B.Tech (Computer Engineering)

Semester	1	2	3	4	5	6	7	8	Discipline /NSS	Credits	CH	IC	PC	PE	OE	Total
I & II									1	**45**		45				**45**
III	HS-200 Social Science and Economics 2	CP-221 Logic System Design (3-0-0) 3	CP-223 Data Structures (3-0-0) 3	CP-225 Probability and Statistics (3-0-0) 3	CP-227 Abstract Algebra (3-0-0) 3	CP-251 LSD Lab (0-0-3) 2	CP-253 Data Structure Lab (0-0-3) 2	Open Elective I 4		**22**		2	16		4	**22**
IV	CP-222 DBMS (3-0-0) 3	CP-224 Microprocessor and Microcontrollers (3-0-0) 3	CP-226 Computer Architecture (3-0-0) 3	CP-228 Theory of Computation (3-1-0) 4	CP-250 DBMS Lab (0-0-3) 2	CP-252 Assembly Language Programming Lab (0-0-3) 2	Open Elective II 4	Creative Arts/ Sports/ NSS/ Hindi 1	Discipline 1	**23**		2	17		4	**23**
V	CP-321 Operating System (3-0-0) 3	CP-323 Compiler Design (3-0-0) 3	CP-325 Computer Network (3-0-0) 3	CP-351 OS Lab (0-0-3) 2	CP-353 System Programming Lab (0-0-3) 2	CP-355 Network Programming Lab (0-0-3) 2	Program Elective I 4	Open Elective III 4		**23**			15	4	4	**23**
VI	CP-322 Software Engg. (3-0-0) 3	CP-324 Design and Analysis of Algorithms (3-0-0) 3	CP-350 Free and Open Source Lab (0-1-3) 3	CP-352 Algorithms Lab (0-0-3) 2	CP-354 Seminar (0-3-0) 3	Program Elective II 4	Open Elective IV 4		Discipline 1	**23**		1	14	4	4	**23**
VII	CP-451 Industrial Training (0-3-0) 3	Program Elective III 4	Program Elective IV 4	Program Elective V 4	CP-453 Design Elective 6	Open Elective V 4				**25**			3	18	4	**25**
VIII	CP-452 Project 10	Program Elective VI 4	Program Elective VII 4						Discipline 1	**19**		1	10	8	0	**19**
								Total Credits		180		**51**	**75**	**34**	**20**	**180**

	Program Elective I
CP-371	Object Oriented Programming
CP-373	Logical and Functional Programming
CP-375	Programming in Java
CP-377	Concurrent Programming

	Program Elective II
CP-372	Digital Signal Processing
CP-374	Wireless Communications
CP-376	VHDL
CP-378	Neural Networks

	Program Elective III		**Program Elective IV**		**Program Elective V**
CP-471	Computer Graphics	CP-481	Advances in Compiler Design	CP-491	Cryptography
CP-473	Speech Recognition	CP-483	Natural Language Processing	CP-493	Embedded Systems
CP-475	Mobile Computing	CP-485	Programming Paradigms	CP-495	Data Mining
CP-477	Software Project Management	CP-487	Wireless & Ad-hoc Networks	CP-497	VLSI Algorithms
CP-479	Data Compression	CP-489	Real Time Systems		

	Program Elective VI		**Program Elective VII**
CP-470	Network Security	CP-480	Evolving Architectures
CP-472	Biometrics	CP-482	Topics in Advanced Computing
CP-474	High Level Synthesis of Digital Syst	CP-484	Software Testing and Validation
CP-476	Selected Topics in Cryptography	CP-486	Machine Learning
CP-478	Digital Image Processing	CP-488	Modelling and Simulation

B.Tech (Computer Engineering) Semester III

HF-201 Humanities and Social Sciences (X-X-X) 2

Please refer to Department of Humanities and Social Sciences.

CP-221 Logic System Design (3-0-0) 3

Introduction to Boolean algebra: Binary connectives, Evaluation of truth functions, Truth – function calculus as Boolean Algebra, Duality, Fundamental theorems of Boolean Algebra and simplification of Boolean expressions.

Realisation of Logic Circuits: Standard forms of Boolean Functions, Minterm and Maxterm, designation of functions. Simplification of functions on Karnaugh maps, incompletely specified functions.

Combinational circuits: Adder, subtract, encoder, decoder, multiplexer, demultiplexer, parity checker and generator.Cubical representation of Boolean functions and determination of prime implicants. Selection of an optimal set of prime implicants, multiple output circuits and map minimization of multiple output circuits. Tabular determination of multiple output prime implicants.

Latches, Flip Flops : JK, SR, D Type and T type Flip Flops and their working principals.

Counters and shift registers: Ripple, decade, up-down counters, Mod-n counters and series, parallel registers. General characteristic of sequential circuits, clock, pulse and level mode sequential circuits. Analysis and design of sequential circuit. Synthesis of state diagrams, finite memory circuits, equivalence relations, equivalent states and circuits, determination of classes of in distinguishable states and simplification by implicant tables. Mealy and Moore machines, state assignment and memory element input equation, Partitioning and state assignment. General pulse-mode circuits, clock input counters, extended state tables.

Asynchronous Mode Circuits: Analysis of a fundamental mode circuits, Synthesis of flow tables, minimization, transition tables, excitation maps and output maps, Cycles and Races, Race free assignments, Hazards in sequential circuits.

Introduction to A/D and D/A converters.

Text/References:

1. Digital Systems and Hardware and Firmware Algorithms: M.Ercegovac and T. Lang, Pearson.
2. Morris-Mano : Logic System and Design, McGraw Hill
3. Hill & Peterson: Switching Theory and Logic Design, John Wiley
4. J.F.Wakerly: Digital Design, Principle and Practices, Pearson.
5. Malvino leech: Digital Electronics

CP-223 Data Structures (3-0-0) 3

Introduction to Data structures.

Arrays: Representation – row-major, column-major, sparse matrix –implementation, addition, multiplication; polynomial – Representation, addition, evaluation and multiplication.

Strings: Representation, operations, string matching - Brute force or naïve, Robin-Karp, Knuth-Morris-Pratt.

Linked List: Static and dynamic implementation, single, double, circular, multiple linked list.

