Course Curriculum

and

Detailed Syllabus

for

Bachelor of Technology Programme

in

Computer Science and Engineering

राष्ट्रीय प्रौद्योगिकी संस्थान गोवा

NATIONAL INSTITUTE OF TECHNOLOGY GOA

कोट्टामल प्लाटू, कुंकोलिम, सालसेट, दक्षिण गोवा, गोवा - ४०३७०३, इंडिया

Kottamoll Plateau, Cuncolim, Salcete, South Goa, Goa- 403703, India

Programme Structure Summary

Sl. No.	Classifications	Course Type	Credits For CGPA	Courses
1	Basic Sciences	**BS**	21	MA→11, PH→5, CY→5
2	Basic Engineering Sciences and Technical Arts	**ES**	21	EM→3, BMC→3, BES→6, CPPS→4, ED→3, WP→2
3	Humanities and Social Sciences	**HU & HS**	7	PC →4, ECO→ 3
4	Indian Knowledge Systems	**IKS**	5	HH →2 and an Open Elective Course →3
5	Others: Liberal Arts, Innovation & Entrepreneurship	**OT**	2	LA→1, IE→1
6	Mandatory Learning Courses	**MLC**	2	PE→0, ES→ 1, PEHV→ 1
7	Department Core	**DC**	83 - 86	Core Theory and Lab courses, Comprehensive Examination →1, Seminar→1, Summer Internship→1, Project Work→5
8	Department Elective (including MOOCs or any other as approved by the Institute)	**DE**	21-27	7-9 Electives
9	Open Elective (including MOOCs or any other as approved by the Institute)	**OE**	0-6	Upto 2 Open Electives
Total Credits			**168**	
10	Minor Program	**MR**	**18**	

Semester-wise Credit Distribution for B.Tech Program

Semester	Total Credits
I	23
II	23
III	24
IV	24
V	22
VI	21
VII	19
VIII	12
Total Credits	**168**

Semester-Wise Distribution of the Courses

1st Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	MA100	Matrices and Advanced Calculus	BS	3-1-0	4
2.	PH100	Engineering Physics	BS	3-0-0	3
3.	CS100	Computer Programming and Problem Solving	ES	3-0-0	3
4.	EE100	Basics of Electrical Engineering	ES	2-0-0	2
5.	ME100	Engineering Mechanics	ES	3-0-0	3
6.	HU100	Liberal Arts	OT	0-0-2	1
7.	PH101	Engineering Physics Lab	BS	0-0-3	2
8.	CS101	Computer Programming Lab	ES	0-0-2	1
9.	EE101	Basics of Electrical Engineering Lab	ES	0-0-3	1*
10.	ME101	Engineering Drawing	ES	1-0-3	3
	Total Credits				**23**

2nd Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	MA150	Differential Equations and Vector Calculus	BS	3-1-0	4
2.	CY150	Engineering Chemistry	BS	3-0-0	3
3.	HU150	Professional Communication	HU	2-0-3	4
4.	EC150	Basics of Electronics Engineering	ES	2-0-0	2
5.	ME150	Basics of Mechanical and Civil Engineering	ES	3-0-0	3
6.	HU151	Health & Happiness	IKS	2-0-0	2
7.	CY151	Engineering Chemistry Lab	BS	0-0-3	2
8.	EC151	Basics of Electronics Engineering Lab	ES	0-0-3	1*
9.	ME151	Workshop Practices	ES	0-0-3	2
10.	PE150	Physical Education	MLC	1-0-2	0#
	Total Credits				**23**

***: Half-Semester Course, #: Non credit Course**

3rd Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	MA201	Probability, Statistics and Queuing Theory	BS	3-0-0	3
2.	CS200	Data Structures	DC	3-1-0	4
3.	CS201	Digital Systems Design	DC	3-0-0	3
4.	CS202	Discrete Mathematics	DC	3-1-0	4
5.	CS203	Object Oriented Programming	DC	3-1-0	4
6.	CS204	Data Structures Lab	DC	0-0-3	2
7.	CS205	Digital Systems Design Lab	DC	0-0-3	2
8.	CS206	Object Oriented Programming Lab	DC	0-0-3	2
Total Credits					**24**

4th Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	IE250	Innovation & Entrepreneurship	OT	1-0-0	1
2.	CS250	Database Systems	DC	3-1-0	4
3.	CS251	Software Engineering	DC	3-0-0	3
4.	CS252	Computer Organization and Architecture	DC	3-1-0	4
5.	CS253	Theory of Computation	DC	3-1-0	4
6.	CS254	Microprocessor and Microcontrollers	DC	3-1-0	4
7.	CS255	Database Systems Lab	DC	0-0-3	2
8.	CS256	Microprocessor and Microcontrollers Lab	DC	0-0-3	2
Total Credits					**24**

5th Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	CS300	Operating Systems	DC	3-1-0	4
2.	CS301	Introduction to Machine Learning	DC	3-0-0	3
3.	CS302	Design and Analysis of Algorithms	DC	3-0-0	3
4.	CS303	Number Theory and Cryptography	DC	3-0-0	3
5.	CS5XX	Elective-I	DE/OE*	3-0-0	3
6.	CS304	Operating Systems Lab	DC	0-0-3	2
7.	CS305	Design and Analysis of Algorithms Lab	DC	0-0-3	2
8.	CS306	Seminar	DC	0-0-2	1
9.	HU350	Professional Ethics and Human Values	MLC	1-0-0	1
Total Credits					**22**

6th Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	CS350	Distributed Systems	DC	3-0-0	3
2.	CS351	Compiler Design	DC	3-0-0	3
3.	CS352	Computer Networks	DC	3-1-0	4
4.	CS5XX	Elective II	DE/OE*	3-0-0	3
5.	CS353	Compiler Design Lab	DC	0-0-3	2
6.	CS354	Computer Networks Lab	DC	0-0-3	2
7.	HS350	Industrial Economics	HU & HS	3-0-0	3
8.	ES300	Environmental Studies	MLC	1-0-0	1
Total Credits					**21**

7th Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	CS5XX	Elective III	DE/OE*	3-0-0	3
2.	CS5XX	Elective IV	DE/OE*	3-0-0	3
3.	CS5XX	Elective V	DE/OE*	3-0-0	3
4.	CS5XX	Elective VI	DE/OE*	3-0-0	3
5.	CS400	Summer Project/Industrial Training	DC	0-0-2	1
6.	CS401	Comprehensive Examination	DC	0-0-0	1
7.	CS402	Major Project I	DC	0-0-3	2
8.	IKS350	Indian Knowledge System	IKS	3-0-0	3
Total Credits					**19**

8th Semester

Sl. No.	Course Code	Course Name	Type	L-T-P	Credits
1.	CS450	Major Project II	DC	0-0-6	3
2.	CS5XX	Elective VII	DE/OE*	3-0-0	3
3.	CS5XX	Elective VIII	DE/OE*	3-0-0	3
4.	CS5XX	Elective IX	DE/OE*	3-0-0	3
Total Credits					**12**

* A student can register only one open elective (OE) per semester and a maximum of two OE in the B.Tech. tenure. This is excluding the Indian Knowledge System (IKS) course offered in 7th Semester. Since IKS is a mandatory OE, students are not allowed to register for an OE in that semester.

Detailed Syllabus

of

Core Courses

(3rd Semester Onwards)

3rd Semester

Course Code	Course Name	L	T	P	Credits
MA201	Probability, Statistics and Queuing Theory	3	0	0	3

Course Objective

The main objective of studying this course is to give overview on axiomatic definition of probability, random variable, distributions, moments, sampling distribution and hypothesis testing. It explains the concepts of probability theory and statistics which are needed for handling various real-world problems.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability

CO1. To find mean and variance of a given probability distribution,

CO2. To test the hypothesis for small and large samples,

CO3. To find the coefficient of correlation and lines of regression,

CO4. To understand the characteristics of a queuing model.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H		M							L
CO2	H	H	H		M							L
CO3	H	H	H		M							L
CO4	H	H	H		M							L

Syllabus

Module 1 : Elements of Probability

Introduction to Probability; Sample Space and Events; Probabilities Defined on Events; Conditional Probabilities; Independent Events; Total Probability; Bayes' Formula

Module 2 : Random Variables and Distribution

Concept of Random Variables; Distribution and Density Function;; Jointly Distributed Random of Variables; Conditional and Joint Density Distribution function; Function of Random Variables; Expected Value: Mean, Variance and moments of random variable; Joint moments, conditional expectation; covariance and correlation.

Some special distributions: Uniform and Gaussian distributions; Bernoulli, Binomial, and Poisson distributions.

Module 3 : Statistics

The Sample Mean; The Central Limit Theorem; The Sample Variance; Sampling Distributions from a Normal Population; Sampling from a Finite Population.

Parameter Estimation; Maximum Likelihood Estimators; Interval Estimates; Estimating the Difference in Means of Two Normal Populations.

Hypothesis Testing; Significance Levels; Tests Concerning the Mean of a Normal Population; Testing the Equality of Means of Two Normal Populations; Hypothesis Tests Concerning the Variance of a Normal Population; Hypothesis Tests in Bernoulli Populations; Tests Concerning the Mean of a Poisson Distribution; Regression (Basic concepts only)

Module 4: Queuing theory

Concepts, applicability, classification, birth and death process, Poisson queues, Characteristics of queuing models - single server (with finite and infinite capacities) model, multiple server (with infinite capacity only) model

Reference Books/Material

1. Ross, Sheldon M. “Introduction to probability and statistics for engineers and scientists”, Academic press, 2020.
2. Ross, Sheldon M. “Introduction to probability models”, Academic press, 2014.
3. Trivedi, Kishor S. “Probability & statistics with reliability, queuing and computer science applications”, John Wiley & Sons, 2008.
4. Ross, Sheldon M. “Stochastic processes”, John Wiley & Sons, 1995.
5. R. A. Johnson, Miller and “Freund’s Probability and Statistics for Engineers”, Pearson Publishers, 9th Edition, 2017.

6. John E. Freund, Benjamin M. Perles, “Modern Elementary Statistics”, 12th Edition, Pearson, 2013.

7. Hamdy A. Taha, “Operations Research: An Introduction”, Pearson, 2017, Tenth Edition.

8. S.C.Gupta and V.K.Kapoor, “Fundamentals of Mathematical Statistics”, 12th Edition, S.Chand & Co, 2020.

9. Kantiswarup, P.K.Gupta and Manmohan Singh, “Operations Research”, Sultan Chand & Sons, 2014.

Course Code	Course Name	L	T	P	Credits
CS200	Data Structures	3	1	0	4

Course Objective

The objective of the course is to familiarize oneself with basic data structures and to develop skills to analyze how the choice of data structures impacts the performance of programs.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability

CO1. To identify how the choice of data structures and algorithm design methods impacts the performance of programs,

CO2. To select the appropriate data structure and algorithm design method for any application,

CO3. To solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, binary search trees, and graphs and writing programs for these solutions,

CO4. To solve problems using algorithm design methods such as the greedy method, divide and conquer, dynamic programming, backtracking, branch and bound and writing programs for these solutions.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	M							
CO2	H	H	H	H	H							
CO3	H	H	H	H	H							
CO4	H	H	H	H	H							

Syllabus

Module 1: Introduction to data structures and algorithms, asymptotic notation for complexity analysis, Time and space complexity analysis; Arrays: one dimensional, multi-dimensional, Elementary operations.

Module 2: Stacks: Representation, elementary operations and applications such as infix to postfix, postfix evaluation, parenthesis matching, Queues: Simple queue, circular queue, dequeue, elementary operations and applications.

Module 3: Linked lists: Linear, circular and doubly linked lists, elementary operations and applications, linked list implementation of stacks and queues.

Module 4: Trees: Binary tree representation, tree traversal, complete binary tree, heap, binary search tree, height balanced trees: AVL tree, Red-Black tree; Rotations, Search and Update Operations on Balanced BSTs, Tries. Hashing: Implementation of Dictionaries, Hash Function, Collisions in Hashing, Separate Chaining, Open Addressing.

Module 5: Graphs: Representation, adjacency list, graph traversals: DFS, BFS, and their applications; Minimum spanning tree: Kruskal's, and Prim's algorithms; Shortest Path Problem- Dijkstra's, Bellman Ford, and Floyd-Warshall algorithms; algorithms on sorting: Selection sort, bubble sort, quick sort, merge sort, heap sort, searching: linear and binary search.

Reference Books/Material

1. Alfred V Aho, John E Hopcroft, Jeffrey D. Ullman, “Data Structures & Algorithms”, Pearson, 2013.
2. Aaron M. Tenenbaum, Yedidyah Langsam, Moshe J. Augenstein, “Data Structures using C”, Third Edition, Pearson, 2009.
3. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, Second Edition, Pearson, 2006.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L.Rivest, Clifford Stein, “Introduction to algorithms”, Third Edition, MIT Press, 2009.
5. Michael T.Goodrich and Roberto Tamassia, “Algorithm Design: Foundations, Analysis and Internet Examples”, Second Edition, Wiley-India, 2006.

Course Code	Course Name	L	T	P	Credits
CS201	Digital Systems Design	3	0	0	3

Course Objective

To understand the working of digital systems. Hardware components of the computer can be studied in greater depth.

Course Outcomes

At the completion of this course, the student shall acquire following knowledge and ability:

CO1. Identify the various number systems and application of these number systems to solve the basic digital arithmetic and logic operation

CO2. Simplification of Boolean logics & expression and to design basic problems using k-map applications.

CO3. Solve problems using various digital arithmetic and logic blocks like adders, subtractor etc. and implementation using various methods of logic optimizing for delays.

CO4. Design a sequential circuit using latches & Flip flops such as various types of counters and their applications in real time scenario.

CO5. To identify the infernal structure of digital gates and designing of various types logic gates, their propagation delays, power dissipation and fanout/fan in issues

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	L							
CO2	H	H	H	H	M							
CO3	H	H	H	H	M							
CO4	M	H	H	H	H	L						
CO5	H	H	H	M	M							

Syllabus

Module 1: Number Systems And Boolean Algebra: Review of binary, octal & hexadecimal number systems, representation of signed numbers, floating point number representation BCD, ASCII, EBCDIC, excess 3 codes, gray code-error detecting & correcting codes.

Boolean algebra: Postulates & theorems of boolean algebra, canonical forms, simplification of logic functions using Karnaugh map, Quine McCaskey method.

Module 2: Combinational Logic Design: Logic gates, implementation of combinational logic functions, encoders & decoders, multiplexers & demultiplexers, code converters, comparator, half adder, full adder, parallel adder, binary adder, parity generator/checker, implementation of logical functions using multiplexers.

Module 3: Sequential Logic Design-I : RS, JK, JK master, slave, D&T flip flops, level triggering and edge triggering, excitation tables, asynchronous & synchronous counters, modulus counters, shift register, Johnson counter, ring counter, timing waveforms, counter applications.

Module 4: Sequential Logic Design-II: Basic models of sequential machines, concept of state table, state diagram, state reduction through partitioning & implementation of synchronous sequential circuits, Introduction to asynchronous sequential logic design.

Module 5: Programmable Logic Devices: Semi Custom design, introduction to PLD's, ROM, PAL, PLA, FPGA, Implementation of digital functions. MOS inverters, CMOS inverters, comparison of performance of various logic families.

Reference Books/Material

1. Alan B.Marcovitz, "Introduction to logic design", 3rd Edition, McGraw-Hill Professional, 2009.
2. Giovanni De Micheli, "Synthesis and optimization of digital circuits", Tata McGraw- Hill Education 2003.
3. Zvi Kohavi, Niraj K. Jha, "Switching and finite automata theory", 3 rd Edition Cambridge University Press, 2011.
4. Douglas A. Pucknell &Kamran Shrayhian, "Basic VLSI design systems and circuits", Prentice Hall 2000.
5. ParagK.Lala, "Fault tolerant & fault testable hardware design", B.S publications, 2002.

Course Code	Course Name	L	T	P	Credits
CS202	Discrete Mathematics	3	1	0	4

Course Objective

This course introduces proof strategies and structures, Counting and combinatorics, elements of graph theory, introduction to abstract algebra and number theory.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability.

CO1. Explain basic concepts of mathematical reasoning. Also understand the different types of proven techniques like mathematical induction.

CO2. Understand the concepts of set theory, i.e., countable and uncountable sets, various types of functions and applications.

CO3. Understand the concepts of relations, partial ordering and equivalence relations. Apply the concepts of generating functions to solve the recurrence relations.

CO4. Apply the concepts of divide and conquer method and principle of inclusion and exclusion to solve some simple algorithms in discrete mathematics.

CO5. Understand various definitions in graph theory and study their properties. Also, understand the basic concept of abstract algebra and number theory

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							
CO5	H		M	H	L							

Syllabus

Module 1: Proof strategies:

Introduction, propositions, predicates, examples of theorems and proofs. Types of proof techniques. Axioms. Mathematical induction, Well-ordering principle, Strong Induction.

Module 2: Basic mathematical structures:

Sets, Russell's paradox, infinite sets, functions, comparing infinite sets using functions. Countable and countably infinite sets.

Module 3: Relations and functions:

Cartesian products and relations, equivalence relations and partitions of a set. More on equivalence relations, partial order relations, Posets. chains, anti-chains, topological sort, applications to (parallel) task scheduling. Lattices, Product and sum principles, Bijection principle, double counting, Handshake lemma, the binomial theorem, Pascal's triangle. Counting techniques: permutations and combinations with and without repetitions, estimating

factorials, solving recurrence relations. Pigeon-hole principle (PHP), its variants and its applications

Module 4: Introduction to graph theory:

Basic terminology, Konigsberg bridge problem, Eulerian graphs, Bipartite graphs, Representation of graphs, Graph isomorphism, Subgraphs, cliques and independent sets. Connected components, cut edges. Matchings, Perfect and maximum matchings, Minimum vertex covers.

Module 5: Abstract algebra:

Definition of an abstract group, subgroups, isomorphism, and cyclic groups, order of subgroups of a group, Lagrange's theorem, Modular arithmetic and applications to cryptography.

Reference Books/Material

1. Kenneth Rosen, “Discrete mathematics and its applications”, TMH, 2011
2. R. P. Grimaldi, “Discrete and Combinatorial Mathematics”, Pearson Education, Fifth Edition, 2007
3. Introduction to Graph Theory, 2nd Edition, by Douglas B West. Eastern Economy Edition published by PHI Learning Pvt Ltd.
4. Discrete Mathematics, 2nd Edition, by Norman L Biggs. Indian Edition published by Oxford University Press.

Course Code	Course Name	L	T	P	Credits
CS203	Object Oriented Programming	3	1	0	4

Course Objective

The objective of the course is to introduce students to the concept of object oriented programming and to develop programming skills to apply different object oriented concepts to solve problems.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Develop an understanding of fundamental concepts of object-oriented programming

CO2. Develop an understanding of object-oriented design and their mapping to object- oriented programming

CO3. Illustrate the object-oriented concepts and develop solutions using C++ and Java

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	M							
CO2	H	H	H	H	M							
CO3	H	H	H	H	H							

Syllabus

Module 1: Principles of OOP: Programming paradigms, basic concepts, benefits of OOP, applications of OOP

Introduction to C++: History of C++, structure of C++, basic data types, type casting, type modifiers, operators and control structures, input and output statements in C++.

Classes and objects: class specification, member function specification, scope resolution operator, access qualifiers, instance creation.

Functions: Function prototyping, function components, passing parameters, call by reference, return by reference, inline functions, default arguments, overloaded function.

Pointers: Array of objects, pointers to objects, this pointer, dynamic allocation operators, dynamic objects.

Module 2: Constructors: Constructors, parameterized constructors, overloaded constructors, constructors with default arguments, copy constructors, static class members and static objects.

Operator overloading: Overloading unary and binary operator, overloading the operator using friend function, stream operator overloading and data conversion.

Module 3: Inheritance: Defining derived classes, single inheritance, protected data with private inheritance, multiple inheritance, multi-level inheritance, hierarchical inheritance, hybrid inheritance, multipath inheritance, constructors in derived and base class, abstract classes, virtual function and dynamic polymorphism, virtual destructor.

Module 4: Exception Handling: Principle of Exception handling, exception handling mechanism, multiple catch, nested try, rethrowing the exception. Streams in C++: Stream classes, formatted and unformatted data, manipulators, user defined manipulators, file streams, file pointer manipulation, file open and close.

Templates: Template functions and Template classes.

Module 5: Object oriented programming using Java: Introduction to Java, bytecode, virtual machines, basic data types, operators, control structures, classes and objects, using Javadoc, packages, arrays, strings, inheritance, interfaces, exception handling, multithreaded programming, Java streams, developing user interfaces in Java.

Reference Books/Material

1. Bjarne Stroustrup, "The C++ Programming Language", 4th Edition, Addison-Wesley, 2013.
2. Stanley B Lippman, "The C++ Primer", 5th Edition, Addison-Wesley, 2012.
3. Herbert Schildt, "C++: the Complete Reference", 4th Edition, McGraw Hill, 2002.
4. Patrick Naughton and Herbert Schildt, "Java 2: The Complete Reference", 5th Edition, McGraw Hill, 2017.
5. Paul. Deitel, Harvey Deitel, "Java: How to program", 8th Edition, Pearson, 2017.

Course Code	Course Name	L	T	P	Credits
CS204	Data Structures Lab	0	0	3	2

Course Objective

The objective of the course is to illustrate the implementation of basic data structures and to develop programming skills to apply appropriate data structures for problem solving.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Implement linear and non-linear data structures for problem solving.

CO2. Design and implement algorithms using the appropriate data structure.

CO3. Implement and analyze searching and sorting techniques.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	M							
CO2	H	H	H	H	H							
CO3	H	H	H	H	H							

Syllabus

List of Experiments

1) Implementation of array operations, Structures & Unions
2) Stacks, Queues, Circular Queues, Priority Queues, Multiple stacks and queues
3) Infix to postfix expression using stack
4) Implementation of linked lists: stacks, queues
5) Implementation of doubly linked lists
6) Tree traversals
7) Open Addressing for Collision handling in Hashing
8) Implementation of DFS and BFS
9) Implementation of Kruskal's, and Prim's algorithms
10) Implementation of sorting algorithms: Bubble Sort, Insertion Sort, Quick Sort, Selection Sort, Merge Sort, Heap Sort

Reference Books/Material

1. Aaron M. Tenenbaum, Yedidyah Langsam, Moshe J. Augenstein, "Data Structures using C", Third Edition, Pearson, 2009.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Second Edition, PHI, 2009.
3. Alfred V Aho, John E Hopcroft, Jeffrey D. Ullman, "Data structures & algorithms", Pearson, 2013.
4. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Second Edition, Pearson Education, 2006.
5. Michael T.Goodrich and Roberto Tamassia, "Algorithm Design: Foundations, Analysis and Internet Examples", Second Edition, Wiley-India, 2006.

Course Code	Course Name	L	T	P	Credits
CS205	Digital Systems Design Lab	0	0	3	2

Course Objective

The course provides practical knowledge in designing the digital logic systems and their verification for the desired output.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Implementation of Boolean expression, Universal gates, Arithmetic block, code conversion using basic/ universal gates and ICs.

CO2. Design and implement combinational/ sequential circuits and computation of delays.

CO3. Implement and analyze arithmetic/sequential logic circuit using simulation tool.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	L							
CO2	H	H	H	H	H	M						
CO3		H	H	H	H	H						

Syllabus

List of Experiments

1) Simplification, realization of boolean expressions using logic gates/universal gates
2) Realization of half/full adder & half/full subtractors using logic gates
3) Realization of parallel adder/subtractors using 7483 chip, BCD to Excess-3code conversion & vice versa, binary to gray code conversion & vice versa
4) MUX/DEMUX – use of 74153,74139 for arithmetic circuits & code converter
5) Realization of one/two bit comparator and study of 7485 magnitude comparator
6) Use of a) Decoder chip to drive LED display & b) Priority encoder
7) Truth table verification of flip-flops: i) JK Master Slave ii) T type iii) D type
8) Realization of 3 bit counters as a sequential circuit & MOD-N counter design (7476,7490,74192,74193)
9) Writing & testing of sequence generator
10) Hardware modeling using Verilog/VHDL; Laboratory exercises and assignments to supplement the lab

Reference Books/Material

1. Morris Mano, “Digital Design”, Pearson Education India,5th Edition, 2013
2. Charles. H. Roth, Jr., “Fundamentals of Logic Design”, Fifth Edition, Thomson Brooks /Cole, 2005.
3. R. J. Tocci, “Digital Systems Principles and Applications”, Prentice Hall
4. D. D. Givone, “Digital Principles and Design”, Tata McGraw Hill.

5. S. Palnitkar, “Verilog HDL: A Guide to Digital Design and Synthesis”, Second Edition, Pearson Education, 2004.
6. J. Bhasker, “A VHDL primer”, 3rd edition, Addison Wesley Longmen, 1999.
7. Douglas Perry, “VHDL: Programming by example”, 4 th ed. McGraw Hill International, 2002.
8. Peter Ashenden, “The Designer Guide to VHDL”,Morgan Kaufmann, 1998

Course Code	Course Name	L	T	P	Credits
CS206	Object Oriented Programming Lab	0	0	3	2

Course Objective

The objective of the course is to introduce students to fundamental programming concepts and methodologies which are essential to build good C++/Java programs.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Demonstrate fundamental programming methodologies in the C++/Java programming language via laboratory experiences.

CO2. Demonstrate the ability to build programs for complex problems.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H							
CO2	H	H	H	H	H							

Syllabus

List of experiments

1) Simple programs in C++
2) Matrix multiplication in C++

3) Operator overloading exercises

4) Matrix manipulation using dynamic memory allocation

5) Overloading dynamic memory allocation operators

6) Practice on templates

7) Implementation of linked list using templates

8) Implementation of sorting algorithms using templates

9) Implementation of stack and queue using exception handling

10) Inheritance based exercise

11) File handling using streams

12) Practice of Java programming

13) File handling using Java streams

14) Multithreaded programming using Java

15) Developing graphical user interfaces using Java

Reference Books/Material

1. BJarne Stroustrup, "The C++ Programming Language", 4th Edition, Addison Wesley, 2013.
2. Stanley B Lippman, "The C++ Primer", 5th Edition, Addison Wesley, 2012.
3. Ira Pohl, "Object Oriented Programming using C++", 2nd Edition, Pearson, 2003.
4. Patrick Naughton and Herbert Schildt, "Java 2: The Complete Reference", 5th Edition, McGraw Hill, 2017.
5. Paul. Deitel, Harvey Deitel, "Java: How to program", 8th Edition, Pearson, 2017.

4th Semester

Course Code	Course Name	L	T	P	Credits
IE250	Innovation & Entrepreneurship	1	0	0	1

Course Objective

- Introducing a project-based learning approach from Ideation to Innovation and Entrepreneurship will be the key process considered here.
- To learn the essential concepts of innovation and entrepreneurship through hands-on activities and the best and most relevant practical examples
- The course is designed to provide the tools necessary for starting independent innovation and businesses
- To give students practical experience in market survey, commercialization, IPR and proactively work in projects in risky market environments

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability.

CO1. To comprehend the basic theories and concepts that underlie a survey study of Innovation, Entrepreneurship and Social Business/ Entrepreneurship

CO2. To understand how to generate good large company or startup business ideas / societal ideas, and refine these ideas, to substantially increase chances for success in the marketplace

CO3. The students will be exposed to the thoughts and strategies of some very effective real-life innovators and entrepreneurs through videos and small cases.

