COMPUTER ENGINEERING COURSE

SCHEME OF INSTRUCTION AND EXAMINATION REVISED COURSE 2019-2020

SEMESTER – III

Course Code	Nomenclature of the Course	Scheme of Instruction Hrs/Week			Scheme of Examination						
		L	T	P	Duratiion (Hrs)	Marks Th	IA	TW**	P	Total	Credits
CE310	Mathematics III	3	1	0	**3**	100	25	25	0	**150**	**4**
CE320	Logic Design	3	0	0	**3**	100	25	0	0	**125**	**3**
CE330	Data Structures	3	0	0	**3**	100	25	0	0	**125**	**3**
CE340	Object Oriented Programming System	3	0	0	**3**	100	25	0	0	**125**	**3**
CE350	Computer Organization	3	1	0	**3**	100	25	25	0	**150**	**4**
CE360	Data Structures Programming Lab	0	0	4	**0**	0	0	25	50	**75**	**2**
CE370	Object Oriented Programming System Lab	0	0	4	**0**	0	0	25	50	**75**	**2**
HM001	Technical Communication	2	0	0	**0**	0	0	75	0	**75**	**2**
AC390	Mathematics I & II (Bridge Course)*	2	0	0	**0**	0	0	0	0	**0**	**0**
	TOTAL	**19**	**2**	**8**		**500**	**125**	**175**	**100**	**900**	**23**

*Applicable to direct second year /lateral entry students
**Term Work marks are to be awarded through continuous evaluation

LEGEND

Abbreviation	Description
L	Lecture
T	Tutorial
P	Practical
O	Oral
Th	Theory
TW	Term Work
IA	Internal Assessment

COMPUTER ENGINEERING COURSE

SCHEME OF INSTRUCTION AND EXAMINATION REVISED COURSE 2019-2020

SEMESTER – IV

Course Code	Nomenclature of the Course	Scheme of Instruction Hrs/Week			Scheme of Examination						
		L	T	P	Duration (Hrs)	Marks					Credits
						Th	IA	TW*	P	Total	
CE410	Discrete Mathematics	3	1	0	**3**	100	25	25	0	**150**	**4**
CE420	Microprocessors & Microcontrollers	3	0	0	**3**	100	25	0	0	**125**	**3**
CE430	Formal Languages & Automata Theory	3	0	0	**3**	100	25	0	0	**125**	**3**
CE440	Modern Algorithm Design Foundation	3	0	0	**3**	100	25	0	0	**125**	**3**
CE450	Object Oriented Software Engineering	3	1	0	**3**	100	25	25	0	**150**	**4**
CE460	Modern Algorithm Design Foundation Lab	0	0	4	**0**	0	0	25	50	**75**	**2**
CE470	Microprocessors & Microcontrollers Lab	0	0	4	**0**	0	0	25	50	**75**	**2**
HM100	Economics for Engineers	3	0	0	**3**	100	25	0	0	**125**	**3**
	TOTAL	18	**2**	8	--	**600**	**150**	**100**	**100**	**950**	**24**

*Term Work marks are to be awarded through continuous evaluation

LEGEND

Abbreviation	Description
L	Lecture
T	Tutorial
P	Practical
O	Oral
Th	Theory
TW	Term Work
IA	Internal Assessment

MATHEMATHICS III

Course Code	CE310		Credits	4	
Scheme of Instruction	L	T	P	TOTAL	
Hours/ Week	3	1	0	40 hrs/sem	
Scheme of Examination	IA	TW	TM	P	O
TOTAL = 150 marks	25	25	100	0	0

Course Outcomes:

The student will be able to:

CO1	Compute the rank, Eigen values and Eigen vectors of a given matrix, which will enable students to handle linear systems.
CO2	Compute Laplace transforms of real valued functions and apply it to solve integral and differential equations.
CO3	Compute Fourier transforms and Z-transforms and be able to apply it in their engineering studies.
CO4	Understand the basic concepts of probability, random variables, mean, variance, standard deviation and probability distributions.

UNIT -1	
MATRICES : Types of matrices, Determinant, inverse of matrix, Elementary transformations, Elementary matrices, Rank of matrix, Reduction to normal form, Canonical form, Rank using elementary transformation, Linear independence and dependence of vectors, System of the form AX = 0, and AX = B, and their solutions, Eigen values, Eigen vectors with properties, Cayley-Hamilton theorem with its applications, minimal polynomial, Diagonalization.	10 hrs
UNIT -2	
LAPLACE TRANSFORMS: Definition. Existence conditions, properties, inverse Laplace transforms. Laplace transform of periodic functions, Convolution theorem, Laplace transform of Dirac-Delta function, Application of Laplace transforms in solving linear differential equations with initial conditions and system of linear simultaneous differential equations.	10 hrs
UNIT -3	
FOURIER TRANSFORM : Fourier Transform, Inverse Fourier transform, Fourier Sine and Cosine transform Convolution and application. Z-TRANSFORM: Definition, region of convergence, properties, Z-transform on impulse function, Convolution theorem, application to difference equations.	10 hrs
UNIT -4	
PROBABILITY: Definition, properties, Axioms of probability, conditional probability, theorem on total probability, Bayes theorem; Random variables-discrete & continuous; Expectation and Variance, Standard deviation, Moment Generating Function & properties, Standard distributions: discrete-Binomial, Geometric & Poisson; continuous-Uniform, Normal, exponential.	10 hrs

TEXTBOOKS

1	B. S. Grewal;Higher Engineering Mathematics; Khanna Publications, New Delhi.
2	Erwin Kreyzing; Advanced Engineering Mathematic; New International Limited.
REFERENCES	
1	P. Kandasamy; Engineering Mathematics; Chand & Co., New Delhi.
2	Srimanta Pal, Subodh C. Bhunia; Engineering Mathematics; Oxford University Press
3	D. S. Chandrasekhraiah; Engineering Mathematics- Part III ; Prism Books Pvt. Ltd.
4	Montgomery, D. C., Probability and Statistics for Engineers; Prentice Hall of India.

