SYLLABI

U.G. Courses offered by the Department of COMPUTER ENGINEERING

REVISED WITH EFFECT FROM 2017-18

DEPARTMENT OF COMPUTER ENGINEERING

ALIGARH MUSLIM UNIVERSITY, ALIGARH

ABOUT THE DEPARTMENT

The Department of Computer Engineering was created in the Faculty of Engineering in the year 1991. The Department is currently running an eight Semester B. Tech. programme in Computer Engineering and a four semester M.Tech programme in Computer Science & Engineering with specialization in Software Engineering. The Department also offers Ph.D. Programme in emerging areas of Computer Engineering.

The Department has the following well equipped laboratories:-

- Software Laboratory
- Hardware Laboratory
- Advanced Computing Laboratory
- Networking Laboratory
- PG Laboratory
- Basic Computing Laboratory

All Laboratories, Staff Rooms and office of the Department are fully networked and have access to high speed University Internet facility.

The Department has highly qualified, motivated and dedicated teaching faculty with education from reputed institutes. They are actively involved in Research and Development activities in the emerging areas of Computer Engineering. The prominent areas of research in the Department are - Computer Networks, Artificial Intelligence & Soft Computing, Web Mining & Information Retrieval, Multimedia Technologies, Software Engineering and Computer Security.

The success of a Department is best judged by the performance of its students. The students find placement in prestigious organizations both at national as well as international level. Three of the students were selected for the elite civil services of Government of India. Students have fared commendably well in various national as well as international level competitions such as GATE, JRF, CAT, GRE, SAT etc. A significant number of students secure admission to higher studies at institutions of international repute.

VISION AND MISSION OF THE INSTITUTE

VISION

To become an institute of excellence in scientific and technical education and research with standards at par with national and international institutes of repute and to serve as quality human resource provider to the society and industry.

MISSION

1. To offer state-of-the-art undergraduate, postgraduate and doctoral programmes.

2. To make policies and atmosphere to attract and retain best faculty.

3. To create an ambience in which new ideas and cutting-edge research flourish through effective curriculum and infrastructure so as to produce the leaders and innovators tomorrow.

4. To produce ethically strong and morally elevated human resource to serve mankind.

5. To undertake collaborative projects and consultancy for long term interaction with the academia and industry.

6. To be among top ten engineering institutes of India.

VISION AND MISSION OF THE DEPARTMENT

VISION

To achieve the status of being one of the best departments in terms of quality of research and technical manpower in the area of computer engineering

MISSION

1. To offer state-of-the-art undergraduate, postgraduate, and doctoral programmes in computer engineering.

2. To provide one of the best working environments to motivate faculty and students to work towards vision of the department and to attract best faculty and students.

3. To develop linkages with industry, other universities/institutes/research laboratories and work in collaboration with them.

4. To use our expertise in computer engineering discipline for helping society in solving problems.

B.TECH

PROGRAMME EDUCATIONAL OBJECTIVES

AND PROGRAM OUTCOMES

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

PEO1. To produce engineering graduates who shall excel in a career utilizing their education in computer engineering

PEO2. To equip graduates with curiosity so that they may continue to enhance their knowledge

PEO3. To produce graduates who are effective in multidisciplinary and diverse professional environment

PEO4. To produce graduates who should provide leadership and demonstrate professional integrity

PROGRAM OUTCOMES (POS)

The students should gain:

(a) an ability to apply knowledge of Mathematics, Science and Engineering

(b) an ability to design and conduct experiments in Computer Engineering, as well as to analyze and interpret data

(c) an ability to design a computer system, component, or process to meet desired needs within realistic constraints

(d) an ability to function on multidisciplinary teams

(e) an ability to identify, formulate, and solve Computer Engineering problems

(f) an understanding of professional and ethical responsibility

(g) an ability to communicate effectively

(h) the broad education necessary to understand the impact of computer engineering solutions in a global, economic, environmental and societal context

(i) a recognition of the need for, and an ability to engage in life-long learning

(j) a knowledge of contemporary issues in computer engineering for managing projects and their financial aspects.

(k) an ability to use the techniques, skills, and modern computer engineering tools necessary for engineering practice.

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. (for all branches)

Course Title — Computer Programming Laboratory

Course Number	**:**	COA1910/CO191
Credits	**:**	1.5
Course Category	**:**	ESA
Pre-requisite(s)	**:**	-
Contact Hours (L-T-P)	**:**	0-0-3
Type of Course	**:**	Practical

Course Objective

To make students of all branches of B.Tech familiar with the programming concepts and to implement the algorithmic approach of problem solving in C language to gain working knowledge of C programming.

Course Outcome

Students will be able to:

1. Understand programming concepts and C language constructs such as operators and data types, control statements, functions etc.
2. Gain algorithm development skills
3. Implement programming problems in C Language

Syllabus

Introduction to Programming Environment, experiments to be conducted in the laboratory consist of, but not limited to, the following:

- Practice of Turbo C as the development environment
- Simple introductory algorithms and programs for getting input, printing formatted output etc.
- Programs introducing elementary C concepts, like variable and names
- Programs using operators
- Programs using control structures
- Programs for repetitive tasks and iterations
- Programs on arrays and strings
- Programs introducing the use of function calls
- Programs introducing basic concept of file handling and storage classes

Books:

1. Kernighan, Brian W., and Dennis M. Ritchie. "The C programming language. "*Prentice-Hall, Englewood Cliffs, New Jersey (I978).*
2. Gottfried "Theory and Problem of Programming with C" Schaum's Outline Series, TMC (Text book)
3. E. Balagurusamy "Programming in ANSI C", McGraw Hill Education India Private Limited (2016) (Text Book)

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x									
2					x						x
3									x		

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title: OBJECT ORIENTED PROGRAMMING

Course Number	**:**	COC2030/CO203
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

The objective of this course is to have students learn basic concepts of object-oriented programming and its difference from procedural programming, implementation of OO concepts and techniques and writing simple application, in both C++ and JAVA.

Course Outcome

Students will be able to:

1. Select suitable programming paradigm for a given problem
2. Learn object oriented programming concepts and features of C++ and Java
3. Apply object oriented concepts in a proposed solution
4. Implement a given solution in C++ and Java

SYLLABUS:

UNIT-I Introduction to Programming

Programming paradigms: Imperative, object oriented, functional and logical programming, Introduction to Object Oriented Programming: Objects, Classes, Encapsulation, Data Hiding, Polymorphism, Inheritance, Abstract Classes, Class Members, Overloading, Overriding

UNIT-II Object Oriented Programming in C++

Program structure, Class, Derived class, Overloading, Template, Exception handling, Scope operator, Virtual function

UNIT-III Some Advanced Concepts in OOP

Abstract Data Types, Pointers, Virtual function, Polymorphism, Generic Programming: Templates: Function Templates, Class Templates, Exception Handling etc.

UNIT-III Object Oriented Programming in Java

Program Structure, Java Architecture, Statements, Classes, Inheritance, Polymorphism, Arrays, API and Packages, GUI in Java

Reference:

1. Robert Lafore; “Object Oriented Programming in C++; Pearson Education
2. Bruce Eckel; “Thinking in Java”; Prentice Hall

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x		x		x						x
2	x	x									x
3			x		x					x	
4		x	x		x					x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title DATA STRUCTURE AND ALGORITHM

Course Number	**:**	COC2060/CO206
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To introduce the concept of data structures including arrays, linked lists, stacks, queues, binary trees, heaps, binary search trees, and graphs etc., and apply these data structures in problem solving. To introduce applications of various data structures and its use in a manner that adds to the efficiency of an algorithm in writing effective programs.

Course Outcome

The student will be able to

1. Learn how the choice of data structures and algorithm design methods impact the performance of programs.
2. Analyze the importance and use of Abstract Data Types (ADTs)
3. Design and implement elementary Data Structures such as arrays, trees, Stacks, Queues, and Hash Tables.
4. Identify algorithms as a pseudo-code to solve some common problems.
5. Explain best, average, and worst-cases of an algorithm using Big-O notation.

Syllabus:

UNIT I : INTRODUCTION
Concept of Data Structures, Basic Terminologies related to data structures, linear and non-linear data structure. Concept and properties of algorithms, How to develop an algorithm, Complexity, Time-Space Tradeoff, Algorithm analysis, Rate of growth: Big Oh notation, other asymptotic notations for complexity of algorithms.

UNIT II: ARRAYS
Arrays, one-dimensional arrays: traversal, selection, searching, insertion and deletion. Sorting: Bubble sort, selection sort, insertion sort, merge sort, quicksort, other sorting methods and their analysis. Multi-dimensional arrays, Representation of arrays in physical memory, Application of arrays.

UNIT III: Abstract Data Types (ADTs)
Abstract Data Types, Stacks, Applications of Stacks - prefix and postfix notations, Queue, Circular Queue, Priority Queue, Deque, Linked Lists, Operations on Linked Lists, Circular linked lists, doubly linked lists, concept of dummy nodes.

UNIT IV: Trees & Graphs
Basic terminologies, Binary Tree, representation and traversal of binary tree; in-order, preorder, and post-order traversal. Different types of binary trees: binary search tree, Heap trees and its application to sorting. Graph, representation and its applications. Other related topics.

Books:

1. Aaron M. Tenenbaum, Langsam "Data Structure using C", Pearson, 2008
2. Lipschutz, "Data structures" Tata McGraw Hill.
3. Goodrich M. Tamassia R., "Data Structures and Algorithms in Java", 3rd ed. Wiley

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1		x			x					x	
2	x									x	
3					x						x
4					x						
5			x								

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title DIGITAL LOGIC AND SYSTEM DESIGN

Course Number	**:**	COC2070/CO207
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To provide introduction to fundamentals of digital logic in order to impart basic knowledge of digital circuits and their logical functions to create digital system.

Course Outcome

Students will be able to:

1. Understand Boolean logic operation, minimization of a Boolean function and its implementation.
2. Analyse and synthesize digital combinational units.
3. Analyse and synthesize digital sequential units.
4. Design and implement simple digital system comprising control unit and data path.

Syllabus

UNIT I. BOOLEAN ALGEBRA AND LOGIC GATES
Introduction, Binary numbers, Base-conversions, Octal and hexadecimal numbers, complements, binary codes, Basic Theorems and properties, Boolean functions and representation in canonical and standard forms, SOP and POS forms, other logic operations, Digital logic gates, Function minimization: Karnaugh map methods, Limitations of K-maps for larger variables, POS-simplification, NAND/NOR implementation, other 2-level implementations, Don't-care conditions, Tabular method.

UNIT II. COMBINATIONAL LOGIC
Hardware aspect of arithmetic logic functions, Half-Adder, Full-Adder, Binary Adder/Subtractor, Parallel Adder, Magnitude Comparator, Demultiplexer, Multiplexer, Parity Checker/Generator, ROM, etc.

UNIT III. SEQUENTIAL LOGIC
Definition and state representation of Flip-Flops, RS, D, JK-M/S, their working characteristics, State Tables, Excitation Tables and triggering. Asynchronous and Synchronous Counters-Design and Analysis, Counter Applications, Description and Operations of Shift Registers, Shift Register/Counters.

UNIT IV. REGISTER TRANSFER LOGIC
Introduction to Register Transfer Logic, macro-operations and micro-operations, introduction to control unit, design and hardware specification of a simple computer, instruction fetch and execute cycle, Introduction to ALU design.

Books:

*1. M. Morris Mano, "Digital Logic and Computer Design", PHI.
*2. Digital Principles and Applications by Leach, Malvino and Saha TMH
3. Digital Design by Mano and Ciletti, Pearson, 5th Ed.

*Text Book

Table: Mapping of Program Outcomes and Course Outcomes for CO-207

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1	x				x						x
2	x	x			x			x			x
3	x	x			x			x			x
4	x		x		x			x	x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — COMPUTER ARCHITECTURE

Course Number	**:**	COC2080/CO208
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	COC2070/CO207
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To know the theory of components of a computer and its operation, and to learn applying the knowledge for designing a minimal computer system.

Course Outcome

Students will be able to:

1. Explain the philosophy of digital systems
2. Interpret the micro-operations
3. Explain and design arithmetic and logic unit
4. Design the control and execution unit to execute a binary instruction
5. Acquire the knowledge of advanced concepts of performance measure and speeding-up of computing

Syllabus

UNIT I INTRODUCTION TO BASIC COMPUTER ARCHITECTURE

Philosophy of digital computers, Method of register, bus and memory transfer, Micro-operations: Arithmetic, Logic and Shift, Introduction to execution of unit-register, ALU design and shift register design, division of unsigned integers, IEEE 754 standard of floating point numbers, Floating point arithmetic operations.

UNIT II MEMORY AND PROCESSOR ORGANIZATION

Introduction to main, cache, auxiliary and virtual memory, concept of address mapping, Instruction and Interrupt cycles, Timing and Control, Design of accumulator logic, Central Processing Unit (CPU): Single accumulator, General register and Stack organization, assembly language, RISC and CISC characteristics.

UNIT III HARDWIRED & MICROPROGRAMMED CONTROL LOGIC DESIGN

Hardwired control design: Computer instruction execution, Design of computer registers, Design of basic computer, PLA Controller. Microprogrammed control design: Basic concepts of Microprogrammed Control Design, Microprogrammed Control design of an example computer, Microprogram sequencer.

UNIT IV ADVANCED CONCEPTS IN COMPUTER ARCHITECTURE

CPU performance measures and Benchmarks, Introduction to pipeline and parallel processor, Parallelism in conventional computers, classification of parallel computers, Pipelining: general consideration and speedup, pipeline structures: arithmetic and instruction pipeline, Introduction to different types of available computers.

Books:

1. *M. Morris Mano “Computer System Architecture” Third edition, Pearson Education India, Pvt. Ltd, reprint 2007
2. *David A. Patterson and John. L. Hennessey “Computer Organization and Design, The Hardware/Software Interface”, 3rd Edition, Morgan Kaufmann Publishers.
3. *Mohamed Rafiquzzaman and Rajan Chandra “Modern Computer Architecture”, Galgotia publication Pvt. Ltd.

*Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1		x	x								
2	x	x			x						x
3		x	x		x						
4	x		x		x						x
5	x		x		x						x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Software Engineering

Course Number	**:**	COC2090/CO209
Credits	**:**	4
Course Category	**:**	DC
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

This course covers practices which are applied during software development. These practices help in developing large size and complex software. With concepts and knowledge gained from this course, one can easily become part of industrial software production.

Course Outcomes

Upon completing the course, the student will be:

1. Familiar with various software development process model**s,** requirement engineering concepts and software design principles.
2. Able to understand software project metrics, quality concepts and estimate effort in software development.
3. Able to understand coding practices, styles and software testing approaches.
4. Able to develop software cooperatively in a team with an understanding about software risk management, configuration management and re-engineering approaches.