Stack: Static and dynamic implementation, expression evaluation, prefix (polish), infix, postfix (inverse polish) expressions, application, multiple stacks, recursion.

Queues: Static and dynamic implementation, applications, circular queue, multiple queue.

Tree: Binary tree, binary search tree, static and dynamic implementation, tree operations - insertion,

deletion and search, tree traversal, Binary heaps. Introduction to AVL trees and B trees.
Sorting: Insertion sort, selection sort, Bubble sort, quick sort, merge sort, heap-sort, radix sort (bucket sort).
Searching: Linear and binary search, hashing.
Graph: Representation of graphs, BFS, DFS, topological sort.

Text/References:

1. Aho A.V., J.E. Hop croft, J.D. Ullman, Data Structures and algorithms, Addison Wesley.
2. Kruse R.L., Data Structure and Program Design, PHI.
3. Horowitz and Sahni: Data Structure in C++ , Glagotia
4. Ellis Horowitz, Sartaj Sahni, Fundamentals of Data Structures
5. Aaron M. Tenenbaum, Y. Langsam, Moshe J. Augenstein, Data Structures Using C
6. Niklaus Wirth, Algorithms + Data Structures = Programs (Prentice-Hall Series in Automatic Computation)
7. Sartaj Sahni, Data Structures, Algorithms, and Applications in C++
8. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++ (2nd Edition)

CP-225 Probability and Statistics (3-0-0) 3

Probability Theorem: Properties of probability, Conditional probability, Independence, Bayes theorem
Discrete Distributions: Probability distribution functions and cumulative distribution functions
Mean and variance; moment-generating functions, Marginal and conditional probability distributions, Some specific discrete distributions
Continuous Distributions: Probability density functions and cumulative distribution functions, Mean and variance; moment generating functions, Marginal and conditional probability distributions, Some specific continuous distributions
Functions of Random Variables: Distribution function technique, Transformation technique, Moment-generating function techniques

Text/References:

1. DeGroot, Morris H., and Mark J. Schervish. Probability and Statistics. 3rd ed. Boston, MA: Addison-Wesley, 2002. ISBN: 0201524880.
2. Feller, William. An Introduction to Probability Theory and Its Applications. 3rd ed., rev. printing. New York, NY: Wiley, 1968. ISBN: 0471257087.
3. Freund, W.J., Mathematical Statistics, 5th Ed., Prentice-Hall, Inc., Englewood Cliffs, N.J., 1994.
4. Hoel, P.G., Mathematical Statistics, 5th Ed., John Wiley & Sons, Inc., NewYork, 1984.
5. Hogg, R.V., & Craig, A.T., Introduction to Mathematical Statistics, 5th Ed.,Prentice-Hall, Inc., Englewood Cliffs, N.J., 1995.
6. Mood, A.M., Graybill, F.A., Boes, D.C., Introduction to the Theory of Statistics, 3rd Ed. McGraw Hill, Inc., New York, 1974.

CP-227 Abstract Algebra (3-0-0) 3

Number Systems: Natural numbers. Counting. Cardinality of finite sets. Laws, Mathematical induction. Prime numbers. Fundamental theorem of arithmetic. Well-ordering principle. Number bases. Modulo arithmetic. Greatest Common Divisor, Euler's extended algorithm, Chinese Remainder Theorem, Primality testing, Integers. Laws of arithmetic. Integer powers and logarithms. Recurrence relations. Number sieves.
Group Theory: Groups, Semi groups and Monoids, Cyclic semi graphs and sub monoids, Subgroups and cosets, Congruence relations on Semi groups, Factor groups and homomorphisms, Morphisms Normal sub groups. Structure of cyclic groups, Permutation groups, dihedral groups, Sylow theorems, abelian groups; solvable groups, Nilpotent groups; groups of small order, elementary applications in coding theory.
Rings: Rings, Subrings, Morphism of rings, ideal and quotient rings, Euclidean domains, Commutative rings; integral domains, noncommutative examples, Structure of Noncommutative Rings, Ideal Theory

of Commutative Rings

Field Theory: Integral domains and Fields, polynomial representation of binary number, Galois fields, primitive roots, discrete logarithms, split search algorithm.

Modules: Sums and products; chain conditions, Composition series; tensor products.

Text/ References:

1. John Fraleigh. *First Course in Abstract Algebra*, Pearson Education.
2. Michael Artin. *Algebra*, Pearson Education.
3. John A. Beachy and William D. Blair. *Abstract Algebra*, Second Edition, Waveland Press.
4. John A. Beachy. Abstract Algebra II, Cambridge University Press, London Mathematical Society Student Texts #47, 1999.

CP-251 LSD Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with CP-221.

1. Truth table verification – NAND gate, NOR gate, OR gate, AND gate, NOT gate.
2. Verifying if NAND gate is a universal gate.
3. Constructing XOR gate using NOR gate only.
4. Realizing given truth table using SOP form.
5. Realizing given truth table using POS form.
6. Design of combinational circuits – half adder, full adder, multiplier.
7. Design of binary-gray encoder.
8. Design of parity generator and detector.
9. Design of one bit error detecting and correcting circuit.
10. Design of flip flops – RS, JK, D and T flip flops.
11. Design of sequential circuits – counters.

Text/References:

1. Digital Systems and Hardware and Firmware Algorithms: M.Ercegovac and T. Lang, Pearson.
2. Morris-Mano : Logic System and Design, Mc Graw Hill
3. Hill & Peterson: Switching Theory and Logic Design, John Wiley
4. J.F.Wakerly: Digital Design, Principle and Practices, Pearson.
5. Malvino leech: Digital Electronics

CP-253 Data Structure Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with CP-223.

Programs in C or C++ for following:

1. Sorting programs: Bubble sort, Merge sort, Insertion sort, Selection sort, and Quick sort.
2. Searching programs: Linear Search, Binary Search.
3. Array implementation of Stack, Queue, Circular Queue, Linked List.
4. Implementation of Stack, Queue, Circular Queue, dynamic memory allocation.
5. Infix to postfix (prefix) conversion.
6. Program for expression evaluation.
7. Implementation of Binary tree. Program for Tree Traversals (preorder, inorder, postorder).
8. Program for graph traversal (BFS, DFS).
9. Program for minimum cost spanning tree, shortest path.