CO4. To understand about IPR, prototyping and financial management.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	H	L	H	M	H	H	L	H	H	H	H
CO2	L	H	L	H	H	H	H	L	H	H	H	H
CO3	L	H	L	H	H	H	H	L	M	H	H	H
CO4	L	H	L	H	H	H	H	H	L	L	H	H

Syllabus

Module 1: Introduction:

Creative thinking, blocks to creativity, factors that influence creative design, engineering design and creative design, influence of society, market pull & technology push, attribute of a creative person Three levels of Design – Viceral, Behavioral and Reflective design.

Qualities and skills required for entrepreneurship, Functions of an entrepreneur, Importance of entrepreneur in economic development.

Module 2: Ideas for Entrepreneurship:

Need or identification of a problem, market survey, data collection, review & analysis, problem definition, challenge statement, problem statement initial specifications, Brain storming, analogy technique or Synectic, check list, trigger words, morphological method, interaction matrix method, analysis of interconnected decision making.

Module 3: Theory of Inventive Problem Solving (TRIZ):

20 key TRIZ principles – multifunction, compensation, nested doll, blessing in disguise, segmentation, separation, symmetry change, opaque & porous, inflate and deflate, recycle & recover, phase transformation, energy, imaging, environment, composition, economical, surface response, static & dynamic, continuous & intermittent, dimensions.

Module 4: Product Design, IPR & Finance:

Detail design, prototyping, product deployment, useful life assessment and recycling and sustainability; patent act, patent laws, Types of entrepreneurs- Based on type of business, based on use of technology, based on motivation, based on stages of development, based on motive, Based on capital ownership, Business Plan, Finance and Funding.

Reference Books/Material

1. C.B.Gupta & N.P.Srinivasan, 'Entrepreneurial Development', Sultan Chand & Sons, 2020, ISBN: 978-93-5161-132-5
2. Floyd Hurt, Rousing Creativity: Think New Now, Crisp Publ Inc. 1999, ISBN 1560525479,

3. Kalevi Rantanen & Ellen Domb, 'Simplified TRIZ' – II edn., Auerbach Publications, Taylor & Francis Group, 2010, ISBN: 978-142-0062-748
4. John Adair, 'The Art of Creative Thinking', Kogan Page Publication, 2011, ISBN 978-0-7494-5483-8

Course Code	Course Name	L	T	P	Credits
CS250	Database Systems	3	1	0	4

Course Objective

This course covers the relational database systems RDBS - the predominant system for business, scientific and engineering applications at present. The topics are reinforced using tools such as Oracle server in labs. The course includes entity-relation model, normalization, relational model, relational algebra, and data access queries as well as an introduction to SQL.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Model Entity-Relationship diagrams for enterprise level databases

CO2. Formulate Queries using SQL and Relational Formal Query Languages

CO3. Apply different normal forms to design the Database

CO4. Uses formal design techniques to produce a database schema

CO5. Summarize concurrency control protocols and recovery algorithms

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	M	M	L	L	H	M	M	M
CO2	H	H	H	H	H	M	L	L	H	M	H	M
CO3	H	H	H	M	M	M	L	L	M	M	M	M
CO4	H	H	H	M	M	M	L	L	M	M	M	M
CO5	H	H	M	M	M	M	L	L	M	L	M	M

Syllabus

Module 1 : Introduction: An overview of database management system, database system vs file system, database system concept and architecture, data model schema and instances, data independence and database language and interfaces, (DDL,DML,DCL), overall database structure, database users. Data modelling using the Entity Relationship model: ER model concepts, notation for ER diagram, mapping constraints, keys, specialization, generalization, aggregation, reduction of an ER diagrams to tables, extended ER model, relationship of higher degree.

Module 2 : Relational data Model and Language: Relational data model concepts, integrity constraints, entity integrity, referential integrity, key constraints, domain constraints, relational algebra, relational calculus, tuple and domain calculus.Introduction on SQL: Characteristics of SQL, advantage of SQL, SQL data type and literals, types of SQL commands, SQL operators and their procedure, tables, views and indexes, queries and sub queries, aggregate functions, insert, update and delete operations, joins, unions, intersection, minus, cursors, triggers, procedures in SQL/PL SQL.

Module 3 : Data Base Design & Normalization: Functional dependencies, primary key, foreign key, candidate key, super key, normal forms, first, second, third normal forms, BCNF, 4th Normal form,5th normal form, loss less join decompositions, canonical cover, redundant cover, synthesis the set of relation , MVD, and JDs,inclusion dependence, transaction processing concept, transaction system, testing of serializability, serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log based recovery, deadlock handling.

Module 4 : Concurrency Control Techniques: Concurrency control, locking techniques for concurrency control, 2PL, time stamping protocols for concurrency control, validation based protocol, multiple granularity, multi version schemes and recovery with concurrent transaction. Storage: Introduction, secondary storage devices, tertiary storage, buffering of blocks, structure of files, file organization, indexing and hashing, types of single level ordered indexes, multilevel indexes, dynamics multilevel indexes using B-trees and B+-Trees, database security.

Reference Books/Material

1. Korth, Silberschatz, "Database System Concepts", 4th ed., TMH, 2003.
2. Elmsari and Navathe, "Fundamentals of Database Systems", 4th ed., A. Wesley, 2004
3. Raghu Ramakrishnan , Johannes Gehrke, " Database Management Systems", 3rd Edition, McGraw- Hill, 2003.
4. J D Ullman, "Principles of database systems", Computer Science Press, 2001.

Course Code	Course Name	L	T	P	Credits
CS251	Software Engineering	3	0	0	3

Course Objective

To introduce the software development life cycles models, analyse the software requirements, introduce various design methods for software development and to develop an ability and skill to test software systems.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the basic concepts of software engineering, software project management, software designs and software testing.

CO2. Choose the appropriate software development models and methodologies and plan the design and testing strategies to manage given software applications.

CO3. Analyze the role, advantages and disadvantages of various software development models, methodologies, design strategies, testing strategies and software project management methods.

CO4. Build a software application for a given real life problem incorporating all the phases of the software development life cycle.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	L	M	L	L	M	H	H	H	H
CO2	H	H	H	M	H	M	L	M	H	H	H	H
CO3	H	H	H	H	H	H	M	M	H	H	H	H
CO4	H	H	H	H	H	H	M	H	H	H	H	H

Syllabus

Module 1 : Introduction, Software development life cycle: software requirements specification, formal requirements specification and verification - axiomatic and algebraic specifications, function-oriented software design. Software life-cycle models and their comparison.

Module 2 : Software Requirement and Analysis: Data dictionary, Data flow diagrams. IEEE standards for software requirements.

Techniques: feasibility analysis, requirements elicitation, validation, rapid prototyping, OO paradigms vs. structured paradigm - OO analysis.

Effort estimation and scheduling: LOC, Function point analysis and Basic COCOMO model. Basic design concepts: Cohesion and its various types, Coupling and its various types.

Module 3 : Software Specifications :Specification document, specification qualities, uses, system modelling: context, interaction, structural, behavioural, DFD, specification techniques using UML, ER diagrams, logic, algebraic specifications: comparison of various techniques, formal specifications – model checking, introduction to binary decision diagrams.

Module 4 : Object Oriented Methodology :Introduction to objects, relationships, unified approach to modelling, use-case modelling, activity, state and interaction diagrams, classification approaches, cohesion, coupling, reuse, case studies - object oriented paradigm, software design: architectural - distributed - data oriented design & object-oriented design - real-time systems design techniques. Object modeling using UML: UML overview, nature and purpose of models. Use case diagrams, class diagrams, activity diagram, sequence diagram, interaction diagram.

Sample Tool- Argo UML, an open-source tool.

Module 5: Taxonomy of software testing, types of S/W test, black box testing, testing boundary conditions, structural testing, test coverage criteria based on data flow mechanisms, regression testing, unit testing, integration testing, validation testing, system testing and debugging. Testing: Software quality Assurance, Walkthroughs, Inspections, Attributes to be tested,

Software reliability and fault-tolerance, software project planning, monitoring, and control, software maintenance, computer-aided software engineering (CASE), software reuse, component-based software development, extreme programming.

Reference Books/Material

1. Roger S Pressman: "Software Engineering – A Practitioner's Approach", 7th Edition, McGraw-Hill, 2009.
2. Rajib Mall, "Fundamentals of Software Engineering", 5th Edition, PHI, 2018.
3. Ian Sommerville: "Software Engineering". 9th Edition, Pearson Education, 2011.
4. S.L. Pfleeger, Software Engineering – Theory and Practice, 2nd Edition, Pearson Education, 2015
5. Paul Ammann, and Jeff Offutt, "Introduction to Software Testing", 1st Edition, Cambridge University Press, 2008.
6. Eric Gamma, "Design Patterns: Elements of Reusable Object-Oriented Software", 1st Edition, Addison-Wesley Longman Publishing, 1995
7. K. C. Shet, "Software Engineering & Quality Assurance", BPB Publications, New Delhi
8. Waman S. Jawadekar, "Software Engineering, Principles and Practice", Tata McGraw Hill

Course Code	Course Name	L	T	P	Credits
CS252	Computer Organization and Architecture	3	1	0	4

Course Objective

To understand the basic hardware and software issues of computer organization to provide an overview on the design principles of digital computing systems. The course also focusses on understanding how the data is represented at machine level and computations are performed at machine level.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability

CO1. To understand how a program gets executed at machine level.

CO2. To understand the basic hardware aspects of a computer system.

CO3. To build various functional units of computer system

CO4. To workout tradeoff involved in designing a modern computer system

CO5. To identify problems in various components of computer systems and comprehend and differentiate various computer architectures and hardware.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	M	L		L						M	
CO2	H	M	H									
CO3	H		H	L								
CO4	H	H	H	L								
CO5	H	H	M	L	M							

Syllabus

Module 1 : Overview of Computer Architecture & Organization, contrast between computer architecture & organization, logical organization of computers; basic operational concepts, bus structures, performance, processor clock, basic performance equation, clock rate, performance measurement, Von Neumann machine, instruction format, execution cycle; instruction types and addressing modes.

Module 2 : Computer Arithmetic: representation of integers and real numbers, fixed point arithmetic, arithmetic and logical unit design, addition and subtraction of signed numbers, design of fast adders, multiplication of positive numbers, signed operand multiplication, fast multiplication, integer division, floating-point numbers and operations.

Module 3 : Basic Concepts of Memory System: Semiconductor RAM memories, ROM memories, speed, size, and cost, cache memories mapping functions, replacement algorithms, performance considerations, virtual memories, secondary storage.

Module 4 : Control Unit Design: Instruction sequencing, instruction interpretation, control memory, hardwired control, microprogrammed control and microprogrammed computers. I/O organization, bus control, Serial I/O (study of asynchronous and synchronous modes, USART & VART), parallel data transfer Program controlled: asynchronous, synchronous & interrupt driven modes, DMA mode, interrupt controller and DMA controller.

Module 5: Organization of CPU: Single vs. multiple data path, ISA, control unit, instruction pipelining, trends in computer architecture, CISC, RISC, VLIW, introduction to ILP, pipeline hazards: structural, data and control, reducing the effects of hazards.

Reference Books/Material

1. Carl Hamacher, ZvonkoVranesic and SafwatZaky, "Computer organization", 5th Edition, Tata McGraw Hill, 2002.
2. Patterson and Hennessy, "Computer Organization and Design", 5th Edition, Morgan Kaufman, 2013
3. Patterson and Hennessy, "Computer architecture: A quantitative approach", Morgan Kaufmann, 2000.
4. J. P. Hayes, "Computer architecture and organization", 3rd Edition, McGraw Hill, 1998.
5. Hwang and Briggs, "Computer architecture and parallel processing", McGraw Hill, 1985.

Course Code	Course Name	L	T	P	Credits
CS253	Theory of Computation	3	1	0	4

Course Objective

This course introduces models of computation: Regular languages models, context-free languages models. Also, this course emphasizes computability and computational complexity theory. Topics include decidable and undecidable problems, reducibility,

recursive function theory, completeness, hierarchy theorems, inherently complex problems, and interactive proof systems.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Describe concept of theorem proofing, finite automata, different type of automata, etc.

CO2. Be familiar with Regular and Non regular Language, i.e., context-free language. Understand the concept of Context Free Grammars and Pushdown Automata. Also, Understanding the Context free languages and grammars, and also Normalizing CFG.

CO3. Examine the properties of formal language and automata, their equivalence and conversion techniques.

CO4. Understand basic properties of Turing machines and computing with Turing machines

CO5. Know the concepts of tractability and decidability, the concepts of NP-completeness and NP-hard problem.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	H	H	M	H	H	L	M	M	H	H	M	H
CO2	H	H	H	H	H	M	L	M	H	H	M	H
CO3	H	M	M	H	H	L	M	M	H	H	H	H
CO4	H	M	H	M	H	M	L	M	H	H	H	H
CO5	H	H	H	M	H	L	L	M	H	M	L	L

Syllabus

Introduction to Automata: Introduction to formal proof, additional forms of proof, inductive proofs, finite automata (FA), deterministic finite automata (DFA), non-deterministic finite automata (NFA), Finite Automata with Epsilon transitions.

Regular expression and languages: Introduction to regular expression, building regular expression, converting DFA to a regular expression, converting regular expression to DFA,

pumping lemma and its applications to prove languages not to be regular, closure properties of regular languages, minimization of automata.

Context free grammars (CFG) and languages: Definition, derivations, parse trees, ambiguity in grammars and languages, pushdown automata (PDA): Definition, Graphical notation, deterministic and nondeterministic, instantaneous descriptions of PDAs, language acceptance by final states and by empty stack, equivalence of the CFG and PDAs, pumping lemma for CFLs, closure properties of CFLs, decision problems for CFLs.

Computability Theory: Introduction to Turing machines, language acceptance by Turing machines, Turing machine transition diagrams, Church-Turing hypothesis, Chomsky hierarchy, recursively enumerable sets, existence of non-recursively enumerable notion of undecidable problems, universality of Turing machine, separation of recursive and recursively enumerable classes, notion of reduction, undecidable problems of Turing machines.

Complexity Theory: Notion of tractability/feasibility, the classes NP and co-NP, polynomial time many-one reduction, completeness under this reduction, NP-completeness of propositional satisfiability, other variants of satisfiability, NP-complete problems from other domains: graphs (clique, vertex cover, independent sets, Hamiltonian cycle), number problem (partition), set cover.

Reference Books/Material

1. J.E. Hopcroft and J.D. Ullman. “Introduction to Automata Theory, Languages of Computations”, Addison-Wesley, 2008.
2. Michael Sipser, “Introduction to the Theory of Computation”, Books/Cole Thomson Learning, 2014.
3. H. R. Lewis and C. Papadimitriou, “Elements of Theory of Computation”, Prentice-Hall, 2015.
4. D. C. Kozen, Automata and Computability, Addison Wesley, 1994.
5. J. C. Martin, Introduction to Languages and the Theory of Computation, McGraw Hill, 2002

Course Code	Course Name	L	T	P	Credits
CS254	Microprocessor and Microcontrollers	3	1	0	4

Course Objective

To introduce microprocessor architecture, interfacing and programming with 80x86 microprocessors. To familiarize the architecture of Microcontrollers using 8051 microcontrollers and also for high end processors as: ARM/PIC.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the organization and design of microcomputer and programming.

CO2. Understand the design of architecture of 80x86 family, its interfacing with peripherals and programming

CO3. Understand the design of architecture of 8051 microcontroller, its interfacing with peripherals and programming in assembly and C language

CO4. Understand the architecture and programming with high end processors

CO5. Design and Develop interfacing models according to applications.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	M	H	M	L	M	L		M		H
CO2	H	H	H	H	H	L	L	L		M	L	H
CO3	H	H	H	H	H	M	L	L	L	M	L	H
CO4	H	H	M	H	M	L	M	M	L	L	L	M
CO5	H	H	H	H	H	L	L	L		M	L	H

Syllabus

Module 1: Organization of microcomputer: Von Neumann and Harvard architecture, Instruction set architectures, data transfer operations and their hardware implementation,

addressing schemes, instruction set design, general purpose register organization, basic operational concepts of CPU and GPU, multiprocessors, multicore processors, fixed point and floating point arithmetic, overview on assembly language

Module 2: Arithmetical algorithms, pipelining, 80x86 Architecture, programming models, segmentation, addressing modes of 80x86, instructions sets of 80x86, I/O addressing in 80x86, programming with 80x86, interfacing with 80x86: interfacing with RAMs, ROMs, peripheral ICs and key-boards, use of 80x86 in electrical engineering

Module 3: Intel 8051 Microcontroller-Architecture, Assembly language of 8051, programmable keyboard/display interface, interface programmable peripheral interface, programmable communication interface, serial and parallel bus (RS232,IEEE488), use of 8051 in electrical engineering, overview of architecture of 8096 16-bit microcontroller.

Module 4 : Introduction to High end Processors: ARMx processors, ARMx Architecture, registers and internal modules, JTAG, GPIO, ADC in ARMx processors, overview on Raspberry Pi/PIC microcontroller.

Reference Books/Material

1. M. Morris Mano, "Computer System Architecture", Pearson, 3 rd edition, 1992
2. Barry B. Brey, "The Intel Microprocessors: Architecture, Programming and Interface", PHI, 1997.
3. Sivarama P. Dandamudi, "Introduction to Assembly Language Programming: From 8086 to Pentium Processors", Springer, 1998.
4. . Muhammad Ali Mazidi, Janice G. Mazidi, Rolin D Mckinlay," 8051 Microcontroller and Embedded systems", Pearson Education, 2005.
5. Krishna Kant, "Microprocessors and Microcontrollers, Architecture, Programming, and System Design-8085,8086,8051,8096",PHI Publication,
6. Andrew N Sloss, Dominic Symes and Chris Wright, ARM System developers guide-Designing and developing system software, Elsevier 2004

Course Code	Course Name	L	T	P	Credits
CS255	Database Systems Lab	0	0	3	2

Course Objective

To obtain working knowledge of a database management system and developing applications using the databases**.**

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Design and implement a database schema for given problem

CO2. Apply the normalization techniques for development of application software to realistic Problems

CO3. Formulate queries using SQL DML/DDL/DCL commands.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	M	M	L	L	H	M	M	M
CO2	H	H	H	M	M	M	L	L	M	M	M	M
CO3	H	H	H	H	H	H	M	L	H	M	M	M

Syllabus

List of Experiments

1) Defining schemas for applications.
2) Creating tables, Renaming tables, Data constraints (Primary key, Foreign key, Not Null), Data insertion into a table.
3) Grouping data, aggregate functions, Oracle functions (mathematical, character functions).
4) Sub-queries, Set operations, Joins.
5) Creation of databases, writing SQL and PL/SQL queries to retrieve information from the databases.
6) Procedures, Functions, Cursors, Triggers, Packages, views, Assertions.

7) Deployment of Forms, Reports Normalization, Query Processing Algorithms in the above application project;
8) Assignment in Design and Implementation of Database systems or packages for applications such as office automation, hotel management, hospital management;

Reference Books/Material

1. Ramez Elmasri, Shamkant B Navathe, "Fundamentals of database systems", 5th ed., 2003.
2. Avi Silberschatz, Henry Korth and S. Sudarshan, "Database Systems Concepts", 5th Edition, TMH, 2005

Course Code	Course Name	L	T	P	Credits
CS256	Microprocessor and Microcontrollers Lab	0	0	3	2

Course Objective

To provide hands on experience on programming and interfacing micro controllers

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability

CO1. To write, debug and execute assembly level program for 8051.

CO2. To obtain practical experience on interfacing I/O devices with 8051.

CO3. To write, debug and execute problems using Embedded C programming.

CO4. To perform mini projects on a real-time problem

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	M	L		L							L
CO2	H	H	M		L							L

CO3	H	H	H		L							L
CO4	M	L	H		L					M		L

Syllabus

List of Experiments:

1) Assembly Level programs for arithmetic (addition, subtraction, multiplication) and logic operations
2) Assembly Level Programs for sorting of numbers
3) Embedded C programs for arithmetic and logic operations
4) Embedded C program for sorting and searching of numbers
5) I/O interface experiments using Assembly Level Program and Embedded C program, Interfacing with A/D and D/A converters
6) Interfacing with stepper motors
7) LCD interfacing to 8051
8) Design and execute programs to generate square wave / Sine Wave/ Pulses/ triangular wave
9) Design and execute a I/O interface problem using Keyboard/push button as input and LED as output device.
10) Mini project

Reference Books/Material

1. M. Morris Mano, "Computer System Architecture", Pearson, 3 rd edition, 1992
2. Barry B. Brey, "The Intel Microprocessors: Architecture, Programming and Interface", PHI, 1997.
3. Sivarama P. Dandamudi, "Introduction to Assembly Language Programming: From 8086 to Pentium Processors", Springer, 1998.
4. . Muhammad Ali Mazidi, Janice G. Mazidi, Rolin D Mckinlay," 8051 Microcontroller and Embedded systems", Pearson Education, 2005.
5. Krishna Kant, "Microprocessors and Microcontrollers, Architecture, Programming, and System Design-8085,8086,8051,8096",PHI Publication,
6. Andrew N Sloss, Dominic Symes and Chris Wright, ARM System developers guide-Designing and developing system software, Elsevier 2004

5th Semester

Course Code	Course Name	L	T	P	Credits
CS300	Operating Systems	3	1	0	4

Course Objective

The objectives of this course are to provide knowledge about the services rendered by operating systems, to provide a detailed discussion of various process management techniques and memory management techniques, to discuss the various file-system design and implementation issues, to discuss the I/O management functionalities and to discuss how the protection domains help to achieve security in a system.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Comprehend the techniques used to implement the process manager

CO2. Comprehend virtual memory abstractions in operating systems

CO3. Design and develop file system interfaces, etc.

CO4. Design access control and protection-based modules for an operating system

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	L			L	L	M		H
CO2	H	H	H	M				L	L	M		H
CO3	H	H	H	M	L			L	L	M		H
CO4	H	H	H	H	L			L	L	M		H

Syllabus

Module 1: Introduction to operating systems: review of computer organization, operating system structures, system calls, system programs, virtual machine.

Module 2: Process and Threads: Process Concept, Process States, Process Description, Process Control Block, PCB as a Data Structure in Contemporary Operating Systems, Process Hierarchy, Process scheduling, Operations on processes, Cooperating processes, Inter-Process-Communication (IPC), Communication in client-server systems, Processes vs Threads, Types of Threads, Multicore and Multithreading, Case Study: Linux & Windows Process and Thread Management and its Related System Calls. CPU Scheduling: Scheduling criteria, Scheduling algorithms, Multiple processor scheduling, Real time scheduling, Algorithm evaluation. Process synchronization: The critical section problem, Synchronization hardware, Semaphores, Classical problems of synchronization. Deadlock: Methods for handling deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

Module 3: Memory Management: Background, Swapping, Contiguous memory allocation, Paging, Segmentation, Segmentation with paging, Virtual memory - Background, Demand Paging, Process Creation, Page replacement, Allocation of frames, Thrashing File System Interface: File Concept, Access methods, Directory Structure, Allocation methods, Free space management, Disk Scheduling, Disk Management, Swap space management, RAID.

Module 4: I/O Management: I/O Device, Organization of the I/O Function, Operating System Design Issue, I/O Buffering, Disk Scheduling, RAID, Disk Cache, Case Study: Linux & Windows I/O.

Module 5: Operating system security & protection, breaches, solutions, mechanisms, Inside attacks, outside attacks, case studies - the UNIX kernel and Microsoft Windows NT.

Reference Books/Material

1. Silberschatz, Galvin and Gagne, "Operating System Concepts", 10/E, John Wiley & Sons, 2018.
2. William Stallings, “Operating Systems –Internals and Design Principles”, 8/E, Pearson Publications, 2014.
3. Andrew S. Tanenbaum, “Modern Operating Systems”, 4/E, Pearson Publications, 2014.
4. Remzi Arpaci-Dusseau, Andrea Arpaci-Dusseau, “Operating Systems: Three Easy Pieces”, Arpaci-Dusseau Books.

CS301	Introduction to Machine Learning	3	0	0	3

Course Objective

Introduce the fundamental concepts of machine learning from a mathematically motivated perspective. The course will cover the important learning paradigms with popular algorithms.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand and appreciate the underlying mathematical relationships within and across machine learning algorithms and the paradigms of supervised and unsupervised learning

CO2. Appreciate machine learning challenges and suggest solutions for the same

CO3. Design and implement various machine learning algorithms in a range of real-world applications

CO4. Suggest supervised / unsupervised machine learning approaches for any application

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	M	M	H							
CO2	H	H	H	M	M							
CO3	H	H	H	H	H							
CO4	M	H	H	M	H							

Syllabus

Module 1: Introduction to Machine learning, Learning paradigms, Pattern analysis tasks: Classification, Regression and Clustering Illustrative applications, Designing a learning system, Mathematical concepts required for machine learning

Module 2: Statistical method for classification based on Bayes decision theory

Discriminant functions and decision surfaces, The Gaussian probability density function, Parameter estimation-Maximum Likelihood method, Mixture models—mixture of Gaussians-GMM, Expectation Maximization method

Module 3: Linear model for regression and classification

Linear basis function models for regression-linear regression and polynomial regression, Model complexity discussion, Linear discriminant functions and hyperplanes (Least squares method, Fisher linear discriminant, perceptron, support vector machines), probabilistic generative models, probabilistic discriminative models (logistic regression).

Module 4: Nonlinear models for classification and Regression

Probabilistic generative models, Artificial Neural networks- multi-layer perceptron/ multi-layer feed forward neural networks, Back-propagation algorithm, Radial basis function neural network, probabilistic neural network, extreme learning machine, Bias Variance dilemma/ decomposition, Evaluation of a hypothesis, Classifier combination: Ensemble methods, Random forests

Module 5: Unsupervised Learning

Data clustering: Partition based approach (k-means, k-medoid etc), hierarchical method for clustering, soft clustering approaches (Fuzzy K-means, Gaussian mixture model), Density based approaches, Graph based approaches to clustering, Cluster purity measures, Dimension Reduction and feature selection. Anomaly/outlier detection

Reference Books/Material

1. C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
2. Shai Shalev-Shwartz and Shai Ben-David, “Understanding Machine Learning: Form Theory to Algorithms”, Cambridge University Press, 2014

3. Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar, "Foundations of Machine Learning" Second Edition, The MIT Press

4. NPTEL course offered by Prof. Balaraman Ravindran, IIT Madaras titled "Introduction to Machine Learning".

5. S. Haykin, Neural Networks and Learning Machines, Prentice Hall of India, 2010

6. S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, 2009.

Course Code	Course Name	L	T	P	Credits
CS302	Design and Analysis of Algorithms	3	0	0	3

Course Objective:

The primary objective of this course is to introduce the concept of algorithm as a precise mathematical concept, and study how to design algorithms, establish their correctness, study their efficiency and memory needs. The course consists of a strong mathematical component in addition to the design of various algorithms.

Course Outcomes:

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Identify various Time and Space complexities of various algorithms

CO2. Understand Tree Traversal method and Greedy Algorithms

CO3. Apply Dynamic Programming concept to solve various problems

CO4. Apply Backtracking, Branch and Bound concept to solve various problems

CO5. Implement different performance analysis methods for non-deterministic algorithms

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	H	H	H	H	H							M	H
CO2	H	H	H	H	H							M	H
CO3	H	H	H	H	H							M	H
CO4	H	H	H	H	H							M	H
CO5	H	H	H	H	H							M	H

Syllabus

Module 1:

Introduction: Algorithm, pseudo code for expressing algorithms, performance analysis-space complexity, time complexity, asymptotic notation-big (O) notation, omega notation, theta notation and little (o) notation, recurrences, probabilistic analysis, disjoint set operations, union and find algorithms.