Syllabus

UNIT I **INTRODUCTION TO SOFTWARE ENGINEERING AND SOFTWARE PROCESS**

Introduction to Software Engineering: Program Vs Software; Characteristics of Software; Evolution of Software Engineering, Software categories, Software Development life cycle, Software Quality. **Software Development Processes**: Waterfall model, Incremental Models – Iterative Model and RAD Model, Evolutionary Models – Prototype and Spiral Model, Component Based Development; Fourth Generation Techniques, Unified Process.

UNIT-II **SOFTWARE METRICS, REQUIREMENT ENGINEERING AND SOFTWARE PLANNING**

Introduction: Software Measurement and Metrics, Software Quality Concepts**, Requirement Engineering:** Activities and approaches, Software Requirement specification; **Software sizing approaches**: Size oriented metrics, Function oriented metric, and evaluation techniques. **Software Project Planning**: Cost Estimation and Evaluation techniques, Software Team Structure.

UNIT-III **SOFTWARE DESIGN**

Introduction to Software Design: Objectives and Principles, Module level concepts Coupling and Cohesion, Design notation and specification; Architectural Design, Component Level Design, Interface Design; Structured Design Methodology, Design Heuristics, Verification, Concepts of Object Oriented Design.

UNIT-IV **CODING, TESTING AND OTHER TOPICS IN SOFTWARE ENGINEERING**

Coding: Programming practices and styles. **Testing**: Introduction to software testing, Testing Fundamentals, Test cases and test criteria. Black box testing, White box testing: Structural testing, Code Verification, Code Coverage and Cyclomatic Complexity. **Other topics in Software engineering**: Software Risk Management, Configuration Management and Software Re-Engineering.

BOOKS:

1. Roger S. Pressman, “Software Engineering: A Practitioner's Approach”, 7th International edition, Mc Graw Hill, 2009
2. Pankaj Jalote, “An Integrated Approach to Software Engineering”, Narosa publishing House, New Delhi, 1995.
3. Fairley, R.E. “Software Engineering Concepts”, McGraw Hill, 1992.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x		x		x	x			
2	x	x	x		x						x
3	x		x		x						x
4	x	x	x	x	x			x		x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — DESIGN AND ANALYSIS OF ALGORITHM

Course Number	**:**	COC2140/CO314
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	COC2060/CO206
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

The course addresses:

1. Designing algorithms for the problems that are encountered in the area of Computer Engineering.
2. Analysis of time and space complexity of algorithms.
3. Graph algorithms and NP Problems.

Course Outcome

Students who completed the course have demonstrated the ability to:

1. Argue about the correctness of algorithms using inductive proofs and invariants.
2. Analyze running times of algorithms using asymptotic analysis.
3. Use different paradigms such as divide-and-conquer, greedy, dynamic programming, etc.
4. Decide whether a solution to the problem exists in a reasonable amount of time. If no such solution exists, one can consider algorithms that try to solve the problem approximately.
5. Develop efficient algorithms for the problems in the area of Computer Engineering and other engineering disciplines.

Syllabus

UNIT I **INTRODUCTION**
Review of algorithmic analysis, Asymptotic Notations; Solving Recurrences: Substitution methods, Recursion-Tree method and Master method.

UNIT II **ALGORITHM DESIGN TECHNIQUES**
Divide and Conquer Approach, Dynamic Programming, Greedy Algorithms, Greedy versus Dynamic Programming, Miscellaneous algorithmic problems

UNIT III **GRAPH ALGORITHMS**
Breadth First Search, Depth First Search; Minimum Spanning Trees, Shortest Path Algorithms, Dijkstra's Algorithm, The Bellman-Ford Algorithm, The Floyd-Warshall's Algorithm; Maximum Flow; Flow Networks, The Ford-Fulkerson Algorithm.

UNIT IV **NP COMPLETENESS**
P and NP classes, Unsolvable problems, NP Completeness and Reducibility, Circuit Satisfiability, Examples and Proofs of NP-Complete problem.

Books:

*1) Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, Third Edition, September 2009, MIT Press USA.

2) Alfred V. Aho, John E. Hopcroft, Jeffery D. Ullman, "The Design and Analysis of Computer Algorithms", Addison-Wesley, 2001.

3) John Kleinberg, Eva Tardos, "Algorithm Design", Addison Wesley

4) Ellis Horowitz, Sartaj Sahni, "Fundamentals of Computer Algorithms", Galgotia Publications, 2001.

*Text Book

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1		x			x						
2	x										
3		x			x						x
4					x						
5	x	x									x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — DATA STRUCTURE AND OOP LABORATORY

Course Number	**:**	COC2910/CO291
Credits	**:**	2
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	0-1-2
Type of Course	**:**	Practical

Course Objective

The objective of this course is to learn the application of data structure in problem solving and understand the importance of object oriented features and their applications.

Course Outcome

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

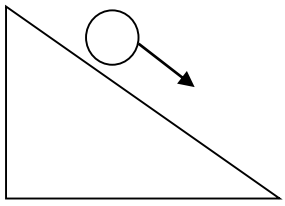
1. Design, implement and debug programs in C, C++ and Java and be able to select suitable language for a given problem
2. Design and implement data structures such as linked list, stack, tree, graphs etc. for selected solution of a given problem.
3. Apply object oriented concepts and develop object oriented solution for a given problem

Syllabus:

Programs covering following topics:

Sample List of Experiments for the Lab

1. Write a program to create and display linked list
2. Write a program to merge two linked list
3. Write a program that construct binary search tree for a given list of numbers.
4. Write a program to multiply arbitrary large numbers
5. Write a program to design a class for queue data structure
6. Write a program to overload operator for string concatenation
7. Simulate the movement of a ball if it is set free on an inclined plane (see figure).



8. Develop an object oriented solution to implement a dictionary which stores words in alphabetical order. Program takes input a word, if that word exists in the dictionary show the message ‘duplicate entry’ otherwise save that word at its correct position.
9. Construct a telephone diary using object oriented concepts. The diary contains name of person, phone number (s), Nickname, group, email address and postal address. A person can have more than one phone numbers. Perform the following operations on the telephone diary:

I. Store a new person detail
II. List the person details belonging to a given group
III. List all the phone numbers of a given person name

10. There are 16 players in a knockout tennis tournament. Players are numbered 1 through 16. In the first stage, each odd number player n plays with player number n+1. In the later stages, winner of a match plays with the winner of the next match. Simulate the tournament. Use random number generator to identify the winner in a match.

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x		x	x	x				x
2	x	x	x		x	x	x				x
3	x	x	x		x	x	x				x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title: DIGITAL DESIGN AND SIMULATION LAB

Course Number	**:**	COC2920
Credits	**:**	2
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	COC2070/CO207
Contact Hours (L-T-P)	**:**	0-1-2
Type of Course	**:**	Lab

Course Objective

Analysis, Design and Synthesis of digital circuits on hardware and its simulation using various tools.

Course Outcome

Students who completed the course have demonstrated the ability to:

1. Understand hardware implementation of simple combinational and sequential circuits.
2. Understand simulation studies of complex digital circuits (Finite State Machine and Combinational circuits).
3. Differentiate between implementation of Moore and Mealy machines.
4. Understand the importance of timing as well as states of FSM

Syllabus

Students are expected to implement various digital circuits. The outlines of the course are as follows.

Awareness of different logic gates and its usage. (AND, OR, NAND, NOR, XOR, XNOR etc.).

Use of logic gates to implement different combinational logic circuits. (MUX, Half Adder, Full Adder, Adders/Subtractors).

Use of logic gates and memory to implement different sequential circuits. (Counters, Registers, Sequence detectors).

Design and implementation (hardware/simulation) of simple control units.

Introduction to VHDL structural Modeling Styles. Design and simulation of simple circuits using VHDL/Verilog.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x	x			x				x
2	x	x	x		x		x				x
3	x	x	x		x	x	x		x		x
4	x	x	x	x	x	x	x		x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title DIGITAL ELECTRONICS

Course Number	**:**	COC3080/CO308
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	ELA1110/EL111, ELA2110/EL211
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

This course is designed to teach students the theory of digital electronics, logic and the implementation of modules required for digital systems with different cost criteria, digital integrated circuits used for building digital systems. It includes the description and the structure of various fundamental components of digital systems like logic gates, memory cells, interfacing units etc.

Course Outcome

Upon successful completion of this course, students should be able to:

1. Explain the structure and fundamental components of digital systems at transistor level with different design criteria.
2. Demonstrate a clear understanding of important concepts and logic family.
3. Design and explain different types of memory: RAM, ROM, PLA, PAL etc.
4. Describe the fundamental architecture of analog-digital interfacing units such as data converters, data acquisition system etc.

Syllabus:

UNIT I: IC LOGIC FAMILIES

Digital IC Terminology and Logic Circuit Families; MOS and CMOS Logic Circuits and Characteristics; CMOS Inverter, NAND, NOR, X-OR, X-NOR Gates; CMOS Complex Gates; CMOS Transmission Gate; CMOS Clocked S-R and D-Flip-Flops; Pseudo NMOS Logic Circuits; Pseudo NMOS Inverter and other Gates; Pass Transistor Logic (PTL) and Complementary Pass Transistor Logic (CPTL); Realization of Different Gates in PTL and CPTL; Bi-CMOS Digital Circuits; Bi-CMOS Inverter and Logic Gates; Comparison of various Logic Families.

UNIT II: MEMORY DEVICES

Memory Terminology, General Memory Operation; CPU Memory Connections; Semiconductors Memories; Types and Architecture; Memory Chip Organization; ROM -- Architecture, Addressing and Timing; MOS ROM; PROM, EPROM EEPROM (EAPROM), CD-ROM; ROM Applications; Programmable Logic Device Arrays (PAL and PLA); ROM/PLD Based Combinational Design; Semiconductor RAM -- RAM Organization; Static RAM, Dynamic RAM; DRAM Structure and Operation; Read/Write Cycles; DRAM Refreshing; Expanding Word Size and Capacity; Sequential Memories.

UNIT III: DYNAMIC LOGIC CIRCUIT AND MEMORY CELLS

Dynamic Logic Circuits; Basic Structure of Dynamic-MOS Logic Circuits; Cascading Dynamic Logic Gates; Dynamic CMOS Ratio-Less Shift Register Stage; Domino CMOS Logic; CMOS SRAM Memory Cell; One Transistor Dynamic RAM Cell; Differential Voltage Sense Amplifier and Address Decoders for SRAM and DRAM; Charge Coupled Device (CCD); Basic Operation of 3-Phase and 2-Phase CCD, CCD Memory Organization.

UNIT IV: INTERFACING WITH THE ANALOG WORLD

Principal of operation of Digital-to-Analog Converters (DACs); Basic Circuits Using Binary Weighted Resistors and R/2R Ladder; DAC specification; DAC Applications; Analog to Digital converters (ADCs); Digital Ramp ADC; Up/Down Digital Ramp ADC, (Tracking ADC), Successive Approximation ADC; Flash ADC, Dual Slope Integrated ADC, Data Acquisition, Sample and Hold Circuits: Multiplexed ADC.

Books:

*1. Ronald. J. Tocci, And Neal. S. Widmer, "Digital Systems – Principles And Applications", Eighth Edition, Pearson Education, New Delhi, 2001.

2. A.S. Sedra and K. C. Smith, "Microelectronic Circuits", Oxford University Press, Vth Edition, 2004.

3. J. Millman and Grabel, "Microelectronics", McGraw Hill, 1987.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1		x	x			x					x
2					x						
3		x	x							x	
4			x								x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — MICROPROCESSOR THEORY AND APPLICATION

Course Number	**:**	COC3090/CO309
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	COC2070/CO207, COC2080/CO208
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To develop an in-depth understanding of the operation of microprocessors, assembly language programming and interfacing.

Course Outcome

Upon successful completion of this course, students should be able to:

1. Understand the instruction set architecture of 8085 microprocessor, machine cycles, and address decoding for memory and I/O interface.
2. Understand instruction set architecture of 8086 microprocessor.
3. Write and analyze assembly language programs in 8085 and 8086.
4. Learn the concept of procedures, interrupt, and interrupt service routine.
5. Explain Input and output modes and interfacing.

Syllabus

UNIT 1 **INTRODUCTION TO 8085 MICROPROCESSOR**
Introduction to microprocessor, microcomputer and assembly language, 8085 microprocessor architecture, 8085 based microcomputer, overview of 8085 instruction set, addressing modes. 8085 assembly language programming.

UNIT II **BUS TIMINGS AND INTERFACING**
8085 machine cycles and bus timings. Timing Diagrams of Opcode Fetch, Memory read/write machine cycle etc. Address decoding to interface memory and I/O. Memory mapped I/O and I/O mapped I/O.

UNIT III **x86 ARCHITECTURE AND PROGRAMMING**
Overview of 8086 microprocessor family. Internal architecture of 8086, addressing modes, instruction set of 8086, assembler directives, assembly language programming of x86 processors (analysis and coding), procedures and macros.

UNIT IV **INTERRUPTS AND PERIPHERAL INTERFACING**
Interrupts, Interrupt service routines, interfacing peripheral devices with microprocessor. Introduction to 8253/8254 Programmable Timer/Counter, 8259 Priority Interrupt Controller, 8255 Programmable Peripheral Interface, and other peripheral devices.

BOOKS:

*1. R.L Gaonkar- "Microprocessor Architecture, Programming and Applications", Penram International Publishing India Pvt Ltd, 5th edition.
*2. Doughlas V. Hall and SSSP Rao "Microprocessors and Interfacing- programming and hardware", McGraw Hill Education, 3rd edition.

3. Barry B. Brey- “The Intel Microprocessors, Architecture Programming and Interfacing”,- Prentice Hall International.

* Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x		x		x						
2	x		x		x						
3	x	x	x		x				x		x
4	x		x		x				x		x
5			x		x				x	x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — OPERATING SYSTEMS

Course Number	**:**	COC3100/CO310
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	COC2080/CO208
Contact Hours (L-T-P)	**:**	4 (3-1-0)
Type of Course	**:**	Theory

Course Objective

Consider and address the issues in the design of modern operating systems. The course explains the main components of modern operating system and their organization schemes. It discusses various issues and problems in the design and implementation of main components-process management, Main Memory Management, File System management, Secondary Storage Management; and methods, algorithms to solve them efficiently. Students will learn modern OS concepts through examples of Linux and Windows XP Operating Systems.

Course Outcome

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

1. Grasp a fundamental understanding of goals, components and evolution of operating systems
2. Learn the concepts of processes, scheduling policies, critical section problem and synchronization
3. Understand Paging, segmentation and virtual memory concepts in modern OSs
4. Understand the concepts of data input/output, storage and file management
5. Understand the main design concepts of Linux and Windows OS

Syllabus:

UNIT I **INTRODUCTION**
Evolution of Operating Systems- Simple Batch Systems, Multiprogramming system, Time-sharing systems, Parallel & Distributed systems, Real Time system; Operating System Structure: System components, Services, Interface to user & programs, concept of virtual machines, System Structure.