LOGIC DESIGN					
Course Code	CE320		**Credits**	**3**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**3**	**0**	**0**	**40 Hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 125 marks	**25**	**0**	**100**	**0**	**0**

Course Outcomes:

The student will be able to:

CO1	Convert the numbers from one radix to another and perform arithmetic operations using Complement Arithmetic.
CO2	Solve Boolean Expressions using Boolean algebra, K-maps and VEM and implement them using logic gates. Design any given combinational circuits and explain their applications.
CO3	Explain different flip flops, registers and their applications.
CO4	Design sequential circuits,state machines ,synchronous and asynchronous counter circuits

UNIT -1	
Introduction: Digital and analog systems, Logic levels and Pulse Waveforms. Number systems – Decimal, Binary, Representation of Signed numbers and binary arithmetic, Octal number system, Hexadecimal number system. Binary codes – Classification, 8421 BCD code, XS-3 code, Gray code, Error correction and detection codes. Logic gates-AND, OR, NOT, Universal , X-OR,X-NOR gates. Boolean algebra: Logic operations, Laws of Boolean Algebra, Duality, Reducing Boolean expressions, Boolean functions and their representations, Boolean expressions in SOP and POS forms, Computation of total gate inputs, Boolean expressions and logic diagrams, Conversion of AOI to NAND / NOR logic.	10 hrs
UNIT -2	
Minimization of Switching Functions: Two, Three, Four variable K-Map, Don't Care Combinations, Quine- McCluskey method. Combinational logic Design: Adders, Subtractors, Binary Parallel Adder/ Subtractor, Look Ahead Carry Adder, Code Converters, Parity generators/checkers , Comparators, Encoders, Decoders, Multiplexers and De-multiplexers, Modular design using IC chips.	10 hrs
UNIT -3	
Flip-flops: Classification of Sequential Circuits, Latches & flip-flops - D flip-flop, JK flip-flop, T flip-flop. Flip-flop operating characteristics, Race around condition, Master slave flip-flop, conversion of one flip-flop to another, Applications of flip-flop.	10 hrs

Shift Registers: Buffer register, Data Transmission in Shift Registers, Serial-In Serial-Out Shift register, Serial-In Parallel-Out Shift register, Parallel-In Serial-Out Shift register, Parallel-In Parallel-Out Shift register, Bidirectional shift register, Universal Shift register, Applications of Shift register.	
UNIT -4	
Counters: Asynchronous counters, Design of asynchronous counters, Synchronous counters, Shift register counters. Sequential Circuits : Finite state model, Memory elements, Synthesis of synchronous sequential circuits, Serial Binary Adders, Sequence Detector.	10 hrs

TEXTBOOKS	
1	A. Anand Kumar; Fundamentals of Digital circuits; PHI, Second Edition
2	Thomas L. Floyd; Digital Fundamentals; Prentice Hall.
REFERENCES	
1	Morris Mano; Digital Logic and Computer Design; PHI Publication. .
2	Malvino& Leach; Digital Principles and Applications; TMH Publication.
3	R. P. Jain; Modern Digital Electronics; TMH Publication.

DATA STRUCTURES

Course Code	CE330		Credits	3	
Scheme of Instruction	L	T	P	TOTAL	
Hours/ Week	3	0	0	40 hrs/sem	
Scheme of Examination	IA	TW	TM	P	O
TOTAL = 125 marks	25	0	100	0	0

Course Outcomes:

The student will be able to:

CO1	Demonstrate the use of data structures like linked lists , stacks and queues
CO2	Explain the applications of linked lists, stacks and queues in Computer Engineering
CO3	Apply the knowledge of data structures to a given problem.
CO4	Illustrate searching, sorting and hashing techniques.

UNIT -1	
Introduction to Data Structures: Linear and Non Linear Data Structures. Linked lists: Concept of Linked Lists. Singly linked lists and its operations Stacks: Basic Stack Operations, Array implementation of Stacks , Polish Notation- Introduction to infix, prefix and postfix expressions Application of Stacks: Conversion of Infix to Postfix, Evaluation of Postfix expression Queues: Basic Queue Operations, Array implementation of Queues, Circular Queues. Application of Queues: Implementation of a palindrome	10 hrs
UNIT -2	
Linked list based implementation of Stacks, Linked list based implementation of Queues Doubly linked lists and circular linked lists and their operations Application of Linked Lists: Addition of two polynomials Binary Trees: Terms associated with binary trees, Strictly binary, Complete binary, Almost complete binary tree, Representation of trees - Linked array representation and Implicit array representation, Traversal in Binary Tree: Preorder, in-order, post-order and Level order traversal. Binary search tree - Insert, Delete, Search.	10 hrs
UNIT -3	
Threaded Binary tree – Insertion and Deletion in-threaded binary tree, Traversal: Inorder traversal of in-threaded binary tree, Preorder traversal of in-threaded binary tree AVL Tree: Insert, Delete with Rotations, Searching and sorting. B-tree: Searching, Insertion, Deletion from leaf node and non-leaf node. Graphs: Directed and undirected graphs, graph terminology, Adjacency matrix, Adjacency list, Graph Traversals - Breadth First Search, Depth First Search.	10 hrs