Module 2:

Divide And Conquer: General method, applications-analysis of binary search, quick sort, merge sort, AND OR Graphs.

Greedy Method: General method, Applications-job sequencing with deadlines, Fractional knapsack problem, minimum cost spanning trees, Single source shortest path problem.

Module 3:

Graphs (Algorithm and Analysis): Breadth first search and traversal, Depth first search and traversal, Spanning trees, connected components and bi-connected components, Articulation points.

Dynamic Programming: General method, applications, Matrix chain multiplication, optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

Module 4:

Backtracking: General method, Applications- n-queen problem, Sum of subsets problem, Graph coloring and Hamiltonian cycles.

Branch And Bound: General method, applications travelling sales person problem, 0/1 knapsack problem- LC branch and bound solution, FIFO branch and bound solution.

Module 5:

Complexity classes, P, NP, Co-NP, NP Hard & NP complete problems. Search/decision, SAT, Cooks theorem, NP Completeness for clique, vertex cover, TSP, set covering & subset sum, approximation algorithms.

Reference Books/Material

1. Corman T. H., Leiserson C. E. and Rivest R. L., & Stein, C, “Introduction to Algorithms", Fourth Edition, Prentice Hall India, 2010
2. Kleinberg, J., & Tardos, E., “Algorithm design”, Pearson Education India. 2006
3. Sanjoy Dasgupta, Christos H. Papadimitriou, and Umesh Vazirani. “Algorithms,” McGraw-Hill Science/Engineering/Math, 2006.
4. Aho, V Hopcroft, John E. Hopcroft and Jefrey D. Ullman "The Design and Analysis of Computer Algorithms", 1st Edition, Pearson Education.
5. Horowitz and Sahni, "Fundamentals of Computer Algorithms", Galgotia Publications, 2000.
6. Baase S., "Computer Algorithms: Introduction to Design and Analysis", Addison Wesley. 2000
7. Donald E. Knuth, "Art of Computer Programming, Volume 1: Fundamental Algorithms", 3rd Edition, Addison Wesley, 2000
8. Anany Levtin, "Introduction to Design and Analysis of Algorithms", Pearson, 2003

Course Code	Course Name	L	T	P	Credits
CS303	Number Theory and Cryptography	3	0	0	3

Course Objective

The objectives of this course are to gain knowledge about the mathematical foundation of the cryptographic algorithms, to understand number theory and algebra for the design of cryptographic algorithms, to get an insight into the working of different existing cryptographic algorithms and to learn how to use cryptographic algorithms in security.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the mathematical background and foundations of the cryptography

CO2. Understand concepts related to cryptography and cryptanalysis

CO3. Define the system to protect determine the security properties that are desired for this system identify the possible threats to these security properties, their likelihood of occurrence and consider possible mitigations against these threats

CO4. Describe and implement of some of the prominent techniques for symmetric key and public-key cryptosystems, digital signature schemes

CO5. Analyze and use methods for cryptography and reflect about limits and applicability of these methods

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	L	H		M	H	M							
CO2		H		H				M		M			
CO3	H	M	M		H				L				
CO4	M	H	L	H						M		H	`
CO5			M										L

Syllabus

Mathematical preliminaries: Number theory and algebra, finite fields- Introduction to Number theory, Modular arithmetic, prime number generation, GCD, Euclidean Algorithm, Extended Euclidean Algorithm, Fast Exponentiation, Chinese Remainder Theorem, Fermat's and Euler's T heorem, Cauchy 's theorem, Quadratic residues, Legendre symbol, Jacobi symbol, Review of abstract, Algebra – Study of Ring Zn, multiplicative group Zn* and finite field Zp, Primality Testing – Fermat test, Carmichael numbers, SolovayStrassen Test, Miller Rabin Test – analysis, Shannon's Theory

Information Security Attacks, Services and Mechanisms, CIA, Security Design Principles, Attack Surface and Attack Trees

Conventional Encryption, Classical Techniques- substitution and transposition ciphers, study of basic cryptanalysis possible on classical ciphers, Modern Techniques- block and stream ciphers and RC4

Principles of Public-Key Cryptosystems, Introduction to Public key infrastructure, Security standards such as X.509, RSA, Diffie-Hellman Key Exchange, Elgamal Cryptosystem, Elliptic Curve Cryptography, Digital signatures

Hash Functions and Data Integrity, Security of Hash Functions-The Random Oracle Model, Iterated Hash Functions- Merkel Damgard Construction, Secure Hash Algorithm (SHA), Message authentication requirements, message authentication codes (MAC) based on hash functions-HMAC and block ciphers-DAA and CMAC, Authenticated Encryption-CCM and GCM

Reference Books/Material

1. J. Katz and Y. Lindell, Introduction to Modern Cryptography, Third edition CRC press, 2020
2. Menezes Bernard, Network Security and Cryptography, 1st Edition, Cengage Learning India, 2010.
3. Neal Koblitz, “Number theory and cryptography”, Springer, 2007.
4. Hans Delfs, Helmut Knebl, “Introduction to Cryptography: Principles and Applications”, Springer.
5. Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone, “Handbook of Applied Cryptography”, CRC Press, 1996.
6. Stinson. D. Cryptography: Theory and Practice, 3rd edition, Chapman & Hall/CRC, 2012.
7. Behrouz A. Forouzan and D. Mukhopadhyay, Cryptography & Network Security, McGraw Hill, New Delhi. , 2nd Edition - 1st reprint 2010.

Course Code	Course Name	L	T	P	Credits
CS304	Operating Systems Lab	0	0	3	2

Course Objective

The objective of the course is to understand the implementation of an operating system. This is to develop sufficient skills in the learner to write a new operating system if needed for developing systems to solve problems.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand how to work in Linux/UNIX environment

CO2. Implement and analyze various CPU scheduling algorithms

CO3. Implement and analyze various page replacement algorithms

CO4. Implement and study various approaches for inter process communication

CO5. Analyze various disk scheduling algorithms by implementing them

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

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

Syllabus

List of Experiments

1) Linux based exercises to practice working in Linux environment
2) Implementation of various CPU scheduling algorithms (FCFS, SJF, Priority).
3) Implementation of various page replacement algorithms (FIFO, Optimal, LRU)\
4) Concurrent programming; use of threads and processes, system calls (fork and v-fork).
5) Implementation of Producer-Consumer problem, Bankers algorithm

6) To simulate concept of semaphores.
7) To simulate concept of inter process communication
8) Implementation of various memory allocation algorithms, (First fit, Best fit and Worst fit), Disk Scheduling algorithms (FCFS, SCAN, SSTF, C-SCAN)
9) Kernel reconfiguration, devicone drivers and systems administration of different operating systems.
10) Writing utilities and OS performance tuning.

Reference Books/Material

1. Silberschatz, Galvin and Gagne, "Operating System Concepts", 10/E, John Wiley & Sons, 2018.
2. William Stallings, "Operating Systems –Internals and Design Principles", 8/E, Pearson Publications, 2014
3. Andrew S. Tanenbaum, "Modern Operating Systems", 4/E, Pearson Publications, 2014.
4. MelinMilenkovic, "Operating Systems: Concepts and Design", McGraw Hill, New York, 2000..

Course Code	Course Name	L	T	P	Credits
CS305	Design and Analysis of Algorithms Laboratory.	0	0	3	2

Course Objective :

The primary objective of this course is to introduce the concept of algorithm as a precise mathematical concept, and study how to design algorithms, establish their correctness, study their efficiency and memory needs. The course consists of a strong mathematical component in addition to the design of various algorithms.

Course Outcomes :

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Calculate the time complexity of algorithm

CO2. Write programs for the problems using Divide and Conquer.

CO3. Write programs for the problems using Greedy Method

CO4. Write programs for the problems using Dynamic programming

CO5. Write programs for the problems using Backtracking

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	H	H	H	H	H							M	H
CO2	H	H	H	H	H							M	H
CO3	H	H	H	H	H							M	H
CO4	H	H	H	H	H							M	H
CO5	H	H	H	H	H							M	H

Syllabus

List of Experiments

1) Write a program to perform operation count for a given pseudo code
2) Write a program to perform Quick Sort for the given list of integer values.
3) Write a Program to perform Merge Sort on the given two lists of integer values.
4) Write a program to find solution for knapsack problem using greedy method.
5) Write a program to find minimum cost spanning tree using Prim's Algorithm.
6) Write a program to find minimum cost spanning tree using Kruskal's Algorithm.
7) Write a program to perform Single source shortest path problem for a given graph,
8) Write a program to find solution for job sequencing with deadlines problem.
9) Write a program for all pairs shortest path problem.
10) Write a program to solve N-QUEENS problem.
11) Write a program to solve Sum of subsets problem for a given set of distinct numbers.
12) Write a program to solve travelling sales person problem using branch and bound.

Reference Books/Material

1. Data Structures and Algorithms by G.A.V. Pai, 2017, TMH.

2. Fundamentals of Computer Algorithms by Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, 2nd edition, University Press.

Course Code	Course Name	L	T	P	Credits
HU350	Professional Ethics and Human Values	1	0	0	1

Course Objectives:

The main objective is to inculcate human values and professional ethics among the students so that they become good human beings, which in turn will bring collective benefits. Also, the students will understand harmony at all levels of existence.

Course Outcomes

After this course, the student shall acquire knowledge of Human Values and ethics and there will be a behaviour change. They will understand the value of harmonious relationships with fellow human beings based on trust, respect, compassion, tolerance, and empathy.

.

CO1. Students will have a fair understanding of Human Values and Professional Ethics

CO2. Students will exemplify good behaviour

CO3. Students will develop a feeling of Empathy

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1						M		H				L
CO2						M		H				L
CO3						H	M					L

Syllabus

Module 1: Introduction to Concepts of Human Values and Ethics- Origin and History (Western and Eastern Perspectives with reference to Socrates, Plato, Plotinus, Epicurus,Thomas Aquinas, Immanuel Kant, Buddha, *The Vedas*, *The Upanishads* and *The Mahabharata*) Ethics – Classification (4 Types), History, and Purposes, Utilitarianism,

Duties, Rights, Responsibility, Virtue, Honesty, Morality, Moral Autonomy, Obligations of Engineering Profession and Moral Propriety.

Module 2: A comprehensive understanding of Existence, Knowledge of Self, Knowledge of Society, Nature vis-à-vis Culture, Anthropocentrism, Deep Ecology, Idea of Cosmos

Module 3: Ability to identify the scope and characteristics of people-friendly and eco-friendly
production systems, technologies, and management models, Engineer's Moral responsibility for Safety and Human Rights, Risk Assessment and Communication, Product Liability, Engineers-Employers Liaison, Whistle-Blowing and its Justification, Cyber Crime and Cyber Ethics, and Ecoethics

Module 4: Case study Discussion of typical holistic technologies, management models and production systems, Strategy for the transition from the present state to Universal Human Order

Module 5: Rapid Reading of texts like *Justice*, *Crime and Punishment*, *The Model Millionaire*, & Films Discussion like *An Inconvenient Truth*, *Modern Times*, and *The Elephant Whisperers* to understand Universal Human Values.

Reference Books/Material

1. A N Tripathy, 2003, *Human Values*, New Age International Publishers.
2. B P Banerjee, 2005, *Foundations of Ethics and Management*, Excel Books
3. B L Bajpai, 2004, *Indian Ethos and Modern Management*, New Royal Book Co., Lucknow. Reprinted 2008.
4. E.F. Schumacher, 1973, *Small is Beautiful: a study of economics as if people mattered*, Blond & Briggs, Britain
5. Ralph T.H. Griffith, (Trans) *The Vedas*
6. Eknath Easwaran, (Trans) *The Upanishads*
7. Peter Brook directed *The Mahabharata* (1989) film [available on Youtube]

6^{th} Semester

Course Code	Course Name	L	T	P	Credits
CS350	Distributed Systems	3	0	0	3

Course Objective

The objectives of this course are to have a broad and up-to-date coverage of the principles and practice in the area of Distributed Systems and understand the heterogeneous systems such as computers, mobile phones, other devices and Internet) and their functionalities.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the concepts of distributed System and design and implementation issues

CO2. Define key mechanism for designing distributed algorithms for different primitives like mutual exclusion, deadlock detection, agreement etc.

CO3. Analyze different types of faults and fault handling techniques in order to implement fault tolerant systems

CO4. Correlate different election algorithms, file system, time synchronization and naming services

CO5. Design and develop distributed programs subject for specific design and performance constraints

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	H		H	M	L								
CO2	H		H	M									
CO3	H		M		L								
CO4	H			H	L								
CO5	H		M	M									

Syllabus

Review of Networking Protocols, Point to Point Communication, Operating Systems, Concurrent Programming, Characteristics and Properties of Distributed Systems, Goals of Distributed Systems, Multiprocessor and Multicomputer Systems, Distributed Operating Systems, Network Operating Systems, Middleware Concept, The Client-Server Model, Design Approaches-Kernel Based-Virtual Machine Based, Application Layering

Layered Protocols, Message Passing-Remote Procedure Calls-Remote Object Invocation, Message Oriented Communication, Stream Oriented Communication, Case Studies
Concept of Threads, Process, Processor Allocation, Process Migration and Related Issues, Software Agents, Scheduling in Distributed System, Load Balancing and Sharing Approaches, Fault Tolerance, Real Time Distributed System

Clock Synchronization, Logical Clocks, Global State, Election Algorithms-The Bully algorithm-A Ring algorithm, Mutual Exclusion-A Centralized Algorithm-A Distributed Algorithm-A token ring Algorithm, Distributed Transactions

Introduction to Replication, Object Replication, Replication as Scaling Technique, Data Centric Consistency Models-Strict-Linearizability and Sequential-Causal-FIFO-Weak-release-Entry, Client Centric Consistency Models-Eventual Consistency-Monotonic Reads and Writes-Read your WritesWrites Follow Reads, Implementation Issues, Distribution Protocols-Replica Placement-Update Propogation-Epidemic Protocols, Consistency Protocols
Introduction, Failure Models, Failure Masking, Process Resilience, Agreem in Faulty Systems, Reliable Client Server communication, Group communication, Distributed Commit, Recovery

Introduction to Distributed Objects, Compile Time Vs Run Time Objects, Persistent and Transient Objects, Enterprise JAVA Beans, Stateful and Stateless Sessions, Global Distributed Shared Objects, Object Servers, Object Adaptors, Implementation of Object References, Static And Dynamic Remote Method Invocations, Replica Framework
Introduction, Architecture, Mechanisms for Building Distributed File Systems-Mounting-CachingHints-Bulk Data Transfer-Encryption, Design Issues-Naming and Name Resolution-Caches on Disk or Main Memory-Writing Policy-Cache consistency-Availability-Scalability-Semantics, Case Studies, Log Structured File Systems

Architecture, Processes, Communication, Naming, Synchronization, Web Proxy Caching, Replication of Web Hosting Systems, Replication of Web Applications

Reference Books/Material

1. Andrew S Tanenbaum, “Distributed systems: Principles and Paradigms”, Second Edition, Pearson Education. Inc 2007
2. George Coulouris, Jean Dollimore, and Tim Kindberg, “ Distributed Systems Concepts and Design”, 5th ed., Pearson Education, 2011
3. Mukesh Singhal and Niranjan G. Shivaratri, “Advanced Concepts in Operating Systems”, TMH, McGraw-Hill, Inc. New York, USA 1994
4. Pradeep K. Sinha, “Distributed Operating System: Concept and design”, PHI, New Delhi 2019
5. W Richard Stevens, “Unix Network Programming: Vol 1, Networking APIS: Sockets & XTI”, Second Edition E, Pearson Education, 1998
6. Colouris, Dollimore, Kindberg, "Distributed Systems Concepts & Design", Fourth Edition, Pearson Ed

Course Code	Course Name	L	T	P	Credits
CS351	Compiler Design (CD)	3	0	0	3

Course Objectives

Describe the steps and algorithms used by language translators.Recognize the underlying formal models such as finite state automata, push-down automata and their connection to language definition through regular expressions and grammars and discuss the effectiveness of optimization.

Course Outcomes:

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the concept of formulating token for various languages

CO2. Apply principles of parsing techniques to do syntax analysis

CO3. Formulate semantic rules for semantic analysis

CO4. Apply optimization techniques for intermediate code

CO5. Generate target code

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	M	L	L	M	M	L	L	M	M	M	M
CO2	L	H	M	M	M	M	L	L	M	M	M	M
CO3	M	M	M	M	M	M	L	L	M	M	M	M
CO4	L	L	M	M	M	M	L	L	M	M	M	M
CO5	L	L	L	M	M	M	L	L	M	M	M	M

Syllabus

Module 1

Introduction to compiler design, Model of a Compilers, Translators, Interpreters, Assemblers, Languages, Computer Architecture vs Compiler Design, Lexical analyzer, Regular expressions and finite automata.

Module2

Introduction to context free grammars, BNF notation, Syntax Analysis.

Module 3

Parsing Techniques: Top-down parsing and Bottom-up parsing, general parsing strategies, brute force approach, recursive descent parser and algorithms, simple LL(1) grammar, bottom-up parsing-handle of a right sentential form, shift reduce parsers, operator precedence parsers, LR, SLR, Canonical LR,LALR grammar and parsers, error recover strategies for different parsing techniques.

Module 4

Symbol table, syntax-directed translation schemes, intermediate code generation, translation schemes for programming language constructs, runtime storage allocation.

Module 5

Code generation and instruction selection: Issues, basic blocks and flow graphs, register allocation, DAG representation of programs, code generation from DAG, peep hole optimization, code generator generators, specifications of machine. Code optimization, source

of optimizations, optimization of basic blocks, loops, global dataflow analysis, solution to iterative dataflow equations.

Reference Books/Material

1. Alfred V. Aho, Ravi Sethi & Jeffrey D. Ullman, "Compilers; Principles, Techniques & Tools", Addison- Wesley Publication, 2001.
2. William A. Barrett et.al, "Compiler Construction, Theory and Practice", Galgotia 2000.
3. Andrew W. Appel, "Modern compiler implementation in c for compiler design", Cambridge University Press, Year 2004

Course Code	Course Name	L	T	P	Credits
CS352	Computer Networks	3	1	0	4

Course Objective

The objective of this course is to introduce concepts and implementation of computer networks; architecture, protocol layers, inter-networking and addressing; network application development, congestion control protocols, routing and switching, wireless networking. The course will also touch upon some recent advances in the field of computer networking.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Explain basic concepts, OSI reference model, services and role of each layer of OSI model and TCP/IP, networks devices and transmission media

CO2. Apply channel allocation, framing, error and flow control techniques. Channel access techniques in wired and wireless network

CO3. Describe the functions of Network Layer i.e., Logical addressing, subnetting & Routing Mechanism

CO4. Explain the different Transport Layer function i.e., Port addressing, Congestion control and Flow control mechanism

CO5. Explain the different protocols used at application layer i.e., HTTP, SNMP, SMTP, FTP, and VPN. Also, recent advance field of computer networking such as DCN, SDN

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							
CO5	H		M	L								

Syllabus

Introduction to Computer Network: Network Topologies, OSI reference model, services and role of each layer of OSI model and TCP/IP, Data Communication: Components of a Data Communication System, Simplex, Half-Duplex and Duplex Modes of Communication; Analog and Digital Signals; Noiseless and Noisy Channels; Bandwidth, Throughput and Latency; Digital and Analog Transmission; Data Encoding and Modulation Techniques; Broadband and Baseband Transmission; Multiplexing, Transmission Media, Transmission Errors, Error Handling Mechanisms.

Data Link Layer and Logical Link Control (LLC) sub-layer: Framing; Error control including Bit-parity, CRC and Hamming Codes; Reliable transmission and Automatic Repeat Request (ARQ) protocols including Stop-and-Wait, Go-back-N, Selective Repeat. Performance analysis of ARQ protocols. Example protocols such as HDLC and PPP. Medium Access Control (MAC) sub-layer: Shared media systems; Bus, Star and Ring topologies; TDMA, FDMA, CSMA, CSMA/CD, Ethernet and IEEE 802.3; IEEE 802.11 including CSMA/CA protocols; Performance analysis; Shared and Switched Ethernet; Related protocols such as ICMP, NAT, ARP and RARP.

Network Layer: Internet Protocol (IP) headers; Routing protocols including distance-vector and link-state approaches; Interior and Exterior Gateway Protocol concepts; Routing Algorithms including Dijkstra's algorithm and distributed Bellman-Ford algorithm; Example protocols: OSPF, RIP, BGP.

Transport Layer: Socket Interface and Socket programming; Reliable end-to-end transmission protocols; UDP header; Details of TCP header and operation including options

headers and congestion control; TCP variants such as Reno, Tahoe, Vegas, Compound and CUBIC.

Application Layer: DNS, SMTP, FTP, SNMP, HTTP, VPN, VLANs, MPLS, Network security: basic concept, Advanced topics: Data center Network (DCN), Software Defined Networking (SDN).

Reference Books/Materials

1. Larry L. Peterson, Bruce S. Davie, "Computer Networks - A Systems Approach", Morgan Kaufmann, sixth edition
2. James F. Kurose, Keith W. Ross, "Computer Networking: A Top-Down Approach", Pearson Education.
3. Andrew S. Tanenbaum, "Computer Network", Prentice Hall of India, 5th Edition.
4. Behrouz A. Forouzan, "Data Communications and Networking", McGraw-Hill.
5. William Stallings, "Data and Computer Communication", Pearson

Course Code	Course Name	L	T	P	Credits
CS 353	Compiler Design Laboratory	0	0	3	2

Course Objectives:

To obtain the practice of writing compilers.

Course Outcomes:

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Write code for tokenize the given C-Program segment and remove: i. white spaces, ii. Horizontal/vertical tabs. iii. Comments of single line or multiline lines

CO2. Write code to do parsing techniques and evaluate the expression to do syntax analysis

CO3. Write code to do check semantic rules for semantic analysis

CO4. Write code to optimization techniques for intermediate code

CO5. Develop a mini compiler for given C-program segment

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

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

Syllabus

List of Experiments

1) Introduction to Flex/Lex& Bison/Yacc tools, Lexing and tokenizing Programs

2) Implementing grammars for C-program segments

3) Parsing and parse trees

4) Type checking

5) Intermediate code generation

6) Simple optimization (constant folding, etc.)

7) Control flow

8) Functions

9) Building a mini compiler (for C Language) and executing Simple problems to demonstrate the Compiler capabilities

Reference Books/Materials

1. Holub A.I., "Compiler Design in C", Prentice Hall India.2000.

2. W. Appel, "Modern Compiler Implementation in C", Cambridge University Press, 1998.

3. V. Aho, M. S. Lam, R. Sethi, J. D. Ullman, "Compilers- Principles,Techniques & Tools", 2/e, Pearson Education, 2007.

Course Code	Course Name	L	T	P	Credits
CS354	Computer Networks Lab	0	0	3	2

Course Objective

To provide students with a theoretical and practical base in computer networks issues. Analyze and compare various networking protocols. Demonstrate the working of different concepts of networking.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the channel allocation, error and flow control techniques, and channel access techniques

CO2. Understand the routing protocols in the implementation level with the exception scenarios

CO3. Experience the packet transmission and concurrent application with TCP using socket programming

CO4. Learn the Wireshark real-time packet analyzer. Understand the IP packet header, the application layer protocols such as HTTP, DNS and DHCP

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	M	M	M	H	L	M	M	H	H	M	H
CO2	H	H	H	H	H	M	L	M	H	H	M	H
CO3	H	H	M	H	H	L	M	M	H	H	H	H
CO4	H	M	H	H	H	M	L	M	H	H	H	H

Syllabus

List of Experiments

Implement the following using any programming language

1. Implement the error detection method such as CRC and Checksum
2. Implementation of channel access protocols in data link layer such as Aloha, CSMA, CSMA/CD protocols. Comparison study on the performance of these protocols.
3. The routing protocols such as RIP, OSPF, BGP.
4. Implementation of basic Client Server program using TCP and UDP Socket.
5. Using TCP/IP sockets, write a client server program to make the client send the file name and to make the server send back the contents of the requested file if present.
6. Implementing fully concurrent application with a TCP server acting as a directory server and client programs allowing concurrent connection and message transfer (Eg. Chat system).
7. Implementation of ARQ protocols in TCP layer, and observe the reliable data transmission over an unreliable source or service.

Using Wireshark

8. Investigate the IP protocol, focusing on the IP datagram. Analyz a trace of IP datagrams sent and received by an execution of the traceroute program
9. Explore several aspects of the HTTP protocol: the basic GET/response interaction, HTTP message formats, retrieving large HTML files, retrieving HTML files with embedded objects, and HTTP authentication and security.
10. Observe DHCP in action, perform several DHCP-related commands and capture the DHCP messages exchanged as a result of executing these commands.
11. Observe the local DNS servers, DNS caching, DNS records and messages, and the TYPE field in the DNS record.

Reference Books/Materials

1. James F.Kurose, Computer networking: Atop-down approach featuring the internet, Pearson Education India

2. Online Resources: Interactive animations, Video notes from Kurose and Ross book. Wire shark assignments, Presentation slides, interactive exercises from the following link: https://gaia.cs.umass.edu/kurose_ross/wireshark.php.

3. https://www.wireshark.org/

Course Code	Course Name	L	T	P	Credits
HS 350	INDUSTRIAL ECONOMICS	3	0	0	3

COURSE OBJECTIVES

1. Identify and analyze the behaviour of a firm under different market situations systematically.
2. Understand and assimilate the issues related to strategic behaviour in firms, R&D and innovation.
3. Have a comprehensive coverage of firms' profitability and efficiency measurements, with applications to India's industrial structure.
4. To understand the rich complexities and paradox of fourth industrial revolution.

COURSE OUTCOMES

At the completion of this course, the student shall acquire knowledge of

CO1. Market structure, conduct and performance

CO2. Strategic behaviour in firms

CO3. Innovation, R&D and the market

CO4. Industrial efficiency and its applications for the Indian economy

Unit-1: Introduction to Economics – Introduction to Industrial economics - nature and scope - concept of firm and industry- types of firms - structure, conduct and performance. [5 hours]

Unit-2: Standard forms of market structure - pricing strategies and output determination of firms - profit maximization, sales maximization (William J. Baumol), utility maximization (Oliver E. Williamson), growth maximization (George K. Yarrow) - equilibrium of firms under perfect competition, monopoly, monopolistic competition and oligopoly - optimum price and output - economies of scale. [10 hours]

Unit-3: Price and non-price competition - strategic behaviour of firms - collusion and mergers - game theory - market failures and information asymmetry - advertising and product differentiation - market entry and exit - concentration and diversification [10 hours]

Unit- 4: Patents and technological change- the economics of patent-innovation and diffusion measures of concentration [5 hours]

Unit- 5: Research and Development (R&D) and market structure -- product and process innovation- R&D and patent race-licensing and incentive to innovate [6 hours]

Unit-6: Economics of the fourth Industrial Revolution – Industrial revolution past, present and Future, Internet-Artificial Intelligence- Blockchain technologies [6 hours]

ESSENTIAL READING

1. Donald A. Hay, Derek J. Morris, *Industrial Economics: Theory and Evidence*, Oxford University Press, 1979
2. Carton, D. and J. Perloff. Modern Industrial Organization (Reading, Massachusetts: Addison-Wesley), 1999.
3. Lall, Sanjaya. Competitiveness, Technology and Skills (Cheltenham: Edward Elgar), 2001.
4. Shy, O. (1996). *Industrial organization: Theory and applications*. MIT Press.