UNIT II **PROCESS MANAGEMENT**
Process states, control block, operations on process, Cooperating process, Concepts of Threading, IPC, Process Scheduling- Basic concept, Type of schedulers, Scheduling criteria, scheduling algorithm, Multiple processor scheduling, Real time scheduling; Process Synchronization: Critical section problem, synchronization hardware, Semaphores, classical problem of synchronization, Monitors.

UNIT III **DEADLOCK AND MEMORY MANAGEMENT**
Deadlock- Characterization, Methods of handling: Deadlock prevention, Deadlock avoidance, Deadlock detection & Recovery; Memory Management- Contiguous allocation, Swapping, Non Contiguous allocation: Paging, Segmentation, Virtual Memory- Basic Concepts, page allocation & replacement policies, Thrashing.

UNIT IV **STORAGE MANAGEMENT & CASE STUDIES**
File System- Concept, Access Methods, Directory, Protection, Implementation of File systems; Storage Structure- Overview, Disk Scheduling, Disk Management, Swap Space Management, Case Studies- Linux, Windows XP.

Books:

*1. Peterson: Silberschatz, Galvin “Operating System Concepts”, Addison Wasely, Vth Addition.
2. William Stallings, “Operating Systems”
3. Milenkovic, Milan: Operating system concepts and Design, McGraw Hill.

*Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x							x	
2	x	x	x		x					x	x
3	x		x		x						
4	x		x		x					x	x
5	x										x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — THEORY OF COMPUTATION

Course Number	**:**	COC3110/CO311
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	AMS2630/AM263, COC2070/CO207
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To build a strong theoretical background for computer science, and to know practical implications of the theory

Course Outcome

Students will be able to:

1. Explain and classify mathematical models for representing finite state systems
2. Design various types of automata and write regular expressions for regular languages, and interconvert automata and regular expressions
3. Explain, analyze, and design context free grammars for context free languages
4. Design and analyze Turing machines
5. Interpret and explain the limits of computing

Syllabus

UNIT I **INTRODUCTION TO FINITE AUTOMATA & REGULAR LANGUAGES**
Strings, Alphabets and Languages, Graphs & Trees, Sequential machine, State tables & diagram, Mealy & Moore machines, State and Machine equivalence. Deterministic and Non-deterministic Finite Automata, Regular Expressions, Regular grammar, Minimization of DFA, Pumping Lemma for Regular sets, Properties of Regular Languages.

UNIT II **CONTEXT FREE GRAMMARS & LANGUAGES**
Context free Grammar, Chomsky Normal form and Greibach Normal form, Pushdown Automata, Context Free languages, Chomsky Classification of languages, Pumping Lemma for context free languages, properties of context free languages.

UNIT III **TURING MACHINES**
Turing Machines, Computing with Turing Machines, Nondeterministic Turing Machines, Church's Thesis, Universal Turing Machines.

UNIT IV **UNCOMPUTABILITY & COMPUTATIONAL COMPLEXITY**
Halting Problems, Unsolvable Problems about Turing Machines, Time bound Turing Machines, The Class P and NP Languages, NP Completeness, Some NP Complete Problems.

Books:

* 1. J.E. Hopcroft, Motwani & J.D. Ullmann, "Introduction to Automata Theory, Languages and Computation", Narosa Publications
2. H.R. Lewis & C.H. Papadimitrou, "Elements of the Theory of Computation", PHI

3. John C. Martin, "Introduction to Languages and the Theory of Computation", McGraw-Hill International
4. Michael Sipser , "Introduction to the Theory of Computation ",Thomson Learning, PWS publishing company
5. D.A. Cohen , "Introduction to Computer Theory", John Wiley
6. Zvi Kohavi, "Switching and Finite Automata Theory", Tata McGraw-Hill

(*Text Book)

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x		x		x				x
2	x	x	x		x		x			x	x
3	x	x	x		x		x			x	x
4	x	x	x		x		x				x
5	x	x	x		x		x				x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — DATA BASE MANAGEMENT SYSTEM

Course Number	**:**	COC3120/CO312
Credits	**:**	4
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	AMS2630/AM263, COC2060/CO206
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

This course will provide students with a general overview of databases, introducing you to database history, modern database systems, the different models used to design a database, and Structured Query Language (SQL), which is the standard language, used to access and manipulates databases.

Course Outcome

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

1. Compare and contrast the database approach and the file system approach.
2. Explain what a database management system is as well as their components and models.
3. Identify how relational algebra / relational calculus is used to construct queries for data definition commands and data manipulation commands in SQL.
4. Understand and apply the process of normalization and design normalized relations.
5. Understand what tables, indexes, and views are as well as their importance and effect.

Syllabus

UNIT I **INTRODUCTION TO DATA BASE CONCEPTS**
Data Base Systems and their needs, Components of DBS, DBS architecture, Data Base Administrator and his role, DBMS and its Components, Data Models – Classification, Physical Model- Storage structure, Indexing Techniques; Conceptual Modeling- E/R Model.

UNIT II **RELATIONAL MODEL**
Relational Model – Definitions, Relational algebra and calculus, Integrity; SQL, Database Programming- Embedded SQL, Dynamic SQL, PL/SQL.

UNIT III **NORMALIZATION, RECOVERY & CONCURRENCY**
Concept of Normalization- Functional Dependencies (FD), First, Second and Third normal forms, BCNF, Fourth & Fifth normal forms; Recovery- Transaction, Commit and Rollback, Sync. Points, Systems and Media recovery; Concurrency- problems, Locking, Serializability, Deadlock.

UNIT IV **ADVANCE TOPICS IN DATABASES**
Security- General considerations, Security in SQL, Views, Grant and Revoke mechanism; Query optimization, Relational Database System Design, Data Mining, Data Warehousing, Parallel & Distributed Data Bases.

Books:

1. Date, C. J. : "Introd. To Data Base Systems", Addison Wesley 6th ed.
2. Elmasri & Navathe “Fundamentals of Database Systems” 5th Edition
3. Abraham Silberchatz, "Data Base System Concepts", McGraw Hill Int.
4. Ullman, "Principles of Database Systems". Galgotia Pub.
5. Ivan Bayross. "Oracle Developer 2000", B. P. B. Publication

* Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x				x		x				
2							x				x
3		x				x	x	x			x
4					x	x	x				x
5			x		x						

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — COMPUTER NETWORKS

Course Number	**:**	COC3130/CO313
Credits	**:**	4
Course Category	**:**	DC
Pre-requisite(s)	**:**	COC2080/CO208, ELA3400/EL340
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

After taking this course the students should be able to:

1. Understand protocol layers and their service models together with delays and losses in networks.
2. Understand principles, architectures, and protocols involved in network applications.
3. Design reliable data transfer protocols and understand congestion control and flow control mechanisms.
4. Understand architecture of a router, routing algorithms, protocols at the network layer, and link layer services and protocols.

Course Outcome

Students who completed the course have demonstrated the ability to:

1. Understand protocol layers and their service models together with how to compute delays and losses in networks.
2. Understand principles, architectures, and protocols involved in network applications.
3. Design protocols for reliable data transfer and understand congestion control and flow control mechanisms.
4. Understand architecture of a router, routing algorithms, protocols at the network layer, and link layer services and protocols.

Syllabus

UNIT I INTRODUCTION

Introduction to Computer Networks, Edge and Core of the Network, Circuit Switching and Packet Switching, Protocol Layers and Their Service Models, Delays and Losses in Packet Switched Networks.

UNIT II APPLICATION LAYER

Principles of Network Applications, Application Layer Services and Protocols, the Web and HTTP, File Transfer: *FTP*; Electronic Mail in the Internet: *SMTP, IMAP, POP*; Domain Name System (DNS), Peer-to-Peer Applications.

UNIT III TRANSPORT LAYER

Transport Layer Services, Connectionless Transport: *UDP*; Principles of Reliable Data Transfer: *Go-Back-N and Selective Repeat*, Connection-Oriented Transport: *TCP*; Principles of Congestion Control, TCP Congestion Control.

UNIT IV NETWORK & LINK LAYER

Virtual Circuit and Datagram Networks, The Internet Protocol: Forwarding and Addressing, Architecture of a Router, Routing Algorithms: *Link-State, Distance Vector and Hierarchical Routing*, Routing in the Internet: *RIP, OSPF, BGP*. Link Layer Services, Multiple Access Protocols, Link-Layer Addressing: *MAC Addresses, ARP, DHCP*, Ethernet.

Books:

*1. J.F. Kurose, K.W. Ross, Computer Networking: A Top-Down Approach Featuring the Internet, 7th Edition, Pearson Education, 2017.

2. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, 5th Edition, Pearson Education, 2012.

3. K.S. Trivedi, Probability and Statistics with Reliability and Queuing and Computer Science Applications, Second Edition, John Wiley, 2002.

*Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x				x						x
2	x				x					x	
3	x		x		x						
4	x				x						x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Computer Graphics

Course Number	**:**	COC3150/CO315
Credits	**:**	4
Course Category	**:**	DC
Pre-requisite(s)	**:**	AMS1110/AM111, AMS1120/AM112
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

This objective of this course is to introduce the fundamentals of computer graphics and its underlying principles. The course aims to explain a number of algorithms that are employed to create the most basic objects in order to render a graphical scene.

Course Outcomes:

At the end of this course, the student is expected to:

1. Define computer graphics and know recent advances, display devices and processing units
2. Derive various algorithms to draw lines and other graphics primitives
3. Understand 2D and 3D viewing and geometric transformations; and classify transformation techniques
4. Learn object representation and techniques for achieving realism; design an illumination scene based on ambient, diffused and specular lighting

Syllabus

UNIT I **INTRODUCTION TO COMPUTER GRAPHICS & GRAPHICS HARDWARE**
A Survey of Computer Graphics, Video Display Devices – Raster scan and Random scan displays, Display processing units, graphics input devices, Graphics Software packages and standards.

UNIT II **POINT PLOTTING TECHNIQUES**
Coordinate systems, Line drawing algorithms: DDA, Bresenham's line drawing Algorithm, Mid-Point Circle Algorithm, Maintaining geometric properties of displayed objects, Fill area primitives, polygon fill areas, output primitives in OpenGL, point, line and character attributes, color models, antialiasing

UNIT III **GEOMETRIC & VIEWING TRANSFORMATIONS**
Basic 2-D Geometric Transformation: Translation, Rotation and Scaling, Matrix representation of transformation in homogenous co-ordinates, Composite transformations, Reflection, Shear, Geometric transformations in 3-D space. 2-D viewing, Clipping window; Normalization and viewport transformation, Clipping algorithms: Cohen-Sutherland line clipping algorithm etc; Sutherland-Hodgman Polygon clipping algorithm, Weiler and Atherton's algorithm etc., 3-D viewing Concepts: Projection Transformations, Orthogonal, Oblique parallel and Perspective projections.

UNIT IV **OBJECT REPRESENTATION AND SURFACE RENDERING**
Polyhedra, Curved surfaces, spline representations, Cubic spline interpolation methods, Bezier and B-Spline Curves, Visible surface detection algorithms: depth buffer, A- Buffer Scan-Line, depth sorting, Ray casting methods etc. Basic illumination models, Diffuse Reflection, Specular Reflection, Gouraud and Phong Surface Rendering, Ray Tracing Methods, Interactive Picture construction techniques: dragging, rubber banding etc.

Books:

1. Hearn & Baker, “Computer Graphics with OpenGL” 3rd Ed., Pearson education *
2. Foley, Dam, Feiner and Hughes “Computer Graphics: Principle and Practice”, 2nd Ed., Pearson Education
3. Edward Angel, “Interactive Computer Graphics: A Top-Down Approach Using OpenGL, 5/E, Addison-Wesley.
4. Neuman & Sproull, “Principle of Interactive Computer Graphics”, McGraw Hill Int.

***Text Book**

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1	x						x			x	
2		x	x				x				
3					x				x		
4	x		x							x	

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title: ALGORITHMS AND OPERATION SYSTEMS LAB

Course Number	**:**	COC3930/CO393
Credits	**:**	2
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	0-1-2
Type of Course	**:**	Lab

Course Objectives

This course is helpful in making students understand simple algorithms and operating system concepts. This course also provides them with an opportunity to implement the algorithms and trace errors while building software.

Course Outcomes

At the end of this course, the students will be able to

1. Design simple algorithms using data structures
2. Understand the procedures defined for different operating system tasks
3. Analyze programs in order to understand their functioning
4. Apply various programming concepts to create simple software
5. Interpret various errors encountered in software development process and trace their causes.

Syllabus/ List of experiments

1. Write a program to implement the Breadth First Search (BFS) algorithm.
2. Write a program to implement the Depth First Search (DFS) algorithm.
3. Write a program to implement Prim's algorithm using Fibonacci heap for finding minimum spanning tree (MST) in a graph
4. Write a program to implement Dijkstra's single source shortest path algorithm.
5. Write a program that draws a polygon with a given set of points as corners and tests whether a given point lies within the polygon or outside the polygon or on the boundary of the polygon.
6. Write a program to find a path from a given source cell to a given destination cell in a square matrix of cell.
7. Write a program to implement the given process scheduling algorithms.
8. Write a program to implement the given page replacement algorithms.
9. Create software that implements various disk scheduling algorithms.
10. Analyze the given program/software in order to understand its functioning and bring the required changes in its functionalities.

Text Book(s)/ Reference Book(s)

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, 7th Edition, Wiley
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Third Edition, September 2009, MIT Press USA.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x		x			x			
2	x	x	x		x						x
3	x				x						x
4	x	x	x		x						x
5	x	x			x			x			

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title MICROPROCESSOR LAB

Course Number	**:**	COC3940/CO394
Credits	**:**	2
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	COC3090/CO309
Contact Hours (L-T-P)	**:**	0-1-2
Type of Course	**:**	Lab

Course Objective

To develop a practical understanding of machine- and assembly-language programming 8085 and 8086/80386 microprocessors and interfacing I/O devices with the 8085.

Course Outcome

Students will be able to:

1. Design device-independent algorithms for machine/assembly language programs
2. Implement and execute assembly language programs on 8085 microprocessor kit
3. Interface various digital devices (I/O Interfacing) with 8085 microprocessor kit
4. Implement assembly language programs for 16/32 bit Processors (8086/80386) using a suitable integrated development environment (IDE).
5. Interfacing C programs with Assembly Language

Syllabus

1. Introduction to 8085 microprocessor kit and basic programs related to registers and memory.
2. Perform arithmetic operations on 8-bit numbers.
3. Perform the arithmetic operations on n-bit string of numbers.
4. Perform searching and sorting of numbers in an array.
5. Writing programs using subroutine and stack.
6. Programs on I/O interfacing devices using 8085 kit.
7. Introduction to Visual studio for 8086/80386 assembly language program.
8. Some complex programs on 8086/80386 assembly language.
9. C Program integration with assembly language.

