Syllabus of B. Tech. Computer Engineering (COE) for 3rd to 8th Semester

Course Title	Linear Algebra	Course No	To be filled by the office		
Specialization	Mathematics	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■	Elective ☐	
Course Objectives	To impart knowledge of basic concepts and applications of Linear Algebra				
Course Outcomes	At the end of the course, a student will be able to show that they get clear understanding of methods of Linear Algebra.				
Contents of the course *(With approximate break up of hours)*	**Linear System of Equations**: Gaussian Elimination—echelon forms—existence, uniqueness and multiplicity of solutions of linear equations. (6) **Vector Spaces**: Definition—linear dependence and independence—spanning sets, basis, and dimension—definition of a subspace—intersection and sum of subspaces—direct sums. (8) **Linear Transformations**: Definition—matrix representation of a linear transformation—change of basis—similarity transformation—invertible transformation—system of linear equations revisited—the four fundamental subspaces associated with a linear transformation. (10) **Inner Products**: Definition—induced norm—orthogonality—Gram-Schmidt orthogonalization process—orthogonal projections—unitary transformations and isometry. (8) **Eigen Decomposition**: Eigenvalues and eigenvectors—characteristic polynomials and eigen spaces—diagonalizability conditions—invariant subspaces—spectral theorem. (10)				
Textbook	1. G. Strang, "Linear Algebra and its Applications," Cengage Learning, 4th Edition, 2005. 2. D. C. Lay, "Linear Algebra and its Applications," Pearson Education, 4th edition, 2011.				
References	1. C. D. Meyer, "Matrix Analysis and Applied Linear Algebra," SIAM, 2000. 2. S. H. Friedberg, A. J. Insel, and L. E. Spence, "Linear Algebra," Pearson Education, 4th Edition, 2002.				

Course Title	Engineering Economics	Course No	To be filled by the office		
Specialization	Management	Structure (LTPC)	3	0	3
Offered for		Status	Core	Elective	
Pre-requisite	Basic Mathematics	To take effect from			
Course Objectives	Help students learn basics of economics and cost analysis to make economically sound design decisions				
Course Outcomes	This course will help students understand: • the basics of micro-economics and cost analysis • Techniques to make economically sound decisions				
Contents of the course *(With approximate break up of hours)*	• Engineering Economic Decisions • Time is Money • Understanding Financial Statements • Cost Concepts and Behaviors • Understanding Money and Its Management • Principles of Investing • Present Worth Analysis • Annual Equivalent Worth Analysis • Rate of Return Analysis • Depreciation • Capital Budgeting Decisions				
Textbook	1. John A. White, Kellie S. Grasman, Kenneth E. Case, Kim LaScola Needy, David B. Pratt, "Fundamentals of Engineering Economic Analysis (First Edition)," Wiley 2014. 2. Chan S.Park, "Contemporary Engineering Economics," Prentice Hall of India, 2002.				
References	1. Blank Tarquin (2005). Engineering Economy. 6th Edition. McGraw-Hill.				

Course Title	Discrete Structures for Computing	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	This course introduces logical reasoning, inferences, and proof techniques. Relations, Functions, Counting principles are also discussed. Graph theory and various properties of graphs are also taught as part of this course.				
Course Outcomes	The learner would appreciate the importance of combinatorics and the various proof techniques, and in particular, in proving the correctness of algorithms. Counting principles learnt as part of the course will help the learner in counting various combinatorial objects				
Contents of the course	Mathematical Reasoning – Propositions – Predicates –First order logic –Methods of proof (10) Set theory – Relations between sets – Operation on sets –Inductive definition of sets (5) Binary relation and digraphs – Special properties of relations – Composition of relations – Closure operations on relations (5) Basic properties of functions – Inductively defined functions – Special classes of functions – Inverse functions, functions , Asymptotic growth of functions –(8) Basic counting techniques – Recurrence systems – Solving recurrence relations. Finite and Infinite sets –Countable and uncountable sets–Cardinal numbers (10) Graph Theory –Graphs – Sub graphs – Isomorphic and Homeomorphic graphs – Paths – Connectivity Bridges of Konisberg – Labeled and Weighted Graphs – Complete, Regular and Bipartite Graphs –Planar Graphs – Coloring (7)				
Textbook	1. K. H. Rosen, “Discrete Mathematics and its Applications,” McGraw Hill, 6th Edition, 2007.				
References	1. D. F. Stanat and D. F. McAllister, “Discrete Mathematics in Computer Science,” Prentice Hall, 1977. 2. R. L. Graham, D. E. Knuth, and O. Patashnik, “Concrete Mathematics,” Addison Wesley, 1994. 3. Busby, Kolman, and Ross, “Discrete Mathematical Structures,” PHI, 6th Edition, 2008. 4. C. L. Liu, “Elements of Discrete Mathematics,” Tata McGraw Hill, 1995.				

Course Title	Digital and Analog Circuits Design	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	To introduce the basic understanding of digital representation, Boolean algebra and the operation of the logic components, combinational and sequential circuits, and to introduce the analog device concepts like diode, FET and op-amp.				
Course Outcomes	Students shall be able to construct digital circuits and systems for real life applications, and design amplifiers, analog to digital and digital to analog converters.				
Contents of the course	Digital Circuits: Number Representation: Fixed point and floating point, 1's and 2's complement. Switching Theory: Boolean algebra, Switching functions, Truth Tables and Algebraic forms, Simplification of Boolean expressions – Algebraic methods, canonical forms and Minimization of functions using K-Maps. (5) Binary Codes: BCD, Gray, Excess 3, Alpha Numeric codes and conversion circuits. (3) Arithmetic circuits: Binary adders and subtractors, multipliers and division, ALU. (5) Synthesis of combinational logic functions using MSIs: mux/demux, decoders/encoders, Priority encoders, Comparators. (2) Sequential Circuits: Latches and Flip-Flops: SR, JK, D, T; Excitation tables. (2) Shift Registers, Counters, Random Access Memory. (3) Synchronous sequential circuits: Finite State Machines- Mealy & Moore types- Basic design steps- Design of counters, sequence generators, and sequence detectors - Design of simple synchronous machines – state minimization. (8) Analog Circuits: Diodes – Basics and Circuits – Clippers, Clampers, rectifiers. (3) Transistors –Basics of Bipolar Junction Transistor and Field Effect Transistors – operating modes, amplifier circuits. (3) Operational amplifiers (op-amp) – Basics and op-amp circuits – non inverting and inverting amplifiers – Signal offset. (3) Analog to Digital and Digital to Analog Conversion and circuits, Applications of Digital ICS: 555 Timer, V to F converters, Introduction to Logic Families, Noise in Digital System. (5)				
Textbook	1. M. Mano and C. Kime, "Logic and Computer Design Fundamentals," Prentice Hall, Upper Saddle River, NJ, 4th Edition, 2008. 2. B. Razavi, "Fundamentals of Microelectronics," Wiley Student Edition, 2010.				
References	1. Sedra and Smith, Microelectronic Circuits, 7th Edition, Oxford University Press. 2. J. F. Wakerly, "Digital Design - Principles and Practices," 3rd Edition, Pearson. 3. M. M. Mano, "Digital Design," PHI, 1979. 4. S. Franco, "Design with Operational Amplifiers and Analog Integrated Circuits," McGraw-Hill Series in Electrical and Computer Engineering, 4th Edition, 2015. 5. R. J. Tocci, N. S. Widmer, and G. L. Moss, "Digital Systems Principles and applications," Pearson Prentice Hall,10th Edition.				