Text/References:

1. Aho A.V., J.E. Hop croft, J.D. Ullman, Data Structures and algorithms, Addison Wesley.
2. Kruse R.L., Data Structure and Program Design, PHI.
3. Horowitz and Sahni: Data Structure in C++ , Glagotia
4. Ellis Horowitz, Sartaj Sahni, Fundamentals of Data Structures

5. Aaron M. Tenenbaum, Y. Langsam, Moshe J. Augenstein, Data Structures Using C
6. Niklaus Wirth, Algorithms + Data Structures = Programs (Prentice-Hall Series in Automatic Computation)
7. Sartaj Sahni, Data Structures, Algorithms, and Applications in C++
8. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++ (2nd Edition)

Open Elective I (X-X-X) 3/4

Please refer to concerned Department.

B.Tech (Computer Engineering) Semester IV

CP-222 Data Base Management System (3-0-0) 3

Need, purpose and goal of DBMS, Three tier architecture, ER Diagram, data models- Relational, Network, Hierarchical and Object Oriented.
Data Base Design: Conceptual data base design, Theory of Normalization
Primitive and Composite data types, concept of physical and logical databases, data abstraction and data independence, data aggregation, Relational Calculus.
SQL : DDL and DML, Relational Algebra.
Application Development using SQL : Host Language interface, embedded SQL programming, Stored procedures and triggers and views, Constraints assertions.
Internal of RDBMS : Physical data organisation in sequential, indexed random and hashed files. Inverted and multilist structures, B trees, B+ trees, Query Optimisation, Join Algorithm, Statistics and Cost Base optimisation.
Transaction Processing, concurrency control, and recovery management. Transaction model properties and state serialisability . Lock base protocols, two phase locking.

Text/References:

1. H.f. Korth and Silberschatz: *Database Systems Concepts*, McGraw Hill
2. Almasri and S.B. Navathe: *Fundamentals of Database Systems*,
3. C.J. Date: *Data Base Design*, Addison Wesley
4. Hansen and Hansen : *DBM and Design*, PHI

CP-224 Microprocessor and Microcontrollers (3-0-0) 3

Introduction to **8085** microprocessor: CPU Architecture, CPU Specifications, CPU Pin Description, System Timing Diagrams, Instructions, Interrupts etc.
Introduction to **8086**/88 microprocessor: CPU Architecture, 8086 CPU Specifications, CPU Pin Description, System Timing Diagrams, Bus Standards, 8086 Address & Data Buses, Segmentation and Paging, Addressing Modes, Accessing Memory, RAM & Direct Memory Access, Memory Mapped I/O, Processor Registers, Data Organization.
Software Architecture: Introduction to Assembly Language Programming, Instruction and timing: instruction classification, instruction formats, addressing modes, instruction timings and status, interrupts.
I/O System Design: 8255 Programmable Peripheral Interface, 8259 Programmable Interrupt Controller, Direct Memory Access: basic concepts of DMA techniques, Description and interfacing of DMA controller 8257. Introduction to Microcontrollers.

Text/References:

1. Douglas V. Hall : Microprocessors and Interfacing, McGraw Hill

2. Gaonkar ; 8085 Architecture, Programming and interfaces, Penram Press
3. John Uffenbeck : The 8086/8088 Family -Design, Programming & Interfacing, Prentice Hall of India Private Limited.
4. Brey : Intel Microprocessor, The 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium & Pentium Pro Processor: Architecture, Programming, and Interfacing, PHI.

CP-226 Computer Architecture (3-0-0) 3

Introduction to computer architecture and organization: Digital components, Von Neumann Machine Architecture, Flynn Classification
Register Transfer Language: Micro operations - data transfer operations, arithmetic, logic and shift micro operations and their hardware implementations as a simple Arithmetic and logic unit.
CPU Organization: Addressing techniques - Immediate, direct, indirect, register, register indirect, index, relative and stack addressing techniques.
Instruction formats: Instruction set design, Instruction types: example for zero address, one address, two address and three address machines, Stack, accumulator and general purpose register organization
Arithmetic Algorithms: Arithmetic and Logic Unit, Adders - Full adder, Ripple carry adder, Carry look ahead adder, Carry select adder, carry save adder, Multiplication - Add and Shift method, Booth's Multiplier, m -Array Multiplier, Division - Restoring and Non restoring method.
Pipelining: Pipeline structure, pipeline performance measures, Pipeline types - Instruction and Arithmetic pipelines.
Memory Organization: Memory device characteristics, RAM organization: 1D and 2D organization, Virtual memory - Paging and Segmentation, High speed memories: Associative and Cache memory Control Unit Design, Hardwired and Micro programmed control unit design implementation techniques. Memory hierarchies.
Input-Output Design: IO interface, Bus structure, Modes of data transfer, Interrupts, Input Output Processor, Serial Communication

Text/References:

1. Computer Architecture: A Quantitative Approach, J.L. Hennessy and D.A. Patterson, 4th Edition Elsevier.

CP-228 Theory of Computation (3-1-0) 4

Introduction to automata theory, formal languages, recursive definitions, regular expressions, finite automata, transition graphs and Kleen's theorem.
Non-determination, finite automata with output, regular languages, minimization of finite automata, pumping lemma for regular languages.
Chomsky classification of languages, regular grammars, context free grammars, simplification of context free grammars, Normal forms of context free grammars.
Push Down Automata Theory: push down automata and languages, push down automata and context free grammars, pumping lemma for context free languages.
Turing hypothesis, Turing machine, Minskey's theorem, TM variation and encoding, Post machines, computability and acceptability.
Elements of prepositional logic and predicate calculus.

Text/ References:

1. Hopcroft, Motwani and Ullman: *Introduction to Automata Theory, Languages and Computation*, Pearson Education.
2. Cohen: *Introduction to Computer Theory*, Addison Wesley.
3. Martin: *Introduction to Languages and Theory of Computation, TMH.*
4. Papadimitriou, *Introduction to Theory of Computing*, Prentice Hall.
5. K.Krishnamurthy: *Theory of Computation*.

CP-252 DBMS Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with CP-222.
1. Conceptual designs using ER diagrams.
2. Design of databases. Based on templates, files and relational basis.
3. Development and implementation of DB system from the fundamentals.
4. Experiments on SQL queries.

