.

5. Write a program to implement the Linear Regression for a sample training data set stored as a .CSV file. Compute Mean Square Error by considering few test data sets.

6. Write a program to implement the Non-linear Regression for a sample training data set stored as a .CSV file. Compute Mean Square Error by considering few test data sets.

7. Write a program to implement the Logistic Regression for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier.

8. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

9. Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions.

10. Write a program to implement Support Vector Machine algorithm to classify the iris data set. Print both correct and wrong predictions.

11. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

12. Write a program to demonstrate the working of the decision tree based CART algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

13. Write a program to construct a Regression tree for cost estimation by assuming any numerical dataset.

14. Write a program to calculate the accuracy, precision, and recall for your data set. Assume a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task.

15. Implement a single neural network and test for different logic gates.

16. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.

REFERENCES:

1. Vijayvargia, Abhishek, Machine Learning with Python: An Approach to Applied Machine

Learning, BPB Publications, 1st edition,2018.

2. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow, Oreilly, March 2017.

3. Dr. M Gopal, Applied Machine Learning, 1st Edition, McGraw-Hill,2018

THIRD YEAR

SEMESTER-VI

Sl No	Course Title	Code		Credit	Type			
					L	T	P	S
1	Introduction to Data Science		MC	4	4	0	0	0
2	Cryptography & Network Security/Artificial Neural Networks		ME	4	4	0	0	0
3	Cloud Computing/Soft Computing		ME	4	4	0	0	0
4	NM Elective-II		NM	4	4	0	0	0
5	Soft-Skill Development-VI		NV	1	1	0	0	0
6	Mentored Seminar-II		NV	2	0	0	0	2
7	SEC3:Logical Ability		SEC	3	3	0	0	0
8	Introduction to Data Science Lab		MC	1	0	0	2	0
Total Credit				**23 Credit**				

INTRODUCTION TO DATA SCIENCE

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Introduction to Data Science	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 6th

THEORY

Learning objectives: *Apply data science techniques to real-world problems: Students should gain practical experience by working on real-world data science projects. They should be able to identify business or research problems, design and implement data science solutions, and evaluate the effectiveness of their models or algorithms.*

Prerequisite: *None.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Data Science	10	10%
Module-II: Descriptive statistics	6	20%
Module-III: Machine Learning Techniques	10	20%
Module-IV: Principles of Data Visualization	6	10%
Module-V: Handling Large Datasets	8	20%

Module-VI: Data Privacy and Security	8	20%

SYLLABUS OUTLINE:

Module-I: Introduction to Data Science [10L]

Overview of data science, Role of data scientists, Data science workflow, Programming for Data Science, Introduction to Python or R programming, Data manipulation and cleaning with pandas or dplyr, Exploratory data analysis, Statistical Analysis for Data Science

Module-II: Descriptive Statistics [6L]

Probability and distributions, Hypothesis testing and confidence intervals, Regression analysis Data Pre-processing and Feature Engineering, Data cleaning and handling missing values, Feature selection and engineering, and Dealing with data imbalances.

Module-III: Machine Learning Techniques [10L]

Supervised learning: classification and regression, Unsupervised learning: clustering and dimensionality reduction, Ensemble methods, Data Visualization

Module-IV: Principles of Data Visualization [6L]

Visualization libraries (matplotlib, ggplot, etc.), Interactive visualizations with tools like Tableau or D3.js, Introduction to Big Data.

Module-V: Handling large Datasets [8L]

Distributed computing frameworks (e.g., Hadoop, Spark), Ethical Considerations in Data Science, and Case Studies.

Module-VI: Data Privacy and Security [8L]

Bias and Fairness in data analysis, Responsible data handling practices, Application of Data Science, Case studies and real-world applications in various domains, Project work: applying data science techniques to a selected problem.

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. VanderPlas, Jake. Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media, 2016.
2. Provost, Foster, and Tom Fawcett. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media, 2013.
3. Grolemund, Garrett, and Hadley Wickham. R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. O'Reilly Media, 2016.

Reference Books:

4. McKinney, Wes. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. O'Reilly Media, 2017.
5. Deisenroth, Marc Peter, A Aldo Faisal, and Cheng Soon Ong. Mathematics for Machine Learning. Cambridge University Press, 2020.
6. Cioara, Jeremy, et al. Python Data Science Essentials. Packt Publishing, 2015.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	1	2	3	-	-	-	-	-	-	-	1
CO2	2	2	2	2	3	-	-	-	-	-	-	3
CO3	2	2	2	3	-	-	-	-	-	-	-	3
CO4	2	1	2	3	3	-	-	-	-	-	-	2
CO5	2	1	2	2	1	-	-	-	-	-	-	1
CO6	2	2	2	2	3	-	-	-	-	-	-	2
Avg	2	1.9	2	2.3	1.6	-	-	-	-	-	-	2.1

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

XXXXXX. CO1: Students should gain a solid understanding of the fundamental concepts and principles of Data Science, including data collection, cleaning, exploration, visualization, statistical analysis, machine learning, and data-driven decision-making.

XXXXXX. CO2: Students should develop proficiency in programming languages commonly used in Data Science, such as Python or R. They should be able to write code to manipulate data, perform statistical analysis, and build machine learning models.

XXXXXX. CO3: Students should acquire skills to effectively manipulate and analyze large and complex datasets. This includes skills in data pre-processing, feature engineering, data transformation, and data visualization.

XXXXXX. CO4: Students should learn various statistical analysis techniques and modeling approaches used in Data Science. This includes understanding of descriptive statistics, inferential statistics, hypothesis testing, regression analysis, time series analysis, and other statistical modeling techniques.

XXXXXX. CO5: Students should become familiar with a range of machine learning algorithms and techniques, such as linear regression, logistic regression, decision trees, random forests, support vector machines, clustering, and neural networks. They should understand the principles behind these algorithms and know how to apply them to real-world problems.

XXXXXX. CO6: Students should develop skills in visualizing and communicating data insights effectively. This includes creating meaningful visualizations, interpreting and presenting results, and effectively communicating findings to both technical and non-technical audiences.

Cryptography & Network Security/Artificial Neural Networks

Cryptography & Network Security

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Cryptography & Network Security	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 6th

<u>THEORY</u>

Learning objectives: *On completion of the course, student will be able to:*
1. To understand basics of Cryptography and Network Security.
2. To be able to secure a message over insecure channel by various means.
3. To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
4. To understand various protocols for network security to protect against the threats in the networks.

Prerequisite: *Understanding of mathematical principles, such as linear algebra, number theory, and combinatorics.*

Course content/Syllabus:

Module no.	No of	Weightage (%)

	lecture/Contact hour	
Module-I: Attacks on Computers & Computer Security	5	
Module-II: Cryptography: Concepts & Techniques	8	
Module-III: Symmetric Key Algorithm	9	
Module-IV: Asymmetric Key Algorithm, Digital Signature and RSA	9	
Module-V: Internet Security Protocols, User Authentication	9	
Module-VI: Electronic Mail Security and Firewall	8	

SYLLABUS OUTLINE:

Module-I: Attacks on Computers & Computer Security [5L]

Introduction, Need for Security, Security approaches, Principles of Security, Types of attack

Module-II: Cryptography: Concepts & Techniques [8L]

Encryption & Decryption, Symmetric & Asymmetric key Cryptography, Key Range & Key Size Introduction, Plaintext & Cipher text, Substitution Techniques, Transposition Techniques,

Module-III: Symmetric Key Algorithm [9L]

Introduction, Algorithm types & Modes, Overview of Symmetric Key Cryptography, DES(Data Encryption Standard) algorithm, IDEA(International Data Encryption Algorithm) algorithm, RC5(Rivest Cipher 5) Algorithm.