UNIT -4	
Searching: Linear Search, Binary Search Hashing: Hash functions, Collision resolution techniques Study of different sorting techniques: Bubble Sort, Selection Sort, Insertion Sort, Radix Sort, Mergesort, Heap sort, Shell sort	10 hrs

TEXTBOOKS	
1	S. K Srivastava, Deepali Srivastava; Data Structures through C in Depth; BPB Publications; 2011.
2	Aaron M. Tenenbaum; Data Structures using C; Pearson Education India .
REFERENCES	
1	Ellis Horowitz and SartajSahni, Fundamentals of Data Structures, Galgotia Book Source, Gurgaon, First edition/Recent edition.
2	Mark Allen Weiss, Data Structures and Algorithm Analysis in C, Pearson Education, 2nd Edition.
3	Gregory L. Heilman, Data Structures, Algorithms and Object Oriented Programming, Tata Mcgraw-Hill, New Delhi, 2002.
4	Jean-Paul Tremblay and Paul G. Sorenson, An Introduction to Data Structures with Applications, Second Edition, Tata McGraw-Hill, New Delhi, 1991.
5	Alfred V. Aho, John E. Hopcroft and Jeffry D. Ullman, Data Structures and Algorithms, Pearson Education, New Delhi, 2006.

OBJECT ORIENTED PROGRAMMING SYSTEM					
Course Code	CE340		**Credits**	**3**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**3**	**0**	**0**	**40 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 125 marks	**25**	**0**	**100**	**0**	**0**

Course Outcomes:

The student will be able to:

CO1	Design algorithms using principles of object oriented programming.
CO2	Demonstrate the concepts of data abstraction, encapsulation, code-reusability and data hiding using 'C++".
CO3	Explain the applications of polymorphism and inheritance in object oriented programming.
CO4	Apply the knowledge of standard template library achieve reusability

UNIT -1	
Basic concepts of Object-Oriented Programming: Objects, Classes, Data Abstraction and Encapsulation, Inheritance, Polymorphism, Dynamic Binding, Message Passing. Benefits of Object-Oriented Programming. Structure of a C++ program, Data types, Constants, tokens, expressions, control structures, functions, recursion, arrays.	10 hrs
UNIT -2	
Classes and Objects, Constructors and destructors, Friend functions and friend classes, Concepts of polymorphism: Function overloading, operator overloading. Overloading types, & rules, explicit & implicit type conversion operators, Pointers.	10 hrs
UNIT -3	
Inheritance: Introduction, Single, Multilevel, Multiple, Hierarchical, Hybrid. Virtual Base Class, Abstract classes. 'this' pointer, pointers to deriver classes Virtual functions, pure virtual functions. I/O streams and classes, managing output with Manipulators, Classes for file streams, file I/O operations and functions. String processing.	10 hrs
UNIT -4	
Functions Templates and Class Templates, Exception handling: Basics of Exception Handling, Exception Handling mechanism, Throwing Mechanism, Throwing Mechanism, Catching mechanism, Re-throwing mechanism. Introduction to the Standard Template Library: Components of STL, Containers and Adapter: stack, queue, priority queue adapter algorithms, Iterators, Applications.	10 hrs

TEXTBOOKS	
1	Paul Deitel and HarreyDietel; C++, How to Program; seventh edition.
2	E Balaguruswamy; Object oriented programming with C++; Tata McGraw Hill. 6th edition
REFERENCES	
1	K R Venugopal, Rajkumar, T. Ravishankar; Mastering C++; Tata McGraw Hill.
2	Stanley Lippman; C++ Primer; Fifth edition.
3	Herbert Schildt; Complete Reference; Fourth edition.
4	BjarneStroustrup; C++ Programming Language; Fourth edition.
5	D Ravichandran; Programming with C++; Third Edition.

`

COMPUTER ORGANISATION

Course Code	CE340		Credits	4	
Scheme of Instruction	L	T	P	TOTAL	
Hours/ Week	3	1	0	40 hrs/sem	
Scheme of Examination	IA	TW	TM	P	O
TOTAL = 150 marks	25	25	100	0	0

Course Outcomes:

The student will be able to:

CO1	Identify high performance architecture designand perform different computer arithmetic operations.
CO2	Create an assembly language program to program a microprocessor system.
CO3	Design a pipeline for consistent execution of instructions with minimum hazards.
CO4	Demonstrate memory hierarchy and its impact on computer cost/performance.

UNIT -1	
Introduction to Computer Organization: Computer components, Functions, interconnection Structure, Bus Interconnection. Computer Arithmetic: Integer Representation-unsigned numbers, signed numbers, signed magnitude, 2's compliment, Biased Representation. Integer Arithmetic: Addition, Subtraction, Multiplication unsigned, signed (Booths Algorithm), Division- unsigned, signed. Floating-Point Representation: IEEE 32 bits, 64 bits. Floating-Point Arithmetic: Addition, Subtraction, Multiplication, Division.	10 hrs
UNIT -2	
Internal Memory: Semiconductor Memory - Memory Hierarchy, Characteristics of Memory System, Semiconductor RAM Memories, Internal Organization of Memory Chip, Static RAM, Asynchronous DRAM, Synchronous DRAM, Connection of Memory to the processor, RAM Bus memory. Cache Memory: Basics of Cache, Structure, Read operation, Elements of Cache Design. Associative Memory: External Memory: Magnetic Disk, RAID, Optical Memory. Virtual Memory: Logical VS Physical Address space, working Principle, Mapping Functions, Replacement Policy. operators, Pointers.	10 hrs