Course Title	Signals, Systems, and Communication	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	The objective of this course is to introduce the students to the concepts of discrete time signals and systems, and their significance in practice. Further, the basics of digital communication like various digital modulation and demodulation techniques are introduced.				
Course Outcomes	At the end of the course, the students will have learnt about digital signal, analyze an LTI system with its impulse and frequency response. Further, students will be able to design an IIR filter (e.g., LPF and HPF). In the digital communication front, students will have learnt various digital modulation techniques and analyze their BER performance.				
Contents of the course	<u>Signal and Systems</u> Types of signals, operation on signals, discrete time systems,-static, dynamic, stable, unstable, causal, LTI system, correlation –auto,cross correlation, properties, computation, Analog to digital conversion (8) <u>Signal Processing</u> Discrete Fourier Transform- Properties, Convolution- circular, linear, comparison (8) Fast Fourier Transform: DIT-FFT (4) Butterworth Filter design: low-pass, high-pass (4) <u>Communications</u> Modulation, need for modulation, Frequency Modulation, (8) ASK,FSK,BPSK-BER performance, QAM. (8)				
Textbook	1. A. Oppenheim, R. Schafer, and J. Buck, "Discrete-Time Signal Processing," Pearson, 2007. 2. S. Haykin and M. Moher, "An Introduction to Analog and Digital Communications," Wiley, 2nd Edition, 2001.				
References	1. S. K. Mitra, "Digital Signal Processing," McGraw Hill, 2nd Edition. 2. B. P. Lathi, "Modern Digital and Analog Communication Systems," Oxford Press, 2008.				

Course Title	Programming and Data Structures	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	The objective of the course is to teach programming (with an emphasis on problem solving) and introduce elementary data structures. The student should, at a rudimentary level, be able to prove correctness (loop invariants, conditioning, etc) and analyze efficiency (using the `O' notation).				
Course Outcomes	At the end of the course, students will be able to design data structures so that efficient algorithms that make use of those data structures to solve a given problem				
Contents of the course	1. Review of Problem Solving using computers, Abstraction, Elementary Data Types: Algorithm design- Correctness via Loop invariants as a way of arguing correctness of programs, preconditions, post conditions associated with a statement. (3 lectures) Complexity and Efficiency via model of computation (notion of time and space), mathematical preliminaries, Elementary asymptotics (big-oh, big-omega, and theta notations). (3 lectures) 2. ADT Array -- searching and sorting on arrays: Linear search, binary search on a sorted array. Bubble sort, Insertion sort, Merge Sort and analysis; Emphasis on the comparison based sorting model. Counting sort, Radix sort, bucket sort. (6 lectures) 3. ADT Linked Lists, Stacks, Queues: List manipulation, insertion, deletion, searching a key, reversal of a list, use of recursion to reverse/search. Doubly linked lists and circular linked lists. (3 lectures) Stacks and queues as dynamic data structures implemented using linked lists. Analyse the ADT operations when implemented using arrays. (3 lectures) 4. ADT Binary Trees: Tree representation, traversal, application of binary trees in Huffman coding. Introduction to expression trees: traversal vs post/pre/infix notation. Recursive traversal and other tree parameters (depth, height, number of nodes etc.) (4 lectures) 5. ADT Dictionary: Binary search trees, balanced binary search trees - AVL Trees. Hashing - collisions, open and closed hashing, properties of good hash functions. (8 lectures) 6. ADT Priority queues: Binary heaps with application to in-place sorting (5 lectures) 7. Graphs: Representations (Matrix and Adjacency List), basic traversal techniques: Depth First Search + Breadth First Search (Stacks and Queues) (7 lectures)				
Textbook	1. M. A. Weiss, “Data Structures and Algorithm Analysis in C,” Addison-Wesley, 1997.				
References	1. Cormen T.H, Leiserson C.E and Rivest R.L, “Introduction to Algorithms,” Prentice Hall India, 2nd Edition, 2001. 2. Aho, Hopcroft and Ullmann, “Data Structures and Algorithms,” Addison Wesley, 1983. 3. Adam Drozdek, “Data structures and Algorithms in C,” 1994. 4. R G Dromey, “How to solve it by Computer,” PHI, 1982. 5. Horowitz, Sahni and Anderson-Freed, “Fundamentals of Data Structures in C,” Silicon Press, 2007.				

Course Title	Digital and Analog Circuits Design Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	To provide hands on design and implementation of analog and digital circuits. Students will build simple digital systems on general purpose PCBs.				
Course Outcomes	Students shall be equipped with the skill set required for the construction of digital and analog circuits for real time applications using ICs.				
Contents of the course	Design and implementation of logic functions, combinational circuits (code converters, half & full adders, comparator, ripple carry adder, priority encoder, Decoders, Seven segment display, multiplexer) – Design of sequential Circuits. Design of 4-bit ALU (Adder, subtractor, logic and shift operations). Design project Static characteristics of rectifiers and filters, clipping and clamping circuits, Op-Amp based amplifier circuits				
Textbook	1. S. Franco, "Design with Operational Amplifiers and Analog Integrated Circuits," McGraw-Hill Series in Electrical and Computer Engineering, 4th Edition, 2015. 2. S. Brown and Z. Vranesic, "Fundamentals of Digital Logic with VHDL Design," TMH, 3rd Edition.				
References	1. R. J. Tocci, N. S.Widmer, and G. L. Moss, "Digital Systems Principles and applications," Pearson Prentice Hall, 10th Edition. 2. D. A. Newman, "Electronic Circuits," TMH, 4th Edition.				

Course Title	Data Structures Practice Using C-Programming	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	Data Structure plays an important role in solving problems efficiently. Unless data are arranged in an efficient way, the algorithms which use the data cannot run efficiently. This course helps students to design and implement data structures to solve real world/mathematical problems.				
Course Outcomes	At the end of the course, students will be able to design efficient data structure which will be used by efficient algorithms to solve real problems.				
Contents of the course	The laboratory component will require the student to write computer programs using a careful choice of data structures (in C language) from scratch, based on the concepts learnt in the theory course. Arrays: Linear and Binary search(1)- Array and Pointer based implementation of list, stack and queue (2) - Application of linked lists – Polynomial manipulations (1) - Representing sets using lists and implementation of set theoretic operations(1) - Expression conversion(1) and evaluation of postfix expressions(1) - Binary trees (1)- binary search trees(2), AVL Trees and dictionary ADT using AVL trees(2)- Heap and Priority queue ADT implementation using Heap(2) –Hashtables(1)				
Textbook	1. M. A. Weiss, “Data Structures and Algorithm Analysis in C++,” Pearson Education, 2nd Edition, 2002.				
References	1. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, “Introduction to Algorithms,” Prentice Hall India, 2nd Edition, 2001. 2. Aho, Hopcroft, and Ullmann, “Data Structures & Algorithms,” Addison Wesley, 1983.				