Module-IV: Asymmetric Key Algorithm, Digital Signature and RSA [9L]

Introduction, Overview of Asymmetric key Cryptography, RSA algorithm, Symmetric & Asymmetric key Cryptography together, Digital Signature, Basic concepts of Message Digest and Hash Function (Algorithms on Message Digest and Hash function not required)

Module-V: Internet Security Protocols, User Authentication [9L]

Basic Concepts, SSL protocol, Authentication Basics, Password, Authentication Token, Certificate based Authentication, Biometric Authentication.

Module-VI: Electronic Mail Security and Firewall [8L]

Basics of mail security, Pretty Good Privacy, S/MIME, Introduction to Firewall, Types of firewall, Firewall Configurations, DMZ Network

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Stallings, William. Cryptography and Network Security: Principles and Practice. 7th ed., Pearson, 2017.
2. Paar, Christof, and Jan Pelzl. Understanding Cryptography: A Textbook for Students and Practitioners. 2nd ed., Springer, 2010.
3. Kaufman, Charlie, Radia Perlman, and Mike Speciner. Network Security: Private Communication in a Public World. 2nd ed., Prentice Hall, 2002.
4. Schneier, Bruce. Applied Cryptography: Protocols, Algorithms, and Source Code in C. 2nd ed., Wiley, 1996.

Reference Books:

1. Ferguson, Niels, Bruce Schneier, and Tadayoshi Kohno. Cryptography Engineering: Design Principles and Practical Applications. Wiley, 2010.

2. Forouzan, Behrouz A., and Debdeep Mukhopadhyay. Cryptography and Network Security. McGraw-Hill Education, 2018.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	3	1								2
CO2	3	1	3	2								
CO3	3	3	3	3	2							2
CO4	3	1	3	2								
CO5	3	3	3	3	2							2
CO6	3	2	3	3	3							
Avg												

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

1XXXXX. CO1: To understand the fundamental of attacks and the need of security

1XXXXX. CO2: To be able to secure a message over insecure channel by various means.

1XXXXX. CO3: Have a strong understanding of different cryptographic algorithms and techniques and be able to use them

1XXXX. CO4: To learn about how to maintain the Confidentiality, Integrity and Availability of a data.

1XXXX. CO5: To understand various protocols for network security to protect against the threats in the networks.

1XXXXX. CO6: To apply methods for authentication, access control, intrusion detection and

prevention. Identify and mitigate software security vulnerabilities in existing systems

ARTIFICIAL NEURAL NETWORKS

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Artificial Neural Networks	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 6th

THEORY

Learning objectives: *On completion of the course, student will be able to: Learn and understand Artificial Neural Network (ANN) concepts, methods and techniques. They can evaluate various architectures of ANN and understand its usage in its various applications*

Prerequisite: *Before learning the concepts of Artificial Neural Network (ANN), you should have a basic knowledge of probability theory, linear algebra and calculus. They should also have programming skills throughout the course.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	6	10%
Module-II: Learning Process	7	15%
Module-III: Single Layer Perceptron	8	20%
Module-IV: Multilayer Perceptron	7	15%
Module-V: Back Propagation	10	20%
Module-VI: Applications of ANN	10	20%

SYLLABUS OUTLINE:

Module-I: Introduction **[6L]**

A Neural Network, Human Brain, Models of a Neuron, Neural Networks viewed as Directed Graphs, Network Architectures, Knowledge Representation, Artificial Intelligence and Neural Networks.

Module-II: Learning Process [**7L**]

Error Correction Learning, Memory Based Learning, Hebbian Learning, Competitive, Boltzmann Learning, Credit Assignment Problem, Memory, Adaption, Statistical Nature of the Learning Process, McCulloch – Pitts Neuron Model, Perceptron Learning Rule, Delta Learning Rule, Competitive Learning Rule, Hebb Net.

Module-III: Single Layer Perceptron **[8L]**

Adaptive Filtering Problem, Unconstrained Organization Techniques, Linear Least Square Filters, Least Mean Square Algorithm, Learning Curves, Learning Rate Annealing Techniques, Perceptron - Convergence Theorem, Relation Between Perceptron and Bayes Classifier for a Gaussian Environment.

Module-IV: Multilayer Perceptron [**7L**]

Back Propagation Algorithm, XOR Problem, Heuristics, Output Representation and Decision Rule, Computer Experiment, Feature Detection.

Module-V: Back Propagation [**10L**]

Back Propagation and Differentiation, Hessian Matrix, Generalization, Cross Validation, Network Pruning Techniques, Virtues and Limitations of Back Propagation Learning, Radial Basis Function Network (RBFN). Accelerated Convergence, Supervised Learning.

Module-VI: Applications of ANN **[10L]**

Pattern Classification, Associative memories, Optimization, Vector quantization, Control applications, Application in Speech and Image Processing.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Neural Networks a Comprehensive Foundations, Simon Haykin, PHI edition.

2. Artificial neural network, B. Yegnanarayana, PHI Publication.

Reference Books:

1. Neural Networks in Computer Intelligence, Li Min Fu, MC GRAW Hill Education, 2003.

2. Introduction to Artificial Neural Systems Jacek M. Zurada, JAICO Publishing House Ed., 2006.

3. Neural Networks, James A Freeman; David M S Kapura, Pearson Education, 2004

<u>CO-PO Mapping</u>

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	2	2	-	-	-	-	-	-	-	-	-	2
CO2	2	2	2	-	-	-	-	-	-	-	-	-
CO3	-	2	-	-	-	-	-	-	-	-	-	-
CO4	-	3	3	-	3	-	-	-	-	-	-	2
CO5	-	2	3	-	-	-	-	-	2	-	-	2
CO6	-	-	3	2	-	-	-	-	-	-	-	2
Avg	2	2.2	2.75	2	3	-	-	-	2	-	-	2

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

XXXXXXXX. CO1: Understand the principles of Neural Networks.

XXXXXXXX. CO2: Identify different types of models of artificial neural networks (ANN).

XXXXXXXX. CO3: Analyse the feedback and feed-forward neural networks.

XXXXXXXX. CO4: Develop neural network models.

XXXXXXXX. CO5: Compare different applications of artificial neural networks.

XXXXXXXX. CO6: Design and develop applications using neural networks.

Cloud Computing/Soft Computing

Cloud Computing

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Cloud Computing	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 6th