UNIT -3	
Input/Output: External Devices, I/O Modules, Programmed I/O, Interrupt Driven I/O, Direct Memory Access, I/O Channel and Processor. CPU Structure and Functions: Processor Organization, Register Organization, Instruction Pipeline, Basic Concepts of Pipelining. RISC CPU Architecture: Instruction Execution Characteristics, Use of Large Register File, Compiler based register optimization, Reduced Instruction Set Architecture, RISC v/s CISC.	10 hrs
UNIT -4	
Buses: Bus interconnections, VGA, Asynchronous v/s Synchronous Buses, PCI Bus, SCSI Control Unit Operation: Micro Operations, Control of the CPU, Hardwired Implementation Micro programmed Control: Basic Concepts, Microinstruction Sequencing, and Microinstruction Execution. Parallel Processing: Multi Processing, Cache Coherence /MESI Protocol.	10 hrs

TEXTBOOKS	
1	William Stalling; A textbook of Computer Organization and Architecture; Edition VI.
2	David A. Patterson, John L. Hennesy ; Computer Organization And Design, Edition III.
REFERENCES	
1	M. Morris Mano ; A textbook of Computer Organization and Architecture.
2	Douglas V. Hall ; Microprocessors and Interfacing.
3	Carl Hamacher, ZvonkoVranesic, SafalZaky ; Computer Organization; Edition

`

Data Structures Programming Lab					
Course Code	CE360		**Credits**	**2**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**0**	**0**	**4**	**28 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 75 marks	**0**	**25**	**0**	**50**	**0**

At least 8 experiments out of below mentioned set are to be implemented using C

1. Implementation of Stack and Queue using Arrays.
2. Implementation of Stack and Queue using Linked lists.
3. Application of Stack: Infix to postfix Conversion, Postfix evaluation.
4. Implementation of Doubly Linked Lists .
5. Implementation of Circular Queues using Linked lists.
6. Implementation of Binary Search tree & its Operations & Traversals.
7. Implementation of Threaded Binary Search Tree.
8. Implementation of AVL Tree.
9. Implementation of Graph representations and Graph traversal techniques.
10. Implementation of Search techniques: Linear Search and Binary Search.
11. Implementation of Sorting techniques: Insertion Sort and Heap Sort.
12. Implementation of Sorting techniques: Merge Sort and Quick Sort.
13. Implementation of hash collision resolution techniques.

`

Object Oriented Programming Lab

Course Code	CE370		Credits	2	
Scheme of Instruction	L	T	P	TOTAL	
Hours/ Week	0	0	4	28 hrs/sem	
Scheme of Examination	IA	TW	TM	P	O
TOTAL = 75 marks	0	25	0	50	0

At least 8 experiments out of below mentioned set are to be implemented inclusive of mini-project using OOP paradigm

1. Basics of C++ (input /output / control statements / array).
2. Classes and objects.
3. Constructors and Destructors.
4. Function Overloading.
5. Operator Overloading.
6. Inheritance and Polymorphism.
7. Console I/O and Files.
8. Templates.
9. Exception Handling.
10. Standard Template Library.
11. Mini project using OOP paradigm

`

TECHNICAL COMMUNICATION					
Course Code	HM001		**Credits**	**2**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**2**	**0**	**0**	**2**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 75 marks	**0**	**75**	**0**	**0**	**0**

Course Outcomes:

The student will be able to:

CO1	Demonstrate precise language skills with suitable vocabulary and apt style.
CO2	Develop life skills/interpersonal skills to progress professionally.
CO3	Apply traits of suitable candidature for a job/higher education.
CO4	Deliver formal presentations and effectively implementing the verbal and non-verbal skills.

UNIT -1	7
Communication **Oral Communication** Listening, Speaking, Reading, Writing (LSRW), Conversational Dialogues, Role Play, Barriers to Oral Communication, Effective Oral Communication, Principles of Communication, Dos and Don'ts of Group Discussion **Global Communication** Social Media, People Analytics, Models of Culture, Cross-Cultural Communication, Compare Cultures of the World, Impact of Cultural Differences on Managerial Communication, Effective Communicator in a Cross-Cultural setting	
UNIT -2	7
Personality Development Social Etiquette, Email Etiquette, Table Etiquette, Telephone Etiquette, SWOC Analysis, Life Coaching, Emotional Intelligence, Leadership, Time Management, Motivation, Goal Setting, Team Work and Collaboration, Critical Thinking and Problem Solving, Professional Attitude, Persuasion, Anxiety and Stress Management, Social Responsibility	
UNIT -3	6
Career Development Resume Building, Interviewing Skills, Job Search, Personal Networking and Branding, Personal Finance, Build Professional Portfolio	
UNIT -4	6
Public Speaking Methods to overcome anxiety, Build Confidence, Use of Media Aids, Craft an Impactful Speech, Design Impactful Presentations, Effective Presentation Delivery	

TEXTBOOKS	
1	Meenakshi Raman and Sangeeta Sharma; Technical Communication: Principles and Practice, 3rd ed; Oxford University Press
2	Meenakshi Raman, Prakash Singh; Business Communication; 2nd ed.; Oxford University

	Press
3	Dr. K. Alex; Soft Skills: Know Yourself and Know The World; 3rd ed; S. Chand Publishing
REFERENCES	
1	Nicky Stanton; Mastering Communication; 5th ed.; Palgrave Master Series; Red Globe Press
2	Ghosh, B. N.; Managing Soft Skills for Personality Development; Tata McGraw Hill; 2012
3	Wallace and Masters; Personal Development for Life and Work;10thedition; Thomson Learning
4	Lehman, Dufrene, Sinha; BCOM : A South-Asian Perspective with CourseMate; 2ndedition; Cengage Learning
5	Ashraf Rizvi; Effective Technical Communication; Tata McGraw-Hill; 2005
6	MolefiKete Asante, William B. Gudykunst, Bella Mody; Handbook of International and Intercultural Communication; 2nd ed.; Sage Publications

MATHEMATICS-I& II (BRIDGE COURSE)					
Course Code	**AC390**		**Credits**	**0**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**2**	**0**	**0**	**28 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 0 marks	**0**	**0**	**0**	**0**	**0**

Course Outline:

This is an audit course.