CP-254 Assembly Language Programming Lab (0-0-3) 2 GS

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with CP-224.
Programming the 8085: 8085 instruction set, data transfer instruction, arithmetic, logic & branch operations. Rotate and compare. Instruction related to stack operations. Programming Techniques: looping, counting and indexing, counters and time delays, subroutines. Interfacing microprocessors with the additional devices like ADC/DAC and plotting them onto CRO.
Programming the 8086: Instruction types: Data Transfer, String, Arithmetic, Logical, Bit Manipulation, Program Transfer and Processor Control, the Processor Flags. Interfacing microprocessors with the additional devices like ADC/DAC and plotting them onto CRO. Stepper motor controller and PPI usages.

Text/References:

1. Gaonkar ; 8085 Architecture, Programming and interfaces, Penram Press
2. John Uffenbeck : The 8086/8088 Family -Design, Programming & Interfacing, Prentice Hall of India Private Limited.

Open Elective II (X-X-X) 3/4

Please refer to concerned Department.

B.Tech (Computer Engineering) Semester V

CP-321 Operating System (3-0-0) 3

Introduction: Need of Operating System, its evolution, types of operating systems, batch, multiprogramming, time sharing systems, real time systems.
Processes and processor management: process concept, systems programmers view of processes, operating systems view of processes, Process scheduling, Schedulers, interprocess communication and synchronization, race condition, mutual exclusion, semaphores, monitors, messages. Deadlocks prevention , avoidance, detection and recovery.
Memory Management: Contiguous allocation- partitioned memory allocation – fixed and variable partitioning, memory management with bit maps – swapping – relocation- protection and sharing. Non contiguous allocation – Paging – principles , page allocation, segmentation. Virtual memory concepts, address translation, management of virtual memory, page replacement policies, protection and sharing, working set model, hardware support.
File management: Command language users view of file system, file system design, disk space management directory structure, shared files, file system performance. File servers, security, protection mechanism. **Input/Output Management**: Device drivers, disk scheduling. Introduction to loaders,

linkers and relocating loaders.
Case study: UNIX/LINUX, Windows.

Text/References:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts. Sixth edition. Addison-Wesley (2003).
2. Andrew Tanenbaum, Modern Operating Systems, Prentice Hall.
3. William Stallings, Operasting Systems, Prentice Hall.
4. Andrew Tanenbaum & Albert Woodhull, Operating Systems: Design and Implementation. Prentice-Hall.
5. Harvey M. Deitel, An introduction to operating systems. Addison-Wesley.

CP-323 Compiler Design (3-0-0) 3

Translators: Introduction to compilers, translators, and interpreters, compilation process.
Lexical Analysis: Finite automata, Regular expressions, Design & implementation of lexical analysers.
Syntax Analysis: Context Free Grammars, Derivation and Parse trees, Bottom-up and Top-down Parsing. Ambiguity, Shift Reduce Parser, Operator Precedence Parser, Predictive Parsers, canonical collection of items, LR parsers.
Syntax directed translation: Syntax directed translation, Attributes, Intermediate codes, Three address codes.
Symbol table organization: Hashing, linked list, tree structures.
Memory allocation: Static and dynamic structure allocation.
Code optimization: Basic blocks, Flow graphs, DAG, Global data flow analysis – ud-chaining, available expressions, Loop optimization.
Code generation: Compilation of expression and control structures. Error detection and recovery.

Text/References:

1. Aho, Ullman and Sethi: Compilers – Principles, techniques and tools, Pearson Education.
2. Tremblay, Sorenson: The Theory and Practice of Compiler Writing, BSP.
3. Holub, Compiler Design in C, PHI.

CP-325 Computer Network (3-0-0) 3

1. Computer network architecture, Physical layer: Hardware, topology, data encoding,
2. Data Link Layer: Logical link Control: Error detection and Correction, ARQ protocols, Framing protocols(HDLC, LLC) Medium Access Control: Multiple access protocols, Channel Allocation, contention, reservation, round robin with Examples.
3. Network Inter connection: Generic switches, switch design issues, switching mechanism : virtual Circuit switching, datagram switching, source route switching, Bridge and bridge learning , Global Addressing scheme, fragmentation and reassembly, Address translation: ARP, RARP, ICMP, IP Scalability Issues, sub netting and super netting (CIDR) , IP Routing, EGP, BGP protocols
4. End to End protocols: End to end issues, UDP and TCP segment formats, connection establishment and termination, state transition sliding window protocol, TCP Flow control, Silly window syndrome, TCP retransmission, RTT Estimation, TCP Congestion Control and congestion avoidance protocols
5. Internet applications : Client server paradigm, DNS, SMTP, RPC, NFS and General network security issues.

Text/References:

1. Data Networks: Bertsekas and Gallagher, Phi.
2. Computer Networking A Top down Approach: J.F.Kurose, Pearson.
3. Computer Networks A Systems Approach: L. Peterson and B. Davie, Elsevier
4. Computer Networks and Internet: D.E. Comer, Pearson

CP-351 OS Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with CP-321.

1. Simple Unix-C programs: Programs using system calls, library function calls to display and write strings
2. Concurrent Programming concepts using fork, semaphore and pipes.
3. Programs for error reporting using errno,perror() function.
4. Programs to simulate process scheduling like FCFS,shortest Jobs First and Round Robin.
5. Programs to simulate page replacement algorithms like FIFO, Optimal and LRU.
6. Programs to simulate Free space management.
7. Programs to simulate virtual memory.
8. Programs to simulate deadlock detection.
9. Any other as per curriculum.

Text/References:

1. Unix concepts and applicaions by Sumitbha Das,TMH applications.
2. Unix Programming by stevens, Pearsons Education.
3. Shell Programming by Yashwant Kanetkar.
4. Operating System concepts by silberschatz, and Peter Galvin.

CP-353 System Programming Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations.

Programming exercises to implement typical lexical analyzers, parsers, intermediate code generation.
Assignments using LEX and YACC tools.
Programming exercises to implement assemblers, editors, debuggers etc.

Text/References:

1. Aho, Ullman and Sethi: Compilers, Pearson Education.
2. Levine, Mason and Brown: Lex and Yacc, O'Reilly.

CP-355 Network Programming Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with CP-325.