BOOKS:

1. R.L Gaonkar- “Microprocessor Architecture, Programming and Applications”, Wiley Eastern Ltd.
2. Douglas V. Hall & SSSP Rao- “Microprocessors & Interfacing”, (TMH Publication, 3 rd Edition, 2012)

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x		x	x				x		
2		x	x	x	x				x	x	x
3	x	x	x	x	x				x	x	x
4	x	x	x	x	x				x	x	x
5		x	x	x	x				x	x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Minor Project Lab

Course Number	**:**	COC3950/CO395
Credits	**:**	3
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	COC2090/CO209
Contact Hours (L-T-P)	**:**	6 (0-0-6)
Type of Course	**:**	Practical

Course Objective

The objective of this lab is to make the students understand the software engineering paradigm by working on a minor project.

1. Familiarize the students with software development life cycle stages
2. Learn to apply the design methodologies
3. Learn to implement, test and deploy the project efficiently
4. Develop the understanding and use of some application development environment like .Net, JVM, Eclipse etc
5. Develop the ability to work in a team

Course Outcome

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

1. Apply software development practices in projects
2. Express project requirements in IEEE SRS format
3. Design using standard design methodologies
4. Implement the project using a programming language and platform
5. Test and debug the project

Syllabus:

List of some sample Minor projects completed in the Lab

- Implementation of MOST architecture for neural network design optimization
- Online examination system
- Social networking website for information sharing
- Identify trends in data using HADOOP
- Auditing software for database
- Promotional website for small scale business
- Online job search website
- Network communication on Android
- Audio/video file splitter
- Adaptive e-test
- TPO Management
- Mobile application for course management
- Dynamic news alert application on android
- Blog website for AMU
- An improved Steganography scheme
- Adaptive CPU scheduling algorithm

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x			x						x
2	x	x	x	x	x		x			x	x
3	x	x			x	x				x	x
4				x	x		x				
5		x	x			x					x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — ADVANCED COMPUTING LAB

Course Number	**:**	COC3960/CO396
Credits	**:**	2
Course Category	**:**	DC
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	0-1-2
Type of Course	**:**	Laboratory

Course Objectives

This course is helpful in making students learn advanced tools for software development. The student is expected to learn programming and debugging tools on windows and non-windows platforms such as Linux/Unix.

Course Outcomes

At the end of this course, the student will be able to:

CO1. Perform programming on the Linux platform.
CO2. Understand programming tools for software development.
CO3. Apply software development tools to debug software.
CO4. Develop algorithms that satisfy various constraints.
CO5. Analyze software behavior to isolate anomalies.

Syllabus/ List of experiments

1. Write a program to solve Dining Philosopher problem with 5 philosophers using semaphores. There are two pairs of Red forks and Spoons and Two pair of Green forks and Spoons. A philosopher can pick similar colored forks and spoons.
2. Write a program to find second shortest path between every pair of nodes in a Graph using threads. The number of threads will be provided by the user.
3. Write a program to find spanning tree for a given graph. Also from the removed edges find whether any other spanning tree is possible. List the other possible spanning tree.
4. Develop an Android Application for student attendance management system. (2 weeks)
5. Install a server and use a database to develop a Web application.
6. Write a program that uses Socket programming to play the tic tac toe game on the Linux platform. Both players will use separate machines.
7. Write a shell script to create two folders ‘old’ and ‘new’, also determine which file have been created earlier than a given date and also later than the given date. Place the files by maintaining their hierarchy in the newly created folders.
8. Implement a game in OpenGL (eg: Tic Tac Toe, Snakes & Ladders) etc. (2 weeks)
9. Write the unit tests for the game (using GTest in C++) and appropriate assertions in functions.

Text Book(s)/ Reference Book(s)

3. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, 7th Edition, Wiley
4. Richard Blum, Linux Command Line And Shell Scripting Bible 2nd Edition, Wiley India, 2011
5. Elliot Rusty Harold, Java Network Programming: Developing Networked Applications, 4th Edition, O’Reilly Media, 2013

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x			x						x
2	x							x		x	
3	x				x					x	x
4					x					x	x
5	x	x						x			

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — COLLOQUIUM

Course Number	**:**	COC3800/CO380
Credits	**:**	2
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	EZH1110/EZ111, EZH2910/EZ291
Contact Hours (L-T-P)	**:**	0-2-0
Type of Course	**:**	Seminar

Course Objective

To promote students for research and innovative work and to develop skills to present technical topics in front of high technical audience.

Course Outcome

On successful completion of the course the students will:

1. Develop their skills at giving oral research presentations and evaluating such presentations
2. Be able to write a short report upon the presentations delivered
3. Be aware of current researches in various field of computer science & engineering.
4. Be a practicing professional and sustain in academic and corporate world.

Syllabus

The colloquium is structured in manner that there is a suitable mix of presentations of the students on diversified areas of computer engineering. The colloquium will be coordinated by a faculty of the department.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x						x				
2				x				x	x		
3				x						x	
4									x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — INFORMATION SECURITY

Course Number	**:**	COC4010
Credits	**:**	4
Course Category	**:**	DC (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To develop an understanding of information assurance as practiced in computer operating systems, distributed systems, networks and representative applications. Develop a basic understanding of cryptography, how it has evolved, and some key encryption techniques used today. Develop an understanding of security policies (such as authentication, integrity and confidentiality), as well as protocols to implement such policies in the form of message exchanges.

Course Outcome

After completion of this course, the student shall be able to:

1. Describe computer and network security fundamental concepts and principles.
2. Describe the inner-workings of popular encryption algorithms, digital signatures, certificates and anti-cracking techniques.
3. Demonstrate the ability to select among available network security technology and protocols such as IDS, IPS, firewalls, VPNs etc.
4. Describe Operating system security models E.g. UNIX, LINUX WINDOWS security and methods for web application and database security.
5. Identify ethical, professional responsibilities, risks and liabilities in computer and network environment, and best practices to write a security policy.

Syllabus

UNIT I **INTRODUCTION:**
Need and basic goals for computer security, security threats etc. Cryptographic building blocks: symmetric and asymmetric key cryptography. Some symmetric key algorithms.

UNIT II **PUBLIC-KEY ENCRYPTION & HASH FUNCTIONS:**
Public key cryptography RSA, Message Authentication, cryptographic hash functions, security of hash functions, Key management Deffie –Hellman key exchange algorithm, digital signature schemes etc., with representative applications for each.

UNIT III **OPERATING SYSTEM SECURITY:**
Low-level protection mechanisms, access control: models for access control, some confidentiality, integrity, and hybrid models of access control such as Bell-LaPadula, Biba, Chinese Wall etc., discretionary v/s mandatory access control. Program flaws: bugs which have security implications such as buffer overflows, race conditions etc. Malicious code: viruses, worms, Trojan horses; how they work and how to defend against them.

UNIT IV **NETWORK SECURITY:**

Problems in network security; kinds of attacks, PKI, key exchange protocols, example protocols such as PGP, Kerberos, IPSEC/VPN, SSL, S/MIME etc. Protocol vulnerabilities: examples of protocol vulnerabilities such as in TCP/IP, denial of service attacks etc. Tools for network security such as firewalls and intrusion detection systems.

Books

- Stallings, W. Cryptography and Network Security, Principles and Practice, 4th edition, Pearson Education , 2005.
- Network Security The Complete Reference, TATA McGraw Hill Publication., 2004
- Tanenbaum, A.S., Computer Networks, 4th edition, prentice Hall
- Stinson, D., Cryptography. Theory and practice, 2nd edition, CRC Press
- Network Security Private Communication in Public World, C. Kaufman, R. Perlman, M. Spicier, 2nd edition PHI.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x										
2		x			x						
3									x		
4			x								x
5						x					

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Compiler Design

Course Number	**:**	COC4060/CO406
Credits	**:**	4
Course Category	**:**	DC
Pre-requisite(s)	**:**	AMS2630/AM263, COC3110/CO311
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

This course aims to familiarize students with compiler techniques. It covers how to build compiler of programming languages. It discusses important language issues, their syntax verification and translation to machine language. But these techniques also find application in several other problems such as string parsing, translator etc.

Course Outcomes

Upon completing the course, the student will:

1. be familiar with the techniques used in lexical analysis, syntax analysis, semantic analysis, intermediate code generation, code optimization and code generation
2. be able to build compiler
3. be able to apply compiler techniques in other domain such as in building translator, parser etc.

Syllabus

UNIT-I Introduction to Compiler & Lexical Analysis

Introduction, Structure of Compiler, Elements of Programming Language, Grammar, Derivation, Syntax Tree, Parse Tree, Ambiguous Grammar, Symbol Table, Lexical Analysis – Specification and Recognition of Tokens, Lookahead Operator, Lexical Errors.

UNIT-II Syntax Analysis and Syntax Directed Translation

Syntax Analysis – Top Down Parsing, Predictive Parser, Bottom Up Parsing, SLR, Canonical and LALR Parsing, Error Handling in Top Down and Bottom Up Parsing, Syntax Directed Definition, Synthesized and Inherited Attribute, S-Attributed Definition, L-Attributed Definition.

UNIT-III Intermediate Code Generation

Type Checking, Intermediate Code Forms, Intermediate Code Generation for Arithmetic Expression, Boolean Expression, if-then-else, goto, while statements etc. Run-Time Environment

UNIT-IV Code Generation & Code Optimization

Issues in the design of Code Generator, Basic Blocks and Flow Graph, Register Allocation and Assignment. Sources of Optimization, Optimization of Basic Blocks, Global Data Flow Analysis – Reaching Definition, Available Expression, Live Variable etc. Loops in Flow Graph.

REFERENCES

1. Aho, Lam, Sethi, Ullman, “Compilers : Principles, Techniques and Tools”, Pearson Education.
2. Aho, Ullman, “ Principles of Compiler Design”, Narosa Publising House.
3. Steven S. Muchnick, “Advanced Compiler Design Implementation”, Harcourt Asia Pte Ltd.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1		x	x								x
2	x	x	x		x						x
3	X				x						x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — NETWORK AND EMBEDDED SYSTEM LAB

Course Number	**:**	COC4950/CO495
Credits	**:**	2
Course Category	**:**	DC (Departmental Core)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	0-1-2
Type of Course	**:**	Lab

Course Objective

The objective of this course is:-
a) To design and simulate a network and configure various routing protocols.
b) To carry out experiments related to embedded systems using VHDL.

Course Outcome

Students who completed the course are able to:

1. Design a network and run various network protocols.
2. Carry out simulation study of various network protocols.
3. Understand the design of subsystem using HDL.
4. Design and simulate a complex digital system, using hierarchical design methodology.

Syllabus

Students are expected to perform the following sample experiments.

1. Static Routing: Consider a network with 3 routers and 2 switches. Design a network such that 3 LANs are connected to 3 routers and there is a WAN link between 2 routers. Implement static routing for the above network. Select IP addressing of your choice
2. RIP Routing: Select 2 routers and 2 switches, such that there is WAN link between the routers and the switches are connected behind the routers on a LAN. Implement RIP routing on the given network. Select IP addressing of your choice.
3. OSPF Routing: Consider a network with 4 routers. Two routers are in area 0 (core network), and 2 routers are in area 10. There are 4 LANs connected to these 4 routers. Between these 4 routers we have WAN links. Implement OSPF routing between these routers, select IP address of your choice.
4. BGP Routing: Implement the BGP protocol on a network with 3 AS in which one AS is acting as a transit AS. Choose a network of your choice.
5. Design and simulation of Digital functional units using different VHDL modeling styles.
6. Design of Finite State Machine using Algorithmic Modeling style.
7. VHDL design of Data Path using mixed design style.
8. VHDL design of control unit and simple basic microcontrollers.

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1		x			x						x
2	x	x			x						x
3		x	x		x						x
4		x	x		x						x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — B.Tech. Project Part-I

Course Number	**:**	COC4980/CO499A
Credits	**:**	4
Course Category	**:**	DC
Pre-requisite(s)	**:**	COC2090/CO209
Contact Hours (L-T-P)	**:**	0-0-8
Type of Course	**:**	Lab

Course Objectives

1. An opportunity to apply the acquired knowledge into practical.
2. Ability to get hands on experience and exposure.
3. Ability to acquire writing and presentation skills.
4. Ability for project planning and execution.

Course Outcomes

Students who complete this course are able to:

1. Design innovative ideas.
2. Conduct literature survey
3. Prepare good technical project reports..
4. Work and show a good team spirit.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x				x		x	x	
2	x	x							x		
3	x	x		x	x	x		x			x
4			x	x	x				x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — B.Tech. Project Part-II

Course Number	**:**	COC4990/CO499B
Credits	**:**	6
Course Category	**:**	DC
Pre-requisite(s)	**:**	COC2090/CO209
Contact Hours (L-T-P)	**:**	0-0-12
Type of Course	**:**	Lab

Course Objectives

1. An opportunity to apply the acquired knowledge into practical.
2. Ability to get hands on experience and exposure.
3. Ability to acquire writing and presentation skills.
4. Ability for project planning and execution.

Course Outcomes

Students who complete this course are able to:

1. Implement innovative ideas.
2. Prepare and present technical project reports.
3. Write technical papers for journals and conferences.
4. Demonstrate team work and leadership.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x				x		x	x	
2	x	x		x		x		x			x
3					x		x			x	
4			x	x	x				x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — NETWORK SECURITY

Course Number	**:**	COE4450/CO445
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To develop an understanding of information assurance as practiced in computer operating systems, distributed systems, networks and representative applications. Develop a basic understanding of cryptography, how it has evolved, and some key encryption techniques used today. Develop an understanding of security policies (such as authentication, integrity and confidentiality), as well as protocols to implement such policies in the form of message exchanges.

Course Outcome

After completion of this course, the student shall be able to:

1. Describe computer and network security fundamental concepts and principles.
2. Describe the inner-workings of popular encryption algorithms, digital signatures, certificates and anti-cracking techniques.
3. Demonstrate the ability to select among available network security technology and protocols such as IDS, IPS, firewalls, VPNs etc.
4. Describe Operating system security models E.g. UNIX, LINUX WINDOWS security and methods for web application and database security.
5. Identify ethical, professional responsibilities, risks and liabilities in computer and network environment, and best practices to write a security policy.

Syllabus

UNIT I **CONVENTIONAL ENCRYPTION:**
Conventional Encryption: Classical and Modern Techniques, DES, AES, Contemporary Symmetric Ciphers, Public-Key Cryptography RSA, Key Management Deffie-Hellman Key exchange algorithm.

UNIT II **PUBLIC-KEY ENCRYPTION & HASH FUNCTIONS:**
Elliptic Curve Cryptography, Message Authentication and Hash Functions, Security of hash functions, Digital Signatures.

UNIT III **NETWORK SECURITY FOUNDATIONS:**
Network Security overview, Security Methodology, Risk Analysis, Defense models: Lollipop, Onion Model, Sample Security Policy Topics, Security Organization, Security Audit, Physical Security, Authentication and authorization Controls.

UNIT IV **NETWORK ARCHITECTURE AND OPERATING SYSTEM SECURITY:**
Network Design Consideration: Design, performance, Availability, Network Device security, Firewalls, VPN, Wireless network security, IDS, Network role based security: E-mail, DNS etc. Operating system security models, CASE: UNIX, LINUX WINDOWS security. Web Application security, Regular Application security, writing secure software, Database security.