This course is compulsory to direct second year/lateral entry students. It is introduced to reduce the knowledge gap in the students.

The syllabus is selected topics from FE110 Mathematics I and FE120 Mathematics II.

The Text books and References are same as shown in FE110 Mathematics I and FE120 Mathematics II.

DISCRETE MATHEMATICAL STRUCTURES

Course Code	CE410		Credits	4	
Scheme of Instruction	L	T	P	TOTAL	
Hours/ Week	3	1	0	40 hrs/sem	
Scheme of Examination	IA	TW	TM	P	O
TOTAL = 150 marks	25	25	100	0	0

Course Outcomes:

The student will be able to:

CO1	Well versed with relations and its various types, including module congruencies relations, which are widely used in computer sciences.
CO2	Well versed in propositional calculus and predicate calculus. Principals of mathematical inductions and Boolean algebra.
CO3	Well versed with the various counting techniques including pigeonhole principle, generating functions and recurrence relations.
CO4	Well versed with graphs and its various types such as Eulerian, Hamiltonian graphs, trees and its applications.

UNIT -1	
Set Theory: Sets, Set Operations, Relations and their properties, Equivalence Relations, partial orderings. **Functions:** One-to-One and Onto Functions, Inverse Function, Composition of functions, some important functions in computer science. **Integers:** Integers and division (excluding applications of congruences and cryptology), primes and greatest common divisors, Integers and algorithms.	10 hrs
UNIT -2	
Propositional Calculus: Propositional logic, propositional equivalences, predicates and quantifiers, rules of inference. **Boolean Algebra:** Boolean functions, representing Boolean functions. **Mathematical Induction:** Principle of Mathematical Induction and applications.	10 hrs
UNIT -3	
Counting: The basics of counting, permutations and combinations, binomial coefficients, pigeonhole principle. **Advanced Counting Techniques:** inclusion –exclusion principle, applications of inclusion –exclusion principle, generating functions, and Recurrence relations, solving linear recurrence relations.	10 hrs
UNIT -4	
Graph theory: Graphs and graph models, graph terminology and special types of graphs, representing graphs and graph isomorphism, connectivity, Euler and	10 hrs

Hamilton paths, shortest path problems, planar graphs, graph coloring. **Trees:** Introduction to Trees, applications of trees, tree traversal, Spanning Trees, Minimal Spanning Trees.	

TEXTBOOKS	
1	Kenneth H. Rosen; Discrete Mathematics and Its Applications; Tata McGraw Hill (6th edition).
2	B Kolman, R.C. Busby and Sharon C. Ross; Discrete Mathematical Structures; Prentice Hall
REFERENCES	
1	J. P. Tremblay and R. Manohar, McGraw Hill; Discrete Mathematical Structures with Applications to Computer Science; New York McGraw Hill.
2	Swapan Kumar Sarkar; Discrete Mathematics; S. Chand Publication.
3	Dr. D. S. C ; Discrete Mathematical Structures; Prism Books Pvt. Ltd.
4	G.V.Kumbhojkar; Discrete Structures And Graph Theory; Pradeep Prakashan.

MICROPROCESSORS & MICROCONTROLLERS					
Course Code	CE420		**Credits**	**3**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**3**	**0**	**0**	**40 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 125 marks	**25**	**0**	**100**	**0**	**0**

Course Outcomes:

The student will be able to:

CO1	To apply the assembly language programming to develop small real life embedded application.
CO2	To understand the architecture of the advanced processor thoroughly to use the resources for programming
CO3	To understand the higher processor architectures descended from 80386 architecture
CO4	To understand architecture and programming model of 8051 microcontroller with interfacing requirement

UNIT -1	
Systems Architecture- Systems Registers, Systems Instructions. **Memory Organization and Segmentation**- Global Descriptor Table, Local Descriptor Table, Interrupt Descriptor Table, Data Types, Registers, Instruction Format, Operand Selection, Interrupts and Exceptions Applications Instruction Set- Data Movement Instructions, Binary Arithmetic Instructions, Decimal Arithmetic Instructions, Logical Instructions, Control Transfer Instructions, String and Character Transfer Instructions, Instructions for Block Structured Language, Flag Control Instructions, Coprocessor Interface Instructions, Segment Register Instructions, Miscellaneous Instructions.	10 hrs
UNIT -2	
Memory Management- Segment Translation, Page Translation, Combining Segment and Page Translation. **Protection**- Need of Protection, Overview of 80386DX Protection Mechanisms, Segment Level Protection, Page Level Protection, Combining Segment and Page Level Protection. Multitasking- Task State Segment, TSS Descriptor, Task Register, Task Gate Descriptor, Task Switching, Task Linking, Task Address Space. **Input-Output-** I/O Addressing, I/O Instructions, Protection and I/O Exceptions and Interrupts- Identifying Interrupts, Enabling and Disabling Interrupts, Priority among Simultaneous Interrupts and Exceptions, Interrupt Descriptor Table (IDT), IDT Descriptors, Interrupt Tasks and Interrupt Procedures, Error Code, and Exception Conditions.	10 hrs
UNIT -3	