Course Title	Probability Theory	Course No	To be filled by the office		
Specialization	Mathematics	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■	Elective ☐	
Course Objectives	To impart knowledge of basic concepts and applications of Probability and Statistics				
Course Outcomes	At the end of the course, a student will be able to apply the knowledge in solving engineering problems				
Contents of the course *(With approximate break up of hours)*	Introduction to Probability: Sets, Events, Axioms of Probability, Conditional Probability and Independence, Bayes Theorem and MAP Decision Rule (8) Random Variables: Definitions, Cumulative Distribution Functions, mass and density functions, joint and conditional distributions, Functions of Random Variables (8) Expectations: Mean, Variance, Moments, Correlation, Chebychev and Schwarz Inequalities, Moment-generating and Characteristic Functions, Chernoff Bounds, Conditional Expectations (8) Random Vectors: Jointly Gaussian random variables, Covariance Matrices, Linear Transformations, Diagonalization of Covariance Matrices (6) Random Sequences: Sequences of independent random variables, correlation functions, wide-sense stationary sequences, LTI filtering of sequences (6) Law of Large Numbers, Central Limit Theorem (6)				
Textbook	1. Stark and Woods, "Probability and Random Processes with Applications to Signal Processing," 3rd Edition, Pearson Education 2002. 2. S. Ross, "A First Course in Probability," 6th Edition, Pearson.				
References	1. J. S. Milton and J. Arnold, Introduction to Probability and Statistics, Tata McGraw Hill Education Private Limited, 4th Edition, 2006. 2. S. Kay, Intuitive Probability and Random Processes Using MATLAB, Springer, 2008. 3. R. M. Gray and L. D. Davisson, "An Introduction to Statistical Signal Processing," Cambridge University Press, 2004.				

Course Title	Sociology of Design	Course No	To be filled by the office		
Specialization	Management	Structure (LTPC)	3	0	3
Offered for	UG	Status	Core ■	Elective □	
Pre-requisite	None	To take effect from			
Course Objectives	Design as a Social Activity – Level 1				
Course Outcomes	This course will help students understand • Design as a social activity involving people, their relationships & values - How designs can emerge out of or be constrained by social patterns of relating • How technology can influence interactions among people, cooperative work, ethical issues around technology interventions • Exposure to techniques like ethnomethodology				
Contents of the course *(With approximate break up of hours)*	Basics concepts of sociology (behavior, interaction, language) **[6]** Historical evolution of Societies (Agrarian, Industrial, Digital) and current human and organizational contexts in which engineers and other professionals work, Personal and corporate social responsibility & ethics **[10]** Relationship between people (age, gender, cultures) and technology - Social and psychological dimensions of technological change, Technology & Work, Co-operative Work & Coordinative Practices, Ethnomethodology, Critical Systems Heuristics **[10]**				
Textbook and References	1. Manuel Castells (1996); The Rise of Network Society. 2. Herbert Blumer (1986); Symbolic Interactionism: Perspective and Method. 3. Herkert, J. (ed.), Social, Ethical, and Policy Implications of Engineering: Selected Readings. New York, NY: IEEE Press, 2000. 4. Heath, C. and Luff, P. (2000); Technology in Action, Cambridge: Cambridge Univ Press. 5. Werner Ulrich (1983), Critical Systems Heuristics, John Wiley, London.				

Course Title	Design and Analysis of Algorithms	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	3
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	Data Structure and Algorithm course is essential to understand many areas in Computer Science and Engineering. This course also trains the students to solve problems using computer.				
Course Outcomes	At the end of the course, students will be able to design data structures and efficient algorithms to solve given problem.				
Contents of the course	Introduction to Asymptotic Notation – Solving Recurrence relations – Master's theorem – Recurrence Tree method (8) Incremental and Decremental Algorithm Design Strategies – case studies, lower bound for sorting (3) Divide & Conquer – Merge – Quick sort – Median Finding- (6) Greedy algorithms – knapsack problem (fractional and 0/1 versions) - Minimum spanning tree – Prims- Kruskal's algorithm- Huffman coding, Set of Intervals (6) Dynamic programming – case studies — LCS-Matrix Multiplication – Knapsack (7) Graph algorithms – Topological sort – Shortest path algorithms – Dijskstra's Algorithm, – Bellman-Ford's Algorithm (5) Solvability & Tractability – Introduction to unsolvable problem-Hatling problem- Introduction to NP-completeness – Search/Decision, SAT, Independent set, VC, X3C, Hamilton circuit, etc Backtracking – n queen problem-subset problem - Branch & Bound- Job Scheduling problem (10)				
Textbook	1. E. Horowitz, S. Sahni, and S. Rajasekaran, "Computer Algorithms," 2nd Edition, Galgotia Publications, 2007.				
References	1. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, "Introduction to Algorithms," Prentice Hall India, 2nd Edition, 2001. 2. Aho, Hopcroft, and Ullmann, "Data Structures & Algorithms," Addison Wesley, 1983.				

Course Title	Database Systems	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	The focus of this course is on database design, architecture, and relational models. Normal forms, internal schema design would also be explored				
Course Outcomes	Learner would appreciate the systematic design and principles involved in any database development. The importance of canonical normal forms and its design in large scale database systems would be a secondary outcome of this course				
Contents of the course	Introduction to Database Systems, Database System Architecture, Schema, Database Models, Relational Model, ER Modelling and case studies. (7) Expressive power of relational databases, Relational Algebra (5) Database Languages, DDL, DML, Structured Query Language (SQL), SQL views, case studies (8) Database Design, Normal Forms (First to third normal form), Boyce codd Normal Form, Database decomposition, Functional Dependencies, Loss-less Join decomposition(8) Transaction Processing and Concurrency control (4) Internal schema Design, Indexing, B-trees, B+ trees (5) Introduction to advanced concepts like Data mining, Data warehousing, XML (5)				
Textbook	1. R. Elmasri and S. B. Navathe, "Fundamentals of Database Systems," Pearson, 4th Edition, 2007.				
References	1. A. Silberschatz, H. F. Korth, and S. Sudharsan, "Database System Concepts," Tata McGraw Hill, 5th Edition, 2006. 2. C. J. Date, A. Kannan, and S. Swamynathan, "An Introduction to Database Systems," Pearson, 8th Edition, 2006.				

Course Title	Computer Organization and Design	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	The course aims to introduce various aspects of computer organization such as Instruction format, Instruction codes, Addressing Modes, processor design and hierarchical memory design, Input and Output Interface design using Programmed Controlled and Interrupt Control way				
Course Outcomes	Students will be able to interface and program various components such as Memory, I/O, etc. with the processor.				
Contents of the course	Introduction: function and structure of a computer, functional components of a computer, performance of a computer system. Instruction set architectures – CISC and RISC architectures.(5) Instructions: Language of the Computer, Operations of the Computer Hardware, Operands of the Computer Hardware, Representing Instructions in the Computer, Logical Operations Instructions for Making Decisions, addressing Modes, Parallelism & Instructions. (5) Arithmetic Design: – Carry look ahead adder, Wallace tree multiplier, Floating–point adder/subtractor, Division. (5) The Processor: Logic Design Conventions, Building a Datapath, A Simple Implementation Scheme (3) An Overview of Pipelining, Pipelined Data path and Control, Data Hazards: Forwarding versus Stalling, Control Hazards, Exceptions and Parallelism via Instructions. (7) Memory Hierarchy: Introduction, Memory Technologies (SRAM, DRAM), The Basics of Caches, Measuring and Improving Cache Performance, Dependable Memory, Virtual Machines, Virtual Memory, A Common Framework for Memory Hierarchy, Using a Finite-State Machine to Control a Simple Cache, Parallelism and Memory Hierarchies: Cache Coherence, Parallelism and Memory Hierarchy: Redundant Arrays of Inexpensive Disks and Implementing Cache Controllers. (9) Input/Output Unit**:** access of I/O devices, I/O ports, I/O control mechanisms – Program Controlled I/O. Interrupt controlled I/O and DMA controlled I/O; I/O interfaces – Serial port, parallel port, USB port, SCSI bus, PCI bus; I/O peripherals – Keyboard, display, secondary storage devices. (8)				
Textbook	1. Patterson and Hennessy, “Computer Organization and Design,” Morgan Kaufmann, 5^{tn} Edition, 2013. 2. C. Hamacher, Z. Vranesic, and S. Zaky, “Computer Organization,” Tata McGraw Hill, 5th Edition, 2002.				
References	1. J. P. Hayes, “Computer Architecture and Organization,” Tata McGraw Hill 1998. 2. M. J. Murdocca, V. P. Heuring, “Computer Architecture and Organization - An Integrated Approach,” John Wiley & Sons Inc., 2007. 3. A. S. Tanenbaum, “Structured Computer Organization,” Prentice Hall,5th Edition, 2006.				