Books

- Stallings, W. Cryptography and Network Security, Principles and Practice, 3rd edition, Prentice Hall, 2002.
- Network Security The Complete Reference, TATA McGraw Hill Publication., 2004
- Tanenbaum, A.S., Computer Networks, 4th edition, prentice Hall
- Stinson, D., Cryptography. Theory and practice, 2nd edition, CRC Press
- Network Security Private Communication in Public World, C. Kaufman, R. Perlman, M. Spicier, 2nd edition PHI.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x										
2		x			x						
3									x		
4			x								x
5						x					

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — DISTRIBUTED AND PARALLEL SYSTEMS

Course Number	**:**	COE4590/CO405
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	COC2080/CO208, COC3100/CO311
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

This course will introduce students to fundamentals parallel and distributed programming. The course will cover the current parallel and distributed architecture, basic issues in parallel/distributed application development, parallel/distributed algorithms, data-structures and programming methodologies, and current technologies. The course will be a "hands-on" course with programming assignments and a final project.

Course Outcome

Upon successful completion of this course, the student will be able to know the following topics:

1. SISD, SIMD, MIMD and networked computers
2. Shared memory and distributed memory computers
3. Static and dynamic interconnections
4. Message passing schemes
5. Different Parallel Algorithms.

Syllabus:

UNIT I INTRODUCTION
Need for high performance computers; Limitations of Von Neumann Architecture, Pipelining, Interleaved memory Organizations, cache, Flynn's and other Classifications; Amdahls Law; Gustufson-Barsis Law; Comparison of two laws, efficiency and speed up Curves; performance measures & metrices.

UNIT IISIMD MODELS
SIMD configurations, Shared memory and mesh- Connected models, Processor organizations - mesh, shuffle, inverse shuffle, butterfly, Hypercube; Inter connection networks & performance parameters, Sorting Algorithm on SIMD models.

UNIT III PIPELINED VECTOR PROCESSOR & MULTIPROCESSORS
Vector Operations; General pipelines, concept of reservation tables; Pipelining of Vector operations; Vectorizing Compilers, case study of Vector Computers- CRAY -1, CRAY –XMP, VP 200, IRAM etc. Tightly coupled and Loosely Coupled Systems; Shared memory and message, Passing models; Properties & Characteristics of Multiprocessors, UMA and NUMA models, Task scheduling, deadlock, cache coherency/ consistency, and their solutions.

UNIT IV PARALLEL ALGORITHMS
Abstract m/c Models; RAM & PRAM Models Analysis of Parallel, Algorithms, Cost Optical algorithms, Concepts of Complexity of algorithms; NP- hard Problem etc, Searching & Sorting algorithms on PRAM Models, Practical models of parallel Computation; BSP, Log GP models, Heterogeneous processing, Parallel programming environments, concept of Grid & Cloud computing.

Books:

*1. Hwang and Briggs, "Computer Architecture and Parallel Processing", McGraw Hill International.
2. Kai Hwang , "Advanced Computer Architecture", McGraw Hill International
3. Quinn M. J., "Designing Efficient Algorithms for Parallel Computers", McGraw-Hill International
4. Saing Soo, YL. Ching, “Parallel processing and parallel Algorithms, springs Publications.

*Text Book

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1			x								x
2				x							
3	x										
4			x								
5	x			x					x	x	

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Soft Computing

Course Number	**:**	COE4440/CO444
Credits	**:**	4
Course Category	**:**	DE
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

To develop an in-depth understanding of soft computing techniques that exploit the tolerance for approximation, uncertainty, imprecision, and partial truth in order to provide intelligent solutions to real world problems.

Course Outcomes

On completion of this course, the student will be able to:

1. Identify and select a suitable Soft Computing methodology to solve the problem
2. Understand & define fuzzy sets and represent these sets by membership functions
3. Describe the relation between real brains and simple artificial neural network models
4. Design genetic algorithms for single and multiple objective optimization problem
5. Analyze and design neuro fuzzy and other hybrid approaches of soft computing techniques for problem solving.

Syllabus

UNIT I **INTRODUCTION**
An overview of Soft Computing, Constituents of Soft Computing and conventional Artificial Intelligence, Introduction to Artificial Neural Networks, Perceptron, Neural Networks Learning Rules, Activation Functions, Derivation of generalized delta learning rule (backpropagation) for Multilayer perceptron.

UNIT II **FUZZY LOGIC**
Fuzzy Sets, Basic Definitions and Terminology, membership function Set-theoretic operation. Fuzzy union, intersection and complement, various T-norm and T-conorm operators, Fuzzy Relations.
Fuzzy Logic, Approximate Reasoning, Compositional Rule of Inference, Mamdani Fuzzy model, Sugeno Fuzzy model, Fuzzy decision making, Fuzzy Control.

UNIT III **GENETIC ALGORITHMS**
Fundamentals of Genetic Algorithm, Encoding, Reproduction, Roulette Wheel, Tournament Selection, Rank Selection etc. Cross over and mutation operators, Introduction to Simulated Annealing. Recent Trends

UNIT IV **HYBRID SYSTEMS**
Hybrid Systems, GA based Fuzzy Systems and Neural Networks Training, Any other applications of soft computing.

Books:

1. *Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications, S. Rajasekaran and G. A. Vijayalakshmi Pai, Prentice Hall India
2. *Fuzzy Logic and its Engineering Applications, T.J. Ross, Mc Graw Hill
3. *Introduction to Artificial Neural Networks, J.M. Zorada, Jaico Publishing House

4. Neuro-Fuzzy and Soft Computing, Jang, Sun and Mizutani, Pearson Education Asia
5. Genetic Algorithms, D. E. Goldberg, Pearson Education Asia

*__Text Book__

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1									x		
2	x		x			x				x	
3							x		x		
4							x			x	
5	x		x						x		

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — EMBEDDED SYSTMS

Course Number	**:**	COE4480/CO448
Credits	**:**	4
Course Category	**:**	DE (Department Elective)
Pre-requisites	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

1. Provide Basic terminology and concepts of Embedded system.
2. Introduce FPGA and CPLD and UML in embedded system design.
3. Introduce Microprocessor/microcontroller architectures and instruction sets.
4. ARM and SHARC processors and Programming concepts.
5. Provide explanation of Design constraints: low power, speed, memory size, real-time behavior.
6. Introduction to real-time operating systems
7. Provide understanding of Basic network and communication protocols.

Course Outcome

Students will be able to:

1. Comprehend the important embedded system terminology and their requirements.
2. Sketch design of embedded system around a microprocessor or microcontroller.
3. Understand different standard processors and their contrast like ARM and SHARC processor.
4. Explain how microprocessor, memory, peripheral components and buses interact in an embedded system.
5. Evaluate how architectural and implementation decisions influence performance and power dissipation in different environments like real-time embedded system.

Syllabus

UNIT I. INTRODUCTION TO EMBEDDED SYSTEMS
Overview of Embedded systems, Design model for Embedded systems, Design description with Unified Modeling Language (UML), Introduction to FPGA and CPLD.

UNIT II. SMALL SCALE EMBEDDED PROCESSOR
Embedded systems Architecture: RISC/CISC Architecture, Von-Neumann and Harvard Architecture, PIC Microcontroller: Architecture, Instruction Set and Interrupt Logic.

UNIT III. ADVANCED EMBEDDED PROCESSORS
Study of ARM and SHARC processors: Architecture, Programming Concepts, CPU performance, CPU power consumption, CPU buses, ARM bus, SHARC bus, Some other Embedded systems processors.

UNIT IV. EMBEDDED SYSTEMS SOFTWARE
Program Design and Analysis, Control and Data Flow graphs, Embedded Operating Systems, Scheduling, Networks for Embedded systems, Various network protocols.

Books:

*1. Computers as Components: principles of embedded computing system design, Wayne Wolf, Morgan Kaufman (Harcourt India).

2. Design with PIC Microcontrollers, John B. Peatman, Pearson Education.
*3. Programming for Embedded Systems by Prasad, Gupta, Das and Sharma, Wiley DreamTech India Pvt Ltd.
4. Embedded System Design: a unified hardware/software by Vahid and Givargis, John Wiley.
5. The design of small-scale embedded systems, Tim Wilmhurst, Palgrave.

*Text books.

Table: Mapping of Program Outcomes and Course Outcomes for CO-448

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1										x	
2	x		x								
3										x	
4		x			x						
5			x		x						

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — COMPUTER NETWORK DESIGN

Course Number	**:**	COE4510/CO451
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

After taking this course the students should be able to:

1. Understand principles of network services and applications.
2. Identify and analyze delays and losses in networks.
3. Design reliable data transfer protocols.
4. Understand how to analyze of congestion control and flow control mechanisms.
5. Design and develop applications that run over networks.

Course Outcome

Students who completed the course have demonstrated the ability to:

1. Understand principles of network services and applications.
2. Compute and analyze delays and losses in networks.
3. Design protocols for reliable data transfer.
4. Analyze mechanisms of congestion control and flow control.
5. Design and develop network applications.

Syllabus

UNIT I INTRODUCTION
Introduction to Computer Networks, Circuit Switching and Packet Switching, Protocol Layers and Their Service Models, Introductory Concepts of Queuing Theory: *Queuing Delays and Losses*, Introduction to Application Layer, Application Layer Services and Protocols: *The Web and HTTP, FTP, Electronic Mail, DNS.*

UNIT II TRANSPORT LAYER
Introduction and Transport Layer Services, Connectionless Transport, Principles of Reliable Data Transfer: *Go-Back-N and Selective Repeat*, Connection-Oriented Transport, Principles of Congestion Control, TCP Congestion Control.

UNIT III NETWORK LAYER
Introduction, Virtual Circuit and Datagram Networks, The Internet Protocol: Forwarding and Addressing, What is Inside a Router?, Routing Algorithms: *Link-State Routing, Distance Vector Routing, Hierarchical Routing*, Routing in the Internet: *RIP, OSPF, BGP*.

UNIT IV LINK LAYER
Introduction to Link Layer Services, Error Detection and Correction Techniques, Multiple Access Protocols, Link-Layer Addressing: *MAC Addresses, ARP, DHCP*, Ethernet, Interconnections: *Hubs and Switches*, The Point-to-Point Protocol.

Books:

*1. J.F. Kurose, K.W. Ross, Computer Networking: A Top-Down Approach Featuring the Internet, 6th Edition, Pearson Education, 2012.

2. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, 5th Edition, Pearson Education, 2012.

3. K.S. Trivedi, Probability and Statistics with Reliability and Queuing and Computer Science Applications, Second Edition, John Wiley, 2002.

*Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x									x	
2		x									x
3			x							x	
4	x		x								x
5		x								x	

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Mobile Computing

Course Number	**:**	COE4520/CO452
Credits	**:**	4
Course Category	**:**	DE
Pre-requisite(s)	**:**	COC3130/CO313
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

Mobile computing and wireless networks is a young and dynamic field. Ubiquitous access to information, anywhere, anyplace, and anytime, will characterize whole new kinds of information systems in the 21st century. These are being enabled by rapidly emerging wireless communications systems such as Cellular transmissions, Personal Communications Systems, Mobile IP, Wireless Local Area networks (LANs). Moreover, the next generation communication systems are expected to provide a range of services to mobile users to support voice, video, multimedia, conventional data, and Internet access in an integrated fashion.

Course Outcomes

At the end of this course students will be able to:

1. Understand the area of wireless networking and mobile computing,
2. Understand the unique network protocol challenges and opportunities presented by wireless communications and host or router mobility.
3. Apply fundamental concepts in mobile wireless systems and mobile computing, mobile IP.
4. Analyze issues associated with small handheld portable devices and design applications that can exploit mobility and location information.

Syllabus

Unit-I INTRODUCTION

Introduction to Wireless Communication Systems: Evolution of mobile radio communications, examples of wireless communication systems, PCS Architecture, Cellular Telephony, 2G, 3G and 4G Networks, AMPS, D-AMPS, GSM, CDMA

Unit-II MOBILITY MANAGEMENT

Handoff, Inter-BS Handoff, Intersystem Handoff, Roaming Management, Handoff Management, MCHO, NCHO, MAHO, Hard Handoff, Soft Handoff

Unit-III 2G/2.5G/2.75G NETWORKS

D-AMPS, Cellular Digital Packet Data, CDPD Architecture, GSM System Overview, GSM Short Message Service (SMS), GPRS Architecture, GGSN, SGSN, EDGE, Mobile Number Portability, VoIP service for Mobile Networks

Unit-IV 3G AND 4G NETWORKS

UMTS, W-CDMA, CDMA2000, Field Trials, IP for 3G, HSDPA, HSUPA, WiMAX, LTE, Advanced LTE, Circuit Switched Fallback (CSFB)

Text Book(s)/ Reference Book(s)References:

1. T.S. Rappaport, "Wireless Communication: Principles and Practice", 2nd ed, Pearson, 2002
2. Yi-Bing Lin, "Wireless & Mobile Network Architectures", John Wiley & Sons, 2001
3. Jochen Schiller, "Mobile Communication", Pearson Education
4. Dave Wisely, Philip Eardley and Louise Burness "IP for 3G - Networking Technologies for Mobile Communications", John Wiley & Sons, 2002
5. Andreas Molisch, "Wireless Communications", 2nd ed, Wiley, 2005

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x					x	x			
2		x	x							x	x
3	x		x				x	x		x	
4	x	x			x				x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title INTERNET TOOLS

Course Number	**:**	COE4310/CO431
Credits	**:**	4
Course Category	**:**	DE
Pre-requisites	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Outcome

1. Explain the fundamental concepts of Cryptography and basic Network protocols.
2. Demonstrate the working of Proxy servers, DNS, torrents, Firewalls and debugging tools.
3. Explain the working of popular encryption algorithms, digital signatures, certificates and key management.
4. Apply encryption methods to ensure message authentication, message integrity and confidentiality.

Syllabus

UNIT I BASIC NETWORK CONCEPTS
Layers of network, Network layer interconnections, Internet Standard, Ports, Proxy servers, Firewall, NAT/PAT, URIs and relative URL, Domain Name System: DNS lookup, records, MX records etc., finger, whois, Content Delivery.

UNIT II PROTOCOLS AND WWW
TCP/IP, UDP, USENET, Internet Control (ICMP), Debugging tools like ping and tracert, IGMP, Internet Management (SNMP). WWW: HTTP, Static and Dynamic web pages with example of JavaScript and php. Markup Languages. Wireshark introduction.

UNIT III ELEMENTARY SOCKET PROGRAMMING
Client/Server model of Interaction, TCP Sockets, TCP Client-Server, UDP Sockets, Socket Options, UNIX System Calls

UNIT IV INERNET SECURITY
Introduction to Cryptography, Encryption Techniques and Authentication Protocols, Digital Signature, Management of Keys. Components of Firewall. Introduction to SSL.