SUPPLEMENTARY READING

1. A. Singh and A.N. Sandhu, *Industrial Economics*, Himalaya Publishing House, Bombay, 1988
2. Ferguson, Paul R. and Glenys J. Ferguson, (1994), Industrial Economics - Issues and Perspectives, Macmillan, London.
3. Stephen Martin, *Advanced Industrial Economics*, Oxford, UK Blackwell Publisher, 2002
4. R. R. Barthwal, *Industrial Economics: An Introductory Textbook*, New Age International Publishers, 2007
5. Hay, Donald A. and Derek J. Morris. Industrial Economics and Organization: Theory and Evidence, 2nd Edition (Oxford: Oxford University Press), 1991.
6. Schmalensee, R., Inter-industry studies of Structure and Performance, in Schmalensee, R. and R. D. Willig (eds.): Handbook of Industrial Organization [Amsterdam: North-Holland] Vols. 2 Chapter 16, pp. 951-1009, 1989.
7. Siddharthan, N. S. and Y.S. Rajan. Global Business, Technology and Knowledge Sharing: Lessons for Developing Country Enterprises (New Delhi: Macmillan), 2002.
8. Tirole, Jean. The Theory of Industrial Organization (Cambridge, MA: The MIT Press), 1988.

Course Code	Course Name	L	T	P	Credits
ES300	Environmental Studies	1	0	0	1

COURSE OBJECTIVES

Understanding environment, its constituents, importance for living, ecosystem, human developmental activities vs environment, climate change, national and international environment related developments, need for public awareness, its protection and conservation activities.

COURSE OUTCOMES

At the completion of this course, the student shall acquire knowledge of

CO1. Understand in-depth knowledge on natural processes and resources that sustain life

CO2. Understand the effect of human interference on the web of life, economy, and quality of human life.

CO3. Develop critical thinking for shaping strategies for environmental protection, conservation of biodiversity, environmental equity, and sustainable development

CO4. Acquire values and attitudes towards understanding complex environmental economic- social challenges, and active participation in solving current environmental problems and preventing the future ones

CO5. Adopt sustainability as a practice in life, society, and industry

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		L	L	H		H	H	H	H	L	L	H
CO2		H	H	M	L	H	H	H	H	L	L	H
CO3		H	H	L		H	H	H	M	L	L	H
CO4		L	L	L		L	H	H	L	L	L	H
CO5		H	H	M	M	M	H	H	H	L	M	H

Syllabus

Module 1

Introduction: Environment, interaction organism, scale of interaction, types of environment, Human interference, environmental ethics, environmental problems, sustainable society, ecological foot prints.

Module 2

Ecosystem: current status, Role of organism, species, Life supporting system, Factors sustaining life, Components of ecosystem, Ecological efficiency, Matters in ecosystem, Major chemical cycles, Role of Species, Classification of species.

Module 3

Biodiversity and species interaction: Biodiversity and Ecosystem, Species interaction, Natural selection, population growth, factor limiting population growth, Population dynamics, Species and reproductive pattern, Biodiversity, Population and Economy, Food and nutrition.

Reference Books/Materials

1. Textbook for Environmental Studies For Undergraduate Courses of all Branches of Higher Education (online book -UGC Website), Erach Bharucha,University Grants Commission , India.
2. Daniel B. Botkin, Edward A. Keller, “Environmental Science: Earth as a living planet”, 8th edition, John Wiley & Sons, Inc., ISBN 978-0-470-52033-8
3. G. Tyler Miller, Jr., Scott Spoolman, “Environmental Science: Problems, Concepts, and Solutions”, 16th edition, Brooks/Cole, ISBN-13: 978-0-495-55671-8
4. William P. Cunningham, Mary Ann Cunningham, “Principles of environmental science: Inquiry & application”, 7th edition, Mcgraw-Hill, ISBN 978–0–07–353251–6
5. William P. Cunningham, Mary Ann Cunningham, “Environmental science: A global concern”, 12th edition, Mcgraw-Hill, ISBN 978–0–07–338325–5

List of Electives

SL No	Course Code and Name
1	CS500 Object Oriented Analysis and Design (OOAD)
2	CS501 Advanced Data Structures (ADS)
3	CS502 Advanced Computer Architecture (ACA)
4	CS503 Principles of Programming Languages (PPL)
5	CS504 Systems Programming
6	CS505 Data Warehousing and Data Mining (DWDM)
7	CS506 Advanced Database Systems (ADBS)
8	CS507 E-Commerce (EC)
9	CS508 Advanced Operating Systems (AOS)
10	CS509 Cyber Laws & Intellectual Property Right (CLIPR)
11	CS510 Information Theory (IT)
12	CS511 Optimization Techniques in Computing (OT)
13	CS512 Soft Computing (SC)
14	CS513 Applied Algorithms (AA)
15	CS514 Network Management(NM)
16	CS515 Software Architecture (SA)
17	CS516 Web Engineering (WE)
18	CS517 Software Project Management (SPM)
19	CS518 Advanced Compilers (AC)
20	CS519 Computer Vision (CV)
21	CS520 Artificial Intelligence (AI)
22	CS521 Multimedia & Virtual Reality (MVR)
23	CS522 Protocol Engineering(PE)
24	CS523 Software Testing (ST)
25	CS524 Mobile Communications (MC)
26	CS525 Network Security(NS)
27	CS526 Parallel Algorithms (PA)
28	CS527 Distributed Algorithms (DA)
29	CS528 Computer Security Audit and Assurance (CSAA)
30	CS529 Big Data Analysis (BDA)
31	CS530 Secure Software Engineering (SSE)
32	CS531 Computer Graphics (CG)
33	CS532 Graph Theory (GT)
34	CS533 Distributed Computing Systems (DCS)
35	CS534 Virtualization and Cloud Computing
36	CS535 Internet of Things
37	CS536 High Performance Computing
38	CS537 Cluster and Grid Computing
39	CS538 Quantum Computing
40	CS539 Advanced topics in IoT
41	CS540 Modern Cryptography
42	CS541 Cryptocurrency and Blockchain Technology
43	CS542 Penetration Testing
44	CS543 Cyber Crime Investigation and Digital Forensics
45	CS544 Malware Analysis and Mitigation

46	CS545 Design and Analysis of Security Protocols
47	CS546 Privacy Preserving Data Publishing
48	CS547 Secure Multi-Party Computation
49	CS548 Public Key Infrastructure and Trust Management
50	CS549 Security Analytics
51	CS550 Deep Learning

Detailed Syllabus

of

Elective Courses

Course Code	Course Name	L	T	P	Credits
CS 500	Object Oriented Analysis and Design (OOAD)	3	0	0	3

Course Objective

To apply an iterative process such as the Unified Process & Analyze software requirements and document them using Use Cases. Perform software analysis and record the results using UML notation. Discuss how object oriented software development affects testing and quality.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand object-oriented systems development and its life cycle

CO2. Apply UML methodologies to model object-oriented systems

CO3. Analyze system requirements, object relationships, and design principles

CO4. Evaluate interface design, software quality, and usability metrics

CO5. Develop a mini-project using object-oriented analysis and design

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	L	L	L	L	-	-	-	L	L	L	L
CO2	H	H	H	L	L	-	-	-	M	M	L	L
CO3	H	H	H	L	L	L	-	-	M	L	L	L
CO4	H	H	H	M	M	L	-	-	M	L	L	L
CO5	H	H	H	M	H	L	-	-	H	M	H	L

Syllabus

Module 1 :

An overview of object oriented systems development, object basics, object oriented systems development life cycle.

Module 2 :

Rumbaugh methodology , Booch methodology , Jacobson methodology , patterns, frameworks, unified approach, unified modeling language , use case , class diagram , interactive diagram , package diagram , collaboration diagram , state diagram , activity diagram.

Module 3 :

Identifying use cases, object analysis, classification, identifying object relationships, attributes and methods, design axioms, designing classes, access layer, object storage, and object interoperability.

Module 4 :

Designing interface objects, software quality assurance, system usability, measuring, user satisfaction, mini project.

Reference Books/Material

1. Ali Bah rami, “Object Oriented Systems Development”, Tata McGraw-Hill, 1999.
2. Martin Fowler, “UML Distilled”, 2nd ed., PHI/Pearson Education, 2002
3. Stephen R. Schach, “Introduction to Object Oriented Analysis and Design”, Tata McGraw-Hill, 2003
4. James Rumbaugh, Ivar Jacobson, Grady Booch “The Unified Modeling Language Reference Manual”, Addison Wesley, 1999
5. Hans-Erik Eriksson, Magnus Penker, Brain Lyons, David Fado, “UML Toolkit”, OMG Press Wiley Publishing Inc., 2004

Course Code	Course Name	L	T	P	Credits
CS 501	Advanced Data Structures (ADS)	3	0	0	3

Course Objective

Advanced Data Structures is about using mathematical objects like trees and graphs to represent computational problems. It aims at the usage of some sophisticated algorithms and methods of analysis.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the fundamental concepts of algorithms, including their design, analysis apply these to sorting and order statistics

CO2. Develop various algorithms and solve problems using basic and advanced data structures

CO3. Apply graph algorithms for solving real-world problems related to shortest paths, spanning trees, and maximum flow in networks

CO4. Evaluate advanced algorithmic techniques such as greedy algorithms and dynamic programming to solve optimization problems efficiently

CO5. Analyze string matching algorithms and NP-completeness for computational complexity.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	L	-	-	-	L	L	M	L
CO2	H	H	H	M	-	-	-	-	M	M	L	L
CO3	H	H	H	M	L	-	-	-	M	L	M	L
CO4	H	H	H	M	L	-	-	-	M	L	M	L
CO5	H	H	H	M	L	-	-	-	M	L	L	L

Syllabus

Module 1 :

General Idea, Hash Function, Separate Chaining, Hash Tables without linked lists: Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Hash Tables in the Standard Library, Universal Hashing, Extendible Hashing.

Module 2 :

Binary Heap: Structure Property, Heap Order Property, Basic Heap Operations: insert, delete, Percolate down, Other Heap Operations. Binomial Queues: Binomial Queue Structure, Binomial Queue Operations, Implementation of Binomial Queue, Priority Queues in the Standard Library.

Module 3 :

Splay Trees, Splaying operations and amortized analysis, Scape Goat Tree, Treap as randomized binary search tree, implementation of search insert and delete in Treap, segment tree basics, structure, construction and operations, Range Minimum Query- applications and optimization, persistent segment tree

Module 4 :

Elementary Graph Algorithms: Topological sort, Single Source Shortest Path Algorithms: Dijkstra's, Bellman-Ford, All-Pairs Shortest Paths: Floyd-Warshall's Algorithm : Greedy algorithms, an activity, selection problem, elements of greedy strategy, Huffman codes. Dynamic programming:

Module 5 :

Disjoint Sets – Equivalence relation, Basic Data Structure, Simple Union and Find algorithms, Smart Union and Path compression algorithm.. String Matching: The naïve string matching algorithm, Rabin-Karp algorithm, Knuth-Morris-Pratt algorithm. NP-Completeness: Polynomial time, Verification, NP-Completeness and reducibility

Reference Books/Material

1. Thomas Cormen, Charles E Leiserson and Ronald D River, "Introduction to Algorithms", PHI, 2001.
2. Mark Allen Weiss, Algorithms, "Data Structures and Problem Solving with C++", Addison Wesley, 2002.

3. Mark Allen Weiss, Algorithms, "Data Structures and Problem Solving with C++", Addison Wesley, 2002
4. EllisHorowitz, SatrajSahni and S.Rajasekaran, "Fundamentals of computer algorithms",Galgotia publications pvt.Ltd.
5. Reema Thareja, S. Rama Sree , "Advanced Data Structures", Oxford University Press, 2018

Course Code	Course Name	L	T	P	Credits
CS 502	Advanced Computer Architecture (ACA)	3	0	0	3

Course Objective

To understand concepts of parallel processing and design choices of implementing parallel execution within a single processor (pipeline, VLIW, and superscalar) and multiprocessor systems. To gain knowledge of the state of the art research topics on advanced computing systems

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1: Understand the parallel processing that characterizes the modern-day processors

CO2: Analyze the design choices of implementing the parallel execution in a single and multi-processor systems

CO3: To apply the concepts of parallel execution in a given scenario of task involving a large program execution

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	L	L	L	L							
CO2	H	H	M	M	M							
CO3	M	M	L	M	L							

Syllabus

Module 1 :

Parallel Computer Models: The state of computing, classification of parallel computers, multiprocessors and multicomputer, multi vector and SIMD computers. Program and network properties: Conditions of parallelism, data and resource dependences, hardware and software parallelism, program partitioning and scheduling, grain size and latency, program flow mechanisms, control flow versus inter connects, hierarchical bus systems, crossbar switch and multiport memory, multistage and combining network

Module 2 :

Advanced Processors: Advanced processor technology, instruction-set architectures, CISC scalar processors, RISC scalar processors, superscalar processors, VLIW architectures, vector and symbolic processors.

Module 3 :

Pipelining: Linear pipeline processor, nonlinear pipeline processor, instruction pipeline design, mechanisms for instruction pipelining, dynamic instruction scheduling, branch handling techniques, branch prediction, arithmetic pipeline design, computer arithmetic principles, static arithmetic pipeline, multifunctional arithmetic pipelining

Module 4 :

Multi Processors: Multiprocessor system interconnect, cache coherence and synchronization mechanisms, message-passing mechanism, scalable, multi-threaded and dataflow architectures: latency-hiding techniques, principles of multithreading, scalable and multithreaded architecture, dataflow and hybrid architectures.

Module 5 :

Parallel Models, languages and compilers: Latency-Hiding techniques environment, synchronization and multiprocessing modes, shared variable program structures, message passing programming development

Reference Books/Material

(1) Dezso Sima, Terence Fountain, Peter Kacsuk, “Advanced computer architectures: A design space approach” , Addison Wesley.

(2) K.Hwang and F.A. Briggs, “Computer architecture and parallel processing”, McGraw Hill Publications

(3) K. Hwang, “Advanced computer architecture-parallelism, scalability, programmability” , McGraw Hill.

(4) J. Hennesy and D. Patterson, “Computer architecture –A quantitative approach”, Morgan Kaufmann, 200.3

Course Code	Course Name	L	T	P	Credits
CS 503	Principles of Programming Languages (PPL)	3	0	0	3

Course Objective

The basic thrust of this course will be on learning the distinctive techniques in the different paradigms and what semantic and compiling issues come up in the various languages considered. The course introduces Imperative Languages, functional programming, declarative programming and semantics of object oriented programming.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1: Understand to express syntax and semantics in formal notation

CO2: Employ to apply suitable programming paradigm for the application

CO3: Design to program in different language paradigms and evaluate their relative benefits

CO4: Understand the programming paradigms of modern programming languages

CO5: Understand the concepts of declarative programming and OOP

CO6: Knowledge to compare the features of various programming languages

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	M	M	M	M	M	L	L	M	L	L	M
CO2	M	M	H	M	M	L	L	L	M	M	M	M
CO3	H	H	M	H	L	L	L	L	M	M	M	M
CO4	H	M	M	M	L	M	L	L	M	M	M	M
CO5	M	L	M	L	M	M	L	L	M	M	M	M
CO6	M	H	M	M	L	M	L	L	H	M	M	M

Syllabus

Module 1 :

Imperative and object-oriented programming, role of types, static and dynamic type checking, scope rules, grouping data and operations, information hiding and abstract data types, objects, inheritance, polymorphism, templates.

Module 2 :

Functional programming, expressions and lists, evaluation, types, type systems, values and operations, function declarations, lexical scope, lists and programming with lists, polymorphic functions, higher order and curried functions, abstract data types.

Module 3 :

Logic programming, review of predicate logic, clausal-form logic, logic as a programming language, unification algorithm, abstract interpreter for logic programs, semantics of logic programs, programming in prolog.

Module 4 :

Lambda calculus and semantic environment and rules.

Reference Books/Material

(1) Kenneth C. Louden, “Programming Languages: Principles and Practice”, 2nd ed., Thomson 2003.

(2) Carlo Ghezzi, Mehdi Jazayeri, “Programming Language Concepts”, 3rd ed., John Wiley & Sons, 1997.

(3) Ravi Sethi, "Programming Languages: Concepts and Constructs", 2 nd ed., Pearson Education Asia.

Course Code	Course Name	L	T	P	Credits
CS504	Systems Programming	3	0	0	3

Course Objective

To understand the relationship between system software and machine architecture to design and implement assemblers, linkers and loaders.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand how to write assembly language program

CO2. Understand the relationship between the system software and machine architecture

CO3. Implement the system software such as assemblers, linkers and loaders

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	L	M	M	M							
CO2	H	M	M	L	M							
CO3	H	M	H	M	M							

Syllabus

Module 1:

Components of a programming system: Assemblers, loaders, macros, compilers, machine Structure: Memory, registers, data, instructions. Machine language: Address modification using instructions as data, address modification using index registers, looping Assembly language.

Module 2:

Assemblers: Basic assembler functions with an example assembler, assembler algorithm and data structures, machine dependent assembler features, machine independent assembler features, one-pass assemblers, multi-pass assemblers, implementation example. Table processing: Searching and sorting.

Module 3:

Loaders and Linkers: Basic loader functions, design of an absolute loader, a simple bootstrap loader, machine dependent loader features, program linking, algorithms and data structures for lining, machine independent loader features, automatic library search, loader design options, dynamic linking and an implementation example.

Module 4:

Macro processors: Basic macro processor functions, macro definition and expansion, macro processor data structures and algorithms, implementation example, discussion of ANSI C macro language.

Module 5:

System Software Tools: Text editors, overview of the editing process, user interface, editor structure, interactive debugging systems, debugging functions and capabilities, relationship with other parts of the system.

Reference Books/Material

1. Leland L. Beck, D. Manjula, "System software: An introduction to systems programming", Pearson education, 3rded, 2007.
2. John J. Donovan, "Systems Programming", Tata McGraw-Hill Edition, 2009
3. D.M. Dhamdhere, "Introduction to system software", Tata McGraw Hill Publications, 2002.
4. John R. Levine, "Linkers & Loaders", Morgan Kaufmann Publishers, 2000.

Course Code	Course Name	L	T	P	Credits
CS505	Data Warehousing and Data Mining (DWDM)	3	0	0	3

Course Objective

Following this course, students will be able to

1) Learn the concepts of database technology,

2) Understand data mining principles and techniques,

3) Discover interesting patterns from large amounts of data to analyze and extract patterns to solve problems, make predictions of outcomes.

4) Evaluate systematically supervised and unsupervised models and algorithms with respect to their accuracy,

5) Design and implement of a data-mining application using sample, realistic data sets and modern tools.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the data Warehouses, Operational Data Stores (ODS) and OLAP characteristics

CO2. Understand the data mining concept, application and their usage

CO3. Analyze the frequent patterns using association analysis algorithms like apriori, FP-growth etc by solving problems

CO4. Understand the concept of classification, different classification algorithms and their applications and design classifiers such as Decision tree, Naïve Bayes', Rule based approach etc.

CO5. Understand the concept of clustering and different cluster analysis methods

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	M	M	L	L	L	M	L	L	M	M	M	M
CO2	M	M	L	M	M	M	L	L	M	M	M	M
CO3	M	M	H	M	M	M	L	L	M	M	M	M
CO4	M	H	M	M	M	M	L	L	M	M	M	M
CO5	M	M	H	M	M	M	L	L	M	M	M	M
CO6	M	H	M	M	L	M	L	M	M	M	M	M

Syllabus

Module 1 :

Introduction to data warehousing, building a data warehouse, mapping the data warehouse to a multiprocessor architecture, OLAP technology for data mining, data warehouse, multidimensional data model, data warehouse architecture, data warehouse implementation, OLAP guidelines, multidimensional versus multi relational OLAP, categories of tools, DBMS schemas for decision support data extraction, cleanup and transformation tools for metadata, development of data cube technology, from data warehousing to data mining, data generalization, efficient methods for data cube computation, further development of data cube and OLAP Technology, attribute-oriented induction.

Module 2 :

Introduction to data mining tasks, objectives (classification, clustering, association rules, sequential patterns, regression, deviation detection).

Module 3 :

Data and preprocessing (data cleaning, feature selection, dimensionality reduction).

Module 4 :

Classification (decision-tree based approach, rule-based approach, instance-based classifiers, Bayesian Approach: Naive and Bayesian networks, classification model evaluation).

Module 5 :

Clustering (partitional methods, hierarchical methods, graph-based methods, density-based methods, cluster validation methods), anomaly/outlier detection (introduction to various types of outliers, statistical-based, density-based and other methods for outlier detection).

Reference Books/Material

1. Jiawei Han and Micheline Kamber, “Data mining: Concepts and techniques”, 2 nd ed., Morgan Kaufmann publishers.
2. Raph Kimball,” Data warehouse toolkit”, John Wiley & Sons Publications
3. Michael. J. Berry, Gordon Linoff, “Data mining techniques: Marketing, sales, customer support”, John Wiley & Sons.

Course Code	Course Name	L	T	P	Credits
CS 506	Advanced Database Systems (ADBS)	3	0	0	3

Course Objective

To develop an appreciation of emerging database trends as they apply to semistructured data, the internet, and object-oriented databases. To explain the process of DB Query processing and evaluation.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand and analyze the fundamentals of relational database systems including: data models, database architectures and ER features, normalization, transactions and concurrency protocols

CO2. Understand the roles that databases play in organizations and familiarize with basic database storage, file organization, database accessing techniques and query processing in relational databases

CO3. Understand the advanced database concepts such as distributed databases, parallel databases, spatial databases, advanced transaction databases and semi-statured databases

CO4. Understand the different techniques used parallel, distributed databases to handle databases

CO5. Understand query processing in advanced databases systems Analyzing advanced concurrency protocols and data security concepts in advanced database systems

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	L	L	M	L	L	M	L	M	L	M	M
CO2	M	M	L	M	M	L	M	L	M	L	M	M
CO3	M	M	L	M	H	M	M	L	M	M	M	M
CO4	M	M	M	M	H	H	M	L	M	M	M	M
CO5	M	M	L	M	M	M	M	L	M	M	M	M
CO6	M	M	M	L	H	H	M	L	M	M	H	M

Syllabus

Module 1 :

Distributed database concepts, overview of client-server architecture and its relationship to distributed databases, concurrency control heterogeneity issues, persistent programming languages, object identity and its implementation, clustering, indexing, client server object bases, cache coherence.

Module 2 :

Parallel databases: Parallel architectures, performance measures, shared nothing/shared disk/shared memory based architectures, data partitioning, intra-operator parallelism, pipelining, scheduling, load balancing, query processing- index based, query optimization: cost estimation, query optimization: algorithms, online query processing and optimization, XML, DTD, XPath, XML indexing, adaptive query processing.

Module 3 :

Advanced transaction models: Save points, sagas, nested transactions, multi-level transactions, Recovery, multilevel recovery, shared disk systems, distributed systems 2PC, 3PC, replication and hot spares, data storage, security and privacy- multidimensional k- anonymity, data stream management.

Module 4 :

Models of spatial data: Conceptual data models for spatial databases (e.g. pictogram enhanced ERDs), logical data models for spatial databases: raster model (map algebra), vector model, spatial query languages, need for spatial operators and relations, SQL3 and ADT. spatial operators, OGIS queries.

Reference Books/Material

1. AviSilberschatz, Henry Korth, and S. Sudarshan, " Database system concepts", 5 th ed., McGraw Hill, 2005.
2. S. Shekhar and S. Chawla, "Spatial databases: A tour, Prentice Hall", 2003.
3. Ralf HartmutGuting, Markus Schneider, "Moving objects databases", Morgan Kaufman, 2005.
4. R. Elmasri and S. Navathe, "Fundamentals of database systems", BenjaminCummings,5th ed., 2007.
5. Raghu Ramakrishnan, "Database management systems", McGraw-Hill, 2000.

6. Ceri S and Pelagatti G, “Distributed databases principles and systems”, 2nd ed., Mc-Graw Hill, 1999.

Course Code	Course Name	L	T	P	Credits
CS 507	E-Commerce (EC)	3	0	0	3

Course Objective

To provide principles of e-commerce from a business perspective.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the concepts of e-commerce applications and different models

CO2. Analyze security risks in e-commerce applications and business models

CO3. To explore newer solutions

CO4. To develop newer e-commerce applications with the advancements in the technologies

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO 12
CO1	M			L					M			
CO2	H		M			L				H	M	
CO3		L	M		M		M			H		M
CO4	M			H				L			M	

Syllabus

Module 1 :

Infrastructure and tools for e-commerce, current trends in e-commerce applications development, the business of internet commerce, enterprise level e-commerce.

Module 2 :

Security and encryption, electronic payment systems, search engines, intelligent agents in ecommerce, on-line auctions, data mining for e-commerce

Module 3 :

Web metrics, recommended systems, knowledge management, mobile e-commerce, legal, ethical and social issues.

Module 4 :

Seminars and mini projects

Reference Books/Material

(1) Henry Chan et al., "E-Commerce-Fundamental and applications", John Wiley & Sons 2002..

(2) G. Winfield Treese and Lawrence C.S., "Designing Systems for Internet Commerce", Pearson Education, LPE, 2002

(3) Fensel, Dieter, Brodie M.L., "Ontologies: A Silver Bullet for Knowledge Management and ECommerce", Allied Publishers, 2004

(4) Zimmermann, Olaf Tomlinson, Mark R.: Peuser, Stefan, "Perspectives on Web Services", Allied Publilshers, 2004

Course Code	Course Name	L	T	P	Credits
CS 508	Advanced Operating Systems (AOS)	3	0	0	3

Course Objective

To provide comprehensive and up-to-date coverage of the major developments in distributed operating system, multi-processor operating system and database operating system.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand distributed system architectures, communication, and synchronization

CO2. Analyze deadlock handling, detection, resolution, and agreement protocols

CO3. Apply RTOS scheduling, IPC mechanisms, and virtualization techniques

CO4. Evaluate and optimize embedded system constraints, power, and networking

CO5. Develop mobile applications using the Android OS

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	M	L	-	-	-	M	-	L
CO2	H	H	H	M	M	L	-	-	-	M	-	L
CO3	H	H	H	H	M	L	-	-	-	M	-	L
CO4	H	H	H	H	H	L	L	M	-	H	L	M
CO5	H	H	H	M	H	L	L	-	M	M	M	M

Syllabus

Module 1 :

Architectures of distributed systems , system architecture types, issues in distributed OS, communication networks, primitives, theoretical foundations, inherent limitations of a distributed system, lamp ports logical clocks, vector clocks, casual ordering of messages, global state, cuts of a distributed computation, termination detection, distributed mutual exclusion.

Module 2 :

Distributed deadlock detection, introduction, deadlock handling strategies in distributed systems, issues in deadlock detection and resolution, control organizations for distributed deadlock detection, centralized, distributed and hierarchical deadlock detection algorithms , agreement protocols.