Course Title	Object Oriented Algorithm Design and Analysis Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	The objective is to introduce object oriented programming (OOP) paradigm and implement algorithms using OOP concepts.				
Course Outcomes	Students would be capable of using OOP concepts effectively while implementing various algorithmic paradigms.				
Contents of the course	The laboratory component will require the student to write computer programs using a careful choice of data structures and algorithmic paradigms (in C++/Java language) from scratch, based on the concepts learnt in the theory course. **OOP concepts:** Object oriented programming - Encapsulation – Constructors – Destructors - Composition – Friend functions/classes – this pointer – Dynamic memory management Operator overloading Reusability – Inheritance – Base & derived classes – Protected members – Constructors –Destructors in derived classes – public/private/protected inheritance–Polymorphism Virtual functions - Templates – Function & Class templates – Streams – Stream input Output Stream format states – Manipulators – Exception handling – Re–throwing exceptions – specifications–and exception handling – Inheritance – STL Case studies involving Data structures and Algorithms using OOPs concepts.				
Textbook	1. P. J. Deitel and H. M. Deitel, "C++ : How To Program," Prentice Hall, 8th Edition, 2011.				
References	1. H. Schildt, "Teach Yourself C++," 3rd Edition, Tata McGraw Hill. 2. R. Lafore, "Object Oriented Programming in C++," 4th Edition, Sams Publishing.				

Course Title	Database Systems Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status	Core ■		Elective ☐
Course Objectives	This course introduces SQL programming. Database design preserving functional dependencies and loss-less decomposition properties would be addressed.				
Course Outcomes	Conceptual design using ER diagrams, programming using structured query language, and database design respecting third normal form shall be the outcomes of this course.				
Contents of the course	Introduction to SQL. Schema, table creation using SQL, Data definition and data manipulation using SQL. Implementation of set theoretic operations on databases. Views using SQL. Implementation of algorithms related to functional dependencies and loss-less decomposition. Indexing using B-trees and B+ trees(creation, insertion, deletion).				
Textbook	1. Loney Koch, Oracle – The complete reference, Tata McGraw Hill, 2002 2. R.Elmasri and S.B.Navathe, Fundamentals of Database Systems, Pearson, 4th Edn, 2007.				
References	1. A. Silberschatz, H. F. Korth, and S. Sudharsan, "Database System Concepts," Tata McGraw Hill, 5th Edition, 2006. 2. C. J. Date, A. Kannan, and S. Swamynathan, "An Introduction to Database Systems," Pearson, 8th Edition, 2006.				

Course Title	Computer Organization & Design Practices	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status	Core ■		Elective □
Course Objectives	Exposure to assembly language programming, instruction set design, and processor design for a given instruction set are given. Accessible macros, interrupt service routines, and simple device driven programs would also be introduced. Computer system design concepts are introduced.				
Course Outcomes	Students would be able to demonstrate programming proficiency using the various addressing modes and data transfer instructions of the target computer, and design microcomputer systems.				
Contents of the course	Exercises will mainly involve writing the assembly language programs. • Execution of assembly language programs. Single-step, break points, Accessing the contents of registers, accessing the contents of memory locations. – Implementation of higher level language assignment statements with arithmetic expressions and logical expressions. • Implementation of control transfer statements. Macros. • Software interrupts. • Operating system function calls. • Interrupt service routines. – Simple device drivers. – Assembly language programming in C language. I/O interfacing and programming. Computer System Design.				
Textbook	1. Patterson and Hennessy, "Computer Organization and Design," Morgan Kaufmann, 5th Edition, 2013.				
References	1. C. Hamacher, Z. Vranesic, and S. Zaky, "Computer Organization," Tata McGraw Hill, 2002.				

Course Title	Entrepreneurship and Management Functions	Course No	To be filled by the office		
Specialization	HMC	Structure (IPC)	3	0	3
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	Systems Thinking and Design	To take effect from			
Course Objectives	The objective of this course is to provide engineering students an exposure to the basic concepts of entrepreneurship and management, with a specific focus on the process of turning an idea into a commercially viable venture.				
Course Outcomes	At the end of the course, the students will learn how to Understand the market & competition Prepare a business case for the product/idea				
Contents of the course	Module 1: Introduction · Division of labor and creation of value · Evolution of organizations, industries and sectors, for profit and non-profit · Role of Entrepreneurs and Managers in value creation · Principles of Management - Planning, Organizing, Resourcing, Directing (4) Module 2: Strategy & Planning · Understanding industry dynamics & competition (Porter's Framework) · Understanding the industry value chain and firm positioning (6) Module 3: Organizing · Typical organizational functions (R&D, Marketing & Sales, HR, Operations) · Cybernetics of organizational functions (Stafford Beer's viable systems model) · Types of organization structures (product, functional, matrix, global) (6) Module 4: Resource Management · Financial management (Sources of funding, how to read a P&L, balance sheet) · Human resource management (Interviewing, compensation, motivation) · Global sourcing and supply chain management (8) Module 5: Management Information & Decision Making (4) Module 6: Legal and Regulatory environment (4)				
Textbook	1. Peter F Drucker, *The Practice of Management*, Harper Collins, 2006, ISBN: 978-0060878979 2. Hentry Mintzberg, *Managing*, Berret-Koehler Publishers, 2009, ISBN: 978-1605098746 3. Michael Porter, *On competition: Updated and Expanded Edition*, HBS, 2008, ISBN: 978-1422126967 4. Vasanta Desai, *Dynamics of Entrepreneurial Development and Management*, Himalaya Publishing House, ISBN:9788183184113.				
References	1. Walter Isaacson, *Steve Jobs*, 2011, ISBN:978-1451648539 2. Eric Ries, *The Lean Startup*, Portfolio Penguin, 2011, ISBN: 978-0307887894 3. Vineet Bajpai, Build from scratch, Jaico books, 2013, ISBN: 9788184952919.				