Books and reference material:

1. Forouzan, "TCP/IP Protocol Suit", 4th ed, Tata McGraw Hill, 2010.
2. Douglas E. Comer: "'INTERNETWORKING WITH TCP/IP (Volume 1)", 4th ed, Prentice Hall.
3. A. S. Tanenbaum: "Computer Network", 5th ed, Prentice Hall.
4. W. R. Stevens: "Unix Network Programming", Prentice Hall.
5. Behrouz Forouzan, "Cryptography & Network Security", McGraw Hill, 2008.
6. William Stallings : "Cryptography and Network Security Principles and Practices", 4th Ed, Prentice Hall, 2005.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x				x	x			x		
2	x	x			x						x
3								x			
4						x			x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title VLSI DESIGN TOOLS & TECHNIQUES

Course Number	**:**	COE4420/CO442
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

This course is designed to teach students the theory of VLSI design flow and design automation. It includes the description and the structure of various fundamental components of digital systems like data path and controller.

Course Outcome

Upon successful completion of this course, students should be able to:

1. Understand the design flow of VLSI system, understand the architecture design and design of various blocks and sub-blocks (combinational and sequential block).
2. Explain the structure and fundamental components of VLSI physical design.
3. Understand various algorithms used for design automation.

Syllabus

UNIT I Introduction to VLSI Design; " Design Abstraction, Design Hierarchy, Design Flow, Tools foe VLSI Design, Introduction to VLSI Technology, Layout Design Rules, Stick Diagram, Transistor Parasitic, Wires and Vias".

UNIT II Logic Gate Design, Combinational Logic Network, Layout Design Methods, Standard Cell Layout Design, Combinational network Delays, Power Optimization, Sequential machines, Sequential system and Clocking Disciplines, Sequential System Design, design Validation.

UNIT III Subsystem Design, Adder, Multiplier, Graph Algorithms for Physical Design – Classes of Graph in Physical Design, Graph Problems in Physical Design, Algorithm For interval Graphs.

UNIT IV Physical Design – Partitioning- System level, board level and chip level Partitioning Floorplanning Methods, Block Placement, Routing- Global Routing and Detailed Routing.

BOOKS:

*1. Wayne Wolf, Modern VLSI Design System on Silicon, Second edition "PHI".
2. N Sherwani Algorithm for VLSI Design Automation "Springer".
3. Sabih H Gerez, Algorithm for VLSI Design Automation "J Wiley".

* Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x								
2		x	x			x					x
3		x	x	x		x	x			x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Multimedia Technologies

Course Number	**:**	COE4500/CO450
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

The primary objective of this course is to introduce basic principles of multimedia, image data structures, and multimedia technologies.

Course Outcomes

On completion of this course, the student will be able to:

1. Understand the basic fundamental of multimedia, essential theory and algorithms
2. Apply multimedia techniques
3. Develop hands-on experience in using computers to process images
4. Develop significant thoughts about lacks of the modem in streaming multimedia

Syllabus

UNIT I INTRODUCTORY CONCEPTS
Definition of 'multimedia', Multimedia systems, and components, Multimedia applications, Digitization principles for text, images, audio and video, Basic compression principles, Text compression, Repetition suppression, Statistical Encoding Dictionary Modeling, Image formats & representation schemes, colour schemes, Image compression principal

UNIT III AUDIO COMPRESSION
Audio compression techniques, Sub-band coding, DPCM, Adaptive Sub-Band Coding, Predictive Coding, CELP Coding Perceptual Coding, Voice Recognition , Types of voice recognition systems, Voice recognition performance

UNIT IV VIDEO COMPRESSION
Fundamentals of video compression, temporal redundancy, spatial redundancy, Frame Types, Motion estimation and compensation, Overview of some video coding standards

UNIT V STREAMING MULTIMEDIA
Principles of streaming multimedia, Scalability, Streaming Servers, Streaming Protocols, QoS parameters, Timing relationships in networked multimedia, transcoders

BOOKS

*1. "Multimedia Communications-Applications, Networks, Protocols & Standards", Fred Halsall, Pearson Education.
2. Introduction to Data Compression, Khalid Sayood, Elsevier, 2nd Edition, 2005
3. Information Theory Coding and Cryptography, Ranjan Bose, Tata McGraw-Hill, 3rd Reprint, 2004.

3. Fundamentals of Multimedia, Ze-Nain Li, Mark S. Drew, Pearson Eductation, Indian Reprint, 2005.
4. Principles of Multimedia, Ranjan parekh, Tata McGraw-Hill, 1st Reprint, 2006.

* Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x				x			x	
2					x				x		
3							x		x		
4	x				x				x		

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — ARTIFICIAL INTELLIGENCE & SOFT COMPUTING

Course Number	**:**	COE4530/CO453
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

Introduce students to artificial intelligence and soft computing concepts and techniques and foster their abilities in designing and implementing solutions for real-world problems. Emphasis on intelligent systems - problem solving, reasoning, decision making, learning, logical foundation. The main goal is to achieve a thorough understanding of the foundations of these paradigms, and also to experiment with their use and practical application in a number of problems.

Course Outcome

Students will be able to:

1. Explain underlying AI concepts its achievements and to develop program for AI problems.
2. Understand knowledge representation and applications to learning, natural language processing, agents, expert systems, and planning.
3. Apply soft computing techniques in making intelligent machines and other problems.
4. Define fuzzy logic and reasoning to handle uncertainty. Understand neural networks concepts and its applications.
5. Apply genetic algorithms to combinatorial optimization problems.

Syllabus

UNIT I **INTRODUCTION TO ARTIFICIAL INTELLEGENCE**
Introduction, Problem Solving, Problem definition as a State Space Search, Production Systems, Heuristic Search, Search strategies-BFS, DFS etc., Mini-Max Search, Alpha-Beta Pruning.

UNIT II **KNOWLEDGE PRESENTATION AND EXPERT SYSTEMS**
Preposition and Predicate Logic, Knowledge representation using predicate logic, Interpretation of formula in predicate logic, Resolution Principle, Resolution in predicate logic, Unification, Structured Knowledge representation, Introduction to expert System, Intelligence Agents

UNIT III **SOFT COMPUTING AND ARTIFICIAL NEURAL NETWORK**
Soft Computing Techniques-an Overview, Fuzzy Sets and Fuzzy Logic: Basic Definitions and Terminology, Fuzzy Decision Making, Rough Sets, Applications in Knowledge Discovery. Fundamentals of ANN, Neuron Networks Architecture, Perceptrons, Backpropagation, Self Organizing Maps (SOM), Applications in Image Processing.

UNIT IV **EVOLUTIONARY COMPUTATION**
Introduction, Evolution Techniques-Inherent Features, Genetic Algorithms Evolutionary Programming, Swarm Intelligence, Applications in Optimization Problems.

Books:

1. Elaine Rich and Kevin Knight “Artificial Intelligence”, Tata McGraw Hill
2. Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications, S. Rajasekaran and G.A. Vijayalakshmi Pai, Prentice Hall India
3. D.W. Patterson “Artificial Intelligence and Expert Systems”, PHI Ltd.

* Text Book

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x			x						
2				x					x	x	
3										x	x
4	x	x	x							x	
5										x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Principles of Machine Learning

Course Number	**:**	COE4610/CO461
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

The objective of this course is to introduce concepts, techniques, and algorithms in machine learning. The course will give the student the basic ideas and intuition behind modern machine learning methods as well as formal understanding of how, why, and when they work

Course Outcome At the end of this course, the student can be expected to:

1. Define what is machine learning and its application.
2. Comprehend various supervised learning algorithms and be able to select feature sets and detect anomalies in the algorithm.
3. Explain unsupervised learning algorithms and reinforcement learning techniques.
4. Be familiar with various advance topics of machine learning and their case studies.

Syllabus

UNIT 1: INTRODUCTION

Introduction to Machine Learning, Application, Recent Development, Classification of Machine learning algorithms

UNIT 2: SUPERVISED LEARNING

Supervised learning, Linear and Logistic regression, Neural Network, Back Propagation Algorithm, Generative learning algorithms, Gaussian discriminant analysis, Naive Bayes, Support vector machines, Model selection and feature selection, Ensemble methods: Bagging, boosting, Evaluating and debugging learning algorithms.

UNIT 3: UNSUPERVISED LEARNING & REINFORCEMENT LEARNING

Clustering, K-means, Mixture of Gaussians, Factor analysis, PCA (Principal components analysis), ICA (Independent components analysis), Markov Decision Process, Value iteration and policy iteration, Linear quadratic regulation (LQR), Q-learning, Value function approximation, Policy search.

UNIT 4: ADVANCE TOPICS

Introduction to Deep learning, Graphical Model, Case Studies-Application to Computer Vision, Big Data

BOOKS:

1. Machine Learning: A Probabilistic Perspective. Kevin Murphy. MIT Press, 2012.
2. Pattern Recognition and Machine Learning. Christopher Bishop. First Edition, Springer, 2006.
3. Pattern Classification. Richard Duda, Peter Hart and David Stock. Second Edition, Wiley-Interscience, 2000.
4. Machine Learning. Tom Mitchell. First Edition, McGraw-Hill, 1997

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Design of Programming Languages

Course Number	**:**	COE4030/CO403
Credits	**:**	4
Course Category	**:**	DE
Pre-requisite(s)	**:**	COC2030/CO203, COC2060/CO206
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

The objectives of this course are the following:

1. To increase capacity of expressing ideas
2. To improve background for choosing appropriate languages
3. To increase ability of learning new languages
4. To better understanding of the significance of implementation
5. To overall advancement of computing

Course Outcomes

On completion of this course, the student will be able to:

1. Explain the different categories of programming languages
2. Remember the history of few well known programming lanugages
3. Analyze semantic issues associated with language implementations
4. Determine basic constructs of designing a programming language
5. Differentiate the imperative, functional, and object oriented languages
6. Describe key features of abstract data types and object oriented paradigms
7. Illustrate the essential for scripting and logic programming languages

Syllabus

UNIT I **PROGRAMMING LANGUAGE CONCEPTS**
Programming languages design concepts and trade-offs; Evolution of major PLs: functional, object oriented and scripting languages; Basic building block of PLs: constants, variables, expressions and statements; Names, binding, type checking, scope rules, syntax and semantics of PLs

UNIT II **BASIC CONSTRUCTS OF PROGRAMMING LANGUAGES**
PLs primitive, character, array, record, union, pointer and reference type data types; PLs expressions, Selection and Iterative statements; Unconditional branching; Guarded commands; Subprograms, Parameter passing methods, Subprograms Implementation

UNIT III **OBJECT ORIENTED AND MODERN PROGRMMING LANGUAGES**
Characteristics of object-oriented programming; Abstraction and Encapsulation constructs; Support for object-oriented programming in Smalltalk, C++, JAVA and Ruby; Concurrency levels, Java and C# Threads, Exception & Event handling in C++ and in Java

UNIT IV **FUCNTIONAL AND LOGIC PROGRAMMING LANGUAGES**
Fundamental of Functional programming languages: lambda expression; An introduction to LISP, Scheme, ML and Haskell, Introduction to predicates calculus and logic programming; PROLOG and its elements; Application of logic programming.

Text Book(s)/ Reference Book(s)

1. *Robert W. Sebesta, "Concepts of Programming Languages" Ninth edition, Pearson Education India, Pte. Ltd, 2010.
2. M. Morris Mano and Charles Kime, "Logic and Computer Design Fundamentals", Fourth edition, PHI, Pvt. Ltd. 2007.
3. W William Stalling, "Computer Organization and Architecture Designing for Performance", Seven edition, Pearson education, Ltd, 2006.
4. Mohamed Rafiquzzaman, "Fundamentals of Digital Logic and Microcomputer Design", Fifth edition, Wiley Interscience, A John Wile & Sons Inc. publication, 2005.

***Text Book**

Course Outcomes	Program Outcomes (POs)										
	A	b	c	d	e	f	g	h	i	j	k
1	X		x				x				
2		x	x				x				x
3				x					x		
4	x							x			
5			x				x				
6		x		x							
7		x					x		x		x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — ARTIFICIAL INTELLIGENCE AND NEURAL NETWORKS

Course Number	**:**	COE4300/CO430
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

Introduce students to artificial intelligence and neural network concepts and foster their abilities in designing and implementing solutions for AI problems. Emphasis on intelligent systems - problem solving, logic, reasoning, decision making, learning, and knowledge representation. The main goal is to achieve a thorough understanding of the foundations of these paradigms, and also to experiment with their use and practical application in a number of problems.

Course Outcome

Students will be able to:

1. Explain underlying AI concepts and its achievements.
2. Understand neural networks and its role in computation for target results and apply it for particular problem like pattern classification and recognition problems.
3. Discuss problem solving methods and strategies.
4. Develop an ability to write programs for AI problem solutions.
5. Deal with prepositional and predicate logic.
6. Address the basic issues of knowledge representation. Extending it to the understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and expert systems, and planning.

Incorporation of all these techniques for extending the capabilities of the technologies to more effective and efficient problem solving methodologies.

Syllabus

UNIT I **INTRODUCTION TO ARTIFICIAL INTELLEGENCE**
Introduction, Problem Solving, Problem definition as a State Space Search, Production Systems, Control strategies, Heuristic Search, Basic problem solving method, Search strategies-BFS, DFS, Hill Climbing etc., Problem Reduction, Mini-Max Search, Alpha-Beta Pruning.

UNIT II **PREPOSITIONAL, PREDICATE LOGIC AND KNOWLEDGE REPRESENTATION**
Preposition and Predicate Logic, Knowledge representation using predicate logic, Interpretation of formula in predicate logic, Resolution Principle-conversion to Clausal form, Resolution in predicate logic, Unification, Resolution Refutation System. Structure representation of knowledge techniques, Introduction to expert Systems.

UNIT III **FUNDAMENTALS OF ARTIFICIAL NEURAL NETWORK**
Fundamentals of ANN, Artificial Neuron Vs Biological Neuron, Activation Function, Single layer ANN, Multi-Layer ANN, Perceptrons, XOR problem.

UNIT IV **AI APPLICATIONS AND BACKPROPAGATION**
Study of AI Languages such as PROLOG and LISP to design and implement AI Techniques; Backpropagation, Training Algorithms, Hopfield Networks, Applications-ALVINN, Pattern Recognition, Handwritten Character Recognition, Travelling Salesman Problem.

Books:

*1. Elaine Rich and Kevin Knight “Artificial Intelligence”, Tata McGraw Hill
2. D.W. Patterson “Artificial Intelligence and Expert Systems”, PHI Ltd.
*3. N.K. Bose and P. Liang “Neural Network Fundamentals with Graphs, Algorithms”, Tata McGraw Hill.

***** Text Book

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1	x			x						x	
2		x			x						
3					x						x
4		x								x	x
5		x								x	
6										x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title: Image Processing

Course Number	**:**	COE4320/CO432
Credits	**:**	4
Course Category	**:**	DE
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objectives

The primary objective of this course is to introduce basic principles of digital images, image data structures, and image processing algorithms.