Module 3 :

Real-Time Operating Systems (RTOS): Characteristics and design considerations for RTOS, Rate-monotonic (RMS), earliest deadline first (EDF), and least slack time (LST) scheduling. Deterministic Scheduling , Interrupt Handling & Latency Optimization: Low-latency ISR design, non-maskable interrupts (NMI), and real-time response tuning, High-Performance IPC Mechanisms: Real-time message passing, shared memory optimizations, and event-driven execution. Real-Time Virtualization: Running RTOS alongside general-purpose OS using hypervisors. Case Studies:

Industry-Standard RTOS, RTOS Comparisons and Implementations, FreeRTOS, VxWorks, QNX Neutrino, RTEMS, Zephyr RTOS

Module 4 :

Embedded System Constraints & Optimizations, Power-Aware RTOS: Dynamic Voltage and Frequency Scaling (DVFS), low-power RTOS (FreeRTOS, Zephyr), Memory-Constrained RTOS Design: Fixed-size memory allocation, real-time garbage collection, and minimizing fragmentation, Kernel Optimizations for Embedded Devices: Preemptive multitasking, lock-free data structures, and microkernel-based RTOS, High-Performance Embedded Networking: Real-time Ethernet, TSN (Time-Sensitive Networking), CAN bus in automotive systems, Security in Embedded RTOS: Secure boot, real-time encryption, and intrusion detection in IoT and critical systems. Case Studies: Cutting-Edge Industry Applications

Module 5 :

Android Software Platform, Android Architecture , Operating System Services., Android Runtime Application Development, Application Structure Application Process Management, Mobile applications development.

Reference Books/Material

1. Mukesh Singhal Niranjan, Shivorothri G., “Advanced Concepts in Operating systems” , McGraw-Hill Education (India) Pvt Limited, 2001
2. Andrew S. Tanenbaum, “Distributed Operating systems” , Pearson Education, 1995
3. K.C. Wang , “Embedded and Real-Time Operating Systems” Second Edition, Springer, 2023
4. Hermann Kopetz “Real-Time Systems: Design Principles for Distributed Embedded Applications”, Springer, Second Edition, 2011
5. Bill Phillips, Chris Stewart, Kristin Marsicano “Android Programming: The Big Nerd Ranch Guide”, Pearson Technology Group, Fourth Edition, 2019
6. Doreen L. Galli, “Distributed operating systems - concepts and practice”, Prentice-Hall 2000.
7. A Silberschatz, “Applied Operating systems Concepts”, Wiley 2000
8. Lubemir F. Bic& Alan C. Shaw, “Operating systems Principles”, Pearson Education, 2003.

Course Code	Course Name	L	T	P	Credits
CS 509	Cyber Laws & Intellectual Property Right (CLIPR)	3	0	0	3

Course Objective

To introduce the cyber world, intellectual property law and cyber law in general to explain about the various facets of cyber-crimes, to enhance the understanding of problems arising out of online transactions and provoke them to find solutions, to clarify the Intellectual Property issues in the cyber space and the growth and development of the law in this regard and to educate about the regulation of cyber space at national and international level.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the concepts of cyber laws and Acts

CO2. Analyze the IP related issues

CO3. Analyze cyber crimes and related laws

CO4. Develop newer models with the technological advancements

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M			L			M			M		
CO2			H			M			M			
CO3		L		M		M		M			H	
CO4	H		M		L					M		M

Syllabus

Module 1 :

Cyber laws and IT act; the rights the various parties have with respect to creating, modifying, using, and distribution, storing and copying digital data..

Module 2 :

Concurrent responsibilities and potential liabilities, intellectual property issues connected with use and management of digital data, the similar act of other countries.

Module 3 :

Computer crime, computer fraud, hacking.

Module 4 :

Unauthorized modification of information, privacy, computer pornography harassment.

Reference Books/Material

1. D. Brainbridge, “Introduction to computer law”, 5thed., Pearson Education,2004.
2. P. Duggal, “Cyber law: the Indian perspective”, 2005.

Course Code	Course Name	L	T	P	Credits
CS 510	Information Theory (IT)	3	0	0	3

Course Objective

Information Theory is the science for measuring, preserving, transmitting, and estimating information in random data. It provides the fundamental limits of performance for transmission of messages generated by a random source over a noisy communication channel. On the one hand, Information Theory has been the driving force behind the revolution in digital communication and has led to various practical data compression and error correcting codes that meet the fundamental theoretical limits of performance. The objective of this course is to introduce the basic notions and results of Information Theory, keeping in mind both its fundamental role in communication theory and its varied applications beyond communication theory.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the overview of probability theory, significance of "Information" with respect to Information Theory

CO2. Explain various methods of generating and detecting different types of error correcting codes

CO3. Explain various methods of generating and detecting different types of error correcting codes

CO4. Furthermore, the student should be able to formalize compression, transmission, and estimation problem in an information theoretic setting

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

Module 1 :

Introduction to Probability: Random Variables, Random variable, Sample space, Conditional probability, Joint probability. Modeling of Information Sources – Self Information, Entropy, Mutual Information. Source Coding Theory and algorithms – Kraft inequality, Huffman algorithm, Arithmetic coding, Lempel Ziv coding. Modeling of Communication channels – Binary symmetric channel, Binary Erasure channel, Channel coding theorem.

Module 2 :

Information measures (entropy): Discrete entropy and the law of large numbers, bounds on typicality, properties of entropy – rate characterization, conditional, relative, joint entropy, mutual information, source coding theorem (loss less for DMS), existence of minimum information, entropy as divergence, entropy rates of various families, entropy rate of Markov sources, comments on complexity

Module 3 :

Source coding for DMS: Existence of good source codes, optimality criterion, Huffman coding and competitive optimality, greedy algorithm via min-max constraint, Shannon-Fano coding, run length coding, rate-distortion function and data compression of speech or image (case study), dictionary and entropy rates

Module 4 :

Lossless data compression: Variable length and fixed length (almost lossless). Linear compression. Slepian-Wolf problem. Ergodic sources: Shannon-McMillan and Birkhoff-Khintchine theorems. Basics of universal data compression. Optimality of Lempel-Ziv.

Reference Books/Material

1. T. Cover, J Thomas, "Elements of information theory", Second Edition, Wiley Press, 2006
2. R. G. Gallager, "Information theory and reliable communication", Cambridge Press , 1970
3. Ranjan Bose, "Information Theory, Coding and Cryptography", Tata McGraw Hill, 2nd edition, 2017

Course Code	Course Name	L	T	P	Credits
CS511	Optimization Techniques in Computing (OT)	3	0	0	3

Course Objective

The objective of this course is to introduce optimization techniques using both linear and non-linear programming. The focus of the course is on convex optimization though some techniques will be covered for non-convex function optimization too. After an adequate introduction to linear algebra and probability theory, students will learn to frame engineering minima maxima problems in the framework of optimization problems.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Cast engineering minima/maxima problems into optimization framework

CO2. Learn efficient computational procedures to solve optimization problems.

CO3. Use any programing language to implement important optimization methods

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							

Syllabus

Module 1 : Mathematical preliminaries

Linear algebra and matrices, Vector space, eigen analysis, Elements of probability theory, Elementary multivariable calculus.

Module 2 : Linear Programming

Introduction to linear programming model: Simplex method, Duality, Karmarkar's method, sensitivity analysis, transportation and assignment problems.

Module 3 : Unconstrained and constraint optimization

One-dimensional search methods, Gradient-based methods, Conjugate direction and quasi-Newton methods

Module 4 : Non-linear problems

Basic theory, method of Lagrange multipliers, Karush-Kuhn-Tucker theory, convex optimization

Module 5: Numerical optimization techniques

Line search methods, gradient methods, Newton's method, conjugate direction methods, quasi-Newton methods, projected gradient methods, penalty methods.

Reference Books/Material

1. N. S. Kambo, “Mathematical Programming Techniques”, East West Press, 1997.
2. E.K.P. Chong and S.H. Zak, “An Introduction to Optimization”, 2nd Ed., Wiley, 2010.

3. D. G. Luenberger and Y. Ye, “Linear and Nonlinear Programming”, 3rd Ed., Springer India, 2010.
4. D.P. Bertsekas, “Nonlinear Programming”, 2nd Ed., Athena Scientific, 1999.
5. M. S. Bazarra, H.D. Sherali, and C. M. Shetty, “Nonlinear Programming: Theory and Algorithms”, 3rd Ed., Wiley, 2006.
6. Komei Fukuda, “Introduction to Optimization”, Institute of Theoretical Computer Science, ETH Zurich, Switzerland, Autumn Semester 2011

Course Code	Course Name	L	T	P	Credits
CS 512	Soft Computing (SC)	3	0	0	3

Course Objective

The course explores the soft computing approaches to consider uncertainty that is inherent in pattern analysis tasks.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understanding of optimization problems, comprehending the fuzzy logic and the concept of fuzziness involved in various systems and fuzzy set theory

CO2. Understanding of the fundamental theory and concepts of neural networks, and Identifying different neural network architectures, algorithms, applications, and their limitations

CO3. Apply genetic algorithms and neural networks to solve real-world problems

CO4. Integrate hybrid soft computing techniques

CO5. Apply soft computing techniques to solve engineering and other societal problems.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	L	H	H	H	H	M	M	H	M
CO2	H	H	H	L	H	H	H	H	M	M	H	M
CO3	H	H	H	L	H	H	H	H	M	M	H	M
CO4	H	H	H	L	H	H	H	H	M	M	H	M
CO5	H	H	H	L	H	H	H	H	M	M	H	M

Syllabus

Module 1 :

Biological neuron, nerve structure and synapse, artificial neuron and its model, activation functions, Neural network architecture: single layer and multilayer feed forward networks, recurrent networks. Various learning techniques; perception and convergence rule, Auto-associative and hetro-associative memory.

Module 2 :

Architecture: Perceptron model, solution, single layer artificial neural network, multilayer perception model; back propagation learning methods, effect of learning rule co-efficient, back propagation algorithm, factors affecting back propagation training, applications.

Module 3 :

Basic concepts of fuzzy logic, fuzzy sets and crisp sets, fuzzy set theory and operations, properties of fuzzy sets, fuzzy and crisp relations, fuzzy to crisp conversion.

Module 4 :

Membership functions, interference in fuzzy logic, fuzzy if-then rules, fuzzy implications and fuzzy algorithms, fuzzyfication and defuzzificataion, fuzzy controller, industrial applications

Module 5 :

Genetic algorithm(GA):Basic concepts, working principle, procedures of GA, flow chart of GA, genetic representations(encoding), initialization and selection, genetic operators, mutation, generational cycle, applications.

Reference Books/Material

1. Satish Kumar, “Neural networks: A classroom approach” , III edition
2. J. S. R. Lang, C. T. Sun and E. Mizutaju “Neuro-Fuzzy and soft computing”, Pearson Education
3. C. -T. Liu and C.S. George Lee “Neural fuzzy System: A neuro fuzzy synergism to intelligent system”, PH 1996
4. V. Kecman “Learning and soft computing” MIT press 2001
5. A Ghosh, S. Dehuri and S. Ghosh(eds), “Multi-objective evolutionary algorithms for knowledge discovery from databases”, Springer 2008
6. S. Bandyopadhyay and S.K. Pal, “Classification and learning using genetic algorithms: applications in bioinformatics and web intelligence”, , SpringerVerlag, 2007
7. S. Rajsekaran& G.A. VijayalakshmiPai, “Neural networks, fuzzy logic and genetic algorithm:synthesis and applications” Prentice Hall of India, 2003

Course Code	Course Name	L	T	P	Credits
CS513	Applied Algorithms (AA)	3	0	0	3

Course Objective

The course provides an overview of some of the essential numerical techniques which are commonly used in the scientific enterprise.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand various algorithm design techniques for solving various optimization problems.

CO2. Implement string matching algorithms like Rabin-Karp and Knuth-Morris-Pratt, and understand parallel algorithm design.

CO3. Develop various advanced computational algorithms based on network flow algorithms, divide-and-conquer techniques for efficient problem-solving.

CO4. Apply graph algorithms and cryptographic techniques for security and internet-related problems.

CO5. Evaluate various randomized and approximation algorithms in computational problems.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	L	-	-	-	L	L	M	L
CO2	H	H	H	M	-	-	-	-	M	M	L	L
CO3	H	H	H	M	L	-	-	-	M	L	M	L
CO4	H	H	H	M	L	-	-	-	M	L	M	L
CO5	H	H	H	M	L	-	-	-	M	L	M	L

Syllabus

Module 1 :

Sequential algorithms: Algorithm design techniques; stable marriage problem, stable matching problem analysis and representative problems. greedy algorithms, interval scheduling, scheduling with deadlines and profits, 1/2 approximation for knapsack. Data compression: Huffman code, (KT4.8) LZ77, gzip.

Module 2 :

String Matching algorithms: Rabin-Karp algorithm, Knuth Morris pratt algorithm. Parallel algorithms: Designing parallel algorithms; combinatorial algorithms.

Module 3 :

Network flows: Bellman ford algorithm. divide-and-conquer, closest points problem. external memory algorithms, online algorithms.

Module 4 :

Graph Algorithms, internet algorithms and security- cryptography algorithms. basics of randomized algorithms. basics of approximation algorithms

Reference Books/Material

1. Alfred V Aho, John E Hopcroft, Jeffery D Ullman, “Data structure and algorithms”, Addison Wesley , 1993
2. J. Kleinberg, E. Tardos, “Algorithm design”. Pearson Education, Addison Wesley, 2006."
3. Michael Jay Quinn, “Designing efficient algorithms for parallel computers”, McGraw Hill 1997.
4. Rajeev Motwani, PrabhakarRaghavan, “Randomized algorithms”, Cambridge University Press,1995.
5. R. E. Tarjan, “Data structures and network algorithms”, SIAM, 1983. (6) Vijay V. Vazirani, “Approximation algorithms”, Springer, 2001.

Course Code	Course Name	L	T	P	Credits
CS514	Network Management(NM)	3	0	0	3

Course Objective

To appreciate the need for interoperable network management, understand general concepts and architecture behind standards based network management. Understand advanced information processing techniques such as distributed object technologies, software agents and internet technologies used for network management

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Apply network management standards and protocols to monitor, control, and optimize network devices and services

CO2. Utilize Intelligent Agents for Heterogeneous Network Management

CO3. Secure Network Infrastructures

CO4. Implement and manage QoS mechanisms in IP networks

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H							
CO2	H	H	H	M	M							
CO3	H	H	H	H	M							
CO4	H	H	H	H	H							

Syllabus

Module 1 :

Data communications and network management overview: Goals, architecture and perspectives, review of information network and technology.

Module 2 :

SNMP and network management- basic foundations: Standards, models and languages, network management organization and information models, communication and functional models.

Module 3 :

Network Management tools, systems and engineering and applications, management of heterogeneous network with intelligent agents, network security management, internet management (IEEE communication May, Oct.03).

Module 4 :

Broadband network management, wired and optical networks management, QoS in IP network, basic methods & theory for survivable network design & operation, network planning, network management standards.

Reference Books/Material

1. M. Subramanian, "Network Management: Principles and Practice", 2nd Edition, Pearson Education, 2010.
2. James F. Kurose and Keith W. Rose, "Computer Networking: A Top Down Approach", 5th Edition, Pearson Education, 2016.
3. J. Burke, "Network Management Concepts and Practice: A Hands-on Approach", Pearson Education, 2003.
4. Larry L. Peterson and Bruce S. Davie, "Computer Networks: A Systems Approach", 6th edition, Morgan Kaufmann, 2020.

Course Code	Course Name	L	T	P	Credits
CS515	Software Architecture (SA)	3	0	0	3

Course Objective

Complex software systems require abstraction and analysis at an architectural level of abstraction. In this course we study, typical software system structures.

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Understand software architectural styles.

CO2. Analyze and evaluate architectural models using tools.

CO3. Apply design patterns and styles in enterprise architectures.

CO4. Develop architectures using ADLs and component-based approaches.

CO5. Evaluate software architecture's role in engineering and product lines.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	M	L	M	L	-	-	L	L	M	L
CO2	H	H	H	M	H	-	-	-	M	L	L	L
CO3	H	H	H	M	M	-	-	-	L	L	L	-
CO4	H	H	H	M	H	-	-	-	M	L	M	-
CO5	H	H	H	M	M	-	-	-	M	L	M	-

Syllabus

Module 1 :

Typical software system structures (architectural styles), techniques for designing and implementing these structures.

Module 2 :

Models for characterizing and reasoning about architectures, and tools architectural modelling. Role of architecture in Software engineering;.

Module 3 :

Enterprise Architectures, Zachman's Framework; Architectural Styles, Design Patterns;

Module 4 :

Architecture Description Languages; Product-line architectures; Component based development.

Reference Books/Material

1. Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad, MiachelStal, Douglas Schmidt, “Pattern oriented software architecture”, Volumes 1 &2, Wiley
2. Len Bass, Paul Clements, Rick Katzman, Ken Bass, “Software architecture in practice”.2nd ed. Addison-Wesley Professional 2003
3. George T. Heineman, William T. Councill, “Component based software engineering”, Addison-Wesley, 2001
4. Kurt Wallnau, Scott Hissam and Robert Seacord, “Building systems from commercial components”, Addison-Wesley 2002

Course Code	Course Name	L	T	P	Credits
CS 516	Web Engineering(WE)	3	0	0	3

Course Objective

To apply the concepts, principles, and methods of Web engineering to Web applications development

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Understand web engineering fundamentals and methodologies for web-based system development

CO2. Develop and manage web applications with real-time features, testing, and quality assurance

CO3. Analyze and apply web metrics for performance evaluation, updates, and maintenance.

CO4. Implement user-centric development approaches considering human and cultural aspects

CO5. Integrate legacy systems with modern web technologies for seamless migration and usability

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	L	H	H	H	H	M	M	H	M
CO2	H	H	H	L	H	H	H	H	M	M	H	M
CO3	H	H	H	L	H	H	H	H	M	M	H	M
CO4	H	H	H	L	H	H	H	H	M	M	H	M
CO5	H	H	H	L	H	H	H	H	M	M	H	M

Syllabus

Module 1 :

Web Engineering Fundamentals: Requirements specification and analysis, web-based systems development methodologies and techniques, migration of legacy systems to web environments.

Module 2 :

Web-application development: Web-based real-time applications development, testing, verification and validation, quality assessment, control and assurance, configuration and project management.

Module 3 :

Web metrics: generating metrics for estimation of development efforts, performance specification and evaluation, update and maintenance.

Module 4 :

User-centric development: Development models, teams, staffing, integration with legacy systems, human and cultural aspects, user-centric development, user modeling and user involvement and feedback, end-user application development.

Reference Books/Material

1. Journal of Web Engineering, Rinton Press, IEEE & ACM Publications
2. Cato and John, " User centered web design", Pearson Education, 2001

Course Code	Course Name	L	T	P	Credits
CS 517	Software Project Management (SPM)	3	0	0	3

Course Objective

This course introduces project management as it relates to the software life cycle. Different software life cycle models and the project management activities in each phase of the life cycle are studied. Project planning activities are introduced, including effort estimation & the use of software metrics. Risk analysis and resource allocation and project scheduling. The course concludes with a project monitoring & control, project contracts & team organization.

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. To understand the basic concepts, terminologies and issues of software project management

CO2. Apply stepwise project planning, lifecycle models, and workflows in software projects.

CO3. Analyze software project management disciplines, process control, and automation.

CO4. Evaluate the cost –benefits of calculations so as to optimize the selection strategy

CO5. Design and tailor software project management processes for industry-specific needs.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	M	M	M	L	L	-	-	M	L	L	L
CO2	H	H	M	L	L	L	-	-	M	M	M	L
CO3	H	H	H	H	L	L	-	-	M	L	L	L
CO4	H	H	H	H	L	L	-	-	L	M	L	L
CO5	H	H	H	H	M	L	-	-	L	M	L	L

Syllabus

Module 1 :

Introduction, project definition, contract management, activities covered by software project management.

Module 2 :

Overview of Project planning, stepwise project planning, life cycle phases, artifacts of the process, model based software architectures, workflows of the process, check points of the process.

Module 3 :

Software management disciplines, iterative process planning, project organizations & responsibilities, process automation, project control & process instrumentation, tailoring the process.

Module 4 :

Modern project profiles, next generation software economics, modern process transitions, the state of practice in software project management, the COCOMO cost estimation model, change of metrics

Reference Books/Material

1. K. Conway, "Software project management: From concept to development", IDG Books, 2001.
2. I. Jacobson, G.Booch, J.Rumbaugh, "The unified software development" Process, Addison Wesley, 1999.
3. Stephan H.Kin, "Metric and models in software quality engineering", Addison Wesley 1995.
4. Walker Royce, "Software Project Management", Addison Wesley,1998.
5. Pankaj Jalote, "Software Project Management in Practice", Pearson Education Inc. Delhi, 2002

Course Code	Course Name	L	T	P	Credits
CS518	Advanced Compilers (AC)	3	0	0	3

Course Objective

Complex software systems require abstraction and analysis at an architectural level of abstraction. In this course we study, typical software system structures.

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Understand fundamentals of compiler design and analysis of graph structures for data and flow control

CO2. Analyze different optimization techniques in compiler and optimizing for memory hierarchies

CO3. Understand the concept such as high performance systems, scalar, vector, multiprocessor, SIMD, message passing architectures

CO4. Analyze high performance systems, scalar, vector, multiprocessor, SIMD, message passing architectures.

CO5. Understand dependence system, GCD test, Banerj's Inequality, exact algorithm, vectorization, concurrentization, array region analysis, loop restructuring transformations

CO6. Analyze dependence system, GCD test, Banerj's Inequality, exact algorithm, vectorization, concurrentization, array region analysis, loop restructuring transformations

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L	M	M	L	M	L	L	L	M	L	L
CO2	L	M	M	M	L	M	M	L	L	M	L	M
CO3	M	M	M	M	L	M	M	L	M	M	M	M
CO4	M	M	M	M	M	M	M	L	M	M	M	M
CO5	M	M	H	M	L	M	M	L	M	M	M	M
CO6	M	M	H	M	L	M	M	L	M	M	M	M

Syllabus

Module 1 :

Overview of compiler design, optimizing compilers, graph structures for control flow analysis of programs, data flow analysis of programs, static single assignment form, data dependence of program, program dependence graph.

Module 2 :

Scalar optimization, loop optimizations,register allocation, instruction scheduling, local methods, graph colouring, code scheduling software pipelining, inter procedural dataflow analysis, optimizing for memory hierarchies.

Module 3 :

High performance systems, scalar, vector, multiprocessor, SIMD, message passing architectures. sequential and parallel loops, data dependence use-def chains

Module 4 :

Dependence system, GCD test, Banerjee's Inequality, exact algorithm, vectorization, concurrentization, array region analysis, loop restructuring transformations

Reference Books/Material

1. Robert “Building an Optimizing Compiler Morgan”,Digital Press, 1998.
2. M. Wolfe, “High Performance Compilers for Parallel Computing”, AddisonWesley, 1996.

3. Steven S. Muchnick, “Advanced Compiler Design and Implementation”, Morgan Kaufmann Publishers, 1997.
4. R. Allen and K. Kennedy, “Optimizing Compilers for Modern Architectures”, Morgan Kaufmann Publishers, 2003.
5. A. Appel,Press, “Modern Compiler Implementation in C”, 1998.
6. A. Aho, M. Lam, R. Sethi and J. Ullman "Compilers: Principles, Techniques, and Tools", 2007.
7. Steven S. Muchnick, “Advanced Compiler Design and Implementation”, Morgan Kaufmann, Elsevier Science, 2003 (8) Michael Wolfe, “High Performance Compilers for Parallel Computing”, Addison Wesley, 1995.

Course Code	Course Name	L	T	P	Credits
CS 519	Computer Vision (CV)	3	0	0	3

Course Objective

The objective of this course is to understand the basic issues in computer vision and major approaches that address them. Even though Computer Vision is being used for many practical applications today, it is still not a solved problem. Hence, definitive solutions are available only rarely. Course also covers the deep learning based approaches to computer vision tasks.

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Understand the fundamental image processing techniques needed for computer viison
CO2. Implement fundamental image processing techniques required for computer vision
CO3. Understand the contribution of deep learning for computer vision
CO4. Implement the complex computer vision tasks using deep learning technique

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	L							
CO2	M	H	H	M	H							
CO3	H	M	M	L	H							
CO4	H	H	H	L	M							

Syllabus

Module 1 :

Introduction and overview, pinhole cameras, radiometry terminology. Sources, shadows and shading: Local shading models- point, line and area sources; photometric stereo. Color: Physics of color; human color perception, Representing color; A model for image color; surface color from image color.

Module 2 :

Linear filters: Linear filters and convolution; shift invariant linear systems- discrete convolution, continuous convolution, edge effects in discrete convolution; Spatial frequency and Fourier transforms; Sampling and aliasing; filters as templates; Normalized correlations and finding patterns. Edge detection: Noise; estimating derivatives; detecting edges. Texture: Representing texture; Analysis using oriented pyramid; Applications; Shape from texture. The geometry and views: Two views.

Module 3 :

Stereopsis: Reconstruction; human stereo; Binocular fusion; using color camera.

Segmentation by clustering: Human vision, applications, segmentation by graph theoretic clustering. Segmentation by fitting a model, Hough transform; fitting lines, fitting curves;

Module 4 :

Deep learning architectures for computer vision: Convolution Neural Networks, Recurrent Neural networks, LSTMs, GRU etc.

Deep Learning Models for computer vision tasks: Object Classification-VGGNET, RESNET, ALEXNET, DENSENET, EFFICIENT NET, MOBILENET, INCEPTION V3, Object Detection-R-CNN, F-RCN, SSD, Retinanet, YOLO, CornerNet, Image Segmentation- U-Net, SegNet, Mask-RCNN, Attention Models-Transformers

Reference Books/Material

1. David A Forsynth and Jean Ponce, "Computer vision- A modern approach", Pearson education series, 2003.
2. Milan Sonka, Vaclav Hlavac and Roger Boyle , "Digital image processing and computer vision", Cengagelearning, 2008.
3. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep learning, MIT Press, 2016
4. Charu C. Aggarwal, Neural Networks and Deep Learning, Springer, 2018
5. Christopher M. Bishop, Hugh Bishop, "Deep Learning, Foundations and concepts", First Edition, 2023, Springer
6. Schalkoff R. J., "Digital image processing and computer vision", John Wiley, 2004.
7. Sonka M., Hlavac V., Boyle R., "Image processing analysis and machine design". PWS Publishers
8. Ballard D., Brown C., "Computer vision", Prentice Hall

Course Code	Course Name	L	T	P	Credits
CS520	Artificial Intelligence (AI)	3	0	0	3

Course Objective

The objective of this course is to introduce concepts and implementation of AI algorithms. By the end of this course, students will be able to understand AI fundamental, significance of AI. Also, the understanding of knowledge bases, planning and reasoning systems.

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Develop Problem-Solving and Search Strategies. Implement various search algorithms such as BFS, DFS, A*, and Minimax. Understanding of the complex problem in first order predicate logic and inference rules

CO2. Understanding of adversarial search and Games

CO3. Understand supervised, unsupervised learning models. Develop and evaluate machine learning models using Python frameworks. Apply Reinforcement Learning Methods, i.e., Understand Markov Decision Processes (MDP) and Q-learning

CO4. Work with Deep Learning and Neural Networks. Understand CNNs for image recognition and RNNs for sequence modeling.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

Module 1 : Problem solving and Reasoning

Solving Problems by Searching, Search in Complex Environments: Uninformed Search Strategies (BFS, DFS, Uniform Cost Search), Informed (Heuristic) Search Strategies (A*, Greedy Best-First Search), Adversarial Search Minimax Algorithm, Alpha-Beta Pruning), Propositional and First-Order Logic Inference in First-Order Logic

Module 2 : Uncertain Knowledge Representation

Quantifying Uncertainty, Probabilistic Reasoning: Bayesian Networks, Markov Models, and Games: Game Theory, Optimal Decisions in Games, Multiagent Decision Making (Non-Cooperative Game Theory, Cooperative Game Theory).