Course Title	Operating Systems	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	Computer Organization and Design	To take effect from			
Course Objectives	This first level course focuses on exposing students to the purpose, structure and functions of an operating system. Operating systems abstraction, mechanisms and their implementation support for concurrency (threads) and synchronization, resource management, scheduling strategies, etc. are explored.				
Course Outcomes	Students shall have a sound understanding of basic concepts relating to the design and implementation of an operating system. Specifics relating to scheduling, multithreading, synchronization, etc. shall help them understand the structure of the operating system (Linux), at the concept and the source code level.				
Contents of the course	Functionalities & Services of an Operating System – System Calls & Types - Process Concept – Process Control Block – Linux System calls for Process creation, Inter Process Communication using Shared memory / Message passing. (10) Concurrency – Multithreaded programming – benefits, challenges, models, Pthreads library in Linux – thread creation, cancellation, thread specific data, Thread pools, Signal handling , Scheduling – Premptive, Non preemptive algorithms FCFS, SJF, SRT, RR – Thread scheduling – contention scope, pthread support for scheduling. (11) Synchronization – Race condition – Critical Section Problem, Solution, Mutex Locks and Semaphores – Priority Inversion, Pthreads synchronization - Producer Consumer problem (multi threaded) example Deadlock characterization – Resource graph – Avoidance & Prevention – Safe state – Bankers algorithm – recovery schemes. (10) Memory management – logical v/s physical address space – Segmentation, Paging, Page table structures , Virtual memory, Page replacement strategies, File Systems – file operations, types, access methods, Directory structure, Mounting file systems. (11)				
Textbook	1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Concepts, John Wiley, 9th Edn, 2015.				
References	1. Andrew S Tanenbaum, Modern Operating Systems, Prentice Hall, 2007. 2. Stallings. W, Operating System: Internals and Design Principles, Prentice Hall, 2009. 3. Gary Nut, Operating Systems: A Modern Perspective, Addison Wesley, 2003.				

Course Title	Computer Networking	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	Computer Organization and Design	To take effect from			
Course Objectives	To introduce the basics of computer networking, error detection and correction techniques, and flow control techniques. Also an exposure to IP addressing and routing and its associated protocols would be given. A highlight of various application layer protocols and its relevance in modern networking world would be discussed.				
Course Outcomes	To be able to design a local area network and analyze the network using performance metrics. To appreciate the importance of subnetting, masking, and nuances involved in setting up a campus network.				
Contents of the course	Evolution of computer networks, creating a small network, Data transfer between nodes, encoding of bits in physical layer, NRZ, Manchester, Differential Manchester, Performance evaluation of a network: propagation delay, transmission delay, RTT, effective bandwidth. (10) Error detection techniques in Data link layer (LRC, CRC, Two dimensional parity check), Hamming Error correcting codes. Data transfer between nodes using stop and wait protocol, sliding window protocol (Go-back-n and selective reject), performance analysis of stop and wait and sliding window protocols. Flow control at data link layer. Introduction to layer-2 devices (switches, bridges) and addressing scheme at Layer-2 (MAC addresses). (10) Creating a small network using Ethernet (IEEE 802.3) Token Ring (IEEE 802.5), Performance evaluation of IEEE 802.3 and 802.5 networks. Introduction to Layer-3 devices, IP addresses, IPv4,IPv6, Error detection at layer-3 using Checksum. IP addressing schemes, subnetting, CIDR (12) Introduction to TCP/IP, IP routing, RIP, OSPF, Circuit and Packet switching, ICMP, Introduction to networking commands: Ping, Traceroute, IPconfig, UDP, congestion control and avoidance. (10) Introduction to DHCP, FTP, HTTP and other application layer protocols. (3)				
Textbook	1. Larry L.Peterson and Bruce S Davie, Computer Networks: A systems Approach, Morgan, 3rd Edn, 2003. 2. William Stallings, Data and Computer Communications, 6th Edn, Pearson, 2000.				
References	1. Andrew S. Tanenbaum, Computer Networks, 4th Edn, 2003.				

Course Title	VLSI System Design	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG and DD Computer Engineering	Status (Core / Elective)	Core		
Prerequisite	Computer Organization and Design	To take effect from			
Course Objectives	The goal of the course is to introduce architecture and design concepts underlying the modern complex VLSI circuits/systems and system-on-chip.				
Course Outcomes	The student would be able to design the digital subsystem using VLSI techniques and can estimate circuit/system performance, and design digital subsystems/system on chip.				
Contents of the course	MOS Transistors, CMOS Logic - Inverter, Logic Gates, Pass Transistors and Transmission Gates, Tristates, Multiplexers, Sequential Circuits. (3) CMOS Fabrication and Layout - Inverter Cross-section, Fabrication process, Layout Design Rules, Gate Layouts, Stick Diagrams. (4) Design Partitioning: Design Abstractions, Structured Design, Behavioral, Structural and Physical Domains. (3) Logic Design, Circuit Design, Physical Design, Design verification, Fabrication, Packaging and Testing. Technology related CAD Issues: Design Rule Checking (DRC), Circuit extraction. (4) Delay: Timing optimization, Transient response, RC Delay Model, Linear Delay Model, Logical Effort of Paths. Statistical timing analysis. (3) Power: Sources of Power Dissipation, Dynamic Power, Static Power, Energy-Delay Optimization, Low Power Architectures. (3) Robustness: Variability, Reliability, scaling, statistical Analysis of Variability, Variation-Tolerant design. (3) Datapath Subsystem, Array Subsystems, Special purpose Subsystems. (4) Design Methodology and Tools - Structured Design Strategies, Design Methods, Design Flows, Design Economics, Data sheets and Documentation. (4) Testing, Debugging and Verification: Testers, text fixtures, and Test Programs, Logic verification Principles, Silicon Debug Principles, Manufacturing Test Principles, Design for Testability. (4) CMOS chip design options: Full custom ASICs, Std. Cell based ASICs, Gate Array based ASICs, Programmable logic structures-PLA, PAL, PROM, FPGA. (7)				
Textbook	1. Weste & Eshraghian: Principles of CMOS VLSI design, Addison Wesley, 4th Edn, 2011.				
References	1. Samir Palnitkar; Verilog HDL - Guide to Digital design and synthesis, Pearson Education, 3rd Edn 2003. 2. Geiger R. L., Allen, P. E. and Strader, N. R., VLSI Design Techniques for Analog and Digital Circuits, McGraw-Hill, 1990. 3. Wolf W., Modern VLSI Design, Pearson Education, 1997.				

Course Title	Automata & Compiler Design	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG and DD Computer Engineering	Status (Core / Elective)	Core		
Prerequisite	----	To take effect from			
Course Objectives	The objective of this course is to train students to design various phases of compiler such as Lexical analyzer, syntax analyzer, semantic analyzer, intermediate code generator, code optimizer and code generator. Students are also exposed to design compiler construction tools such as Lexical Analyser generator and parser generator. Fundamentals of automata theory and applications of finite sate machine and pushdown automaton in compiler design are also taught in this course.				
Course Outcomes	At the end of the course, students will be able to design a programming language and compiler for the same. Students will also be able to write large programs.				
Contents of the course	Introduction to phases of compiler– DFA – NFA to DFA —regular expression and its application to give syntax of word -regular expression to NFA, Construction of NFA without epsilon moves from regular expression- Regular grammar-regular grammar to automata, and automata to regular grammar-Minimization of automata- Pumping lemma application-Lexical analyzer Design (12) Context free grammar & its application to give syntax of program statement – Types of parsing – Top down & bottom up–Recursive descent– Predictive–Shift reduce–Operator precedence–SLR (10) Semantic analysis - Intermediate code generation: Declaration – Assignment statements – Boolean expressions – looping and branching statements (7) Back patching and procedure calls code generator design issues – Runtime storage management – Code Optimization: Basic blocks – Flow graphs – Next use information – Code generator case study – Directed acyclic graph representation of basic blocks – Peephole optimization technique Introduction to code optimization (10) Storage optimization & allocation strategies).Assembly Code Generation: from syntax tree and Directed acyclic graph - from three address code. (5)				
Textbook	1. Alfred Aho, Ravi Sethi and Jeffrey D Ullman, Compilers Principles, Techniques and Tools, Pearson Education, 2003.				
References	1. Levine J.R, Mason T, Brown D, Lex & Yacc, OReilly Associates, 1992. 2. Allen I. Holub, Compiler Design in C, Prentice Hall, 2003. 3. Kamala Krithivasan and R Rama, Introduction to Formal Languages, Automata Theory and Computation, Pearson Education, 2009.				