THEORY

Learning objectives: *On completion of the course, student will be able to:*

- *Students will learn the evolution strategy and technologies related to Cloud Computing.*
- *Students will learn the basic Cloud types and delivery models and develop an understanding of the risk and compliance responsibilities and Challenges for each Cloud type and service delivery model.*
- *The student will also learn how to apply trust-based security model to real-world security problems.*
- *An overview of the concepts, processes, and best practices needed to successfully secure information within Cloud infrastructures.*

Prerequisite: *Familiarity with Operating Systems. Understanding of Virtualization, Basics of Networking. Basic Understanding of Different Types of Cloud.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Definition of Cloud Computing, Architecture and Concept	10	
Module-II: Use of Platforms in Cloud Computing	10	
Module-III: Cloud Infrastructure	10	
Module-IV: Cloud Management and Storage	8	
Module-V: Cloud Security and Privacy	6	
Module-VI: Concepts of Services and Applications	4	

SYLLABUS OUTLINE:

Module-I: Definition of Cloud Computing, Architecture and Concept [10L]

1. Evolution of cloud computing, Distributed systems, Grid Computing, Cluster computing, Mobile computing, Definition of Cloud Computing: Defining a Cloud, Cloud Types – NIST 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

2. Cloud Architecture: Cloud Infrastructure, Architectural Framework of Cloud Infrastructure, Virtualization versus Traditional Approach.

Module-II: Use of Platforms in Cloud Computing [10L]

1. Concepts of Abstraction and Virtualization

Layered Structure and Virtualization, Mapping Technique of Virtual Machine to Physical Machine, Virtualization Model for Cloud Computing and its representation.

2. 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, Hypervisor Classification, Examples: VMware, vSphere Machine imaging (including mention of Open Virtualization Format – OVF)

Module-III: Cloud Infrastructure [10L]

1. Concepts of Platform as a Service

Definition of services, Distinction between SaaS and PaaS (knowledge of Salesforce.com and Force.com), Application development. Use of PaaS Application frameworks

2. Use of Google Web Services

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.

3. Use of Amazon Web Services

Amazon Web Service components and services: Amazon Elastic Cloud, Amazon Simple Storage system, Amazon Elastic Block Store, Amazon SimpleDB and Relational Database Service

4. Use of Microsoft Cloud Services

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-IV: Cloud Management and Storage [**8L**]

Types of services required in implementation – Consulting, Configuration, Customization and Support

1. Cloud Management

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)

2. Live Migration of Virtual Machines:

Need of Live Migration of Virtual Machine, A Designing Process of Live Migration, and Security Issues during live migration

3. Cloud Database:

Non-Relational Data Models, Heterogeneous Databases in DaaS, MongoDB, CAP Theorem, Commercial Cloud Database Platform

Module-V: Cloud Security and Privacy [**6L**]

1. Concepts of Cloud Security

Infrastructure Security: The Network Level, The Host Level, The Application Level, Data Security and Storage, Public and Private cloud Computing Security, Distributed-Denial-of-Service Attacks. Shared Cloud Computing Services, Phishing and Social Engineering Attacks System Vulnerabilities

2. Auditing and Compliance in Cloud Environment:

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-VI: Concepts of Services and Applications [4L]

1. Service Oriented Architecture: Basic concepts of message-based transactions, Protocol stack for an SOA architecture, Event-driven SOA, workflow and Co-ordination of Multiple components.

2. Applications in the Cloud: Concepts of cloud transactions, functionality mapping, Application attributes, Cloud service attributes, System abstraction and Cloud Bursting, Applications and Cloud APIs

3. Cloud-based Storage: Customer-facing data, Distributed-access data, Data backups, Sensitive data, Synchronized data, Large databases, Public and private Cloud Storage, Cloud Storage Service, Utility Storage, Storage Virtualization, Cooperative Storage Cloud

4. Integration of cloud with Wireless Sensor Network, A framework of Cloud and WSN. Different Applications in WSN in cloud infrastructure.

5. Webmail Services: Cloud mail services including Google Gmail, Mail2Web, Windows Live Hotmail, Yahoo mail, concepts of Syndication services

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. "*Cloud computing: A practical approach*", Anthony T. Velte, Tata Mcgraw-Hill
2. "*Cloud Computing: Principles and Paradigms*", Rajkumar Buyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, Print ISBN:9780470887998 |Online ISBN:9780470940105
3. "*Cloud Computing Solutions: Architecture, Data Storage, Implementation and Security*", S. Pal, Dac-Nhuong Le, P. K. Pattnaik, John Wiley & Sons Inc, 2020 [ISBN: 9781119681656]

Reference Books:

1. "*Cloud Computing Bible",* Barrie Sosinsky, Wiley India Pvt. Ltd, 2013
2. "*Building applications in cloud: Concept, Patterns and Projects*", Moyer, Pearson

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	2	1	1							2
CO2	2	2	3	3	3							
CO3	2	2	3	3	3							2
CO4	3	3	2	2	2							
CO5	2	3	2	2	2							2
CO6	1	2	3	3	3							
Avg												

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

1XXXXX. CO1: To be able to articulate the business model concepts, architecture and infrastructure of cloud computing, including cloud service models and deployment models.

1XXXXX. CO2: To be able to apply and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms.

1XXXXX. CO3: To be able to explore some important cloud computing driven commercial systems such as Google Apps, Microsoft Azure and Amazon Web Services and other businesses cloud applications.

1XXXXX. CO4: To be able to analyse the core issues of cloud computing such as security, privacy, interoperability, and its impact on cloud application.

1XXXXX. CO5: To be able to analyze the flow of service oriented architecture and protocol stack.

1XXXXX. CO6: To be able to evaluate different cloud applications in different platforms.

Soft Computing

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Soft Computing	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 6th

THEORY

Learning objectives: *On completion of the course, student will be able to: Demonstrate artificial intelligence in terms of linguistic variable concepts related to design of modern AI tools in several domain including healthcare, finance, agriculture etc. Analyse the performance of AI tools with data availability. This course is intended to teach the basics application in AI application.*

Prerequisite: *A strong mathematical background. Proficiency with algorithm set theory, mathematical logic, Programming skills python, Perl, MATLAB, etc. Critical thinking and problem-solving skills.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	3	
Module-II: Fuzzy Sets	12	
Module-III: Fuzzy Systems	12	
Module-IV: Artificial Neural Network	12	
Module-V: Genetic Algorithm	4	
Module-VI: Associated Soft Computing Techniques	5	

SYLLABUS OUTLINE:

Module-I: Introduction [3L]

Introduction to soft computing, requirement, soft computing versus hard computing, different tool and techniques and applications. Computational Intelligence versus Machine Learning Basics.

Module-II: Fuzzy Sets [12L]

Introduction, Fuzzy sets versus crisp sets, operations on Classical sets, properties of classical sets, Fuzzy set operations, properties of fuzzy sets, cardinality, operations, Fuzzy relations and properties of fuzzy relations.

Module-III: Fuzzy Systems [12L]

Membership functions: Features of membership functions, standard forms and boundaries, fuzzification, for fuzzy sets, Defuzzification methods: Lamba Cuts, Alpha cuts Fuzzy Logic, Approximate reasoning and Fuzzy Implication. Fuzzy Rule based Systems: Linguistic Hedges, Fuzzy Rule based system – Aggregation of fuzzy Rules, Fuzzy Inference System- Mamdani Fuzzy Models – Sugeno Fuzzy Models. Applications of Fuzzy Logic, fuzzy logic controllers, fuzzy pattern recognition, fuzzy image processing.