1. Programming for data encoding, CRC detection and Correction.
2. Estimation of network delay through OS utilities.
3. Simulation and Emulation of Bus and Star topology, DLC, MAC protocols using Benchmark LAN trainer kits.
4. Packet measurement and observation using network sniffing tools.
5. Use of sniffers for protocol dynamics.
6. Introduction to Socket programming and application development for internet.

Text/References:

1. Computer Networks and Internet: D.E. Comer, Pearson
2. TCP/IP Illustrated, W. Stevens, Vol 1-2, Pearson Eds.

Program Elective I (3-0-2) 4

One/More of the following courses as offered by Department.
CP-371 Object Oriented Programming

CP-373 Logical and Functional Programming
CP-375 Programming in Java
CP-377 Concurrent Programming

Open Elective III (X-X-X) 3/4

Please refer to concerned Department.

B.Tech (Computer Engineering) Semester VI

CP-322 Software Engg. (3-0-0) 3

Introductory Concepts: Historical perspective, System Definition, Software Life Cycle, Software Engineering paradigms.
System analysis: Feasibility study requirement analysis, Cost benefit analysis, Planning systems, Analysis tools and techniques.
System Design: design fundamentals, Modular Design, Data and procedural design, object oriented design.
System Development: Code documentation, Program design paradigms. Verification, Validation and Testing: testing methods, Formal Program Verification, Testing Strategies.
Software Maintenance: Maintenance Characteristics, Maintainability, Maintenance tasks and side effects.

Text/References:

1. Pressman R.S: Software Engineering: A Practitioner approach, McGraw Hill.
2. Sommerville I: Software Engineering, Addison Wesley
3. Ghezzi C. Jazayeri M and Mandrioli: Fundamentals of Software Engg. , PHI.

CP-324 Design and Analysis of Algorithms (3-0-0) 3

Algorithm Analysis: Asymptotic notation, solution of recurrence, model of computation, time and space complexities, average and worst case analysis, Amortized analysis.
Algorithm Design Techniques: Greedy algorithm, dynamic programming, divide and conquer, backtracking, branch and bound.
Graph Algorithms: Shortest path algorithms, Disjoint set operations, minimum spanning tree algorithm, network flow, matching, coverings, applications of DFS:- bi-connectivity, Euler circuits, strongly connected components, topological sort, and articulation point.
Dynamic Programming: Chained matrix multiplication, longest common subsequence.
Divide and Conquer: Order Statistics – finding the median, exponentiation, matrix multiplication, LCS.
Computational Geometry: Line segments, Optimal polygon triangulation.
Approximate Algorithm: Travelling Salesman Problem, vertex-cover problem.
Primality testing, Integer factorization, Randomized algorithms, Probabilistic algorithms.
String Matching algorithms: Rabin Karp, KMP, Boyer Moore.
Matrix Algorithms – Strassen Matrix multiplication, LUP decomposition.
Construction of codes: Shannon Fano and Huffman codes.
Introduction to problem classes – NP, NPC, NP-Hard.

Text/References:

1. Cormen, Leiserson, Rivest: *Introduction to Algorithms*, Prentice Hall of India.
2. Horowitz and Sahani: *Fundamental of Computer algorithms*.
3. Aho A.V , J.D Ulman: *Design and analysis of Algorithms*, Addison Wesley

4. Brassard : *Fundamental of Algorithmics*, PHI.
5. W.W. Peterson and E. J. Weldon: *Error correcting codes*.
6. Sara Baase, Allen Van Gelder: *Computer Algorithms: Introduction to Design and Analysis*, Pearson Education.

CP-352 Algorithms Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations.

1. Implementation of graph algorithms – DFS, Shortest Path, MST, articulation point, topological sorting.
2. Implementation of Network Flow, matching, covering.
3. Implementation of pattern matching algorithms.
4. Determination of k^{th} shortest number in a given sequence.
5. Dynamic programming – Applications to problem solving.
6. Approximation algorithms for NP problems.
7. Randomized algorithms implementation.

Text/References:

1. Cormen, Leiserson, Rivest: *Introduction to Algorithms*, Prentice Hall of India.
2. Horowitz and Sahani: *Fundamental of Computer algorithms*.
3. Aho A.V , J.D Ulman: *Design and analysis of Algorithms*, Addison Wesley
4. Brassard : *Fundamental of Algorithmics*, PHI.
5. W.W. Peterson and E. J. Weldon: *Error correcting codes*.
6. Sara Baase, Allen Van Gelder: *Computer Algorithms: Introduction to Design and Analysis*, Pearson Education.

CP-354 Seminar (0-2-0) 2

The topics selection covering the latest and relevant topics related to the emerging areas. Ideally, some recent reputed journal papers abstraction and presentation shall be encouraged for presentation. The evaluation shall be continuous and through components evaluation viz. content, coverage, depth, presentation, response to the queries, and seminar report. In case of unsatisfactory performance, an X grade can be awarded for extension work during summer term.

Program Elective II (3-0-2) 4

One/More of the following courses as offered by Department.
CP-372 Digital Signal Processing
CP-374 Wireless Communications
CP-376 VHDL
CP-378 Neural Networks

Open Elective IV (X-X-X) 3/4

Please refer to concerned Department.

Open Elective V (X-X-X) 3/4

Please refer to concerned Department.

B.Tech (Computer Engineering) Semester VII

CP-451 Industrial Training (0-2-0) 2

The evaluation shall be continuous and through components evaluation viz. content, coverage, depth, presentation, demonstration, response to the queries, and training report. In case of unsatisfactory performance, and failure extra credit course from the department equivalent to CP-451 can be permitted through consent of DUGC.

CP-453 Free and Open Source Lab (0-0-3) 2

The following proposed coverage are broad guiding areas lab. The instructor offering the course in consultation with the theory offered can adopt further variations.

1. Linux basics and installation and management of the Linux.
2. Different types of software development environment (Eclipse)
3. make and other software construction utilities on Linux.
4. Version control and managing project in open source.
5. Managing large software development through wiki or alike project management tools.
6. Introduction to scripting for system management.

