NEP based Course Structure & Curriculum

For
B. Tech. Programme
In

Computer Science & Engineering

Motilal Nehru National Institute of Technology Allahabad Department of Computer Science & Engineering

COMPLETE COURSE STRUCTURE OF B.TECH. PROGRAMME WITH RESPECT TO NEP-2020

Vision and Mission of the Institute

VISION

To attain a distinct identity for the Institute through technology innovation, knowledge creation and dissemination for the benefit of the society.

MISSION

- To nurture an eco-system for continuous enhancement of value-based teaching and learning process in the emerging areas of technology.
- To train quality human and knowledge resources in the service of society.
- To develop sustainable products and technologies.

DEPARTMENT VISION

The Computer Science & Engineering Department of Motilal Nehru National Institute of Technology Allahabad would continue to provide leading programs in the field of computing. The graduates produced would be globally recognized as innovative and well-prepared computing professionals. These graduates would be leaders and innovators in the industry, education, and other walks of social life. The faculty working in the department would influence the national and international agenda through quality in teaching and research. Department would strive for inter-disciplinary pursuits that reinforces the impact of computation in other disciplines.

DEPARTMENT MISSION

The Department Computer Science & Engineering has following missions.

- To offer high quality undergraduate and graduate programs, to train the students in different aspects of computing discipline and to offer other training programs to enhance, augment, and/or update technical skills of stake holders.
- To provide a learning environment that helps students to be successful in their professional lives and to prepare students to be lifelong learners by offering a solid theoretical foundation in computing along with applied computing experiences and educating them about their professional, and ethical responsibilities.
- To have high quality faculty and staff with an appropriate know how and degrees with a commitment to remain professionally advanced through ongoing research and professional development programs.
- To attract and retain a diverse and multicultural population of student, faculty, and staff.

B.Tech. Course Structure 1st Year I Semester

S.No.	Course & Course code	Cat.	L	T	P	Credits	Contact hours	Remarks
1.	Chemistry (CYN11502)	**CEF** (EE)	2	1	2	4	5	Branch Specific Physics and Chemistry Courses (Alternatively in each semester)
2.	Mathematics-I (MAN11101)	**CEF** (EE)	3	1	0	4	4	Common course for all branches
3.	English Language & Technical Communication (HSN11600)	**PCE**(SA)	2	0	2	3	4	Common course for all branches *As per the clause 23.13 of the NEP 2020
4.	Core Engineering Essentials Courses-I (Flexible L-T-P) Computer-Programming (CSN11101)	**CEE** (CES)	2	0	2	3	4	Branch specific course for the students of the branch only
5.	Core Engineering Essentials Courses-II (Flexible L-T-P) Computer Organization (CSN11102)	**CEE** (CES)	3	0	0	3	3	Branch specific course for the students of that branch only
6.	Discrete Mathematics (CS12105)	**PCE**(EE)	2		0	2	2	Common course (Alternatively in each semester) #If any department does not want to adopt this/these course(s) for the specific branch, the department may float branch-specific course in its/their place
7.	Environment and Climate Change@	**PCE** (HSS)	2	0	0	2	2	Common course for all branches @Audit Course
8.	Extra Academic Activity-A/ Extra Academic Activity-B (EAN11700-EAN11710)	**EAA** (SA)	-	-	4	2&	4**	Common course for all branches (with different titles) **Engagement beyond Academic Activity Duration &The evaluation of grading system should be worked out
	Total		14	2	14	23	26+4**	

B.Tech. Course Structure 1st Year II Semester

S.No.	Course	Cat.	L	T	P	Credits	Contact hours	Remarks
1.	Physics (PHN12502)	**CEF** (EE)	2	1	2	4	5	Branch Specific Physics and Chemistry Courses (Alternatively in each semester)
2.	Mathematics-II (MAN12106)	**CEF** (EE)	3	1	0	4	4	Branch specific Mathematics Course
3.	Introducing to Artificial Intelligence and Machine Learning* (CSN12601)	**PCE**(EE)	2	0	2	3	4	Common course for all branches *As per the clause23.13 of the NEP 2020
4.	Core Engineering Supporting Course^{s} (Flexible L-T-P) Data Structures * (CSN12401)	**CES** (EE)	2	0	2	3	4	Courses to be floated by each department ^{S}Only for the students of other branches (***maximum 180 students**).
5.	Core Engineering Essentials Courses-III (Flexible L-T-P) Data Structures (CSN12101)	**CEE** (CES)	2	0	2	3	4	Branch specific course for the students of that branch only
6.	Programming Paradigms (CSN12602)	**PCE** (CES)	2	0	0	2	2	Common course (Alternatively in each semester) $^{\#}$If any department does not want to adopt this/these course(s) for the specific branch, the department may float branch-specific courses in its/their place
7.	Extra Academic Activity-B/ Extra Academic Activity-A (EAN12700-EAN12710)	**EAA** (SA)	0	0	4	2$^{\&}$	4**	Common course for all branches (with different titles) **Engagement beyond Academic Activity Duration $^{\&}$The evaluation of grading system should be work out
	Total		14	2	12	21	23+4**	

Note: Flexible L-T-P structure indicates the course may have any combination of L-T-P with assigned credits as listed above.

B.Tech. Course Structure 2nd Year SEMESTER-III

S.No.	Cat.	Course & Course code	L	T	P	Credits	Contact hours	Remarks
1	CES	Analysis of Algorithms (CSN13400)	4	0	2	5	6	
2	CES	Computer Architecture (CSN13401)	4	0	0	4	4	
3	PCE	Operation Research (CSN13600)	3	0	0	3	3	
4	CEE	Object Oriented Programming (CSN13101)	2	0	2	3	4	
5	CES	Automata Theory (CSN13402)	4	0	0	4	4	
6	EAA	Extra Academic Activity-B-II (EAN13700-EAN13710)	0	0	4	2	4	
					Total =	21		

B.Tech. Course Structure 2nd Year SEMESTER-IV

S.No.	Cat.	Course	L	T	P	Credits	Contact hours	Remarks
1	CES	Microprocessors & Its Application (CSN14400)	4	0	2	5	6	
2	CES	Operating System (CSN14401)	4	0	2	5	6	
3	CEE	Object Oriented Modelling (CSN14101)	4	0	0	4	4	
4	CES	Compiler Construction (CSN14402)	3	0	0	3	3	
5	CES	Artificial Intelligence (CSN14403)	4	0	2	5	6	
6	EAA	Extra Academic Activity-B-II (EAN14700-EAN14710)	0	0	4	2	4	
7	MN	Minor Course@	4	0	0	4@	4@	
					Total =	24@		

B.Tech. Course Structure 3rd Year SEMESTER-V

S.No.	Cat.	Course	L	T	P	Credits	Contact hours	Remarks
1	CES	Computer Networks (CSN15400)	4	0	2	5	6	
2	CES	Database Management System (CSN15401)	4	0	2	5	6	
3	CES	Embedded Systems (CSN15402)	4	0	2	5	6	
4	CEE	Software Engineering & Project Management (CSN15101)	4	0	0	4	4	
5	CES	Cryptography & Network	4	0	0	4	4	

		Security (CSN15403)						
6	MN	Minor Course@	4	0	0	4@	4@	
7	HN	Honours Course#	4	0	0	4#	4#	
8	RS	Research Course$	4	0	0	4$	4$	
					Total =	23@#$		

B.Tech. Course Structure 3rd Year SEMESTER-VI

S.No.	Cat.	Course	L	T	P	Credits	Contact hours	Remarks
1	CEE	Wireless and Mobile Networks (CSN16101)	4	0	0	4	4	
2	CEE	Data Mining and Warehousing (CSN16102)	4	0	2	5	6	
3	CES	Distributed Systems (CSN16400)	4	0	2	5	6	
4	CEE	Professional Elective-I (CSN16250-CSN16255)	3	0	0	3	3	
5	CES	Professional Elective-II (CSN16401-CSN16405)	3	0	0	3	3	
6	MN	Minor Course@	4	0	0	4@	4@	
7	HN	Honours Course#	4	0	0	4#	4#	
8	RS	Research Course$	4	0	0	4$	4$	
					Total =	20@#$		

B.Tech. Course Structure 4th Year SEMESTER-VII

S.No.	Cat.	Course	L	T	P	Credits	Contact hours	Remarks
1	CEE	Image Processing and Computer Vision (CSN17101)	4	0	2	5	6	
2	CES	Formal Methods (CSN17400)	4	0	0	4	4	
3	PCE	Machine Learning with Python (CSN17600)	3	0	2	4	5	
4	CEE	Professional Elective-III (CSN17250-CSN17259)	3	0	0	3	3	
5	CEE	Mini Project (CSN17351)	0	0	0	6	0	
6	MN	Minor Course@	4	0	0	4@	4@	
7	HN	Honours Course#	4	0	0	4#	4#	
8	RS	Research Course$	4	0	0	4$	4$	
					Total =	22@#$		

B.Tech. Course Structure 4th Year SEMESTER-VIII

S.No.	Cat.	Course	L	T	P	Credits	Contact hours	Remarks
1		Industrial Training/Group Project/Research Projects (CSN18351)				14		

Semester	Total Credits
I	23
II	21
III	21
IV	24
V	23
VI	20
VII	22
VIII	14
GRAND TOTAL	**168**

@: A student can opt for one or two Minor courses in IV, V, VI and VII semesters as per the Minor courses offered by the other departments. Total credits in these semesters will be based on the credits of the Minor course(s) opted by the student in a particular semester. In this case the degree of the student will be B.Tech. (Computer Science & Engineering) with Minor in ( )

#: 1. A student can opt for one or more Honours courses in V, VI and VII semesters as per the Honours courses offered by the department. Total credits in these semesters will be based on the credits of the Honours course(s) opted by the student in a particular semester. In this case the degree of the student will be B.Tech. (Computer Science & Engineering) with Honours

2. A student, opting for Honours registration, is required to complete a total of 16 credits of the Honours courses offered by the department of Computer Science & Engineering. Honours courses will be from any of the M.Tech. programme of Department of Computer Science & Engineering.

$: 1. A student can opt for one or more research courses in V, VI and VII semesters as per the Research courses offered by the department. Total credits in these semesters will be based on the credits of the Research course(s) opted by the student in a particular semester. In this case the degree of the student will be B.Tech. (Computer Science & Engineering) with Research. In VII semester the research course will be a research project which may continue in VIII semester as Major Project.

2. A student, opting for Research registration, is required to complete a total of 16 credits of the Research courses offered by the department. Research courses will be from any one of the M.Tech. programme of Department of Computer Science & Engineering.

Professional Elective I - Core Engineering Essentials (CEE) Elective Courses	
Sno.	**Subject Name**
1.	Design Pattern (CSN16250)
2.	Neural Network (CSN16251)
3.	SOSE(Service Oriented Software Engg.) (CSN16252)
4.	XMl Based Applications (CSN16253)
5.	Real Time Systems (CSN16254)
6.	E-Commerce (CSN16255)

Professional Elective II-Core Engineering Supporting (CES) Elective Courses	
Sno.	**Subject Name**
1.	Data Compression (CSN16401)
2.	Functional Programming (CSN16402)
3.	Genetic Algorithm (CSN16403)
4.	Network Administration (CSN16404)
5.	IoT : Architecture & Protocol (CSN16405)

Professional Elective III (Core Engineering Essentials Elective Courses)	
Sno.	**Subject Name**
1	Distributed & Parallel Algorithms (CSN17250)
2	Gaming and Animation (CSN17251)
3	Information Retrieval (CSN17252)
4	Pattern Recognition (CSN17253)
5	Semantic Web (Web Ontology) (CSN17254)
6	Software Metrics & Quality Assurance (CSN17255)
7	Software Testing (CSN17256)
8	Theory of Virtualization (CSN17257)
9	Web Mining (CSN17258)
10	Cloud & Edge Computing (CSN17259)

Minor courses offered by the Department of Computer Science & Engineering

S.No.	Cat.	Course	Credits	Contact hours	Remarks
1.	CEE	Computer Organization CSN14001	4	4	
2.	CEE	Object Oriented Programming CSN14002	4	4	
3.	CES	Analysis of Algorithms CSN15001	4	4	
4.	CES	Operating System CSN15002	4	4	
5.	CES	Database Management System CSN16001	4	4	
6.	PCE	Machine Learning with Python CSN17001	4	4	

Honors courses offered by the Department of Computer Science & Engineering

S.No.	Cat.	Course	Credits	Contact hours	Remarks
1.	CEE	Network Security CSN15003	4	4	
2.	CES	Network Protocol Design & Implementation CSN15004	4	4	
3.	CEE	Advanced Computer Network CSN16002	4	4	
4.	CES	Software Defined Networking CSN16003	4	4	
5.	CES	Data Center Networking CSN16004	4	4	
6.	CEE	Distributed Ledger Technology CSN17002	4	4	
7.	CES	IOT Security & Trust CSN17003	4	4	
8.	CEE	IOT, ML and Deep Learning CSN17004	4	4	

Research courses offered by the Department of Computer Science & Engineering

S.No.	Cat.	Course	Credits	Contact hours	Remarks
1.	CES	Introduction to Bioinformatics and Genomics CSN15005	4	4	
2.	CEE	Soft Computing CSN15006	4	4	
3.	CEE	Data Security and Application Security CSN16005	4	4	
4.	CEE	Data Analytics CSN16006	4	4	
5.	CES	High Performance Computing CSN16007	4	4	
6.	CEE	Big Data CSN17005	4	4	
7.	CES	Medical Image Processing CSN17006	4	4	

Syllabus
Semester-I

COMPUTER PROGRAMMING
Syllabus

Prerequisite: NIL

L-T-P: 2-0-2, Credits**: 3Type:** Core Essential Subjects (CES)

Course Objectives

Students undergoing this course are expected to:	
1	To explore the basic understanding of how a computer works in general.
2	To explore how to solve a computational problem using a computer via writing a program inC programming language.
3	To explore the basic concept of logic thinking, and how to represent a logic using Algorithm and Flowchart.
4	To explore the basic syntax, such as variables, operators, conditionals and loops of C programming language.
5	To explore arrays, pointers, strings and functions in C.
6	To explore how to create user-defined data types using structure and unions.
7	To explore how to operate on files in C.
8	To explore the underlying workings of various macros, and how macro-processor processesthe macros.

Course Outcomes

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO1	To understand the basics of working of a computer.	Understanding
CO2	To obtain the basic knowledge about how to think logic given a computational problem, represent that via Algorithms and Flowchart, and implement the logic in C.	Understanding
CO3	To have an understanding of concepts of C syntax and semantics	Analyzing
CO4	To understand and apply the concept of Pointers and Functions	Applying
CO5	Understand how to operate on files.	Understanding

Mapping Course Outcomes with Program Outcomes

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	2	2	1	1	2					2
CO2	3	3	2	2	1	2				
CO3	2	1	1	3	2	2			2	1
CO4	3	1	3	3	1	2				2
CO5	1	2	1		2					
CO (Average) Course Average	**2.2**	**1.8**	**1.6**	**1.8**	**1.6**	**1.2**	**-**	**-**	**0.4**	**1**

3-High, 2-Medium, 1-Low

Syllabus

Unit 1: Working of a computer – Data representation in various bases, Binary System, Floating-point representation, basic components of computer architecture, operating system, and how they work together, A brief history of programming languages up to C, Stages of a compilation process, An overview of logic thinking – Algorithm, Flowchart.

Unit 2: Basic Syntax of C – Structure of a C Program, Printing on a screen, escape sequences, variables, operators, writing arithmetic and logic expressions, printing variables, format specifiers, reading from a console. **Conditional statements –** if, else, else if statements. Writing conditions. Nested conditional statements. **Loops –** for, while, do-while loops and their various formats, infinite loops, break, continue. Nested loops and conditional statements. **Switch –** switch-case statements and their use. Thinking and implementation of logics involving conditional, loops, switch-case. Patterns.

Unit 3: Functions – Syntax of C functions, parameters, arguments, return statements. **Array –** Single-dimensional, multi-dimensional arrays, accessing and updating array elements. **Pointers –** Memory address of a variable and concept of pointers. Single-dimensional and multi-dimensional pointers. Arrays as pointers. Dynamic Memory Management – malloc, calloc, realloc, free. Concept of heap and stack segments. Passing and returning arrays and pointers to and from a function. Call by value, call by reference. **Strings –** Reading and Writing strings. Allocating strings dynamically. String functions.

Unit 4: Structures and Unions – User-created data types, Structure, Union, accessing and updating their members, pointers to structure and union. **Storage class specifiers** – auto, extern, static, register. **Macros – #**include, #define, #ifdef, #ifndef, #endif. **Files –** Opening a file, opening modes, reading from a file, writing to a file: fopen, fread, fwrite, fprintf, fscanf, fseek, fclose.

Text Books

1. Yashavant Kanetkar, “Let us C”, 17th edition, BPB Publications.
2. R. J. Dromey, “How to solve it by Computer”, Pearson India.
3. Herbert Schield, “Complete Reference in C”, 4th edition, TMH.
4. Yashavant Kanetkar, “Understanding Pointers in C”, 3rd Edition, BPB Publication.
5. K. N King, C Programming: A Modern Approach, Second Edition.

Reference Books

1. Brian W. Kernighan, Dennis M. Ritchie, “The C Programming Language”, 2nd edition, Prentice Hall.

Computer Organization
Syllabus

Prerequisite: Basic knowledge prior to digital electronics, Basic functional units of a computer system
L-T-P: 3-0-0, Credits**:** 3**Type:** Core Essential Subjects (CES)

Course Objectives

Students undergoing this course are expected to:	
1	This course will discuss the basic concepts of computer architecture and organization that can help the participants to have a clear view as to how a computer system works. Examples and illustrations will be mostly based on a popular Reduced Instruction Set Computer (RISC) platform.
2	This course qualitatively and quantitatively examines computer design trade-offs and teaches the fundamentals of computer architecture and organization, including CPU, memory, registers, arithmetic unit, control unit, and input/output components
3	Understand the fundamentals of computer architecture.
4	Design and implement single-cycle and pipelined data paths for a given instruction set architecture.
5	Understand the performance trade-offs involved in designing the memory subsystem, including cache, main memory and virtual memory.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Study of the basic structure and operation of a digital Computer system.	Identify/Knowledge
CO-2	Analysis of the design of arithmetic & logic unit and understanding of the fixed point and floating-point arithmetic operations.	Analyzing
CO-3	Implementation of control unit techniques and the concept of Pipelining	Applying
CO-4	Understanding the hierarchical memory system, cache memories and virtual memory	Analyzing/understanding
CO-5	Understanding the different ways of communicating with I/O devices and standard I/O interfaces	Understanding

Mapping Course Outcomes with Program Outcomes

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	2	1	1	2					2
CO2	3	3	2	2	1	2				
CO3	2	1	1	3	2	2			2	1
CO4	3	1	3	3	1	2				2
CO5	1	2	1		2					
CO (Average) Course Average	**2.2**	**1.8**	**1.6**	**1.8**	**1.6**	**1.2**	**-**	**-**	**0.4**	**1**

3-High, 2-Medium, 1-Low

UNIT 1: Introduction: Functional units of digital system and their interconnections, buses, bus architecture, types of buses and bus arbitration. Register, bus and memory transfer. Processor organization, general registers organization, stack organization and addressing modes.

UNIT II: Arithmetic and logic unit: Look ahead carries adders. Multiplication: Signed operand multiplication, Booths algorithm and array multiplier. Division and logic operations. Floating point arithmetic operation, Arithmetic & logic unit design. IEEE Standard for Floating Point Numbers.

Unit III: Control Unit: Instruction types, formats, instruction cycles and sub cycles (fetch and execute etc), micro operations, execution of a complete instruction. Program Control, Reduced Instruction Set Computer, Pipelining. Hardwire and micro programmed control: micro programme sequencing, concept of horizontal and vertical microprogramming.

Unit IV: Memory: Basic concept and hierarchy, semiconductor RAM memories, 2D & 2 ½ D memory organizations. ROM memories. Cache memories: concept and design issues & performance, address mapping and replacement Auxiliary memories: magnetic disk, magnetic tape and optical disks Virtual memory: concept implementation.

UNIT V: Input / Output: Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. Modes of Data Transfer: Programmed 1/0, interrupt initiated 1O and Direct Memory Access., I/O channels and processors. Serial Communication: Synchronous & asynchronous communication, standard communication interfaces.

Text Books

1. Computer System Architecture - M. Mano
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky Computer Organization, McGraw-Hill Fifth Edition, Reprint 2012
3. John P. Hayes, Computer Architecture and Organization, Tata McGraw Hill, Third Edition, 1998, Reference books
4. William Stallings, Computer Organization and Architecture-Designing for Performance, Pearson Education, Seventhedition, 2006.

Reference Books

1. Behrooz Parahami, “Computer Architecture”, Oxford University Press, Eighth Impression, 2011.
2. Fundamentals of Microprocessors & Microcontrollers by B.RAMDHANPAT RAI PUBLICATIONS, NEW DELHI
3. David A. Patterson and John L. Hennessy, “Computer Architecture-A Quantitative Approach”, Elsevier, a division ofteed India Private Limited, Fifth edition, 20121
4. Swati Saxena Computer Organization, Dhanpat Rai Publications, NEW DELHI

Discrete Mathematics
Syllabus

Prerequisite*:* Basic of Mathematics
L-T-P: 2-0-0, **Credits: 2 Type:**EngineeringEssentials (EE)

Course Description

This course is aimed at Computer Science majors who have never taken any type of mathematical theory courses before, though it is also a useful course for developing general reasoning and problem-solving skills. For those that continue studying Computer Science, this course serves as excellent preparation for the required course. However, all students taking this course should benefit by improving their reasoning and abstract thinking skills, learning how to construct sound, logical arguments, and learning to detect flaws in unsound arguments.

Course Outcomes

Students undergoing this course are expected to:		**Bloom's taxonomy**
CO-1	To learn the expression of mathematical properties formally via the formal language of propositional logic, predicate logic and various proofing methods.	Understanding/ Comprehension
CO-2	Understand set operations, various types of relations and their representations, solving recurrence relations and also be able to verify simple mathematical properties that these objects possess.	Understanding /Analyze
CO-3	Understand various types of graphs, paths, spanning trees, planarity of graphs and coloring theorems. Graph Theory application in real world scenario	Understanding Apply
CO-4	Comprehend the discrete structures of lattices,	Understanding/ Comprehension:
CO-5	Recognize Algebraic structures; Groups, Subgroups, Rings, Fields with extension to concepts of vector spaces, dimensions, linear transformations. Applications of algebraic structure in Cryptographic algorithms, Network management, and various algorithms.	Evaluate

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Syllabus

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**
CO1	3	1	3	2	3	1	-	-	-	-
CO2	3	1	3	2	3	1	-	-	-	-
CO3	3	1	3	2	3	1	-	-	-	-
CO4	3	1	3	2	3	1	-	-	-	-
CO5	3	1	3	2	3	1	-	-	-	-
CO (Average) Course Average	**3**	**1**	**3**	**2**	**3**	**1**	**0**	**0**	**0**	**0**

UNIT 1: Formal logic

Simple and compound statement. logical operators. Implication and double implication, Tautologies and contradictions. Valid arguments and fallacy. Propositional functions and quantifiers. Notion of proofs.

UNIT 2: Set, Relations and Functions

Different types of relations, their compositions and inverses. Different types of functions, Recursively defined functions, Recursive algorithms, generating functions and solutions of recurrence relations, Complexity of algorithms, Big-o notation, Euclidean algorithm for finding GCD, Evaluation of polynomial using Horner's method, Russian Peasant method for multiplication. Generating Functions and related issues Recurrence relations and Generating, Functions.

UNIT 3: Graph Theory

Graphs and their basic properties – degree, path, cycle, subgraph, isomorphism, Eulerian and Hamiltonian walk, Matrix representation of Graphs and properties, Planar Graphs, Homeomorphism, Kuratowski's theorem, Spanning trees, shortest spanning tree, Algorithms for finding shortest spanning tree Graph colorings. Four color problem, Digraphs and related definitions, connectivity in diagraphs.

UNIT 4: Lattice and Boolean Algebra

Ordered Sets and Lattices: Partial order relations and Hasse diagram, Supremum and infimum, total ordering, lattices – bounded, distributive, complemented, modular.

UNIT 5: Group Theory and related issues

Algebraic System and Group Theory, Rings; Integral Domains, Division Rings, Fields related issues.

Text Books

1. Rosen: Discrete Mathematics and Its Applications Seventh Edition 7th Edition by Kenneth Rosen, McGraw Hill.
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2010/

Reference book

1. Manohar: Discrete Mathematical Structure with Application to Computer Science", J.P Trembley, & R. Manohar.
2. Schaum's DM: Lipschutz S. Schaum's Outlines of Theory and Problems of Discrete Mathematics., 2016. Schaum's Abstract Algebra: Jaisingh LR, Ayres F. Schaum's Outline of Abstract Algebra. McGraw
3. Levin, Oscar. "Discrete mathematics: An open introduction." (2021).
4. Epp, S. S. (2010). Discrete mathematics with applications. Cengage learning.

Syllabus
Semester-II

Prerequisite: NIL
COURSE CODE: CSN11601,
Type: Core Essential Subjects (CES)

Course Description

Course Overview:

Artificial Intelligence & Machine learning has revolutionized various industries by enabling computers to learn from data and make intelligent decisions. This course, "**Introduction to Artificial Intelligence and Machine Learning**," is designed to provide participants with a comprehensive understanding of AI and machine learning concepts and practical skills in building machine learning models using Python. By the end of this course, participants will have the knowledge and hands-on experience required to apply machine learning techniques using Python to solve real-world problems and make data-driven decisions.

Course Outcomes:

Sr. No.	CO Number	Course Outcome	Bloom's Taxonomy
1	CO1	Understand the basics of programming languages for problem-solving.	Understanding
2	CO2	Understand data storage, processing, and its use in model evaluation	Understanding
3	CO3	Explain the basics of AI and its applications.	Analyzing
4	CO4	Apply the machine learning algorithms for theclassification and regression problems.	Applying
5	CO5	Develops the ability to analyze problems and theirsolutions using clustering-based approaches	Understanding

CO-PO Mapping:

CO\PO	PO1	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	**3**	**3**	**2**	**2**	**3**	**1**	**-**	**-**	**-**	**-**	**-**
CO2	**2**	**3**	**3**	**3**	**2**	**2**	**-**	**-**	**-**	**-**	**-**
CO3	**2**	**2**	**2**	**2**	**3**	**-**	**1**	**-**	**-**	**-**	**-**
CO4	**3**	**3**	**3**	**3**	**3**	**2**	**1**	**-**	**-**	**-**	**-**
CO5	**3**	**3**	**3**	**3**	**3**	**2**	**1**	**-**	**-**	**-**	**-**
Average	**2.6**	**2.8**	**2.6**	**2.6**	**2.8**	**1.4**	**0.6**	**-**	**-**	**-**	**-**

Syllabus

Unit-1: Fundamentals of Programming: Basics of Programming Languages, Algorithms, Data types, Expressions and Operators, Control Statements, Iterations, and Functions.

Unit-2: Programming Constructs: Data Import and Export, Operations on Data: Traversing, Searching, and Arranging the data, String Operations, Mathematical operations on arrays.

Unit 3: Introduction to Artificial Intelligence: Evolution of AI, Various Approaches to AI, Intelligence and Machines, Intelligent Agents, Solving Problems by Searching, Considering the uses of AI for Society.

Unit-4:Introduction to Machine Learning: What and Why? Types of Machine Learning:Supervised, Unsupervised and Reinforcement learning, Under-fitting vs Over- fitting Problem, Training, Testing and Validation Process,Applications of Machine Learning, Linear Regression, Naïve Bayes Classifier, K-Means.

Text Books:

1. Russell, Stuart, and Peter Norvig. "Artificial intelligence: a modern approach."(2002).

2. Mueller, John Paul, and Luca Massaron. *Artificial intelligence for dummies*. John Wiley & Sons, 2021.

Reference Books:

1. Alpaydin, Ethem. *Introduction to machine learning*. MIT Press, 2020. Michalski, Ryszard Stanislaw, Jaime Guillermo Carbonell, and Tom M. Mitchell, eds. *Machine learning: An artificial intelligence approach*. Springer Science & Business Media, 2013

DATA STRUCTURES
Syllabus

Prerequisite: C programming and Basic of Mathematics.
L-T-P: 2-0-2, Credits: 3 **Type: Engineering Essential/ Core Essential**

Course Description

This course introduces the student's fundamentals of data structures and takes them forward to software design along with the course on Algorithms. It details how the choice of data structures impacts the performance of programs for given software application. This is a precursor to DBMS and Operating Systems. A lab course is associated with it to strengthen the concepts.

Course Objectives

Students undergoing this course are expected to:	
1	Give Knowledge of elementary Data organization, Complexity, Revision of Programming concepts
2	Give Basic& Advanced Knowledge of Array, Stack ,Queue& Linked List
3	Give Basic & Advanced Knowledge of Tree
4	Give Basic & Advanced Knowledge of Searching and Sorting
5	Give Basic & Advanced Knowledge of Tree Graph
6	Give Knowledge of elementary Data organization, Complexity, Revision of Programming concepts

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics**	**Bloom's taxonomy**
CO-1	Students Understood elementary Data organization, Complexity, Revision of Programming concepts	Identify/Knowledge
CO-2	Students Understood Basic &Advanced Knowledge of Array, Stack ,Queue& Linked List	Understanding /Analyzing
CO-3	Students Understood Basic Advanced Knowledge of Tree	Understanding
CO-4	Students Understood Basic& Advanced Knowledge of Searching and Sorting	Applying
CO-5	Students Understood Basic& Advanced Knowledge of Tree Graph	Understanding/Comprehension

Mapping Course Outcomes with Program Outcomes

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	3	2	3	2	2	0	0	1	2	0
CO2	2	2	3	3	1	0	1	1	2	1
CO3	1	3	3	2	2	0	2	2	1	1
CO4	2	3	3	3	2	1	1	1	1	1
CO5	2	2	3	3	2	1	1	1	1	1
CO (Average) Course Average	**2**	**2.4**	**3**	**2.6**	**1.8**	**0.4**	**1**	**1.2**	**1.4**	**0.8**

3-High, 2-Medium, 1-Low

Syllabus

UNIT-1: ***Introduction:*** Basic Terminology, Elementary Data Organization, Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic notations: Theta, Big-O, and Omega, Time-Space trade-off. Abstract Data Types (ADT)

UNIT-II:***Arrays***: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Application of arrays, Sparse Matrices and their representations.

Linked Lists: Array Implementation and Dynamic Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition, Generalized Linked List

Stacks: Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Recursion, Tower of Hanoi Problem, Simulating Recursion, Principles of recursion, Tail recursion, Removal of recursion

Queues: Abstract Data Type, Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Deque and Priority Queue.