Course Outcomes

On completion of this course, the student will be able to:

1. Understand the basic fundamental of digital image processing
2. Cover the essential theory and algorithms that are usually used in image processing
3. Apply image processing techniques in both the spatial and frequency domains
4. Expose technologies and issues that are specific to image processing systems
5. Develop hands-on experience in using computers to process images
6. Write image processing programs in a high-level language such as C++, Java, Matlab
7. Perform image manipulations and analysis in many different fields
8. Develop significant thoughts about lacks of the modern in image processing

Syllabus

UNIT I **FUNDAMENTAL OF IMAGE PROCESSING**
Basic Image Models and Definitions, image perception, image representation, operations, perspective transformation; Image Transformation: 2-D Fourier Transform and spectral analysis, Discrete Cosine Transform, Discrete Wavelet Transform

UNIT II **IMAGE ENHANCEMENT & COMPRESSION**
Image enhancement: histogram processing, spatial-filtering, frequency-domain filtering; image restoration: linear degradation model, inverse filtering, Wiener filtering; image compression: lossy and lossless compression, entropy coding, transform coding, subband coding, image compression standards

UNIT III **IMAGE SEGMENTATION, REPRESENTATION & ANALYSIS**
Derivative of Gaussian, Line & Canny edge detection, Edge linking & boundary detection, Border Tracing, Region based segmentation, , Image representation Schemes: AR, MA and ARMA representation, Image decomposition, Feature extraction, Image texture analysis

UNIT IV **ADVANCED TOPICS IN IMAGE PROCESSING**
Morphological image processing: binary morphology- erosion, dilation, opening and closing operations, applications; Basic gray-scale morphology operations; color models and color image processing, Case Studies and experimentation

Text Book(s)/ Reference Book(s)

Books:

1. *Rafael C. Gonzalez and Richard E. Woods, “Digital Image Processing”, 3/E, Prentice Hall Publisher, 2008.
2. Maria Petrou and Costas Petrou, “Image Processing The Fundamentals”, Second Edition, John Wiley & Sons Ltd, 2010.
3. Dwayne Philips, “Image Processing in C”, Second Edition, R & D Publication, Electronic Edition, 2000.

***Text Book**

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	I	j	k
1	x						x			x	
2		x	x				x				
3					x				x		
4	x		x							x	
5							x		x		
6		x					x			x	
7	x				x				x		
8			x		x				x		

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — INFORMATION TECHNOLOGY

Course Number	**:**	COE4360/CO436
Credits	**:**	4
Course Category	**:**	DE
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

The objective of this course is to have students learn the importance of information system, information privacy and security and also identify trends in the world of IT; Interacting with computer systems and cultivate skills in development and implementation of information system.

Course Outcome

Students will be able to:

1. Define Information, its need and quality value.
2. Understand the basic concepts and technologies used in the field of information systems.
3. Describe how information technology is used in business and the trends of the technologies.
4. Use various input and output devices of computer systems.
5. Analyze performance of drives and its interface standards.
6. Do system analysis.
7. Explain the need of information security and various techniques for the same.
8. Use current techniques, skills and tools necessary for computing practice.

SYLLABUS:

UNIT I INTRODUCTION

Information concepts and processing, definition of Information, need of Information. Quality, value, concept of Information, Categories and levels of information, Introduction to Information Technology, Technological convergence, developments in Comp. & Communication technology. Practical uses of communication & connectivity.

UNIT II INTERACTING WITH THE COMPUTER

Standard methods of input, Alternative methods of input, Optical I/p devices, Output devices-Monitors, Sound systems, Printers & Plotters, CPU's used in personal computers, Types of storage devices – Magnetic & Optical storage devices. Measuring drive performance, drive interface standards.

UNIT III DESIGN AND ANALYSIS OF INFORMATION SYSTEM

System planning, Approaches to system planning. Role of a system Analyst, File design – Type of file, file structure & organization, structure method, design considerations, Management Information Systems.

UNIT IV INTERNET

Principles of data security, Maintenance and troubleshooting, security concepts, security software, Failure & recovery, security on the Internet or Network, Viruses & the Internet, Firewall, Web Techniques: WAP, ASP, Sampling.

UNIT V APPLICATIONS OF INFORMATION TECHNOLOGY

Introduction to information representation in digital Media, Graphics, Animation, audio, video, Introduction to JPEG, MPEG, Office Automation, CAD, Business data processing, Scientific computations, Application packages, Use of computers in Entertainment and Education, Multimedia, Multilingual Applications, Railway Reservation Systems, E-Commerce, Major IT Projects in India, E-governance, Nicnet, Ernet.

REFERENCE:

*1. Peter Norton's "Introduction to Computers.", Tata Mc Graw Hill 2001.
2. J. Kanter, "Management Information Systems", Prentice Hall, 1995.
3. Curtin, Foley, Sen, Morin "Information Technology – The breaking wave".

Tata Mc Graw Hill 1999.

* Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	I	j	k
1	x						x				
2	x				x						
3							x				
4											x
5		x			x						
6		x									
7										x	
8											x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — ADVANCED MICROPROCESSOR SYSTEMS AND DESIGN

Course Number	**:**	COE4390/CO439
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

This course aims to develop background knowledge as well as core expertise in microprocessor and microcontroller. Topics covered are 16 bit microprocessors, 32 bit microprocessors and some modern microprocessors.

Course Outcome

Students will be able to

1. Explain the concepts and basic architecture of 8086 family.
2. Understand the differences between 16 bit and 32 bit microprocessors.
3. Write assembly language program in 8086 for various application.
4. Explain the use of a memory hierarchy to reduce effective memory access times.
5. Discuss the problems of using caches in a multiprocessor environment, know the operation of several cache coherency protocols and be able to predict the bus traffic given a sequence of CPU "memory" accesses.
6. Know about the different computer architectures SISD, MISD, SIMD, MIMD.
7. Design new microprocessor based systems.

Syllabus:

UNIT I **16 BIT MICROPROCESSOR**
Introduction to 16 bit microprocessors-the 8086 family; 8086 architecture, instruction set, simple programming examples, interrupts and interfacing; 8086 architecture, addressing modes, memory management, protection mechanisms and interrupt handling.

UNIT II **32 BIT MICROPROCESSOR SYSTEMS**
Introduction to 32 bit microprocessors, architectures-conventional, unconventional, memory and storage devices; communications interfaces; software options and development systems, applications.

UNIT III **INTRODUCTION TO ARCHITECTURES**
Conventional computer architectures-the Von Neumann architectures, alternative computer architectures-Dataflow architectures, SIMD, MIMD, Classifications-control flow program organization, Dataflow program organization, reduction program organization, Machine organization-centralized, packet communication, expression manipulation, Multiprocessing systems; systolic and wavefront arrays, pipelines, cache memory inter leaving, virtual memory management, extended processing sets, reduced instruction set computers, High integrity processors, digital signal processing chips-Harvard architecture, Hypercube architectures for supercomputers.

UNIT IV **SURVEY OF ADVANCED MICROPROCESSORS**
Survey of advanced microprocessors like MC68020, Z80000, Intel 80386, Inmos T414, Am29300, NCR/32-000, Fairchild clipper and TMS 34010 Graphics system processor.

BOOKS:

*1. Yu-Ching liu and G.A. Gikson "Microcomputer Systems; The 8086/8088 Family", Prentice Hall,1994.

2. Ed. Titus and Laurson, "16-bit Microprocessors', Howard & Co., 1995.

*3. A.P. Mathur, "Introduction to Microprocessors" , McGraw Hill, 1996.

*Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x						x		
2					x						
3				x							
4	x										
5			x								
6		x	x		x						
7	x								x	x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — PATTERN RECOGNITION AND COMPUTER VISION

Course Number	**:**	COE4410/CO441
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Prerequisites	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

1. introduce basic concepts of Pattern recognition
2. present various feature extraction and classification methods
3. Illustrate the fundamental concepts, theories, and algorithms for pattern recognition and machine learning, which are used in computer vision.
4. Illustrate different pre and post image processing Techniques and Transformations.
5. Introduction to Bayesian decision theory, Feed Forward Neural Nets and Recurrent Networks, nearest neighbor indexing and hashing and some transformation techniques.

Course Outcome

Students will be able to:

1. Explain basic theories and techniques in computer vision and pattern recognition
2. Identify various approaches of computer vision and pattern recognition and design the components of the systems for computer vision and pattern recognition
3. Describe and discuss the basic functions and methods for image processing.
4. Design simple systems for computer vision and pattern recognition which can handle certain problem.
5. Gain skills in evaluating, experimenting with, and optimizing the performance of the systems for computer vision and pattern recognition

Syllabus

UNIT I. Introduction to Pattern Recognition and concept learning from examples, Features, Feature spaces, Pattern classes / concepts, Decision Regions and decision boundaries, Bay's decision theory.

UNIT II. Maximum likelihood estimation, Supervised learning in Feed Forward Neural Nets and Recurrent Networks Unsupervised learning – Self – organizing feature extraction. Adaptation rename theory, Theory of feature mapping, Application to character and image pattern recognition.

UNIT III. Image formation and image acquisition, Edge deletion, Image segmentation and feature extraction, Hough transform, Reflections map, Photometric stereo, shape for shading, Extended Gaussian image, Shape from contours, Stereo vision – Imaging geometries, Camera calibration, matching and interpolation Range image analysis techniques.

UNIT IV. Motion field and Optical Flow, Structure from motion, Image sequence analysis, Shape descriptions and recognition of 2 – D and 3 – D objects, Applications – Inspection systems, Document image understanding.

BOOKS:

*1 T. Mitchall, “Machine Learning”, Tata McGraw Hill,

2 Fukunaga K. “Introduction to Statistical Pattern Recognition”, (2nd edition), Academic Press, Boston.

* Text Book

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1										x	
2					x					x	
3					x					x	
4		x	x		x						x
5	x									x	x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — INTERNET PROTOCOLS

Course Number	**:**	COE4490/CO449
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisites	**:**	COC3130/CO313
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

1. To learn how the Internet is structured into layers and the various protocols at each layer with emphasis on the transport and application layers.
2. To explores the development of TCP/IP applications and their associated protocols. It utilizes hands-on programming and makes use of network monitoring tools. It includes detailed coverage of TCP, UDP, HTTP, FTP, and SMTP protocols.
3. To master the development of client-server Internet applications using the sockets and other higher-level APIs.

Course Outcome

Students will be able to:

1. Demonstrate understanding of the TCP/IP model and relevant protocols in each layer.
2. Describe the IP addressing, Internet domain names and recognize the role of the DNS servers.
3. Explain the operation and related issues of various common Internet applications and protocols including: HTTP, SMTP, POP, FTP, Telnet, IGMP, etc.
4. Identify and apply various socket programming concepts and mechanisms.
5. Use effectively the socket interface to develop Client-Server Internet applications.
6. Practice software engineering principles and methods in building network-aware applications.

Syllabus

UNIT-I — **INTRODUCTION**
Internet Architecture, Addresses IPV4 & IPV6,IP, BOOTP,ICMP

UNIT-II — **TCP & UDP**
TCP: Connections, Flow Control, Segment Format, Retransmission, TCP state machine, Other features
UDP: Format checksum computation, UDP multiplexing and other features

UNIT-III — **MAJOR PROTOCOLS**
GGP EGP, RIP, OSPF, HELLO, IGMP , SNMP, Telnet DHCP, WAP

UNIT-IV — **SOCKET PROGRAMMING AND APPLICATIONS.**
Internet programming.
Unix System Calls.
Socket Programming, Introduction to Web server & search engine. Introduction to Internet Languages like Java & HTML.

Books:

1. Internetworking with TCP/IP (Vol. I) by Douglas E. Comer PHI
2. Computer Networks by A. S. Tannenbaum PHI
3. Unix Network Programming by W.R. Stevens. PHI
4. TCP/IP Protocol Suite by Forouzan B. A TMH

Table: Mapping of Program Outcomes and Course Outcomes for CO-449

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1	x									x	
2										x	
3										x	
4			x		x						x
5			x		x						x
6	x										

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Big Data Analytics

Course Number	**:**	COE4560/CO456
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective The purpose of this course is to introduce the students with big data storage systems and important algorithms that form the basis of big data processing. The course also introduces the students with major application areas of big data analytics.

Course Outcome On successful completion of this course, students will be able to:

1. Understand the concept and challenges of big data
2. Explain the basics of big data storage systems
3. Explain the various algorithms used for big data processing.
4. Describe and use the large-scale analytics tools available to solve big data problems
5. Describe the major application areas of big data.

Syllabus

UNIT I: Introduction to big data: introduction to big data, the four dimensions of big data,:- volume, velocity, variety, veracity, drivers for big data, introducing the storage, query stack, revisit useful technologies and concepts, real-time big data analytics

UNIT II: Distributed file systems: Hadoop Distributed File System, Google File System, Data Consistency, Distributed Hash-table, Key-value Storage Model, Document Storage Model, Graph Storage Models

UNIT III: Scalable algorithms:- Mining large graphs, with focus on social networks and web graphs, Centrality, similarity, all-distance sketches, community detection, link analysis, spectral techniques. Map-reduce, Pig Latin, and NoSQL, Algorithms for detecting similar items. Recommendation systems, Data stream analysis algorithms, Clustering algorithms, Detecting frequent items.

UNIT IV: Big Data Applications/ Issues: Advertising on the web, web page quality ranking, mining social-networking group, human interaction with big-data. Privacy, Visualization, Compliance and Security, Structured vs Unstructured Data.

Books:

1. Mining of massive datasets, Anand Rajaraman, Jure Leskovec and Jeffrey Ullman, 2014.
2 An introduction to information retrieval, Christopher D. Manning, Prabhakar Rahgavan, Hinrich Schutze, 2009
3 Data-Intensive Text Processing with MapReduce, Jimmy Lin and Chris Dyer, 2010.

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Cloud Computing

Course Number	**:**	COE4580/CO458
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective This course will cover the study of various algorithms involved in better implementing the cloud-based systems starting through fundamentals of deployment.

Course Outcome On successful completion of this course, students will be able to:

1. Learn algorithms involved in better implementing the cloud-based systems.
2. Learn the fundamentals of deployment in cloud computing.
3. Be able to develop application in cloud environment.

Syllabus

UNIT I: Introduction: Distributed Computing and Enabling Technologies, Cloud Fundamentals: Cloud Definition, Evolution, Architecture, Applications, deployment models, and service models.

UNIT II: Virtualization: Issues with virtualization, virtualization technologies and architectures, Internals of virtual machine monitors/hypervisors, virtualization of data centers, and Issues with Multi-tenancy.

UNIT III: Interoperability and Service Monitoring: Issues with interoperability, Vendor lock-in, Interoperability approaches. SLA Management, Metering Issues, and Report generation. Resource Management and Load Balancing: Distributed Management of Virtual Infrastructures, Server consolidation, Dynamic provisioning and resource management, Resource Optimization, Resource dynamic reconfiguration, Scheduling Techniques for Advance Reservation, Capacity Management to meet

UNIT IV: Migration and Fault Tolerance: Broad Aspects of Migration into Cloud, Migration of virtual Machines and techniques. Fault Tolerance Mechanisms, Security, Advances: Grid of Clouds, Green Cloud, Mobile Cloud Computing.