Course Title	Computer Networking Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	----	To take effect from			
Course Objectives	To understand basic networking commands, MAC/IP addressing, file transfer between two systems, etc. Simulation of error control techniques and flow control techniques using well-known protocols would be addressed as part of this course.				
Course Outcomes	Learner would be comfortable in design, testing, and trouble shooting aspects associated with local area networking. Learner would also appreciate the importance of error detecting codes and flow control techniques.				
Contents of the course	Connecting two nodes using Ethernet cable and study the performance evaluation parameters such as delay, effective bandwidth - Basic Networking commands – Ping, IPConfig, Traceroute, NSlookup - Introduction to Socket Programming. File transfer using TCP. Echo, Chat between two or more clients using socket programming - Simulation of Stop and Wait Protocol - Simulation of Stop and Wait protocol with NACK, Modelling of ACK, NACK drops, etc., - Modelling and simulation of Sliding window protocol - Sliding window protocol with ACK/NACK drops, frame drops etc., - Performance evaluation through simulation of IEEE 802.3/802.5 networks - Implementation of OSPF. Introduction to NS2/OPNET simulator, Case studies.				
Textbook	1. Larry L.Peterson and Bruce S Davie, Computer Networks: A systems Approach, 3rd Edn, Morgan, 2003. 2. William Stallings, Data and Computer Communications, 6th Edn, Pearson, 2000.				
References	1. Andrew S. Tanenbaum, Computer Networks, 4th Edn, 2003				

Course Title	Operating Systems Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	----	To take effect from			
Course Objectives	The course aims to equip the student with implementation level constructs / support in Linux for various concepts such as process management, concurrency, scheduling, deadlock avoidance, etc.				
Course Outcomes	The student shall be able to relate the operating system concepts listed above to the Linux operating system and support for the same available through various system calls.				
Contents of the course	Linux System Calls for process creation, management – Applications such as command prompt simulator using fork – Interprocess Communication using Shared Memory and Pipes – Producer Consumer – Applications using pipes / shm – Concurrency – Multithreading –Pthread support – Applications such as merge sort, min-max-average, etc. in a multi threaded fashion – Scheduling – pthread interfaces setschedpolicy – getschedpolicy based applications – Synchronization – threaded solution for classical problems like dining philosophers, readers writers, etc. using mutex locks and semaphores - Deadlock detection / avoidance algorithms.				
Textbook	1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Concepts, John Wiley, 9th Edn, 2015.				
References	1. Robert Love, Linux Systems Programming, O Reilly Media, 2nd Edition 2. D Butlar, J Farrell, B Nichols, Pthreads Programming, O Reilly Media, 1996				

Course Title	VLSI System Design Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	B.Tech Computer Engineering	Status (Core / Elective)	Core		
Prerequisite	----	To take effect from			
Course Objectives	The lab course is intended to give exposure to the design of different functional components of a computer system using Verilog and development kits, and use VLSI Design flow to generate RTL to GDS-II format.				
Course Outcomes	The student would be able to model and design any digital system at circuit/layout level. They will also be able to design an ASIC using RTL codes.				
Contents of the course	Design at circuit level and layout level for Datapath Subsystem Design: Addition/Sbtraction, one/zero Detectors, comparators, counters, shifters, multiplication, SRAM, DRAM, ROM, Flash, CAM – Delay, Area and Power Analysis using EDA Tools. Simple Digital System design using Verilog HDL – VLSI Design flow from RTL to GDS-II using EDA Tools.				
Textbook	1. Samir Palnitkar; Verilog HDL - Guide to Digital design and synthesis, 3rd Edn, Pearson Education, 2003.				
References	1. Weste & Eshraghian: Principles of CMOS VLSI design, 4th Edn, Addison Wesley 2011.				

Course Title	Design for Quality and Reliability	Course No	To be filled by the office		
Specialization	Design	Structure (IPC)	3	0	3
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	Measurements and Data Analysis Lab (Probability and Statistics)	To take effect from			
Course Objectives	The objectives of the course are to help engineering students understand: (1) To understand concepts of quality & reliability (2) To evaluate the overall reliability of a system from component reliability.				
Course Outcomes	Attending the course would enable the student to: 1. Model repairable and non-repairable systems and calculate failure rate, repair rate, reliability and availability 2. Use various probability density distributions significant to reliability calculations 3. Fit a given failure data set of a product into a Weibull distribution and estimate the reliability parameters.				
Contents of the course	Module 1: Concepts of Product Quality • Quality Function Deployment / House of Quality • Six Sigma (6) Module 2: Concepts of Reliability · Basic concepts of repairable and non-repairable systems · Reliability, Availability and Maintainability (6) Module 3: Failure data analysis · Fitting discrete and continuous distributions to failure data sets, Weibull analysis, estimation of important reliability parameters (8) Module 4: Calculation of System Reliability from Component reliabilities · Markov modeling of repairable and non-repairable systems · Reliability Logic Diagrams · Fault-tree analysis (8) Module 5: Preventive and Predictive maintenance Failure Modes and Effects Analysis. (4)				
Textbook	1. Louis Cohen, Joseph P. Ficalora, *Quality Function Deployment and Six Sigma: A QFD Handbook*, Prentice Hall, Second Edition, 2009, ISBN: 9780137035441 2. VNA Naikan, *Reliability Engineering and Life Testing*, PHI Learning, 2010, ISBN: 978-8120335936 3. Singiresu S Rao, *Reliability Engineering*, Pearson Education, 2014, ISBN: 978-0136015727				
References	1. Patrick O Connor, *Practical Reliability Engineering*, John Wiley, Student ed., 2009, ISBN:9780470979815 2. 2. B.L. Hansen & P.M. Ghare, *Quality Control and Applications*, Prentice-Hall, 1997, ISBN: 9780137452255				