Module 3 : Deep Learning:

Neural Networks and Perceptrons, Backpropagation and Gradient Descent, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) and LSTMs, Transformers and Attention Mechanisms.

Module 4 : Reinforcement Learning (RL)

Introduction, Components of RL, Learning from Rewards, Active and Passive Reinforcement Learning, Policy search: Policy Evaluation, Policy Iteration, Value Iteration, Evaluation Metrics (Accuracy, Precision, Recall, F1-Score).

Reference Books/Material

1. Russell, Stuart J., and Peter Norvig. Artificial intelligence: a modern approach. pearson, 2016.
2. Deepak Khemani, "A first course in artificial intelligence", Tata McGraw Hill,2013
3. Bishop, Christopher M. Pattern recognition and machine learning by Christopher M. Bishop. Springer Science+ Business Media, LLC, 2006
4. Goodfellow, AC Ian. "Deep learning-ian goodfellow, yoshua bengio, aaron courville-google books." 2016
5. Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. Vol. 1. No. 1, Cambridge: MIT press, 1998
6. Koller, Daphne, and Nir Friedman. Probabilistic graphical models: principles and techniques. MIT press, 2009

Course Code	Course Name	L	T	P	Credits
CS521	Multimedia & Virtual Reality (MVR)	3	0	0	3

Course Objective

To provide basic knowledge of multimedia and overview of the tools & taxonomy of multimedia authoring, Including data representation for images, video & audio. To understand data compression & multimedia communication & retrieval

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Understand the basic principles of sound, sound cards, and audio manipulation in multimedia systems

CO2. Perform digital image processing and editing, and utilize computer animation techniques effectively

CO3. Apply animation software to create basic computer-generated animations

CO4. Define and understand the concept of virtual reality

CO5. Explore the use of virtual reality

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

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

Syllabus

Module 1 :

Introduction to multimedia technology and its applications, multimedia hardware and software essentials. multimedia graphics fundamentals. multimedia audio - sound card fundamentals

Module 2 :

MIDI fundamentals: digital video production techniques, image processing - digital image fundamentals, digital image development and editing, computer animation techniques, animation software. multimedia file formats – growth pace of multimedia in IT industry.

Module 3 :

Concepts of virtual reality and its effectiveness in real time applications, virtual reality tools, introduction to scientific visualization and virtual reality, hardware requirements, sound, animation techniques, VR on flight simulation.

Module 4 :

VR on CAD / CAM processing : Virtual banks, compression and decompression techniques, CASE study of multimedia workstations

Reference Books/Material

1. Mark S Drew and Ze Nian Li, "Fundamentals of Multimedia", Pearson Education.
2. Tay Vaughan, "Multimedia: Making it Work", 9th edition, McGraw-Hill, 2014.
3. Steve Aukstakalnis, "Practical Augmented Reality", Addison-Wesley, 2016.
4. Tony Parisi, "Learning Virtual Reality", O'Reilly Media, 2015.

Course Code	Course Name	L	T	P	Credits
CS522	Protocol Engineering(PE)	3	0	0	3

Course Objective

Characterize protocol engineering. Compare and contrast various Internet protocols such as TCP/IP, DNS, DHCP, LDAP, and IPsec.

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Apply life cycle models, top-down and bottom-up approaches to the design and development of network protocols

CO2. Identify and describe service specifications, service data units (SDU), service elements, and communication modes in protocol specification

CO3. Design protocol data formats

CO4. Apply theoretical principles to real-world protocol data formats

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	M							
CO2	H	H	H	H	H							
CO3	H	H	H	M	M							
CO4	H	H	H	H	H							

Syllabus

Module 1 :

Review of Communication Network: Overview of computer network protocol, OSI reference model, Basic design concept: Protocol as a system, life cycle model, architectural design phase,top down approach ,bottom up approach ,separation of concern.

Module 2 :

Requirement specification: service specification service data unit service elements, communication mode, Protocol architecture:Basic protocol concept, protocol layer, protocol entity, protocol element protocol data unit.

Module 3 :

Protocol structuring, design and specification protocol structuring, the users of pdu service structuring, generic protocol function, five elements of protocol specification, rules of design, specification language, message sequence chart, petri net finite state machine

Module 4 :

Protocol Data Format: Abstract Syntax format design principles, ASN.1, ASN.1 record structure ASN.1 encoding rule, XML Syntax, DTD and XML schemas example, Case of protocol data format customer information: XML-based customer information, ASN.1 binary-encoded based XML schema and ASN.1 cooperation.

Reference Books/Material

1. Andrew S. Tanenbaum and David J. Wetherall, "Computer Networks", 5th Edition, Pearson, 2010.
2. William Stallings, "Data and Computer Communications", 10th Edition, Pearson, 2013.
3. Pallapa Venkataram, Sunilkumar S. Manvi, B. Sathish Babu, "Communication Protocol Engineering", 2nd Edition, PHI, 2014.

Course Code	Course Name	L	T	P	Credits
CS523	Software Testing (ST)	3	0	0	3

Course Objective

To discuss the distinctions between validation tests and defect testing. To describe strategies for generating system test cases. To gain the techniques and skills on how to use modern software testing tools to support software testing projects

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Understand software testing principles and the role of defects in software quality.

CO2. Design effective test cases using black-box and white-box testing techniques

CO3. Implement various testing levels, including unit, integration, system, and regression testing.

CO4. Develop test plans and strategies for efficient software validation and quality assurance

CO5. Monitor and control testing processes using review techniques, metrics, and reporting methods

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H	H	L	M	M	M	H	M
CO2	H	H	H	H	H	H	L	M	M	M	H	M

CO3	H	H	H	H	H	H	L	M	M	M	H	M
CO4	H	H	H	H	H	H	L	M	M	M	H	M
CO5	H	H	H	H	H	H	L	M	M	M	H	M

Syllabus

Module 1 :

Testing as an engineering activity, role of process in software quality, testing as a process, basic definitions, software testing principles, the tester's role in a software development organization, origins of defects, defect classes, the defect repository and test design, defect examples, developer / tester support for developing a defect repository.

Module 2 :

Introduction to testing design strategies, the smarter tester, test case design strategies, using black box approach to test case design, random testing, equivalence class partitioning, boundary value analysis, other black box test design approaches, black box testing and cots, using white box approach to test design, test adequacy criteria, coverage and control flow graphs, covering code logic, paths, their role in white box based test design – additional white box test design approaches, evaluating test adequacy criteria.

Module 3 :

The need for levels of testing, unit test, unit test planning, designing the unit tests, the class as a testable unit, the test harness, running the unit tests and recording results, integration tests, designing integration tests, integration test planning, system test, the different types, regression testing, alpha, beta and acceptance tests.

Module 4 :

Basic concepts, testing and debugging goals and policies, test planning, test plan components, test plan attachments, locating test items, reporting test results, the role of three groups in test planning and policy development, process and the engineering disciplines, introducing the test specialist, skills needed by a test specialist, building a testing group.

Module 5 :

Defining terms, measurements and milestones for controlling and monitoring, status meetings, reports and control issues, criteria for test completion, scm, types of reviews, developing a review program, components of review plans, reporting review results.

Reference Books/Material

1. Glenford J. Myers, "The art of software testing", John Wiley & Sons, 1979.
2. Boris Beizer, Black "Testing: Techniques for functional testing of software and systems", John Wiley & Sons, 1995.
3. William Perry, "Software testing: Effective methods for software testing", John Wiley, 1995.
4. Cem Kaner, Jack Falk, Hung Quoc Nguyen, "Testing computer software", 2nd Ed, Intl.

Course Code	Course Name	L	T	P	Credits
CS524	Mobile Communications (MC)	3	0	0	3

Course Objective

By the end of this course, students will be able to Understand the Fundamentals of Mobile Communication. Explain the evolution of mobile communication from 1G to 6G. Understand cellular concepts, including frequency reuse, handoff, and roaming

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Explain the evolution of mobile communication from 1G to 6G. Understand cellular concepts, including frequency reuse, handoff, and roaming.

CO2. Understand the 5G New Radio (NR) architecture and its enhancements over LTE. Describe the 5G Core Network (5GC) and its service-based architecture (SBA), and role of Software-Defined Networking (SDN) and Network Function Virtualization (NFV) in 5G.

CO3. Analyze how AI/ML optimizes 5G networks, including traffic prediction and resource allocation.

CO4. Study 5G-enabled smart cities, IoT, and industrial automation. Analyze the role of Artificial Intelligence (AI) and Quantum Communication in 6G networks. Explore AI for next-generation networks.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

Introduction to Mobile Communication and Cellular Network Architecture: Evolution of Mobile Communication (1G to 6G), Basics of Wireless Communication, Cellular Concept & Frequency Reuse, Handoff & Roaming. Electromagnetic Spectrum & Propagation, Multiplexing (FDMA, TDMA, CDMA, OFDMA), spread Spectrum Techniques (FHSS, DSSS). Mobile Radio Propagation (Path Loss, Fading, Shadowing), Channel Allocation Strategies, Call Processing & Handoff Mechanisms, Interference & Capacity Planning. GSM & GPRS Networks. 3G & 4G LTE Networks.

5G Core Network and Protocols: Introduction to 5G Communication, 5G Network Architecture, 5G Spectrum and Frequency Bands, 5G Radio Access Technologies. key features and requirements of 5G, such as enhanced Mobile Broadband (eMBB), Ultra-Reliable Low Latency Communication (URLLC), and Massive Machine-Type Communication (mMTC).Service-Based Architecture (SBA), Network Slicing in 5G Multi-Access Edge Computing (MEC), Network Function Virtualization (NFV) and Software-Defined Networking (SDN).

AI/ML for 5G Communication: Need for AI/ML in 5G Networks, AI for Dynamic Resource Allocation, AI in 5G Network Optimization, AI/ML in 5G Core Networks: AI for 5G Core

Network Traffic Prediction & Optimization, AI in Multi-Access Edge Computing (MEC), Reinforcement Learning for Handoff & Mobility Management

5G use cases and Future Trends Beyond 5G (6G): Case Studies on AI-Enabled 5G Deployment AI-driven Smart Cities and IoT, 5G and AI for Autonomous Vehicles and V2X Communication, AI in 5G Healthcare (Remote Surgery, Telemedicine), AI-Industrial Automation in 5G, AI in 6G Networks: Intelligent Network Evolution, Quantum AI for Wireless Communication, AI-powered Edge Computing & Federated Learning in 6G,

Reference Books/Material

1. Rodriguez, Jonathan. *Fundamentals of 5G mobile networks*. John Wiley & Sons, 2015.
2. Dahlman, Erik, Stefan Parkvall, and Johan Skold. *5G NR: The next generation wireless access technology*. Academic Press, 2020.
3. Marsch, Patrick, et al., eds. *5G system design: architectural and functional considerations and long term research*. John Wiley & Sons, 2018.
4. Eldar, Yonina C., et al., eds. *Machine learning and wireless communications*. Cambridge University Press, 2022.

Course Code	Course Name	L	T	P	Credits
CS525	Network Security(NS)	3	0	0	3

Course Objective

To provide extensive, detailed and critical understanding of the concepts, issues, principles and theories of network security.

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Understand different attacks at network level and their root cause

CO2. Analyze different solutions to network security

CO3. Analyze the working of different security protocols at network layers

CO4. Explore newer solutions for addressing the current and future challenges

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L		M				L					
CO2	M	L		L	L				M			
CO3	M	M	L		M	L		M			M	
CO4	H	H		M		L				M		L

Syllabus

Module 1 :

Introduction to network security and associated techniques, Firewall design principles: Packet filtering, Gateways: Circuit-level gateways; application-level gateways,

Module 2 :

Firewall Configurations, Intrusion Control: Detection; Anomaly-Based IDS Intrusion Recovery; Vulnerability Scanners; Login, Audit, and Sniffers,

Module 3 :

Communication Security Network Access Layer;- Internet Layer - Transport Layer;

Module 4 :

Application Layer - Message Security Risk Analysis, Policies, Procedures and Enforcement. Special Topics : DOS Mitigation ,VPNs Special Topics: Viruses, SPAM. Network protocols and vulnerabilities, Network defenses, Denial of service attacks, Malware.

Reference Books/Material

1. C. Kaufman, R. Perlman, M. Speciner, “Network security: Private communication in a public world”, Prentice Hall, 2002.
2. William Stallings, “Network security essentials”, 2/e, Pearson Education, 2003.

Course Code	Course Name	L	T	P	Credits
CS526	Parallel Algorithms (PA)	3	0	0	3

Course Objective

To introduce techniques for the design of efficient parallel algorithms and their implementation.

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Analyze parallel models and measure parallel algorithm complexity for optimization.

CO2. Implement parallel design techniques like divide and conquer, pipelining, partitioning.

CO3. Develop parallel algorithms for lists, trees, sorting, and searching problems.

CO4. Design efficient parallel graph algorithms for traversal, connectivity, and paths.

CO5. Apply parallel computing techniques to polynomial and matrix computations efficiently

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H			M		L		
CO2	H	H	H	H	H			M		L		
CO3	H	H	H	H	H			M		L		
CO4	H	H	H	H	H			M		M		
CO5	H	H	H	H	H			M		L		

Syllabus

Module 1 :

Parallel processing, parallel models, performance of parallel algorithms, complexity measure for parallel algorithms.

Module 2 :

Techniques for designing parallel algorithms, pointer jumping technique, divide and conquer, partitioning strategy, pipelining, accelerated cascading, symmetry breaking.

Module 3 :

Lists and trees, list ranking, Euler-tour technique, Tree contraction, computation of tree functions, merging, parallel sorting algorithms. parallel combinatorial algorithms: permutations with and without repetitions combinations, derangements. parallel searching algorithms: maximum/minimum, median, K-th largest/smallest element.

Module 4 :

Parallel graph algorithms, parallel graph search &, tree traversal algorithms, parallel algorithms for connectivity problems, parallel algorithms for path problems., Ear decomposition, Polynomial and matrix computations, General dense matrices.

Reference Books/Material

1. Jaja, J. "An introduction to parallel algorithms", Addison- Wesley, Reading, MA, 1992.
2. Gibbons A., W.Rytter, "Efficient parallel algorithms", Cambridge university Press; Cambridge, 1988
3. H. Sparkias and A. Gibbon, "Lecture notes on parallel computation", Cambridge University Press, 1993.
4. K. Hwang and F. A. Briggs, "Computer architecture and parallel processing", McGraw Hill Inc., 1985.
5. S. Akl., "Design and analysis of parallel algorithms", Prentice Hall Inc, 1992

Course Code	Course Name	L	T	P	Credits
CS527	Distributed Algorithms (DA)	3	0	0	3

Course Objective

To introduce the main algorithmic techniques in the framework of distributed models of computing; to define the most significant complexity parameters and the computational limits of parallelism and concurrency.

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Analyze distributed algorithm models and complexity measures for efficient computation

CO2. Implement leader election algorithms in rings and general networks.

CO3. Develop consensus algorithms handling stopping and Byzantine process failures

CO4. Design asynchronous network algorithms using shared memory for agreement

CO5. Apply synchronizers and logical time for fault-tolerant distributed systems

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

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

Syllabus

Module 1 :

Distributed Algorithms: models and complexity measures. Modeling: Synchronous network model, asynchronous system model,asynchronous shared memory model, asynchronous network model. partially synchronous system model.

Module 2 :

Leader election in synchronous ring: Basic algorithm, non-comparison based algorithm, time slice and variablespeeds algorithm. Lower bounds on the algorithms. Leader election in a general network.

Module 3 :

Distributed consensus with process failures: Algorithms for stopping failures, algorithms for byzantine failures. approximate agreement.

Module 4 :

Consensus: Agreement using read/write shared memory. Basic asynchronous network algorithms: Leader election in a ring algorithms, leader election in arbitrary network.

Module 5 :

Synchonizers, safe synchronizer implementations. algorithm tolerating process failures. adding logical time to asynchronous networks. applications. termination detection for diffusing algorithms. The chandylamport algorithms, mutual exclusion , general resource allocation algorithms .

Reference Books/Material

1. Nancy & Lynch, Distributed Algorithms, Harcour Asia, 2001.

Course Code	Course Name	L	T	P	Credits
CS528	Computer Security Audit and Assurance (CSAA)	3	0	0	3

Course Objective

To introduce students to the concepts of Information Assurance and how to secure such information using appropriate systems and technologies, presenting introductory aspects on computer audit including auditing information systems auditing computerized systems, auditing applications etc. Also, to introduce students to the key management and Public Key Infrastructure.

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Understand security audit requirements and its fundamentals

CO2. Analyze different systems and technologies to security audit and compliance frameworks

CO3. Explore computer and network security and related aspects

CO4. Develop newer solutions with technological advancements

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M			M			L					
CO2		M				L		M			L	M
CO3	L		M		L		M			M		
CO4	H	M				M			M		L	

Syllabus

Module 1 :

Security policy frameworks; practices and procedures, business practice disclosures. Information Systems in Global Context · Threats to Information Systems · Security Considerations in Mobile and Wireless Computing · Information Security Management in Organizations · Building Blocks of Information Security · Information Security Risk Analysis · Overview of Physical Security for Information Systems · Perimeter Security for Physical Protection · Biometrics Controls for Security · Biometrics-based Security: Issues and Challenges · Network Security in Perspective.

Module 2 :

·Networking and Digital Communication Fundamentals · Cryptography and Encryption · Intrusion Detection for Securing the Networks · Firewalls for Network Protection · Virtual Private Networks for Security · Security of Wireless Networks · Business Applications Security: An EAI Perspective · Security of Electronic Mail Systems · Security of Databases · Security of Operating Systems · Security Models, Frameworks, Standards and Methodologies · ISO 17799/ISO 27001 · Systems Security Engineering Capability Maturity Model - The SSE-CMM · COBIT, COSOERM and SAS 70.

Module 3 :

Information Security: Other Models and Methodologies · Laws and Legal Framework for Information Security · Security Metrics · Privacy - Fundamental Concepts and Principles · Privacy - Business Challenges · Privacy - Technological Impacts · Web Services and Privacy · Staffing the Security Function · Business Continuity and Disaster Recovery Planning. Policy

authority and practices, information security practices, personal and physical security practices, operation management practices .

Module 4 :

Auditing for Security · Privacy Best Practices in Organizations · Asset Management · Ethical Issues and Intellectual Property Concerns for InfoSec Professionals. PKI's and key management schemes, key generation, key storage, backup, recovery and distribution. XML frameworks for security policy specification, certificate management life cycle.

Reference Books/Material

1. W K Brotby, Information security management metrics, CRC press 2009.
2. Nina Godbole, Information systems security: security management, metrics, frameworks and best practices, John Wiley and sons Ltd. 2009.

Course Code	Course Name	L	T	P	Credits
CS529	Big Data Analysis (BDA)	3	0	0	3

Course Objective

This course focusses on understanding the Big Data Platform and its Use cases. The course involves an overview of Apache Hadoop. Further the security and privacy aspects are also discussed.

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Understand Big Data and its analytics in the real world

CO2. Analyze the Big Data framework like Hadoop to efficiently store and process Big Data to generate analytics

CO3. Design of Algorithms to solve Data Intensive Problems using Map Reduce Paradigm

CO4. Design and Implementation of Big Data Analytics to solve data intensive problems and to generate analytics

CO5. Analyze Security and Privacy issues

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	M	M	L	M	M	L	L	M	L	M	M
CO2	H	M	H	M	M	M	L	M	M	M	M	M
CO3	M	M	H	M	M	M	L	M	M	M	M	M
CO4	M	M	H	M	M	M	M	L	M	M	M	M
CO5	M	M	M	M	M	M	H	M	M	M	M	M

Syllabus

Module 1 :

Overview of big data, stages of analytical evolution, state of the practice in analytics, the data scientist.

Module 2 :

Big data analytics in industry verticals, data analytics lifecycle, operationalizing basic data analytic methods using R, advanced analytics - analytics for unstructured data - map reduce and Hadoop, the Hadoop ecosystem, in-database analytics.

Module 3 :

Data Visualization Techniques, Stream Computing Challenges, Systems architecture, Main memory data management techniques, energy-efficient data processing, benchmarking.

Module 4 :

Security and Privacy, Failover and reliability.

Reference Books/Material

1. Bill Franks, Taming, “The big data tidal wave”, 1st ed., Wiley, 2012
2. Frank J. Ohlhorst, “Big data analytics”, 1st ed., Wiley, 2012.

Course Code	Course Name	L	T	P	Credits
CS530	Secure Software Engineering (SSE)	3	0	0	3

Course Objective

This course focuses on secure software engineering process and details the secure programming and software security.

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Understand software security threats, vulnerabilities, and risk management strategies effectively

CO2. Apply secure design principles, patterns, and programming techniques efficiently

CO3. Utilize cryptographic techniques, code reviews, and static analysis securely

CO4. Implement secure testing methods and software security program development]

CO5. Develop comprehensive security measures to protect software from threats

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H			M		L		
CO2	H	H	H	H	H			M		L		
CO3	H	H	H	H	H			M		L		
CO4	H	H	H	H	H			M		M		
CO5	H	H	H	H	H			H		H		

Syllabus

Module 1 :

Definition of software security, threats and vulnerabilities, risk management, security requirements.

Module 2 :

Principles of secure design and patterns, secure programming, validation of the data

Module 3 :

Secure usage of cryptography, code reviews and static analysis.

Module 4 :

Secure testing, creating a software security programs.

Reference Books/Material

1. Julia H Allen, Sean J Barnum, Robert J Ellison, Gary McGraw, Nancy M Read, "Software Security Engineering: A Guide to Project Managers", Addison Wesley, 2008.
2. Ross J Anderson, "Security Engineering: A Guide to Building Dependable Distributed Systems", Wiley, 2008.
3. Howard M and LeBlanc D, "Writing Secure Code", Microsoft Press, 2003.

Course Code	Course Name	L	T	P	Credits
CS 531	Computer Graphics (CG)	3	0	0	3

Course Objective

To have an introduction to computer graphics to develop abilities to comprehend contemporary issues and address them

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Obtain insight into the basics of computer graphics and multimedia

CO2. Analyze and design solutions to problems pertaining to graphics and multimedia

CO3. Implement various 2D and 3D objects transformation techniques and apply viewing technologies into real-world applications

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	M	H	M	H							
CO2	M	H	M	L	M							
CO3	M	H	M	M	M							

Syllabus

Module 1 :

Introduction to graphics hardware devices, display devices, primitive operations, the display-file interpreter, display file structure, and graphics file formats. text mode graphics function, graphic mode graphics functions shapes, colors, co-ordinate systems, applications of computer graphics.

Module 2 :

Basic concepts in line drawing, line drawing algorithms: DDAalgorithms, Bresenham's algorithm Circle generating algorithms: DDA circle drawing algorithm, Bresenham's circle drawing algorithm, midpoint circle algorithm, polygons, types of polygons, polygon representation, entering polygons, inside –outside test, polygon filling: Flood fill, scan-line algorithm.

Module 3 :

2D transformation: scaling, Reflection, shearing, Rotation, Translation, Rotation about an arbitrary point. 3D Transformation: scaling, rotation, translation, rotation about arbitrary axis. Viewing transformation, normalization, transformation. Line clipping: Cohen-Sutherland, Line

clipping algorithm, midpoint subdivision algorithm Polygon clipping: Sutherland–Hodgeman Polygon clipping algorithm.

Module 4 :

Curve generation: arc generation using DDA algorithm. Interpolation, B-Spline, Bezier curves. Fractals: Hilbert's Curve, Koch curve, Fractal lines, Fractal Surfaces. Raster scan display, Random scan display Need for graphics standards, Graphics standards, Advantages of Graphics standards, Hazards of Graphics standards. Graphical user interface Open GL: What is Open GL, How OpenGL works, Open GL and animation, Graphical processors: GPUs.

Reference Books/Material

1. Ronald Hearn &MPauline Baker, "Computer graphics", 2nd ed., PES, 2003.
2. James D. Foley, Andrews van Dam, Steven K Feimer, John F Hughes, "Computer graphics principles and practice", 2nd ed., Addison Wesley, 1996.
3. William Newman and Robert Sproull, "Principles of Interactive Computer Graphics", Tata McGraw-Hill,1973.

Course Code	Course Name	L	T	P	Credits
CS 532	Graph Theory (GT)	3	0	0	3

Course Objective

This is an introductory course about properties and applications of graphs. It aims at the usage of graph theoretic methods for modeling problems and proofs in discrete mathematics.

Course Outcomes

Upon successful completion of the course, the student must be able to:

CO1. Apply graph representation techniques

CO2. Implement and apply graph traversal algorithms

CO3. Apply graph coloring techniques

CO4. Use graph theory to solve real-world problems such as network design, resource allocation problems etc.

CO5. Identify which graph algorithm or technique is best suited for a given real-world problem by analyzing the graph's structure and constraints.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

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

Syllabus

Module 1 :

Definitions, pictorial representation of a graph, isomorphic graphs, sub graphs, matrix representations of graphs, degree of a vertex, special graphs, complements, larger graphs from smaller graphs, connected graphs and shortest paths, walks, trails, paths, cycles, connected graphs, cut-vertices and cut-edges, blocks, connectivity, weighted graphs and shortest paths, weighted graphs, Dijkstra's shortest path algorithm, Floyd-Warshall shortest path algorithm.

Module 2 :

Trees, Definitions and characterizations, number of trees, Cayley's formula, minimum spanning trees, Kruskal's algorithm, Prim's algorithm, bipartite graphs, Eulerian graphs, Fleury's algorithm, Chinese Postman problem.

Module 3 :

Hamilton Graphs, necessary conditions and sufficient conditions, independent sets, coverings and matchings, matchings in bipartite graphs, Hall's theorem, Konig's theorem, perfect matching's in graphs, vertex Colorings, basic definitions, cliques and chromatic number, greedy coloring algorithm

Module 4 :

Edge colorings, Gupta-Vizing theorem, class-1 and class-2 graphs, edge-coloring of bipartite, graphs, planar graphs, basic concepts, Euler's formula and its consequences, characterizations of planar graphs, 5-color-theorem, directed graphs, directed walks, paths and cycles, Eulerian and Hamilton digraphs.

Reference Books/Material

1. Adrian Bondy, U. S. R. Murty, "Graph Theory", Springer, 2008.
2. Reinhard Diestel, "Graph Theory", 5th Edition, Springer, 2016.
3. Douglas B. West, "Introduction to Graph Theory", Pearson, 2000.

Course Code	Course Name	L	T	P	Credits
CS 533	Distributed Computing Systems (DCS)	3	0	0	3

Course Objective

This course covers abstractions and implementation techniques for the design of distributed systems. It focuses on server design, network programming, naming, storage systems, security, and fault tolerance.

Course Outcomes

Upon successful completion of the course, the student must be able to

CO1. Analyze and implement message-passing mechanisms
CO2. Design and apply synchronization techniques in distributed systems
CO3. Solve problems using distributed algorithms
CO4. Develop fault-tolerant distributed systems
CO5. Implement and analyze distributed databases and file systems

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

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

Syllabus

Module 1 :

Introduction Distributed Systems and applications, Distributed vs parallel systems, models of distributed systems, Message Passing mechanisms IPC and RPC.

Module 2 :

Clock synchronization, physical & logical clocks, vector clocks, verifying clock algorithms, mutual exclusion using time stamp, election algorithms, Distributed mutual exclusion using time stamps, token & quorums, centralized & distributed algorithms, proof of correctness & complexity, drinking philosophers problem, Implementation & performance evaluation of DME Algorithms.