Program Elective III (3-0-2) 4

One/More of the following courses as offered by Department.
CP-471 Computer Graphics
CP-473 Speech Recognition
CP-475 Mobile Computing
CP-477 Software Project Management
CP-479 Data Compression

Program Elective IV (3-0-2) 4

One/More of the following courses as offered by Department.
CP-481 Advances in Compiler Design
CP-483 Natural Language Processing
CP-485 Programming Paradigms
CP-487 Wireless and Ad hoc Networks
CP-489 Real Time Systems

Program Elective V (3-0-2) 4

One/More of the following courses as offered by Department.
CP-491 Cryptography
CP-493 Embedded Systems
CP-495 Data Mining
CP-497 VLSI Algorithms

CP-455 Project Lab (0-2-3) 5

Objective of this elective is to facilitate transfer of knowledge acquired by a student to a field of his own choice for application to solving a problem. Student is expected to collect and study relevant material under mentorship of a faculty member working in similar area; identify a suitable problem and propose methodology towards its solution. Alternately a student can explore hardware implementation of existing solution(s). This elective shall act as prequel to project work for next semester. The project coordinator(s) from the department for continuity shall coordinate this course. Grouping and division shall be applicable as defined in the major project of final semester.

Open Elective VI (X-X-X) 3/4

Please refer to concerned Department.

B.Tech (Computer Engineering) Semester VIII

CP-452 Major Project (0-10-0) 10

The major project covers lab component of the final semester work. The evaluation of project shall be continuous and will be done through project coordinator(s). The evaluation mechanism shall be evolved based on the existing practices through DUGC rectified from time to time. Ideally the project should comprise with group size of two students shall be limited to maximum 4 students and the groups shall be evenly distributed among faculty through coordinator(s). Internal and external components shall not exceed 40% each of the overall marks.

CP-454 Group Discussions (0-0-3) 2

This lab will remain only if Industrial training is given weightage of 02 credits.

Program Elective VI (3-0-2) 4

One/More of the following courses as offered by Department.
CP-470 Network Security
CP-472 Biometrics
CP-474 High Level Synthesis of Digital Systems
CP-476 Selected Topics in Cryptography
CP-478 Digital Image Processing

Program Elective VII (3-0-2) 4

One/More of the following courses as offered by Department.
CP-480 Evolving Architectures
CP-482 Topics in Advanced Computing
CP-484 Software Testing and Validation
CP-486 Machine Learning
CP-488 Modeling and Simulation

Open Elective VII (X-X-X) 3/4

Please refer to concerned Department.

Program Elective I (Semester V)

CP-371 Object Oriented Programming (3-0-2) 4

Object Oriented Programming and Design: Review of abstraction, objects and other basics, Encapsulation, Information hiding, method, Signature, Classes and Instances, Polymorphism and inheritance.

C++ Programming Basics: Fundamentals, variables and assignments, Input and Output, Data types and expressions, flow of control, subprograms, top-down design, predefined functions, user defined functions, procedural abstractions, local variables, overloading function names, operator overloading, parameter passing, this pointer, destructors, copy constructor, overloading the assignment operator, virtual functions, function calling functions, friend functions, recursive functions, recursive member functions. Static member function.

C++ Object oriented concepts: Objects and classes, use of file for I/O, formatting output with stream functions, Character I/O, inheritance, structures for diverse data, structures as function arguments, initializing structures, defining classes and member functions, public and private members, constructors for initialization, standard C++ classes, derived classes, flow of control, use of Boolean expressions, multiway branches, use and design of loops. Friend function and friend class.

C++ Data structures and Advanced Topics: Arrays – programming with arrays, arrays of classes, arrays as function arguments, strings, Multidimensional arrays, Arrays of strings, pointers Dynamic arrays, Classes and dynamic arrays, Base classes, access control, Templates- generic classes and functions, namespaces. Standard Template Library.

Text/References:

1. Balaguruswamy: Object-oriented Programming with C++.
2. Robert Lafore: C++ Programming
3. Ashok N. Kamthane : Object Oriented with C++, Pearson Education

CP-373 Logical and Functional Programming (3-0-2) 4

Introduction to logic programming, Prolog - Lists, cut operator, and sorting, Data structures, text strings, operators - extensions of SWI Prolog, Searching state space, clause management, and parsing in Prolog Introduction to functional programming, lambda calculus, Programming language Haskell - introduction, lists, User-defined data types, type classes, and arrays in Haskell. Input/Ouput in Haskell - type classes IO and Monad, Simple applications/programs in Haskell.

Text/References:

1. Nilsson, Maluszynski: Logic, Programming and Prolog, John Wiley.
2. Thompson: Haskell, The Craft of Functional Programming, Addison-Wesley.

CP-375 Programming in Java (3-0-2) 4

Introduction: Internet, Java as a tool for internet applications, Byte Code and its advantages.

Object Oriented Programming and Design: Review of Abstraction, Objects and other basics, Encapsulation, Information hiding, Method, Signature, Classes and Instances, Polymorphism, Inheritance, Exceptions and Exception Handling with reference to object modeling, Coupling and Cohesion in object oriented software. Object Oriented Design – Process, Exploration and Analysis.

Java Programming Basics: Fundamentals: Variables and assignments, Input and Output, Data Types and Expressions, Flow of control, Local variables, Overloading Parameter passing, this pointer,

Java Object Oriented Concepts: Objects and Classes: Use of file for I/O, Formatting output with stream functions, Character I/O, Inheritance, Public and private members, Constructors for initializations, Derived classes, Flow of Control

Java Data Structures and Advanced Topics

Arrays – Programming with arrays, arrays of classes, arrays as function arguments, Strings, Multidimensional arrays, Arrays of strings, vectors, Base classes.

Introduction to Java Applets

Text/References:

1. Herbert Schildt: JAVA 2 - The Complete Reference, TMH, Delhi
2. U.K. Chakraborty and D.G. Dastidar: Software and Systems - An Introduction, Wheeler Publishing, Delhi.
3. Joseph O’Neil and Herb Schildt: Teach Yourself JAVA, TMH, Delhi.

CP-377 Concurrent Programming (3-0-2) 4

Concurrent versus sequential programming. Concurrent programming constructs and race condition. Synchronisation primitives. Processes and threads. Interprocess communication. Livelock and deadlocks, starvation, and deadlock prevention. Issues and challenges in concurrent programming paradigm and current trends.