Course Title	Computer Architecture	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	3	0	3
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	Computer Organization and Design	To take effect from			
Course Objectives	The course aims to expose students to the concepts involved in the design of computer systems covering aspects such as instruction sets, pipelining, caches, physical memory, virtual memory, superscalar and out-of-order instruction execution, vector processor and multi-threading				
Course Outcomes	Students will have the ability to design a computer system addressing issues related to Instruction level, data level and thread level parallelisms.				
Contents of the course	Fundamentals of Quantitative, Design and Analysis Computers. (3) Memory Hierarchy Design: Optimizations of Cache Performance, Memory Technology and Optimizations, Virtual Memory and Virtual Machines. (7) Instruction-Level Parallelism and Its Exploitation: ILP Concepts and Challenges, Overcoming Data Hazards with Static and Dynamic Scheduling, Reducing Branch Costs with Advanced Branch Prediction, Static and Dynamic Scheduling, Hardware-Based Speculation, Studies of the Limitations of ILP. (12) Multi-Threading: Exploiting Thread-Level Parallelism to Improve Uniprocessor Throughput (5) Data-Level Parallelism in Vector, SIMD, and GPU Architectures: Vector Architecture, Detecting and Enhancing Loop-Level Parallelism. (5) Thread-Level Parallelism: Centralized Shared-Memory Architectures, Performance of Symmetric Shared-Memory Multiprocessors, Distributed Shared-Memory and Directory-Based Coherence, Synchronization, Models of Memory Consistency, Multicore Processors and Their Performance. (5) Warehouse-Scale Computers to Exploit Request-Level and Data-Level Parallelism: Programming Models and Workloads for Warehouse-Scale Computers, Computer Architecture of Warehouse-Scale Computers, Physical Infrastructure and Costs of Warehouse-Scale Computers, Cloud Computing: The Return of Utility Computing. (5)				
Textbook	1. John L. Hennessy and David A. Patterson, Computer Architecture, Fifth Edition: A Quantitative Approach, The Morgan Kaufmann, 5th Edn, 2012.				
References	1. John P. Shen and Mikko H. Lipasti, Modern Processor Design: Fundamentals of Superscalar Processors, Waveland Press, 1st Edn, 2005, 2. D.M. Harris and S.L. Harris. Digital Design and Computer Architecture, 2nd Edn. Morgan Kaufmann, 2012. 3. M. Johnson. Superscalar Microprocessor Design, Prentice Hall, 1991.				

Course Title	Computer Architecture Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	----	To take effect from			
Course Objectives	The course aims to be a hands on to the supplementing theory course with exposure to issues related to computer systems design on instruction level ad thread level parallelism.				
Course Outcomes	Students will have the ability to design multi core systems for a given specification using electronic design automation tools.				
Contents of the course	Incrementally design, implement, test, and evaluate a complete multi-core system with an integrated collection of processors, memories. A processor includes – pipeline arithmetic operation, register file, branch predictors, hardware based instruction scheduling and commit, cache design, MESI.				
Textbook	1. John L. Hennessy and David A. Patterson, Computer Architecture, Fifth Edition: A Quantitative Approach, The Morgan Kaufmann, 5th Edn, 2012. 2. Samir Palnitkar, Verilog HDL: A Guide to Digital De sign and Synthesis, Second Edition, Prentice Hall, 2003.				
References	1. John P. Shen and Mikko H. Lipasti, Modern Processor Design: Fundamentals of Superscalar Processors, Waveland Press, 1st Edn, 2005, 2. D.M. Harris and S.L. Harris. Digital Design and Computer Architecture, 2nd Edn Morgan Kaufmann, 2012. 3. M. Johnson. Superscalar Microprocessor Design, Prentice Hall, 1991.				

Course Title	Artificial Intelligence	Course No *(will be assigned)*				
Specialization	Computer Science & Engineering	Structure (LTPC)	3	0	0	3
Offered for	UG	Status	Core ■		Elective ☐	
Faculty		Type	New ☐		Modification ☐	
Pre-requisite		To take effect from				
Submission date		Date of approval by Senate				
Objectives	The objective of this course is to train the students to understand different types of AI agents, various AI search algorithms, fundamentals of knowledge representation, building of simple knowledge-based systems and to apply knowledge representation, reasoning, and machine learning techniques to solve real-world problems.					
Contents of the course	Introduction: AI problems,Agents and Environments, Structure of Agents, Problem Solving Agents (4 hrs) Basic Search Strategies: Problem Spaces, Uninformed Search (Breadth-First, Depth-First Search, Depth-first with Iterative Deepening), Heuristic Search (Hill Climbing, Generic Best-First, A*), Constraint Satisfaction (Backtracking, Local Search) (7 hrs) Advanced Search: Constructing Search Trees, Stochastic Search, A* Search Implementation, Minimax Search, Alpha-Beta Pruning (7 hrs) Basic Knowledge Representation and Reasoning: Propositional Logic, First-Order Logic, Forward Chaining and Backward Chaining, Introduction to Probabilistic Reasoning, Bayes Theorem (7 hrs) Advanced Knowledge Representation and Reasoning: Knowledge Representation Issues, Non-monotonic Reasoning, Other Knowledge Representation Schemes (5 hrs). Reasoning Under Uncertainty: Basic probability, Acting Under Uncertainty, Bayes' Rule, Representing Knowledge in an Uncertain Domain, Bayesian Networks (6 hrs) Basic Machine Learning: Forms of Learning, Decision Trees, Nearest Neighbor Algorithm, Statistical-Based Learning such as Naïve Bayesian Classifier. (8 hrs)					
Textbook	Russell, S. and Norvig, P, *Artificial Intelligence: A Modern Approach*, Third Edition, Prentice-Hall, 2010					
References	1. Artificial Intelligence, Elaine Rich, Kevin Knight, Shivasankar B. Nair, The McGraw Hill publications, Third Edition, 2009 2. George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education, 6th ed., 2009. 3. Artificial Intelligence a new synthesis: Nils J. Nilson, Morgan Kaufmann Publishers, 1998					

Course Title	Artificial Intelligence Practice	Course No *(will be assigned)*				
Specialization	Computer Science & Engineering	Structure (LTPC)	0	0	3	2
Offered for	UG	Status	Core ☑		Elective ☐	
Faculty		Type	New ☐		Modification ☐	
Pre-requisite		To take effect from				
Submission date		Date of approval by Senate				
Objectives	This course helps the students to get exposed to the use of AI-techniques such as searching, constraint satisfaction, knowledge representation and machine learning for solving classical AI-problems. With this practical exposure, the students will be able to build systems for solving real-world problems.					
Contents of the course	Solving travelling salesman problem using BFS, DFS and A* algorithms – Implementation of tic-tac-toe game problem using minimax algorithm with alpha beta pruning- Use Backtracking technique for solving the 8 queen problem- Develop the program for solving the 8 Puzzle problem using Hill Climbing Technique – Solve a simple time schedule problem using Constraint Satisfaction Procedure- Develop a simple question answering system by representing knowledge about a particular event- Develop disease identification system using Naïve Bayesian Classifier technique- Develop a decision tree classification system using C4.5 algorithm and extend your implementation to Random Forests and do the performance analysis - Implement K-nearest-neighbours (K-NN) algorithm for both classification and regression.					
Textbooks	1. Artificial Intelligence: Problems and their solutions, Danny Kopec, Shweta Shetty and Christopher Pileggi, Mercury Publishers, 2014. 2. Machine Learning for Big Data: Hands-On for Developers and Technical Professionals, Jason Bell, Wiley Publications, 2014.					
References	1. Russell, S. and Norvig, P, *Artificial Intelligence: A Modern Approach*, Third Edition, Prentice-Hall, 2010 2. Artificial Intelligence, Elaine Rich, Kevin Knight, Shivasankar B. Nair, The McGraw Hill publications, Third Edition, 2009					