Initialization- Processor State after Reset, Software Initialization for Real Address Mode, Switching to Protected Mode, Software Initialization for Protected Mode, Initialization Example, TLB Testing **Debugging-** Debugging Features of the Architecture, Debug Registers, Debug Exceptions, Breakpoint Exception **Virtual 8086 Mode-** Executing 8086 Code, Structure of V86 Stack, Entering and Leaving Virtual 8086 Mode. 80387 NDP- Control Register bits for Coprocessor support, 80387 Register Stack, Data Types, Load and Store Instructions, Trigonometric and Transcendental Instructions, Interfacing signals of 80386DX with 80387	10 hrs
UNIT -4	
Architecture of 8051 – Special Function Registers(SFRs) - I/O Pins Ports and Circuits - Instruction set - Addressing modes - Assembly language programming. Programming 8051 Timers - Serial Port Programming - Interrupts Programming – LCD & Keyboard Interfacing - ADC, DAC & Sensor Interfacing - External Memory Interface- Stepper Motor and Waveform generation.	10 hrs

TEXTBOOKS	
1	Brey, Barry B, ―8086/8088, 80286, 80386 and 80486 Assembly Language Programming‖, Prentice Hall, ISBN: 13: 9780023142475.
2	Mohammad Rafiquzzaman, ―Microprocessors: Theory and Applications: Intel and Motorola", Prentice Hall, ISBN:-10:0966498011, 13:978:0966498011.
REFERENCES	
1	Microcontrollers –hardware ,architecture, programming- By Kenneth Ayala ,Second edition
2	James Turley, ―Advanced 80386 Programming Techniques‖, McGraw-Hill, ISBN: 10: 0078813425, 13: 978-0078813429.
3	Walter A. Triebel, ―The 80386Dx Microprocessor: Hardware‖, Software, and Interfacing, Pearson Education, ISBN: 0137877307, 9780137877300.
4	Muhammad Ali Mazidi, Janice Mazidi, DannyCausey -The x86 PC Assembly Language, Design and Interfacing, Fifth Edition, Pearson publications. ISBN 978-93-325-8404-4.Intel 80386 Programmer's Reference Manual 1986, Intel Corporation, Order no.: 231630- 011, December 1995.

FORMAL LANGUAGE AND AUTOMATA THEORY

Course Code	CE430		Credits	3	
Scheme of Instruction	L	T	P	TOTAL	
Hours/ Week	3	0	0	40 hrs/sem	
Scheme of Examination	IA	TW	TM	P	O
TOTAL = 125 marks	25	0	100	0	0

Course Outcomes:

The student will be able to:

CO1	Identify formal language classes and explain the properties of languages, grammars and automata.
CO2	Apply the techniques to transform between equivalent deterministic and non-deterministic finite automata and regular expressions.
CO3	Design grammars and automata (recognizers) for different language classes. Perform the Simplification of automata and Context free grammars.
CO4	Explain the concepts of context-free languages, pushdown automata and Turing recognizable languages.

UNIT -1	
Introduction: Languages, Grammars and Automata. Finite Automata: Deterministic Finite Accepters, Nondeterministic Finite Accepters, Equivalence of Deterministic and Nondeterministic Finite Accepters, Reduction of the Number of States in Finite Automata.	10 hrs
UNIT -2	
Regular Languages and Regular Grammars: Regular Expressions, Connection Between Regular Expressions and Regular Languages, Regular Grammars, Closure properties of Regular languages , A Pumping Lemma for regular languages. Finite State Transducers: Mealy Machine, Moore Machine, Moore and Mealy Machine Equivalence.	10 hrs
UNIT -3	
Context-Free Languages: Examples of Context Free Languages, Leftmost and Rightmost Derivations, Derivation Trees, Parsing and Ambiguity, Methods for Transforming Context Free Grammars, Chomsky Normal Form, and GreibachNormal Form. Nondeterministic Pushdown Automata, Pushdown Automata and Context-Free Languages, Deterministic Pushdown Automata, Pumping Lemma for Context-Free Languages. Closure of Context Free languages.	10 hrs
UNIT -4	
Turing Machine: Standard Turing Machine, Combining Turing `s for Complicated Tasks, Turing's Thesis. Turing Machines with More Complex Storage. Nondeterministic Turing Machines. A Universal Turing Machine. Linear Bounded Automata. Computability and Decidability: Turing Machine Halting Problem. Unrestricted Grammars, Context-Sensitive Grammars.	10 hrs

TEXTBOOKS	
1	Peter Linz; An introduction to Formal Languages and Automata; Jones & Bartlett Learning, 2006
2	John C Martin; Introduction to languages and the theory of computation; Tata McGraw Hill, Fourth Edition, 2010.
REFERENCES	
1	John E. Hopcraft and Jeffery D. Ullman; Introduction to Automata Theory, Languages and Computation; Narosa Publishing House.
2	Michael Sipser; Introduction to Theory of Computation; PWS Publishing Company.
3	A.A Puntambekar; Formal Languages and Automata Theory; Technical Publications Pune.
4	K.L.P Mishra, N. Chandrasekaran; Theory of Computer Science – Automata, languages and Computation; PHI Publications; Third Edition ; 2008.

MODERN ALGORITHM DESIGN FOUNDATION					
Course Code	CE440		**Credits**	**3**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**3**	**0**	**0**	**40 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 125 marks	**25**	**0**	**100**	**0**	**0**

Course Outcomes:

The student will be able to:

CO1	Demonstrate how the different algorithm design approaches are used to solve various classes of engineering problems.
CO2	Compute and analyze the time and space complexities of algorithms and understand their rate of growth.
CO3	Implement the algorithms with help of different data structures.
CO4	Describe the different algorithm classes P, NP, and NP-Complete, Randomized, Probabilistic, Approximation.