Unit III:***Trees:*** Basic terminology, k-ary trees, Binary Trees, Binary Tree Representation: Array Representation and Dynamic Representation, Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Array and Linked Representation of Binary trees, Tree Traversal algorithms: In order, Preorder and Post order, Binary Search Trees, Threaded Binary trees, Traversing Threaded Binary trees, Forest, Huffman algorithm, Heap, B/B+ Tree, AVL tree

Unit IV: *Searching*& Sorting:

Sequential search, Binary Search, Comparison and Analysis

Internal Sorting: Bubble Sort, Selection Sort, Insertion Sort, Two Way Merge Sort, Heap Sort, Quick Sort Hashing

UNIT V: ***Graphs:*** Terminology, Sequential and linked Representations of Graphs: Adjacency Matrices, Adjacency List, Adjacency Multi list, Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Shortest Path algorithm: Dijikstra Algorithm

Text Books:

1. Aaron M. Tenenbaum, YedidyahLangsam and Moshe J. Augenstein "Data Structures Using C and C++", PHI

Reference Books:

1. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publication
2. Donald Knuth, "The Art of Computer Programming", vol. 1 and vol. 3.
3. Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with applications", McGraw Hill
4. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education
   Lipschutz, "Data Structures" Schaum's Outline Series, TMH

Programming Paradigms
Syllabus

Prerequisite: Computer Programming and Discrete Mathematics
L-T-P: 4-0-0, **Credits:** 4**Type:** Core Essential Subjects (CES)

Course Description

Students undergoing this course are expected to:	
1	The course is aimed at making the student familiar with the general concepts common to all programming languages so as to facilitate learning new languages.
2	Language paradigms (i.e., logic, functional, procedural, object-oriented) are compared and implementation strategies are discussed.
3	Presents examples of important programming languages and paradigms such as LISP, ALGOL, ADA, ML, Prolog, and C++. Students write sample programs in some of the languages studied.
4	The languages are used to illustrate programming language constructs such as binding, binding times, data types and implementation, operations (assignment data-type creation, pattern matching), data control, storage management, parameter passing, and operating environment
5	The model provided by a programming language to discuss concepts, formulate algorithms, and reason about problem solutions. Programming languages define models tailored to thinking about and solving problems in intended application areas

Course Objectives

Students undergoing this course are expected to

- Knowledge of, and ability to use, language features used in current programming languages.
- An ability to program in different language paradigms and evaluate their relative benefits.
- An understanding of the key concepts in the implementation of common features of programming languages.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics**	**Bloom's taxonomy**
CO-1	Study of the basics of Computer Programming Languages & Language Translation Issues:	Identify/Knowledge
CO-2	Study of the basics of the Data, Data Types, and Basic Statements ,Primitive Data Types, Array, Record & Union , Pointers & References etc.	Knowledge/Understanding/ Comprehension:
CO-3	Study of the Subprograms (Functions) & their Implementations	Applying/Comprehension
CO-4	Understanding the Object-Orientation, Concurrency, and Event handling	Analyzing/Applying
CO-5	Understanding the basics of the Functional and Logic Programming Languages With Introduction to Programming with functional Programming Languages	Creating/Evaluate

Mapping Course Outcomes with Program Outcomes:

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	1	2	3	2	2	3		1	2	2
CO2	2	2	2	1	2	2			2	2
CO3	2	2	2		2	2		2		2
CO4	2	2	3	3	2	3		2	1	3
CO5	2	2	2	2	2	2		1		3

CO (Average) Course Average	**1.8**	**2**	**2.4**	**1.6**	**2**	**2.4**	-	1.2	**1**	**2.4**

3-High, 2-Medium, 1-Low

Syllabus

UNIT 1: Introduction: Role of Programming Languages: Why Programming Languages, Towards Higher-Level Languages, Programming Paradigms, Programming Environments Language Description: Syntactic Structure, Language Translation Issues: Programming Language Syntax, Stages in Translation, Formal Translation Models

UNIT II: Data, Data Types, and Basic Statements: Names , Variables , Binding, Type Checking, Scope, Scope Rules , Lifetime and Garbage Collection, Primitive Data Types, Strings, Array Types, Associative Arrays ,Record Types, Union Types, Pointers and References, Arithmetic Expressions , Overloaded Operators, Type Conversions, Relational and Boolean Expressions, Assignment Statements, Mixed Mode Assignments, Control Structures, Selection ,Iterations, Branching, Guarded Statements

UNIT III: Subprograms and Implementations: Subprograms, Design Issues, Local Referencing, Parameter Passing, Overloaded Methods, Generic Methods, Design Issues for Functions, Semantics of Call and Return, Implementing Simple Subprograms, Stack and Dynamic Local Variables, Nested Subprograms, Dynamic Scoping.

UNIT IV: Object-Orientation, Concurrency, and Event handling: Grouping of Data and Operations — Constructs for Programming Structures, Abstraction Information Hiding, Program Design with Modules, Defined Types, Object Oriented Programming — Concept of Object, Inheritance, Derived Classes and Information Hiding — Templates, Semaphores, Monitors, Message Passing, Threads, Statement Level Concurrency Exception Handling

(Using C++ and Java as Example Language).

UNIT V: Functional and Logic Programming Languages: Fundamentals of Functional Programming Languages, Programming with ML, Introduction to Logic and Logic Programming — Introduction to Programming with HASKELL, SCHEME& SCALA

Text Books:

1. “Programming Languages: Design and Implementations” , Terrance Pratt, Marvin V. Zelkowitz, T.V.Gopa1, Fourth ed., Prentice Hall
2. “Programming in HASKELL” by Graham Hutton , Cambridge University Press
3. “ The Scheme Programming language” by R.Kent , Dybvig
4. “Programming in SCALA by Bill Venners & Martin Odersky , ARTIMA Press, Mount View, California

Reference Books:

1. Concepts of Programming Languages, Robert W. Sebesta, 10th Ed.,Pearson
2. Programming Language Design Concept”, David A. Watt, Willey India
3. “Programming languages: Concepts and Constructs”, Ravi Sethi, Second Ed.,Pearson.
4. “Types and programming Languages”, Benjamin C. Pierce. The MIT Press Cambridge, Massachusetts

Syllabus
Semester-III

Analysis of Algorithms
Syllabus

Prerequisite: Engineering Mathematics, Data Structures and Discrete Mathematics
L-T-P: 3-0-0, **Credits:** 3 **Type:** Core Essential Subjects(CES)

Course Description

Algorithms are recipes for solving computational problems. This course teaches techniques for the design and analysis of efficient algorithms for various applications, emphasizing methods useful in practice.

Course Objective

Students undergoing this course are expected to:	
1	To understand the problems in computing
2	To use/design the most efficient of the available/suitable techniques to solve these problems
3	To obtain efficiency of the algorithm is to be analyzed in the domains of time, space and energy
4	A lab course is also associated with it to strengthen the concepts.

Course Outcomes

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO-1	**Apply** knowledge of computing and mathematics for designing and implementing problem solving algorithms.	Applying
CO-2	**Analyze** the best, average and worst-case performance of the algorithms using asymptotic bounds.	Analyzing
CO-3	**Choose** and **apply** best suitable algorithm to find optimal solution of engineering problems	Creating/Applying
CO-4	**Demonstrate** a familiarity with various classes of algorithms and data structures.	Identify/Knowledge
CO-5	**Realize** the limits of computer algorithms via P, NP and NP-complete problems.	Identify/Knowledge

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	3	2	2	2	3	1	-	-	1	-
CO2	3	2	3	2	3	2	1	2	-	-
CO3	3	2	3	2	3	2	1		-	-
CO4	3	3	2	2	3	1	-	-	1	-
CO5	3	2	3	2	3	3	-	-	2	3
CO (Course Average)	**3.0**	**2.2**	**2.6**	**2.0**	**3.0**	**1.8**	**0.4**	**0.4**	**0.8**	**0.6**

UNIT 1: Sorting and complexity analysis: Time and Space Complexity, Different Asymptotic notations and their mathematical significance. Analysis of sorting algorithms of different complexity classes. Heaps, Heat Sort and Priority Queues.

Divide and Conquer Algorithms: Binary Search, Merge sort, Multiplication of Large Integers, Closest Pair, Strassen's Algorithm, Recurrences and Masters' Method, Quick Sort and Order Statistics.

UNIT II: Greedy Algorithms with Graph Recapitulation: Graph representation and Traversals, Minimum Spanning Trees and Shortest Paths in a Graph, Dial's Algorithm, Travelling Salesman Problem, Knapsack Problem, Job Sequencing Problem, Vertex Cover Problem, Maximum Network Flow, Huffman Coding and Encoding, Coin Change Problem, Discussion on Load Balancing Problem.

UNIT III: Dynamic Programming: Principle of Optimality, Cases where Greedy Algorithms Fails, 0/1 Knapsack, Coin Change Problem, Matrix Chain Multiplication, Optimal Binary Search Tree, Longest Common Subsequence, All-pairs Shortest Path Problem, Rod Cutting Problem, Memorization.

Backtracking: Hamiltonian Cycle, N-Queens Problem, Maze Problem, Sum of Subset Problem, Designing Generic Backtracking Algorithm.

UNIT IV: Number Theoretic Algorithms: The GCD, Modular Arithmetic, Chinese Remainder Theorem, Cyclic Groups, Linear Programming and Duality, Primality Testing.

Text Processing: Naïve String Matching, Rabin Karp Algorithm, String Matching using Finite Automata, KMP Algorithm.

UNIT V: NP-Completeness and Approximation algorithm:

P class, NP class, NP hard class, NP complete class – their interrelationship, Satisfiability Problem and Reducibility Examples. Necessity of Approximation Scheme, Polynomial Time Approximation with Examples.

Text Books

1. Cormen, Thomas H., et al. Introduction to algorithms. MIT press, 2009.
2. Horowitz, Ellis, SartajSahni, and Susan Anderson-Freed. Fundamentals of data structures. Vol. 20. Potomac, MD: Computer science press, 1976.
3. Sedgewick, Robert, and Philippe Flajolet. An introduction to the analysis of algorithms. Pearson Education India, 2013.

Reference Books

1. Levitin, Anany. Introduction to Design and Analysis of Algorithms, 2/E. Pearson Education India, 2008.
2. Kozen, Dexter C. The design and analysis of algorithms. Springer Science & Business Media, 1992.

Computer Architecture
Syllabus

Prerequisite: Digital Electronics and Computer Organizations
L-T-P: 4-0-0,   **Credits:** 4**Type:** Core Elective Subjects (CEL)

Course Description

The purpose of this course is to cultivate an understanding of modern computing technology through an in-depth study of the interface between hardware and software. In this course, you will study the history of modern computing technology before learning about modern computer architecture and a number of its essential features, including instruction sets, processor arithmetic and control, the Von Neumann architecture, pipelining, memory management, storage, and other input/output topics. The course will conclude with a look at the recent switch from sequential processing to parallel processing by looking at the parallel computing models and their programming implications.

Course Objectives

Students undergoing this course are expected to:	
1	Learn classes of computers, and new trends and developments in computer architecture
2	Learn about the performance metrics of microprocessors, memory, networks, and disks, pipelining, instruction set architectures, memory addressing
3	Understand the various techniques to enhance processors ability to exploit Instruction-level parallelism (ILP), and its challenges, exploiting ILP using dynamic scheduling, multiple issue, and speculation
4	Learn how data flow computer architectures works
5	State and understand memory hierarchy design, memory access time formula, performance improvement techniques, and tradeoffs, virtual memory and various cache mapping techniques and cache coherence problem.
6	understand the concept of Centralized shared memory architectures, Distributed shared memory architectures

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Students acquires basic knowledge of new trends and developments in computer architecture, performance metrics of microprocessors, memory, networks, and disks	Identify/Knowledge
CO-2	Students learn the concepts of parallelism in hardware/software and their mismatch.	Analyze Identify/Knowledge
CO-3	Will learn various techniques to enhance processors ability to exploit Instruction-level parallelism (ILP), and its challenges, exploiting ILP using dynamic scheduling, multiple issue, and speculation.	knowledge/ Analyzing/ Understanding
CO-4	Will get to know the working of data flow computer architecture	Applying/ Understanding
CO-5	Understand the concept of memory hierarchy design, memory access time formula, performance improvement techniques, and trade-offs, virtual memory and various cache mapping techniques and cache coherence problem.	Create/ Understanding
CO-6	Students learn the concepts of concept of Centralized shared memory architectures, Distributed shared memory architectures and their applications.	Understanding/ Applying

Mapping Course Outcomes with Program Outcomes

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	2	3	2		1		1		1	
CO2	3	2	2	1	2	3	2			1
CO3	2	2	3	3	2	3	1		1	
CO4	3	2	3	1	2	3	1		1	
CO5	2	3	3	2	2	3			2	
CO6	3	1	2	3	2	3			2	1
CO (Average) Course Average	**2.5**	**2.1**	**2.5**	**1.6**	**1.8**	**2.5**	**.83**		**1.16**	**0.33**

3-High, 2-Medium, 1-Low

Syllabus

UNIT 1: Introduction, History of Computing,

UNIT II: Fundamentals of computer Design, Performance related issues- Performance Parameters Measuring Performance- Instruction Set Architecture Design – compiler related issues.

UNITIII: Pipeline hazards- Overcoming hazards- Instruction set design and pipelining- Parallelism Concepts – Dynamic Scheduling – Dynamic hardware branch prediction.

UNIT IV: Multi-core, Super scalar, VLIW and vector processors – compiler support for ILP –extracting parallelism – speculation – performance.

UNIT V: Centralized shared memory architectures, Distributed shared memory architectures – synchronization – memory organization and cache coherence issues.

Text Books:

1. Advanced Computer Architecture: Parallelism, Scalability and Programmability by Kai Hwang
2. Computer Organization and Design, The Hardware/Software Interface by Patterson and Hennessey,
3. Advanced Computer Architecture: A System Design Approach by Richard Y. Kain

Reference Books:

1. Microprocessor Architecture: From Simple Pipelines to Chip Multiprocessors by Jean-Loup Baer

Operation Research
Syllabus

Prerequisite: Basic Engineering Mathematics and Probability Theory
L-T-P: 4-0-0, **Credits:** 4**Type:** Core Essential Subjects (CES)

Course Description

This course module on Operations Research aims to introduce students to use quantitative methods and techniques for effective decisions–making; model formulation and applications that are used in solving business decision problems.

Course objective

Students undergoing this course are expected to:	
1	Understand some common operations research models and algorithms.
2	Refers to the science of informed decision making.
3	To provide rational basis for decision-making by analyzing and modeling complex situations, and to utilize this understanding to predict system behavior and improve system performance.
4	Applying OR involves problem formalization, model construction and validation; other activities include a computational part, analysis of solutions, arriving at conclusions, and implementation of the decision.
5	Use the concepts of mathematical modeling, statistical analysis and optimization techniques on operations research.
6	To understand the emphasis is on applications rather than the details of methodology.
7	To know that operation research act as a tool to the courses namely data mining, business intelligence and decision support systems.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Identify** and develop operational research models from the verbal description of the real system.	Identify/Knowledge
CO-2	**Analyze** any real-life system with limited constraints and depict it in a model form.	Analyzing
CO-3	**Understand** a variety of real-life problems like assignment, transportation, a traveling salesman, inventory management, project planning, etc.	Understanding/Comprehension
CO-4	**Design** and **solve** operational research models using appropriate methods.	Applying
CO-5	**Learn** to propose the best strategy in any organization using decision-making methods in the domains of cost, time, and benefit.	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	3	1	2	2	3	1	–	–	1	–
CO2	3	1	3	2	3	2	–	–	–	–
CO3	3	2	3	2	3	2	–	–	–	–
CO4	3	3	2	2	3	1	–	–	1	–
CO5	3	3	3	2	3	3	–	–	2	–
CO (Average) Course Average	**3.0**	**2.0**	**2.6**	**2.0**	**3.0**	**1.8**	**0.0**	**0.0**	**0.8**	**0.0**

Syllabus

UNIT 1: Introduction to OR Models, Graphical Method for LPP, Convex sets, Simplex Method, Big M Method, Two Phase Method, Multiple solutions of LPP, Unbounded solution of LPP, Infeasible solution of LPP, Revised Simplex Method, Case studies, Primal Dual Construction, Duality Principle, Primal-Dual relationship of solutions, Dual Simplex Method, Sensitivity Analysis.

UNIT II: The Transportation Model, Definitions, Formulation, North-West Corner Method, Vogel's Approximation Method, u-v method, Post optimality Analysis,

The Assignment Model: Definition and mathematical representation, The Hungarian method

Network Optimization Models: Sequencing and Scheduling (Critical Path Methods and PERT)

UNIT III: Dynamic Programming: Formulation, Optimal subdivision problem, Solution of L.P.P. by dynamic programming, applications, Deterministic and probabilistic dynamic programming.

UNIT IV: Integer Programming: Non-Linear Programming Constrained and Unconstrained optimization, Multi-variable unconstrained optimization, Kuhn-Tucker Conditions, Queuing Theory.

Unit V: Decision Theory, The Theory of Games, Pure and Mixed strategies Games, n-person zero sum games, Basic Queuing Models, Inventory Models, Simulation Modeling.

Text Books

1. Taha, Hamdy A. Operations research: an introduction. Pearson Education India, 2013.
2. Hillier, Frederick S., and Gerald J. Lieberman. "Introduction to Operations Research, Burr Ridge, IL." 10th Edition (2017)
3. Operations Research, P K Gupta and D S Hira, S. Chand and Company LTD. Publications, New Delhi – 2007
4. Operations Research, Theory and Applications, Sixth Edition K Sharma, Trinity Press, Laxmi Publications Pvt. Ltd. 2016.

Reference Books

1. Churchman, C. West, Russell L. Ackoff, and E. Leonard Arnoff. "Introduction to operations research." (1957).
2. Hillier, Frederick S. Introduction to operations research. Tata McGraw-Hill Education, 2012.

Object Oriented Programming

Prerequisite: Basics of Programming
L-T-P: 2-0-2, Credits: 3 **Type:** Core Elective Subjects (CEL)

Course Description

This is an introductory course, where students learn and practice essential programming skills using the Java programming language. This course provides an overview of Object Oriented Programming (OOP) concepts using Java/C++. It helps to understand basic OOP concepts and assist in applying these concepts. The principles behind OOP discussed. It covers object-oriented principles such as classes, objects, abstraction, composition, Inheritance, polymorphism, and interfaces. These concepts can be implemented in the Java language. Along the way, many of the Java library classes are seen that can be organized to solve a variety of problems. The Java collection classes are studied. Additional topics include exception handling, database connectivity with JDBC, and multi-threading. The course is programming intensive. By the end of this course student will able to understand the basics of OOP and be prepared to take on more complex challenges. A lab course is associated with it to strengthen the concepts.

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO-1	Understand the Basic concept of Object Orientation, object identity and Encapsulation.	Understanding/ Comprehension:
CO-2	Understand the Basic concept of Basic Structural Modeling.	Understanding/ Comprehension
CO-3	Know the knowledge of Object-oriented design, Object design.	Identify/Knowledge/Remember
CO-4	Know the knowledge of Basics of Java or C++.	Identify/Knowledge/Remember
CO-5	Understand the Basics of object and class in Java or C++.	Understanding Analyzing

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	2	-	-	1	-	-	-	2	-	-
CO2	2	2	-	1	-	-	-	-	-	-
CO3	2		-	1	-	-	-	-	-	-
CO4	2		-	1	-	-	-	2	-	-
CO5	1	1	2	1	-	-	-	1	-	-
CO (Average) Course Average	**1.8**	**0.6**	**0.4**	**1.0**	**0.0**	**0.0**	**0.0**	**1.0**	**0.0**	**0.0**

UNIT I: INTRODUCTION

The meaning of Object Orientation, object identity, Encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object-oriented modeling, Introduction to UML, conceptual model of the UML, Architecture.

UNIT II: BASICS OF JAVA OR C++

Core Java: Introduction to Object Oriented Software development through Java. Classes and Objects.

UNIT-III Abstraction, Inheritance, Polymorphism, Nested classes and interfaces, Exceptions, Strings, Packages, The I/O Package.

UNIT IV: Advanced Java: Event Handling, AWT, Swing, Applets, Multi-Threading, Generic, The collection frameworks, Introduction to socket programming, Java Database connectivity (JDBC).

TextBooks:

1. Kathy Sierra and Bert Bates, “Head First Java”, 2nd edition, O’Reilly
2. Herbert Schildt , “Java : The Complete Reference”, 9th edition, Oracle Press
3. Cay S. Horstmann and Gary Cornell , “Core Java Volume I & II”, 10th edition, Prentice-Hall
4. Tony Gaddis, “Starting Out with Java: From Control Structures through Objects”, 6th edition,Pearson
5. David Flanagan, “Java in a Nutshell”, 5th edition, O’Reilly

Automata Theory
Syllabus

Prerequisite: Discrete Mathematics
L-T-P: 3-0-0**, Credits:** 3 **Type:** Core Essential Subjects (CES)

Course Description

The theory of computation is the concepts of *automata, formal languages, grammar, algorithms, computability, decidability*, and *complexity*. Theory provides a simple, elegant view of the complex machine that we call a computer. Theory possesses a high degree of permanence and stability, in contrast with the ever-changing paradigms of the technology, development, and management of computer systems. Further, parts of the theory have direct bearing on practice, such as Automata on circuit design, compiler design, and search algorithms; Formal Languages and Grammars on compiler design; and Complexity on cryptography and optimization problems in manufacturing, business, and management

Course Objectives

Students undergoing this course are expected to:	
1	Introduce students to the mathematical foundations of computation including automata theory; the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability.
2	Enhance/develop students' ability to understand and conduct mathematical proofs for computation and algorithms.
3	This subject is very useful in order to develop a logical mind
4	Knowledge of Finite Automata theory will give you understanding of advanced concepts of this subjects like Pushdown Automata and Turing Machine
5	you will be approaching towards the end of this course you will be able to grasp all the concepts and logics that you need to further proceed deeper in this subject

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Learn and Identify the behavior of a system through finite state machine.	Identify/Knowledge
CO-2	Use knowledge of determinism, non-determinism, minimization, equivalence issues of Finite Automata	Analyze Identify/Knowledge
CO-3	Demonstrate the understanding of key notions, such as algorithm, computability, decidability, and complexity through problem solving	knowledge/Analyzing/Understanding
CO-4	Understand context-free grammars, properties of languages, grammars and Automata with rigorously formal mathematical method.	Applying/Understanding
CO-5	Interpret and design different types of PDA. Design Turing machine to solve problems and Explain the decidability and intractability of computational problem	Create/Understanding
CO-6	Apply Automata Theory concepts in engineering applications like designing of compilers	Understanding/Applying

Mapping Course Outcomes with Program Outcomes:

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	2	2	2	1	3	3		1		1
CO2	3	2	1	3	2	2				
CO3	3	3	2	3	2	3		1		
CO4	3	2	1	3	2	3				2
CO5	3	3	2	3	2	3		1		
CO6	2	3	3	2	2	1			2	
CO (Average) Course Average	**2.6**	**2.5**	**1.8**	**2.5**	**2.1**	**2.5**		**0.5**	**0.3**	**0.5**

3-High, 2-Medium, 1-Low

Syllabus

UNIT 1: Introduction, inductive Proofs Relations and Functions

UNIT II: Regular Languages DFA, NFA Machines and their equivalence, Regular Expressions, Equivalence of Regular Expressions and Finite State Machines, Closure Properties of Regular Languages Proving Non-Regularity

UNITIII: Context-free Languages Context-free Grammars, Derivations, Leftmost, Rightmost, Inherent Ambiguity, Parse Trees, Normal Forms, Proof of Containment of the Regular Languages Pushdown Automata, Equivalence of PDAs and Context free Grammars Closure Properties of Context-free Languages

UNIT IV: Pumping Lemma for both Regular & Context-free Languages, Proving Some Languages are not Context-free.

UNIT V: Recursive and Recursively Enumerable Languages, Turing Machines Definition of Recursive and Recursively Enumerable, Church's Hypothesis, Computable Functions, Methods for Turing Machine Construction

Text Books:

1. "Theory of Computer science "Automata Languages and Computation, Mishra and Chandrasekaran PHI.
2. "Switching & Finite Automata", ZVI Kohavi,2nd Edition., Tata McGraw Hill.
3. "Introduction to languages and the Theory of Computation", John C Martin, TMH 2.

Reference Books:

1. "Introduction to Automata Theory Language and Computation", Hopcroft and Ullman, Pearson Education.
2. "An Introduction to Formal Language and Automata", PETER LINZ, Narosa Publishing House.

Syllabus
Semester-IV

Micro Processors and Its Application
Syllabus

Prerequisite: Computer Organizations& Computer Programming
L-T-P: 3-0-2, **Credits:** 4**Type:** Core Essential Subjects (CES)

Course Description

1. Know the history of Microprocessor
2. Define Microprocessor
3. Know the internal architecture of a microprocessor
4. Explain different types of busses and basic architecture of computer

Course Objectives

Students undergoing this course are expected to:	
1	This course deals with the systematic study of the Architecture and programming issues of 8 bit 8085-microprocessor and interfacing with other peripheral ICs and co-processor
2	Learn the different types of serial communication techniques.
3	This subject deals about the basic 16-bit (8086) processor and an 8-bit (8051) controllers, their architecture, internal organization and their functions, interfacing an external device with the processors/ controllers.
4	Understand concepts of register transfer logic and arithmetic operations
5	An introductory course in microprocessor software and hardware: architecture, timing sequence, operation, and programming – and discussion of appropriate software diagnostic language and tools.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Understand the basic architecture of 8085.	Understanding /Knowledge
CO-2	Impart the knowledge about the instruction set of 8085 Microprocessor.	Understanding/ Comprehension
CO-3	Understand the basic idea about the data transfer schemes and its applications	Applying/Understand
CO-4	Develop skill in simple program writing for 8085 Microprocessor	Analyzing/Create
CO-5	Select and apply the pattern for 8085 Microprocessor.	Applying

Mapping Course Outcomes with Program Outcomes

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	1	2	2							
CO2		1	2	3	2					

CO3			1	3	2	1				
CO4		2	2		3			1		
CO5	2					3	2			1
CO (Average) Course Average	0.6	1.0	1.4	1.2	1.4	0.8	0.4	0.2		0.2

3-High, 2-Medium, 1-Low

Syllabus

UNIT-1: Introduction to Microprocessors & Microcomputers History and Evolution, types of microprocessors, Microcomputer Programming Languages, Microcomputer Architecture, Intel 8085 Microprocessor, Register Architecture, Bus Organization, Registers, ALU, Control section, Instruction set of 8085, Instruction format, Addressing modes, Types of Instructions.

UNIT-II: Assembly Language Programming and Timing Diagram Assembly language programming in 8085, Macros, Labels and Directives, Microprocessor timings, Micro instructions, Instruction cycle, Machine cycles, T-states, State transition diagrams, Timing diagram for different machine cycles.

UNIT III: Serial I/O, Interrupts and Comparison of Contemporary Microprocessors Serial I/O using SID, SOD. Interrupts in 8085, RST instructions, Issues in implementing interrupts, multiple interrupts and priorities, Daisy chaining, interrupt handling in 8085, Enabling, Disabling & masking of interrupts.

UNIT IV: Data Transfer techniques Data transfer techniques, Parallel & Programmed data transfer using 8155. Programmable parallel ports & handshake input/output, Asynchronous and Synchronous data transfer using 8251. PIC (8259), PPI (8255), DMA controller (8257).

UNIT V: Microprocessor Interfacing Techniques Interfacing Traffic Light Interface, Stepper Motor, 4 Digit 7 Segment LED , Elevator, Musical Tone Generator & 8 Channel 12Bit ADC with Multiplexor &A/D converters, D/A converters.

UNIT VI: Architecture of Typical 16-Bit Microprocessors (Intel 8086) Introduction to a 16 bit microprocessor, Memory address space and data organization, Segment registers and Memory segmentation, Generating a memory address, I/O address space, Addressing modes, Comparison of 8086 & 8088, Basic configurations of 8086/8088, Min. Mode, Max. Mode & System timing, Introduction to Instruction Set of 8086.

Text Books

1. Ramesh Gaonkar , Microprocessor Architecture, Programming, and Application with the 8085, Penram publications.
2. John Ufferbeck, The 8080/85 Family: Design, Programming & Interfacing, PHI India Microprocessor & Interfacing - Douglas Hall, TMH.
3. Savaliya M. T., 8086 Programming and Advance Processor Architecture, Wiley India.
4. Triebel& Singh, The 8088 and 8086 Microprocessors, , Pearson Education

Reference Books

1. J H Hennessy and D A Patterson, Computer Architecture: A Quantitative Approach.
2. Bhurchandi K and A K Ray, Advanced Microprocessor and Peripherals.
3. Brey, The Intel Microprocessors: 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium pro Processor.

Operating System
Syllabus

Prerequisite: Computer Programming, Data Structures and Computer Organization
L-T-P: 3-0-2, **Credits:** 4 **Type:** Core Essential Subjects (CES)

Course Description

In this course students will study the basic facilities provided in modern operating systems. The emphasis will be on understanding general concepts that are applicable to a wide range of operating systems, rather than a discussion of the features of any one specific system. However, for gaining in-depth knowledge the pedagogical operating systems such as XV6 can be used to demonstrate the working at code level. Topics that will be covered in the course include protected kernels, processes and threads, concurrency and synchronization, memory management, virtual memory, file systems, secondary storage, protection, and security.

Course Objectives

Students undergoing this course are expected to:	
1	gain extensive knowledge on principles and modules of operating systems
2	understand key mechanisms in design of operating systems modules
3	understand process management, concurrent processes and threads, memory management, virtual memory concepts, deadlocks
4	compare performance of processor scheduling algorithms
5	produce algorithmic solutions to process synchronization problems
6	use modern operating system calls such as Linux process and synchronization libraries

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics**	**Bloom's taxonomy**
CO-1	Demonstrate the knowledge of Operating Systems both at system call level and code level.	Identify/ Knowledge
CO-2	Compare and analyze the different implementation approach of system programming and operating system abstractions.	Creating/ Analyzing
CO-3	Formulate synchronization problem and develop the solution for same	Analyzing, Creating
CO-4	Interpret various OS functions used in different operating systems.	Applying
CO-5	Learning about real life applications of Operating System in every field.	Understanding/Comprehension

Mapping Course Outcomes with Program Outcomes

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	-	-	-	-	2	-	-	-	2
CO2	-	-	-	-	-	-	2	-	-	-
CO3	-	-	-	-	-	-	-	-	-	-
CO4	-	3	-	-	-	-	-	-	-	-
CO5	2	-	-	-	-	-	-	-	-	2
CO (Average) Course Average	**0.8**	**0.6**	**-**	**-**	**-**	**0.4**	**0.4**	**-**	**-**	**0.8**

3-High, 2-Medium, 1-Low

Syllabus

UNIT-1: Introduction and Overview
UNIT-II: Process fundamentals, scheduling, synchronization.
UNIT III: Inter-process communication, Deadlock.
UNIT IV: Memory management and virtual memory.
UNIT V: File system and secondary storage.
UNIT VI: Protection and security issues, Case studies e.g., Linux, Solaris and Android.

Text Books:

1. Operating Systems, by William Stallings.
2. Operating Systems Concepts by Silberschatz, Galvin, and Gagne.
3. The Design of the UNIX Operating System, by Maurice J. Bach.
4. Advanced Programming in the UNIX Environment, by W. R. Stevens & S. A.Rago.

Reference Books:

1. The Design and implementation of the 4.4 BSD UNIX operating system by Marshall Kirk McKusick, Keith Bostic, Michael J. Karels, John S. Quarterman.
2. xv6: A simple, Unix-like teaching operating system by Russ Cox, FransKaashoek and Robert Morris.