Module 3 :

Leader election algorithms, global states, global predicates, termination detection, Control of distributed computation, disjunctive predicates, performance evaluation of leader election algorithms on simulated environments.

Module 4 :

Distributed File Systems and Services, Shared data, Synchronization Transaction and Concurrency Control. Distributed databases, Name service, Timing & Coordination, Replication, Security and Fault Tolerance.

Reference Books/Material

1. Vijay K Garg "Elements of Distributed Computing", 1st Edition, Wiley, 2002
2. Pradeep Sinha, "Distributed Operating Systems- Concepts and Design", PHI, 2007
3. A.S. Tanenbaum and M.V. Steen, "Distributed Systems -Principles and Paradigms", 2nd Edition, PHI, 2016
4. George Couloris, Jean Dollimore, Tim Kindberg, and Gordon Blair, "Distributed Systems: Concepts & Design", 5th Edition, Pearson, 2017

Course Code	Course Name	L	T	P	Credits
CS534	Virtualization and Cloud Computing	3	0	0	3

Course Objective

The objectives of this course are to provide an in-depth and comprehensive knowledge of the Cloud Computing fundamental issues, technologies, applications and implementations, to understand the underlying principle of cloud virtualization, cloud storage, data management and data visualization, to expose the students to the frontier areas of Cloud Computing, to motivate students to do programming and experiment with the various cloud computing environments, to shed light on the Security issues in Cloud Computing and to introduce about the Cloud Standards.

Course Outcomes

At the completion of this course, the student will be able to

CO1. Articulate the main concepts, key technologies, strengths, and limitations of cloud computing and the possible applications for state-of-the-art cloud computing

CO2. Design and develop highly scalable cloud-based applications by creating and configuring virtual machines on the cloud and building private cloud

CO3. Explain the core issues of cloud computing such as security, privacy, and interoperability

CO4. Provide the appropriate cloud computing solutions and recommendations according to the applications used

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

History of Centralized and Distributed Computing - Overview of Distributed Computing, Cluster computing, Grid computing

Technologies for Network based systems- System models for Distributed and cloud computing- Software environments for distributed systems and clouds

Introduction to Cloud Computing- Cloud issues and challenges - Properties - Characteristics - Service models, Deployment models. Cloud resources: Network and API - Virtual and Physical computational resources - Data-storage

Virtualization concepts - Types of Virtualization- Introduction to Various Hypervisors, Different approaches to virtualization, Hypervisors Machine Image Virtual Machine (VM), Process VM vs System VM, Resource Virtualization: Server, Storage, Network Full Virtualization vs Para Virtualization, Operating System Support for Virtualization, Virtual Machine (resource) Provisioning and Manageability VM Placement, VM Migration, High Availability (HA)/Disaster Recovery (DR) using Virtualization, Moving VMs

Service models - Infrastructure as a Service (IaaS) - Resource Virtualization: Server, Storage, Network - Case studies. Platform as a Service (PaaS) - Cloud platform & Management: Computation, Storage - Case studies. Software as a Service (SaaS) - Web services - Web 2.0 - Web OS - Case studies – Anything as a service (XaaS)

Cloud Programming and Software Environments – Parallel and Distributed Programming paradigms – Programming on Amazon AWS and Microsoft Azure – Programming support of Google App Engine – Emerging Cloud software Environment

Service Management in Cloud Computing - Service Level Agreements (SLAs) Billing & Accounting Economics of scaling Managing Data: Database & Data Stores in Cloud, Large Scale Data Processing

Task Scheduling in Cloud - Scheduling Algorithms for Computing Clouds Fair Queuing Start Time Fair Queuing Borrowed Virtual Time Cloud Scheduling Subject to Deadlines Scheduling MapReduce Applications Subject to Deadlines

Cloud Security - Cloud Security Risks, Trust, Operating System Security, VM Security, Security of Virtualization, Security Risks Posted by Shared Images, Security Risks Posted by Management OS, Data privacy and security Issues, Identity & Access Management, Access Control, Authentication in cloud computing,
Case Study - Microsoft Azure, Amazon EC2

Reference Books/Material

1. Dan C Marinescu, Cloud Computing, Theory and Practice, MK Elsevier
2. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley
3. Kai Hwang, Geoffrey C. Fox and Jack J. Dongarra, “Distributed and cloud computing from Parallel Processing to the Internet of Things”, Morgan Kaufmann, Elsevier – 2012
4. Barrie Sosinsky, “ Cloud Computing Bible” John Wiley & Sons, 2010
5. Tim Mather, Subra Kumaraswamy, and Shahed Latif, Cloud Security and Privacy An Enterprise Perspective on Risks and Compliance, O'Reilly 2009
6. Jim Smith, Ravi Nair, Virtual Machines: Versatile Platforms for Systems and Processes, MK Elsevier

Course Code	Course Name	L	T	P	Credits
CS535	Internet of Things	3	0	0	3

Course Objective

The objectives of this course are to understand the fundamentals of Internet of Things and its building blocks along with their architecture, to understand the enabling technologies of IoT like wireless sensor networks, fog computing, cloud computing etc. in everyday life, to understand the layered protocols and standards designed for IoT and the current research on it and to learn the advancement in IP addressing and messaging protocols in IoT eco system.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Thorough about the technology behind the IoT and associated technologies

CO2. Use IoT technologies in real life like design of smart city, smart agriculture etc

CO3. To gain knowledge about the state-of-the-art methodologies in IoT application domains

CO4. Use sensors for data collection, use the communication technologies for data transmission, and analyze the data for various applications

CO5. Do the performance analysis of the protocols like efficiency, throughput, delay, packet delivery ratio etc during data transmission

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

Definition of Internet of Things: Life in IoT ecosystem, Characteristics of IoT, IoT components, Pillars of IoT, IoT Design goals, Applications of IoT, Why IoT? Examples of IoT ecosystem, Definition of THINGS, examples of things

IoT protocol stack layers, physical or sensor layer processing and control action layer hardware interface layer, RF layer session / message layer user experience layer application layer Sensor, Networks Design principles of connected devices, IoT Architecture Reference Models, Physical design of IoT, Logical design of IoT

Wireless sensor networks, Sensing & Actuation, cloud computing IaaS PaaS SaaS, Fog layer, Role of fog layer in IoT eco system, big data analysis, embedded systems, security protocols and architecture, Communication protocols, web services microcontrollers and their interface to sensors, ARM microcontroller

Addressing and identification, IP address IPV4 IPV6 address formats embedding IPV4 into IPV6 neighbour discovery in IPV6, IPV6 packet, IPV6 frame format, IPV6 extension header

Message queuing telemetry transport (MQTT), architecture of MQTT, MQTT message format, publish- subscribe architecture in MQTT, client and broker architecture, MQTT broker constrained application protocol (CoAP), CoAP architecture, CoAP messages, CoAP request / response model, HTTP vs MQTT /CoAP

Bluetooth, Bluetooth low energy (BLE), BLE over Bluetooth, BLE features, BLE components, BLE protocol stack, IEEE 802.15.X, architecture of LR-WPAN, 6LoWPAN, Zigbee, Wireless HART RFD FFD IEEE 802.15.4 network topologies, Link quality indication clear channel assessment

Low power wide area networking, Domain specific IoT, Routing protocol RPL low power and lossy networks, IoT and M2M (Machine-to-Machine) communication, Interoperability in IoT

Reference Books/Material

1. Rajkumar Buyaa and Amir V Dastjerdi, Internet of things: Principles and Paradigms, Morgan Kaufmann
2. A Bahga & V Madisetti, Internet of Things: A Hands On Approach, Universities Press
3. Adrian McEwen and Hakim Cassimally, Designing the Internet of Things, Wiley
4. Olivier Hersent, David Boswarthick and Omar Elloumi, The Internet of Things: Key applications and Protocols, Wiley

Course Code	Course Name	L	T	P	Credits
CS536	High Performance Computing	3	0	0	3

Course Objective

The objectives of this course are to introduce various High-performance computing environments and implementation issues.

Course Outcomes

At the completion of this course, the student will be able to

CO1. Learn concepts, issues and limitations related to parallel computing architecture and software development

CO2. Apply different parallel models of computation, parallel architectures, interconnections and various memory organization in modern high performance architectures

CO3. Analyze the algorithms to map them onto parallel architectures for parallelism

CO4. Evaluate the performance of different architectures and parallel algorithms with different aspects of real time problems

CO5. Design parallel programs for shared-memory architectures and distributed-memory architectures using modern tools like OpenMP and MPI, respectively for given problems

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

Levels of Parallelism (Instruction, Transaction, Task, Thread, Memory, Function), Models (SIMD, MIMD, SIMT, SPMD, Dataflow Models, Demand-driven Computation etc.), Architectures: N-wide Superscalar Architectures, Multi-core, Multi-threaded

Fundamental design issues in Parallel Computing, Synchronization, Scheduling, Job Allocation, Job Partitioning, Dependency Analysis, Mapping Parallel Algorithms onto Parallel Architectures, Performance Analysis of Parallel Algorithms

Fundamental Limitations Facing Parallel Computing, Bandwidth Limitations, Latency Limitations, Latency Hiding/Tolerating Techniques and their Limitations, Power-Aware Computing and Communication, Power-Aware Processing Techniques, Power-Aware Memory Design, Power-Aware Interconnect Design, Software Power Management

Programming Languages and Programming-Language Extensions for HPC, Inter-Process Communication, Synchronization, Mutual Exclusion, Basics of Parallel Architecture, Parallel Programming Parallel Programming with OpenMP and (Posix) Threads, Message Passing with MPI

Parallel programming with cuda, Processor Architecture, Interconnect, Communication, Memory Organization, and Programming Models in High Performance Computing Architectures: (Examples: IBM CELL BE, Nvidia Tesla GPU, Intel Larrabee Micro architecture and Intel Nehalem Micro architecture), Memory Hierarchy and Transaction Specific Memory Design, Thread Organization

Petascale Computing, Optics in Parallel Computing, Quantum Computersfor Job Hunting,2009, New Delhi,CUP (2 Audio CDs)

Reference Books/Material

1. 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
2. A.Y.Zomaya, "Parallel and Distributed Computing Handbook", 1995. McGrawHill Professional
3. Michael J.Quinn, "Parallel Computing: Theory and Practice", Second Edition, 1994, Tata McGrawHill, Inc
4. M.Sasikumar, Dinesh Shikhare, P.Ravi Prakash, "Introduction to Parallel Processing", 2000, PHI
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
8. https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html

Course Code	Course Name	L	T	P	Credits
CS537	Cluster and Grid Computing	3	0	0	3

Course Objective

The objectives of this course are to understand the cluster and grid computers, to understand task scheduling and resource allocation in cluster and grid environment, to understand middleware architecture in Cluster and Grid Environment, and to understand the cluster and grid computing platform as an alternative to traditional supercomputers.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Analyze and design cluster and grid computing environment

CO2. Understand the use of Globus tools and SOAP standards by following Cluster and Grid Systems

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								

Syllabus

Introduction: High Performance Computing (HPC), Grand Challenge Problems Computational and communication intensive, Parallel Architectures Classifications SMP, MPP, NUMA, Clusters and Components of a Parallel Machine, Conventional Supercomputers and its limitations

Multi-processor and Multi Computer based Distributed Systems. Cluster and Grids: Cluster Components Processor/machine

High Speed Interconnections goals, topology, latency, bandwidth, Example Interconnect: Myrinet, Inifiniband, QsNet, Fast Ethernet, Gigabit Ethernet, Light weight Messaging system/Light weight communication Protocols

Cluster Middleware Job/Resource Management System, Load balancing, Scheduling of parallel processes, Enforcing policies, GUI

Introduction to programming tools such as PVM, MPI, Cluster Operating Systems Examples: Linux, MOSIX, CONDOR, Characteristics of Grid, Computational services, Computational Grids, Data grids/Storage grids, management and applications, Different components of Grid Grid fabric, Grid middleware

Grid applications and portal, Globus toolkit Ver.2.4, web services, MDS,GRAM, Grid Security – Cryptography, Authentication, Integrity, Digital Signature, Digital Certificates, Certificate Authority, MD 5, RSA, GSI, GSSAPI, Directory Service, LDAP,GRID FTP,GASS Fault Tolerance: Fault detection and diagnosis of Clusters and Grids. Recent advances in cluster and grid computing.

Reference Books/Material

1. R. K. Buyya, High Performance Cluster Computing: Programming and Applications, PHI , 2021
2. D. Janakiram, Grid Computing, Tata Mcgraw Hill , 2005
3. J. J. Jos & R. K. Buyya, High Performance Cluster Computing: Architecture and Systems, PHI , 1999
4. P. Jalote, Fault Tolerance in Distributed Systems, Prentice Hall, 1994
5. Sanjaya Kumar Panda and Pabitra Mohan Khilar, "MSSA: A M-Level Sufferage-based Scheduling Algorithm in Grid Environment", Proceedings of 9th International Conference on Distributed Computing and Internet Technology (ICDCIT), Lecture Notes in Computer Science, Springer, Bhubaneswar, Vol. 7753, 5th – 8th Feb 2013, pp. 410 – 419, ISSN: 0302-9743.(Springer)
6. Sanjaya Kumar Panda and Pabitra Mohan Khilar, "A Two-Step QoS Priority for Scheduling in Grid", Proceedings of The Second IEEE International Conference on Parallel, Distributed and Grid Computing (PDGC), IEEE, Waknaghat, 6th - 8th Dec 2012, pp. 502 – 507

Course Code	Course Name	L	T	P	Credits
CS538	Quantum Computing	3	0	0	3

Course Objective

The objectives of this course are to impart knowledge about the quantum-mechanical phenomena such as superposition and entanglement to perform computation, to introduce the fundamental concepts, Quantum Computing and to enable the students to understand the quantum computing and quantum information in depth.

Course Outcomes

At the completion of this course, the student will be able to

CO1. Understand quantum computation

CO2. Understand Hilber space, entanglement and basics of quantum mechanics

CO3. Compare between classical and quantum information theory

CO4. Demonstrate quantum algorithms such as Shor's and Grover's

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

Introduction to Quantum Computation: Quantum bits, Bloch sphere presentation of a qubit, multiple qubits. Background Mathematics and Physics, Hilber space, Probabilities and measurements, entanglement, density operators and correlation, basics of quantum mechanics, Measurements in bases other than computational basis

Quantum Circuits: single qubit gates, multiple qubit gates, design of quantum circuits. Quantum Information and Cryptography: Comparison between classical and quantum information theory

Bell states, Quantum teleportation, Quantum Cryptography, no cloning theorem. Quantum Algorithms: Classical computation on quantum computers

Relationship between quantum and classical complexity classes

Deutsch’s algorithm, Deutsch’s-Jozsa algorithm, Shor factorization, Grover search

Noise and error correction: Graph states and codes, Quantum error correction, fault-tolerant computation

Advance topic; Quantum resistance cryptography

Reference Books/Material

1. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press. 2002
2. Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics, World Scientific. 2004
3. Pittenger A. O., An Introduction to Quantum Computing Algorithms, 2000

Course Code	Course Name	L	T	P	Credits
CS539	Advanced Topics in IoT	3	0	0	3

Course Objective

The objectives of this course are to understand Industry 4.0 (or the Industrial Internet), its applications in the business world, to give deep insights into how smartness is being harnessed from data and appreciate what needs to be done in order to overcome some of the challenges, to provide knowledge of designing Industrial IOT Systems for various applications.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Understand the drivers and enablers of Industry 4.0

CO2. Analyze the smartness in Smart Factories, Smart cities, smart products and smart services

CO3. Design and analysis of the various systems used in a manufacturing plant and their role in an Industry 4.0 world

CO4. Identify, formulate and solve engineering problems by using Industrial IoT

CO5. Implement real field problem by gained knowledge of Industrial applications with IoT capability

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H	M	L							
CO2	H		H	M								
CO3	H		M		L							
CO4	H			H	L							

Syllabus

Industry 4.0: Globalization and Emerging Issues, The Fourth Revolution, LEAN Production Systems, Smart and Connected Business Perspective, Smart Factories

Industry 4.0: Cyber Physical Systems and Next Generation Sensors, Collaborative Platform and Product Lifecycle Management, Augmented Reality and Virtual Reality, Artifical Intelligence, Big Data and Advanced Analysis

IIoT-Introduction, Industrial IoT: Business Model and Reference Architecture: IIoT-Business Models, Industrial IoT- Layers: IIoT Sensing, IIoT Processing, IIoT Communication, IIoT Networking

Big Data Analytics and Software Defined Networks, Machine Learning and Data Science, Julia Programming, Data Management with Hadoop

Industrial IoT: Security and Fog Computing - Cloud Computing in IIoT, Fog Computing in IIoT, Security in IIoT

Industrial IoT- Application Domains: Oil, chemical and pharmaceutical industry, Applications of UAVs in Industries, Real case studies: Milk Processing and Packaging Industries, Manufacturing Industries

Reference Books/Material

1. Industry 4.0: The Industrial Internet of Things Alasdair Gilchrist Publications: Apress
2. The Concept Industry 4.0 An Empirical Analysis of Technologies and Applications in Production Logistics Authors: Bartodziej, Christoph Jan Springer: Publication in the field of economic science.
3. Embedded System: Architecture, Programming and Design by Rajkamal, TMH3.
4. Dr. OvidiuVermesan, Dr. Peter Friess, “Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems”, River Publishers

Course Code	Course Name	L	T	P	Credits
CS540	Modern Cryptography	3	0	0	3

Course Objective

The objectives of this course are to understand to understand group theory, number theory, and discrete probability, to analyze probabilistic algorithms, to develop the ability to model security problems and to write security proofs, to understand basic computational problems that are important for cryptography such as the factoring problem, the RSA problem, the discrete-logarithm problem and to understand and analyze the design of advanced cryptosystems related to lattice based cryptography, homomorphic encryption, and attribute-based encryption.

Course Outcomes

At the completion of this course, the student will be able to

CO1. Understand formal security definitions, security assumptions and security proofs of modern cryptosystems

CO2. Demonstrate familiarity with modern day cryptosystems and prove its security strengths with respect to the state of the art cryptanalytic attacks

CO3. Analyze and evaluate the security strengths of newer cryptosystems

CO4. Analyze different security models and protocols

CO5. Evaluate encrypted data using encrypted data processing techniques

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO 12
CO1	L	H		M	H	M						
CO2		H		H				M		M		
CO3	H	M	M		H				L			
CO4	M	H	L	H						M		H
CO5			M									

Syllabus

Classical Cryptography and Modern Cryptography, Principles of Modern Cryptography, formal Definitions, Precise Assumptions, Proofs of Security, Provable Security and RealWorld Security. Formal Definitions, Shannon's Theory, one-Time Pad, Limitations of Perfect Secrecy

Defining Computationally Secure Encryption, Semantic Security, Constructing Secure Encryption Schemes-Pseudorandom Generators and Stream Ciphers, Proofs by Reduction, Cryptanalytic Attacks-Chosen-Plaintext Attacks and CPA-Security, Constructing CPA-Secure Encryption Schemes, Pseudorandom Functions and Block Ciphers, Cpa-Secure Encryption From Pseudorandom Functions, Chosen-Ciphertext Attacks- Defining CCA-Security

Hash Functions-one-Wayness and Collision Resistance, Merkle–Damgard Construction, Attacks on Hash Functions-Birthday Attacks, Random-oracle Model, Merkle Trees
Message Authentication Codes – formal Definitions, Design, and Proof of Security, HMAC, CBC-MAC, Authenticated Encryption, information-Theoretic Macs, Limitations on information-Theoretic Macs

Algorithms for Factoring-Pollard's P − 1 Algorithm, Pollard's Rho Algorithm , Quadratic Sieve Algorithm, Algorithms for Computing Discrete Logarithms- Pohlig–Hellman Algorithm, BabyStep/Giant-Step Algorithm, Discrete Logarithms From Collisions, index Calculus Algorithm

RSA Encryption, Security Against Chosen-Plaintext Attacks, Security Against Chosen Ciphertext Attacks, RSA Implementation Issues and Pitfalls, Computational Diffie Hellman/Decisional Diffie-Hellman Based Encryption, Elliptic Curve Cryptography-Elliptic
Curve Over Finite Fields and Binary Fields, Elliptic Curve Discrete Logarithm Problem, Cryptosystems Based on Elliptic Curve, Introduction to Pairing-Based Cryptography, 3-Party Non-Interactive Key-Exchange from Pairings, Short Signatures From Pairings, Identity-Based Encryption from Pairings

Homomorphic Signatures, Partial Homomorphic Encryption, Somewhat Homomorphic Encryption, Fully Homomorphic Encryption, Dual Regev Encryption, Attribute-Based Encryption

Secure Multi-Party Computation, Oblivious Transfer Protocols, Yao's Garbled Circuits, Shamir Secret Sharing, Computing on Secret-Shared Data, SMPC in the Preprocessing Model: OT Correlations and Beaver Triples

Overview of Post-Quantum Cryptography, Introduction to Lattice-Based Cryptography, The Short Integer Solutions (SIS) Problem, Lattice Trapdoors, and Lattice-Based Signatures, The Learning With Errors (LWE) Problem, Regev's Public-Key Encryption Scheme from LWE

Reference Books/Material

1. J. Katz and Y. Lindell, Introduction to Modern Cryptography, 3rd Ed. CRC press, 2020.
2. Douglas R. Stinson, Cryptography: Theory and Practice, Third Edition, Publisher: Chapman and Hall/CRC, 2005.
3. Goldreich, Foundations of Cryptography, Cambridge University Press, 2005 (Vol. 1 and 2).
4. Boneh, Dan, and Victor Shoup. "A graduate course in applied cryptography." https://crypto.stanford.edu/~dabo/cryptobook/BonehShoup_0_4.pdf (2017).
5. Bellare, Mihir, and Phillip Rogaway. "Introduction to modern cryptography." UCSD CSE 207 (2005).

Course Code	Course Name	L	T	P	Credits
CS541	Cryptocurrency and Blockchain Technology	3	0	0	3

Course Objective

The objectives of this course are to demonstrate a familiarity with the concepts related to blockchain technology, to understand the different components of the Blockchain system and interact with them securely, to design, build and deploy smart contracts and distributed applications and to assess the strengths and weaknesses of blockchain enabled decentralization in different application scenarios

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Have knowledge about the design principles of blockchain and smart contracts

CO2. Familiarize with the existing cryptocurrencies and blockchain platforms

CO3. Program and demonstrate the working of different consensus mechanisms

CO4. Design, build, and deploy distributed applications and smart contracts by identifying the need of blockchains to find the solution to the real-world problems

CO5. Evaluate security, privacy, and efficiency of a given blockchain use case

\

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	H			H	M						
CO2		H		H				M		M		
CO3	H	M			H				L			
CO4	M	H	L	H			L			M		H
CO5			M		H			H				

Syllabus

Introduction to Blockchain and Digital Currency, Evolution, Blockchain as Public ledger, Structure of a Block, Transactions, Merkel Trees, Peer-to-Peer Networks, Double Spend Problem, Decentralization Applications, Characteristics, Benefits and Challenges

The consensus problem - Asynchronous Byzantine Agreement - AAP protocol and its analysis - Nakamoto Consensus on permission-less, nameless, peer-to-peer network - Abstract Models for BLOCKCHAIN - GARAY model - RLA Model - Proof of Work (PoW) as random oracle - formal treatment of consistency, liveness and fairness - Proof of Stake (PoS) based Chains - Hybrid models (PoW + PoS)

Hash Functions, Public Key Cryptosystem, Public Key Generation, Digital Signature, Zero Knowledge Proof, k-Anonymity, ECC

Bitcoin Concept- Merkle Tree - hardness of mining - transaction verifiability - anonymity, Attacks on Bitcoin- Double-spend attacks, Selfish mining, Security of Transactions in Bitcoin, Privacy in Bitcoin, mathematical analysis of properties of Bitcoin

Ethereum - Ethereum Virtual Machine (EVM) - Wallets for Ethereum – Solidity, DApps - Smart Contracts - some attacks on smart contracts

Zero Knowledge proofs and protocols in Blockchain - Succinct non interactive argument for Knowledge (SNARK) - pairing on Elliptic curves – Zcash, Stellar Introduction-Stellar Transactions, Stellar Consensus Protocol, Monero Introduction- Monero Ring Signatures

Permissioned Blockchain – Hyperledger, Blockchain Applications & Use Cases, Consensus Protocols- The consensus problem- Byzantine Generals problem, Asynchronous Byzantine Agreement, Consensus mechanisms. Introduction of other blockchains- IOTA, Ripple, Corda etc. Advantages of integrating Blockchain to IoT, Trust Building, Cost Reduction, Accelerate Data Exchanges, Scaled Security for IoT. Applications of smart contracts, Blockchain Application in various areas- Banking, Supply Chain, Healthcare, Real-Estate, Judiciary, IoT, Insurance, Energy etc.

Reference Books/Material

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016.
2. Roger Wattenhofer, “Blockchain Science: Distributed Ledger Technology”, independently Published, ISBN-10 : 1793471738, 2019.
3. Andreas M. Antonopoulos, “Mastering Bitcoin: Programming the Open Blockchain”, Shroff/O'Reilly, 2017.
4. Alan T. Norman, “Blockchain Technology Explained: the Ultimate Beginner s Guide About Blockchain Wallet, Mining, Bitcoin, Ethereum, Litecoin, Zcash, Monero, Ripple, Dash, IOTA and Smart Contracts”, Amazon Digital Services, 2017.
5. William Magnuson, “Blockchain Democracy- Technology, Law and the Rule of the Crowd”, Cambridge University Press, 2020.
6. Pethuru Raj, Kavita Saini, Chellammal Surianarayanan, “Blockchain Technology and Applications”, CRC Press, 2021.
7. Chandramouli Subramanian, “Blockchain Technology”, Universities Press, 2020.
8. Joseph Bonneau et al, SoK: Research perspectives and challenges for Bitcoin and cryptocurrency, IEEE Symposium on security and Privacy, 2015.
9. J.A.Garay et al, The bitcoin backbone protocol - analysis and applications EUROCRYPT 2015 LNCS VOl 9057, (VOLII), pp 281-310.
10. R.Pass et al, Analysis of Blockchain protocol in Asynchronous networks, EUROCRYPT 2017, (eprint.iacr.org/2016/454).
11. R.Pass et al, Fruitchain, a fair blockchain, PODC 2017 (eprint.iacr.org/2016/916).
12. Bahga, Arshdeep, and Vijay Madisetti. "Blockchain applications: a hands-on approach", VPT, 2017.

Course Code	Course Name	L	T	P	Credits
CS542	Penetration Testing	3	0	0	3

Course Objective

The objectives of this course are to understand the core foundations of ethics in regards to computer security, learn about the hacker mindset and the history of hackers, understand basic networking and security technologies, to perform penetration testing, explore various hacking tools/techniques and to learn about basic system defense infrastructure.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Acquire knowledge of core concepts related to information security and ethical hacking

CO2. Install, configure, and use different state of the art hacking software on a closed network environment

CO3. Analyze the vulnerabilities related to computer system and networks using state of the art tools and technologies

CO4. Implement effective solutions for ethical hacking in different environments

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	H			H	M						
CO2		H		H				M		M		
CO3	H				H				L		M	
CO4	M	H	L	H			L			M		H

Syllabus

Ethical Hacking: Introduction, Networking & Basics, Footprinting and Reconnaissance, Google Hacking, Scanning Networks

Enumeration Vulnerability scanning using NMAP and Nessus. Creating a secure hacking environment. System Hacking: password cracking, privilege escalation, application execution. Malware and Virus. ARP spoofing and MAC attack.