UNIT -1	
Introduction: Algorithm Specification, Performance Analysis, and Analyzing of algorithms: Insertion sort, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem. Divide and Conquer: General method, Binary search, Finding the min and max, Merge sort, Quick sort: Sorting by partitioning, Selection: Finding the kth smallest element, Stassen's Matrix Multiplication.	10 hrs
UNIT -2	
Greedy Method: General Method, Knapsack Problem, Minimum cost Spanning tree, Single soured shortest path. Dynamic Programming: General Method, Multistage Graphs, All pair shortest paths, Single source shortest path with general weights, Optimal Binary Search Tree, 0/1 knapsack problem, Travelling salesperson problem.	10 hrs
UNIT -3	
Backtracking: General Method, n-queens problem, Sum of subsets problem, graph colouring, Hamiltonian Cycles, 0/1 knapsack problem. Branch-and-Bound: General Method, 0/1 knapsack, Travelling salesperson problem.	10 hrs
UNIT -4	
Internet Algorithms: String and pattern matching, Tries, Text compression, Text similarity testing.NP-hard and NP-complete problems: Basic concepts, Cooks theorem, Introduction: Randomized Algorithms, Probabilistic	10 hrs

algorithms, Approximation algorithms.

TEXTBOOKS	
1	Fundamentals of Computer Algorithms – E. Horowitz et al, 2nd Edition UP.
2	Introduction to Algorithms, 3th Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
REFERENCES	
1	Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley.
2	Fundamentals of Algorithmics, Gilles Brassard, Paul Bratley, PHI
3	Algorithm Design, 1ST Edition, Jon Kleinberg and ÉvaTardos, Pearson.
4	Algorithms -- A Creative Approach, 3RD Edition, UdiManber, Addison-Wesley,Reading, MA.

OBJECT ORIENTED SOFTWARE ENGINEERING					
Course Code	CE450		**Credits**	**4**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**3**	**1**	**0**	**40 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 150 marks	**25**	**25**	**100**	**0**	**0**

Course Outcomes:

The student will be able to:

CO1	Specify a software system.
CO2	Create an object-oriented design for object oriented software engineering.
CO3	Implement with readable, reusable, modular, object-oriented techniques.
CO4	Test for validity, correctness and completenessand understand .software management process

UNIT -1	
Introduction to Software Engineering:Scope of software engineering- Historical aspects, Economic aspects, Maintenance aspects, Specification and design aspects, Team programming aspects. **The Software Process-** Client, Developer and User Phases of SDLC Life Cycle, Requirement phase, Specification phase, Design phase, Implementation phase Integration phase, Maintenance phase, **Software Life Cycle Models** Build and Fix Model, Waterfall, Rapid Prototyping Model, Incremental Model, Extreme Programming, Synchronize and Stabilize Model, Spiral Model, Object Oriented Life Cycle Model. **Software Metrics** Capability Maturity Model. **Estimating Duration and Cost** Metrics for size of product, Techniques for cost estimation and models, **Teams :** Team Organization Democratic Team Approach, Classical chief Programmer Team Approach, Synchronize and Stabilize Teams	10 hrs
UNIT -2	
Object Oriented Software Engineering: Object Oriented System Development, Object Oriented Terminology, Types of Cohesion, Types of Coupling, Data Encapsulation, Software re-usability, Portability, Interoperability, CASE tools in use for Object Oriented Software Engineering. **Requirement Phase:** Techniques for Requirement Elicitation and Analysis Metrics for Requirement Phase, Testing and CASE tools for Requirement Phase. **Specification Phase:** Specification Document, Metrics for Specification Phase, Testing and CASE tools for Specification Phase **Analysis Phase:** OO Analysis, Use Case Modeling, Class Modeling, Dynamic Modeling, Testing and CASE tools for Analysis Phase **Design Phase:** Action oriented Design and Abstraction, DFA, Data Oriented Design, Object Oriented Design, Testing and CASE tools for Design Phase	10 hrs
UNIT -3	
Software Quality Assurance: Quality Concepts, Quality Movement, Software Reviews, Formal Technical Reviews, Formal Approaches to SQA, Statistical	10 hrs

SQA, Software Reliability, SQA Plan **Software Testing:** Fundamentals, Test Case Designs, White Box Testing, Basic Path Testing, Control Structure Testing, Black Box Testing, Testing for specialized environment **Software Testing Strategies:** Strategic Approach to Software Testing, Strategic Issues, Unit Testing, Integration Testing, Validation Testing, Organizational approaches to testing, Software testing tools- for classical engineering and object oriented engineering Software testing standards **Object Oriented Testing**	
UNIT -4	
Software Project management: Managing software project, Project planning Process planning- Standard process, Requirement change management, Quality Planning, Risk management, Project management plan, Team structure, Communication, Team development and configuration management. Project execution, Project monitoring and control Project Closure, Performing closure analysis, Closure analysis report.	10 hrs

TEXTBOOKS	
1	Object Oriented and Classical Software Engineering- Stephen R.Schah(TMH)
2	Software Project Management in practice- Pankaj Jalote- PEA
REFERENCES	
1	Software Engineering – A practitioner's approach – by Roger S. Pressman, McGraw Hill
2	A discipline for Software Engineering – by Watts S. Humprey, Pearson Education
3	Software Engineering – by K. K. Aggarwal and Yogesh Singh, New Age Publications
4	'Ed-Kit'- Software testing in real world. Addison Wesley 1995
5	Effective methods for software testing(second edition) John-Wiley 1999
6	Software testing techniques(2nd edition) Van Nostrand Rein loud 1990
7	The art of software testing, Jon Wiley Mayers G.J.