Object Oriented Modeling

Prerequisite: Basic Concepts of Object-Oriented Programming,
L-T-P: 3-0-0, **Credits: 3** **Type:** Core Elective Subjects (CEL)

Course Description

In this course students will study the fundamental principles of object-oriented approaches to modeling software requirements and design. Topics include strategies for identifying objects and classes of objects, specification of software requirements and design, the design of class hierarchies, software reuse considerations, graphical notations, system implementation using object-oriented and object-based programming languages, and comparison of object-oriented approaches to more traditional approaches based on functional decomposition.

Course Objectives

Students undergoing this course are expected to:	
CO1	Analyze and Design solutions for real world problems using Object Oriented methodology.
CO2	Create a requirements model using UML class notations and use-cases based on statements of user requirements, and to analyze requirements models given to them for correctness and quality.
CO3	Create the OO design of a system from the requirements model in terms of a high-level architecture description, and low-level models of structural organization and dynamic behavior using UML class, object, and sequence diagrams.
CO4	Comprehend enough Java to see how to create software that implements the OO designs modeled using UML.
CO5	Comprehend the nature of design patterns by understanding a small number of examples from different pattern categories, and to be able to apply these patterns in creating an OO design
CO6	Given OO design heuristics, patterns or published guidance, evaluate a design for applicability, reasonableness, and relation to other design criteria.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Analyze and Design** solutions for real world problems through the Object Oriented methodology.	Identify/Knowledge/Remember
CO-2	**Create** requirements model through the UML class notations, and also to analyze requirements models with the reference of their correctness and quality.	Creating
CO-3	**Create** the OO design of a system from the requirements model in terms of a high-level architecture description, and low-level models of structural organization and dynamic behavior using UML class, object, and sequence diagrams.	Creating
CO-4	**Comprehend** enough Java to see how to create software that implements the OO designs modeled using UML.	Understanding/ Comprehension.
CO-5	**Comprehend** the nature of design patterns by understanding a small number of examples from different pattern categories, and to be able to apply these patterns in creating an OO design	Analyzing
CO-6	Use OO design heuristics, patterns or published guidance, evaluate a design for applicability, reasonableness, and relation to other design criteria	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	3	3	-	-	-	-	-	-	-	-
CO2	-	-	2	3	-	-	-	-	-	-
CO3	-	-	-	-	3	2	-	-	-	-
CO4	2	3	-	-	-	-	-	-	-	-
CO5	-	-	-	-	3	3	-	-	-	-
CO6	-	-	-	-	2	3	-	-	-	-
CO (Average) Course Average	**0.83**	**1.0**	**0.33**	**0.5**	**1.33**	**1.33**	**0.0**	**0.0**	**0.0**	**0.0**

Syllabus

UNIT 1: Introduction to Object Orientation: Abstractions, Encapsulation, Interfaces & their Implementations, Abstract Data Types, Design by Contract, Classes, Implementation sharing & 'this' reference, Single inheritance, Dynamic Binding, Methods and Messages, Polymorphism,7 Sub-typing in Inheritance, \ Engineering Properties: building abstractions and hierarchies. 08

UNIT 2: Unified Modeling Languages (UML): Overview of Unified Modeling Language (UML), Static and Dynamic Models, Use-Case Diagrams, UML Class Diagrams, Object Diagram, Sequence Diagram, Activity Diagram, Collaboration Diagram, Deployment Diagrams. 08

UNIT 3: Object Oriented Analysis & Design: Identifying Relationships, Attributes, and Methods, Associations, Inheritance Relationships, A Part of Relationship-Aggregation, Class Responsibilities, Object Responsibilities. Object Oriented Design Processes, Design Properties. 08

UNIT 4: Logical Architecture: Packages, Subsystems, View Layer /UI Layer/Boundary Objects, Application Logic Layer/ Business Logic Layer /Domain Objects/ Control Objects, Access Layer/Technical Services Layer/Entity Objects 06

UNIT 5: Design Pattern and frameworks: Design Reuse, Creational Patterns with Example Applications, Structural Patterns with Example Applications, Behavioral Patterns with Example Applications, Application Frameworks. 10

TextBooks

1. Larman, Craig. Applying UML and patterns: an introduction to object oriented analysis and design and iterative development. Pearson Education India, 2012.
2. Gamma, Erich, et al. Design patterns: elements of reusable object-oriented software. Pearson Deutschland GmbH, 1995.
3. Blaha, Michael, and James Rumbaugh. Object-oriented modeling and design with UML. Pearson Education India, 2005.

Reference Books

1. Buschmann, Frank, et al. "Pattern-Oriented Software Architecture, Volume 1: A System of Patterns (1996)." Cited on: 20.
2. Booch, Grady, et al. "Object-oriented analysis and design with applications." ACM SIGSOFT software engineering notes 33.5 (2008): 29-29.

Compiler Construction
Syllabus

Prerequisite: Theory of Computation, Data Structures
L-T-P: 3-0-0, **Credits:** 3**Type:** Core Essential Subjects (CES)

Course Description

This course introduces the principles behind the construction of compilers, including automata, lexical analysis, syntactic analysis, constructing parser, translation, and code generation.

Course Objective

Students undergoing this course are expected to:	
1	To become aware about the principles behind the construction of compilers, including automata, lexical analysis, syntactic analysis, constructing parser, translation, and code generation.
2	To implement front end of the compiler by means of generating Intermediate codes.
3	To implement code optimization techniques.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Describe** the basic concepts of language translation, the compilation and its different phases.	Identify/Knowledge
CO-2	**Classify** the lexical, syntactic and semantic **analysis** into meaningful phases for a compiler to undertake language translation	Analyzing
CO-3	**Apply** the lexical, syntactic and semantic specifications to build lexical, syntax analyzer and semantic analyzer.	Applying
CO-4	**Describe** the specification of symbol table of a compiler and storage organization in run-time environment of a program.	Understanding/ Comprehension
CO-5	**Demonstrate** the intermediate code and machine code representations and its generation.	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	3	1	3	3	3	3	–	–	1	–
CO2	3	3	2	3	3	3	–	–	–	–
CO3	3	1	2	2	3	3	–	–	1	–
CO4	3	2	3	3	3	3	–	–	–	–
CO5	3	1	2	2	2	3	–	–	2	–
CO (Average) Course Average	**3.0**	**1.6**	**2.4**	**2.6**	**2.8**	**3.0**	**0.4**	**0.0**	**0.8**	**0.0**

Syllabus

UNIT I: Introduction: Compiler and Translators, Analysis-synthesis model of compilation, various phases of a compiler, Tool based approach to compiler construction.

Lexical Analysis: Interface with input, parser and symbol table, token, lexeme and patterns. Difficulties in lexical analysis. Error reporting. Implementation. Regular definition, Transition diagrams, LEX.

UNIT II: Syntax Analysis: CFGs, ambiguity, associativity, precedence, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing, bottom up parsing, operator precedence grammars, LR parsers (SLR, CLR, LALR),YACC.

Semantic Analysis: Inherited and Synthesized attributes, dependency graph, evaluation order, bottom up and top down evaluation of attributes, L- and S-attributed definitions, type system, type conversion, overloaded functions and operators.

UNIT III: Intermediate Representation and Symbol Table: Intermediate representations, issues, high-level, medium and low-level IR, three-address code, symbol table, data structure for symbol table, storage allocation information, global symbol table structure.

Intermediate Code Generation: Intermediate code generation, translation of declarations, assignments, control flow, Boolean expressions and procedure calls. Implementation issues.

UNIT IV: Runtime System: storage organization, activation tree, activation record, parameter passing, symbol table, dynamic storage allocation.

Code Generation: Code generation and Instruction selection, basic blocks and flow graphs, register allocation, code generation, dag representation of programs, code generation from DAGs, peephole optimization, code generator generators, specifications of machine.

Text Books

1. Aho, Alfred V., et al. Compilers: principles, techniques and tools. 2020.

2. Aho, Alfred V., et al. Compilers: Principles, Techniques, & Tools. Pearson Education India, 2007.

Reference Books

1. Cooper, Keith, and Linda Torczon. Engineering a compiler. Elsevier, 2011.

2. Andrew W. Appel, “Modern Compiler Implementation in ML”, Cambridge UniversityPress

3. Andrew W. Appel, “Modern Compiler Implementation in C”, Cambridge UniversityPress

Artificial Intelligence
Syllabus

Prerequisite: Engineering Mathematics, Programming Languages, Good Analytical Skills, Basic Knowledge of Statistics and Modeling.

L-T-P: 3-0-2, **Credits:** 4 **Type:** Core Essential Subjects (CES)

Course Description

This is an introductory course to explain the design of artificial intelligence and expert systems. Course starts with the explanation of search techniques used to solve the problems. Various search methods like DFS, BFS, A* and others are discussed in detail. Different methods (rule-based methods, probabilistic methods, optimization methods and learning techniques) are discussed in detail. Need of Designing Expert System is presented in detail.

Course Objectives

Students undergoing this course are expected to:	
1	Able to write artificial intelligence problems
2	Able to understand the concept of expert system
3	Able to apply these techniques for solving real life problems
4	knowledge of Artificial Intelligence concepts including the essentials of statistics required for Data Science, Python programming, and Machine Learning
5	A role in this domain places you on the path to an exciting, evolving career that is predicted to grow sharply into 2025 and beyond.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Describe the modern view of AI as the study of agents that receive percepts from the Environment and perform actions.	Identify/Knowledge
CO-2	Demonstrate awareness of informed search and exploration methods.	Analyze Identify/Knowledge
CO-3	Explain about AI techniques for knowledge representation, planning and uncertainty Management.	Knowledge/Analyzing/Understanding
CO-4	Develop knowledge of decision making and learning methods.	Applying/Understanding
CO-5	Describe the use of AI to solve English Communication problems	Create/Understanding
CO-6	Explain the concept of Knowledge Representation	Understanding/Applying

Mapping Course Outcomes with Program Outcomes:

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1		3		3						
CO2	2			3		2				
CO3				3	2	2	2			
CO4		2		3	3					
CO5	2			3		2				
CO6	1	2		2	1	3				
CO (Course Average)	**0.83**	**1.16**		**2.8**	**1**	**1.8**	**0.3**			

3-High, 2-Medium, 1-Low

Syllabus

UNIT-I: Introduction to AI: Intelligent agents – Perception –Natural language processing – Problem – Solving agents – Searching for solutions: Uniformed search strategies-Informed search strategies.

UNIT-II: KNOWLEDGE AND REASONING

Adversarial search – Optimal and imperfect decisions – Alpha, Beta pruning – Logical agents: Propositional logic – First order logic – Syntax and semantics – Using first order logic – Inference in first order logic.

UNIT III: UNCERTAIN KNOWLEDGE AND REASONING

Uncertainty – Acting under uncertainty – Basic probability notation – Axioms of probability – Bayes' rule – Probabilistic reasoning – Making simple decisions

UNIT IV: PLANNING AND LEARNING

Planning: Planning problem – Partial order planning – Planning and acting in non-deterministic domains – Learning: Learning decision trees – Knowledge in learning – Neural networks – Reinforcement learning – Passive and active.

UNIT V: EXPERT SYSTEMS

Definition – Features of an expert system – Organization – Characteristics – Prospector – Knowledge Representation in expert systems – Expert system tools – MYCIN – EMYCIN.

Text Books:

1. Stuart Russel and Peter Norvig 'Artificial Intelligence A Modern Approach', Second Edition, Pearson Education, 2003 / PHI.
2. Donald A. Waterman, '**A** Guide to Expert Systems', Pearson Education, Donald A. Waterman.

Reference Books:

1. *C.S. Krishnamoorthy, S. Rajeev*Artificial Intelligence and Expert Systems1996
2. I Gupta, G Nagpal, Artificial Intelligence and Expert Systems,2020

Syllabus
Semester-V

Computer Networks
Syllabus

Prerequisite: C or Java Programming, Course in Algorithms, Course in Probability, Operating System
L-T-P: 3-0-2, **Credits:** 4**Type:** Core Essential Subjects (CES)

Course Description

This course covers the concept of computer networks with the context of Internet. It requires the prior knowledge communication foundations, computer organization, basic algorithms, data structures and C programming. This course also covers the fundamental principles, elements, and protocols of computer networks. The computer networks course investigate the working of different protocols, need of the various protocols, and their performance trade-offs. This course prepares foundations for wireless networks and distributed systems. A lab course is also associated with the course of computer networks.

Course Objectives

Students undergoing this course are expected to:	
CO1	Build an understanding of the fundamental concepts of computer networking.
CO2	Familiarize the student with the basic taxonomy and terminology of the computer networking area.
CO3	Student could analyze and evaluate different protocols and Internet components.
CO4	Gaining practical experience in installation, monitoring, and troubleshooting of a network
CO5	Students are capable to design an individual networks.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Understand and describe** concepts, taxonomy and terminology of computer network area.	Identify/Knowledge/Remember
CO-2	**Analyze and evaluate** network layer protocols.	Analyzing
CO-3	**Analyze and evaluate** Internet components.	Understanding/ Comprehension:
CO-4	**Analyzing and monitoring** the behavior of packets on the network.	Analyzing
CO-5	Design network.	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**
CO1	2	1	3	2	2	2	-	-	-	-
CO2	-	-	2	2	2	-	-	-	-	-
CO3	-	-	3	3	3	-	-	-	-	-
CO4	3	3	3	3	3	2	3	2	2	-
CO5	2	2	2	2	2	3	2	1	2	-
CO (Average) Course Average	**1.4**	**1.2**	**2.6**	**2.4**	**2.4**	**1.4**	**2.5**	**06**	**0.8**	**0.0**

Syllabus

UNIT 1: Introduction, Fundamental requirements of network, OSI & TCP/IP model
UNIT 2: Physical layer issues, Link layer, Medium Access protocols (IEEE 802.3 ...) and related issues
UNIT 3: Network layer: IP and other protocols, Data and Control plane, Routing protocols and LAN design.
UNIT 4: Transport layer Protocols and related Issues
UNIT 5: Basic client server architecture, introduction to different application layer protocols like ftp, telnet, mail(SMTP), HTTP, DNS, DHCP and peer to peer

Text Books

1. Kurose, James F. Computer networking A top-down approach featuring the internet, 3/E. Pearson Education India, 2005.
2. Peterson, Larry L., and Bruce S. Davie. Computer networks: a systems approach. Elsevier, 2007.
3. Forouzan, Behrouz A. "Data Communication and Networking, by McGraw." (2014).
4. Schmidt, Douglas, and Stephen D. Huston. C++ Network Programming, Volume I: Mastering Complexity with ACE and Patterns. FT Press, 2001.

Reference Books

1. Lowe, Doug. Networking all-in-one desk reference for dummies. John Wiley & Sons, 2004.
2. Tanenbaum, Andrew S. "Computer networks." Prentice-Hall of India Private Limited, 2006.

Database Management System
Syllabus

Prerequisite: Basic of Data Structures.
L-T-P: 3-0-2, **Credits:** 4 Type**:** Core Essential Subjects (CES)

Course Description

In this course students will study the basic functions and capabilities of database management systems (DBMS). Emphasis is placed on the use of DBMS in solving information processing problems which will include database design case studies as well as SQL programming assignments along with transactions. A lab course is associated with it to strengthen the concepts.

Course Objective

Students undergoing this course are expected to:	
1	Understand the basic concepts and the applications of database systems.
2	Understand the relational database design principles.
3	Master the basics of SQL and construct queries using SQL.
4	Familiar with the basic issues of transaction processing and concurrency control.
5	Familiar with database storage structures and access techniques.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Demonstrate** the basic elements of a relational database management system.	Identify/Knowledge
CO-2	**Design** entity relationship and convert entity relationship diagrams into RDBMS.	Creating
CO-3	**Create** SQL queries on the respect data into RDBMS and formulate SQL queries on the data by understanding the concepts of normalization and query optimization.	Analyzing/ Creating
CO-4	**Apply** and **Relate** the concept of transaction, concurrency control and recovery in database.	Applying
CO-5	**Recognize** and **Identify** the use of indexing and hashing technique used in database design.	Understanding/ Comprehension

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	-	-	2	1	-	-	-	2	2
CO2	2	-	2	2	2	-	-	-	-	2
CO3	1	2	2	2	2	-	1	2	-	1

CO4	-	-	-	3	3	-	2	-	1	-
CO5	1	3	2	-	-	3	-	-	-	1
CO (Average) Course Average	**1.2**	**1.0**	**1.2**	**1.8**	**1.6**	**0.6**	**0.3**	**0.4**	**0.6**	**1.2**

Syllabus

UNIT-I: Database system concept and architecture, Entity Relationship and Enhanced E-R.
UNIT-II: Relational Data Model and Relational Algebra, SQL, Indexing, Query Optimization.
UNIT III: Relational Database Design, Normalization principles and normal forms.
UNIT IV: Transaction concept and concurrency control.
UNIT V: Data storage, indexing, query processing and physical design.
UNIT VI: DBMS: Case Study

Text Books

1. Silberschatz, Abraham, Henry F. Korth, and Shashank Sudarshan. Database system concepts. Vol. 4. New York: McGraw-Hill, 1997.
2. Elmasri, R., et al. Fundamentals of Database Systems. Addison-Wesleypublisher, 2000.
3. Garcia-Molina, Hector, Jeffrey D. Ullman, and Jennifer Widom. Database system implementation. Vol. 672. Upper Saddle River, NJ:: Prentice Hall, 2000.
4. Ramakrishnan, Raghu, Johannes Gehrke, and Johannes Gehrke. Database management systems. Vol. 3. New York: McGraw-Hill, 2003.

Reference Books

1. Garcia-Molina, Hector. Database systems: the complete book. Pearson Education India, 2008.
2. Garcia-Molina, Hector, Jeffrey D. Ullman, and Jennifer Widom. Database system implementation. Vol. 672. Upper Saddle River, NJ:: Prentice Hall, 2000.

Embedded Systems
Syllabus

Prerequisite: Digital Electronics, Knowledge of Microcontrollers and C programming.
L-T-P: 3-0-2, **Credits:** 4**Type:** Core Essential Subjects (CES)

Course Description

1. To have knowledge about the basic working of a microcontroller system and its programming in assembly language.
2. To provide experience to integrate hardware and software for microcontroller applications systems.
3. To design & develop a microcontroller-based applications.
4. To acquaint students with methods of executive device control and to give them opportunity to apply and test those methods in practice

Course Objectives

Students undergoing this course are expected to:	
1	This course emphasizes on comprehensive treatment of embedded hardware and real time operating systems along with case studies, in tune with the requirements of Industry
2	this course is to enable the students to understand embedded-system programming and apply that knowledge to design and develop embedded solutions
3	It forms basic skills in embedded systems design
4	Those skills is usable in designing digital control units for consumer electronics, industrial automation, telecommunication systems, etc.
5	Acquire the ability to present engineering designs

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Identify features of various microcontrollers and would have acquired skills to analyse the real world problem domains in the context of Embedded system.	Identify/ Knowledge /Apply
CO-2	Select appropriate microcontroller for different application.	Understanding/ Analyze
CO-3	Interface microcontroller with hardware for given application.	Applying
CO-4	Write and execute assembly language programs (software) for given application.	Analyzing/Evaluate
CO-5	Develop small microcontroller based applications.	Creating/Apply

Mapping Course Outcomes with Program Outcomes

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	1	1	-	-	-	-	-	-	1
CO2	1	3	2	2	1	-	-	-	-	-
CO3	2	1	1	1	2	2	-	-	-	-
CO4	1	2	-	1	2	3	-	-	-	-
CO5	1	2	2	3	1	-	-	-	-	1

CO (Average) Course Average	**1.4**	**1.8**	**1.2**	**1.4**	**1.2**	**1**				**0.4**

3-High, 2-Medium, 1-Low

Syllabus

UNIT 1: Introduction: Embedded systems and general purpose computer systems, Embedded systems overview, Embedded system classifications and applications, Design challenge – optimizing design metrics, embedded processor technology, IC technology, Design technology, General-purpose processors: Software, microprocessor and micro-controller, current trends.

UNIT II: Microprocessors and microcontrollers: Hierarchy of microcontrollers, architectures of microcontroller Harvard , Von Neumann RISC and CISC, overview of 8051 microcontroller, Pin description of the 8051, Blocks of 8051 microcontroller, ALU, PC, DPTR, PSW, Internal RAM, Internal ROM, Latch, SFRs, General purpose registers, Timer/Counter, Interrupt, Ports, Clock circuit, reset Circuit; Memory organization of 8051

UNIT III: Assembly language programming of 8051: Introduction, Assembling and running an 8051 program, The program counter and ROM space in the 8051, data types and directives, flag bits and the PSW register, register banks and stack, Jump, Loop, and Call instructions: Loop and jump instructions, Call instructions, Time delay for various 8051 chips;I/O port programming: 8051 I/O programming, I/O bit manipulation programming, 8051 addressing modes, Immediate and register addressing, Accessing memory using various addressing modes, Bit addresses for I/O and RAM.

UNIT IV: 8051 Programming in C: data types and time delay in 8051, I/O programming, logic operations, data conversion programs, accessing code ROM space, , Design and test of DS89C4x0 trainer, Programming 8051 timers, Counter programming, programming timers 0 and 1 in 8051, Basics of serial communication, serial port programming in Assembly, 8051 interrupts: Programming timer interrupts, Programming external hardware interrupts, Programming the serial communication interrupt, Interrupt priority in the 8051/52.

UNIT V: LCD interfacing, Keyboard interfacing, Parallel and serial ADC, DAC interfacing, Sensor interfacing and signal conditioning, memory address decoding, 8031/51 interfacing with external ROM, 8051 data memory space, accessing external data memory in 8051,8051 interfacing with the 8255, Programming the 8255, 8255 interfacing, 8051 C programming for the 8255, Motor control: relay and opto-isolators, Stepper motor interfacing, DC motor interfacing and PWM.

Text Books

1. Frank Vahid& Tony Givargis, Embedded System Design, 2nd Edition, John Wiley
2. MazidiKinlay ,8051 microcontroller and Embedded Systems Second Edition, Pearson Education Limited
3. Kenneth Ayala J. Thomson Delmar learning, The 8051 Microcontroller: Architecture, Programming, and Applications,(latest Edition)
4. Raj Kamal, Embedded Systems- Architecture, Programming and Design 3rd Edition

Reference Books

1. Lyla B Das, “Embedded Systems an Integrated Approach”, 1st edition, Pearson, 2012
2. David E. Simon, “An Embedded Software Primer”, 1st edition, Pearson Education, 2008.
3. Wayne Wolf, “Computers as Components-principles of Embedded Computer system Design”, 1st edition, Elsevier, 2009.

Software Engineering & Project Management
Syllabus

Prerequisite: Data Structures & Algorithms, Object Oriented Programming and Modeling and Database Management Systems
L-T-P: 3-0-0, **Credits: 3 Type:** Core Elective (CEL)

Course Description

In this course students will study the fundamentals of software engineering, including understanding system requirements, effective methods of design, coding, and testing, team software development, and the application of engineering tools. The course will combine a strong technical focus with a mini project (offered alongside), providing the opportunity to practice engineering knowledge, skills, and practices in a realistic development setting. Further, this course is aimed at introducing the primary concepts of project management related to managing software development projects. Students shall come to know how to successfully plan and implement a software project management activity, and to complete a specific project in time with the available budget.

Course Objectives

Students undergoing this course are expected to:	
1	The course assists to understand the basic theory of software engineering, and to apply these basic theoretical principles to a group software development project.
2	Knowledge of basic SW engineering methods and practices, and their appropriate application
3	Describe software engineering layered technology and Process frame work.
4	Understanding on quality control and how to ensure good quality software.
5	Describe software measurement and software risks

Course Outcomes

CO No.	**Course Outcomes (Action verb should be in italics**	**Bloom's taxonomy**
CO-1	Identify, formulate, and solve software engineering problems. Define principals and working of software process models: including the specification, design, implementation, and testing phases of software systems that meet specification, performance, maintenance and quality requirements	Identify/Knowledge
CO-2	Analyze, elicit and specify software requirements through a productive working relationship with various stakeholders of the project. Also, prepare SRS and software process metrics for a given requirement	Analyse/Identify/ Knowledge
CO-3	Understand various design techniques and solutions in one or more application domains using software engineering approaches that integrate ethical, social, legal and economic concerns	Analyzing/Understanding
CO-4	Develop the code from the design and effectively apply relevant standards. Prepare a Test Suite to perform testing and quality management	Applying/Understanding
CO-5	Monitor the progress of projects and to assess the risk of slippage	Applying/Understanding
CO-6	Create a Quality Assurance plan for a given project according to IEEE standards	Create/Understanding/ Comprehension
CO-7	Understand and analyse software management principles, software reuse, and an ability to engage in life-long learning.	Understand/Create

Mapping Course Outcomes with Program Outcomes:

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	3		1							3
CO2	2	1		3		3	2			2
CO3	2		3			3	1			2
CO4	3	2	3			2	2			3
CO5	3	1	2			2	2			3
CO6	3	1				1				3
CO7	1				2		2	2	3	1
CO (Average) Course Average	**2.3**	**0.6**	**1.6**	**0.5**	**0.3**	**1.5**	**1.16**	**0.3**	**0.5**	**2.3**

3-High, 2-Medium, 1-Low

Syllabus

UNIT 1: Introduction, Software life-cycle models, Software requirements, Requirements Specification, Software design, Software Development Models, Rapid application Development,

UNIT II: Software integration and testing, IEEE Software Engineering Standards, Software Reuse

UNIT III: Software Maintenance, Software Effort Estimation, COCOMO, FPA, Activity duration estimation

UNIT IV: Risk management, Identification, Analysis and abatement of risk, Cost Monitoring

UNIT V: Software quality Assurance, Software quality in Project Planning, Software quality Standards, Product quality management, SEICMM model. Application of SPM.

Cryptography & Network Security
Syllabus

Prerequisite: Computer Networks and Discrete Mathematics
L-T-P: 3-0-2, **Credits:** 4 **Type:** Core Essential Subjects (CES)

Course Description

In this course students will study the essential mathematical foundations for Information Security. This course features a rigorous introduction to modern cryptography, with an emphasis on the fundamental cryptographic primitives of public-key encryption, digital signatures, pseudo-random number generation, and basic protocols and their computational complexity requirements. After crediting these course students can look forward to wireless network security and E-commerce courses.

Course Objectives

Students undergoing this course are expected 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	Learning the concepts of Message authentication and Hash functions.
4	To learn about how to maintain the Confidentiality, Integrity and Availability of a data
5	To understand various protocols for network security to protect against the threats in the networks.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Identify** attacks types, security threats and their countermeasures	Identify/Knowledge
CO-2	**Understanding** and Symmetric and Asymmetric cryptography and basics of cryptanalysis	Understanding/ Comprehension:
CO-3	**Demonstrates and apply** cryptographic algorithms and protocols in attaining security in Network	Applying
CO-4	**Evaluate** security level and performance issues of different encryption and authentication techniques.	Analyzing
CO-5	**Develop** secure communication protocol	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	-	-	2	1	-	-	-	-	-
CO2	2	-	2	2	2	-	-	-	-	-
CO3	1	2	2	2	2	-	-	2	-	-
CO4	-	-	-	3	3	-	-	-	-	-
CO5	1	3	2	-	-	3	-	-	-	-
CO (Average) Course Average	**1.2**	**1.0**	**1.2**	**1.8**	**1.6**	**0.6**	**0.0**	0.4	**0.0**	**0.0**

UNIT 1: Introduction to Cryptography, Block Cipher and Stream Cipher

Introduction to security attacks - services and mechanism - introduction to cryptography - Conventional Encryption: Conventional encryption model - classical encryption techniques - substitution ciphers and transposition ciphers – cryptanalysis – steganography - stream and block ciphers - Modern Block Ciphers: Block ciphers principals - Shannon's theory of confusion and diffusion - fiestal structure - data encryption standard(DES) - strength of DES - differential and linear crypt-analysis of DES - block cipher modes of operations - triple DES – AES.

UNIT II: Mathematical Background for Cryptography

Modulo Arithmetic, Introduction, Prime Number Generation, Shannon's Theory of Perfect Secrecy GCD Euclid's Algorithm, Algebraic Structures (Groups), Chinese Remainder Theorem.

Unit III: Public key cryptography and Authentication requirements) Principles of public key crypto systems - RSA algorithm - security of RSA - key management – Diffle-Hellman key functions - message authentication code - hash functions - birthday attacks – security of hash functions and MACS exchange algorithm - introductory idea of Elliptic curve cryptography – Elgamel encryption - Message Authentication and Hash Function: Authentication requirements - authentication

UNIT IV (Integrity checks and Authentication algorithms): MD5 message digest algorithm - Secure hash algorithm (SHA) Digital Signatures: Digital Signatures - authentication protocols - digital signature standards (DSS) - proof of digital signature algorithm - Authentication Applications: Kerberos and X.509 - directory authentication service - electronic mail security-pretty good privacy (PGP) - S/MIME.

UNIT V (IPSecurity, Key Management, Web Security and System Security) IP Security: Architecture - Authentication header - Encapsulating security payloads - combining security associations - key management. Secure socket layer and transport layer security - secure electronic transaction (SET) - System Security: Intruders - Viruses and related threads - firewall design principals – trusted systems.

Text Books

1. Mao, Wenbo. Modern cryptography: theory and practice. Pearson Education India, 2003.
2. Schneier, Bruce. Applied cryptography: protocols, algorithms, and source code in C. john wiley& sons, 2007.
3. Stinson, Douglas R. Cryptography: theory and practice. Chapman and Hall/CRC, 2005.

Reference Books

1. Buchmann, Johannes. Introduction to cryptography. Vol. 335. New York: Springer, 2004.
2. Stallings, William. Cryptography and network security, 4/E. Pearson Education India, 2006.

Syllabus
Semester-VI

Wireless and Mobile Networks
Syllabus

Prerequisite: Computer Networks
L-T-P: 4-0-0, **Credits: 4Type:** Minors/Honors

Course Description

This course will cover the area of mobile and wireless networking, looking at the unique network protocol challenges and opportunities presented by wireless communication and host or router mobility. Although the course will touch on some of the important physical layer properties of wireless communications, the focus will be on network protocols above the physical layer, with an emphasis on the media access control, network, and transport protocol layers. Additionally, this course also covers short range wireless networks (Bluetooth, ZigBee) and wide range wireless networks (3G, 4G, WiMax etc.).