Module-IV: Artificial Neural Network [12L]

Introduction and basic models, biological neurons and artificial neural network. Learning Methods: Mc-pitt , Hebb’s learning, Perceptron, Adaline and Madaline networks, single layer network, Multilayer feed forward network, Back-propagation network, Different issue regarding convergence multilayer perceptron, Competitive learning, Self-Organizing Maps, Hopfield Networks, Associative Memories, Boltzmann Machine and applications.

Module-V: Genetic Algorithm [4L]

Introduction, different operators of GA: crossover and mutation, analysis of selection operations, Hypothesis and building block, Multi-objective Genetic Algorithm (MOGA), GA in search and optimization and applications.

Module-VI: Advanced Search Technique [5L]

Simulated Annealing, Tabu search, Ant colony optimization (ACO), Particle Swarm Optimization (PSO). Hybrid Systems: Neural Network based Fuzzy system, Fuzzy Logic based Neural Networks.

Pedagogy for Course Delivery: *Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA):*NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Jang, Jyh-Shing Roger. "Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence." Prentice Hall, 1997.
2. Bezdek, James C., and Sankar K. Pal. "Fuzzy Models for Pattern Recognition: Methods That Search for Structures in Data." IEEE Press, 1992.
3. Jain, Lakhmi C., et al. "Neuro-Fuzzy and Soft Computing: A Computational Approach." CRC Press, 2017.
4. Engelbrecht, Andries P. "Computational Intelligence: An Introduction." Wiley, 2007.
5. Gupta, Madan M. "Soft Computing and Intelligent Systems: Theory and Applications." Academic Press, 2000.

Reference Books:

1. Kosko, Bart. "Fuzzy Thinking: The New Science of Fuzzy Logic." Hyperion, 1993.
2. Pedrycz, Witold, and Fernando Gomide. "An Introduction to Fuzzy Sets: Analysis and Design." MIT Press, 1998.
3. Zimmermann, Hans-Jürgen. "Fuzzy Set Theory—and Its Applications." Kluwer Academic Publishers, 1991.
4. Bouchon-Meunier, Bernadette, et al. "Uncertainty in Knowledge Bases." Springer, 1991.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	3	2	2								
CO2	3	3	3	3	2	1						3
CO3	2	3	3	3	2	2						2
CO4	1	2	3	3	3	1						2
CO5	1	3	3	3	2	2						3
CO6	1	2	3	3	3	2						3
Avg	**1.83**	**2.67**	**2.83**	**2.83**	**2.40**	**1.60**						**2.60**

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

CO1: To Understand intelligent systems leveraging the paradigm of soft computing techniques.

CO2: To get the knowledge solutions by various soft computing approaches for finding the optimal solutions.

CO3: To Recognize the feasibility of applying a soft computing methodology for a particular problem

CO4: To Design the methodology to solve optimization problems using fuzzy logic, genetic algorithms and neural networks.

CO5: To Design hybrid system to revise the principles of soft computing in various applications

CO6: To Analyse the applications of Soft Computing Systems

NM Elective-II

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Soft-Skill Development-VI

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Mentored Seminar-II

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

SEC3: Logical Ability

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Introduction to Data Science Lab

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Introduction to Data Science Lab	**COURSE CREDIT :** 01 [0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 2nd

List of Experiments

Introduction to Data Science:
Overview of data science and its applications
Introduction to data analysis pipeline
Tools and libraries commonly used in data science (e.g., Python, R, pandas, NumPy, matplotlib, seaborn)

Data Wrangling and Preprocessing:
Data acquisition from various sources (CSV, databases, APIs, web scraping)
Data cleaning and preprocessing techniques

Handling missing data and outliers
Data transformation and normalization

Exploratory Data Analysis (EDA):
Summary statistics and data visualization
Distribution plots, scatter plots, pair plots, etc.
Correlation analysis
Feature engineering and selection

Statistical Analysis:
Probability distributions and hypothesis testing
Parametric and non-parametric tests
Regression analysis
Time series analysis

FOURTH YEAR

SEMESTER-VII

Sl No	Course Title	Code		Credit	Type			
					L	T	P	S
1	Deep Learning/Mobile Computing		ME	4	4	0	0	0
2	NM Elective III		NM	4	4	0	0	0
3	Project-I / Fundamentals of Blockchain and Applications/Data Warehousing & Data Mining		Project	4	0	0	0	4
4	Summer Internship		INT	4	0	0	0	4
Total Credit				**16 Credit**				

DEEP LEARNING/MOBILE COMPUTING

DEEP LEARNING

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Deep Learning	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 7th

THEORY

Learning objectives: *The main objective of this course is to make students comfortable with tools and techniques required in handling large amounts of datasets. They will also uncover various deep learning methods in NLP, Neural Networks etc. Several libraries and datasets publicly available will be used to illustrate the application of these algorithms. This will help students in developing skills required to gain experience of doing independent research and study*

Prerequisite: *Before learning the concepts of Deep Learning, you should have a strong knowledge of linear algebra and calculus, Machine Learning, AI. Programming knowledge in Python, TensorFlow*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	6	
Module-II: Deep Networks	10	
Module-III: Dimentionality Reduction	8	
Module-IV: Deep Learning Models	10	
Module-V: Optimization AND Generalization	10	
Module-VI: Case study and applications	4	

SYLLABUS OUTLINE:

Module-I: Introduction: [6L]

Introduction to machine learning- Linear models (SVMs and Perceptrons, logistic regression)- Intro to Neural Nets: What a shallow network computes- Training a network: loss functions, back propagation and stochastic gradient descent- Neural networks as universal function approximates

Module-II: Deep Networks: [**10L**]

A Probabilistic Theory of Deep Learning, Backpropagation and regularization, batch normalization- VC Dimension and Neural Nets- Deep Vs Shallow Networks-Convolutional Networks, Architecture- Generative Adversarial Networks (GAN)

Module-III: Dimentionality Reduction: [8L]

Linear (PCA, LDA) and manifolds, metric learning - Auto encoders and its Architecture, and dimensionality reduction in networks - Introduction to Convnet - Architectures – AlexNet, VGG, Inception, ResNet - Training a Convnet: weights initialization, batch normalization, hyper parameter optimization

Module-IV: Deep Learning Models: [10L]

Introduction to CNNs, Architecture, Convolution/pooling layers, CNN Applications, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet. Introduction to RNNs, Back propagation through time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT, GRU.

Module-V: Optimization AND Generalization: [10L]

Optimization in deep learning– Non-convex optimization for deep networks- Stochastic Optimization Generalization in neural networks- Spatial Transformer Networks- Recurrent networks, LSTM - Recurrent Neural Network Language Models- Word-Level RNNs & Deep Reinforcement Learning - Computational & Artificial Neuroscience

Module-VI: Case study and applications: [4L]

Image net- Detection-Audio Wave Net-Natural Language Processing Word2Vec - Joint Detection-Bioinformatics- Face Recognition- Scene Understanding Gathering Image Captions

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Hands-On Unsupervised learning with Python by Giuseppe Bonaccorso – Packt publication

2. Python Deep Learning by Daniel Slater, Gianmario Spacagna and Peter Roelants – Packt Publication

3. Machine Learning with Tensorflow by Nishant Shukla

4. Nikhil Buduma, “Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithm”, O’Reilly, 2017.

5. Ian Goodfellow, YoshuaBengio and Aaron Courville, “Deep Learning”, MIT Press, 2016.

Reference Books:

1. Deep Learning with Keras by Antonio Gulli and Sujit Pal
2. Machine Learning for OpenCV by Micheal Beyeler

3. Géron, Aurélien. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow." 2nd ed., O'Reilly Media, 2019.

4. Ramsundar, Bharath, and Reza Bosagh Zadeh. "TensorFlow for Deep Learning: From Linear Regression to Reinforcement Learning." O'Reilly Media, 2018.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	2							2
CO2	3	3	3	2	3							2
CO3	3	3	3	3	2							2
CO4	3	2	3	3	3							1
CO5	2	2	3	3	3	1						1
CO6	1	3	3	3	3	2						3
Avg	**2.50**	**2.67**	**2.83**	**2.67**	**2.67**	**1.50**						**1.83**

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

1XXXXX. CO1: To be able to understand and apply the mathematical, statistical and computational challenges of building neural networks. (Understand, Apply)

1XXXXX. CO2: To be able to understand, apply the concepts of deep learning. (Apply)

1XXXXX. CO3: To be able to compare different dimensional reduction techniques in deep learning framework. (Evaluate)

1XXXXX. CO4: To be able to implement architectures and optimization methods for deep neural network training. (Apply)

1XXXXX. CO5: To be able to evaluate optimization techniques for different neural networks. (Evaluate)

1XXXXX. CO6: To be able to establish relevant learning algorithms for different applications. (Create)

MOBILE COMPUTING

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Mobile Computing	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 7th

THEORY

Learning objectives: *On completion of the course, student will be able to:*
1. To understand the basic concepts of mobile computing.
2. To learn the basics of mobile telecommunication system .
3. To be familiar with the network layer protocols and Ad-Hoc networks.
4. To know the basis of transport and application layer protocols.
5. To gain knowledge about different mobile platforms and application development.

Prerequisite: *Before learning the concepts of Mobile Computing, you should have a basic Basic knowledge of Data Communication Networks.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Mobile Computing	6	
Module-II: Mobile Telecommunication System	10	
Module-III: Mobile Network Layer	10	
Module-IV: Mobile Transport and Application Layer	6	
Module-V: Cognitive Radio Networks	8	
Module-VI: Mobile Platforms and Applications	8	

SYLLABUS OUTLINE:

Module-I: Introduction to Mobile Computing: [6L]

Introduction to Mobile Computing, Applications of Mobile Computing- Generations of Mobile Communication Technologies- Multiplexing, Spread spectrum -MAC Protocols, SDMA- TDMA- FDMA- CDMA. Concept of location management (HLR and VLR), Handoff strategies; Different types of handoffs (soft, hard, horizontal, vertical).

Module-II: Mobile Telecommunication System: [10L]

Introduction to Cellular Systems, GSM, Services & Architecture, Protocols, Connection Establishment, Frequency Allocation, Routing, Mobility Management, Security,GPRS Architecture, GPRS Network Nodes. Mobile Data Communication, UMTS, Architecture, Handover, Security.

Module-III: Mobile Network Layer : [10L]

Mobile IP, DHCP, AdHoc– Proactive protocol-DSDV, Reactive Routing Protocols, DSR, AODV , Hybrid routing –ZRP, Multicast Routing- ODMRP, Vehicular Ad Hoc networks (VANET) –MANET Vs VANET, Security.

Module-IV: Mobile Transport and Application Layers: [6L]

Mobile TCP– WAP, Architecture, WDP, WTLS, WTP , WSP, WAE, WTA Architecture, WML.

Module-V: Cognitive Radio Networks: [**8L**]

Fixed and dynamic spectrum access; Direct and indirect spectrum sensing; Spectrum sharing; Interoperability and co-existence issues; Applications of cognitive radio networks.

Module-VI: Mobile Platforms and Applications : [8L]

Mobile Device Operating Systems, Special Constraints & Requirements, Commercial Mobile Operating Systems, Software Development Kit: iOS, Android, BlackBerry, Windows Phone, M-Commerce, Structure, Pros & Cons, Mobile Payment System, Security Issues.

***Pedagogy for Course Delivery**: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

***List of Professional Skill Development Activities (PSDA)**:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Talukdar, Soubhik, and Akhilesh Mohan. "Mobile Computing: Concepts, Methodologies, Tools, and Applications." IGI Global, 2008.
2. Li, Mo. "Mobile Computing: Technology, Applications, and Service Creation." McGraw-Hill Professional, 2006.
3. Prasant Kumar Pattnaik, Rajib Mall, Fundamentals of Mobile Computing 2nd Ed, PHI, 2015

Reference Books:

1. Asoke K. Talukder, Hasan Ahmed, and Roopa R. Yavagal. "Mobile Computing: Technology, Applications, and Service Creation." McGraw-Hill Education, 2007.
2. Sarma, Debasis, and Kuan Yew Wong. "Mobile Computing: Implementing Pervasive Information and Communications Technologies." Elsevier, 2007.

CO-PO Mapping: **Avg has to be rechecked**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	1								1
CO2	3	2	3	1	1							1
CO3	3	3	3	2	1							2
CO4	3	2	3	3	1							2
CO5	3	3	1	2	2	1						2
CO6	3	1	3	2	3	1						2
Avg												

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: *(CO)*

1XXXXX. CO1: To be able to understand the basics of mobile telecommunication systems. (Understand)

1XXXXX. CO2: To be able to illustrate the GPRS systems in wireless networks. (Apply)

1XXXXX. CO3: To be able to determine the functionality of MAC, network layer and protocols. (Apply)

1XXXXX. CO4: To be able to explain the functionality of Transport and Application layers. (Analyze)

1XXXXX. CO5: To be able to evaluate the effectiveness of different mobile computing frameworks. (Evaluate)

1XXXXX. CO6: To be able to Develop a mobile application using android/blackberry/ios/Windows SDK. (Create)

NM Elective III

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Project-I / Fundamentals of Blockchain and Applications/Data Warehousing & Data Mining

Project-I

Depend on the Supervisor

Fundamentals of Blockchain and Applications

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Fundamentals of Blockchain and Applications	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 7th

DATA WAREHOUSING & DATA MINING

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Data Warehousing & Data Mining	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 7th

THEORY

Learning objectives: *On completion of the course, student will be able to:*

1. *To understand the principles of Data warehousing and Data Mining.*
2. *To be familiar with the Data warehouse architecture and its Implementation.*
3. *To know the Architecture of a Data Mining system.*
4. *To understand the various Data preprocessing Methods.*
5. *To perform classification and prediction of data.*

Prerequisite: *Before learning the concepts of Mobile Computing, you should have a basic Basic knowledge of Data Communication Networks.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Data Warehousing and Business Analysis	8	
Module-II: Data Mining	10	
Module-III: Classification and Prediction	10	
Module-IV: Cluster Analysis	10	
Module-V: Data Mining cases	10	

SYLLABUS OUTLINE:

Module-I: Data Warehousing and Business Analysis [8L]

Data Warehousing and Business Analysis: - Data warehousing Components –Building a Data warehouse –Data Warehouse Architecture – DBMS Schemas for Decision Support – Data Extraction, Clean-up, and Transformation Tools –Metadata – reporting – Query tools and Applications – Online Analytical Processing (OLAP) – OLAP and Multidimensional Data Analysis.

Module-II: Data Mining [10L]

Data Mining: - Data Mining Functionalities – Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation- Architecture Of A Typical Data Mining Systems- Classification Of Data Mining Systems.

Association Rule Mining: - Efficient and Scalable Frequent Item set Mining Methods – Mining Various Kinds of Association Rules – Association Mining to Correlation Analysis – Constraint-Based Association Mining.

Module-III: Classification and Prediction [10L]

Classification and Prediction: - Issues Regarding Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section.

Module-IV: Cluster Analysis [10L]

Cluster Analysis: - Types of Data in Cluster Analysis – A Categorization of Major Clustering Methods – Partitioning Methods – Hierarchical methods – Density-Based Methods – Grid-Based Methods – Model-Based Clustering Methods – Clustering High-Dimensional Data – Constraint-Based Cluster Analysis – Outlier Analysis.

Module-V: Data Mining cases [10L]

Mining Object, Spatial, Multimedia, Text and Web Data:

Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Mining the World Wide Web.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Jiawei Han, Micheline Kamber and Jian Pei“Data Mining Concepts and Techniques”, Third Edition, Elsevier, 2011.

Reference Books:

1. Alex Berson and Stephen J. Smith “Data Warehousing, Data Mining & OLAP”, Tata McGraw – Hill Edition, Tenth Reprint 2007.
2. K.P. Soman, Shyam Diwakar and V. Ajay “Insight into Data mining Theory and Practice”, Easter Economy Edition, Prentice Hall of India, 2006.
3. G. K. Gupta “Introduction to Data Mining with Case Studies”, Easter Economy Edition, Prentice Hall of India, 2006.
4. Pang-Ning Tan, Michael Steinbach and Vipin Kumar “Introduction to Data Mining”, Pearson Education, 2007.

CO-PO Mapping: Avg has to be rechecked

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg												

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome: (CO)

XXXX. CO.1: Understand the functionality of the various data mining and data warehousing component Knowledge. (Understand)

XXXX. CO.2: Appreciate the strengths and limitations of various data mining and data warehousing models Apply. (Create)

XXXX. CO.3 Explain the analyzing techniques of various data. (Analyze)

XXXX. CO.4: Describe different methodologies used in data mining and data ware housing. (Analyze)

XXXX. CO.5: Compare different approaches of data ware housing and data mining with various technologies. (Evaluating)

Summer Internship

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Fourth Year

SEMESTER-VIII

Sl No	Course Title	Code		Credit	Type			
					L	T	P	S
1	NM Elective-IV		NM	4	4	0	0	0
2	Project-II / Distributed Systems/Introduction to Cognitive Science		Project	4	0	0	0	4
3	Project-II / Natural Language Processing/Introduction to Augmented Reality & Virtual Reality		Project	4	0	0	0	4
Total Credit				**12 Credit**				

NM Elective-IV

Provided from the Respective Department / School

Project-II / Distributed Systems/Introduction to Cognitive Science

Project-II

Depend on the Supervisor

Distributed Systems

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Distributed Systems	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 8th

THEORY

Learning objectives: *To introduce the fundamental concepts and issues of managing large volume of shared data in a parallel and distributed environment, and to provide insight into related research problems.*

Prerequisite: *Basic knowledge of Database Management Systems.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	6	
Module-II: Distributed Database Design	10	
Module-III: Distributed Query Optimization	8	
Module-IV: Reliability issues in DDBS	10	
Module-V: Parallel Database Systems	8	
Module-VI: Advanced Topics Mobile	6	

SYLLABUS OUTLINE:

Module-I: Introduction [6L]

Distributed data processing; What is a DDBS; Advantages and disadvantages of DDBS; Problem areas; Overview of database and computer network concepts DISTRIBUTED DATABASE MANAGEMENT SYSTEM ARCHITECTURE Transparencies in a distributed DBMS; Distributed DBMS architecture; Global directory issues

Module-II: Distributed Database Design [10L]

Alternative design strategies; Distributed design issues; Fragmentation; Data allocation

SEMANTICS DATA CONTROL: View management; Data security; Semantic Integrity Control,

QUERY PROCESSING ISSUES: Objectives of query processing; Characterization of query processors; Layers of query processing; Query decomposition; Localization of distributed data**.**

Module-III: Distributed Query Optimization [8L]

Factors governing query optimization; Centralized query optimization; Ordering of fragment queries; Distributed query optimization algorithms

TRANSACTION MANAGEMENT: The transaction concept; Goals of transaction management; Characteristics of transactions; Taxonomy of transaction models

CONCURRENCY CONTROL: Concurrency control in centralized database systems; Concurrency control in DDBSs; Distributed concurrency control algorithms; Deadlock management**.**

Module-IV: Reliability issues in DDBS [**10L**]

Types of failures; Reliability techniques; Commit protocols; Recovery protocols Algorithm

Module-V: Parallel Database Systems [8L]

Parallel architectures; parallel query processing

Module-V: Advanced Topics Mobile [6L]

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

Text & Reference books:

Text Books:

1. Principles of Distributed Database Systems, M.T. Ozsu and PValduriez, Prentice-Hall, 1991.

2. Distributed Database Systems, D. Bell and J. Grimson, AddisonWesley, 1992.

Reference books:

3. Thinking In Systems, Donella H. Meadows, Diana Wright
4. Designing Data Intensive Applications, Martin Kleppmann

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

CO-PO Mapping

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

CO3	2	1	2		1		3					1
CO4	2	2										1
CO5	2	1			1							
CO6	2	2	3	2	1							2
Avg	1.5	1.6	1.6	0.6	0.8	0.5	1					1.16

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*:*

XXXXXX**. CO1.** Define the characterization of Distributed Systems, Theoretical Foundation for Distributed System and Concepts in Message Passing Systems.

XXXXXX**. CO2.** Explain the Distributed Mutual Exclusion and Distributed Deadlock Detection.

XXXXXX**. CO3.** Apply the Agreement Protocols and Distributed Resource Management.

XXXXXX**. CO4.** Analyse the Failure Recovery in Distributed Systems and Fault Tolerance.

XXXXXX**. CO5.** Evaluate the Transactions and Concurrency Control, Distributed Transactions and Replication.

XXXXXX**. CO6**. Design the parallel database systems.

Introduction to Cognitive Science

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Introduction to Cognitive Science	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER: 8th**

THEORY

Learning objectives: *Students should develop a solid understanding of the theoretical and conceptual foundations of Cognitive Science. This includes exploring the historical development of Cognitive Science as a field, understanding the interdisciplinary nature of the field, and gaining knowledge of key theories and models of cognition.*
Prerequisite: *None.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to AI	10	10%
Module-II: Introduction to Linguistics	6	20%
Module-III: Visual Cognition	10	20%
Module-IV: Culture and Cognition	6	10%
Module-V: Judgement and Decision Making	8	20%
Module-VI: Cognitive disorders	8	20%

SYLLABUS OUTLINE:

Module-I: Introduction to AI [10L]

Introduction, Intelligent Control, Expert System, Adaptive Fuzzy Inference System, Real-time System, A Practical Approach to Neural Network Model, network Topology, Feedforward Network, Feedback Network, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Human Activity Recognition (HAR), Prediction & Analysis using Machine Learning.

Module-II: Introduction to Linguistics [6L]

Overview of the field of modern linguistics and basic skills in linguistic analysis, language learning, and change, Human activities, and linguistics contribute to many other fields of inquiry, including anthropology, psychology, philosophy, law and the natural sciences.

Module-III: Visual Cognition [10L]

Image-capturing methods, perceptual organization, depth and categorization, and contemporary research on vision to give an overview of cognitive processes in general. Furthermore, the course deals with visual perceptual learning, attention and gaze control, and Mathematical Methods for Cognitive Science, regression analysis, Principal Component Analysis, basics of probability and statistics, hypothesis testing, bootstrapping, estimation and decision theory, classification, clustering, time series analysis, information theory.

Module-IV: Culture and Cognition [6L]

relationship between human culture and human cognitive capabilities, Cultural learning allows humans to build on existing knowledge and make collective advancements, Learning and Memory, learn (encode), store, and retrieve (remember).

Module-V: Judgement and Decision Making [8L]

Basic models and strategies of decision-making and look at applications of these models in a variety of fields, including consumer choice, medicine, law and many others, systematic flaws observed in people's actual decisions, the uniquely psychological factors that influence decision-making (e.g., emotion), and the neural systems that underlie the decisions of both humans and non-human animals.

Module-VI: Cognitive Disorder [8L]

Understand different categories of mental health disorders that primarily affect learning, memory, perception, and problem solving, and include amnesia, dementia, and delirium.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Pijush Dutta, Souvik Pal, Asok Kumar, Korhan Cengiz, "Artificial Intelligence for Cognitive Modeling: Theory and Practice", CRC press, 2023, ISBN 9781032105703
2. "Cognitive Science: An Introduction to the Study of Mind" by Jay D. Friedenberg and Gordon W. Silverman
3. "Cognitive Science: A Philosophical Introduction" by Jean-Pierre Dupuy
4. "Cognitive Science: An Introduction to the Science of the Mind" by José Luis Bermúdez
5. "Cognitive Science: An Introduction" by Neil A. Stillings, Steven E. Weisler, Christopher H. Chase, et al.
6. "Cognitive Psychology: A Student's Handbook" by Michael W. Eysenck and Mark T. Keane
7. "Cognitive Science: An Introduction to Mind and Brain" by Daniel Kolak and William Hirstein

Reference Books:

1. "Cognitive Neuroscience: The Biology of the Mind" by Michael S. Gazzaniga, Richard B. Ivry, and George R. Mangun
2. "Cognitive Psychology and Cognitive Neuroscience" by Michael D. Rugg and Michael G. H. Coles
3. "The Oxford Handbook of Cognitive Science" edited by Susan E. F. Chipman, Julian F. Linnell, and Robert W. Lurz
4. "Cognitive Science: Foundations and Applications" edited by Jean-Pierre Thibaut and Martin H. Fischer

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	1	2	1	-	-	1	-	1	-	-	1
CO2	-	2	2	2	3	-	-	1	1	-	-	-
CO3	1	2	1	1	-	-	1	-	-	-	-	1
CO4	1	1	1	-	3	-	-	1	-	-	-	2
CO5	1	1	2	2	1	-	1	1	1	-	-	1
CO6	2	2	1	2	3	-	-	1	1	-	-	1

Avg	1.6	1.9	2	1.8	1.6	-	0.5	0.6	0.6	-	-	1.2

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

XXXX. CO1: Understand the interdisciplinary nature of cognitive science: Students should gain an appreciation for the multidisciplinary nature of cognitive science, which draws from fields such as psychology, neuroscience, linguistics, philosophy, computer science, and anthropology.

XXXX. CO2: Comprehend basic concepts and theories: Students should acquire a solid understanding of fundamental concepts and theories in cognitive science, such as perception, attention, memory, learning, language processing, decision-making, problem-solving, and consciousness.

XXXX. CO3: Analyze and evaluate research methodologies: Students should develop critical thinking skills and be able to analyze and evaluate research methodologies used in cognitive science, including experimental design, data collection techniques, and statistical analysis.

XXXX. CO4: Apply cognitive science principles to real-world problems: Students should be able to apply cognitive science principles to real-world scenarios, such as human-computer interaction, education, language acquisition, artificial intelligence, and cognitive disorders.

XXXX. CO5: Demonstrate knowledge of cognitive neuroscience: Students should have a basic understanding of cognitive neuroscience, including brain anatomy and function, neural correlates of cognitive processes, and the use of neuroimaging techniques in cognitive research.

XXXX. CO6: Communicate effectively about cognitive science: Students should be able to articulate and communicate concepts, theories, and research findings in cognitive science through oral presentations, written reports, and class discussions.

Project-II / Natural Language Processing/Introduction to Augmented Reality & Virtual Reality

Project-II

Depend on the Supervisor

Natural Language Processing

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Natural Language Processing	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER: 8**th

THEORY

Learning objectives: *On completion of the course, student will be able to: Extract information from text automatically using concepts and methods from natural language processing (NLP). Develop speech-based applications that use speech analysis (phonetics, speech recognition, and synthesis) and can analyze the syntax, semantics, and pragmatics of a statement written in a natural language.*

Prerequisite: *Before learning the concepts of Natural Language Processing, you should have a basic knowledge prior to Design and Analysis of Algorithms, Formal Language and Automata, Compiler Design etc.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to NLP	6	13
Module-II: Word Level and Syntactic Analysis	6	13
Module-III: Extracting Relations from Text	8	17
Module-IV: Automatic Document Separation	10	21
Module-V: Parsing	8	17
Module-VI: Applications of NLP	10	21

SYLLABUS OUTLINE:

Module-I: Introduction to NLP [6L]

Overview: Origins and challenges of NLP-Language and Grammar-Processing Indian Languages- NLP Applications-Information Retrieval. Language Modeling: Various Grammar- based Language Models-Statistical Language Model.

Module-II: Word Level and Syntactic Analysis [6L]

Word Level Analysis: Regular Expressions-Finite State Automata-Morphological Parsing-Spelling Error Detection and correction-Words and Word classes-Part-of Speech Tagging. Syntactic Analysis: Context-free Grammar Constituency- Parsing-Probabilistic Parsing.

Module-III: Extracting Relations from Text: [8L]

Domain Knowledge and Knowledge Roles, Frame Semantics and Semantic Role Labeling, Learning to Annotate Cases with Knowledge Roles and Evaluations.

Module-IV: Automatic Document Separation: [**10L**]

Data Preparation, Document Separation as a Sequence Mapping Problem, Results.

Module-V: Parsing: [**8L**]

Parsing, probabilistic parsing. Meaning representation, semantic analysis, lexical semantics, Word Sense Disambiguation, machine learning approaches, dictionary based approaches.

Module-VI: Applications of NLP: [10L]

Applications of NLP: Spell-checking, Text Summarization, Information Retrieval, Machine Translation.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*
Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Tanveer Siddiqui, U.S. Tiwary, “Natural Language Processing and Information Retrieval”, Oxford University Press, 2008.

2. Anne Kao and Stephen R. Poteet (Eds), “Natural LanguageProcessing and Text Mining”, Springer-Verlag London Limited 2007.

Reference Books:

1. Daniel Jurafsky and James H Martin, “Speech and Language Processing: Anintroduction to Natural Language Processing, Computational Linguistics and SpeechRecognition”, 2nd Edition, Prentice Hall, 2008.
2. James Allen, “Natural Language Understanding”, 2nd edition, Benjamin/Cummingspublishing company, 1995.

3. Gerald J. Kowalski and Mark.T. Maybury, “Information Storage and Retrieval systems”, Kluwer academic Publishers, 2000.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	1
CO2	-	2	-	-	-	-	-	-	-	-	-	
CO3	2	2	-	-	-	-	-	-	-	-	-	1
CO4	-	3	3	2	-	-	-	-	-	-	-	2
CO5	1	-	3	3	3	-	-	-	-	-	-	1
CO6	1	-	3	2	2	1	-	-	-	-	-	1
Avg	**1.75**	**2.25**	**3**	**2.33**	**2.5**	**1**	**-**	**-**	**-**	**-**	**-**	**1.5**

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

***Course learning outcome**: (CO)*

1XXXXX. CO1: To understand the fundamental concepts and techniques of natural language processing. (BT2)

1XXXXX. CO2: To distinguish among the various techniques, taking into account the assumptions, strengths, and weaknesses of each. (BT2)

1XXXXX. CO3: To understand appropriate descriptions, visualizations, and statistics to communicate the problems and their solutions. (BT2)

1XXXXX. CO4: Analyze large volume text data generated from a range of real-world applications. Analyze large volume text data generated from a range of real-world applications. (BT4)

1XXXXX. CO5: Apply machine learning algorithms to natural language processing. (BT5)

1XXXXX. CO6: Develop speech-based applications that use speech analysis (phonetics, speech recognition, and synthesis). (BT6)

Introduction to Augmented Reality & Virtual Reality

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Introduction to Augmented Reality & Virtual Reality	**COURSE CREDIT :** 04 [4-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER: 8**th

THEORY

Learning objectives: *The primary objective is to develop a comprehensive understanding of Augmented Reality and Virtual Reality technology, including its principles, components, and applications. Students should be able to explain the fundamental concepts and working principles of AR, VR as well as differentiate them from other related technologies like virtual reality.*

Prerequisite: *Before learning the concepts of AR and VR, you should have a basic knowledge prior to Audio video and multimedia basics.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Virtual Reality	6	15%
Module-II: Representing Virtual World	10	20%
Module-III: The Geometry of Virtual Worlds and Human Vision	6	15%
Module-IV: Visual Perception and Rendering	10	15%
Module-V: Motion Tracking	8	15%
Module-VI: Interaction & Audio	8	20%

SYLLABUS OUTLINE:

Module-I: Introduction to Virtual Reality [6L]

Defining Virtual Reality, History of VR, Human Physiology and Perception, Key Elements of Virtual Reality Experience, Virtual Reality System, Interface to the Virtual World-Input & output- Visual, Aural & Haptic Displays, Applications of Virtual Reality.

Module-II: Representing Virtual World [10L]

Representation of the Virtual World, Visual Representation in VR, Aural Representation in VR and Haptic Representation in VR.

Module-III: The Geometry of Virtual Worlds and Human Vision: [6L]

Geometric Models, Changing Position and Orientation, Axis-Angle Representations of Rotation, Viewing Transformations, Chaining the Transformations, Human Eye, eye movements & implications for VR.

Module-IV: Visual Perception and Rendering: [10L]

Visual Perception - Perception of Depth, Perception of Motion, Perception of Colour, Combining Sources of Information Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates.

Module-V: Motion Tracking: [8L]

Motion in Real and Virtual Worlds- Velocities and Accelerations, The Vestibular System, Physics in the Virtual World, Mismatched Motion and Vection Tracking- Tracking 2D & 3D Orientation, Tracking Position and Orientation, Tracking Attached Bodies.

Module-VI: Interaction & Audio: [8L]

Interaction - Motor Programs and Remapping, Locomotion, Manipulation, Social Interaction. Audio - The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering.

Pedagogy for Course Delivery*: Hybrid Mode (Offline Class/Presentation/Video/MOODLE/NPTEL)*

List of Professional Skill Development Activities (PSDA)*:NA*

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Continuous assessment: Quiz/assessment/presentation/problem solving etc.

Text & Reference books:

Text Books:

1. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016
2. Understanding Virtual Reality: Interface, Application and Design, William R Sherman and Alan B Craig, (The Morgan Kaufmann Series in Computer Graphics)". Morgan Kaufmann Publishers, San Francisco, CA, 2002
3. Developing Virtual Reality Applications: Foundations of Effective Design, Alan B Craig, William R Sherman and Jeffrey D Will, Morgan Kaufmann, 2009.

Reference Books:

1. Gerard Jounghyun Kim, "Designing Virtual Systems: The Structured Approach", 2005.
2. Doug A Bowman, Ernest Kuijff, Joseph J LaViola, Jr and Ivan Poupyrev, "3D User Interfaces,
Theory and Practice", Addison Wesley, USA, 2005.
3. Oliver Bimber and Ramesh Raskar, "Spatial Augmented Reality: Meging Real and Virtual Worlds", 2005.
4. Burdea, Grigore C and Philippe Coiffet, "Virtual Reality Technology", Wiley Interscience, India, 2003.

CO-PO Mapping

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

CO5	1	1	3	3	1	-	-	-	1	-	-	1
CO6	2	1	1	2	3	-	-	-	1	-	-	1
Avg	1.6	1.9	2	1.8	1.6	-	-	-	0.6	-	-	1.2

Highly Correlated: **3**
Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXXX. CO1: Will be able to explain the basics of Augmented Reality and Virtual Reality.

1XXXXXX. CO2: Define different representations of Virtual World Haptics with this representation.

1XXXXXX. CO3: Analyse some of the design issues in terms of Changing Position and Orientation, Axis-Angle Representations of Rotation, and Viewing Transformations.

1XXXXXX. CO4: Visual Rendering -Ray Tracing and Shading Models, Rasterization.

1XXXXXX. CO5: Tracking- Tracking 2D & 3D Orientation, Tracking Position and Orientation, Tracking Attached Bodies.

1XXXXXX. CO6: Motor Programs and Remapping, Locomotion, Manipulation, Social Interaction. Audio -The Physics of Sound