Cryptography, Wireless Hacking, Firewall & Honeypots, IDS & IPS, Vulnerability, Penetration Testing, Session Hijacking, Hacking Web Servers, SQL Injection, Cross Site Scripting, Exploit Writing, Buffer Overflow, Sniffers, Session Hijacking and Denial of Service

Web Server Hacking, Web Applications and Database Attacks: OWASP principles, Web Server Hacking, Web Application Hacking, Database Hacking, Wireless and mobile Hacking

Reverse Engineering, Email Hacking, Incident Handling & Response, Bluetooth Hacking, Mobile Phone Hacking Basic Ethical Hacking Tools and Usage of These Tools in a Professional Environment. Legal, Professional and Ethical Issues Likely to Face the Domain of Ethical Hacking.

Ethical Responsibilities, Professional Integrity and Making Appropriate Use of the Tools and Techniques Associated With Ethical Hacking

Reference Books/Material

1. Ankit Fadia, "An Unofficial Guide to Ethical Hacking", Macmillan India, 2006.
2. Kimberly Graves, “CEH: Official Certified Ethical Hacker Review Guide”, Wiley Publishing Inc., 2007.ISBN: 978-0-7821-4437-6
3. Dominic Chell, Tyrone Erasmus, Shaun Colley, Oflie Whitehouse,” The Mobile Application Hacker’s Handbook”, 2nd Edition, Wiley, 2015.
4. Michael Gregg, "Certified Ethical Hacker (CEH) Cert Guide", 2nd Edition, Pearson India, 2014.
5. Rafay Baloch, “Ethical Hacking and Penetration Testing Guide”, 2nd Edition, CRC Press, 2017.
6. Allen Harper, Shome Harris, Jonathan Ness,Chris Eagle, Gideon Lenkey, TerronVilliams “Gray Hat Hacking The Ethical Hakers Handbook”, 3rd Edition, TMH, 2011.
7. Patrick Engebretson, “The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy”, 2nd Edition, Elsevier, 2013.

8. Aaron E. Earle, “Wireless Security Handbook”, Auerbach publications, Taylor & Francis Group, 2006.

9. Jon Erickson “HACKING: The art of Exploitation”, 2nd Edition, William Pollock No Starch Press, 2008.

Course Code	Course Name	L	T	P	Credits
CS543	Cyber Crime Investigation and Digital Forensics	3	0	0	3

Course Objective

The objectives of this course are to understand various types of cyber-attacks and cyber-crimes, to learn threats and risks within context of the cyber security, to have an overview of the cyber laws & concepts of cyber forensics, to study the defensive techniques against cyber attacks and to understand the basic digital forensics and techniques for conducting the forensic examination on different digital devices.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Analyze and evaluate the cyber security needs of an organization
CO2. Understand cyber Security Regulations and Roles of International Law.
CO3. Design and develop a security architecture for an organization.
CO4. Understand fundamental concepts of data privacy attacks
CO5. Know how to apply forensic analysis tools to recover important evidence for identifying computer crime
CO6. Be well-trained as next-generation computer crime investigators

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L			H	M						
CO2				H				L		M		

CO3	H				H				L		M	
CO4	M	H	L	H			L			M		H
CO5			L		H			M				
CO6	M				M							M

Syllabus

Introduction and Overview of Cyber Crime, Nature and Scope of Cyber Crime, Types of Cyber Crime: Social Engineering, Categories of Cyber Crime, Property Cyber Crime
Cyber Crime Issues: Unauthorized Access to Computers, Computer Intrusions, White collar Crimes, Viruses and Malicious Code, Internet Hacking and Cracking, Virus Attacks, Pornography, Software Piracy, Intellectual Property, Mail Bombs, Exploitation, Stalking and Obscenity in Internet, Digital laws and legislation, Law Enforcement Roles and Responses

The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Cybercrime and Punishment, Cyber Law, Technology and Students: Indian Scenario. Private Ordering Solutions, Regulation and Jurisdiction For Global Cyber Security, Copyright Source of Risks, Pirates, Internet Infringement, Fair Use, Postings, Criminal Liability, First Amendments, Data Losing, Cyber Ethics - Legal Developments, Cyber Security in Society, Security in Cyber Laws Case Studies, General Law and Cyber Law-A Swift Analysis

Introduction to Cyber Crime Investigation, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidences, Password Cracking

Computer forensics fundamentals, Benefits of forensics, computer crimes, computer forensics evidence and courts, legal concerns and private issues

Understanding Computing Investigations – Procedure for corporate High-Tech investigations, understanding data recovery work station and software, conducting and investigations

Data acquisition- understanding storage formats and digital evidence, determining the best acquisition method, acquisition tools, validating data acquisitions, performing RAID data acquisitions, remote network acquisition tools, other forensics acquisitions tools

Processing crimes and incident scenes, securing a computer incident or crime, seizing digital evidence at scene, storing digital evidence, obtaining digital hash, reviewing case

Current computer forensics tools- software, hardware tools, validating and testing forensic software, addressing data-hiding techniques, performing remote acquisitions, E-Mail investigations- investigating email crime and violations, understanding E-Mail servers, specialized E-Mail forensics tool

Reference Books/Material

1. Warren G. Kruse II and Jay G. Heiser, "Computer Forensics: Incident Response Essentials", Addison Wesley, 2002.
2. Nelson, B, Phillips, A, Enfinger, F, Stuart, C., "Guide to Computer Forensics and Investigations, 2nd ed., Thomson Course Technology, 2006, ISBN: 0-619-21706-5.
3. Sunit Belapure and Nina Godbole, Cyber "Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, 1st Edition, Wiley India Pvt. Ltd, 2011.
4. Mark F Grady, Fransesco Parisi, "The Law and Economics of Cyber Security", 1st Edition, Cambridge University Press, 2006.
5. Jonathan Rosenoer, "Cyber Law: The law of the Internet", 1st Edition, Springer-Verlag, 1997.
6. J. Vacca, "Computer Forensics: Computer Crime Scene Investigation", 2nd Edition, Charles River Media, 2005, ISBN: 1-58450-389.

Course Code	Course Name	L	T	P	Credits
CS544	Malware Analysis and Mitigation	3	0	0	3

Course Objective

The objectives of this course are to identify and describe common traits of malware, to examine and analyse malwares using static and dynamic analysis techniques, to apply different tools for malware detection, to evaluate potential threats due to malware activity on system or network, and to create malware analysis report from studied technique and develop mitigation strategies.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Have the knowledge of different types of malware, its behavior and analysis techniques

CO2. Apply different tools and techniques for malware data acquisition and analysis

CO3. Analyze and evaluate the effect of malware on system and network

CO4. Create malware analysis report and suggest suitable preventive measures

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L			H	M						
CO2				H				L		M		
CO3	M		M		H				L			
CO4	M			H			L			M		H

Syllabus

Introduction to Malwares, Different Types of Malwares, Characteristics of Malwares

Identification and Initial Assessment of Malwares, Antivirus Scanning, Hashing, Finding Strings, Packed and Obfuscated Malware, File Formats, Linked Libraries and Functions, X86 Architecture and Disassembly, Recognizing C Code Constructs In Assembly, Analyzing Malicious C Programs, Shellcode Analysis

Sandboxes, Process Monitors, Process Explorer, Registry Snapshots, Faking A Network, Packet Sniffing, Source and Assembly Level Debugger, Kernel and User Level Debugging, Exceptions

Malware Behaviour, Covert Malware Launching, Data Encoding, Malware Focused Network Signatures

Anti-Disassembly, Anti-Debugging, Anti-Virtual Machine Techniques, Packers and Unpacking Problems in large scale classification: scalability; triage methods; Required FP rate Hiding: Polymorphism: compression encryption virtualization; Metamorphism: high level code obfuscation engines, on-board metamorphic engines, semantics-preserving rewritings; Frankenstein

The theory of malware: Rice's theorem and the undecidability of semantic equivalence; Adleman's proof of the undecidability of the presence of a virus; Cohen's experiments on detectability and self-obfuscation Advanced Dynamic Analysis: debugging tools and concepts, Malware Behavior - malicious activities and techniques, Analyzing Windows programs – WinAPI, Handles, Networking, COM, Data Encoding, Malware Countermeasures, Covert Launching and Execution, Anti Analysis- Anti Disassembly, VM, Debugging -, Packers – packing and upacking, Intro to Kernel – Kernel basics, Windows Kernel API, Windows Drivers, Kernel Debugging - Rootkit Techniques- Hooking, Patching, Kernel Object Manipulation, Rootkit Anti- forensics, Covert analysis.

Reference Books/Material

1. Michael Sikorski, andrewHonig, Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software publisher William Pollock, 2012.
2. Michael Hale Ligh, andrew Case, Jamie Levy, AAron Walters, The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory, 2014.
3. Ligh, M., Adair, S., Hartstein, B., Richard, M., "Malware analyst's cookbook and DVD: tools and techniques for fighting malicious code", Wiley Publishing, 2010.
4. Marak V., "Windows malware analysis essentials", Packt Publishing Ltd, 2015.
5. Dang, B., Gazet, A., Bachaalany, E., "Practical reverse engineering: x86, x64, ARM, Windows kernel, reversing tools, and obfuscation", John Wiley & Sons, 2014.
6. Jamie Butler and Greg Hoglund, Rootkits: Subverting the Windows Kernel, AddisonWesley, 2005.
7. Reverend Bill Blunden, The Rootkit Arsenal: Escape and Evasion in the Dark Corners of the System, Second Edition, Jones & Bartlett, 2012.

Course Code	Course Name	L	T	P	Credits
CS545	Design and Analysis of Security Protocols	3	0	0	3

Course Objective

The objectives of this course are to understand concepts of security protocols and its analysis, to understand how applications can communicate securely and what tools and protocols exist in order to offer different levels of security, to get knowledge and the ability to critically analyze and design secure networks, applications and systems, to give hands-on experience in using

automated tools and formal techniques to analyze and evaluate cryptographic protocols and other security mechanisms and to analyze various existing protocols in terms of the goals.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. Understand different authentication techniques, key exchange protocols and security issues while designing the protocols
CO2. Get a hands-on exposure to the principles and techniques used in security systems, as well as designing security protocols
CO3. Analyze the security protocols against different attacks
CO4. Design a key agreement or key transport or key establishment protocol satisfying various security goals

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L			H	M						
CO2				H				L		M		
CO3	L		L		H				L		M	
CO4	M			M		L	L			M		H

Syllabus

Introduction to Computer Security, Security Protocols, Security Analysis

Overview of SSL/TLS, Creating an Abstract Model, Coding Up in Murphi, Specification and Verification of Security Properties

Key Management, Kerberos, Public-Key infrastructure, Security Properties and Attacks on Them, Needham-Schroeder Lowe Protocol, Diffie-Hellman Key Exchange, IPSec, IKE

Fundamental Limitation of Contract-Signing and Fair-Exchange, Trusted Third Party, Optimistic Contract-Signing, Asokan-Shoup-Waidner Protocol, Desirable Properties (Fairness, Timeliness, Accountability, Balance), Abuse-Free Contract-Signing

Hashed Password Files and Salt, Web Authentication Issues: Sniffing, Phishing, Spyware, Password-Authenticated Key Exchange Protocols

Crowds System, Probabilistic Notions of Anonymity, Markov Chains, Prism, PCTL Logic, Probabilistic Fair Exchange

Protocol Analysis Using Theorem Proving, inductive Proofs, Isabelle Theorem Prover, Verifying the Secure Electronic Transactions (Set) Protocols Using Isabelle

Rabin's Beacon, Rabin's Contract Signing Protocol, BGMR Probabilistic Contract Signing, formal Model for the BGMR Protocol

The Problem of Fair Exchange, Protocol As A Game Tree, Alternating Transition Systems, Alternating-Time Temporal Logic, Mocha Model Checker

Yahalom Protocol: Secrecy, Authentication, Non-Repudiation, Anonymity; Dolev-Yao Threat Model, Needham- Schroeder Public-Key Protocol and Its Security Analysis. Wireless

Networking Protocol, Logic for Computer Security Protocols: Floyd-Hoare Logic of Programs, Ban Logic, Compositional Logic for Proving Security Properties of Protocols, Probabilistic Polynomial-Time Process Calculus for Security Protocol Analysis

Reference Books/Material

1. Peter Ryan, Steve Schneider, Michael Goldsmith, Gavin Lowe, Bill Roscoe: Modelling & Analysis of Security Protocols, Addison Wesley, 2000.
2. Stephen W. Mancini, "Automating Security Protocol Analysis", Biblioscholar, 2012.
3. Ulysess Black, "internet Security Protocols: Protecting IP Traffic", Prentice Hall PTR; 1st edition, ISBN-10: 0130142492, ISBN-13: 978-0130142498, 2000.
4. Giampaolo Bella, "formal Correctness of Security Protocols", Springer, 2007.
5. Dinesh Goyal, S. Balamurugan, Sheng-Lung Peng, O.P. Verma, "Design and Analysis of Security Protocol for Communication, Scrivener Publishing, 2020.
6. Reverend Bill Blunden, The Rootkit Arsenal: Escape and Evasion in the Dark Corners of the System, Second Edition, Jones & Bartlett, 2012.

Course Code	Course Name	L	T	P	Credits
CS546	Privacy Preserving Data Publishing	3	0	0	3

Course Objective

The objectives of this course are to understand privacy threats and model, while publishing the data, to understand and compare different techniques for privacy preservation, to get knowledge and the ability to critically analyze and design privacy preservation algorithms and to analyze issues with data mining and the existing solutions.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. apply anonymization methods for sensitive data protection

CO2. apply state-of art techniques for data privacy protection

CO3. design privacy preserving algorithms for real-world applications

CO4. identify security and privacy issues in OLAP systems

CO5. apply information metrics for Maximizing the preservation of information in the anonymization process

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L			H	M						
CO2		M		H				L				
CO3	L				M				L		L	
CO4						L	L			M		H
CO5			M				M				M	

Syllabus

Privacy issues and privacy models

Anonymization: Operations on Anonymization, Information metrics, Anonymization methods for the transaction data, trajectory data, social networks data, and textual data. Collaborative Anonymization

Access control methods for outsourced data, Use of Fragmentation and Encryption to Protect Data Privacy;
Security and Privacy in OLAP systems.Extended Data publishing Scenarios, Anonymization for Data Mining and social media data, Privacy preserving data mining

Reference Books/Material

1. Benjamin C.M. Fung, Ke Wang, Ada Wai-Chee Fu and Philip S. Yu, "Introduction to PrivacyPreserving Data Publishing: Concepts and Techniques", 1st Edition, Chapman & Hall/CRC, 2010.
2. Charu C. Aggarwal, "Privacy-Preserving Data Mining: Models and Algorithms", 1st Edition, Springer, 2008.

Course Code	Course Name	L	T	P	Credits
CS547	Secure Multi-Party Computation	3	0	0	3

Course Objective

The objectives of this course are to understand real world notion of secure multi party computation, to analyze various models of secure computation, to privacy threats and model, while publishing the data, to analyze and design semi honest and active security secure computation protocols for solving real world problems and to understand the existing zero knowledge proof systems.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. understand real world notions related to secure multi party computation
CO2. distinguish various models with respect to secure computation
CO3. apply semi honest and active security secure computation protocols on solving real world problems
CO4. apply zero knowledge proofs

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L			H	M						
CO2			L	H				L				
CO3	L								L		L	
CO4		H								M		H
CO5	M		L		L		M				M	

Syllabus

Introduction to secure computing: Definition-Secure Computation-Computational/statistical Indistinguishability; Real-Ideal World or Simulation based Security notions, secret sharing;

Models of Secure Computation: Honest vs. Dishonest majority settings; semi-honest vs active (malicious) adversary; static vs. adaptive computation; computational vs. information theoretic security; synchronous vs. asynchronous network

Oblivious Transfer and its extension: Oblivious transfer protocols - Definitions, constructions, and applications

Secure computation with semi-honest security: Honest-majority Setting- Secret Sharing, BenOrGoldwasser-Wigderson (BGW) Construction, Optimizations (MPC in preprocessing mode and circuit randomization), Cramer-Damgaard-Neilsen (CDN) Construction; Dishonest majority Setting Oblivious Transfers (OT), two-party Goldreich-Micali-Wigderson (GMW) construction, Optimizations of GMW (Random input OT and OT extension), Yao construction, BMR construction and multi-party GMW construction

Secure computation with Active security: Honest Majority Setting.-Verifiable Secret Sharing, BGW Construction with active security, Hyper-invertible Matrices and Beerliova-Hirt (BH) Construction, Information Checking Protocol; Dishonest majority Setting-Commitment Schemes, Zero-knowledge, GMW Compiler for active corruption, Cut-and-Choose OT and Lindell-Pinkas Construction

Zero-knowledge proof systems: zero-knowledge proofs of knowledge; non-interactive zero-knowledge Broadcast & Byzantine Agreement : Dolev-Strong Broadcast; Exponential

Information Gathering (EIG) construction for BA; Berman-Garay-Perry (BGP) construction for BA; Multi-valued Broadcast and BA

Practical Secure Computation: Secure Set Intersection; Privacy Preserving Biometrics & Genomics; Secure Cloud Computing

Reference Books/Material

1. Manoj M. Prabhakaran, Amit Sahai, "Secure Multi-Party Computation, Cryptography and Information security series Vol 10", IOS Press, 2013.

2. Ronald Cramer, Ivan Damgaard and JesperBuus Nielsen, "Secure Multiparty Computation and Secret Sharing - An Information Theoretic Approach", Cambridge Press. (Book Draft), 2015.

Course Code	Course Name	L	T	P	Credits
CS548	Public Key Infrastructure and Trust Management	3	0	0	3

Course Objective

The objectives of this course are to introduce students to concepts of public key infrastructures, understand Identity Management Process and its importance in public key infrastructure analyse the requirements of PKI standards and to understand the trust management.

Course Outcomes

At the completion of this course, the student will be able to:

CO1. analyze Core PKI services: Authentication, Integrity, and confidentiality

CO2. design Certificates using Trust Models, PKI Considerations and Electronic Legislation

CO3. identify PKIX standardization Requirements

CO4. distinguish Public key certificate management models

CO5. understand implementation of a public key infrastructure, including the technology, policy, standards, and long-term maintenance considerations.

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M				H	M						
CO2				H				L				
CO3	M								L		L	
CO4		M						M		H		H
CO5	M		L				M				M	

Syllabus

Pervasive security services, Building a comprehensive security model, Public key cryptosystems, Authentication protocols, Key management techniques

Services offered by PKI- components of a fully functional PKI: Certification authority, Certificate repository, Certificate revocation, Key backup and recovery, Automatic key update, Key history management, Cross-certification, Support for non-repudiation, Time stamping, Client software, PKI & IPSec, PKI Technologies, PKI Solutions Interoperability

PKI architectures – Single CA, Hierarchial PKI, Mesh PKI, Trust Lists, Bridge CA

PKI standards : X.509: Components of X.509: Tamper evident envelope, Basic certificate contents, certificate extensions Generating, using and validating digital signatures, Building a Certification Authority and extending trust through PKI, Integrating a PKI with existing directory systems, Linking PKIs using cross-certification, Identifying certificate components, P2P trust, Web of Trust, Simple PKI (SPKI) / Simple Distributed Security Infrastructure (SDSI): Representing certificates in terms of SExpressions- Certificate Chain Discovery - Distinct Advantages of SPKI/SDSI over X.509. PKI application : Smart card integration with PKI's

Access Control Mechanisms: Discretionary Access Control (DAC) – Mandatory Access Control (MAC) – Role Based Access Control (RBAC)

Revocation- Anonymity-Privacy issues

Trust Management: Policy based Trust Management System- Social network based Trust Management System- Reputation based Trust Management System (DMRep, EigenRep, P2Prep)- Framework for Trust Establishment

Risks Impact on E-Commerce and E- Business: Information Risk – Technology Risk – Business Risk

Reference Books/Material

1. Desmedt, Yvo G. (Ed.), Secure Public Key Infrastructure Standards, PGP and Beyond, Springer, 2012.
2. Jan Camenisch and Costas Lambrinoudakis, Public Key Infrastructures, Services and Applications, 7th European Workshop, EuroPKI 2010.
3. Philip. Robinson, Harald. Vogt, Waleed. Wagealla ,"Privacy, Security, and Trust Within the Context of Pervasive Computing", 1/E, Springer, 2004.
4. David Chadwick, Gansen Zhao, "Public Key Infrastructure: Second European PKI Workshop: Research and Applications, EuroPKI 2005, Canterbury, UK, 2005, Revised Selected Papers (Lecture Notes in Computer Science) ", 1/E, Springer; 2005.
5. Ketil Stølen , William H. Winsborough ,Fabio Massacci, "Trust Management: 4th International Conference, iTrust 2006, Proceedings (Lecture Notes in Computer Science) ", 1/E, Springer, 2006.

Course Code	Course Name	L	T	P	Credits
CS549	Security Analytics	3	0	0	3

Course Objective

The objectives of this course are to discuss the intersection of two ubiquitous concepts: Security and Machine Learning, to understand machine learning algorithms and techniques, and their applications within the security domain, to learn about attacks against computer systems leveraging machine learning algorithms, as well as defense techniques to mitigate such attacks during learning and inference and to analyze and compare various machine learning systems for security. o have a good, familiarity of main algorithms, and required programming skills to be able to design

Course Outcomes

At the completion of this course, the student will be able to:

CO1. understand of fundamental concepts of ML

CO2. evaluate the security of ML systems

CO3. design and analyze the data-driven solutions

CO4. apply and analyze ML based solutions to the address the real world security problems

Relationship of Course Outcomes to Program Outcomes

H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M				H	M						
CO2		L		H				L				
CO3						L			L		L	
CO4		L			L			M		H		H
CO5	M		L				M				M	

Syllabus

Machine Learning Introduction: A brief history and applications, Data-driven problem solving, learning stages, feature engineering, training, model selection, generalization, optimization, prediction overfitting, underfitting, and regularization

Machine Learning Algorithms for Security: Linear Models for Regression- The Ordinary Least Squares (OLS), regularization, Ridge and Lasso regression, Loss Functions and Optimization-Convex optimization, Gradient descent (GD) and stochastic gradient descent (SGD), Nearest Neighbors (NN)-Introduction to supervised learning, similarity measures, decision boundaries, model selection for NN, and metric learning, Decision Trees-Entropy, Information gain, Decision Trees, Regularization, Clustering-Introduction to unsupervised learning, k-means, hierarchical clustering, k-means++, Mixture of Gaussians and EM algorithm, Naive Bayes Classifier-Bayes' theorem, generative models, independence between features, Logistic Regression-Discriminative models, Predicting probabilities, logit function, the exponential loss and loss minimization viewpoint, Gradient Descent (GD) for LR, Feature Selection-Filter and Wrapper methods, Feature Projection-Principal Component Analysis (PCA), Fisher's Linear Discriminant, Analysis Artificial Neural Networks-Binary classification, Perceptron algorithm and its convergence, multiple layer perceptron, an overview of Deep Neural Networks and its key challenges,

Convolutional Neural Networks- Convolution operation, CNN Architectures, filters, padding, sparse connectivity, weight sharing, stride, and Pooling

Security of Machine Learning Systems- Overview of key concepts, Threat, vulnerabilities, risk, adversarial capabilities, trust and security model, poisoning attacks, model extraction, adversarial examples, and threats to privacy, Perturbation Attack Strategies-Adversarial samples, misclassification and targeted attacks, FGSM, JSMA, and CW attacks, Adversarial Examples in Physical-world-Noisy physical environments, dynamic physical conditions including different viewpoint angles and distances, Adversarial Examples in Constrained Domains-Semantics of features and capability of only controlling a subset of features, Transferability & Black-Box Attacks- Substitute models, intra and cross-technique transferability, Evaluating Robustness of ML systems-Defense evaluations and adversarial example defenses, Defenses Against Adversarial Attacks - Pre-processing and robust optimization techniques, and empirical and theoretic approaches, Generative Adversarial Networks (GANs) - Fake data generation and its applications, Introduction to Privacy in ML Models - Model reconstruction, model inversion, membership inference attacks, and privacy preserving ML, Future Research Directions- open research topics

Reference Books/Material

1. Pattern Recognition and Machine Learning, Christopher Bishop, Springer (2006)
2. Machine Learning A Probabilistic Perspective, Kevin P Murphy, MIT Press (2012)
3. Deep Learning, Ian Goodfellow, Yoshua Bengio, and Aaron Courville, MIT press (2016)
4. Adversarial Machine Learning, Yevgeniy Vorobeychik and Murat Kantarcioglu, Morgan & Claypool Publishers (2018)
5. Machine Learning and Security: Protecting Systems with Data and Algorithms, Clarence Chio and David Freeman, O'Reilly Media (2018)

Course Code	Course Name	L	T	P	Credits
CS550	Deep Learning	3	0	0	3

Course Objective

To introduce the building blocks of deep neural networks architectures. The course consists of deep learning algorithms and its problem settings. The course also covers the idea of representation learning and transfer learning.

Course Outcomes

At the completion of this course, the student shall acquire knowledge and ability to

CO1. Understand the deep learning concepts and be ready to apply it to applications

CO2. Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains

CO3. Incorporate transfer of knowledge in machine learning algorithms

CO4. Implement deep learning algorithms and solve real-world problems

Relationship of Course Outcomes to Program Outcomes
H = High correlation; M = Medium correlation; L = Low correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	H	H						
CO2	M	H	H	M	H	H						
CO3	M	M	H	M	H	H						
CO4	M	H	H	M	H	H						

Syllabus

Module 1:

Basics of artificial neural networks (ANN): Artificial neurons, Computational models of neurons, Structure of neural networks, Functional units of ANN for pattern recognition tasks.

Feedforward neural networks: Perceptron model, Multilayer feedforward neural networks (MLFFNNs), Backpropagation learning, Empirical risk minimization, Regularization

Module 2:

Deep neural networks (DNNs): Difficulty of training DNNs, Greedy layerwise training, Optimization for training DNNs, Newer optimization methods for neural networks (AdaGrad, RMSProp, Adam), Second order methods for training, Regularization methods (dropout, drop connect, batch normalization)

Autoencoders: Autoencoder architecture, dimension reduction using autoencoders, denoising autoencoders, visualization of weights

Module 3:

Convolution neural networks (CNNs): Introduction to CNNs – convolution, pooling, Deep CNNs, Different deep CNN architectures – LeNet, AlexNet, VGG, PlacesNet. Training a CNNs:

weights initialization, batch normalization, hyperparameter optimization, Understanding and visualizing CNNs.

Module 4:

Recurrent neural networks (RNNs): Sequence modeling using RNNs, Back propagation through time, Long Short Term Memory (LSTM), Gated RNN Architecture.

Attention mechanism and Generative models: Encoder-decoder architecture, attention mechanism, transformers. Introduction to generative models - generative adversarial networks (GANs).

Reference Books/Material

1. Ian Goodfellow, Yoshua Bengio and Aaron Courville, *Deep learning,* MIT Press, 2016
2. Charu C. Aggarwal, *Neural Networks and Deep Learning*, Springer, 2018
3. Christopher M. Bishop, Hugh Bishop, "Deep Learning, Foundations and concepts", First Edition, 2023, Springer
4. S. Haykin, *Neural Networks and Learning Machines,* Prentice Hall of India, 2010
   Satish Kumar, *Neural Networks - A Class Room Approach*, Second Edition, Tata McGraw-Hill, 2013
5. B. Yegnanarayana, *Artificial Neural Networks*, Prentice- Hall of India, 1999.
6. C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.