Course Objectives

Students undergoing this course are expected to:	
1	To provide an overview of Wireless/Mobile Networks and its applications
2	To appreciate the contribution of Wireless Communication networks to overall technological growth.
3	To explain the various terminology, principles, devices, schemes, concepts, algorithms, and different methodologies used in Wireless Communication Networks.
4	To enable students to compare short, medium and wide area wireless networks
5	To understand the effect of wireless media and mobility on network protocol stack.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics**	**Bloom's taxonomy**
CO-1	Understand fundamentals of wireless/mobile networks	Identify/Knowledge
CO-2	Classify multiple media access techniques in wireless/mobile communication.	Identify/Knowledge
CO-3	Analyze mobility, scalability, and their unique characteristics in wireless and mobile networks	Analyzing/Understanding
CO-4	Demonstrate basic skills for wide area wireless networks design.	Applying/Understanding
CO-5	Students Understood Basic & Advanced Knowledge of Tree Graph	Understanding/Comprehension

Mapping Course Outcomes with Program Outcomes:

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	-	-	-	2	-	3	-	-	-	-
CO2	-	-	-	-	2	-	-	-	-	-
CO3	2	-	2	-	-	2	-	1	-	-
CO4	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	2	-	-	2	-
CO (Course Average)	**0.4**	**-**	**0.4**	**0.4**	**0.4**	**1.4**	**-**	**0.2**	**0.4**	**-**

3-High, 2-Medium, 1-Low

Syllabus

UNIT-1: Medium range wireless communication: WiFi and IEEE 802.11
UNIT-II: Short range wireless communication: Bluetooth, Zigbee etc.
UNIT III: Wide area wireless communication: Cellular Networks: 3G and 4G/LTE, WiMax
UNIT IV: Routing techniques for mobile nodes on the Internet, particularly Mobile IP and its variants.
UNIT V: Routing techniques in multi-hop wireless ad hoc networks.
UNIT VI: Effects of mobility and wireless transmissions on reliable transport protocols such as TCP.

Text Books

1. Mobile Communications 2ndEdition by Jochen Schiller, Pearson 2010.
2. Adhoc Networking by Charles Perkins, Pearson, 2008

Reference Books:

1. Petri Launiainen, A Brief History of Everything Wireless,2018.

Data Mining and Warehousing
Syllabus

Prerequisite: DBMS, Algorithms, Engineering Mathematics.
L-T-P: 3-0-2, **Credits:** 4**Type:** Core Elective (CEL)

Course Description

This course teaches students how to use OLAP tools to understand a data warehouse, data gathering strategies and processes, and data pre-processing. This course will go over the various data mining models and approaches such as from protein sequences and structures to biological imaging, sensor readings, and chemical data, the volume of data is increasing at an unparalleled rate. The capacity to mine for information inherent in the collection must be provided in order to make this large amount of data more valuable than just a digital data storage structure. The standard algorithms for such data mining techniques will be covered in this course. Recent trends in text data mining, graph mining, spatio-temporal data mining, and other areas will be highlighted.

Course Objectives

Students undergoing this course are expected to:	
1	To teach data warehousing and data mining principles, concepts, and applications.
2	To introduce data mining as a critical stage in the knowledge recovery process.
3	To instill the concepts of conceptual, logical, and physical data warehouse design, as well as OLAP applications and deployment.
4	To instill basic notions that serve as the foundation for data mining.
5	Create a data warehouse or data mart to give information to management in a way that they can understand.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	For any organization, **create** a data mart or data warehouse.	Creating
CO-2	**Use** data mining techniques to extract knowledge.	Apply, Knowledge
CO-3	Learn how to **use** new data-mining technologies.	Applying, Understanding
CO-4	**Research** current data mining trends such as web mining, spatial-temporal mining, big data, and bioinformatics.	Identify/Knowledge

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	3	2	2	2	2	-	2	2	2
CO2	2	2	2	3	2	2	-	2	2	3
CO3	-	3	-	2	-	2	-	2	3	2
CO4	2	2	2	3	2	2	2	2	2	3

CO (Average) Course Average	**1.5**	**2.5**	**1.5**	**2.5**	**1.5**	**2.0**	**0.5**	**2.0**	**1.0**	**0.5**

Syllabus

UNIT I: Introduction of data mining and Data warehouse
Introduction and overview of Data Mining – History – Strategies – Techniques – Applications – Challenges – Future- Types of Data, Motivation and significance of data mining, data mining functionalities, interestingness measures, classification of data mining system, major issues in data mining.– Data Warehouses – Data Processing - Quality Measure – OLAP – Sampling.

UNIT II:Data pre-processing, data types, input and output of data mining algorithm Need, data summarization, data cleaning, data integration and transformation, data reduction techniques – Singular Value Decomposition (SVD), Discrete Fourier Transform (DFT), Discrete Wavelet Transform (DWT), data discretization and concept hierarchy generalization. Different Types of features – Concept Learning – Output of Data Mining Algorithms.

UNIT III: Data warehouse and OLAP technology, Data cube computation and data generalization
Data warehouse definition, multidimensional data model(s), data warehouse architecture, OLAP server types, data warehouse implementation, on-line analytical processing and mining, Efficient methods for data cube computation, discovery driven exploration of data cubes, complex aggregation, attribute oriented induction for data generalization.

UNIT IV:Classification and prediction: Definition, decision tree induction, Bayesian classification, rule-based classification, classification by back propagation and support vector machines, associative classification, lazy learners, ensemble method, prediction, accuracy and error measures.

UNIT V:Mining frequent patterns, associations and correlations:
Basic concepts, efficient and scalable frequent item set mining algorithms, A-prior, FP Growth, ECLAT mining various kinds of association rules – multilevel and multidimensional, association rule mining versus correlation analysis, constraint-based association mining, Advanced concepts-, handling categorical and continues attributes, sequential patterns, sub graph patterns, Infrequent patterns.

UNIT VI:Cluster analysis: Definition, clustering algorithms –partitioning, hierarchical, density based, grid based and model based; Clustering high dimensional data, constraint based cluster analysis, outlier analysis – density based and distance based, prototype based clustering.

UNIT VII:Data mining on complex data and applications: Algorithms for mining of spatial data, multimedia data, text data; Data mining applications, social impacts of data mining, trends in data mining, Applications such as Bioinformatics, Information Retrieval, Link Analysis, Search Engines, Big Data etc. and case studies and projects 3(L)

Text Books

1. Han, J. and Kamber, M., “Data Mining - Concepts and Techniques”, 3rd Ed., Morgan Kaufmann Series. 2011.
2. Ali, A. B. M. S. and Wasimi, S. A., “Data Mining - Methods and Techniques”, Cengage Publishers. 2009.
3. Tan, P.N., Steinbach, M. and Kumar, V., “Introduction to Data Mining”, Addison Wesley –Pearson. 2008.
4. Pujari, A. K., “Data Mining Techniques”, 4th Ed., Sangam Books. 2008.
5. D. Hand, H. Mannila, and P. Smyth. Principles of Data Mining, MIT Press, 2001.

6. Recent literature from ACM SIGMOD, VLDB, IEEE Trans. Knowledge & Data Engg.,Data Mining and Knowledge Discovery, ACM SIGKDD, IEEE ICDM, SIAM Data Mining,ICML.

Reference Books

1. **I**nmon, William H. "The data warehouse and data mining." Communications of the ACM 39.11 (1996): 49-51.
2. Huang, Xuegang. "Data Warehouse Architecture: Practices and Trends." *Evolving Application Domains of Data Warehousing and Mining: Trends and Solutions*. IGI Global, 2010. 1-15.
3. Wu, Xindong, and Vipin Kumar, eds. The top ten algorithms in data mining. CRC press, 2009

Distributed Systems
Syllabus

Prerequisite: Operating Systems and Computer Networks
L-T-P: 3-0-2,   **Credits:** 4**Type:** Core Essential Subjects (CES)

Course Description

Students will be introduced to the need and wide applicability of the distributed architecture in our daily life and its applications in Industry. The foundational concepts of distributed systems with the introduction to inter-process communications with least number of message exchanges, need of synchronous and asynchronous applications are covered. The course takes up design and implementation of efficient tools, protocols and systems in a distributed environment. Topics such as resource and process management along with naming, synchronization, replication, and caching are covered. Introduction to distributed file system, Concurrency control and Distributed Transaction Processing recovery are studied. Students may note that the courses namely web services, cloud computing, virtualization, peer-to-peer and Internet computing all have distributed systems concepts at their foundation.

Course Objectives

Students undergoing this course are expected to:	
CO-1	To provide students with the fundamentals and essentials of distributed computing
CO-2	To provide students a basic understanding of Internet of Things and the related standards and protocols.
CO-3	To explore challenges in the integration of IoT, edge and cloud computing
CO-4	To expose students with challenges and applicability of distributed computing in different domains.
CO-5	Understanding of design and development of distributed applications.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Understand** the need and wide applicability of the distributed architecture and applications in Industry	Understanding/ Comprehension:
CO-2	**Understand** and **explain** the need of communication among geographically distributed nodes and need of synchronous and asynchronous applications with least exchange of messages..	Identify/Knowledge/Remember
CO-3	**Explain** various issues and design process of resource and process management.	Understanding/ Comprehension:
CO-4	**Understanding** the need and concepts relating to Distributed Transaction Processing and importance of Naming System'.	Understanding/ Comprehension:
CO-5	**Discuss** application and design of the course in Distributed Ledger Technology and other Distributed Applications.	Analyzing

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	3	3	-	3	2	3	3	3	-	2
CO2	2	-	3	3	3	-	-	2	3	3
CO3	1	3	3	3	2	3	3	-	3	-
CO4	2	2	2	-	3	3	-	3	3	3
CO5	3	3	3	3	3	3	2	3	2	3
CO (Average) Course Average	**2.2**	**2.2**	**2.2**	**2.4**	**2.6**	**2.4**	**1.6**	**1.6**	**2.2**	**2.2**

Syllabus

UNIT I: Overview and fundamentals of distributed systems, Network & Inter-networking, Architectures, Idem potency, Byzantine and Fail Stop failures, Challenges.

UNIT II: Distributed Communication Paradigms - Message Passing, Remote Procedure calls, Distributed Shared Memory, Synchronization, Distributed Deadlock Avoidance, Prevention Detection & Recovery, Hierarchical and Fully Distributed approaches for deadlock detection and recovery.

UNIT III: Distributed Resource Management need and issues, Load balancing and load sharing, Issues in designing load balancing algorithms, Migration limiting and state information exchange policies, Distributed Process Management and its need, Process migration policy and mechanism, Process address space transfer mechanisms and handling message passing.

UNIT IV: Introduction to distributed transactions, Distributed File Management. Naming Issues and challenges, name resolution, Object locating mechanism, Name servers and context binding

UNIT V: Introduction to Fog, Edge and Cloud Computing and its Applications, Introduction to Distributed Ledger Technology and other Distributed Applications.

TextBooks:

1. Sinha, Pradeep K. "Distributed operating system." Concept and Design (1998).
2. Coulouris, George, Jean Dollimore, and Tim Kindberg. "Distributed Systems: Concepts and Design Edition 3." System 2.11: 15.

Reference Books:

1. Van Steen, Maarten, and Andrew S. Tanenbaum. Distributed systems. Leiden, The Netherlands: Maarten van Steen, 2017.

Syllabus
Semester-VII

Image Processing and Computer Vision
Syllabus

Prerequisite: Engineering Mathematics and Basics of Machine Learning
L-T-P: 3-0-2, **Credits:** 4**Type:** Core Elective (CEL)

Course Description

This course cover in image representation and description, image filtering, segmentation techniques, image geometry, dynamic vision and motion analysis, matching and analysis.

Course Objectives
Course Outcomes

Students undergoing this course are expected to:	
1	This course is designed for students: To introduce the fundamentals of image processing and computer vision.
2	It includes image formation, color models, image transforms methods, and image enhancement and restoration techniques.
3	The student can introduce and apply segmentation, feature extraction, and pattern analysis techniques in real-world applications such as classification, scene understanding, object recognition, etc.

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Explore** machine learning concepts, Python tools and libraries available for machine learning, and employ machine learning techniques on real-world data using Python.	Identify/Knowledge
CO-2	**Identify** the problems in which Classification is to be used, and the problems in which Regression is to be used, and solve such problems using Python tools and libraries.	Understanding/ Comprehension:
CO-3	**Identify** the problems in which Support Vector Machine is to be employed, and the problems in which Decision Trees are to be employed, and solve such problems using Python tools and libraries.	Identify Applying
CO-4	Combine multiple models to solve machine learning problems, wherever it will be necessary, using Ensemble learning and Random Forests, and solve them in Python.	Analyzing
CO-5	**Perform** Dimensionality Reduction, in whichever problem it will be necessary, using Python.	Applying

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	3	2	2	3	3	-	-	-	-	-
CO2	3	3	3	1	3	-	-	-	-	-
CO3	2	3	3	1	3	-	-	-	-	-

CO4	3	3	3	1	3	-	-	-	-	-
CO5	2	3	3	3	3	2	-	-	2	-
CO (Average) Course Average	**2.6**	**2.8**	**2.8**	**1.8**	**3**	**0.4**	**-**	**-**	**0.4**	**-**

Syllabus

UNIT I: Introduction: Introduction and Goals of Computer Vision and Image Processing, Image acquisition and formation models, Sampling and quantization, Relationships between pixels, Elements of visual perception, brightness, contrast, hue, saturation, Color image fundamentals - RGB, HSI models, Mathematical preliminaries,2D-Image Transforms.

UNIT II: Image Enhancement and Restoration: Histogram equalization and specification techniques, Noise distributions, Spatial averaging, Directional Smoothing, Median, Geometric mean, Harmonic mean, Contra harmonic mean filters, Homomorphic filtering, Image Restoration - degradation model, Unconstrained restoration - Lagrange multiplier and Constrained restoration, Inverse filtering-removal of blur caused by uniform linear motion, Wiener filtering.

UNIT III: Segmentation: Thresholding, Region based segmentation, Region growing, Region splitting and Merging, Segmentation by morphological watersheds: basic concepts, Watershed segmentation algorithm, Fuzzy c-means clustering.

UNIT IV: Image Descriptors and Features: Edge detection, Line detectors, Corner detectors, Gaussian derivative filters, Orientation histogram, HOG, Scale Invariant Feature Transform, speeded up Robust Features, Scale-space analysis-Image Pyramid, Gabor filters and DWT, Object Boundary and Shape Representations.

UNIT V: Pattern Analysis: PCA, LDA, ICA, Non-parametric methods, Clustering: K-means, K-medoids, Gaussian mixture models, Classification: Bayes theorem, k-NN, ANN models, Motion analysis: Background subtraction and modeling, spatio-temporal analysis, motion parameter estimations.

Text Books

1. Gonzales R., Woods R.: “Digital Image Processing”, Fourth Edition, Pearson Prentice-Hall, 2018.
2. D.A. Forsyth, J. Ponce: “Computer Vision A Modern Approach”. Pearson Prentice-Hall, 2nd edition (1 January 2015)
3. A.K. Jain. “Fundamentals of Digital Image Processing”, Published by Prentice Hall, Upper Saddle River, NJ.

Reference Books

1. Bhunia, Manaskamal, “Computer Vision and image processing: Fundamentals and applications” CRC Press,2019.
2. S. Sridhar. “Digital Image Processing”, Oxford University Press; Second edition (27June 2016)

Formal Methods
Syllabus

Prerequisite: Automata, Discrete Mathematics and any Programming language.
L-T-P: 4-0-0, **Credit:** 4**Type:** Core Essential Subjects (CES)

Course Description

In this course students will understand mechanisms and principles involved in the use of formal methods for the design and development of high intensity concurrent and distributed systems. It is based on the application of formal logic, process algebra, mathematical models and Model Checking for the automated verification. After crediting the course it is expected that the students will acquire an understanding of logical concepts and their application to the analysis of software design.

Course Objectives

Students undergoing this course are expected to:	
1	Understanding mechanism and principle involved in the use of formal method for the design and development of high intensity concurrent and distributed system
2	Objective of this course to specification, verification, automated theorem proving, model checking and correctness of system design..
3	The purpose is to construct, with high confidence, systems that behave according to their specification
4	Students will learn how to use various tools, including theorem proving and model checking tools, and will work in groups to apply the tools to various domains

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Explain the potential and limitations of using logic based verification methods for assessing and improving software correctness and identify what can and cannot be analyzed with certain logics and proof methods.	Knowledge/ understanding
CO-2	The use of formal methods for software and hardware design is motivated by the expectation that, as in other engineering disciplines, performing appropriate mathematical analysis can contribute to the reliability and robustness of a design.	Understanding/ Analyze
CO-3	It helps disambiguate system specifications and articulate implicit assumptions. They also expose flaws in system requirements, and their rigor enables a better understanding of the problem.	Create/ Understanding
CO-4	Formal methods are techniques used by software engineers to design safety-critical systems and their components they are techniques that involve mathematical expressions to model "abstract representation" of the system.	Understanding/ Applying
CO-5	Apply techniques for computer science engineering problems and Employ abstraction, modeling, and rigorous reasoning when approaching the development of correctly functioning software.	Applying

Mapping Course Outcomes with Program Outcomes

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	2	2	3		2		1	2	1	
CO2	2	3		2		3	2			2
CO3				2	1	2		2	1	
CO4	3	2	3		2	2				1
CO5	2	2	2	2	3		2			2
CO (Average) Course Average	**1.8**	**1.8**	**1.6**	**1.2**	**1.6**	**1.4**	**1**	**0.8**	**0.4**	**1**

3-High, 2-Medium, 1-Low

Syllabus

UNIT-1: Introduction, Hardware and Software verification, Model Checking, Reactive systems, Process algebras. (7)

UNIT-II: Modeling Systems, The language CCS, CCS process constructions, CCS formally, Introduction to Pi-calculus, Concurrency Work Bench. (9)

UNITIII: Theory of fixed points and bisimulation equivalence, Posets and complete lattices, Tarski's fixed point theorem, Behavioural equivalences, Trace equivalence, Strong bisimilarity, Weak bisimilarity, Bisimulation as a fixed point. (8).

UNIT IV: Modalities and Capabilities, Safety Properties and Invariants, Liveness Properties, Fairness, Hennessy–Milner logic, HML with recursion, Temporal Properties and Modal Mu-Calculus. (8)

UNIT V: Verifying temporal properties, CTL Model Checking algorithms, The fixed point characterization of CTL. (8)

Text Books:

1. Robin Milner, Communication and Concurrency.
2. Robin Milner, Pi-calculus
3. Michael Huth and Mark Ryan, Logic in computer science

Reference Books:

1. Clarke, Grumberg, Peled,Model Checking.
2. Colin Sterling, Modal and Temporal Properties of Processes.

Machine Learning with Python
Syllabus

Prerequisite: Computer Programming, Basics of Mathematics and Statistics
L-T-P: 3-0-0,**Credits:** 3 **Type:** Core Elective (CEL)

Course Description

This course contents various machine learning algorithms and its application in real life. Its contents supervised, unsupervised, reinforcement learning, clustering, bayes theorem concept and application of machine learning.

Course Objectives

Students undergoing this course are expected to:	
1	This course is designed for students: To introduce the fundamental concepts of machine learning and its applications.
2	Students may explore the understanding of machine learning with their underlying mathematical concepts, issues and challenges, including data, model selection, model complexity, strengths and weaknesses and their relevant applications.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	***Understanding*** the nature of data, and model selection.	Identify/Knowledge
CO-2	Describe the various classification algorithms.	Comprehension
CO-3	Develop and ***apply*** clustering base algorithm for finding relationships between data variables.	Applying
CO-4	***Understanding*** of back propagation algorithm for the training of neural networks	Understanding
CO-5	***Design*** and implement various machine learning algorithms in a range of real-world applications.	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	3	3	2	3	-	-	-	-	-
CO2	2	3	3	2	3	-	-	-	-	-
CO3	2	3	3	3	3	1	-	-	-	-
CO4	3	3	3	3	3	-	-	-	-	-
CO5	2	3	3	3	3	2	-	-	2	-
CO	**2.2**	**3**	**3**	**2.6**	**3**	**0.6**	**-**	**-**	**0.4**	**-**

(Course Average)										

Syllabus

UNIT I: Introduction to Machine Learning; Supervised Learning, Unsupervised Learning, Reinforcement Learning**,** Machine learning basics: capacity, over fitting and under fitting, hyper parameters and validation sets, bias & variance.

UNIT II: Bayes Theorem Concept Learning Maximum Likelihood classifier, Bayesian Belief Networks, Decision-trees, *univariate and multivariate decision tree*, network equivalent of decision trees, over fitting and evaluation.

UNIT III: Logistic Regression, Support Vector Machine, Kernel function and Kernel SVM, Instant based Learning: k- Nearest Neighbor, locally weighted Regression, Radial Bases Functions; Case Based Learning.

UNIT IV: Clustering: k-means, adaptive hierarchical clustering, and Gaussian mixture model, Neural network: Perception, multilayer network, back propagation.

UNIT V: Applications of machine learning (Case studies): Business Analytics, Natural language processing, Computer vision, Internet of Things, bio-informatics etc.

Text Books

1. Tom M. Mitchell, ―Machine Learning, McGraw-Hill Education (India) Private Limited, 2013.
2. Mohri, M., Rostamizadedh, A., and Talwalkar, A., Foundations of Machine Learning, The MIT Press (2012)

Reference Books

1. EthemAlpaydin, (2004) "Introduction to Machine Learning (Adaptive Computation and Machine Learning)", The MIT Press
2. T. astie, R. Tibshirani, J. H. Friedman, "The Elements of Statistical Learning", Springer (2nd ed.), 2009

Professional Elective I & II

Data Compression

Prerequisites:

Basic data structures and algorithms, Fundamental concepts of computer architecture.

Objective:

Develop theoretical foundations of data compression, concepts and algorithms for lossy and lossless data compression, signal modeling and its extension to compression with applications to speech, image and video processing.

Course Description

The course discusses the theory and methods of data compression of signals, images, and video. Data Compression is the computational problem of how to encode a data file (text, image, audio, video) so that the new file has fewer bits the original file. Techniques covered include: Quantization, Vector Quantization, Differential Schemes, Filterbanks and Subband Coding, Wavelet Transform, JPEG 2000, and MPEG. Coverage of selected topics of recent research issues in data compression is also taken up.

Course Outline (To be covered in 30 lectures)

1. Information theoretic foundations, Arithmetic coding (6)
2. Dictionary techniques, Context modeling (6)
3. Lossless image compression, Lossy coding preliminaries (6)
4. Scalar and vector quantization (6)
5. Differential encoding, Transform coding (6)

Students undergoing this course are expected to:	
CO-1	Understand the mathematical basics and concepts applicable to compression.
CO-2	Understand the formats of data storage such as image, video, sound and etc. Later analyse the concepts of lossy and lossless compression techniques.
CO-3	Understand the standard mechanisms of compression and decompression and apply them on real time data.
CO-4	To gain knowledge on theory and practical compression techniques of scalar and vector quantization, Differential encoding, Transform coding and etc.
CO-5	To develop skills for using recent and new data compression techniques to solve practical problems in a variety of disciplines such as vision processing, speech processing, text processing and etc.

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**
CO1	3		2		2				-	-
CO2	3	3	2		1				-	-
CO3			2	3	2	2			-	1
CO4			2	2	1	2			-	-

CO5		3	2	1	2	3		1	-	1
CO (Average) Course Average	**1.2**	**1.2**	**2**	**1.2**	**1.6**	**1.4**	**0**	**0.2**	**0**	**0.4**

Text Books

1. Introduction to Data Compression by Sayood, Khalid,
2. Data Compression: The Complete Reference by M. Nelson,

Design Patterns

Prerequisites: Prior knowledge of object-oriented programming is essential for this course. The students are expected to be proficient in Java, Principle of Programming Languages.

Objective:

- Understand and be able to apply incremental/iterative development
- Understand common design patterns
- Be able to identify appropriate patterns for design problems
- Be able to evaluate the quality software source code
- Be able to refactor badly designed program properly using patterns

Course Description

This course is an introduction to software design patterns. Each pattern represents a best practice solution to a software problem in context of some application. The course will cover both the rationale and benefits of object-oriented software design patterns. Several example problems need to be studied to investigate the development of good design patterns. Specific patterns, such as Observer, State, Adapter, Strategy, Decorator and Abstract Factory would be covered.

Course Outline

1. Introduction To Design Patterns, Introduction To Java, Some OO Design Principles , The Observer Pattern, The Template Method Pattern (6)
2. Factory Patterns: Factory Method and Abstract Factory, The Singleton Pattern, The Iterator Pattern, The Composite Pattern, The Facade Pattern (6)
3. The State and Strategy Patterns, Functors and the Command Pattern, The Proxy Pattern (5)
4. RMI, The Adapter Pattern, The Decorator Pattern (4)
5. Dynamic Proxies In Java, The Chain of Responsibility Pattern, Concurrency Patterns, The Visitor Pattern, Anti Patterns (5)
6. Layer, Pipe and Filters, Black Board Broker, Case Studies (4)

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Understand and be able to apply incremental/iterative development	Understanding/ Comprehension
CO-2	Understand common design patterns	Understanding/ Knowledge
CO-3	Be able to identify appropriate patterns for design problems	Identify/Knowledge
CO-4	Be able to evaluate the quality software source code.	Analyzing /Identify
CO-5	Be able to refactor badly designed program properly using patterns.	Creating /Analyzing

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	1	1	2	1	1	1	1	1	2	2
CO2	2	2	2	1	2	1	2	1	1	1	2	2
CO3	2	2	2	1	2	2	2	1	1	2	2	2
CO4	2	3	3	3	3	2	2	1	2	2	1	2
CO5	2	2	2	2	2	3	2	1	3	2	2	2
CO (Average)	2	2	2	2	2	2	2	1	2	2	2	2

Text Books

1. Design Patterns - Elements Of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides,
2. Head First Design Patterns, Eric Freeman and Elisabeth Freeman
3. Applied Java Patterns, Stephen Stelting and Olav Maassen,
4. Java Design Patterns - A Tutorial, James W. Cooper,
5. Refactoring To Patterns, Joshua Kerievsky

Functional Programming

Prerequisites: Basic Mathematics.

Objective:

- master foundational techniques from the paradigm of functional programming.
- be trained in using abstraction to structure programs.
- be able to explain and use recursion in general, as well as know how to distinguish between recursive and iterative processes.
- be able to write and use higher-order functions.
- master techniques for delayed evaluation for working with infinite data structures such as streams.
- have insight in different models for understanding how code is evaluated.

Course Description

This course aims to make functional techniques and thought patterns part of programming skills of the students. This course presents the functional programming paradigm, based on the idea of functions as "first-class" values that can be computed and operated. Functional languages provide great power of expression while maintaining simplicity, making it easier to write correct and maintainable software. Upon successful completion of the course, students would be able to analyze problems and apply functional programming concepts and techniques to solve the problems.

Students undergoing this course are expected to:	
CO-1	Master foundational techniques from the paradigm of functional programming.
CO-2	Trained in using abstraction to structure programs.
CO-3	Able to explain and use recursion in general, as well as know how to distinguish between recursive and iterative processes.
CO-4	Able to write and use higher-order functions.
CO-5	Master techniques for delayed evaluation for working with infinite data structures such as streams.

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**
CO1	2	-	-	1	2	-	-	2	-	-
CO2	2	2	-	1	-	-	-	-	-	-
CO3	2		-	1	2	2	-	-	-	-
CO4	2		-	1	-	-	-	2	-	-
CO5	1	1	2	1	-	1	-	1	-	-
CO (Average) Course Average	**1.8**	**0.6**	**0.4**	**1**	**0.8**	**0.6**	-	1	**-**	**-**

Course Outline (To be covered in 30 lectures)

1. Introduction, Problem Solving with Functional Language, Programming with functions, List constructors and selectors, Recursive functions, Accumulating parameters, Local definitions, Higher Order functions, Dot notation, and example simple functional programs (12)
2. Un-typed and Typed Lambda Calculus and Combinators, Term structure and substitution, alpha and Beta reductions and Beta Equality, Normal Form, Combinators, Church Numerals, Reduction Rules, Y-Combinator, Bracket Abstraction, Standard Combinator Expressions, Typed Lambda Calculus and Reduction Rules (10)
3. Lambda Calculus Semantics: Reduction Machines SECD Machine , Graph Reduction Machine, Lazy/delayed Evaluation, (8)

Text Books

1. Functional Programming : Application and Implementation by Peter Henderson
2. Lambda Calculus, Combinators and Functional Programming by G. Revesz
3. Lambda Calculus and Combinators : An Introduction by J. Roger Hindley and Jonathan P. Seldin

Genetic Algorithms
Syllabus

Prerequisite: Engineering Mathematics and Algorithms
L-T-P: 4-0-0**, Credits:** 4 Type**:** Core Elective Subjects (CEL)

Course Description

This is an introductory course to explain the basic concepts of designing nature-inspired algorithms. All-important nature inspired algorithms are Genetic Algorithm, Particle Swarm Optimization Algorithm and Differential Evolution Algorithm are discussed in this course. It also presents how these algorithms can be used to solve science and engineering-related problems. Application of these algorithms in software testing and cloud computing is also presented in the course in addition to solving optimization problems, these algorithms also help solve prediction-related problems and machine learning problems. These algorithms are designed by mapping some natural phenomena in the form of a computer program. Evolutionary algorithms and swarm intelligence algorithms are mostly used to solve optimization problems. After reading this book, you will be able to design your optimization algorithm and use those problems to solve real-life problems.

Course Objectives

Students undergoing this course are expected to:	
1	At the end of this course, you will be fully familiar with concepts of evolutionary computation and will be able to implement genetic algorithms from scratch and also, utilize them to solve your own optimization problems.
2	Genetic Algorithms (GAs) are members of a general class of optimization algorithms, known as Evolutionary Algorithms (EAs), which simulate a fictional environment based on theory of evolution to deal with various types of mathematical problem, especially those related to optimization
3	Explain the theory behind nature inspired algorithm, apply randomized algorithms for solving optimization problems
4	Genetic algorithms are a type of optimization algorithm, meaning they are used to find the optimal solution(s) to a given computational problem that maximizes or minimizes a particular function.
5	This feature allows them to find solutions to problems that other optimization methods cannot handle due to a lack of continuity, derivatives, linearity, or other features.
6	Able to understand the difference between evolutionary and swarm intelligence-based method

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	How to design randomized algorithm	Identify/Knowledge
CO-2	Difference between deterministic and randomized algorithms	Analyze Identify/Knowledge
CO-3	How to map natural phenomena to design nature inspired algorithm	knowledge/Analyzing/Understanding
CO-4	Applications of Nature Inspired algorithm	Apply
CO-5	Apply Evolutionary Computation Methods to find solutions to complex problems and Summaries current research in genetic algorithms and evolutionary computing.	Apply

Mapping Course Outcomes with Program Outcomes:

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1		3	3	1	2	2				
CO2		2	1							
CO3		3	3	2	2	3			2	
CO4		2	2		2	2			2	2
CO5		3	3		3	3				3
CO (Average) Course Average	**2.2**	**2.6**	**2.4**	**0.6**	**1.8**	**2**			**0.8**	**1**

3-High, 2-Medium, 1-Low

Syllabus

UNIT 1: Introduction

1.1. Basic Terminology
1.2. Optimization Problem
1.3. Single Objective and Multi objective Optimization problems
1.4. Fundamental Differences, Approaches to solve Multi Objective Optimization
1.5. Classical optimization algorithms and Nature Inspired Algorithms

UNIT II: Evolutionary Algorithms

2.1. Binary GA
2.2. Real Parameter GA
2.3. Differential Evolution Algorithm

UNIT III:Swarm Intelligence based Algorithm

3.1 Particle Swarm Optimization Algorithm
3.2. Grey Wolf Optimization Algorithm
3.3Other Popular algorithms like Ant colony, Ant bee colony

UNIT IV:Applications of Nature Inspired Algorithms in Testing

4.1 Test case generation using nature inspired algorithm
4.2 Test case prioritization using nature inspired algorithms.
4.3 Workflow Scheduling in cloud Computing

UNIT V:Application of nature inspired algorithms in machine learning and cloud computing

5.1 Application of Binary GA in Classification
5.2 Workflow Scheduling

Text Books:

1. Genetic Algorithms in Search, Optimization and Machine Learning, David E. Goldberg
2. Genetic Algorithms + Data Structures = Evolution Programs, Michalewicz, Zbigniew

Reference Books:

1. Hardcover, Bauer Richard J, genetic Algorithms and Investment Strategies,Publisher: John Wiley & Sons Inc
2. Sivanandam, Springer India, Introduction to Genetic Algorithms,2013

Network Administration

Prerequisites: Basic knowledge of Computer Networks.

Objective: To learn about the network and how the data route.

Course Description

The course is designed to provide students with essential knowledge and skills that an effective network administrator must possess. It provides an overview of the essential TCP/IP protocols, and discusses how to properly configure and manage the network services based on these protocols (including DNS, DHCP, AD/LDAP directory services, print and file servers, NFS/NIS, and routing services). The course also takes up various issues like Configuration management, accounting management, Fault and disaster management, security management and performance management.

Students undergoing this course are expected to:	
CO-1	Understanding of the Linux operating systems and commands
CO-2	Apply the knowledge of System administration/ Linux in configuration of Networks
CO-3	Describe the Network protocols like TCP, IP, UDP, DNS etc.
CO-4	Know how to configure DHCP, NIS, NFS, LDAP and Samba
CO-5	Apply the knowledge of UNIX networks commands to troubleshoot the network

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**
CO1	1	-	-	2	-	-	-	-	-	2
CO2	1	2	-	2	-	-	-	-	-	2
CO3	1		-	2	-	-	-	-	-	2
CO4	1	2	-	2	-	-	-	-	-	2
CO5	1	1	-	2	-	-	-	-	-	2
CO (Average) Course Average	**1**	**1**	**-**	**2**	**-**	**-**	-	-	**-**	**2**

Course Outline (To be covered in 30 lectures)

1. Introduction, Basic System Administration (3)

2. Windows Installation, Linux Installation and Package Management, Backup and Security, Monitoring and Managing Processes/Daemons, Scripting basics and start-up scripts (8)

3. Unix Networking, Network Protocols - TCP, IP, UDP, NetBIOS, TCP/IP Concepts and Configuration - the basics, Sub netting Implementation, Basic Network Trouble-Shooting and Monitoring Tools (8)

4. Server configuration and management, DHCP, NIS, NFS, LDAP and Samba (6)

5. Apache Web Server with PHP, DNS, BIND and Send mail, Tools like Webmin, Webalizer, and Phpmyadmin; Security and firewall (5)

Text Books

1. TCP/IP Network Administration?, by Craig Hunt,
2. Neural Networks and Learning Machines by S. Haykin
3. *Artificial Neural Networks by Robert J. Schalkoff*
4. Multi-Objective Optimization Using Evolutionary Algorithms by Deb Kalyanmoy
5. Genetic Algorithms + Data Structures = Evolution Programs by Z Michalewicz

Neural Networks

1. Neural Network(OE)

Prerequisites: Multivariate calculus and linear algebra.

Objective:

- gain familiarity with a wide variety of neural network models and their applications
- Develop capabilities for creating and using neural network models.
- develop knowledge of the state-of-the-art in neural networks, and
- Gain some mathematical understanding of neural network models.
- Gain experience in using computational tools such as neural networks to perform computational experiments leading to new theoretical insights.

Course Description

The course is an introduction to neural networks. Neural networks provide a model of computation drastically different from traditional computers. Typically, neural networks are not explicitly programmed to perform a given task; rather, they learn to do the task from examples of desired input/output behavior. The course introduces biological information processing followed by an overview of the most important artificial neural network architectures and algorithms such as perceptrons, backpropagation, Hopfield and Boltzmann networks, self-organizing maps, adaptive resonance theory, reinforcement learning, and neuroevolution.

PO1	Ability to apply knowledge of computing, mathematics, science, engineering and humanities.
PO2	Ability to design and conduct experiments, as well as to analyze and interpret data.
PO3	Ability to analyze a problem, identify and define the computing requirements appropriate to its solutions
PO4	Ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional
PO5	Ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
PO6	Ability to design, implement, and evaluate a software or a software/hardware system, component or process to meet desired needs within realistic constraints such as memory, run time efficiency.
PO7	Understanding of professional, ethical, legal, political security and social issues and responsibilities.
PO8	Ability to communicate effectively with knowledge of contemporary issues.
PO9	Ability to analyze the local and global impact of computing on individuals, organizations, and society.
PO10	Understanding of the need for acquiring an ability to engage in lifelong learning and continuing professional development

CO1	Gain familiarity with a wide variety of neural network models and their applications
CO2	Develop capabilities for creating and using neural network models.
CO3	Develop knowledge of the state-of-the-art in neural networks
CO4	Gain some mathematical understanding of neural network models.
CO5	Gain experience in using computational tools such as neural networks to perform computational experiments leading to new theoretical insights.

Course Outline (To be covered in 30 lectures)

1. Introduction, Brain Physiology, Neuron Model and Network Architectures (4)
2. Nonlinear dynamical system theory (6)
3. The Hopfield Model, Spin Glasses, Stochastic Neural Networks, Boltzmann Machine (8)
4. Multilayer Feedforward Networks For Supervised Learning(6)
5. Unsupervised and Competitive Learning Algorithms, Bifurcating Neural Networks (6)

Text Books

1. Neural Networks: A Comprehensive Foundation by S. Haykin,
2. Neural Networks and Learning Machines by S. Haykin
3. Artificial Neural Networks by Robert J. Schalkoff
4. Multi-Objective Optimization Using Evolutionary Algorithms by Deb Kalyanmoy
5. Genetic Algorithms + Data Structures = Evolution Programs by Z Michalewicz

Cos and Pos mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	3	2	3	2	2	1	1	-	-	-
CO2	2	3	2	3	1	2	1	-	1	-
CO3	3	2	2	2	2	1	2	1	-	-
CO4	3	3	3	2	3	2	1	-	-	-
CO5	3	3	1	2	1	1	2	3	2	3
avg	2.8	2.6	2.2	2.2	1.8	1.4	1.4	0.8	0.6	0.6

Course Outline (To be covered in 30 lectures)

1. Introduction, Brain Physiology, Neuron Model and Network Architectures (4)
2. Nonlinear dynamical system theory (6)
3. The Hopfield Model, Spin Glasses, Stochastic Neural Networks, Boltzmann Machine (8)
4. Multilayer Feedforward Networks For Supervised Learning(6)
5. Unsupervised and Competitive Learning Algorithms, Bifurcating Neural Networks (6)

Text Books

1. Neural Networks: A Comprehensive Foundation by S. Haykin,
2. Neural Networks and Learning Machines by S. Haykin
3. *Artificial Neural Networks by Robert J. Schalkoff*
4. Multi-Objective Optimization Using Evolutionary Algorithms by Deb Kalyanmoy
5. Genetic Algorithms + Data Structures = Evolution Programs by Z Michalewicz

Service Oriented Software Engineering
Syllabus

Objective:

1. To introduce the idea of service-oriented architectures
2. To explain the notion of a reusable service, based on web service standards, that provides a mechanism for inter-organisational computing;
3. To describe the service engineering process that is intended to produce reusable web services.
4. To introduce service composition as a means of application development;
5. To show how business process models may be used as a basis for the design of service-oriented systems

Prerequisites: Software Engineering, Service-oriented analysis and design, Service oriented Modeling.

Course Description

Service oriented software development paradigm is becoming the delivery model by all major IT companies. This course is intended to introduce the students with this paradigm. In this course students shall study the fundamentals of Service Oriented Software Engineering. Prerequisite for this course is course on Software Engineering.

Course Outcomes (COs)

CO#	Course Outcomes
CO1	To understand the concepts of Service orientation
CO2	To learn and apply knowledge of Service oriented Software architecture
CO3	To analyze requirements &design process
CO4	To describe service testing and estimation models
CO5	To summarize cloud based services models

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)

2. Moderately (Medium)

3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	3	1	3	3	3	3	–	–	2	–
CO2	3	3	2	3	3	3	–	–	–	–
CO3	3	3	2	2	2	2	–	–	–	–
CO4	3	2	3	3	3	3	–	–	–	–
CO5	3	2	2	2	2	3	–	–	2	–
CO (Average) Course Average	**3.0**	**2.2**	**2.4**	**2.6**	**2.6**	**2.8**	**0.0**	**0.0**	**0.8**	**0.0**

Course Outline (To be covered in 30 lectures)

1. Concepts of Service orientation (8)
2. Service oriented Software architecture concepts   (5)
3. Requirements Analysis & Design Process (7)
4. Service Testing and Estimation models (6)
5. Cloud based services models (4)

Text Books

1. Service Oriented Architecture – Concept Technology & Design by Thomas Earl
2. Enterprise SOA – Designing IT for Business Innovation by Woods & Mattem
3. Web Service Essentials, Eiban Cerami, O'Reilly

XML Based Applications

Objective:

1. To familiarize students with various XML based applications with the help of case studies.
2. To be able to develop new applications using XML schema.

Prerequisites: Fundamental concepts of XML including document and language creation and implementation, XML Schema

Course Description

This course introduces students to the basic concepts of the extensible markup language (XML). XML has made a major impact in almost every aspect of software development. Designed as an open, extensible, self-describing language, it has become the world-wide standard for data and document delivery on the Web. Students will be instructed as to the purpose of an XML document and what it consists of, in how a Document Type Definition (DTD) or schema is used to validate an XML document and the extensible style language (XSL) to transform XML documents into HTML/XHTML. XML-related technologies continue to develop, to enable validation, navigation, transformation, linking, querying, description, and messaging of data. Students would be exposed to such wide range of application domains.

Students undergoing this course are expected to:	
CO-1	Develop XML files using concept of XML DOM, XSLT and XML Namespaces.
CO-2	Implement programs to validate the XML Documents with respect to given XML schemas and DTD
CO-3	Develop an interactive website using jQuery or AJAX
CO-4	Develop solution to complex problems using appropriate web services and content management software
CO-5	Develop pages using suitable client side and server side web technologies.

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	2	-	-	1	-	-	-	2	-	-
CO2	2	2	-	1	-	2	-	-	-	-
CO3	2		-	1	-	2	-	-	-	1
CO4	2		-	1	-	-	-	2	-	-
CO5	1	1	2	1	-	-	-	1	-	1
CO (Average) Course Average	**1.8**	**0.6**	**0.4**	**1**	**-**	**0.8**	-	1	**-**	**0.4**

Course Outline (To be covered in 30 lectures)

1. Emerging Technologies; XML Documents: Syntax, Well formed and Valid; CCS and XHTML;Document Type Definition(DTD); XML Schema : XSD, XDR, Examples; JavaScript (12)
2. SAX and DOM Parser and APIs, Example of API usage; XPATH, XLink, Xpointer; XSL: XSLT (10)
3. Applications: RDF and RDFS, JENA API, Case Study (8)

Books:

1. XML The Microsoft Way By Peter G. Aitken
2. Learning XML By Erik T. Ray and Christopher R. Maden
3. XML How to Program By Harvey M. Deitel, Paul J. Deitel, Tem R. Nieto, Ted Lin and Praveen Sadhu

Real Time Systems
Syllabus

Prerequisite: Operating System and Computer Architecture
L-T-P: 3-0-0**, Credits:** 3 **Type:** Core Essential Subjects (CES)

Course Description

This course content application of different type real time system, Precedence constraints and Data dependency, Real-time Scheduling: Scheduling hierarchies, Priority driven scheduling of periodic tasks, Fixed priority versus Dynamic Priority Algorithms, Optimality of RM and DM scheduling, Uncontrolled Priority Inversion.

Course Objectives

Students undergoing this course are expected to:	
1	Students will be able to understand the basics of Real Time systems
2	Students will be able to analyze Periodic Task model and scheduling of the same
3	Student will be able to understand the scheduling concepts of a periodic Task
4	Student will be able comprehend the scheduling of real time tasks with resource constraints.

Course Outcomes

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO-1	To obtain the basic **knowledge** about the Real Time System and its applications, Types of Real time systems, Processor and Resource models.	Identify/Knowledge
CO-2	To obtain the **basic knowledge** of Real Time system Parameters, Task Models, Scheduling Hierarchies and Types of Scheduling.	Understanding/ Comprehension
CO-3	To have an **understanding** of Priority Driven scheduling of Periodic Tasks, Static and Dynamic scheduling of Periodic tasks along with feasibility analysis of the same.	Understanding
CO-4	To **understand** and study the effects of practical factors in scheduling of Periodic Task, Scheduling A periodic task in presence of Periodic one with Server based approaches.	Understanding /Analyzing
CO-5	To **comprehend** about the Resource allocation and access in presence of Periodic and A periodic Tasks, Avoiding Deadlock and understanding priority ceiling protocol	Comprehension

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	2	2	2	-	-	-	2	-	-	-
CO2	1	2	3	2	3	2	1	-	-	-
CO3	2	3	2	2	2	2	3	-	-	-

CO4	1	2	2	3	2	2	3	-	-	-
CO5	1	2	2	3	2	2	1	-	-	-
CO (Course Average)	**1.4**	**2.5**	**2.5**	**2.0**	**1.8**	**1.6**	2.0	0.0	**0.0**	**0.0**

Syllabus

UNIT 1: Introduction: Applications, different type of real-time systems: Hard Real time system, Soft Real time Systems, reference models, Processor and Resources, Precedence constraints and Data dependency, Functional Parameters.

UNIT II: Release Time, Deadline, Periodic Task Model, Hyper-period, Aperiodic Task Model, Sporadic Task Model, Real-time Scheduling: Scheduling hierarchies, commonly used scheduling approaches, Clock Driven Approach, Weighted Round Robin Approach, Priority Driven Approach.

UNIT III: Priority driven scheduling of periodic tasks, Dynamic versus Static System, Off-Line Versus Online Scheduling, Fixed priority versus Dynamic Priority Algorithms, Maximum Schedulable Utilization, Rate Monotonic Scheduling (RM), deadline Monotonic Scheduling (DM), Earliest Deadline first Scheduling (EDF), Least Slack Time first Scheduling (LST), Tick Scheduling, (10).

UNIT IV: Optimality of RM and DM scheduling, Practical Factors, Scheduling of aperiodic and sporadic tasks: Conventional Approaches, Deferrable server, sporadic servers, constant utilization, total utilization and weighted fair queue servers, slack stealing approaches. (6)

UNIT IV: Resource access control: NPCS, Priority inheritance protocol, Uncontrolled Priority Inversion, Priority ceiling protocol, Slack based ceiling protocol, Use of Priority Ceiling protocol in Dynamic Priority system, Multiprocessor priority ceiling protocol. (6)

Text Books

1. Real Time System, Jane W.S. Liu,1999.
2. Laplante, Phillip A. “Real-time systems design and analysis - an engineer's handbook.” (1992).

Reference Books

1. Hard Real Time Computing Systems Predictablle Scheduling Algorithms and applications By Giorgio C. Buttazzo ,2011
2. Real Time Design Patterns: Robust Scalable Architecture for Real Time System by Bruce Powel Douglass ,2003.
3. Real Time System: Scheduling, Analysis and Verification by Albert M. K. Cheng,2003.

IoT: Architecture and Protocol
Syllabus

Prerequisite: Operating System, Computer Networks and Wireless Networks
L-T-P: 3-0-0**, Credits:** 3 **Type:** Core Elective (CEL)

Course Description

IOT architectural model, IOT reference model Domain, information, functional and communication model, IOT in practice: IOT in logistics, health, environmental monitoring, smart cities. IOT reference architecture functional, information, deployment and operational views.

Course Objectives

Students undergoing this course are expected to:	
1	Understand the different architecture of IOT
2	Explore and build different IOT based application
3	Analyze real world problems and map it into IoT based application
4	Understand the concept of application protocols used in IoT

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	***Comprehend*** the essentials of IoT and its applications	Comprehension
CO-2	Learn and ***understand*** the concepts of IoT Standardized Architecture and IoT reference architecture and able to create customized IoT architectures	Understanding:
CO-3	***Possess*** working knowledge in current technologies	Identify/Knowledge
CO-4	***Analyze*** various IoT Application layer protocols and apply IP based protocols for IoT.them in Python.	Analyzing
CO-5	Design IoT-based systems for real-world problems	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	3	0	3	2	1	0	1	1	0	2
CO2	2	1	2	2	2	1	1	1	0	1
CO3	2	1	2	1	2	1	1	1	0	2
CO4	2	3	2	2	1	3	2	1	1	1
CO5	2	2	2	2	2	3	2	1	3	2

CO (Average) Course Average	2	1	2	2	2	2	1	1	1	2

Syllabus

UNIT 1: Introduction to IoT, Applications of IoT, Use cases of IoT, Scope–Sensors for IoT Applications–Structure of IoT– IoT Map Device.

UNIT 2: Drivers Behind New Network Architectures, Comparing IoT Architectures, IoT World Forum (IoTWF) Standardized Architecture, A Simplified IoT Architecture, The Core IoT Functional Stack, IoT Data Management and Compute Stack.

UNIT 3: The IoT Architectural Reference Model as Enabler, IoT in Practice: Examples: IoT in Logistics and Health, IoT Reference Model: Domain, information, functional and communication models. IoT Reference Architecture: Architecture, Functional, information, deployment and operation views; SOA based Architecture, API-based Architecture, OPENIoT Architecture for IoT/Cloud Convergence.

UNIT 4: Application Protocols for IoT: CoAP, MQTT, UPnP, XMPP. SCADA, WebSocket; IP-based protocols: 6LoWPAN, RPL; Authentication Protocols; IEEE 802.15.4.

UNIT 5: Case study: Cloud-Based Smart-Facilities Management, Healthcare, and Environment Monitoring System, Public Safety: Emergency Response IoT Architecture, IoT Public Safety Information Processing, School Bus Safety.

Text Books

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, C, “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, Cisco Press, 2017.
2. Bassi, Alessandro, et al, “Enabling things to talk”, Springer-Verlag Berlin An, 2016.
3. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, CISCO Press, 2017.

Reference Books

1. Hersent, Olivier, David Boswarthick, and Omar Elloumi. The internet of things: Key applications and protocols. John Wiley & Sons, 2011.
2. Buyya, Rajkumar, and Amir VahidDastjerdi, eds. Internet of Things: Principles and paradigms. Elsevier, 2016.
3. Dillon, Roberto, ed. The digital gaming handbook. CRC Press, 2020

E-Commerce

Prerequisites: Computer Information Systems , Business Data Management , System Analysis and Design .

Objective:

- have an understanding of essential e-Commerce concepts and technologies and skills related to the management and application of e-Commerce and e-Business approaches .
- have an understanding of the technological, capital and social infrastructure for commercial activities such as buying and selling, marketing and advertising, supply-chain management etc.
- have hands on, real-life experience with electronic commerce applications .
- be able to define and explain the main issues facing businesses engaged in the planning and implementation of e-Business strategies .
- identify and define the main e-Business models currently being adopted by organizations
- have an understanding and ability to assess the strategic relevance of e-Commerce in shaping both inter-organisational relationships and intra-organisational structures and processes
- critically evaluate the design of e-Business sites and discuss human, organisational and social implications of electronic commerce

Course Description

The growth of the Internet continues to have a tremendous influence on business. Companies and organizations of all types and sizes are rethinking their strategies and how they run their operations. This new course in the Temple E-Marketing program challenges students to explore the realities and implications of e-commerce from a marketer's perspective. Business-to-consumer (B2C) and business-to-business (B2B) e-commerce markets are examined. The course introduces students to a wide range of electronic commerce issues for marketers, as a foundation for continual learning in the dynamic e-commerce environment.

Course Outcomes

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO-1	Have an understanding of essential e-Commerce concepts and technologies and skills related to the management and application of e-Commerce and e-Business approaches	Understanding/ Comprehension
CO-2	Have an understanding of the technological, capital and social infrastructure for commercial activities such as buying and selling, marketing and advertising, supply-chain management etc.	Understanding/ Knowledge
CO-3	Have hands on, real-life experience with electronic commerce applications.	Analyzing/Evaluating
CO-4	Be able to define and explain the main issues facing businesses engaged in the planning and implementation of e-Business strategies.	Identify/Knowledge
CO-5	Identify and define the main e-Business models currently being adopted by organizations	Identify/Analyzing
CO-6	Have an understanding and ability to assess the strategic relevance of e-Commerce in shaping both inter-organisationalrelationships and intra-organisational structures and processes	Identify /Analyzing

CO-7	Critically evaluate the design of e-Business sites and discuss human, organisational and social implications of electronic commerce	Evaluating/Analyzing

Mapping Course Outcomes with Program Outcomes:
Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	1	1	2	1	2	1	1	1	2	2
CO2	2	2	2	1	2	1	2	1	1	1	2	2
CO3	2	2	2	1	2	2	2	1	1	2	2	2
CO4	2	3	3	3	3	2	2	1	2	2	1	2
CO5	2	2	2	2	2	3	2	1	3	2	2	2
CO6	2	2	2	2	2	2	2	2	2	2	2	2
CO7	2	2	2	3	2	2	2	3	2	2	2	2
CO (Average)	2	2	2	2	2	2	2	1	2	2	2	2

Course Outline (To be covered in 30 lectures)

1. Introduction to e-Commerce and Network Infrastructure for e-commerce. [4]
2. E-commerce Models, e-Advertising & Marketing [6]
3. Electronic Payment Systems and Electronic Data Exchange [6]
4. E-commerce Security [4]
5. e-CRM [6]
6. Mobile Commerce [4]

Text Books

1. Introduction to E-commerce by Jeffrey F.Rayport & Bernard J.Jaworski
2. Frontiers of E-commerce by Kalakota & Winston
3. E-Commerce- Strategy technologies and Applications by David Whiteley
4. E-Commerce-Concepts, Models & Strategies by C.S.V. Murthy
5. E-Commerce by Perry

Professional Elective III (Pool-2)

Distributed & Parallel Algorithms
Syllabus

Prerequisite: Design and Analysis of Algorithms, Computer Architecture, Operating System
L-T-P: 4-0-0, **Credits:** 4 **Type:** Core Electives Subjects (CEL)

Course Description

This course is designed to examine the fundamental principles of distributed and parallel systems, and provide students hands-on experience in developing distributed and parallel algorithm. While we still look at issues in distributed operating systems, this course will address distributed and parallel systems in a broader sense.

Course Objective

Students undergoing this course are expected to:	
1	To analyze the parallelism.
2	To identify the conditions of parallelism.
3	To study different parallel interconnection systems.
4	To address distributed and parallel systems in a broader sense

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Understand** the requirements for parallel and distributed systems and how they can be used to build concurrent and distributed systems.	Understanding/Comprehension
CO-2	To **learn** and **apply** knowledge of parallel and distributed computing techniques and Methodologies.	Creating
CO-3	**Apply** the basic algorithmic techniques and design algorithms in shared memory as well as distributed memory environment.	Applying
CO-4	**Understand** the memory hierarchy and cost-performance tradeoffs in parallel and distributed systems.	Identify/Knowledge
CO-5	To gain experience in the design, development, and performance **analysis** of parallel and distributed applications.	Analyzing

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	3	1	3	3	3	3	–	–	2	–
CO2	3	3	2	3	3	3	–	–	–	–
CO3	3	3	2	2	2	2	–	–	–	–
CO4	3	2	3	3	3	3	–	–	–	–

CO5	3	2	2	2	2	3	–	–	2	–
CO (Average) Course Average	**3.0**	**2.2**	**2.4**	**2.6**	**2.6**	**2.8**	**0.0**	**0.0**	**0.8**	**0.0**

SYLLABUS

UNIT I: Parallel Algorithms: Parallel Programming Models: Shared-memory model (PRAM, MIMD, SIMD), network model (line, ring, mesh, hypercube), performance measurement of parallel algorithms.

UNIT II: Algorithm Design Techniques for PRAM Models: Balancing, divide and conquer, parallel prefix computation, pointer jumping, symmetry breaking, pipelining, accelerated cascading.

Algorithms for PRAM Models: List ranking, sorting and searching, tree algorithms, graph algorithms, string algorithms.

Algorithms for Network Models: Matrix algorithms, sorting, graph algorithms, routing, Relationship with PRAM models.

Parallel Complexity: Lower bounds for PRAM models, the complexity class NC, P- completeness.

UNIT III: Distributed Algorithms: Basic concepts. Models of computation: shared memory and message passing systems, synchronous and asynchronous systems. Logical time and event ordering. Global state and snapshot algorithms, clock synchronization.

Distributed Operating Systems: Mutual exclusion, deadlock detection Classical Algorithms: Leader election, termination detection, distributed graph algorithms.

UNIT IV: Fault Tolerance and Recovery: basic concepts, fault models, agreement problems and its applications, commit protocols, voting protocols, check pointing and recovery, reliable communication.

Security and Authentication: basic concepts, Kerberos. Resource sharing and load balancing.

Text Books

1. Quinn, Michael J. Parallel computing theory and practice. McGraw-Hill, Inc., 1994.
2. Tanenbaum, Andrew S. Distributed operating systems. Pearson Education India, 1995.
3. JéJé, Joseph. "An introduction to parallel algorithms." Reading, MA: Addison-Wesley 10 (1992): 133889.
4. Xavier, C., and Sundararaja S. Iyengar. Introduction to parallel algorithms. Vol. 1. John Wiley & Sons, 1998.
5. MukeshSinghal and Niranjan G. Shivaratri. 1994. Advanced Concepts in Operating Systems. McGraw-Hill, Inc., USA.

Reference Books

1. Hughes, Cameron, and Tracey Hughes. Parallel and distributed programming using C++. Addison-Wesley Professional, 2004.
2. Quinn, Michael J. "Parallel programming." TMH CSE 526 (2003): 105.
3. Attiya, Hagit, and Jennifer Welch. Distributed computing: fundamentals, simulations, and advanced topics. Vol. 19. John Wiley & Sons, 2004.
4. Wilkinson, Barry, and Michael Allen. Parallel Programming-Techniques and Applications using a network of parallel workstations & Computers. Andi Publisher, 2010.

Gaming and Animation
Syllabus

Prerequisites: This course requires general familiarity with computer concepts, an interest in and experience with games, and a vivid imagination.

Objective: This course gives students a solid understanding of designing, modeling and implementing a game.

Course Description

The purpose of this course is to give students a thorough understanding of computer animation and gaming. The course introduces camera and vehicle animation, parent/child hierarchies, character rigging, character animation, facial animation, lip syncing, physical simulations, motion capture for gaming.

After undergoing this course students will be able to–

CO1: generate animated sequences from the development of the original concept through design to final film or video production.

CO2: integrate the concepts, principles and theories involved in the physics of animation in all aspects of drawing.

CO3: create 2D and 3D characters and environments that reflect the integration of graphic clarity, design principles, performance principles and theoretical constructs.

CO4: visually conceptualize in a clear and concise way the artistic direction for a game or specific game assets using traditional rendering methods or digitally facilitated rendering methods.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	1	3	3	2	1.5	1	-	-	-	-
CO2	1.5	2	3	3	2.5	1	1	-	-	-
CO3	2	1.5	1.5	2	2	2	1	-	-	-
CO4	2	2	3	3	2	2	1	-	1	-

Course Outline (To be covered in 30 lectures)

1. Introduction, Fundamental Principles of Animation and gaming (6)
2. Rigging & Posing Techniques, Fundamentals of Character Animation, Facial Animation and Lip Sync Techniques (8)
3. Fundamentals of Motion Capture, Principles of Motion Simulation (6)
4. Game design principles and processes (8)

Text Books

1. Fundamentals of Game Design. By E. Adams.
2. The Art of Game Design by J. Schell
3. Computer Animation: Algorithms and Techniques by Rick Parent

Information Retrieval
Syllabus

Prerequisites:

Basic knowledge of web design, Basic Programming, data structures, Algorithms, Basic linear algebra, Basic statistics.

Objective:

To give students a solid understanding of:

- the genesis and variety of information retrieval situations.
- the variety of information retrieval models and techniques.
- design principles for information retrieval systems.
- methods for implementing information retrieval systems.
- characteristics of operational and experimental information retrieval systems.
- methods and principles for the evaluation of information retrieval systems.

Course Description

This course will cover traditional material, as well as recent advances in Information Retrieval (IR). The course includes the study of indexing, processing, and querying textual data basic retrieval models, algorithms, and IR system implementations. The course will also address advanced topics in IR, including Natural Language Processing techniques, and Web agents.

CO1: The students are able to understand the basics of information theory, vector-space, probability models and similarity measures of query and document.

CO2: The students are able to implement clustering algorithms and perform regression analysis to identify the information content in data.

CO3: The students are able to estimate the probability of relevance and understand natural language systems to build semantic networks for text.

CO4: The students are able to evaluate the performance of cross language information, query, and document and phrase translation methods.

CO5: The students are able to design the method to build inverted index and apply the various models to desired applications.

Program Outcomes (POs):

Students in the B Tech CSE program will attain:

PO1	Ability to apply knowledge of computing, mathematics, science, engineering and humanities.
PO2	Ability to design and conduct experiments, as well as to analyze and interpret data.
PO3	Ability to analyze a problem, identify and define the computing requirements appropriate to its solutions
PO4	Ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional
PO5	Ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
PO6	Ability to design, implement, and evaluate a software or a software/hardware system, component or process to meet desired needs within realistic constraints such as memory, run time efficiency.
PO7	Understanding of professional, ethical, legal, political security and social issues and

	responsibilities.
PO8	Ability to communicate effectively with knowledge of contemporary issues.
PO9	Ability to analyze the local and global impact of computing on individuals, organizations, and society.
PO10	Understanding of the need for acquiring an ability to engage in lifelong learning and continuing professional development

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	M	H	M	M	M	H	L	-	-
CO2	H	H	H	H	H	H	L	M	M	-
CO3	M	M	M	H	H	H	M	L	L	L
CO4	M	M	H	H	H	H	L	M	L	L
CO5	M	M	H	H	H	H	M	M	H	H

Course Outline (To be covered in 30 lectures)

1. Introduction to IR models and methods, Text analysis / Web spidering Text properties (5)
2. Vector-based model, Boolean model, Probabilistic model, other IR models; IR evaluation and IR test collections; Relevance feedback, query expansion (8)
3. Web search: link based and content based; Query-based and content sensitive link analysis; Search engine technologies (8)
4. Text classification and clustering; Question answering on offline and online collections (5)
5. Personalized IR, Cross-language IR, Web 2.0, (4)

Text Books

1. Introduction to Information Retrieval by Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze (available online)
2. Information Retrieval: Algorithms and Heuristics. By D.A. Grossman, O. Frieder
3. *Readings in Information Retrieval by K.Sparck Jones and P. Willett*

Pattern Recognition
Syllabus

Prerequisites: Analysis of algorithms, Calculus, Introductory Statistics, Linear Algebra.

***Objective**:*

- Learn the fundamental concepts and applications of pattern recognition.
- Learn the concepts of Bayes decision theory.
- Understand the concepts of linear and nonlinear classifiers.
- Understand the concepts of feature selection and generation techniques.
- Understand the concepts of supervised learning and system evaluation.
- Develop some applications of pattern recognition.

Course Description

The emphasis of the course is on algorithms used for pattern recognition. Pattern Recognition is assigning a meaningful or classifying label to the elements of the input data. It uses the concepts of classification and clustering to separate the interclass elements. This information can then be used to classify or recognize new data using supervised or unsupervised learning methods and classifiers such as Support Vector Machine, Hidden Markov Model and Linear Discriminate Analysis. Pattern recognition has several important applications in the fields of data mining, artificial intelligence, networking and image processing. The prerequisites of the course are basic knowledge of statistics and linear algebra along with the concepts of probability theory.

CO1: A good knowledge of Bayesian decision theory and Bayesian learning.

CO2: Fundamental understanding of classifiers such as linear discriminant function, quadratic discriminant function, nearest neighbor rule, neural network and SVM.

CO3: A good understanding of feature selection algorithms.

CO4: Ability to evaluate the performance of various classifiers on real-world datasets.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	1	2	2	2.5	1.5	1	-	-	-	-
CO2	1.5	2	2.5	2	2.5	1	1	-	-	-
CO3	2	1.5	1.5	2	2	2	1	-	-	-
CO4	2	2	3	3	2	2	1	-	-	-

Course Outline (To be covered in 30 lectures)

1. Introduction to Pattern Recognition, Feature Detection, Classification, Decision Theory, ROC Curves, Likelihood Ratio Test, Linear and Quadratic Discriminants, Fisher Discriminant, Sufficient Statistics, Coping with Missing or Noisy Features, Template-based Recognition, Feature Extraction, Eigenvector and Multilinear Analysis (10)

2. Training Methods, Maximum Likelihood and Bayesian Parameter Estimation, Linear Discriminant/Perceptron Learning, Optimization by Gradient Descent, Support Vector Machines, K-Nearest-Neighbor Classification (6)

3. Non-parametric Classification, Density Estimation, Parzen Estimation,Unsupervised Learning, Clustering, Vector Quantization, K-means, Mixture Modeling, Expectation-Maximization (6)

4. Hidden Markov Models, Viterbi Algorithm, Baum-Welch Algorithm, Linear Dynamical Systems, Kalman Filtering, Decision Trees, Multi-layer Perceptrons, Reinforcement Learning with Human Interaction (8)

Text Books

1. Pattern Classification by Richard O. Duda, Peter E. Hart and David G. Stork

2. Pattern Recognition and Machine Learning by C. M. Bishop

3. Pattern Recognition by S. Theodoridis and K. Koutroumbas

Semantic Web

Syllabus

Prerequisites: Basic Web technology like html.

Objective: The aim of the course is to make the students familiar with the Semantic Web, with technologies used on the Semantic Web, and with applications using Semantic Web technologies. The course will focus on the theoretical background of various languages on the Semantic Web such as RDF, SPARQL, OWL, and F-Logic (Programming), and the practical use of these languages on the Semantic Web. In addition, the course will focus on important application areas for Semantic Web technology, namely Web Services and Life Sciences.

Course Description

This course introduces techniques that are useful stand-alone and can be integrated for building a semantic web. It will review XML with Document Type Definitions and Schemas; transformation/inference rules in XSLT, metadata with RDF (Resource Description Framework); metadata taxonomies with RDF Schema; description logic and the W3C ontology language OWL 2; as well as integrating these techniques for ontology/rule-based multi-agent systems. Students may note that besides enabling quick and accurate web search, semantic web may also allow the development of intelligent internet agents and facilitate communication between multitudes of heterogeneous web-accessible devices.

CO 1: The students are able to understand the concepts of web search and methods to response the user queries.

CO 2: The students are able to compute the similarity of pairs of documents, process and analyze streaming data.

CO 3: The students are able to discovery of association rules in text, social network analysis, web structure mining and web usage mining.

CO 4: The students are able to design and implement a crawler application to collect and index documents from the web.

CO 5: The students are able to analyze text and apply the knowledge and techniques of web search to solve the many real world application problems.

Program Outcomes (POs):

Students in the B Tech CSE program will attain:

PO1	Ability to apply knowledge of computing, mathematics, science, engineering and humanities.
PO2	Ability to design and conduct experiments, as well as to analyze and interpret data.
PO3	Ability to analyze a problem, identify and define the computing requirements appropriate to its solutions
PO4	Ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional
PO5	Ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
PO6	Ability to design, implement, and evaluate a software or a software/hardware system, component or process to meet desired needs within realistic constraints such as memory, run time efficiency.
PO7	Understanding of professional, ethical, legal, political security and social issues and

	responsibilities.
PO8	Ability to communicate effectively with knowledge of contemporary issues.
PO9	Ability to analyze the local and global impact of computing on individuals, organizations, and society.
PO10	Understanding of the need for acquiring an ability to engage in lifelong learning and continuing professional development

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	H	H	M	M	M	H	L	L	-
CO2	H	M	M	H	M	H	L	M	L	-
CO3	M	M	M	M	H	H	H	L	L	L
CO4	M	M	H	H	M	M	L	M	L	L
CO5	M	M	M	H	H	H	H	M	H	H

Course Outline (To be covered in 30 lectures)

1. Review of XML; Meta-model and Meta-data, RDF & RDFS; OWL; Ontology Engineering and tools (12)
2. Description Logic(DL); Programming with DL; Example Application (12)
3. Knowledge Acquisition and Management System,   (6)

Text Books

1. A Semantic Web Primer by Antoniou, Grigoris and Frank van Harmelen
2. The Description Logic Handbook: Theory, Implementation and Applications by Franz Baader, Deborah L. Guinness, Daniele Nardi, and Peter F. Patel-Schneider (Eds.)
3. An Introduction to Description Logic by Daniele Nardi and Ronald J. Brachman

Software Metrics & Quality Assurance
Syllabus

Prerequisites: Software engineering process, analysis, designs etc.

Objective: This course introduces concepts, metrics, and models in software quality assurance. The course covers components of software quality assurance systems before, during, and after software development. It presents a framework for software quality assurance and discuss individual components in the framework such as planning, reviews, testing, configuration management, and so on. It also discusses metrics and models for software quality as a product, in process, and in maintenance. The course will include case studies and hands on experiences. Students will develop an understanding of software quality and approaches to assure software quality.

Course Description

In this course students will study the foundational concepts of measurement of various aspects of software during the entire course of its development. The course takes up various existing metrics and tools that measure various activities of the software development. Topics such as Property-oriented measurement, Meaningfulness in measurement, Measurement quality, Measurement process, Scale, Measurement validation, Object-oriented measurement are covered. Students may note that the course is credited only after having undergone Software Engineering.

COURSE OUTCOMES: After completion of the course:

CO1	Student shall be able to understand technical knowhow of Software Metrices. and apply various software metrics, which determines the quality level of software
CO2	Identify and evaluate the quality level of internal and external attributes of the software product
CO3	Compare and Pick out the right reliability model for evaluating the software
CO4	Design new metrics and reliability models for evaluating the quality level of the software based on the requirement
CO5	Evaluate the reliability of any given software product

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	3	-	3	-	-	-	-	-	-	-
CO2	2	2	2	2	-	-	-	-	1	1
CO3	3	2	3	3	2	-	-	-	1	1
CO4	2	3	2	-	2	2	-	-	-	-
CO5	-	1	1	-	1	-	1	-	1	1
Average	2	1.6	2.2	1	1	0.4	0.2	-	0.6	0.6

Course Outline (To be covered in 30 lectures)

1. The state of IT project management & basics of measurement (6)
2. Measuring internal product attributes: size and structure (6)
3. Measuring cost and effort (6)
4. Measuring external product attributes: Quality & Reliability (6)
5. Software test metrics (6)

Text Books

1. Software Metrics: A Rigorous and Practical Approach by N.E. Fenton and S.L. Pfleeger
2. Metrics and Models in Software Quality Engineering by Stephen H. Kan
3. Software Project Management in practice by Pankaj Jalote
4. Software Project Management by Bob Hughes and Mike Cotterell

Software Testing
Syllabus

Prerequisites: Software engineering and Software project management.

Objectives: To understand the fundamental of software testing, different approaches to testing, managing test cases and different testing strategies.

Course Description

In this course students shall study the fundamentals of testing, various approaches to testing, managing test cases and various testing strategies. Students may note that the course is credited only after having undergone Software Engineering and/or Software Project Management.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Explain the fundamentals of software testing. Software Testing Models.	Identify/Knowledge
CO-2	Develop skills to identify the correct approach of software testing in different applications and apply them to test the software working.	Identification/ Applying
CO-3	Learn to develop different test cases for applying in software testing and also learn to use these test cases to test the software working.	Creation/ Applying
CO-4	Study and implementation of different testing strategies like preventive, reactive approach, analytical, heuristic, cofiguration management and their use with example in software testing.	Analyzing/ Use
CO-5	Apply the mutation testing and its use in object oriented software and use of software testing tools. Also learn advantages to use mutation testing in object oriented software.	Knowledge/ Applying

Mapping Course Outcomes with Program Outcomes:

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**
CO1	3	-	-	-	-	1	-	-	-	-
CO2	3	2	-	2	2	3	-	-	-	-
CO3	1	3	2	3	1	3	-	-	-	-
CO4	2	2	3	-	2	3	-	-	-	-
CO5	-	3	2	2	-	2	-	-	-	-
CO (Average) Course Average	**2**	**2**	**1**	**1**	**1**	**2**	-	-	**-**	**-**

Course Outline (To be covered in 30 lectures)

1. Fundamentals of Testing and its current state of art (8)
2. Various approaches to Testing (6)
3. Test planning and Management (6)
4. Test Strategies - Preventive, Reactive Approach, Analytical, Heuristic, Configuration Management (6)
5. Mutation Testing & Testing Object Oriented Software (4)

Text Books

1. Software Testing Techniques by Borris Beizer
2. Software Testing – A Craftman's Approach by Paul C. Jorgensen
3. Software Testing by Hambling, Samaroo & Williams.
4. Software Testing Practice: Test Management by Spillner, Rossner, Winter & Linz

Theory of Virtualization
Syllabus

Prerequisites: Operating system and Computer network.

Objectives: Understanding the skills and knowledge related to the concepts and principles of virtualization, the mechanisms and techniques of building virtualized system and virtualization-enabled processing scenario.

Course Description

This course provides description of the concepts of virtualization and the properties of virtualization that make it a powerful technology. It contrast different forms of virtualization and focus on system level virtualization which has become very popular lately in the computer industry. It describes various architectures for implementing system-level virtualization. Upon completion of this course, students will possess the skills and knowledge related to the concepts and principles of virtualization, the mechanisms and techniques of building virtualized systems, as well as the various virtualization-enabled computing paradigms. Further, they will also gain knowledge about some State-of-the-art virtualization software and systems through their course projects. The basic courses on Operating System and Computer Networks are prerequisites.

Students undergoing this course are expected to:	
CO-1	Understanding of the concepts of virtualization and the properties of virtualization.
CO-2	Apply the knowledge of Hardware/Server virtualization in configuration of virtualization-enabled processing scenario.
CO-3	Know how to configure Network virtualization.
CO-4	Apply the knowledge of Virtual machines to building virtualized systems, as well as the various virtualization-enabled computing paradigms.

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	2	-	1	1	1	-	-	-	-	-
CO2	2	2	2	2	1	2	-	-	-	-
CO3	2	2	2	1	2	2	-	-	-	-
CO4	2	2	2	2	1	2	-	-	-	-
CO (Average) Course Average	**2**	**1.5**	**1.75**	**1.5**	**1.25**	**1.5**	0	0	**0**	**0**

Course Outline (To be covered in 30 lectures)

1. Introduction, Overview of virtualization (8)
2. Hardware/Server virtualization (8)
3. Network virtualization (8)
4. Virtual machines (6)

Text Books

1. Virtual Machines: Versatile Platforms for Systems and Processes by James E. Smith, Ravi Nair,
2. Virtualization: From the Desktop to the Enterprise by Chris Wolf, Erick M. Halter
3. Network virtualization by Kumar Reddy, Victor Moreno,
4. Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center by David Marshall, Wade A. Reynolds,

Web Mining
Syllabus

Prerequisites: Data mining, Data Base.

Objectives: Web usage mining is the process of extracting useful information from server. Web Usage Mining is the application of data mining techniques to discover interesting usage patterns from Web data in order to understand and better serve the needs of Web-based applications.

Course Description

The course is an introduction to web mining technologies. Though the Web is rich with information, gathering and making sense of this data is difficult because the documents of the Web are largely unorganized. The course will cover machine learning techniques to mine the Web and other information networks, social networks, and social media. Applications to search, retrieval, classification, and recommendation would be studied. Various models to explain the dynamics of Web processes will also be emphasized.

Course Outline (To be covered in 30 lectures)

1. Introduction, Practical web mining applications overview (3)
2. Natural Language Processing methods used for web information retrieval (6)
3. Web Content Mining (5)
4. Web Structure Mining (5)
5. Web Usage Mining (6)
6. Specific applications and case studies (5)

CO 1: The students are able to understand the concepts of web search and methods to response the user queries.

CO 2: The students are able to compute the similarity of pairs of documents, process and analyze streaming data.

CO 3: The students are able to discovery of association rules in text, social network analysis, web structure mining and web usage mining.

CO 4: The students are able to design and implement a crawler application to collect and index documents from the web.

CO 5: The students are able to analyze text and apply the knowledge and techniques of web search to solve the many real world application problems.

Program Outcomes (POs):

Students in the B Tech CSE program will attain:

PO1	Ability to apply knowledge of computing, mathematics, science, engineering and humanities.
PO2	Ability to design and conduct experiments, as well as to analyze and interpret data.
PO3	Ability to analyze a problem, identify and define the computing requirements appropriate to its solutions
PO4	Ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional
PO5	Ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that

	demonstrates comprehension of the tradeoffs involved in design choices.
PO6	Ability to design, implement, and evaluate a software or a software/hardware system, component or process to meet desired needs within realistic constraints such as memory, run time efficiency.
PO7	Understanding of professional, ethical, legal, political security and social issues and responsibilities.
PO8	Ability to communicate effectively with knowledge of contemporary issues.
PO9	Ability to analyze the local and global impact of computing on individuals, organizations, and society.
PO10	Understanding of the need for acquiring an ability to engage in lifelong learning and continuing professional development

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	H	H	M	M	M	H	L	L	-
CO2	H	M	M	H	M	H	L	M	L	-
CO3	M	M	M	M	H	H	H	L	L	L
CO4	M	M	H	H	M	M	L	M	L	L
CO5	M	M	M	H	H	H	H	M	H	H

Text Books

1. Web data mining: exploring hyperlinks, contents, and usage data by LIU, B.
2. Mining the Web - Discovering knowledge from hypertext data, by Soumen Chakrabarti,
3. *Ontology learning and population from text : algorithms, evaluation and applications by CIMIANO, P.*

Cloud and Edge Computing
Syllabus

Prerequisite: Computer Networks and Distributed Computing.
L-T-P: 4-0-0**, Credits:** 4**Type:** Minors/Honors (MR)

Course Description

Cloud Computing has continuously evolved and shall become a de facto infrastructure in many business and research organizations to deliver various business and scientific applications to end users. The goal of this course is to introduce the underlying technologies that created the current cloud computing and infrastructure. Also, almost everything is on the verge of getting connected to a Cloud, and as this trend continues, in the near future almost all devices and appliances will include IoT modules which will use sensor data collection and control/management based on Clouds and Edge devices. This paves way to explore edge and fog computing. Hence, knowledge of the core technologies and platforms of Cloud and edge computing will enable us with the tools to become a true leader in the product and application development.

Course Objectives

Students undergoing this course are expected to:	
CO1	To provide students with the fundamentals and essentials of edge and cloud computing.
CO2	To provide students a basic understanding of Internet of Things and the related standards and protocols.
CO3	To explore challenges in the integration of IoT, edge and cloud computing
CO4	To expose students with challenges and applicability of edge computing in different domains.
CO5	Understanding of design and development of IoT based applications.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Describe** the main concepts, key technologies, strengths, and limitations of cloud and edge computing and the possible applications of state-of-the-art cloud computing paradigm.	Identify/Knowledge/Remember
CO-2	**Understand** need of Edge devices in cloud computing application.	Understanding/ Comprehension:
CO-3	**Identify the need** for the JSON and Restful services in application development.	Applying
CO-4	**Choose** the appropriate technologies, algorithms, and approaches for IoT based application domain.	Analyzing
CO-5	**Design** applications	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)

3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	3	3	-	3	2	3	3	3	-	2
CO2	3	-	3	3	3	-	2	2	3	3
CO3	1	3	3	3	2	3	3	-	3	-
CO4	2	2	2	-	3	3	-	3	3	3
CO5	3	3	3	3	3	3	2	3	3	3
CO (Average) Course Average	**2.4**	**2.2**	**2.2**	**2.4**	**2.6**	**2.4**	**2.0**	**2.2**	**2.4**	**2.2**

Syllabus

UNIT I: Overview of computing paradigm: Recent trends in Computing - Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing. Evolution of cloud computing - Business driver for adopting cloud computing, Pros and Cons of Cloud Computing, Need of Edge Computing, Introduction to Transducers.

UNIT II: Cloud Computing Architecture: Cloud computing stack - Comparison with traditional computing architecture (client/server), Services provided at various levels, How Cloud Computing Works, Role of Networks in Cloud computing, protocols used, Service oriented Architecture, Role of Web services, WSDL, Software as a Service (SaaS), Web 2.0, Case Study on SaaS, Restful Services.

UNIT III: Overview and Application of IoT, IoT and WSN, IoT Ecosystem, IoT Data Link Protocol - IEEE 02.15.4e, IEEE 802.11 ah, Wireless HART, Z-Wave, Bluetooth Low Energy, Lora WAN, Network Layer Routing Protocols, Network Layer Encapsulation Protocols - 6LoWPAN, Session Layer Protocols – MQTT, SMQTT, AMQP, CoAP

UNIT IV: Why Integrate IoT, Cloud and Edge computing, Need of Edge computing, Introduction to Fog Computing, Edge devices, Edge computing implementations, Opportunities and challenges in Edge computing, Challenges and issues

UNIT V: Current Trends in Cloud, IoT and Edge Computing and their Application

Text Books

1. Dan C. Marinescu, “Cloud Computing Theory and Practice,” Morgan Kaufmann, 2014
2. Michael Miller, “Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online,” Pearson education, 2012
3. Pethuru Raj and Anupama C. Raman, “The Internet of Things: Enabling Technologies, Platforms, and Use Cases,” CRC Press, 2017
4. RajkumarBuyya, and Satish Narayana Srirama, “Fog and edge computing: principles and paradigms,” John Wiley & Sons, 2019
5. Perry Lea, “IoT and Edge Computing for Architects: Implementing Edge and IoT Systems from Sensors to Clouds with Communication Systems, Analytics, and Security,” 2nd Edition, Packet publishing, 2020.

References

1. Serpanos, Dimitrios, and Marilyn Wolf (2017). Internet of things (IoT) Systems: Architectures, Algorithms, Methodologies. Springer. DOI: https://doi.org/10.1007/978-3-319-69715-4
2. Internet Society (2015), The Internet of Things (IoT): An Overview, Understanding the Issues and

Challenges of a More Connected World, https://www.internetsociety.org/resources/doc/2015/iot-overview

3. Al Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., &Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 17(4), 23472376.
4. Brandon Butler, What is edge computing and how it's changing the network, reprint from https://www.networkworld.com/article/3224893/internet-of-things/what-is-edge-computing-and-how-it-s-changing-the-network.html
5. Skala, K., Davidovic, D., Afgan, E., Sovic, I., &Sojat, Z. (2015). Scalable distributed computing hierarchy: Cloud, fog and dew computing. Open Journal of Cloud Computing (OJCC), 2(1), 1624.
6. Yu, W., Liang, F., He, X., Hatcher, W. G., Lu, C., Lin, J., & Yang, X. (2018). A survey on the edge computing for the Internet of Things. IEEE Access, 6, 69006919.
7. Mahmud, R., Kotagiri, R., &Buyya, R. (2018). Fog computing: A taxonomy, survey and future directions. In Internet of Everything (pp. 103-130). Springer, Singapore.

Course Works in Minor Basket

Database Management System
Syllabus

Prerequisite: Basic of Data Structures.
L-T-P: 3-0-2, **Credits:** 4 Type**:** Core Essential Subjects (CES)

Course Description

In this course students will study the basic functions and capabilities of database management systems (DBMS). Emphasis is placed on the use of DBMS in solving information processing problems which will include database design case studies as well as SQL programming assignments along with transactions. A lab course is associated with it to strengthen the concepts.

Course Objective

Students undergoing this course are expected to:	
1	Understand the basic concepts and the applications of database systems.
2	Understand the relational database design principles.
3	Master the basics of SQL and construct queries using SQL.
4	Familiar with the basic issues of transaction processing and concurrency control.
5	Familiar with database storage structures and access techniques.

Course Outcomes

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO-1	**Demonstrate** the basic elements of a relational database management system.	Identify/Knowledge
CO-2	**Design** entity relationship and convert entity relationship diagrams into RDBMS.	Creating
CO-3	**Create** SQL queries on the respect data into RDBMS and formulate SQL queries on the data by understanding the concepts of normalization and query optimization.	Analyzing/ Creating
CO-4	**Apply** and **Relate** the concept of transaction, concurrency control and recovery in database.	Applying
CO-5	**Recognize** and **Identify** the use of indexing and hashing technique used in database design.	Understanding/Comprehension

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

4. Slightly (Low)
5. Moderately (Medium)
6. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	2	-	-	2	1	-	-	-	2	2
CO2	2	-	2	2	2	-	-	-	-	2
CO3	1	2	2	2	2	-	1	2	-	1

CO4	-	-	-	3	3	-	2	-	1	-
CO5	1	3	2	-	-	3	-	-	-	1
CO (Average) Course Average	**1.2**	**1.0**	**1.2**	**1.8**	**1.6**	**0.6**	**0.3**	**0.4**	**0.6**	**1.2**

Syllabus

UNIT-I: Database system concept and architecture, Entity Relationship and Enhanced E-R.
UNIT-II: Relational Data Model and Relational Algebra, SQL, Indexing, Query Optimization.
UNIT III: Relational Database Design, Normalization principles and normal forms.
UNIT IV: Transaction concept and concurrency control.
UNIT V: Data storage, indexing, query processing and physical design.
UNIT VI: DBMS: Case Study

Text Books

1. Silberschatz, Abraham, Henry F. Korth, and Shashank Sudarshan. Database system concepts. Vol. 4. New York: McGraw-Hill, 1997.
2. Elmasri, R., et al. Fundamentals of Database Systems. Addison-Wesleypublisher, 2000.
3. Garcia-Molina, Hector, Jeffrey D. Ullman, and Jennifer Widom. Database system implementation. Vol. 672. Upper Saddle River, NJ:: Prentice Hall, 2000.
4. Ramakrishnan, Raghu, Johannes Gehrke, and Johannes Gehrke. Database management systems. Vol. 3. New York: McGraw-Hill, 2003.

Reference Books

1. Garcia-Molina, Hector. Database systems: the complete book. Pearson Education India, 2008.

2. Garcia-Molina, Hector, Jeffrey D. Ullman, and Jennifer Widom. Database system implementation. Vol. 672. Upper Saddle River, NJ:: Prentice Hall, 2000.

Operating System
Syllabus

Prerequisite: Computer Programming, Data Structures and Computer Organization
L-T-P: 3-0-2, **Credits:** 4 **Type:** Core Essential Subjects (CES)

Course Description

In this course students will study the basic facilities provided in modern operating systems. The emphasis will be on understanding general concepts that are applicable to a wide range of operating systems, rather than a discussion of the features of any one specific system. However, for gaining in-depth knowledge the pedagogical operating systems such as XV6 can be used to demonstrate the working at code level. Topics that will be covered in the course include protected kernels, processes and threads, concurrency and synchronization, memory management, virtual memory, file systems, secondary storage, protection, and security.

Course Objectives

Students undergoing this course are expected to:	
1	gain extensive knowledge on principles and modules of operating systems
2	understand key mechanisms in design of operating systems modules
3	understand process management, concurrent processes and threads, memory management, virtual memory concepts, deadlocks
4	compare performance of processor scheduling algorithms
5	produce algorithmic solutions to process synchronization problems
6	use modern operating system calls such as Linux process and synchronization libraries

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics**	**Bloom's taxonomy**
CO-1	Demonstrate the knowledge of Operating Systems both at system call level and code level.	Identify/ Knowledge
CO-2	Compare and analyze the different implementation approach of system programming and operating system abstractions.	Creating/ Analyzing
CO-3	Formulate synchronization problem and develop the solution for same	Analyzing, Creating
CO-4	Interpret various OS functions used in different operating systems.	Applying
CO-5	Learning about real life applications of Operating System in every field.	Understanding/Comprehension

Mapping Course Outcomes with Program Outcomes

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	-	-	-	-	2	-	-	-	2
CO2	-	-	-	-	-	-	2	-	-	-
CO3	-	-	-	-	-	-	-	-	-	-
CO4	-	3	-	-	-	-	-	-	-	-
CO5	2	-	-	-	-	-	-	-	-	2
CO (Average) Course Average	**0.8**	**0.6**	**-**	**-**	**-**	**0.4**	**0.4**	**-**	**-**	**0.8**

3-High, 2-Medium, 1-Low

Syllabus

UNIT-1: Introduction and Overview

UNIT-II: Process fundamentals, scheduling, synchronization.

UNIT III: Inter-process communication, Deadlock.

UNIT IV: Memory management and virtual memory.

UNIT V: File system and secondary storage.

UNIT VI: Protection and security issues, Case studies e.g., Linux, Solaris and Android.

Text Books:

1. Operating Systems, by William Stallings.
2. Operating Systems Concepts by Silberschatz, Galvin, and Gagne.
3. The Design of the UNIX Operating System, by Maurice J. Bach.
4. Advanced Programming in the UNIX Environment, by W. R. Stevens & S. A.Rago.

Reference Books:

1. The Design and implementation of the 4.4 BSD UNIX operating system by Marshall Kirk McKusick, Keith Bostic, Michael J. Karels, John S. Quarterman.
2. xv6: A simple, Unix-like teaching operating system by Russ Cox, FransKaashoek and Robert Morris.

Object Oriented Programming

Prerequisite: Basics of Programming
L-T-P: 2-0-2, Credits: 3 **Type:** Core Elective Subjects (CEL)

Course Description

This is an introductory course, where students learn and practice essential programming skills using the Java programming language. This course provides an overview of Object Oriented
Programming (OOP) concepts using Java/C++. It helps to understand basic OOP concepts and assist in applying these concepts. The principles behind OOP discussed. It covers object-oriented principles such as classes, objects, abstraction, composition, Inheritance, polymorphism, and interfaces. These concepts can be implemented in the Java language. Along the way, many of the Java library classes are seen that can be organized to solve a variety of problems. The Java collection classes are studied. Additional topics include exception handling, database connectivity with JDBC, and multi-threading. The course is programming intensive. By the end of this course student will able to understand the basics of OOP and be prepared to take on more complex challenges. A lab course is associated with it to strengthen the concepts.

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO-1	Understand the Basic concept of Object Orientation, object identity and Encapsulation.	Understanding/ Comprehension:
CO-2	Understand the Basic concept of Basic Structural Modeling.	Understanding/ Comprehension
CO-3	Know the knowledge of Object-oriented design, Object design.	Identify/Knowledge/Remember
CO-4	Know the knowledge of Basics of Java or C++.	Identify/Knowledge/Remember
CO-5	Understand the Basics of object and class in Java or C++.	Understanding Analyzing

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	2	-	-	1	-	-	-	2	-	-
CO2	2	2	-	1	-	-	-	-	-	-
CO3	2		-	1	-	-	-	-	-	-
CO4	2		-	1	-	-	-	2	-	-
CO5	1	1	2	1	-	-	-	1	-	-
CO (Average) Course Average	**1.8**	**0.6**	**0.4**	**1.0**	**0.0**	**0.0**	**0.0**	**1.0**	**0.0**	**0.0**

UNIT I: INTRODUCTION

The meaning of Object Orientation, object identity, Encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object-oriented modeling, Introduction to UML, conceptual model of the UML, Architecture.

UNIT II: BASICS OF JAVA OR C++

Core Java: Introduction to Object Oriented Software development through Java. Classes and Objects.

UNIT-III Abstraction, Inheritance, Polymorphism, Nested classes and interfaces, Exceptions, Strings, Packages, The I/O Package.

UNIT IV: Advanced Java: Event Handling, AWT, Swing, Applets, Multi-Threading, Generic, The collection frameworks, Introduction to socket programming, Java Database connectivity (JDBC).

TextBooks:

1. Kathy Sierra and Bert Bates, “Head First Java”, 2nd edition, O’Reilly
2. Herbert Schildt , “Java : The Complete Reference”, 9th edition, Oracle Press
3. Cay S. Horstmann and Gary Cornell , “Core Java Volume I & II”, 10th edition, Prentice-Hall
4. Tony Gaddis, “Starting Out with Java: From Control Structures through Objects”, 6th edition, Pearson
5. David Flanagan, “Java in a Nutshell”, 5th edition, O’Reilly

Analysis of Algorithms
Syllabus

Prerequisite: Engineering Mathematics, Data Structures and Discrete Mathematics
L-T-P: 3-0-0**, Credits:** 3 **Type:** Core Essential Subjects(CES)

Course Description

Algorithms are recipes for solving computational problems. This course teaches techniques for the design and analysis of efficient algorithms for various applications, emphasizing methods useful in practice.

Course Objective

Students undergoing this course are expected to:	
1	To understand the problems in computing
2	To use/design the most efficient of the available/suitable techniques to solve these problems
3	To obtain efficiency of the algorithm is to be analyzed in the domains of time, space and energy
4	A lab course is also associated with it to strengthen the concepts.

Course Outcomes

CO Numbers	Course Outcomes (Action verb should be in italics)	Bloom's taxonomy
CO-1	**Apply** knowledge of computing and mathematics for designing and implementing problem solving algorithms.	Applying
CO-2	**Analyze** the best, average and worst-case performance of the algorithms using asymptotic bounds.	Analyzing
CO-3	**Choose** and **apply** best suitable algorithm to find optimal solution of engineering problems	Creating/Applying
CO-4	**Demonstrate** a familiarity with various classes of algorithms and data structures.	Identify/Knowledge
CO-5	**Realize** the limits of computer algorithms via P, NP and NP-complete problems.	Identify/Knowledge

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

4. Slightly (Low)
5. Moderately (Medium)
6. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	3	2	2	2	3	1	-	-	1	-
CO2	3	2	3	2	3	2	1	2	-	-
CO3	3	2	3	2	3	2	1		-	-
CO4	3	3	2	2	3	1	-	-	1	-
CO5	3	2	3	2	3	3	-	-	2	3

CO (Course Average)	**3.0**	**2.2**	**2.6**	**2.0**	**3.0**	**1.8**	**0.4**	**0.4**	**0.8**	**0.6**

Syllabus

UNIT 1: Sorting and complexity analysis: Time and Space Complexity, Different Asymptotic notations and their mathematical significance. Analysis of sorting algorithms of different complexity classes. Heaps, Heat Sort and Priority Queues.

Divide and Conquer Algorithms: Binary Search, Merge sort, Multiplication of Large Integers, Closest Pair, Strassen's Algorithm, Recurrences and Masters' Method, Quick Sort and Order Statistics.