Text/References:

1. Mordechai Ben-Ari. *Principles of Concurrent and Distributed Programming*, Prentice-Hall International.
2. Greg Andrews. *Concurrent Programming: Principles and Practice*, Addison Wesley.
3. Gadi Taubenfeld. *Synchronization Algorithms and Concurrent Programming,* Pearson.
4. M. Ben-Ari. *Principles of Concurrent Programming,* Prentice Hall.
5. Fred B. Schneider. *On Concurrent Programming,* Springer.
6. Brinch Hansen. *The Origins of Concurrent Programming: From Semaphores to Remote Procedure Calls*,

Program Elective II (Semester VI)

CP-372 Digital Signal Processing (3-0-2) 4

Introduction to Continuous time Systems ,idea about Linear Time Invariant System LTI systems. Fourier Transforms.

Discrete Time Systems: Sampling and aliasing, LTIs , Representation of Sequences by Fourier Transform and properties of Fourier Transform.

Z-Transform, Structures for discrete system, DFT, Computation of DFT. FIR Filters, frequency response of FIR filters. IIR Filters, spectrum analysis. FFT Algorithms

Text/References:

1. *Discrete Time Signal Processing* by Alan V Oppenheim, Ronald W Schafer.- PHI
2. *Digital Signal Processing Primer*: K.Steiglitz, Pearson.
3. *Signal and Systems*: S.Haykin and Veen, Wiley
4. *DSP First: A Multimedia Approach*: J.F.McClellan, Schafer and Yodar, Pearson.

CP-374 Wireless Communications (3-0-2) 4

History of wireless communication, and future trends. Wireless Generations and Standards. Cellular Concept and Cellular System Fundamentals .Trunking Cell Splitting and Sectoring. Mobile Radio signal propagation, path loss and channel models. Large Scale Path Loss. Small Scale Path Loss - Rayleigh and

Rician Fading. Analog Modulation Schemes for Wireless Communication - AM/FM. Digital Modulation Techniques for Wireless Communication Preliminaries. Baseband Modulation Schemes Bandpass Modulation Techniques. Fading Counteraction – Diversity, Coding and Interleaving. Source and Channel Coding. Speech Coding for Wireless Communications. Adaptive Equalization. Multipath Propagation, Doppler. Multiplexing and Multiple Access techniques. TDMA, FDMA , ALOHA - Packet Radio, Spread Spectrum-CDMA ,Frequency Hopped Spread Spectrum, Inter-Symbol Interference (ISI), ISI mitigation; Equalization, Random Access Protocols. Wireless Networking, Wireless Standard. Third generation systems and advanced topics Wideband-CDMA, MCCDMA.
OFDM principles: Comparison of OFDM and CDMA. WLAN and Bluetooth.

Text/References:

1. Wireless Communications: Principles and Practice, 2nd edition, T. Rappaport, Prentice Hall, 2002
2. K. Pahlavan & P. Krishnamurthy, Principles of Wireless Networks, Prentice Hall:
3. Wireless Communications Systems, A. Goldsmith, Cambridge, 2005
4.

CP-376 VHDL (3-0-2) 4

1. Overview of VHDL, fundamentals of VHDL, Lexical elements Data types and objects
2. Data Flow style: Conditional and selected Concurrent assignment, block assignment If and wait statement, Design for synthesizability
3. Structural style: Instantiation and component declaration, statement configuration declaration, generate statement, examples of structural design
4. Behavioural Style : Signal assignment, statement like case, process and wait loop, exit etc., concurrent signal assignment statements, function and procedures, file I/O operations and Testbenches.

Text/References:

1. Peter J. Ashenden ,” The Designer's Guide to VHDL", published by Morgan Kaufmann” Kaufmann Pub.

1. SS Limaye,” Digital Design with VHDL”, CMR
2. Douglas Parry, “ VHDL Programming by Example”, MGH
3. Xilinx, “ Programmable Logic Design Quick Start Hand Book II ed.
4. Xilinx,” A CPLD VHDL Introduction Application Notes”

CP-378 Neural Networks (3-0-2) 4

Neural Architecture: Neuron model, transfer function, hamming and Hopfield network, perceptron, learning rule, recurrent networks.
Back propagation: generalized delta rule, limitations, modifications – momentum, variable learning rate, conjugate gradient.
Learning: Supervised, associative, competitive, unsupervised learning.
Unsupervised learning: Self-organizing maps, Adaptive Resonance Theory.
Neural network applications: Pattern classification, function approximation.

Text/ References:

1. Simon Haykin: *Neural Networks: A Comprehensive Foundation (2nd Edition)*
2. Christopher M. Bishop: *Neural Networks for Pattern Recognition*
3. James A. Freeman, David M. Skapura: *Neural Networks*, Pearson Education.
4. Martin T. Hagan: *Neural Network Design,* Thomson Learning.

Program Elective III (Semester VII)

CP-471 Computer Graphics (3-0-2) 4

Introduction to computer graphics: Vector and Raster graphics, Graphic primitives and attributes. Computer graphics devices **-** CRT, plasma, LCD, plotters, Scan conversion of line, circle and ellipse.
Viewing Transformations: Coordinate system - world, device and normalized device coordinates, Window and Viewport and viewing transformations
Filling and Clipping - Flood fill and seed fill algorithms and scan line polygon filling algorithms, Cohen Sutherland clipping algorithms for Polygon
Geometric transformations - 2D and 3D transformations: Translation, Scaling, rotation, Shearing, reflection etc., Transformations about an arbitrary axis
Projections: Parallel - Orthographic, Plans and Elevations, Axonometric - Isometric, Diametric, trimetric, Perspective - One point, two point, three point.
Hidden surface removal: Object space and image space algorithms, Back space removal, Z-buffer, scan line, area subdivision, painters, BSP tree, Floating horizon and ray tracing methods
Curves: Spline representations, Curve representation techniques, Continuity constraints, Hermite Interpolation, Bezier curves, B-Spline curves

Text/References:

1. Foley, Van Dam. Computer Graphics: Principles and Practic. Addison Wesley.
2. Hearn and Baker. Computer Graphics. PHI.
3. Rogers and Adams. Mathematical Elements of Computer Graphics. McGraw Hill.
4. Rogers and Adams. Procedural Elements of Computer Graphics. McGraw Hill.