Course Title	Embedded Systems Practice	Course No	To be filled by the office		
Specialization	Computer Engineering	Structure (IPC)	0	3	2
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	----	To take effect from			
Course Objectives	In this course fundamental practices in the context of embedded systems will be covered. Hands-on experiments will be performed involving TI ARM Cortex-M microcontroller LaunchPad IDE (and booster packs), rapid prototyping of embedded systems using open source microcontrollers (Arduino, Raspberry Pi, BeagleBone Black), wireless networked embedded systems using Arduino shields, and Internet of Things concepts such as smart automation.				
Course Outcomes	At the end of the course, a student will be able to, 1. Understand how embedded systems interfaces operate (GPIO, interrupts, ADC/DAC, etc.) using the ARM Cortex LaunchPad IDE and booster packs 2. Perform experiments in sound, video (gaming) and mobile robots, with LCD displays, stepper and DC motors and RC servos 3. Rapid prototype embedded systems using open source microcontrollers (such as Arduino, Raspberry Pi, BeagleBone Black, and Intel Edison/Galileo). 4. Build wireless networked embedded systems using Arduino shields and modules (e.g., GPS, GSM/GPRS, Bluetooth, RFID, and ZigBee). 5. Conduct experiments in Internet of Things (e.g., using Arduino Yun, Intel and Microsoft Developer Kits)				
Contents of the course	Experiments in GPIO, serial interfacing, interrupts, data acquisition with ADC, sound and video, DAC Experiments in control of RC servos, stepper motors, DC motors, and design of video games and mobile robots Data acquisition and real-time control with Arduino, Raspberry Pi, and BeagleBone Black microcontrollers, shields, and add-on boards Experiments in wireless networked systems, using shields and modules, for GPS, GSM/GPRS, ZibBee, Bluetooth, and RFID Experiments in IOT for smart automation, with Intel and Microsoft development kits				
Textbook	1. IIITDM Kancheepuram –Embedded Systems Practice Manual.				
References	1. Jonathan Valvano and Ramesh Yerraballi, 2014, “Embedded Systems – Shape the World” (ebook). 2. T. Igoe, 2007, “Making things talk”, O’Reilly Press.				

Course Title	Product Design Practice	Course No	To be filled by the office		
Specialization	Design	Structure (IPC)	0	2	2
Offered for	UG	Status (Core / Elective)	Core		
Prerequisite	Design Realization, Product Realization	To take effect from			
Course Objectives	Students will develop cross-discipline products and prototype them using product realization tools in a multi- disciplinary team setting.				
Course Outcomes	By the end of the course, the students would be able to • Develop cross disciplinary idea • conceive, design and prototype an innovative idea • work in cross-functional groups and to apply the concepts learnt in theory to a practical problem • manage group projects, maintain timeliness and follow method oriented approach to problem solving				
Contents of the course	This course is an inter-disciplinary team-based product design and prototyping course. The concept of the course is to provide hands-on learning experience in interdisciplinary fields of engineering and exposure to the context of a "real" product design problems. In this course students will design a product by following the systematic product design process. A team consist of students from different discipline will choose their own innovative product and while designing, students will consider many issues like market opportunities, formal requirements and constraints, the environment in which the product will be used, product look and feel; technical legitimacy, and manufacturing considerations for the products. During the course, students will learn and put in to practice team working, project management and product realization practices commonly found in product developers in industry. Throughout the semester, the student teams have several opportunities to present their progress to their fellow students and faculty.				
Textbooks	1. Carl Liu, Innovative Product Design Practice, Kindle Edition, ASIN: B00B29V9RQ 2. Bjarki Hall grimsson, Prototyping and Model making for Product Design, 2012, Laurance King Publishing Limited, ISBN-13: 978-1856698764.				

Course Title	Systems Thinking for Design	Course No	To be filled by the office		
Specialization	Design	Structure (IPC)	3	0	3
Offered for	UG	Status	Core ■	Elective ☐	
Pre-requisite	Matrix Methods	To take effect from			
Course Objectives	Design for effectiveness – Level 1				
Course Outcomes	This course will help students understand • The importance of modeling systems to realize effective designs • Abstraction of key elements from problem situations • Use of specific techniques to model problems in a holistic manner				
Contents of the course	• Real-world problems & the need for inter-disciplinary approaches [2] • Basic concepts of systems thinking (parts, relations, patterns) [10] • Technique #1: Rich Pictures • Technique #2: Mapping Stakeholder, Needs, Alterables, Constraints [10] • Technique #3: Structural Modeling (Hierarchical decomposition) [10] • Technique #4: Influence Diagrams (Self-regulating systems) [10]				
Textbook	1. Hitchins, Derek K. (2007) Systems Engineering: A 21st Century Systems Methodology, John Wiley, ISBN: 978-0-470-05856-5. 2. Wilson, Brian (1991) Systems: Concepts, Methodologies and Applications. 2nd Edition, Wiley. ISBN: 0471927163. 3. Hutchinson, William; Systems Thinking and Associated Methodologies, Praxis Education. ISBN: 0 646 34145 6.				
References	1. Gerald Wienberg (2001), An introduction to general systems thinking, Dorset House Publishing. 2. Sage, A.P. (1977); Methodology for Large Scale Systems, McGraw Hill, New York.				

Course Title	Sustainable Design	Course No	To be filled by the office		
Specialization	Design	Structure (IPC)	3	0	3
Offered for	B. Tech. All streams	Status (Core / Elective)	Core		
Prerequisite	Earth Environment and Design	To take effect from			
Course Objectives	The objective of this course is to prepare engineering students to address product design from a broader, holistic perspective, integrating environmental responsibility into the core of the design process.				
Course Outcomes	Upon completion of the course students are expected to demonstrate knowledge, skill and abilities in the following areas:(a) To equip the design student with specific environmentally-responsive tools, principles and methodologies in preparation for professional application. Management(b) To use a variety of techniques to communicate effectively (sketches, illustrations, photographs, persuasive writing, presentation skills, etc.).				
Contents of the course	Module 1: Introduction, Definitions, History • The environmental origins of sustainability• theory of sustainability. (6) Module 2: Environmentally-responsive design methodologies • Industrial ecology• dematerialization • Design for reuse / modularity• design for recycling • Remanufacturing: issues/problems, current and future developments (12) Module 3: Alternative resources • Alternative energy • Alternative materials • Sustainable packaging. (14) Module 4: life-cycle assessment methods. (8)				
Textbook	1. Victor Papanek, *The Green Imperative*, 1995, ISBN: 978-0500278468 2. William McDonough and Michael Braungart, *Cradle to Cradle*, 2009, ISBN: 978-0099535478 3. Stuart Walker (2006), *Sustainable by Design: Explorations in Theory and Practice*, ISBN: 978-1844073535 4. Charter, Tischner, *Sustainable Solutions*, Green Leaf Publishing, 2001, ISBN: 978-1874719366.				
References	1. Cattanach, Holdreith, Reinke, Sibik, *The Handbook of Environmentally Conscious Manufacturing*, 1995, ISBN: 9780786301478 2. Sim van der Ryn, Stuart Cowan, *Ecological Design*, 1995, ISBN: 978-1559633895 3. Paul Hawken, *The Ecology of Commerce*, 2010, Collins Business Essentials, ISBN: 978-0061252792 4. Nattrass & Altomare, *The Natural Step for Business*, New Society Publishers, 1999, ISBN: 978-0865713840.				