UNIT II: Greedy Algorithms with Graph Recapitulation: Graph representation and Traversals, Minimum Spanning Trees and Shortest Paths in a Graph, Dial's Algorithm, Travelling Salesman Problem, Knapsack Problem, Job Sequencing Problem, Vertex Cover Problem, Maximum Network Flow, Huffman Coding and Encoding, Coin Change Problem, Discussion on Load Balancing Problem.

UNIT III: Dynamic Programming: Principle of Optimality, Cases where Greedy Algorithms Fails, 0/1 Knapsack, Coin Change Problem, Matrix Chain Multiplication, Optimal Binary Search Tree, Longest Common Subsequence, All-pairs Shortest Path Problem, Rod Cutting Problem, Memorization.

Backtracking: Hamiltonian Cycle, N-Queens Problem, Maze Problem, Sum of Subset Problem, Designing Generic Backtracking Algorithm.

UNIT IV: Number Theoretic Algorithms: The GCD, Modular Arithmetic, Chinese Remainder Theorem, Cyclic Groups, Linear Programming and Duality, Primality Testing.

Text Processing: Naïve String Matching, Rabin Karp Algorithm, String Matching using Finite Automata, KMP Algorithm.

UNIT V: NP-Completeness and Approximation algorithm:

P class, NP class, NP hard class, NP complete class – their interrelationship, Satisfiability Problem and Reducibility Examples. Necessity of Approximation Scheme, Polynomial Time Approximation with Examples.

Text Books

1. Cormen, Thomas H., et al. Introduction to algorithms. MIT press, 2009.
2. Horowitz, Ellis, SartajSahni, and Susan Anderson-Freed. Fundamentals of data structures. Vol. 20. Potomac, MD: Computer science press, 1976.
3. Sedgewick, Robert, and Philippe Flajolet. An introduction to the analysis of algorithms. Pearson Education India, 2013.

Reference Books

1. Levitin, Anany. Introduction to Design and Analysis of Algorithms, 2/E. Pearson Education India, 2008.
2. Kozen, Dexter C. The design and analysis of algorithms. Springer Science & Business Media, 1992.

Machine Learning with Python
Syllabus

Prerequisite: Computer Programming, Basics of Mathematics and Statistics
L-T-P: 3-0-0,**Credits:** 3 **Type:** Core Elective (CEL)

Course Description

This course contents various machine learning algorithms and its application in real life. Its contents supervised, unsupervised, reinforcement learning, clustering, bayes theorem concept and application of machine learning.

Course Objectives

Students undergoing this course are expected to:	
1	This course is designed for students: To introduce the fundamental concepts of machine learning and its applications.
2	Students may explore the understanding of machine learning with their underlying mathematical concepts, issues and challenges, including data, model selection, model complexity, strengths and weaknesses and their relevant applications.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	***Understanding*** the nature of data, and model selection.	Identify/Knowledge
CO-2	Describe the various classification algorithms.	Comprehension
CO-3	Develop and ***apply*** clustering base algorithm for finding relationships between data variables.	Applying
CO-4	***Understanding*** of back propagation algorithm for the training of neural networks	Understanding
CO-5	***Design*** and implement various machine learning algorithms in a range of real-world applications.	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	3	3	2	3	-	-	-	-	-
CO2	2	3	3	2	3	-	-	-	-	-
CO3	2	3	3	3	3	1	-	-	-	-
CO4	3	3	3	3	3	-	-	-	-	-
CO5	2	3	3	3	3	2	-	-	2	-
CO (Course Average)	**2.2**	**3**	**3**	**2.6**	**3**	**0.6**	**-**	**-**	**0.4**	**-**

Syllabus

UNIT I: Introduction to Machine Learning; Supervised Learning, Unsupervised Learning, Reinforcement Learning, Machine learning basics: capacity, over fitting and under fitting, hyper parameters and validation sets, bias & variance.

UNIT II: Bayes Theorem Concept Learning Maximum Likelihood classifier, Bayesian Belief Networks, Decision-trees, *univariate and multivariate decision tree*, network equivalent of decision trees, over fitting and evaluation.

UNIT III: Logistic Regression, Support Vector Machine, Kernel function and Kernel SVM, Instant based Learning: k- Nearest Neighbor, locally weighted Regression, Radial Bases Functions; Case Based Learning.

UNIT IV: Clustering: k-means, adaptive hierarchical clustering, and Gaussian mixture model, Neural network: Perception, multilayer network, back propagation.

UNIT V: Applications of machine learning (Case studies): Business Analytics, Natural language processing, Computer vision, Internet of Things, bio-informatics etc.

Text Books

1. Tom M. Mitchell, ―Machine Learning, McGraw-Hill Education (India) Private Limited, 2013.
2. Mohri, M., Rostamizadedh, A., and Talwalkar, A., Foundations of Machine Learning, The MIT Press (2012)

Reference Books

1. EthemAlpaydin, (2004) "Introduction to Machine Learning (Adaptive Computation and Machine Learning)", The MIT Press
2. T. astie, R. Tibshirani, J. H. Friedman, "The Elements of Statistical Learning", Springer (2nd ed.), 2009

Computer Organization
Syllabus

Prerequisite: Basic knowledge prior to digital electronics, Basic functional units of a computer system
L-T-P: 3-0-0, Credits**:** 3**Type:** Core Essential Subjects (CES)

Course Objectives

Students undergoing this course are expected to:	
1	This course will discuss the basic concepts of computer architecture and organization that can help the participants to have a clear view as to how a computer system works. Examples and illustrations will be mostly based on a popular Reduced Instruction Set Computer (RISC) platform.
2	This course qualitatively and quantitatively examines computer design trade-offs and teaches the fundamentals of computer architecture and organization, including CPU, memory, registers, arithmetic unit, control unit, and input/output components
3	Understand the fundamentals of computer architecture.
4	Design and implement single-cycle and pipelined data paths for a given instruction set architecture.
5	Understand the performance trade-offs involved in designing the memory subsystem, including cache, main memory and virtual memory.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Study of the basic structure and operation of a digital Computer system.	Identify/Knowledge
CO-2	Analysis of the design of arithmetic & logic unit and understanding of the fixed point and floating-point arithmetic operations.	Analyzing
CO-3	Implementation of control unit techniques and the concept of Pipelining	Applying
CO-4	Understanding the hierarchical memory system, cache memories and virtual memory	Analyzing/understanding
CO-5	Understanding the different ways of communicating with I/O devices and standard I/O interfaces	Understanding

Mapping Course Outcomes with Program Outcomes

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	2	1	1	2					2
CO2	3	3	2	2	1	2				
CO3	2	1	1	3	2	2			2	1
CO4	3	1	3	3	1	2				2
CO5	1	2	1		2					
CO (Average) Course Average	**2.2**	**1.8**	**1.6**	**1.8**	**1.6**	**1.2**	**-**	**-**	**0.4**	**1**

3-High, 2-Medium, 1-Low

Syllabus

UNIT 1: Introduction: Functional units of digital system and their interconnections, buses, bus architecture, types of buses and bus arbitration. Register, bus and memory transfer. Processor organization, general registers organization, stack organization and addressing modes.

UNIT II: Arithmetic and logic unit: Look ahead carries adders. Multiplication: Signed operand multiplication, Booths algorithm and array multiplier. Division and logic operations. Floating point arithmetic operation, Arithmetic & logic unit design. IEEE Standard for Floating Point Numbers.

Unit III: Control Unit: Instruction types, formats, instruction cycles and sub cycles (fetch and execute etc), micro operations, execution of a complete instruction. Program Control, Reduced Instruction Set Computer, Pipelining. Hardwire and micro programmed control: micro programme sequencing, concept of horizontal and vertical microprogramming.

Unit IV: Memory: Basic concept and hierarchy, semiconductor RAM memories, 2D & 2 ½ D memory organizations. ROM memories. Cache memories: concept and design issues & performance, address mapping and replacement Auxiliary memories: magnetic disk, magnetic tape and optical disks Virtual memory: concept implementation.

UNIT V: Input / Output: Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. Modes of Data Transfer: Programmed 1/0, interrupt initiated 1O and Direct Memory Access., I/O channels and processors. Serial Communication: Synchronous & asynchronous communication, standard communication interfaces.

Text Books

1. Computer System Architecture - M. Mano
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky Computer Organization, McGraw-Hill Fifth Edition, Reprint 2012
3. John P. Hayes, Computer Architecture and Organization, Tata McGraw Hill, Third Edition, 1998, Reference books
4. William Stallings, Computer Organization and Architecture-Designing for Performance, Pearson Education, Seventhedition, 2006.

Reference Books

1. Behrooz Parahami, “Computer Architecture”, Oxford University Press, Eighth Impression, 2011.
2. Fundamentals of Microprocessors & Microcontrollers by B.RAMDHANPAT RAI PUBLICATIONS, NEW DELHI
3. David A. Patterson and John L. Hennessy, “Computer Architecture-A Quantitative Approach”, Elsevier, a division ofteed India Private Limited, Fifth edition, 20121
4. Swati Saxena Computer Organization, Dhanpat Rai Publications, NEW DELHI

Course Works in Honors Basket

Data Center Networking
Syllabus

Prerequisite: Computer Networks
L-T-P: 4-0-0**, Credits:** 4**Type:** Minors/Honors(MR)

Course Description

Multipath Topologies and Routing, Virtual Networking, Data Centre Load Balancing, Fault-Tolerance, Performance Isolation, Network acceleration and In-Network Computing, Atomic Multicast and Network Accelerated Consensus, Explicit Congestion Control and Packet Scheduling, Protocol Independent Programmable Networking Hardware, Network Function Virtualization.

Course Objectives

Students undergoing this course are expected to:	
1	To introduce students a systematic understanding of modern data center networks.
2	To cover most of the relevant technologies to implement the current data centers, including switch fabrics, virtualization, network infrastructure, network protocols, and etc.
3	To give knowledge about interconnection networks, including network topology, clos networks, and routing protocols.
4	After having undergone this course, the students shall be able to understand many different aspects of data center networking in a way that is both comprehensive and current.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	To **understand** the concepts behind the data center networking over the traditional networking.	Identify/Knowledge
CO-2	To **analyze** data center topologies and virtualized environment.	Analyzing
CO-3	To **explain** software-defined networking principles and OpenFlow.	Applying
CO-4	To **recognize** the routing protocols for data center networking and their key design principles during deployment.	Analyzing

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Syllabus

UNIT I: Data center evolution and switch fabrics: Networking basics, types of data center, cloud data centers and cloud networking characteristics, enterprise cloud and virtualized data centers, types of data center network topology, clos topology, switch fabric architecture, switch fabric congestion management, flow control and traffic management.

UNIT II: Cloud data center networking: Traditional multi-tiered enterprise networks, data center network switch types, flat data center networks, rack scale architectures, network function virtualization, data center bridging, remote direct memory access.

UNIT III: Virtualization and networking: Virtual machines, virtual switching, PCI express and edge virtual bridging, VM migration, multi-tenant environments, traditional network tunneling protocols, VXLAN and NVGRE protocols.

UNIT IV: Software-defined storage and networking: Advanced storage technologies, storage communication protocols, software-defined storage, storage in cloud data centers, data center software, OpenStack and OpenFlow, controllers Routing protocols: BGP, OSPF, network virtualization, container networking, multicast routing.

Text Books

1. Gary Lee, "Cloud Networking - Understanding Cloud-based Data Center Networks", Elsevier, 2014.
2. Dinesh G. Dutt, "Cloud Native Data Center Networking," O'Reilly Media, 2019.

Reference Books

1. Kevin Corbin, Ron Fuller, and David Jansen, "NX-OS and Cisco Nexus Switching: Next-Generation Data Center Architectures" Cisco Press, first edition, 2010.
2. SilvanoGai, and Claudio DeSanti, "I/O Consolidation in the Data Center" Cisco Press; first edition, 2009.

Distributed Ledger Technology
Syllabus

Prerequisite: Distributed Computing
L-T-P: 4-0-0**, Credits:** 4**Type:** Core ElectiveSubjects (CEL)

Course Description

This course covers the basic idea: Block-chain and distributed ledger technology. It also discusses the origins, development, and popularization. This course includes the cryptocurrencies and cryptoassets, other application such as Smart contracts, D Apps, DAOs, Stable coins. This course also includes the future disruption: the potential of block-chain. Areas for potential abuse and misuse: block-chains really unhakable there other reasons to be cautious of the evolution of block-chain.

Course Objectives

Students undergoing this course are expected to:	
CO1	The course aims to develop an understanding of the need of distributed ledger technology and its wide application domain.
CO2	The technical ideas to be understood include the key cryptographic constructs, economic incentive mechanisms and algorithms, Distributed algorithms underpinning cryptocurrencies such as Bitcoin and Ethereum.
CO3	The concept of smart contract shall be introduced along with fault tolerant consensus, mining incentives and storage.
CO4	The course also aims to introduce students to the broader context of the range of alternate designs for distributed ledger technologies, the range of potential applications for these technologies, and the socio-economic, ethical and legal dimensions of their use.
CO5	Shall provide the students the ability to design and develop e-governance applications for benefit of society.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Use basic cryptographic concepts including private/public keys, signatures, hash functions, Merkle trees and distinguish pros and cons of centralized versus decentralized databases.	Identify/Knowledge/Remember
CO-2	Understand alternative approaches to FLP Problem, Byzantine fault-tolerant consensus including proof of work, proof of stake, etc.;	Understanding/ Comprehension:
CO-3	Characterize the basic setup of Bitcoin, including storage, mining, and payments;	Applying
CO-4	Understand the limits of privacy with distributed ledgers and confidential transactions.	Analyzing
CO-5	Elaborate inherent scalability limits of distributed ledgers and potential solutions with channeling, horizontal scaling and second-layer, off-chain transactions;	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	3	3	2	3	2	3	3	3	2	2
CO2	3	-	3	3	3	-	2	2	3	3
CO3	1	3	3	3	2	3	3	-	3	2
CO4	2	2	2	-	3	3	1	3	3	3
CO5	3	3	3	3	3	3	2	3	2	3
CO (Average) Course Average	**2.4**	**1.6**	**2.6**	**2.4**	**2.6**	**2.4**	**2.2**	**1.6**	**2.6**	**2.6**

Syllabus

UNIT I: Overview of blockchain technology, Introduction to DLT - Functionality, Applications and Use Cases, Challenges and Constraints, Philosophy and Implications, Introduction to Hashes and its Significance, Introduction to Transactions, Blocks and blockchain, Hash function, Digital Signature

UNIT II: Technological and Cryptographic Elements in Blockchain, Need for a Decentralized Ledger System, Centralized Trusted Party - Advantages and Disadvantages, Security, Integrity and Privacy Issues of a Decentralized System, Trust and Coordination, Barriers to Blockchain Adoption

UNIT III: Introduction to Bitcoin, Bitcoin protocol, Classical FLP problem and its application in consensus, Byzantine General problem and Fault Tolerance, Mining Mechanism, Energy usage, Distributed Consensus, Merkle Tree, Transactions and Fee, Anonymity, Reward, Soft & Hard Fork, Private and Public blockchain, Bitcoin Transaction structure, Double Spending Problem, Privacy in blockchains, Legal considerations

UNIT IV: Introduction to Consensus Problem, Distributed Consensus, Fischer, Lynch and Patterson (FLP) Problem, Nakamoto consensus, Proof of Work, Proof of Stake, Delegated Proof of Stake, Leased Proof of Stake, Proof of Elapsed Time, Tangle, Proof of Burn, Difficulty Level, Energy utilization and alternate, Consensus in Ethereum

UNIT V: Applications of DLT in e-Governance, Application of DLT in Banking and Finance, Design and Architecture of these applications, Use of IoT& Cloud in DLT, Application of Restful services in DLT Applications, Recent Developments and Trends in Distributed Ledger Technology, Semester Project

Text Books

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press, July 2016.

References

1. Casey, Michael, et al. "The impact of blockchain technology on finance: A catalyst for change." (2018). Chapter 1 (pages 1 – 7);

2. Yaga, D., et al. Blockchain technology overview. National Institute of Standards and Technology, Gaithersburg. MD, Technical report NIST IR 8202, October 2018, NIST IR 8202. https://doi. org/10.6028/NIST. IR. 8202. https://nvlpubs. nist. gov/nistpubs/ir/2018/NIST. IR. 8202. pdf. Accessed 02 Jan, 2020. (pages 9 – 23, sections 1 & 2)

3. Lamport, Leslie, Robert Shostak, and Marshall Pease. "The Byzantine Generals Problem', ACM Transactions on Programming Languages and Systems (TOPLAS), 4(3), (July 1982) (required 382-387).

4. A (Short) Guide to Blockchain Consensus Protocols' CoinDesk (March 4, 2017)

5. Wattenhofer, The Science of the Blockchain

6. Antonopoulos, Mastering Bitcoin: Unlocking Digital Cryptocurrencies

7. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System

8. DR. Gavin Wood, "ETHEREUM: A Secure Decentralized Transaction Ledger, "Yellow paper.2014.

9. Nicola Atzei, Massimo Bartoletti, and TizianaCimoli, A survey of attacks on Ethereum smart contracts

Network Security
Syllabus

Prerequisite: Basics of ComputerNetworks , Cryptography
L-T-P: 3-0-2, **Credits:** 4**Type:** Minors/Honors (MR)

Course Description

A network security course discusses different types of malicious attacks and various methods of responding to them. Students learn how to protect computer networks by using security codes. The course also examines malware, social engineering attacks, operating system holes, web security, privacy and e-mail hacking.

Course Objectives

Students undergoing this course are expected to:	
CO-1	To understand basics of Cryptography and Network Security.
CO-2	To be able to secure a message over insecure channel by various means.
CO-3	To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
CO-4	To understand various protocols for network security to protect against the threats in the networks.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Provide security of the data over the network.	Understanding/ Comprehension:
CO-2	Do research in the emerging areas of cryptography and network security.	Analyzing and creating
CO-3	Implement various networking protocols.	Understanding/ Comprehension
CO-4	Protect any network from the threats in the world	creating
CO-5	Various network security applications, IPSec, Firewall, IDS, Web security, Email security, and Malicious software etc.	Identify/Knowledge/Remember

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

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

CO (Average) Course Average	**2.0**	**1.2**	**0.6**	**1.6**	**2.0**	**0.0**	**0.0**	**1.0**	**0.0**	**0.0**

Syllabus

UNIT I: Introduction to Cryptography and Block Ciphers

Introduction to security attacks - services and mechanism - introduction to cryptography -Conventional Encryption: Conventional encryption model - classical encryption techniques -substitution ciphers and transposition ciphers – crypt analysis – steganography - stream and block ciphers - Modern Block Ciphers: Block ciphers principals - Shannon's theory of confusion and diffusion - fiestal structure - data encryption standard(DES) - strength of DES - differential and linear crypt analysis of DES - block cipher modes of operations - triple DES – AES

UNIT II: Confidentiality and Modular Arithmetic

Confidentiality using conventional encryption - traffic confidentiality - key distribution - random number generation - Introduction to graph - ring and field - prime and relative prime numbers - modular arithmetic - Fermat's and Euler's theorem - primality testing - Euclid's Algorithm - Chinese Remainder theorem - discrete algorithms.

UNIT III: Public key cryptography and Authentication requirements

Principles of public key cryptographic systems - RSA algorithm - security of RSA - key management – Diffle-Hellman key exchange algorithm - introductory idea of Elliptic curve cryptography – Elgamel encryption - Message Authentication and Hash Function: Authentication requirements - authentication functions - message authentication code - hash functions - birthday attacks – security of hash functions and MACS.

UNIT IV: Integrity checks and Authentication algorithms

MD5 message digest algorithm - Secure hash algorithm (SHA) Digital Signatures: Digital Signatures - authentication protocols - digital signature standards (DSS) - proof of digital signature algorithm - Authentication Applications: Kerberos and X.509 - directory authentication service - electronic mail security-pretty good privacy (PGP) - S/MIME.

UNIT V:IP Security and Key Management

IP Security: Architecture - Authentication header - Encapsulating security payloads - combining security associations - key management.

UNIT VI: Web and System Security

Web Security: Secure socket layer and transport layer security - secure electronic transaction (SET) - System Security: Intruders - Viruses and related threads - firewall design principals – trusted systems.

TextBooks:

1. William Stallings, "Crpyptography and Network security Principles and Practices", Pearson/PHI.
2. Wade Trappe, Lawrence C Washington, "Introduction to Cryptography with coding theory", Pearson.

Reference Books:

1. W. Mao, "Modern Cryptography – Theory and Practice", Pearson Education.
2. Charles P. Pfleeger, Shari Lawrence Pfleeger – Security in computing – Prentice Hall of India.

Advanced Computer Network
Syllabus

Prerequisite: Computer networks, packet switching, routing and flow control, congestion control and quality-of-service, Internet protocols (IP, TCP, BGP), network security, network management, software defined networking, and the design of network services (multimedia, file, and web servers).

L-T-P: 4-0-0**, Credits:** 4**Type:** Minors/Honors(MR)

Course Description

This course is design to introduce advance topics in Computer networks, algorithms, protocols at the application transport, network, and medium access layers as well as experimentation simulation and modeling technique will be covered. The course explores emerging research challenging in the field of information and content centric networks. Course project are required and involve programming and working with the Linux operating system**.**

Course Objectives

Students undergoing this course are expected to:	
1	Students will be able to design a network with appropriate protocols selected according requirement.
2	Students will be able to analyze different routing protocols and traffic engineering methods deployed in networking.
3	Student will be able to know working of modern networks.
4	Student will be able to differentiate between the development of fixed and wireless networks.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	To **understand** the basics of Network components and environment and OSI reference model	Identify/Knowledge
CO-2	To obtain the basic knowledge about and the TCP-IP reference model.	Understanding/ Comprehension
CO-3	To have an understanding of concepts of protocols, network interfaces, and design/performance issues in local area networks and wide area networks.	Applying
CO-4	To understand and study the Routing Concept at various layers.	Understanding/ Comprehension:
CO-5	To comprehend about Network Virtualization, SDN and Wireless Sensor Network	Understanding/ Comprehension:

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)

2. Moderately (Medium)

3. Substantially (High)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	2	2	2	2	2	-	-	-	-
CO2	2	2	2	2	2	3	-	-	-	-
CO3	2	2	3	3	3	3	-	-	-	-
CO4	2	1	1	1	2	2	-	2	2	-

CO5	1	2	2	2	2	2	2	2	2	-
CO (Average) Course Average	**1.8**	**1.8**	**2**	**2**	**2.2**	**2.4**	0.4	0.8	**0.8**	**-**

Syllabus

UNIT 1: Review of Basic Network Architectures: OSI reference model, Review of Element of Network, Networking Device and media connection, Optical Networks., MAC layer issues, Ethernet 802.3, ARP, IP addressing and Subnetting, NAT and PAT, Variable Length Subnet Masking, CIDR, Traffic Characterization (CBR, VBR).

UNIT II: Link layer: switching, multiple access, error recovery: Data Link Layer MAC Layer: Local Area Network Technologies: Fast Ethernet, Gigabit Ethernet, IEEE 802.11 WLAN, Bluetooth. TCP Architecture, Barkley API, Application protocol and TCP/IP suits, TCP mechanics, congestion control, Switching Paradigms, Multiplexing. Error Control, Flow Control, The Internet Protocols-IPV4 & IPV6, UDP & TCP.

UNIT III: Network Device, Routing algorithms, BGP, Advanced routing concepts, Router architectures, internetworking: Interdomain Routing, Multicast Routing Protocols, Multi-Protocol Label Switching, and Virtual Networks. Transport layer protocols, UDP mechanics. Socket Programming.

UNIT IV: Naming, DNS, DDNS, Paradigms for Communication in Internet, Caching, Issues of Scaling in Internet and Distributed Systems, Caching Techniques for Web, Protocols to Support Streaming Media, Multimedia Transport Protocols, Content Delivery Networks, Overlay and P2P Networks. Network virtualization, software defined networking Applications and Other Networking Technologies: RTP, RTSP, SIP, VoIP.

UNIT V: Overview of Wireless Sensor Network (WSN), Sensor Node Architecture, Sensor Network Protocols-MAC Protocols & Energy Efficiency, Routing Protocols- Data centric, Hierarchical Location based, Energy Efficient Routing etc. Vision for 4G & 5G Wireless Standards, Overview of Mobile Internet ProtocolDoS Attack, Security Systems, SSH, PGP, TLS, IPSEC Mitigation in Internet, Security in MPLS.

Text Books

1. Computer Networks: A Systems Approach, by Peterson and Davie, 5th Ed. Morgan Kauffman, 2011
2. Computer Networking: Top Down Approach, by Kurose and Ross, 6th Ed. Pearson, 2011
3. Internetworking with TCP/IP: Principles, Protocols, and Architecture, Douglas E. Commer, Prentice Hall, 6th Edition, 2013

Reference Book

1. SDN: Software Defined Networks, Thomas D. Nadeau, Ken Gray, Kindle Ed., O'Reilly, 2013
2. Data Communication and Network, Behrouz A. Forouzan, McGraw Hill Education,2007.
3. Computer Networks, Andrew S. Tanenbaum, David J. Wetherall, Prentice-Hall, 5th Edition, 2010

Software Defined Networking
Syllabus

Prerequisite: Computer Networks
L-T-P: 4-0-0, **Credits:** 4**Type:** Minors/Honors(MR)

Course Description

This course coves the data center networking technologies, more specifically software-defined networking. It also covers the history behind SDN, description of networks in data-centers, a concrete data-center network architecture (Microsoft VL2), and traffic engineering.

Course Objectives

Students undergoing this course are expected to:	
CO1	Explain the key benefits of SDN by the separation of data and control planes.
CO2	Interpret the SDN data plane devices and OpenFlow Protocols
CO3	Implement the operation of SDN control plane with different controllers
CO4	Apply techniques that enable applications to control the underlying network using SDN
CO5	Describe Network Functions Virtualization components and their roles in SDN

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	**Explain** in detail the operation of the SDN control plane, Data Plane, Network Function Virtualization	Identify/Knowledge/Remember
CO-2	**Describe** the key benefits of SDN, in particular those benefits brought about by the separation of data and control planes	Understanding/ Comprehension:
CO-3	**Demonstrates** how to enable applications to control the underlying network using SDN.	Applying
CO-4	**Examine** how SDN and NFV go together in order to improve network performance.	Analyzing
CO-5	**Design** and configure an SDN-friendly network emulator and **develop** a sample SDN for a given task	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

If there is no correlation, the cell is to be left blank or put (-).

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	1	-	-	2	1	-	-	3	-
CO2	1	-	2	2	-	-	-	3	-
CO3	1	2	-	2	-	-	-	-	-

CO4	-	-	-	-	-	2	-	-	-
CO5	1	1	-	-	-	2	-	-	-
CO (Course Average)	**0.8**	**0.6**	**0.4**	**1.2**	**0.2**	**0.4**	**0.0**	1.2	**0.0**

Syllabus

UNIT 1: SDN Background and Motivation, Evolving network requirements-The SDN Approach: Requirements, SDN Architecture, Characteristics of Software-Defined Networking, SDN and NFV-Related Standards: Standards-Developing Organizations, Industry Consortia, Open Development Initiatives.

UNIT II: SDN Data plane and OpenFlow, SDN data plane: Data plane Functions, Data plane protocols, OpenFlow logical network Device: Flow table Structure, Flow Table Pipeline, The Use of Multiple Tables, Group Table- OpenFlow Protocol.

UNIT III: SDN Control Plane, SDN Control Plane Architecture: Control Plane Functions, Southbound Interface, Northbound Interface, Routing, Open source controller: Floodlight, open Daylight, ONOS etc.

UNIT IV: SDN Application Plane SDN Application Plane Architecture: Northbound Interface, Network Applications, User Interface- Network Services Abstraction Layer: Abstractions in SDN, Frenetic- Traffic Engineering Measurement and Monitoring.

UNIT V: Network Functions Virtualization: Background and Motivation for NFV- Virtual Machines- NFV Concepts: Simple Example of the Use of NFV, NFV Principles, High-Level NFV Framework, NFV Benefits and Requirements- NFV Reference Architecture: NFV Management and Orchestration.

Text Books

1. William Stallings, “Foundations of Modern Networking”, Pearson Ltd.,2016.
2. Software Defined Networks: A Comprehensive Approach by Paul Goransson and Chuck Black, Morgan Kaufmann Publications, 2014
3. SDN - Software Defined Networks by Thomas D. Nadeau & Ken Gray, O'Reilly, 2013

Reference Books/Research Papers

1. Feamster, Nick, Jennifer Rexford, and Ellen Zegura. "The road to SDN: an intellectual history of programmable networks." ACM SIGCOMM Computer Communication Review 44.2 (2014): 87-98.
2. Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76.

Network Protocol Design & Implementation

Unit 1: Internet Architecture: Open Source Implementations. OSI and TCP/IP Standards, Introduction to switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks,Error Detection & Correction: Introduction, Block coding, Linear block codes, Cyclic codes, Checksum. Multiple access protocols, addressing, Ethernet.

Unit 2: Internet Protocols Layers: Introduction, Data-Plane Protocols: Internet Protocols IPv4,IPv6. Control-Plane Protocols: Address Management, Routing Protocols, Error Reporting.Internetworking basics, Comparison of IPv4 and IPv6 Headers.

Unit 3: Introduction and transport layer services, Multiplexing and Demultiplexing, Connectionless transport (UDP), Connection-oriented transport (TCP), Congestion control, TCP congestion control, Principles of computer applications, Web and HTTP, FTP, SNMP, P2P, E-mail, DNS, Socket programming with TCP and UDP

Unit 4: Protocol Structure and Design, Specification and Modeling: Validation Models, Network Configuration Verification Approaches, Correctness Requirements, Finite State Machines, Conformance Testing, Synthesis and Validation, Protocol Simulator and Validator

Unit 5: Sequential Programming in PROMELA, Verification of Sequential Programs, Concurrency and Synchronization, Verification with Temporal Logic, Channels in PROMELA

Kurose and Ross,Computer Networking- A Top-Down Approach, Pearson, 5th edition
Behrouz A. Forouzan, "Data Communications and Networking," Tata McGraw-Hill,
Fourth Edition
Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, "Computer Networks: An Open Source
Approach", McGraw Hill Publisher, 2011
DESIGN AND VALIDATION OF COMPUTER PROTOCOLS Gerard J. Holzmann Bell Laboratories Murray Hill
Principles of the Spin Model CheckerGerard J. Holzmann Pasadena, California May 2007

CO1 Understand and Contrast the concept of internet protocols, OSI & TCP/IP reference models and discuss the functionalities of each layer in these models.

CO2 Discuss and Analyze flow control and error detection and correction mechanisms and apply them using standard data link layer protocols.

CO3 Design subnets and evaluate the IPV4/IPV6 addresses to fulfill the network requirements of an organization.

CO4 Explain the details of Transport Layer Protocols (UDP, TCP, HTTP, DNS, and SMTP) and suggest appropriate protocol in reliable/ unreliable communication.