CP-473 Speech Recognition (3-0-2) 4

Overview of Speech Recognition; What is Speech; Why is it important; Applications and issues. Speech Production; Mechanism of speech production; Categories of sounds; Sound units in indian languages. Nature of Speech Signal; Source-system characteristics; Segmental and suprasegmental features; Temporal and spectral parameters for sound units in indian languages. Basics of Digital Signal Processing; Signals and systems; Discrete fourier transform; Digital filtering; Stochastic processes. Speech Signal Processing Methods: Short-time spectrum analysis; Spectrograms; Linear prediction analysis; Cepstrum analysis. Speech Recognition; Isolated word recognition; Connected word recognition Continuous Speech Recognition; Speech recognition problem; Hidden markov models. Other Applications: Word spotting; Speaker recognition; Speech enhancement; Speech synthesis; Practical issues in speech Recognition.

Text/References:

1. Spoken Language Processing: A Guide to Theory, Algorithm and System Development by Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, Raj Reddy Prentice Hall PTR; ISBN: 0130226165
2. Speech Communications : Human & Machine by Douglas O'Shaughnessy, IEEE Press, Hardcover 2nd edition, 1999; ISBN: 0780334493.
3. Digital Processing of Speech Signals, Rabiner and Schafer, Prentice Hall, 1978.
4. Fundamentals of Speech Recognition, Rabiner and Juang, Prentice Hall, 1994.
5. Speech and Audio Signal Processing : Processing and Perception of Speech and Music by Nelson Morgan and Ben Gold, July 1999, John Wiley & Sons, ISBN: 0471351547
6. Discrete-Time Speech Signal Processing: Principles and Practice by Thomas F. Quatieri Publisher: Prentice Hall; ISBN: 013242942X; 1st edition (October 29, 2001)
7. Speech Processing and Synthesis Toolboxes by Donald G. Childers, John Wiley & Sons, September 1999; ISBN: 0471349593

CP-475 Mobile Computing (3-0-2) 4

Introduction to mobile computing: principles, classification & overview of devices, operating systems. Wireless transmission: brief overview, multipath propagation, hidden & exposed terminals. Medium access control & protocols: SDMA, FDMA, TDMA, DAMA, FAMA, PRMA, Reservation TDMA, polling, CSMA/CA, CDMA etc.
Wireless LAN: infrastructure & ad-hoc networks, IEEE 802.11, HIPERLAN.
Mobile network layer: mobile IP, DHCP, infrastructure & Ad-hoc
routing. Mobile transport layer: indirect TCP, snooping TCP, mobile TCP etc. mobile support, WWW & mobility, WAP.

Text/ References:

1. Principles of mobile computing Hansmann & Merk., Springer
2. Mobile communications Jochen Schiller , Pearson
3. 802.11 wireless networks Matthew S.Gast, O'REILLY.
4. Wireless LANs: Davis & McGuffin, McGraw Hill
5. Mobile Communications Handbook by Jerry D. Gybson
6. Mobile Communications Handbook by Raymond Steel

CP-477 Software Project Management (3-0-2) 4

Software Project Management Concept: The Management Spectrum, People, Product, Process & Project. Software Process & Project Matrix: Software Measurement, Size Oriented Matrices, Function Oriented Matrices.
Software Project Planning: Objectives, Decomposition Techniques and Empirical Estimation Model. Risk Analyses and Management: Risk Identification, Projection, Risk Identification, Projection, Risk Refinement, Risk Monitoring and Management.
Project Scheduling & Tracking, Software Quality Assurance, Software Configuration Management.

Text/References:

1. R. S. Pressman, Software Engineering
2. P. Jalote, Software Project Management in Practice.
3. B. Hughest & M. Cotterell, Software Project Management.

CP-479 Data Compression (3-0-2) 4

Compression: Need, Lossless v/s lossy compression, review of information theory, prefix codes, uniquely decodable code.
Lossless Compression: Huffman coding – minimum variance, optimal, non-binary, extended, adaptive. Applications and limitations of Huffman codes, Run length encoding, Arithmetic coding, Predictive coding – Burrows-Wheeler transform, Delta modulation, Adaptive delta modulation
Dictionary based compression - Lempel-Ziv-Welch, LZ77 and LZ-78
Lossy Compression Techniques – JPEG and its application
Error detection and correction: Parity, 1,2,n dimensions, Hamming codes, p-out-of-q codes
Quantization: Scalar and Vector Quantization.

Texts/References:

1. Khalid Sayood, Introduction to Data Compression, Morgan Kauffman
2. Greg A. Harris, Darrel R. Hankerson, Peter D. Jr. Johnson, Introduction to Information Theory and Data Compression, Second Edition, Chapman and Hall.
3. Saloman, Data Compression, Springer Verlag.
4. Nelson, The Data Compression book, Hungry Minds

Program Elective IV (Semester VII)

CP-481 Advances in Compiler Design (3-0-2) 4

A Tour of Compiler Design, LR Parsers – SLR parsers, Canonical LR and LALR parsers, Lex and Yacc Tools, Control-flow Analysis, Control-flow Graphs, Basic Blocks, Data-flow Analysis, Dependence Analysis, Global Optimizations, Loop Optimizations, Dominators, Loop-invariant computations, Code motion, Data Dependence Analysis in Loops, Loop Scheduling, Runtime System Architectures and Automatic Memory Management Techniques.

Text/References:

1. Aho, Alfred V., Sethi, Ravi, Ullman, Jeffrey D., Compilers: Principles, Techniques and Tools, Addison-Wesley.
2. Steven Muchnick, Advanced Compiler Design & Implementation, Morgan Kaufmann.
3. Keith Cooper and Linda Torczon, Engineering a Compiler, Morgan Kaufmann.

CP-483 Natural Language Processing (3-0-2) 4

Introduction; Goals of Natural Language Processing and Computational Linguistics. Finite State Automata and Transducers, Morphology. Parsing: Context Free Grammars, Generalized Phrase Structure Grammar, Earley Parsing ALgorithm. Transformational Grammar, Computational Models and Knowledge Representation. