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	1
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	1
10	Digital Electronics Lab		MC	1	0	0	1
Total Credit				**L+P=(18+4)=22 Credit**			

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	Signal 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	1
10	Computer Organization Lab		MC	1	0	0	1
11	Signals and Systems Lab		MC	1	0	0	1
Total Credit				**L+P=(22+3)=25 Credit**			

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-I		NV	1	0	0	1
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	1
11	Computer Architecture Lab		MC	1	0	0	1
12	Object Oriented Programming Lab		MC	1	0	0	2
Total Credit				**L+P=(20+5)=25 Credit**			

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	1
10	Database Management System Lab		MC	1	0	0	1
11	Artificial Intelligence Lab		MC	1	0	0	1
Total Credit				**L+P=(22+3)=25 Credit**			

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/IoT		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	1	0
9	Software Engineering Lab		MC	1	0	0	1	0
10	Digital Image Processing Lab /Machine Learning Lab /IoT Lab		ME	1	0	0	1	0
Total Credit				**L+P+S=(19+3+2)=24 Credit**				

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/Embedded Systems		ME	4	4	0	0	0
3	Cloud Computing/Soft Computing/Wireless Sensor Network		ME	4	4	0	0	0
4	NM Elective-II		NM	4	4	0	0	0
5	Soft-Skill Development-V		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	1	0
Total Credit				**L+P+S=(20+1+2)=23 Credit**				

FOURTH YEAR

SEMESTER-VII

Sl No	Course Title	Code		Credit	Type			
					L	T	P	S
1	Cyber Security/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				**L+P+S=(8+0+8)=16 Credit**				

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				**L+P+S=(4+0+8)=12 Credit**				

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	1
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	1
10	Digital Electronics Lab		MC	1	0	0	1
Total Credit				**L+P=(18+4)=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	8%
Module-II: Abstract Algebra	6	12%
Module-III: Combinatories	6	13%
Module-IV: Fundamental concepts of Graph Theory	10	21%
Module-V: Tree and Network flow	10	21%

Module VI: Logic	12	25%

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: [10L]

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: [10L]

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: [12L]

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**]

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.

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. E. Balagurusamy Programming in ANSI C, 5th Edition, Tata McGraw-Hill Publications

2. Byron S Gottfried "Programming with C" Second edition, Tata McGrawhill, 2007 (Paper back)

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 3. Yashwant Kanetkar Let us C, 6th Edition , BPB publication

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	8	
Module-II: Boolean Algebra	8	
Module-III: Logic Families	4	
Module-IV: Combinational Logic	8	
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 for Computer Science
PENDING

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 [0-0-1]
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.*

C PROGRAMMING LAB EXPERIMENTS

****

****

Write C programs to perform the following:

(SIMPLE ARITHMETIC PROGRAMS)

1. Add two numbers.
2. Find out the area and perimeter of a rectangle.
3. Input three decimal numbers and find their sum and average.
4. Input two numbers and swap them –
   a. Using a third variable
   b. Without using a third variable
5. Input temperature in Celsius and convert it to Fahrenheit.

(IF-ELSE & IF-ELSE-IF LADDER)

6. Write a C program to input two numbers and find maximum between two numbers using conditional/ternary operator
7. Input a character and change its case. (lowercase to uppercase and vice versa).
8. Input a number and check whether it is odd or even and display accordingly.
9. Find the largest and smallest among three numbers supplied by user.
10. Check whether an input year is a leap year or not.

(OPERATORS & SWITCH-CASE)

11. Input two numbers and swap them using bitwise operators.
12. Convert an unsigned integer to its equivalent binary using bitwise operator.
13. Find the maximum of three numbers using ternary operator.
14. Implement simple arithmetic calculator using switch case.
15. Input a character from the user and check if it is a vowel or a consonant.

(LOOP CONTROL STRUCTURE)

16. Print the following patterns up to n no. of lines:

```
1
1 2 3
1 2 3 4 5
1 2 3 4 5 6 7
```

```
*
* *
* * *
* * * *
```

```
    *
   * * *
  * * * * *
* * * * * * *
   * * *
    *
```

17. Input two numbers and find their HCF and LCM.
18. Input a number and find the sum of its digits using while/do-while loop.
19. Input a number and check if it is a Palindrome number or not while/do-while loop.
20. Input a number and check if it is a prime number or not.
21. Input a number and check whether it is an Armstrong number or not using while/do-while loop.

(FUNCTIONS)

22. Input a number n and find its factorial using a user defined function long int fact(int)
23. Input a number and check if it a Krishnamurthy number.
24. Find the sum of first n prime numbers using as user defined function to check for prime. Input the value of n fromthe user.
25. Input a limit n and print all prime fibonacci numbers up to n using a user defined function int prime(int) whichreturns a 1 if the argument is a prime or else 0.
26. Input a limit n and print all twin prime numbers up to n.

(POINTERS & CALL BY ADDRESS)

27. Swap two numbers using call by address.
28. Find the sum and product of two numbers using call by address.
29. Input two sides of a rectangle and find their area and perimeter using call by address.
30. Find the lcm and hcf of two numbers using call by address.

(ARRAYS)

31. Write a C program to declare, initialize, input elements in array and print array.
32. Write a C program to find sum of all elements in an array
33. Write a C program to find maximum and Minimum element in an array
34. Write a C program to input elements in array and count negative elements in array.

35. Input a number and find its 2’s complement.

36. Input a decimal number and convert it into its equivalent binary.
37. Input a decimal number and convert it into its equivalent octal.
38. Input a decimal number and convert it into its equivalent hexadecimal.
39. Input a binary number and convert it into its equivalent decimal.
40. Perform any base to any base conversion.
41. Input an array of n elements remove all duplicate elements and print the new array.
42. Input an array of n elements in sorted order and perform binary search on it.
43. Write a C program to read elements in a matrix and find the sum of elements of each row and columns of matrix. C program to calculate sum of rows and columns of matrix.
44. Input a matrix of size (m x n), transpose it and print the final transposed matrix.
45. Input two matrices of any order and add them to produce a third matrix.
46. Input two matrices of any order and multiply them.
47. Input a matrix of size (m x n) and check whether it is a sparse matrix or not. A sparse matrix is a matrix where thenumber of zero elements is greater than the number of non-zero elements.

(POINTER TO ARRAYS, ARRAY OF POINTERS & DYNAMIC ARRAYS)

48. Find the sum of the elements of an array using an user defined function int sum(int*,int). The function should acceptthe array and the number of elements as arguments.
49. Sort an array of n elements using an user defined function int selection(int*,int), employing selection sort technique.The function should accept the array and the number of elements as arguments.
50. Create a dynamic 1D array to store n elements and perform binary search on them.
51. Create a dynamic 2D array to store (m x n) elements and find its upper triangular matrix.

(STRINGS)

52. Implement strlen(), strcat(), strrev(), strcpy(), strcmp(), strcmpi() without using standard library functions.
53. Enter a sentence and find number of vowels, consonants, spaces and special characters.
54. Input a string that contains digits as well as characters. Find the sum of the digits.
55. Input a string and find sum of the ASCII values of all characters.
56. Input a string and check if it a palindrome or not.
57. Input a string and count the number of words in it.
58. Input a name and find its initial (e.g., Subhash Chandra Bose should be printed as S. C. B).
59. Input a string and delete all consecutive occurrences of characters.

(STRUCTURE, UNION, ENUM)

60. Write a C program to design a structure named XYZ with one integer and one float member. Declare 3 variablesV1, V2 and V3 of this structure. Input the values of members V1 and V2. Finally, add members of V1 and V2 and keep the results in V3 and print it.

61. A student of honours class can have following attributes: roll (int), age (int), name (char []), sex (char), marks (int). Write a program to input 'n' student records with the above attributes and find out how many of them are eligible tovote. Also determine the 1st boy or girl (mention Mr. or Miss).

62. An angle is measured in degree and minutes. Write a program to specify a structure angle to measure an angle in degree and minutes. Using a function angle sumangle (angle, angle), find the resultant angle after adding the two angles and display the result with the help of the function void display (angle).

63. Accept two complex numbers and display their sum, difference and the modulus of the result after addition and subtraction using the following specifications:

```
struct complex
 {
float real, img;
 };
complex sum (complex,
complex);
complex difference
(complex, complex);
void display modulus
(complex);
```

RECURSION

64. Print the sum of natural numbers up to n using recursion.
65. Find the factorial of a number using recursion.
66. Find the nth Fibonacci number using recursion. The value of n should be taken as input.
67. Input two numbers and find their GCD using recursion.

(FILE HANDLING)

68. Input some numbers into a file and read them.
69. Input some characters from the user and write it in a file "one.txt" and read the characters from the file.
70. Copy one file to another.
71. Write a program that displays the program itself.

DIGITAL ELECTRONICS LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Digital Electronics Lab	**COURSE CREDIT :** 01 [0-0-1]
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	1
10	Computer Organization Lab		MC	1	0	0	1
11	Signals and Systems Lab		MC	1	0	0	1
Total Credit				**L+P=(22+3)=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	10L	21%
Module V: Inner Product Space	14L	30%
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 [10L]

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 [14L]

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 and 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:** MC
CODE: XXXXXX	**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 [0-0-1]
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 [0-0-1]
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-I		NV	1	0	0	1
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	1
11	Computer Architecture Lab		MC	1	0	0	1
12	Object Oriented Programming Lab		MC	2	0	0	2
Total Credit				**L+P=(20+5)=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: *Upon completion of this course, students will be able to do the following: Analyse the asymptotic performance of algorithms. Write rigorous correctness proofs for algorithms. Demonstrate a familiarity with major algorithms and data structures.*

Prerequisite: *Discrete Math and Data Structure.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction	4	10%
Module-II: Foundations of Design and Analysis	8	25%
Module-III: Sorting Algorithms	8	15%

Module-IV: Graphs	8	25%
Module-V: Optimization Problems	4	15%
Module-VI: Selected Topics	4	10%

SYLLABUS OUTLINE:

Module-I: Introduction [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: Foundations of Design and Analysis [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 Algorithms [8L]

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:Graphs [8L]

Graph representations/storage implementations – adjacency matrix, adjacency list,

Graph traversal and connectivity – Depth-first search (DFS), Breadth-first search (BFS), Topological sort, Strongly connected components

Single-Source Shortest Paths - Bellman-Ford algorithm, Dijkstra's algorithm; All-Pairs Shortest Paths – Shortest paths and matrix multiplication, Floyd-Warshall algorithm.

Disjoint Set Manipulation: UNION - FIND Algorithms

MST- Prim's algorithm, Kruskal's Algorithm.

Module-V:Optimization Problems [4L]

Knapsack Problem, Bin-Packing Problem.

Module-VI: Selected Topics [4L]

Longest Common Sub sequence (LCS) problem,Euclid's algorithm for finding GCD, Integer exponentiation, Polynomial Evaluation.String Matching problem: Naïve String-Matching Algorithm, Knuth-Morris-Pratt (KMP) Algorithm.

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	3	1	1								1
CO2	3	3	3	1								1
CO3	3	3	3	1								1
CO4	3	3	3	1								1
CO5	3	3	3	1								1
CO6	3	3	3	1								1
Avg												

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

Course learning outcome: (CO)

1XXXXX. CO1: To be able to utilize various asymptotic notations to compute the complexity of different algorithms.

1XXXXX. CO2: To be able to explain the different standard design techniques such as divide & conquer, greedy, dynamic programming, backtracking.

1XXXXXX. CO3: To be able to infer the complexity of various sorting algorithm.

1XXXXXX. CO4: To be able to make use of various graph algorithms for solving some unknown problems.

1XXXXXX. CO5: To be able to illustrate the design technique for several combinatorial problems.

1XXXXX. CO6: To be able to relate various algorithm strategies like Branch & Bound, KMP 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 [7**L**]

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

Module-V: Multiprocessor Architecture [8**L**]

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

CO-PO Mapping

	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 [10**L**]

Signed number representation, fixed and floating point representations, character representation. Computer arithmetic - integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication - shift-and-add, Booth multiplier, carry save multiplier, etc. Division - restoring and non-restoring techniques, floating point arithmetic.

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 [4**L**]

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 [4**L**]

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-I
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

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.

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	1
10	Database Management System Lab		MC	1	0	0	1
11	Artificial Intelligence Lab		MC	1	0	0	1
Total Credit				**L+P=(22+3)=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/Cont act 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 [8**L**]

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 : [6**L**]

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

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

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: *Data Structure and Algorithm.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Amortized Analysis	4	15%
Module-II: Linear time sorting	6	20%
Module-III: Approximation algorithm	6	15%
Module-IV: Computational Geometry	4	15%
Module-V: NP Completeness	6	15%
Module-VI: Advanced Topics	10	20%

SYLLABUS OUTLINE:

Module-I: Amortized Analysis [4L]

Aggregate Method, Accounting Method and Potential Method, solving problems.

Module-II: Linear time sorting [**6L**]

Lower bounds for sorting, Counting sort, Radix sort, bucket sort.

Module-III: Approximation algorithm : [6L]

Vertex cover problem, travelling salesman problem, set covering problem, randomization and linear programming, subset sum problem.

Module-IV: Computational Geometry : [4**L**]

Line-segment properties, Convex hull, closest pair points pair of segment intersection

Module-V: NP Completeness : [**6L**]

P class, NP class, NP-Complete class, NP-Hard class, Decision vs Optimization problems. Cook's Theorem, Clique Decision Problem.

Module-VI: Advanced Topics : [10L]

Different types of algorithms and solution of simultaneous equations, DFT & FFT algorithm; integer multiplication schemes, Approximation algorithms, Necessity of approximation scheme, performance guarantee, Polynomial time approximation schemes..

***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: 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: 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	15%
Module-II: Lexical Analysis	9	20%
Module-III: Syntax Analysis	7	15%

Module-IV: Syntax directed translation and Type Checking	10	15%
Module-V: Run time environments and Intermediate Code Generation	7	15%
Module-VI: Code optimization and Code generations	7	20%

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: [**9L**]

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 : [**10L**]

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 : [**7L**]

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 : [7L]

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	2%
Module II: Linear Programming	14	30%
Module III: Transportation and Assignment problems	12	25%
Module IV: PERT – CPM	6	13%
Module V: Inventory Control	6	13%
Module VI: Queuing Theory	8	17%

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: Algorithm-II	**COURSE CREDIT :** 03 [3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** ME
CODE: XXXXXX	**SEMESTER:** 4th

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 [0-0-1]
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

Exp. No.	*Experiment Name*	*CO Mapping*
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.	**CO1**
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	**CO2**

	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.	
3.	a) Write a C program for implementing the Producer Consumer problem using Thread Synchronization. b) 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.	**CO3**
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.	**CO4**
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.	**CO5**
6.	Write a C program to calculate the seek time by applying FCFS, SSTF, SCAN,C-SCAN algorithms	**CO6**

DATABASE MANAGEMENT SYSTEM LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Database Management System Lab	**COURSE CREDIT :** 01 [0-0-1]
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

Exp. No.	***Experiment Name***	***CO Mapping***
1.	**customer (cust_id number,cust_fname varchar,cust_lname varchar ,territory varchar,cred_lmt number, mngr_id number, marital_status varchar,sex varchar ,income number)** Create and insert given data in table customer. Alter table and Add column stay_from_year Set value of stay_from_year as 2001 for Italy/America and 2003 otherwise Display credit limit attribute for America Delete the record corresponding to Meg Sen Show all attributes for Italy// Show all data in the territory Italy If territory is India and status is Single set value of credit to 7000 Rename cust_fname to first_name Rename cust_lname to last_name Create table cust1 from the old table customer(copy structure as well as data using CTAS statement). Create tables cust2 without values of cust1 using CTAS statement. Drop column income from cust1. Rename table cust1 to cust_one Insert values into cust2 table from customer table Insert values into cust3 table with attributescust_ id, f_name, l_name,Income from customer table where income > 50000 alter the table cust4 change cust id to varchar(6) and income to number(5) Add new attribute mngr_name to cust4 and insert 5 records Add attribute territory to cust4 Drop table cust3 and then bring it back. Increase the size of the column custid by 5. Suppose the customer with id no C63 has changed her	**CO1, CO2**

	last name & now it is just same as the customer with id no C68. Update customer set lname=(select lname from customer where cid=C63) where cid=C68. Display the records where territory=America & crd_lmt=25000. Display the records of all Indian customers whose income>20000. Display the name of the customer having crd_lmt between 2000 and 7000. Display the records of the customers having income 20000,24000,300,4500 using only one query. Display the records in ascending order of first name Display the records in descending order of income. Insert a duplicate record and display all the records. Suppose your friend wants to select a name from the names of the customers. Show the different names of the student.	
2.	Create a table "Student" with following structure: Student (Roll, Name, Age, Course, Math, Physics, Computer, Blrthday) Details of Attributes:- Roll Number (6) Name Varchar2 (30) Age Number (6) Course Varchar2 (5) Math Number (6, 2) Physics Number (6, 2) Computer Number (6, 2) Birthday Date 2. Create table MSc from the Student table with the same fields and same structure but without any data. 3. Display the stricture of MSc table. 4. Create table "MCA" from the Student table with the same fields and same structure but without any data. Rename Course with Department and Name with First Name. 5. Display the structure of MCA table. 6. Insert following records into the Student table:- • 1, Rahul, l9,BCA,79.5,67,89,15-jun-93 • 2,Kuna1,21,BCA,68,76,59.5,16-aug-91 • 3,Aditi,20,MSc,90,73,56,20-sep-92 • 4,Sumit,20,MCA,57.5,78,81,07-dec-91 • 5,Anirban,22,MCA,80,68,63,l5-sep-94 • 6,Kumkum,21,BCA,72,54.5,60,08-feb-95 • 7,Suman,2l,BCA,91.5,32,61,10-mar-94 • 8,Rohit,22,MSc,85,76,92,19-apr-92 7. Display all the students details from Student table. 8. Find out the details of the students with roll no 5 from Student table.	**CO1**

<table>
<tr>
<td></td>
<td>
9. Show the roll, name, marks of all subjects for all students from Student table.
10. Insert data in the “MCA“ table from “Student“ table where course is MCA.
11. Update the Math marks of the student with Roll no 7 from 91 to 95 in the “Student“ table.
12. Delete the details of the student with Roll no 2 from the “Student“ table.
</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>
Create table dept with the following attributes:
<table>
<tr><th>Column name</th><th>Data type(size)</th><th>Constraints</th></tr>
<tr><td>dept_id</td><td>number(3)</td><td>primary key</td></tr>
<tr><td>dept_name</td><td>varchar2(15)</td><td>------</td></tr>
</table>
Insert 4 depts with names and id’s 90, 69, 100 and 110.

Create table cust_100 with the following attributes:
<table>
<tr><th>Column name</th><th>Data type(size)</th><th>Constraints</th></tr>
<tr><td>emp_id</td><td>number(3)</td><td>Primary key</td></tr>
<tr><td>first_name</td><td>varchar2(10)</td><td>Initial letter capital</td></tr>
<tr><td>last_name</td><td>varchar2(10)</td><td>Initial letter capital and not null</td></tr>
<tr><td>e_mail</td><td>varchar2(20)</td><td>All upper case</td></tr>
<tr><td>ph_no</td><td>varchar2(15)</td><td>------</td></tr>
<tr><td>hire_date</td><td>Date</td><td>Should be > than 01-jan-1980</td></tr>
<tr><td>job_id</td><td>varchar2(10)</td><td>Must begin with FI or AD or IT</td></tr>
<tr><td>salary</td><td>number(8,2)</td><td>≥ 4000 & ≤ 25000</td></tr>
<tr><td>mgr_id</td><td>number(3)</td><td>------</td></tr>
<tr><td>dept_id</td><td>number(3)</td><td>Foreign key, refer table dept</td></tr>
</table>
Add 10 records to cust_100

Drop column mrg_id

Add column mgr_id and make it self referenced such that first 4 id’s correspond to first emp_id, next 4 correspond to fifth emp_id and the last 2 correspond to the ninth emp_id.
</td>
<td>CO1, CO2</td>
</tr>
<tr>
<td>4.</td>
<td>A database is being constructed for storing sales information system. A product can be described with a unique product number, product name, selling price, manufacturer name. The product can sale to a particular client and each client have it own unique client number, client name, client addresses, city, pin code, state and total balance to be required to paid. Each client order to buy product from the salesman. In the order, it has unique sales order number, sales order date, client number, salesman number (unique), billed whole payment by the party or not and its delivery date. The salesman have the name, addresses, city, pin code, state, salary of the sales man, delivery date, total quantity ordered, product rate.</td>
<td>CO1</td>
</tr>
</table>

	Q.1.1. Write the SQL queries for the following – (a) Retrieve the list of names and the cities of all the clients. (b) List the various products available. (c) Find the names of all clients having 'a' as the second letter in their names. (d) List all the clients who are located in TEZPUR. (e) Find the products whose selling price is greater than 2000 and less than or equal to 5000 (f) Add a new column NEW_PRICE into the product_master table. (g) Rename the column product_rate of Sales_Order_Details to new_product_rate. (h) List the products in sorted order of their description. (i) Display the order number and date on which the clients placed their order. (j) Delete all the records having delivery date before 25th August, 2022. (k) Change the delivery date of order number ON01008 to 16-08-22 (l) Change the bal_due of client_no CN01003 to 1200 (m)Find the product with description as 'HDD1034' and 'DVDRW' (n) List the names, city and state of the clients not in the state of 'ASSAM' (o) List of all orders that were cancelled in the of March. 2022	
5.	Create the following tables. Dept (D_no, D_name); D_no is the primary key and D_name should be not null Student (s_id , name, address ,dept_no ,age); Sid is the primary key Name should enter in upper case, Dept_no is the foreign key references to dept table also add constraints that will check the age should not below 18 years. Execute the following query 1. Insert records into the department table 2. Insert records into the student table.	**CO2**