Books:

1. Cloud Computing Principles and Paradigms, Rajkumar Buyya, James Broberg, Andrzej Goscinski, Wiley Publishers, 2011.
2. Cloud Computing Bible, Barrie Sosinsky, Wiley Publishers, 2010.
3. Mastering Cloud computing, Rajkumar Buyya, Christian Vacchiola, S Thamarai Selvi, McGraw Hill, 2013.

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — Principles of Cyber Physical Systems

Course Number	**:**	COE4600/CO460
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective This course aims to develop an understanding of the underlying principles behind integration of computation (software), networking, and physical processes to design useful engineering systems. It primarily deals with effective orchestration of software and physical processes.

Course Outcome On successful completion of this course, students will be able to:

1. Understand the significance of the integration of computation with the physical process.
2. Build high confidence systems with real-time and concurrent behaviors.
3. Analyze computational systems that interact with physical processes.
4. Design useful applications of cyber physical systems.

Syllabus

UNIT I: Foundations of Cyber Physical Systems, Modeling Dynamic Behaviours, Continuous Dynamics: Properties of Systems, Feedback Control, Discrete Dynamics: The notion of state, Finite-State Machines, Extended State Machines, Hybrid Systems, Composition of State Machine, Concurrent Models of Computation.

UNIT II: Models of sensors and actuators, Common Sensors, actuators, Embedded Processors, Types of Processor, Parallelism, Memory Technologies,

UNIT III: Input and Output, Real-Time Task Scheduling. Types of Real-Time tasks and their characteristics. Multitasking, Different Task Scheduling algorithms

UNIT IV: Analysis and verification: Invariants and temporal logic, Equivalence and refinement, reachability analysis and model checking, Quantitative analysis, Security and Privacy, applications of Cyber Physical Systems

Books:

1. E. A. Lee and S. A. Seshia, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, Second Edition, MIT Press, 2017.
2. R.Mall, Real Time Systems: Theory and Practice, Pearson Education,2007.
3. Hermann Kopetz and Gunther Bauer, The Time-Triggered Architecture, Proceedings of the IEEE 91(1): 112 - 126, 2013.
4. Edward A. Lee. Cyber-Physical Systems: Design Challenges. IRORC 2008.

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x	x	x		x						
2		x	x					x			
3			x		x					x	
4	x		x								x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech.

Course Title — REAL TIME SYSTEMS

Course Number	**:**	COE4550/CO455
Credits	**:**	4
Course Category	**:**	DE (Departmental Elective)
Pre-requisite(s)	**:**	COC309/CO309
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Outcome On successful completion of this course, students will be able to:

1. Characterise real-time systems and describe their functions
2. Understand Real Time System requirement, design, and performance analysis
3. Apply formal methods to the analysis and design of real-time systems

Syllabus

UNIT I: Introduction to real time system, Issues in real-time computing. A basic model of a Real-Time system. Types of Real Time Systems. Timing Constraints. Classification of Timing Constraints. Modeling Timing Constraints. Hard and soft real time systems.

UNIT II: Real-Time Task Scheduling. Types of Real-Time tasks and their characteristics. Different Task Scheduling algorithms

UNIT III: Handling Resource sharing and dependencies among real-time tasks. Features of Real-Time Operating Systems. Real-Time Communications. Real-Time Databases..

UNIT IV: Computer Controlled Systems. Real-Time Control. Discrete PID Controller. Fuzzy Controller. Real-Time modeling and case studies.

BOOKS:

1. R.Mall, Real Time Systems: Theory and Practice, Pearson Education,2007.
2. C.M.Krishna and K.G.Shin, Real Time Systems, Tata McGraw Hill,2010.
3. Jane Liu, Real Time Systems, Pearson Education, 2000

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. (for all branches)

Course Title — SELECTED TOPICS IN COMPUTER ENGINEERING-I

Course Number	**:**	COO4460/CO446N
Credits	**:**	4
Course Category	**:**	OE (Open Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	4 (3-1-0)
Type of Course	**:**	Theory

Course Objective

The course discusses few important topics from Data Structures and Operating System Concepts. The objective of this course is to familiarize students, of different branches of engineering, with the basics of Data structures and their use in fundamental algorithms for common problems like sorting, searching etc. The another objective of the course is to familiarize the students with the functions of operating systems and the basic operating system concepts like process, resource scheduling and memory management techniques.

Course Outcome

Students will be able to:

1. Learn how the choice of data structures and algorithm design methods impact the performance of programs.
2. Analyze the importance and use of Abstract Data Types (ADTs)
3. Design and implement elementary Data Structures such as arrays, trees, Stacks, Queues, and Hash Tables.
4. Explain the process concept and various process scheduling algorithms in operating system.
5. Explain the concept of memory management like swapping, paging, segmentation, page replacement algorithms etc in a given constraints.

SYLLABUS:

UNIT I **INTRODUCTION**

Concept of Data Structures, Basic Terminologies related to data structures, linear and non-linear data structure. Concept and properties of algorithms, How to develop an algorithm, Complexity, Time-Space Tradeoff, Algorithm analysis, Rate of growth: Big Oh notation, other asymptotic notations for complexity of algorithms.

UNIT II **ARRAYS**

Arrays, one-dimensional arrays: traversal, selection, searching, insertion and deletion. Sorting: Bubble sort, selection sort, insertion sort, merge sort, quicksort, other sorting methods and their analysis. Multi-dimensional arrays, Representation of arrays in physical memory, Application of arrays.

UNIT III **ABSTRACT DATA TYPES (ADTs)**

Abstract Data Types, Stacks, Applications of Stacks - prefix and postfix notations, Queue, Circular Queue, Priority Queue, Dequeue, Linked Lists, Operations on Linked Lists, Binary Tree, representation and traversal of binary tree.

UNIT IV **OPERATING SYSTEM & MEMORY MANAGEMENT**

Introduction, Process Concept, Process States, Process Control Block, Process Scheduling, Scheduling Algorithms, Memory management: swapping, paging, segmentation, virtual memory, page replacements algorithms.

Books:

*1. Aho, Hopcroft, Ullman, "Data Structures and Algorithms", Pearson Education

*2. Lipschutz, "Data structures" Tata McGraw Hill.

*3. Silberschatz, Galvin "Operating System Concepts", 7th ed, Addisson Wesley, 2006

4. A.K.Sharma, "Data Structure using C", Pearson, 2011

5. Goodrich M. Tamassia R., "Data Structures and Algorithms in Java", 3rd ed. Wiley

* Text Books

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x						x				
2	x				x		x				
3	x				x		x				
4							x	x			
5							x		x		

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. (for all branches)

Course Title — SELECTED TOPICS IN COMPUTER ENGINEERING-II

Course Number	**:**	COO4470/CO447N
Credits	**:**	4
Course Category	**:**	OE (Open Elective)
Pre-requisite(s)	**:**	
Contact Hours (L-T-P)	**:**	4 (3-1-0)
Type of Course	**:**	Theory

Course Objective

The objective of this course is to study the fundamental concepts of Computer networking and Database Management Systems. in this course, students will learn about protocols governing various layers of OSI and TCP/IP models like flow control, techniques for routing information in packets and data organization in relational databases and maintaining its a*tomicity, consistency, isolation, durability*.

Course Outcome

Students will be able to:

1. Define computer network concepts, OSI and TCP/IP Model.
2. Explain data link layer emphasizing on multiple access of media and collision detection.
3. Explain network layer emphasizing on routing algorithm and transport layer.
4. Describe architecture and fundamental elements of Relational Database management system.
5. Learn Normalization, Recovery and Concurrency techniques

SYLLABUS:

UNIT I **COMPUTER NETWORKS**

Protocol and Protocol Structure, Layered Architecture, ISO-OSI Reference Model, TCP/IP Reference Model, Physical and Data Layer Concepts: Transmission Media, MAC Layer, Flow Control , CSMA/CD

UNIT II **TCP/IP PROTOCOLS & ROUTING ALGORITHMS**

Networks Layer : IP, Family & Protocol, Routing Algorithms Transport layer : QOS, Connection Management; Flow Control, Buffering and Crash Recovery Introduction to Telnet, FTP, SMTP, SNMP, PPP, SLIP

UNIT III **DATA BASE MANAGEMENT SYSTEM**

Data Base Systems and their needs, Components of DBS, DBS architecture, Data Base Administrator and his role, DBMS and its Components, Relational Model: Domains, Attributes & keys, Extension, Intension, Integrity Rules and Constraints, SQL

UNIT IV **NORMALIZATION, RECOVERY AND CONCURRENCY**

Normalization & Functional Dependencies, Introduction to Recovery, Transaction, Commit, Rollback, Sync, Concurrency control problems

Books:

*1. A.S. Tanenbaum , "Computer Networks" 4th edition,
*2. Abraham Silberchatz, " Data Base System Concepts", McGraw Hill

3. Forouzan, “Data Communication and Networking”, Tata McGraw Hill
4. Willian Stalbings , “Data Communication and Networks”,
5. Date, C.J.: “Introduction to Data base System”, Addison Wesley
6. Elmasri & Navathe “Fundamentals of Database Systems” 5th Edition

*Text Books

Course Outcomes	Program Outcomes (POs)										
	a	b	c	d	e	f	g	h	i	j	k
1	x						x	x		x	
2					x		x				
3					x		x				
4							x	x			x
5	x				x			x			x

2017-2018
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. (for Electronics Engineering)

Course Title — COMPUTER ARCHITECTURE

Course Number	**:**	CO460
Credits	**:**	4
Course Category	**:**	ESA (Engineering Science in Arts)
Pre-requisite(s)	**:**	Digital Logic
Contact Hours (L-T-P)	**:**	3-1-0
Type of Course	**:**	Theory

Course Objective

To provide the basic knowledge to understand the hardware operation of digital computers, theory of components of a computer and its operation, and to learn applying the knowledge for designing a minimal computer system. The course concludes with a look at the recent switch from sequential processing to parallel processing.

Course Outcome

Students will be able to:

1. Explain the techniques that computers use to communicate with the input and output devices
2. Explain the importance of memory hierarchy in computer designs, and explains how memory design impacts overall hardware performance
3. Explain and design arithmetic and logic unit
4. Interpret the micro-operations and design of accumulator logic
5. Design the control and execution unit to execute a binary instruction
6. Acquire the knowledge of advanced concepts of performance measure and speeding-up of computing, and characteristics and interconnection of multiprocessors.

Syllabus

UNIT I I/O AND MEMORY ORGANIZATION
Introduction to Digital computers, Peripheral Devices, Input-Output Interface, Modes of Transfer, Direct Memory Access (DMA), Input-Output Processor (IOP): CPU-IOP Communication, Memory hierarchy: Main, Auxiliary and Associative memory, Cache memory and cache mapping, Virtual memory and address mapping, Memory management hardware.

UNIT II MICRO-OPERATION AND COMPUTER DESIGN
Method of Register, Bus and Memory transfer, Micro-operations: Arithmetic, Logic and Shift, Arithmetic logic shift unit, Instruction code, Instruction and interrupt cycle, Timing and control, memory-reference, register-reference and input-output instructions, complete computer description, Design of Accumulator Logic.

UNIT III CONTROL LOGIC AND PROCESSOR ORGANIZATION
Control memory, Address sequencing, Micro-programmed and Hard-wired control, Microinstruction format, Design of control unit, Micro-program sequencer, Central Processing Unit (CPU): Single accumulator, General register and Stack organizations, RISC and CISC characteristics.

UNIT IV PIPELINE AND PARALLEL PROCESSORS
Parallel processing, Pipelining: general consideration and speedup, Arithmetic and instruction pipeline, Characteristics of multiprocessors, Interconnection structures, Interprocessor arbitration, Interprocessor communication and synchronization, Cache coherence and MESI protocol.

BOOKS:

1. M. Morris Mano, “Computer System Architecture” Third edition, Pearson Education India, Pte. Ltd, reprint 2007.
2. M. Morris Mano and Charles Kime, “Logic and Computer Design Fundamentals”, Fourth edition, PHI, Pvt. Ltd. 2007.
3. William Stalling, “Computer Organization and Architecture Designing for Performance”, Seven edition, Pearson education, Ltd, 2006.
4. Mohamed Rafiquzzaman, “Fundamentals of Digital Logic and Microcomputer Design”, Fifth edition, Wiley Interscience, A John Wile & Sons Inc. publication, 2005

Course Outcomes	**Program Outcomes (POs)**										
	a	b	c	d	e	f	g	h	i	j	k
1		x	x				x				
2	x	x			x						x
3		x			x		x				
4	x	x					x				
5	x				x		x				x
6	x				x		x		x		x

w.e.f. 2019 – 2020
DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. (for Electronics Engineering)

Course Title : DATA STRUCTURES AND PROGRAMMING

Course Number	**:**	COA3600
Credits	**:**	3
Contact Hours (L-T-P)	**:**	2-1-0
Type of Course	**:**	Theory

Course Outcome

1. The student will be familiar with commonly used Data Structures and Algorithms.
2. The student will be able to design and develop Data Structures for a given problem.
3. The student will be able to write programs in C language to solve common problems using concept of Data Structures such as arrays, linked list, tree etc. and Algorithms.

UNIT I **Introduction to Data Structures and Algorithms**
Review of structures and Pointers in C; Data Structures: Concept, Organization, Classification, Operation; Algorithms: Introduction, Definition, Time and Space Complexity, Notations (Big-O, omega, Theta)

UNIT II **Arrays, Search and Sorting Algorithms**
Arrays: Definition, Accessing Elements, Search, Deletion, Merge; representation of Arrays, and pointers in physical Memory; Binary Search Algorithms, Sorting Algorithms: bubble Sort, Insertion Sort, Selection Sort, Merge Sort, Quick Sort.

UNIT III **Linked Lists, Stacks and Queues**
Link Lists: Singly Linked List, Circularly Linked, Doubly Linked; Traversal, insertion and Deletion; Stacks: Array Representation, Linked Representation, Operations on Stacks, Multiple Stacks, Applications;
Queues: Array Representation, Linked Representation, Operations on Queues, Types, Applications.

UNIT IV **Trees and Graphs**
Trees: Types (General, Forests, Binary, Binary Search), Operations on Binary Search Trees; Heaps: Binary, Binomial, Fibonacci; Graphs: Terminology, Representations, Traversal Algorithms, Shortest Path Algorithms, Searching; Hashing: Hash Tables, Hash Functions

Books:

*1. Lipschutz, "Data Structures with C" Tata McGraw Hill.
2. Tanenbaum, "Data Structures using C", Pearson.
3. Goodrich M. Tamassia R., "Data Structures and Algorithms in Java", 3rd ed. Wiley

* Text Book