CO5 Ability to design and evaluate the protocol structures and test the validation of the model's correctness.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	M	M	L	L	M	H	M	-	-
CO2	H	H	M	M	M	M	M	H	-	-
CO3	H	H	M	M	H	M	M	M	-	L
CO4	H	M	H	H	H	H	M	H	M	-
CO5	H	H	H	H	H	H	M	M	M	M

IoT Security & Trust
Syllabus

Prerequisite: IoT, Cryptography
L-T-P: 4-0-0**, Credits:** 4 **Type:** Minors/Honors(MR)

Course Description

IoT Security and Trust Course is designed to help CSE professionals strengthen their knowledge about the Internet of Things (IoT) and the security platforms related to it. They will also be able to identify the security, privacy and safety concerns related to the implementation of an IoT infrastructure.

Course Objectives

Students undergoing this course are expected to:	
1	Have the knowledge of cryptographic algorithms and protocols
2	Gain the understanding of IoT architecture and their security requirements
3	Analyse the vulnerability in IoT communication architecture
4	Know basics of cloud security and cyber crime

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	To **understand** the basics of the security principles and methodologies for Internet of Things.	Identify/Knowledge
CO-2	To **obtain** the basic knowledge about application challenges and security requirement	Identify/Knowledge
CO-3	To **understand** the Vulnerability models for IoT	Remember
CO-4	To have an **understanding** of Trust and Identity models for IoT.	Understanding
CO-5	To **comprehend** about cloud security and understanding about Cyber Crime.	Comprehension

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	2	2	2	2	2	-	-	2	-
CO2	2	2	2	2	2	2	-	-	-	-
CO3	2	2	2	2	2	2	-	-	2	-
CO4	1	2	2	2	2	2	-	2	2	-
CO5	1	-	1	1	1	1	1	2	1	2
CO (Average) Course Average	**1.6**	**1.6**	**1.8**	**1.8**	**1.8**	**1.8**	**0.2**	**0.8**	**1.4**	**0.4**

Syllabus

UNIT I: Cryptography – Need and the Mathematical basics- History of cryptography, Cryptographic primitives and its role in IoT, Encryption and Decryption, Hashes, Digital Signatures, Random number generation, Cipher suites, key management fundamentals, symmetric ciphers, block ciphers, DES, AES. Public-key cryptography, RSA, Daffier-Hellman.

UNIT II: Security Requirements in IoT Architecture, Security Concerns in IoT Applications, IOT security frame work, Security in hardware, Boot process, OS & Kernel application, Insufficient Authentication/Authorization, Insecure Access Control, Threats to Access Control, Privacy, and Availability, cryptographic controls built into IoT messaging and communication protocols – IoT Node Authentication.

UNIT III: Vulnerabilities – Vulnerability of IoT and elementary blocks of IoT Threat modeling, Key elements, Security Identity management Models and Identity management in IoT, Approaches using User-centric, Device-centric and Hybrid; Identity lifecycle, Authentication credentials, IoT Authorization with Publish / Subscribe schemes and access control, Identity and Trust. Identity management framework Establishment: Cryptosystem, Mutual establishment phases.

UNIT IV: Trust management lifecycle, Trust and Trust models for IoT, Web of trust models, self-organizing Things, Preventing unauthorized access. Capability-based access control schemes, Concepts, identity-based and identity-driven, Light weight cryptography, need and methods, IoT use cases Comparison on security analysis. Concerns in data dissemination, Lightweight and robust schemes for Privacy protection.

UNIT V: Cloud services and IoT, offerings related to IoT from cloud service providers, Cloud security, Digital identity management in cloud, Classical solutions, alternative solutions, Cloud IoT security controls, an enterprise IoT cloud security architecture, New directions in cloud enabled IoT computing. Cyber Crimes and Laws – Hackers – Dealing with the rise tide of Cyber Crimes – Cyber Forensics and incident Response – Network Forensics.

Text Books

1. John R. Vacca, “Computer and Information Security Handbook”, Elsevier, 2013. Parikshit Narendra Mahalle, Poonam N. Railkar, “Identity Management for Internet of Things”, River Publishers, 2015.
2. William Stallings, “Cryptography and Network security: Principles and Practice”, 5th Edition, 2014, Pearson Education, Indi.

Reference Books

1. Maryline Laurent, SamiaBouzefrane, “Digital Identity Management”, Elsevier, 2015.
2. Alasdair Gilchrist, “IoT security Issues”, Oreilly publications, 2017.

IoT, ML and Deep Learning
Syllabus

Prerequisite: Basics of Computer Programming, Mathematics and Basics of Machine Learning
L-T-P: 4-0-0, **Credits:** 4**Type:** Minors/Honors (MR)

Course Description

This Course focuses on hands-on IoT concepts such as sensing, actuation and communication. Develop intelligent systems and analyse data in science and engineering using machine Learning concept. Various machine Learning algorithms and DLT.

Course Objectives

Students undergoing this course are expected to:	
1	To cover the fundamentals of IoT, machine learning and deep learning techniques.
2	To provide a basic understanding of IoT architecture and its protocols
3	To explore students various machine learning and deep learning techniques.
4	After having undergone this course, students able to identify the challenges faced in the implementation of machine learning and deep learning techniques through various case studies

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	To **understand** and remember core concepts of IoT, machine learning, and deep learning.	Identify/Knowledge
CO-2	To **explain** various IoT protocols, and IoT enabling platforms.	Identify/Knowledge
CO-3	To **analyze** the difference among different types of learning, clustering and classification methods in machine learning.	Comprehension/ Analyzing
CO-4	To **understand** and analyze performance of convolution neural networks and recurrent neural networks.	Analyzing/Understanding
CO-5	To apply machine learning and deep learning techniques for various applications	Creating

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)
2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**P10**
CO1	2	2	2	2	2	2	-	-	-	-
CO2	2	2	2	2	2	2	-	-	-	-
CO3	2	2	2	-	-	-	-	-	-	-
CO4	1	2	2	-	-	-	-	-	-	-
CO5	1	1	1	2	2	2	-	-	-	-

CO (Average) Course Average	**1.6**	**1.8**	**1.8**	**1.5**	**1.5**	**1.5**	**-**	**-**	**-**	**-**

Syllabus

UNIT I: Internet of Things (IoT): Introduction, components of IoT, IoT architecture, communication models in IoT, IoT protocols, IoT enabling platforms, applications, big data analytics

UNIT II: Machine Learning (ML): Introduction, types of learning, hypothesis space and inductive bias, evaluation, cross-validation, clustering, classification, k-nearest neighbours, support vector machines, decision trees, ensemble methods

UNIT III: Deep Learning Techniques (DLT): Introduction, feed forward neural networks, deep networks, convolution neural networks, stacking, striding and pooling, activation function, recurrent neural networks, auto-encoders

UNIT IV: Case Studies: Transportation/logistics, smart city, smart home, industry control, agriculture and smart healthcare applications, and latest research papers

Text Books

1. Pethuru Raj and Anupama C. Raman, “The Internet of Things: Enabling Technologies, Platforms, and Use Cases,” CRC Press, 2017
2. Wiley Editorial, “Internet of Things, An Indian Adaptation: Concepts and Applications,” Wiley, 2021.
3. John PaulMueller, and Luca Massaron. “Machine learning for dummies,” John Wiley & Sons, 2021.

Reference Books

1. AlpaydinEthem, “Introduction to Machine Learning” PHI, 2015
2. Ian Goodfellow,YoshuaBengio, and Aaron Courville, “Deep Learning,” The MIT Press, 2016.
3. Josh Patterson, and Adam Gibson, “Deep Learning: A Practitioner's Approach,” First Edition, O’Reilly, 2017.

Course Works in Research Basket

Course Name: Soft Computing

Syllabus

Prerequisite: Nil **L-T-P: 4-0-0, Credits: 4**

Soft Computing

Course Objectives:

The students hould be motivated to:

- ✓ appreciate the need and application of Soft Computing
- ✓ learn the various soft computing techniques
- ✓ be familiar with design of various neural networks and its applications
- ✓ be exposed to Fuzzy Sets and Crisp Sets
- ✓ get exposed to fuzzy logic & genetic programming.

Course Outcomes:

Upon completion of the course, the student should be able to:

1. Understand the architectures of several neural network paradigms
2. Apply fuzzy logic and reasoning to handle uncertainty and solve various computing problems
3. Evaluate and compare Neuro-Fuzzy Modeling approaches for a given problem
4. Identify and describe soft computing models and their applications in building intelligent systems.

CO-PO mapping

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	M	M	L	M					
CO2	H	H	H	M	H					
CO3	M	M	H	L	H					
CO4	H	H	H	M	M				M	M

Course Outline (To be covered in 40 Lectures)

Unit 1: Introduction to Soft Computing, Introduction to Fuzzy logic, Fuzzy membership functions, Properties of Fuzzy Sets, Operations on Fuzzy sets, Fuzzy relations, Operations on Crisp and Fuzzy Relations (8 L)

Unit 2: Fuzzy relations, Fuzzy proposition, Two Valued Logic vs. Multi valued Logic, Fuzzy propositions vs. Crisp propositions, Canonical Representation of Fuzzy Proposition, Fuzzy implications, Fuzzy inferences, Interpretation of Fuzzy Rules (8 L)

Unit 3: Defuzzification Techniques, Need for Defuzzification, Defuzzification Techniques - Lambda-cut or Alpha Cut method, Weighted average method, Maxima method and Centroid Methods, Weighted Average Method, Fuzzy logic controller (8 L)

Unit 4: Concept of GA, Genetic algorithms - Basic concepts, encoding, fitness function, reproduction, Differences in GA and traditional optimization methods GA Operators: Encoding, GA Operators: Selection, Crossover & Mutation, MOEA Approaches: Pareto, Introduction to ANN, ANN Architecture, ANN Training, Applications of ANN (10 L)

Unit 5: Recent Trends in Soft Computing and its Applications (6 L)

TextBooks:

1. J.S.R.Jang,C.T.SunandE.Mizutani,"Neuro-FuzzyandSoftComputing",PHI- Pearson Education 2004.
2. S.N. Sivanandamand S.N. Deepa, "Principlesof Soft Computing", Wiley India Pvt Ltd, 2016.

Reference Books:

1. David E. Goldberg, "Genetic Algorithm in Search optimization and Machine Learning "Pearson Education India,2013.
2. S. Rajasekaran and G. A. Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm : Synthesis & Applications", Prentice-Hall of India Pvt.Ltd.,2006.
3. GeorgeJ.Klir, UteSt. Clair, BoYuan, "Fuzzy Set Theory: Foundations and Applications" PrenticeHall,1997.
4. JamesA. Freeman, David M.Skapura," Neural Networks Algorithms, Applications, and Programming Techniques, Pearson EducationIndia,1991.

Course Name: Introduction to Bioinformatics and Genomics

Syllabus

Prerequisite: Nil **L-T-P: 4-0-0, Credits: 4**

Introduction to Bioinformatics and Genomics

Course Objectives

This course is designed for students: To get introduced with biology, molecular biology, mathematics, information technologies and computer science and try to develop the understanding of different methodologies used in computational biology for the analysis and development of models for better interpretation of biological data to extract knowledge. They are also able to apply the computational methods for the sequence alignment, database searches, protein structure prediction, drug discovery, DNA microarray and genome analysis etc.

Course Outcomes

1. Describe the basic concepts of Genomics, Bioinformatics and its significance in Biological data analysis.
2. Understanding of algorithms and tools for the sequence alignment and database searches.
3. Applying the computational methods for the protein structure prediction and genome analysis.
4. Analysis of microarray, genome assembling and sequencing data using computational techniques.

CO-PO mapping

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	L	M	M	L	L		M		H	
CO2	H	H	H	H	M					
CO3	M	H	H	M	M					
CO4	H	H	H	H	M		M		H	M

Course Outline (To be covered in 40 Lectures)

Unit I:Introduction: Systems approach in biology, Genomes and Molecular evaluation, Central dogma of molecular biology, Genome sequencing, Bioinformatics objectives and overviews, Major Biological databases and bioinformatics tools, Interdisciplinary nature of Bioinformatics, Overview of the bioinformatics applications.

Unit II: Sequence Alignment: Access to sequence data and related information,Pairwise sequence alignments: Dot matrix for sequence alignment, Dynamic programming for Local and Global alignment, Scoring matrices, and Sequence-based database searches: Need of sequence based database search, BLAST and FASTA algorithms, various versions of basic BLAST and FASTA, Multiple sequence alignment.

Unit III:Protein Structure Prediction & DNA Microarray: Secondary structure prediction (Statistical method: Chou Fasman and GOR method, Neural Network and Nearest neighbor method) and Tertiary structures prediction (Homology Modeling); Structure visualization methods (RASMOL, CHIME etc.); Protein Structure alignment and analysis, Introduction to DNA microarray, microarray data analysis.

Unit IV: Recent/Current Trends in Bioinformatics i.e. Analysis of gene expression profile for cancer, Analysis of Genome assembling and sequencing using machine learning, proteomics in drug discovery, Single cell sequencing, Precision medicine etc.

Text Books

1. Lesk, Arthur. *Introduction to bioinformatics*. Oxford university press, 2019.
2. Pevsner, Jonathan. *Bioinformatics and functional genomics*. John Wiley & Sons, 2015.
3. Gibas, Cynthia, Per Jambeck, and James Fenton. *Developing bioinformatics computer skills*. " O'Reilly Media, Inc.", 2001.

Reference Books

1. Pevzner, Pavel. *Computational molecular biology: an algorithmic approach*. MIT press, 2000.
2. Jones, Neil C., Pavel A. Pevzner, and Pavel Pevzner. An introduction to bioinformatics algorithms. MIT press, 2004
3. Misener, Stephen, and Stephen A. Krawetz, eds. *Bioinformatics methods and protocols*. Vol. 132. Springer Science & Business Media, 2000.

Course Name: Big Data

Syllabus

Prerequisite: Nil **L-T-P: 4-0-0, Credits: 4**

Big Data

Course Objectives

This course is intended to introduce you with Big Data, its types, characteristics, limitations etc. This will help in understanding and learning to work with Big Data Ecosystem technologies, which include HDFS, MapReduce, Hive, Pig, Yarn, Scoop, & Spark.

Course Outcomes

After this course, students should be able to

CO1: Understand the characteristics, history and background of Big data and Hadoop.

CO2: Capable of doing data modelling and applying Multivariate Data Analysis.

CO3:Understand the ecosystem components like HDFS, MapReduce, Scoop, Hive, Pig, Mahout, HBase.

CO4: Understand the Apache spark Architecture and its usage.

CO5: Work in Big data use cases.

Course Outline (To be covered in 40 Lectures)

Unit I: Introduction to Big Data. Large dimension small size multivariate data analysis; tackling the problems of estimation and inference. Classification of Big Data; Screening and Variable Selection.

Unit II: Lasso Regression; Projection Methods. Introduction to Markov Chain Monte Carlo (MCMC) Simulations; MCMC techniques for Bayesian Modeling of Big Data.

Unit III: Introduction to Hadoop, History and Milestones of Hadoop, Hadoop Architecture, HDFS, Map Reduce, Yarn, Hive, Scoop, Pig, Mahout, Hbase, Spark.

Unit IV: Big Data — Use Cases.

Text Books

1. Tom White, “Hadoop: The Definitive Guide”, O’Reilly Media.
2. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data”, McGraw-Hill Publishing.
3. Hastie, T., Tibshirani, R. and Friedman, J. “The Elements of Statistical Learning”. Berlin: Springer, 2009.
4. Buhlmann, P. and van de Geer, S. “Statistics for High-Dimensional Data: Methods, Theory and Applications.” Berlin: Springer, 2011.
5. Cressie, N., “Statistics for Spatial Data”, Revised Edition. NJ: Wiley Classics, 2015.
6. Gamerman, D., Hedibert, F. L. “Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference”, 2nd ed. FL: Chapman and Hall/CRC, 2006.
7. Lecture Notes based on selected recent papers on Big Data Modeling and Analysis.

Program Outcomes (POs)

Students in the M.Tech. CSE (Specialization in Data Science) program will attain:

PO1: ability to apply knowledge of computing, mathematics, statistics, science and engineering fundamentals appropriate to the discipline.

PO2: ability to apply knowledge of machine learning and computational intelligence techniques for solving real world problems encountered in Industries.

PO3: ability to design, implement, and evaluate an intelligent system, process, component, or program to meet desired needs with appropriate consideration for public health and safety, cultural, societal and environmental considerations.

PO4: ability to design and conduct experiments, as well as to analyze and interpret real world data.

PO5:understand the professional and ethical responsibilities in engineering practice.

PO6: ability to use current techniques, skills, and modern tools necessary for computing practice.

PO7:demonstrate the knowledge and understanding of engineering and management principles in the area of data science to manage research based projects in multidisciplinary environment.

PO8: ability to communicate effectively with a range of audiences

- ✓ Knowledge Outcome: pt. 1, 2
- ✓ Technical Outcome: pt. 3, 4
- ✓ Soft Skills Outcome: pt. 7,8
- ✓ Behavioural Outcome: pt. 3, 5

Program Educational Objectives (PEOs):

Objective 1. Excel in professional career, research and /or higher education by acquiring knowledge in mathematics, statistics, computing, science and engineering principles.

Objective 2. Analyze real life problems, design appropriate intelligent system to provide solutions that are technically sound, economically feasible and socially acceptable.

Objective 3. Exhibit professionalism, ethical attitude, communication skills, in their profession and adapt to current trends by engaging in lifelong learning.

Cos and Pos mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	2	2	1	2	-
CO2	3	3	2	1	1	2	1	2
CO3	3	2	2	3	1	2	2	1
CO4	3	2	2	2	2	2	1	2
CO5	3	3	1	1	1	2	2	2
Avg	3	2.6	2.0	1.8	1.4	1.8	1.6	1.4

Course Name: High Performance Computing

Syllabus

Prerequisite: Nil **L-T-P: 4-0-0, Credits: 4**

Course Objectives

- ✓ **Introduce the concepts of modern processors and parallel computing paradigms.**
- ✓ Introduce the parallel programming concepts using OpenMP and MPI.
- ✓ Introduce the current research trends and technologies in HPC.

Course Outcomes

CO1: Learning about modern day processors, parallelism in processing and advance architectures.

CO2: Gain understanding related to performance evaluation and improvement of processor architectures, and optimization of design constraints.

CO3: Gain knowledge about programming of parallel architectures, and load sharing for higher throughput.

CO4: Leaning of current technologies in HPC example Nvidia GPU, Google TPU and Intel VPU
CO5: Design parallel programs for shared-memory architectures and distributed-memory architectures using modern tools like OpenMP and MPI

Course Outline (to be covered in 40 lectures)

Unit –I:

Modern Processors: Stored Program Computer Architecture, General purpose cache-based microprocessor, Performance based metrics and benchmarks, Moore's Law, Pipelining, Superscalarity, SIMD, Memory Hierarchies, Cache-mapping, prefetch, Multicore processors, Multithreaded processors, Vector Processors, Design Principles, Maximum performance estimates, Programming for vector architecture.

Unit-II:

Shared-memory parallel programming with OpenMP, Introduction to OpenMP, Parallel execution, Data scoping, OpenMP work sharing for loops, Synchronization Reductions, Loop scheduling, Miscellaneous, Case study: OpenMP parallel Jacobi algorithm, Advanced OpenMPWavefront parallelization, Efficient OpenMP programming, Profiling OpenMP programs.

Unit-III:

Distributed-memory parallel programming with MPI, Message passing MPI example, Messages and point-to-point communication, Collective communication, Non-blocking point-to-point communication, Virtual topologies, Example: MPI parallelization of Jacobi solver, Communication parameters, Synchronization serialization, Implicit serialization and synchronization–contention, Reducing communication overhead, Aggregating messages, Non-blocking vs asynchronous communication.

Unit-IV:

Hybrid parallelization with MPI and OpenMP, Basic MPI/OpenMP programming models, Vector mode implementation, Task mode implementation;

Case studies: Nvidia-GPU Computing, CUDA Introduction, Google TPU, Intel VPU

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

On the scale of 3

Text Books

1. Georg Hager and Gerhard Wellein, “Introduction to High Performance Computing for Scientists and Engineers”, Chapman & Hall / CRC, 2011.
2. Charles Severance, Kevin Dowd, High Performance Computing, O'Reilly Media, 2nd Edition, 1998.
3. Michael Quinn, “Parallel Programming in C with MPI and OpenMP”, Indian edition, McGraw Hill Education, 2017.
4. John L. Hennessy and David A. Patterson, “Computer Architecture -- A Quantitative Approach”, 4th Edition, Morgan Kaufmann Publishers, 2017, ISBN 13: 978-0-12-370490-0.
5. Barbara Chapman, Gabriele Jost and Ruud van der Pas, “Using OpenMP: portable shared memory parallel programming”, The MIT Press, 2008, ISBN-13: 978-0-262-53302-7.
6. Marc Snir, Jack Dongarra, Janusz S. Kowalik, Steven Huss-Lederman, Steve W. Otto, David W. Walker, “MPI: The Complete Reference, Volume2”, The MIT Press, 1998, ISBN: 9780262571234.
7. Pacheco S. Peter, “Parallel Programming with MPI”, Morgan Kaufman Publishers, 1992, Paperback ISBN: 9781558603394.

Course Name: Medical Image Processing

Syllabus

Prerequisite: Nil **L-T-P: 4-0-0, Credits: 4**

Medical Image Processing

Course Objectives

- ✓ To study the image fundamentals like: image enhancement, denoising, deconvolution, filtering, edge detection, segmentation, registration and mathematical transforms necessary for image processing.
- ✓ To describe and understand image modalities: X-ray, MRI, CT, PACs etc.
- ✓ To study image restoration procedures and feature extraction methods.
- ✓ To study the image segmentation techniques and classification approaches.
- ✓ To deploy the gained knowledge for developing the new medical image based application software's and projects.

Course Outcomes

- ✓ Gains knowledge on fundamental concepts of medical image processing.
- ✓ Understands the medical image modalities representation formats, along with storage challenges and solutions.
- ✓ Understands the feature extraction methods for various image modalities and their registration techniques.
- ✓ Understands the latest advancements in medial image: segmentation techniques, machine learning and deep learning techniques suitable for project development.
- ✓ Can design and develop new projects/software's using the acquired knowledge as team or as an individual.

Students undergoing this course are expected to:	
CO-1	To understand the medical image fundamentals, segmentation, registration and mathematical transforms necessary for medical image processing.
CO-2	To understand image modalities such as: X-ray, MRI, CT, PACs etc.
CO-3	Design and implement techniques for the detection of regions of interest.
CO-4	To study and understand feature extraction methods, segmentation techniques, validation and evaluation models for better classification and recognition.
CO-5	To deploy the gained knowledge for developing the new medical image based application software's and projects.

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**
CO1	3	2	2							
CO2	2	3	2							
CO3	2	1	1		2	3				
CO4	2	2	2	2	2	3				1
CO5	1	1	1	2	3	3				3
CO (Average) Course Average	**2**	**1.8**	**1.6**	**0.8**	**1.4**	**1.8**	**0**	**0**	**0**	**0.8**

Course Outline (To be covered in 40 Lectures)

Unit I: Fundamentals of Medical Image Processing. (10L): Image enhancement in spatial domain, Image enhancement in frequency domain, Image restoration, Image segmentation, Image compression.

Unit II: Medical Imaging Modalities and Picture Archiving and Communication Systems. (5L): Ultrasonic Image, Magnetic Resonance Image, Computed Tomography Image, X-ray Image. Architecture and operation of PACS, PACS-based imaging informatics, Integration of PACS with Hospital Information Systems and Radiology Information Systems, Standards, interoperability and workflow protocols.

Unit III: Registration and Feature Extraction Methods: (12L): Shape related methods, Fourier descriptors, Texture Analysis methods, point descriptors. Image restoration: Degradation models, blur models, spectral blur estimation and super resolution method.

Unit IV: Segmentation and classification methods (13L): k-means, region growing, particle swarm optimization, edge based, snakes and contours based segmentation approaches. Classification Methods: SVM, ANN and deep learning classifications methods. Recent/Current Trends in Medical Image Processing Applications: Breast Tissue Detection, Tumor Imaging and treatment, Angiography, Bone x-ray analysis etc.

Text Books

1. Medical Image Analysis, Atam P. Dhawan, 2nd Edition, John Wiley & Sons.
2. Yoo, Terry S. Insight into images: principles and practice for segmentation, registration, and image analysis. AK Peters/CRC Press, 2004.
3. Bankman, Isaac, ed. Handbook of medical image processing and analysis. Elsevier, 2008.

Reference Books

1. Bankman, Isaac N., and SerbanMorcovescu. "Handbook of medical imaging. processing and analysis." Medical Physics 29.1 (2002): 107-107.
2. Parker, Jim R. Algorithms for image processing and computer vision. John Wiley & Sons, 2010.
3. Bernstein, Matt A., Kevin F. King, and Xiaohong Joe Zhou. Handbook of MRI pulse sequences. Elsevier, 2004.
4. Shen, Dinggang, Guorong Wu, and Heung-Il Suk. "Deep learning in medical image analysis." Annual review of biomedical engineering 19 (2017): 221-248.

Course Name: Data Security and Application Security

Syllabus

Prerequisite: Basics of Computer Networks and Operating System **L-T-P: 4-0-0, Credits: 4**

Data Security and Application Security

Course Objectives

Students undergoing this course are expected to:

- Understand of data security tools such as access control mechanisms, authentication tools and cryptographic constructs.
- Familiar with current state of the art in data and application security domains.
- Learn the basic concepts cryptographic techniques to provide data security.
- Learn the security issues related to web application and mobile applications
- Build novel security solutions for current network infrastructure to meet the requirement of current state of art.

Course Outcomes

At the end of the course, students should be able to:

- Identify security vulnerabilities and threat and able to countermeasure it.
- Write security policies to provide data confidentialities and integrity.
- Demonstrates how security solutions meet the security requirements.
- Evaluate and analyse the security algorithms and protocol in terms of performance and security.
- Develop security product that meet the current research and industry requirements.

Course Outcomes

CO Numbers	**Course Outcomes (Action verb should be in italics)**	**Bloom's taxonomy**
CO-1	Understand terms related to Cryptography, Attack and Threats types	Understanding/ Comprehension
CO-2	Understanding and identifying various authentication, integrity and access control mechanisms	Understanding/ Knowledge
CO-3	Be able to identify appropriate attacks, threats for web and mobile data and their applications problems	Identify/Knowledge
CO-4	Be able to evaluate the quality software source code.	Analyzing /Identify
CO-5	Critically evaluate the data and application models and their implications in the real life experience.	Evaluate/Analyzing

Mapping Course Outcomes with Program Outcomes:

Correlation Matrix

Note: Enter numbers 1, 2 or 3, where the correlation levels are matching

1. Slightly (Low)

2. Moderately (Medium)
3. Substantially (High)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10
CO1	3	2	2	2	2	2	1	2	0	2
CO2	2	2	2	2	2	1	1	2	0	1
CO3	2	2	2	1	2	2	1	2	1	2
CO4	2	1	2	2	1	2	2	2	2	1
CO5	2	2	2	2	2	3	2	2	3	2
CO (Average) Course Average	2	2	2	2	2	2	1	1	2	2

Course Outline (To be covered in 40 Lectures)

UNIT-1: Introduction to data and application security, Information and network security policies: Confidentiality Policies (The Bell-LaPadula Model), Integrity Policies (Biba Integrity Model), Hybrid Policies (Chinese wall model, Clark-Wilson Models). [10]

UNIT-II: Information security basics, Data Security, Threats to Data Security due to Malware, viruses, Trojan, Spyware and Botnets System security, User Security, Program security, Operating System Security [10]

UNIT-III: Introduction to encryption (Symmetric and Asymmetric), Security services - Confidentiality, Authentication, Integrity, Access Control mechanisms, Web security, Mobile application security, Database Security, E-Mail Security, DNS Security, IDS systems, Firewall [10]

UNIT-IV: Recent/Current Trends in this Course [10]

Text Books

1. Matt Bishop, Introduction to Computer Security, Eighth Impression, 2012, Pearson
2. Eric Cole, Network Security Bible, 2nd Edition, 2009, Wiley.
3. Dieter Gollmann, Computer Security, 3ed, 2014, Wiley
4. Stallings William, Cryptography and Network Security - Principles and Practice | Seventh Edition, 2017.

Course Name: Data Analytics

Syllabus

Prerequisite: Nil **L-T-P: 4-0-0, Credits: 4**

Data Analytics

Course Objectives

- ✓ Gather sufficient relevant data, conduct data analytics using scientific methods, and make appropriate and powerful connections between quantitative analysis and real-world problems.
- ✓ Demonstrate a sophisticated understanding of the concepts and methods; know the exact scopes and possible limitations of each method; and show capability of using data analytics skills to provide constructive guidance in decision making.
- ✓ Show substantial understanding of the real problems; conduct deep data analytics using correct methods; and draw reasonable conclusions with sufficient explanation and elaboration.
- ✓ Write an insightful and well-organized report for a real-world case study, including thoughtful and convincing details.

Course Outcomes

CO 1: The students are able to know about data, data analytics methods, conduct data analytics using scientific methods.

CO 2: The students are able to gain in-depth of mathematics and statistics skills necessary to apply high-level analytical thinking to data analysis problems.

CO3: The students are able to an exposure to the basics of computer programming and analytics software to implement the data analysis problems.

CO4: The students are able to practice data analytics techniques on real-world data sets and to compile a project portfolio to show their knowledge potential.

CO 5: The students are able to take better business decisions by using advanced techniques in data analytics.

Program Outcomes (POs):

Students in the B Tech CSE program will attain:

PO1	Ability to apply knowledge of computing, mathematics, science, engineering and humanities.
PO2	Ability to design and conduct experiments, as well as to analyze and interpret data.
PO3	Ability to analyze a problem, identify and define the computing requirements appropriate to its solutions
PO4	Ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional
PO5	Ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
PO6	Ability to design, implement, and evaluate a software or a software/hardware system, component or process to meet desired needs within realistic constraints such as memory, run time efficiency.

PO7	Understanding of professional, ethical, legal, political security and social issues and responsibilities.
PO8	Ability to communicate effectively with knowledge of contemporary issues.
PO9	Ability to analyze the local and global impact of computing on individuals, organizations, and society.
PO10	Understanding of the need for acquiring an ability to engage in lifelong learning and continuing professional development

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	H	H	H	M	M	L	L	-	-
CO2	H	H	H	M	M	H	L	M	M	-
CO3	H	M	M	H	H	H	M	H	L	L
CO4	M	M	H	H	H	H	L	H	L	L
CO5	M	M	H	H	H	H	M	M	H	H

Course Outline (To be covered in 40 Lectures)

Unit I: Introduction, processing the information and getting to know the data, Data Summarization and Visualization.

Unit II: Standard Linear Regression, Local Polynomial Regression, Statistical modeling and Nonlinear Regression.

Unit III: Logistic Regression, Model Selection, Classification, Clustering, Decision Trees.

Unit IV: Reducing the Dimensions in regression, Text mining and sentiment analysis and Network Data.

Text Books

1. Data Mining and Business Analytics with R, by Johannes Ledolter; Publisher: Wiley (2013), ISBN-13: 978-1118447147;
2. An Introduction to Statistical Learning with Application in R, by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani; Publisher: Springer (2013); ISBN-13: 978-1461471370;

Reference Book

1. Elements of Statistical Learning: Data Mining, Inference, and Prediction, by Trevor Hastie, Robert Tibshirani and Jerome Friedman.