	3. List total no. of students. 4. Alter table student by adding a new attribute marks into it. 5. Calculate total summation of marks. 6. Show the maximum marks among all the students. 7. Show the minimum marks among all the students. 8. Display how many students are studying in 'CSE' department. 9. Display the name of the students whose name starts with A. 10. Find the department name od youngest student 11. Find the address of those student who are coming from Delhi. 12. Find the number of the student who are not coming from Delhi. 13. Find the name of the students whose department is neither CSE nor ECE. 14. Find the name of the students who are studying 'Physics' 15. Display the details of the students whose age is in between 19 and 21. 16. Display the students details whose department name contains at least single 'A' Find the number of students who are studying CSE	
6.	Suppliers(Sid, SName, Address) Parts(Pis, Pname, Color) Catalog(Sid, Pid, Price) Write the query 1. Find the name of all red parts. 2. Find all prices for parts that are red or green 3. Find the sid's of all suppliers who supply a parts that is red or green 4. Find the names of all suppliers who supply a part that is not green not red 5. Find the names of all suppliers whose name contains atleast single 'a'.	**CO3**
7.	Customer(**Cus t id : integer,** cust_name: string) Item(**item_id: integer**, item_name: string, price: integer) Sale(**bill_no: integer**, bill_data: date, **cust_id: integer, item_id: integer**, qty_sold: integer) For the above schema, perform the following— Create the tables with the appropriate integrity constraints	**CO1, CO2**

<table>
<tr>
<td></td>
<td>
Insert around 10 records in each of the tables
List all the bills for the current date with the customer names and item numbers
List the total Bill details with the quantity sold, price of the item and the final amount
List the details of the customer who have bought a product which has a price>200
Give a count of how many products have been bought by each customer
Give a list of products bought by a customer having cust_id as 5
List the item details which are sold as of today
Create a view which lists out the bill_no, bill_date, cust_id, item_id, price, qty_sold, amount

Create a view which lists the daily sales date wise for the last one week.
</td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td>
-Create the following Table: Manage (ID number(3),Name varchar2(20),location varchar2(10))
<table>
<tr><th>ID</th><th>NAME</th><th>LOCATION</th></tr>
<tr><td>108</td><td>Ram</td><td>Kolkata</td></tr>
<tr><td>107</td><td>Kaushik</td><td>Naihati</td></tr>
<tr><td>106</td><td>Subhendu</td><td>Narayanpara</td></tr>
<tr><td>105</td><td>Arti</td><td>Bhatpara</td></tr>
</table>
B) Create TOUR table
<table>
<tr><th>TOUR_ID</th><th>TOUR_SPOT</th><th>FARE</th><th>TYPE</th></tr>
<tr><td>TI</td><td>Gangtok</td><td>6000</td><td>Delux</td></tr>
<tr><td>T2</td><td>Puri</td><td>2000</td><td>General</td></tr>
<tr><td>T3</td><td>Nainital</td><td>9000</td><td>Delux</td></tr>
<tr><td>T4</td><td>Andaman</td><td>12000</td><td>General</td></tr>
<tr><td>T5</td><td>Madurai</td><td>7000</td><td>Deux</td></tr>
</table>
<u>Express the following SQL:</u>

Create a view TOURVIEW for deluxe type tour containing two fields, tour-spotnames and fares.
Find all the Tour spots for fare greater than 8000 and Delux type tour from a) TOUR table, b) TOURVIEW view
insert three rows to TOURVIEW.the location will be Bangladesh, Delhi, Hyderabad. 4)Display these Records.Are they seen in TOURVIEW? Are they seen in TOUR Table? 5)Make them Seen in TOURVIEW.
Create a view Tour_type_num containing two fields i.e. tour_type and its total number of count. try to insert records via this view. Errors!!,Explain why?
</td>
<td>CO1, CO2</td>
</tr>
<tr>
<td>9.</td>
<td>
Oracle table DUAL
Types of functions
Numeric Functions
(ABS,POWER,ROUND,SQRT,EXP,GREATEST,LEAST,MOD,TRUNC,FLOOR,CEIL)
</td>
<td>CO2, CO3</td>
</tr>
</table>

String Functions
(LOWER,UPPER,SUBSTR,INSTR,LPAD,RPAD,
TRIM,LTRIM,RTRIM,LENGTH,INITCAP,
SOUNDEX)

Conversion Functions
(TO_CHAR, TO_NUMBER,TO_DATE)

Date Functions
(SYSDATE, SYSTIMESTAMP, ADD_MONTHS,
LAST_DAY, MONTHS_BETWEEN, NEXT_DAY,
ROUND)

Create the following tables with the data types and constraints

Sailor(sid, sname, mname, surname, rating, age)

Constraints:

sid primary key and start with small s
sname initial letter capital
surname not null
rating default zero
age not null

Boat(bid, bname, color)

constrainsts

bid primary key & start with small b
bname all upper case
color red,green,blue

sailor_boat(sid, bid, day, SID_BID)

constraints

sid Foreign key referencing sailor
bid Foreign key referencing boat
day <1-JAN-2000
(SID, BID) composite primary key

For the above schema, perform the following query—

Find the names and ages of all sailors.
Show names under the heading of names_of_sailors and add 2 to age.
Select all records from sailors in ascending order by name;
Show all sailors name.
Select all distinct sailors name.
Show all distinct sailors names, ratings who have rating between 5 and 10.
Select all records from sailors in ascending order by rating and descending order by age.
Select all records from sailors whose rating>7.
Find records for sailor name Horatio and age=35.4.
Find records for sailor name Horatio or age=35.4.
Select names of sailors who have reserved boat 104.
Find sid of sailors who have reserved red boat
Select records for name beginning with ‘B’.
Select records for name containing ‘B’/’b’.
Select names for rating present.

	Select names for rating absent. Find color of boats reserved by Lubber. Find names of sailors who have reserved a red boat or a green boat. Find names of sailors who have reserved a red boat but not a green boat. Find names of sailors who have reserved boat 103. Find names of sailors who have reserved red boat. 22)Find names of sailors whose rating is>10/20/30 using multirow subquery operator. 23)Find names of sailors whose rating is>10 & 20 & 30 using multirow subquery operator. 24) Find average age of sailors with rating 10. 25) Find the name of sailor who are older than oldest sailor of rating=10. 26) Find the age of youngest sailor for each rating level. 27) Find sailor with highest rating. 28) Find sailors who have reserved all boats.	
10.	1. Oracle table DUAL Types of functions Numeric Functions (ABS,POWER,ROUND,SQRT,EXP,GREATEST,LEAST,MOD,TRUNC,FLOOR,CEIL) String Functions (LOWER,UPPER,SUBSTR,INSTR,LPAD,RPAD,TRIM,LTRIM,RTRIM,LENGTH,INITCAP, SOUNDEX) Conversion Functions (TO_CHAR, TO_NUMBER,TO_DATE) Date Functions (SYSDATE, SYSTIMESTAMP, ADD_MONTHS, LAST_DAY, MONTHS_BETWEEN, NEXT_DAY, ROUND) Create the following tables with the data types and constraints **Sailor(sid, sname, mname, surname, rating, age)** **Constraints:** sid primary key and start with small s sname initial letter capital surname not null rating default zero age not null **Boat(bid, bname, color)** **constrainsts** bid primary key & start with small b bname all upper case	**CO2, CO1**

	color red,green,blue **sailor_boat(sid, bid, day, SID_BID)** **constraints** sid Foreign key referencing sailor bid Foreign key referencing boat day <1-JAN-2000 (SID, BID) composite primary key 1. Display s_name with left side padding by at least 3 *. 2. Display length of each name. 3. Display all sailors names in uppercase. 4. Display all sailors' names in lower case. 5. Display all sailors names in sentence case. 6. Display 4th to 7th letter of sailors name. 7. Display 4th and 7th letter of sailors name. 8. Concat s_id, s_name. 9. Display square root of rating. 10. Display floor values of all ages. 11. Display ceiling values of all ages. 12. Select all s_name with 1st 2 letters off. 13. List months between today and reservation date. 14. Select day between today and reservation date. 15. Shift all reservation day by 2 months. 16. Shift all reservation day earlier by 3 months. 17. Suppose after sailing they enjoy their next Monday as holiday. Find that day.	

ARTIFICIAL INTELLIGENCE LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Artificial Intelligence Lab	**COURSE CREDIT :** 01 [0-0-1]
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/IoT		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	1	0
9	Software Engineering Lab		MC	1	0	0	1	0
10	Digital Image Processing Lab /Machine Learning Lab /IoT Lab		ME	1	0	0	1	0
Total Credit				**L+P+S=(19+3+2)=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	8	
Module-II: Image Enhancement Techniques	10	
Module-III: Image Segmentation	10	
Module-IV: Morphological Operations	6	
Module-V: Image Registration	4	
Module-VI: Color Image Processing	10	

SYLLABUS OUTLINE:

Module-I: Introduction to Image processing: [8L]

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: [**10L**]

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: [10L]

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: [10L]

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	12	
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 [12L]

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

INTERNET OF THINGS

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Internet of Things	**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: understand each component that makes up an IoT system. Differentiate between the levels of IoT stack and be familiar with key technologies and protocols employed at each layer of stack. This course is intended to teach the basics involved in sensors, microcontrollers and microprocessors that will be used to prepare a smart system.*

Prerequisite: *Knowledge of Sensor nodes, Networks, Python language.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Internet of Things	6	17%
Module-II: IoT Architecture	4	11%
Module-III: IoT Protocols	8	22%
Module-IV: Web of Things	4	11%
Module-V: IoT Applications	8	22%

Module-VI: IoT and Beyond	6	17%

SYLLABUS OUTLINE:

Module-I: Introduction to Internet of Things [6L]

Introduction-Definition & Characteristics of IoT, Physical Design of IoT- Things in IoT, IoT Protocols, Logical Design of IoT- IoT Functional Blocks, IoT Communication Models, IoT Communication APIs, IoT Enabling Technologies- Wireless Sensor Networks, Cloud Computing, Big Data Analytics, Communication Protocols, Embedded Systems, The Identifiers in IoT.

Module-II: IoT Architecture [**4L**]

IoT Open source architecture (OIC) - OIC Architecture & Design principles- IoT Devices and deployment models, An Open source IoT stack architecture, Sensors and actuators for IoT applications, Concepts of IoT Integration with Sensors and Cloud.

Module-III: IoT Protocols [8L]

Protocol Standardization for IoT, M2M-Difference between IoT and M2M, SDN and NFV for IoT-Software Defined Networking, M2M and WSN Protocols, SCADA and RFID Protocols, Issues with IoT Standardization, Unified Data Standards, Protocols, IEEE802.15.4.

Wireless Technologies for IoT: WPAN Technologies for IoT: IEEE 802.15.4, Zigbee, HART, NFC, Z-Wave, BLE, Bacnet, Modbus.

IP Based Protocols for IoT: IPv6, 6LowPAN, RPL, REST, AMPQ, CoAP, MQTT. Edge connectivity and protocols.

Module-IV: Web of Things [**4L**]

WEB OF THINGS - Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT – Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals and Business Intelligence.

Module-V: IoT Applications [**8L**]

IOT APPLICATIONS - IoT applications for industry: Future Factory Concepts, Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT, Legal challenges, IoT design Ethics, IoT in Environmental Protection

Module-VI: IoT and Beyond [6L]

Use of Big Data and Visualization in IoT, Industry 4.0 Concepts, Overview of RFID, Low-power design (Bluetooth Low Energy), range extension techniques (data mining and mesh

networking), and data-intensive IoT for continuous recognition applications. Overview of Android / IOS App Development tools & Internet of Everything

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.

Text & Reference books:

Text Books:

1. Internet of Things, A Hands on Approach, by Arshdeep Bahga & Vijay audisetti, University Press.

2. "Sheng-Lung Peng, S. Pal, Lianfen Huang (Eds.), "Principles of Internet of Things (IoT) Ecosystems: Insight Paradigm", 2019. [Springer] [ISBN: 978-3030335953]"

Reference Books:

3. The Internet of Things, by Michael Millen, Pearson

4. "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", by Pethuru Raj and Anupama C. Raman (CRC Press)

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3					1						2
CO2		3	2									
CO3			3		3							
CO4		3		2								
CO5			3		3		2					
CO6			3		3		3					2
Avg	0.5	1	1.83	0.33	1.5	0.17	0.83	-	-	-	-	0.67

Highly Correlated: **3**

Moderately Correlated**: 2**
Slightly Correlated: **1**

Course learning outcome*: (CO)*

1XXXXX. CO1: To be able to understand the various concepts, terminologies and architecture of IoT systems.

1XXXXX. CO2: To be able to use sensors and actuators for design and architecture of IoT.

1XXXXX. CO3: To be able to understand and apply various protocols for design of IoT systems.

1XXXXX. CO4:. To be able to apply various techniques of web applications and analytics in IoT.

1XXXXX. CO5: To be able to analyze various applications of IoT.

1XXXXX. CO6: To be able to develop different APIs to connect IoT related technologies.

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 [0-0-1]
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 [0-0-1]
DEPARTMENT: Computer Science	**CATEGORY:** MC

CODE: XXXXXX	**SEMESTER:** 5th

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 [0-0-1]
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

Exp. No.	***Experiment Name***	***CO Mapping***
1.	Write a program to add, subtract and multiply two matrix. Show the output for positive and negative test cases.	**CO1**
2.	Write a program to inverse a matrix. User will provide the matrix size.	**CO1**
3.	Write a program to read the gray scale sample image and store it in a 2D matrix and do the following (don't use any inbuilt library function). a. Find the dimension of the image matrix. b. Find the maximum and minimum intensity levels. c. Inverse the intensity level of the image matrix and then display/save the modified image.	**CO1**
4.	Write a program to read the sample grayscale image and store it in a 2D matrix and do the following (try to avoid inbuilt library function). a. Rotate the image 180 degree and then display/save the modified image. Hint: In 180 degree rotation, the intensity value in pixel 1,1 will be placed at the last row and last column. b. Rotate the image 45 degree and then display/save the modified image.	**CO1**

	Rotate the image 90 degree and then display/save the modified image.	
5.	Write a code to read a gray scale image file and then do the followings without using inbuilt functions. a. Increase the brightness of the image. b. Automatically increase the brightness to the maximum level without losing the property of the image. c. Enhance the contrast of the image. d. Plot the histogram. e. Equalized the histogram of the image and then display the image.	**CO1, CO2**
6.	Write a code to read a gray scale image file and then do the followings without using inbuilt functions. a. Increase contrast of an image using logarithimic function, square function, square root function, exponential function, power function.	**CO2**
7.	Perform correlation and convolution operation. Show the corresponding output. Image 1 2 0 6 7 5 4 3 1 1 0 2 3 4 5 1 2 3 1 0 7 6 4 2 Mask 1 0 1 1	**CO1, CO2**
8.	Write a program for mirror image generation. You can take any image for your references.	**CO2**
9.	a) Program of sharpen image using gradient mask without using any inbuilt function. b) Program for morphological operation: erosion and dilation. You can show the output performance of all these operation on any image.	**CO3, CO4**
10.	Enhance the given "sample" image using a. Laplacian filter b. Median Filter c. Average Filter d. Gaussian Filter And display the enhanced images separately.	**CO2**
11.	Write a program to add the following noises to an image: a. Salt & pepper b. Speckle c. Poisson d. Gaussian	**CO2**
12.	To write a program to perform following thresholding operations- a) Otsu Thresholding b) Adaptive Thresholding c) Multilevel Thresholding	**CO4**

13.	a. Segment the green area and red area from the given image. b. Segment the text area (if any). c. Segment the only white color area .	**CO4**
14.	Find only the diagonal edges from the previous given sample image.	**CO6**
15.	Write a program to segment the text area from the given image. 	**CO6, CO5**

MACHINE LEARNING LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Machine Learning Lab	**COURSE CREDIT :** 01 [0-0-1]
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 IoT sensors, microcontrollers and able to apply those using Python Programming.

List of practical

Sl No	***List of Experiment***	***CO Mapping***
1	Design a Decision Tree algorithm for the multiclass problem and predict which drug among drugs such as drug X, drug Y, and drug C should be given to a patient using the decision tree algorithm. Find the accuracy of the decision tree in predicting the correct drug for the patient. **Dataset: drug.csv**	CO2, CO4
2	Design KNN algorithm for the classification and predict which category a customer belongs to on the basis of the data provided by a telecommunications firm using KNN classifier. Find the accuracy of the KNN algorithm in predicting the category of a customer. **Dataset: teleCust.csv**	CO2, CO5

3	Design SVM classifier with different Kernal tricks and tolerance and predict if a customer has a benign tumour or malignant tumour (cancer) based on the features provided using SVM classifier. Use the following kernel for the SVM algorithm: a) Linear b) Polynomial c) RBF d) Sigmoid Find the following metrics for each of the SVM algorithms: 1) Accuracy 2) Recall 3) Precision 4) F1-Score 5) Error rates 6) Confusion Matrix **Dataset: samples_cancer.csv**	CO2, CO5
4	Design a Naïve Bayes model for both Gaussian and Multinomial distribution and predict if a person is diabetic or not, based on the features provided using Naïve Bayes algorithm. Find the accuracy and F1 scores of both models. **Dataset: pima-indians-diabetes.data.csv**	CO2, CO4
5	Design a prediction model by applying SVM, Naive Bayes, Decision tree and KNN to predict diabetes based on features set. Compare the four classification algorithms with performance metrics such as accuracy, recall, precision, and F1- score. Also, design the heat map confusion matrix for the above algorithms and construct ROC curve for comparison. **Dataset: pima-indians-diabetes.data.csv**	CO2, CO5
6	6. Perform linear regression algorithm to predict house price from given features. Consider both univariate and multivariate cases in this problem. Elaborate gradient descent algorithm and hyper-parameter tuning for the best result with predefined convergent criteria. **Dataset: a) Housing price dataset, b) Fuel consumption dataset**	CO2, CO5
7	7. Perform a nonlinear regression algorithm to predict China's GDP from the year 1960 to 2014 from given features. Elaborate gradient descent algorithm and hyper-parameter tuning for the best result with predefined convergent criteria. **Dataset: China's GDP dataset**	CO2, CO4
8	Perform a logistic regression algorithm to classify dissatisfied customers. Utilize gradient descent with regularization for hyper-parameter tuning for the given dataset. Also, generate log-loss curve for this problem. **Dataset: Telecom Churn dataset**	CO2, CO4
9	Design the artificial neural network that computes the functionality of XOR, NOR, and NAND gates.	CO4
10	Perform a Multilayer perception neural network to classify flower type. Utilize a number of hidden layer 5 and 200 to 400 iterations with learning rate. Try with different loss functions/ activation functions such as MSE, Cross entropy, sigmoid, tanh, ReLU along with different optimizer GD, SGD, Adam. Illustrate the result with performance metrics and observe the Weight, Loss curve, and accuracy curve. **Dataset: Iris dataset**	CO4, CO5
11	Perform k- means clustering algorithm for customer segmentation from	CO2, CO4

	given features. Utilize Euclidean distance and Manhattan distance for this problem and measure the performance of inertia with cluster validity index. Also, plotting in terms of 2D and 3D clusters this problem. **Dataset: Customer segmentation dataset**	
12	Design and implement two hierarchical clustering models as Agglomerative algorithm and a Divisive algorithm to group several vehicles. Utilize single, complete, and average linkage to define the cluster. Also, draw the dendrogram for this problem. **Dataset: Vehicle dataset**	CO2, CO5

INTERNET OF THINGS LAB

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Internet of Things Lab	**COURSE CREDIT :** 01 [0-0-1]
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 IoT sensors, microcontrollers and able to apply those using Python Programming.

List of practical

Exp. No.	***Experiment Name***	***CO Mapping***
1.	Install python in LINUX and write a program to understand different data types in python.	**CO3**
2.	Write a program for arithmetic operations in python.	**CO3**
3.	Write a program for looping statement in python. Study and Install IDE of arduino and its different types.	**CO3**
4.	Write a program using arduino IDE for blinking LED	**CO3**
5.	Write a program for RGB LED using Arduino.	**CO3**
6.	Study the temperature sensor and write program for monitoring temperature using arduino.	**CO3**
7.	Study and implement RFID, NFC & GSM module using arduino.	**CO3**
8.	Study and implement MQTT protocol using arduino.	**CO3**

9.	Study and configure Raspberry Pi.	**CO4**
10.	WAP for LED blinking using Raspberry Pi.	**CO4**
11.	Study and implement Zigbee Protocol using Arduino or Raspberry Pi	**CO3**

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/Embedded Systems		ME	4	4	0	0	0
3	Cloud Computing/Soft Computing/Wireless Sensor Network		ME	4	4	0	0	0
4	NM Elective-II		NM	4	4	0	0	0
5	Soft-Skill Development-V		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	1	0
Total Credit				**L+P+S=(20+1+2)=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/Embedded Systems

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

THEORY

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

CO-PO Mapping

	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.

EMBEDDED SYSTEMS

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Embedded Systems	**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: 1. To introduce the Building Blocks of Embedded System*
2. To Educate in Various Embedded Development Strategies
3. To Introduce Bus Communication in processors, Input/output interfacing.
4. To impart knowledge in various processor scheduling algorithms.
5. To introduce Basics of Real time operating system and example tutorials to discuss on one real time operating system tool

Prerequisite: *Strong knowledge of computer architecture and organization*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Embedded Systems Basics	6	
Module-II: Embedded Hardware	10	
Module-III: Embedded Software	10	
Module-IV: Embedded system Design	10	
Module-V: Introduction to RTOS	8	
Module-VI: Case Studies and Applications of embedded systems	4	

SYLLABUS OUTLINE:

Module-I: Embedded Systems Basics [6L]

An Embedded System-Definition, Examples, Current Technologies, Integration in system Design, Embedded system design flow, hardware design concepts, software development, processor in an embedded system and other hardware units, introduction to processor based embedded system design concepts.

Module-II: Embedded Hardware [10L]

Embedded hardware building blocks, Embedded Processors – ISA architecture models, Internal processor design, processor performance, Board Memory – ROM, RAM, Auxiliary Memory, Memory Management of External Memory, Board Memory and performance. Embedded board Input / output – Serial versus Parallel I/O, interfacing the I/O components, I/O components and performance, Board buses – Bus arbitration and timing, Integrating the Bus with other board components, Bus performance

Module-III: Embedded Software [10L]

Device drivers, Device Drivers for interrupt-Handling, Memory device drivers, On-board bus device drivers, Board I/O drivers, Explanation about above drivers with suitable examples .Embedded operating systems – Multitasking and process Management, Memory Management, I/O and file system management, OS standards example – POSIX, OS performance guidelines, Board support packages, Middleware and Application Software – Middle ware, Middleware examples, Application layer software examples.

Module-IV: Embedded system Design [10L]

Concepts of concurrency, processes, threads, mutual exclusion and inter-process communication, Models and languages for embedded software, Synchronous approach to embedded system design, Scheduling paradigms, Scheduling, Algorithms, Hardware/Software Co-design.

Module-V: Introduction to RTOS [8L]

Basic Design using RTOS, Interfacing, RISC Processor: Architecture, Memory, Reset and Interrupt, Functions, Parallel I/O ports, Timers/Counters, Serial Communication, Analog Interfaces.

Module-VI: Case Studies and Applications of embedded systems [4L]

Embedded Product Development Life Cycle – Description – Objectives -Phases – Approaches. Recent Trends in Embedded Computing.

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. Lee, Marilyn Wolf. "Embedded Systems: A Contemporary Design Tool." Wiley, 2008.
2. Valvano, Jonathan W. "Embedded Systems: Introduction to Arm Cortex-M Microcontrollers." 6th ed., Cengage Learning, 2020.
3. Yiu, Jonathan. "The Definitive Guide to Arm Cortex-M3 and Cortex-M4 Processors." 3rd ed., Newnes, 2013.
4. Simon, David E., et al. "An Embedded Software Primer." Pearson, 1999.
5. Mazidi, Muhammad Ali, Janice Gillispie Mazidi, and Danny Causey. "The 8051 Microcontroller and Embedded Systems Using Assembly and C." 2nd ed., Pearson, 2006

Reference Books:

1. Ganssle, Jack. "The Art of Designing Embedded Systems." 2nd ed., Newnes, 2008.
2. Eady, Fredrick M. "Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers." Newnes, 2005.
3. Kim, K. C., and Marilyn Wolf. "Real-Time Systems." Pearson, 2000.
4. Liu, Sam S. "Real-Time Systems." Pearson, 2000.
5. Valvano, Jonathan W. "Embedded Systems: Real-Time Interfacing to Arm Cortex-M Microcontrollers." 2nd ed., Cengage Learning, 2012.

CO-PO Mapping

	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)*

1XXXXX. CO1:.To able to define an Embedded System and understand it's design flow.

1XXXXX. CO2: To be able to demonstrate Embedded Hardware building blocks and various Embedded Processor architecture models

1XXXXX. CO3: To be able to design various device drivers.

1XXXXX. CO4: To be able to design an embedded system

1XXXXX. CO5: To be able to execute various ECAD tools in the design of the embedded systems

1XXXXX. CO6: To be able to experiment of Embedded Systems

Cloud Computing/Soft Computing/Wireless Sensor Network

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	6	
Module-II: Use of Platforms in Cloud Computing	6	
Module-III: Cloud Infrastructure	6	
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 [6L]

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 [6L]

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 [6L]

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

It has to be prepared

	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	8	
Module-IV: Artificial Neural Network	8	
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 [8L]

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 [8L]

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)*

1XXXXX. CO1: To Understand intelligent systems leveraging the paradigm of soft computing techniques.

1XXXXX. CO2: To get the knowledge solutions by various soft computing approaches for finding the optimal solutions.

1XXXXX. CO3: To Recognize the feasibility of applying a soft computing methodology for a particular problem

1XXXXX. CO4: To Design the methodology to solve optimization problems using fuzzy logic, genetic algorithms and neural networks.

1XXXXX. CO5: To Design hybrid system to revise the principles of soft computing in various applications

1XXXXX. CO6: To Analyse the applications of Soft Computing Systems

WIRELESS SENSOR NETWORK

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Wireless Sensor Network	**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:*

1. To understand the basics of Sensor Networks.
2. To learn various fundamental and emerging protocols of all layers.
3. To study about the issues pertaining to major obstacles in establishment and efficient management of Ad-hoc and sensor networks.
4. To understand the nature and protocols of sensor networks.
5. To understand various security practices and protocols of Sensor Networks.

Prerequisite: *Before learning the concepts of Wireless Sensor Network, you should have a basic knowledge of Data Communication Networks, and Computer Networks.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Characteristics of WSN	8	
Module-II: Medium Access Control Protocols	6	
Module-III: Routing Challenges and Design	8	
Module-IV: Embedded Operating Systems	6	
Module-V: Introduction to Tiny OS	4	
Module-VI: Applications of WSN	4	

SYLLABUS OUTLINE:

Module-I: Characteristics of WSN [8L]

Characteristic requirements for WSN - Challenges for WSNs – WSN vs Adhoc Networks - Sensor node architecture – Commercially available sensor nodes –Imote, IRIS, Mica Mote, EYES nodes, BTnodes, TelosB, Sunspot -Physical layer and transceiver design considerations in WSNs, Energy usage profile, Choice of modulation scheme, Dynamic modulation scaling, Antenna considerations.

Module-II: Medium Access Control Protocols [**6L**]

Fundamentals of MAC protocols - Low duty cycle protocols and wakeup concepts - Contention based protocols - Schedule-based protocols - SMAC - BMAC - Traffic-adaptive medium access protocol (TRAMA) - The IEEE 802.15.4 MAC protocol.

Module-III: Routing Challenges and Design: [8L]

Issues in Wireless Sensor Networks, Flooding and gossiping – Data centric Routing – SPIN – Directed Diffusion – Energy aware routing - Gradient-based routing - Rumor Routing – COUGAR – ACQUIRE – Hierarchical Routing - LEACH, PEGASIS – Location Based Routing – GF, GAF, GEAR, GPSR – Real Time routing Protocols – TEEN, APTEEN, SPEED, RAP - Data aggregation - data aggregation operations - Aggregate Queries in Sensor Networks - Aggregation Techniques – TAG, Tiny DB.

Module-IV: Embedded Operating Systems: [**6L**]

Operating Systems for Wireless Sensor Networks – Introduction - Operating System Design Issues - Examples of Operating Systems – TinyOS – Mate – MagnetOS – MANTIS - OSPM - EYES OS – SenOS – EMERALDS – PicOS.

Module-V: Introduction to Tiny OS : [**4L**]

NesC – Interfaces and Modules- Configurations and Wiring - Generic Components - Programming in Tiny OS using NesC, Emulator TOSSIM.

Module-VI: Applications of WSN: [4L]

WSN Applications - Home Control - Building Automation - Industrial Automation - Medical Applications - Reconfigurable Sensor Networks - Highway Monitoring - Military Applications - Civil and Environmental Engineering Applications - Wildfire Instrumentation - Habitat Monitoring - Nanoscopic Sensor Applications – Case Study: IEEE 802.15.4 LR-WPANs Standard - Target detection and tracking - Contour/edge detection - Field sampling.

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. Holger Karl and Andreas Willig. "Protocols and Architectures for Wireless Sensor Networks." Wiley, 2005.
2. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. "Wireless Sensor Networks: A Survey." Computer Networks, vol. 38, no. 4, 2002, pp. 393-422.
3. Feng Zhao, Leonidas Guibas. "Wireless Sensor Networks: An Information Processing Approach." Morgan Kaufmann, 2004.
4. Jun Zheng and Abbas Jamalipour. "Wireless Sensor Networks: A Networking Perspective." Wiley, 2009.
5. Anna Hac. "Wireless Sensor Network Designs." Wiley, 2003.

Reference Books:

1. C.S. Raghavendra, Krishna M. Sivalingam, and Taieb Znati. "Wireless Sensor Networks." Springer, 2004.
2. Hossam S. Hassanein and Hossam M. Sharaieh. "Wireless Sensor Networks: A Networking Perspective." Wiley, 2016.
3. Tony D. Givargis. "Embedded Sensor Networks." IEEE Computer Society, 2009.
4. Feng Zhao, Leonidas Guibas, Ji Liu, and Younghun Jung. "Information Processing in Wireless Sensor Networks." Elsevier, 2004.

CO-PO Mapping : **It has to be prepared**

	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)

1XXXXX. CO1: To be able to understand wireless architecture and sensor node architectures. (Understand)

1XXXXX. CO2: To demonstrate knowledge of MAC protocols developed for WSN. (Apply)

1XXXXXX. CO3: To demonstrate knowledge of routing protocols developed for WSN. (Apply)

1XXXXXX. CO4: To analyze the characteristics of Operating systems for sensor nodes. (Analyze)

1XXXXXX. CO5: To be able to evaluate different WSN Standards. (Evaluate)

1XXXXX. CO6: To be able to establish a Sensor network environment for different type of applications. (Create)

NM Elective-II

PROVIDED BY RESPECTIVE DEPARTMENT / SCHOOL

Soft-Skill Development-V

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 [0-0-1]
DEPARTMENT: Computer Science	**CATEGORY:** MC
CODE: XXXXXX	**SEMESTER:** 2nd

FOURTH YEAR

SEMESTER-VII

Sl No	Course Title	Code		Credit	Type			
					L	T	P	S
1	Cyber Security/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				**L+P+S=(8+0+8)=16 Credit**				

CYBER SECURITY/DEEP LEARNING/MOBILE COMPUTING

CYBER SECURITY

SCHOOL : School of Engineering	**COURSE TYPE:** L-T-P
NAME: Cyber security	**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 prepare students with the technical knowledge and skills needed to protect and defend computer systems and networks.*
2. *To develop graduates that can plan, implement, and monitor cyber security mechanisms to help ensure the protection of information technology assets.*
3. *To develop graduates that can identify, analyze, and remediate computer security breaches.*

Prerequisite: *Before learning the concepts of Cyber Security, you should have a basic knowledge of Mathematics.*

Course content/Syllabus:

Module no.	No of lecture/Contact hour	Weightage (%)
Module-I: Introduction to Cyber security	8	
Module-II: Cyber Crime and Cyber law	8	
Module-III: Social Media Overview and Security	6	
Module-IV: E-Commerce and Digital Payments	10	
Module-V: Digital Devices Security	8	
Module-VI: Tools and Technologies for Cyber Security	8	

SYLLABUS OUTLINE:

Module-I: Introduction to Cyber security: [8L]

Defining Cyberspace and Overview of Computer and Web-technology, Architecture of cyberspace, Communication and web technology, Internet, World wide web, Advent of internet, Internet infrastructure for data transfer and governance, Internet society, Regulation of cyberspace, Concept of cyber security, Issues and challenges of cyber security.

Module-II: Cyber Crime and Cyber law: [**8L**]

Classification of cyber-crimes, Common cybercrimes- Cyber Crime targeting computers and mobiles, Cyber Crime against women and children, financial frauds, social engineering attacks, malware and ransom ware attacks, zero day and zero click attacks, Cybercriminals modus-operandi, Reporting of Cyber Crimes, Remedial and mitigation measures, Legal perspective of Cyber Crime, IT Act 2000 and its amendments, Cyber Crime and offences, Organisations dealing with Cyber Crime and Cyber security in India, Case studies..

Module-III: Social Media Overview and Security: [6L]

Introduction to Social networks. Types of Social media, Social media platforms, Social media monitoring, Hashtag, Viral content, Social media marketing, Social media privacy, Challenges, opportunities and pitfalls in online social network, Security issues related to social media, Flagging and reporting of inappropriate content, Laws regarding posting of inappropriate content, Best practices for the use of Social media, Case studies.

Module-IV: E-Commerce and Digital Payments: [**10L**]

Definition of E- Commerce, Main components of E-Commerce, Elements of E-Commerce security, E-Commerce threats, E-Commerce security best practices, Introduction to digital payments, Components of digital payment and stake holders, Modes of digital payments- Banking Cards, Unified Payment Interface (UPI), e-Wallets, Unstructured Supplementary

Service Data (USSD), Aadhar enabled payments, Digital payments related common frauds and preventive measures. RBI guidelines on digital payments and customer protection in unauthorised banking transactions. Relevant provisions of Payment Settlement Act,2007.

Module-V: Digital Devices Security : [**8L**]

End Point device and Mobile phone security, Password policy, Security patch management, Data backup, Downloading and management of third party software, Device security policy, Cyber Security best practices.

Module-VI: Tools and Technologies for Cyber Security: **[8L]**

Significance of host firewall and Ant-virus, Management of host firewall and Anti-virus, Wi-Fi security, Configuration of basic security policy and permissions.

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. Dieter Gollmann. "Computer Security." 3rd ed., John Wiley & Sons, 2011.
2. William Stallings. "Network Security Essentials." 6th ed., Pearson, 2017.
3. Michael T. Goodrich and Roberto Tamassia. "Introduction to Computer Security." Pearson, 2011.
4. Adam Shostack and Andrew Stewart. "The New School of Information Security." Addison-Wesley, 2008.
5. Chuck Easttom. "Computer Security Fundamentals." 3rd ed., Pearson, 2015.

Reference Books:

1. Eric Cole. "Cybersecurity for Dummies." 1st ed., John Wiley & Sons, 2019.
2. Peter J. R. Smith. "Cyber Security." Cambridge University Press, 2015.
3. Bruce Schneier. "Secrets and Lies: Digital Security in a Networked World." John Wiley & Sons, 2000.
4. Eugene H. Spafford and Gary R. Wassermann. "Web Security, Privacy & Commerce." 2nd ed., O'Reilly Media, 2001.
5. Richard Stiennon. "Surviving Cyberwar." Government Institutes, 2010.

CO-PO Mapping: **It Has to be prepared.**

	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)*

1XXXXX. CO1: After completion of this course, students would be able to understand the concept of Cyber security and issues and challenges associated with it. (Understand)

1XXXXX. CO2: Students, at the end of this course, should be able to understand the cybercrimes, their nature, legal remedies and as to how report the crimes through available platforms and procedures. (Understand)

1XXXXX. CO3: On completion of this course, students should be able to appreciate various privacy and security concerns on online Social media and understand the reporting procedure of inappropriate content, underlying legal aspects and best practices for the use of Social media platforms. (Knowledge)

1XXXXX. CO4: After the completion of this course, students would be able to understand the basic concepts related to E-Commerce and digital payments. They will become familiar with various digital payment modes and related cyber security aspects, RBI guidelines and preventive measures against digital payment frauds. (Analyze)

1XXXXXX. CO5: Students, after completion of this course will be able to understand the basic security aspects related to Computer and Mobiles. (Analyze)

1XXXXX. CO6: The students will be able to use basic tools and technologies to protect their devices. (Apply)

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	6	
Module-V: Optimization AND Generalization	6	
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: [6L]

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: [6L]

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	8	
Module-III: Mobile Network Layer	8	
Module-IV: Mobile Transport and Application Layer	6	
Module-V: Cognitive Radio Networks	4	
Module-VI: Mobile Platforms and Applications	4	

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: [**8L**]

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 : [8L]

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: [**4L**]

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 : [4L]

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

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				**L+P+S=(4+0+8)=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	10%
Module-II: Distributed Database Design	8	15%
Module-III: Distributed Query Optimization	6	20%
Module-IV: Reliability issues in DDBS	6	20%
Module-V: Parallel Database Systems	4	15%
Module-VI: Advanced Topics Mobile	6	20%

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 [8L]

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 [6L]

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 [6L]

Types of failures; Reliability techniques; Commit protocols; Recovery protocols Algorithm

Module-V: Parallel Database Systems [4]

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: 8**th

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	Weightage (%)

	hour	
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: Discourse and 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 [4L]

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 [4L]

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: [8L]

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: [8L]

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: [6L]

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: [6L]

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