MODERN ALGORITHM DESIGN FOUNDATION LAB					
Course Code	CE460		**Credits**	**2**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**0**	**0**	**4**	**28 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 75 marks	**0**	**25**	**0**	**50**	**0**

At least 8 experiments out of below mentioned set are to be implemented.

1. Write a program to implement binary search using divide and conquer.
2. Write a program to implement Merge Sort using divide and conquer.
3. Write a program to implement Quick Sort using divide and conquer.
4. Write a program to implement minimum cost spanning trees using greedy approach.
5. Write a program to implement single source shortest path algorithm using greedy approach.
6. Write a program to implement 0/1 knapsack problem using dynamic programming.
7. Write a program to implement OBST using dynamic programming.
8. Write a program to implement single source shortest path algorithm using dynamic programming.
9. Write a program to implement sum of subset problem using backtracking.
10. Write a program to implement graph colouring problem using backtracking.
11. Write a program to implement pattern matching algorithms
12. Write a program to implement text compression and text similarity testing.

MICROPROCESSORS & MICROCONTROLLERS LAB					
Course Code	CE460		**Credits**	**2**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**0**	**0**	**4**	**28 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 75 marks	**0**	**25**	**0**	**50**	**0**

At least 8 experiments out of below mentioned set are to be implemented inclusive of mini-project on 8051

1. Write a program which illustrates the programming constructs of higher level language in 80386 assembly coding.
2. Write a program which contains the following macros
    a. For calculating the Fibonacci series for N
    b. For entering the value of N.
    c. For displaying the numbers.
3. Write above program using procedures.
4. Write a procedure to implement the following sorting algorithms.
    a. Bubble Sort
    b. Insertion sort
    c. Selection sort.
5. Write a program to implement the following searching algorithms.
    a. Linear Search
    b. Binary Search
6. Write a procedures to implement the library routine to:
    a. Input integer number(Read integer number)
    b. Output Integer number(write Integer Number)
    c. Input string
    d. Output String
7. Write a program to make use of int 10h for the following:
    a. Sets the video mode and clears the screen
    b. Makes a window of a specific size and color
    c. Sets the cursor at a specified position within the window.
    d. Displays 10 times the character '*' at the cursor position.
8. Write a program to display 'Hello World' vertically downwards at the centre of the screen.
9. Store a password in memory. Enter another password through the keyboard and verify if it matches the stored password. The password entered should not be displayed as such , but each letter should be displayed as '*'.

Mini Project on 8051 microcontroller for Hardware implementation of any one application.

ECONOMICS FOR ENGINEERS					
Course Code	**EE 470**		**Credits**	**3**	
Scheme of Instruction	**L**	**T**	**P**	**TOTAL**	
Hours/ Week	**3**	**0**	**0**	**40 hrs/sem**	
Scheme of Examination	**IA**	**TW**	**TM**	**P**	**O**
TOTAL = 125 marks	**25**	**0**	**100**	**0**	**0**

Course Outcomes:

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

CO1	To acquire the skills to apply the basics of economics to Engineering
CO2	To evaluate the economic theories, cost concepts and pricing policies
CO3	To calculate National Income, Inflation and Price Index
CO4	To evaluate the different measures of Economic Growth & Development.

UNIT 1	
Central concepts of Economics- Definitions of Economics , Scarcity and Efficiency, Nature of Economics: Positive and normative economics, Microeconomics and Macroeconomics Basic Elements of Supply and Demand- The Demand Schedule, The Demand Curve, Market Demand , Forces behind the Demand Curve, Shifts in Demand. The Supply Schedule The Supply Curve, Forces behind the Supply Curve , Shifts in Supply. Equilibrium of Supply and Demand, Effect of a Shift in Supply or Demand. Supply and Demand: Elasticity and Applications to major economic issues **Estimation/Forecasting of Demand:** Meaning, importance, methods – trend, exponential smoothing, regression analysis	10 Hours
UNIT 2	
Microeconomics: Demand & Consumer Behaviour- Choice & Utility Theory. Production and Business Organization, Theory of Production and Marginal Products Basic Concepts, The Nature of the Firm, Big, Small, and Infinitesimal Businesses. Economic Analysis of Costs, Total Cost: Fixed and Variable. Production, Cost Theory, and Decisions of the Firm. Market structures.Perfect and imperfect competition, oligopoly, monopoly.	10 Hours
UNIT 3	
Macroeconomics: Key Concepts of Macroeconomics. Objectives and Instruments of Macroeconomics. Aggregate Supply and Demand. **National Income Terms:** -Gross Domestic Product: The Yardstick of an Economy's Performance. Real vs. Nominal GDP. Net Domestic Product, GNP, National Income, Per capita income, Disposable Income, Price Index, Inflation. Consumption and Investment- Consumption, Income, and Saving, Investment. Determinants of Investment. Monetary Policy and the Economy .Government Control of the Economy- The Tools of Government Policy	10 Hours
UNIT 4	
Economic Growth and Development: Economic Growth- The Long-Term Significance of Growth, The Four Wheels of Growth. Economic Development- meaning, criteria, measures of development- Per Capita Income, Index of	10 Hours

Human Development . Financial markets- Structure, Participants, functions. Capital market- Instruments, Players, trading - Primary and secondary market - Role of stock exchanges and stock indices. Money market	

	Textbooks
1	P.A. Samuelson & W.D. Nordhaus, Economics, 19th Edition McGraw Hill, New York, 1995.
2	A. Koutsoyiannis, Modern Microeconomics, Macmillan, 1975.
3	O.P. Khanna , Economics for Engineers,VK Global Publications Private Limited.
	References
1	Chandra P., Fundamentals of Financial Management, Tata McGraw Hill Education Private Limited, New Delhi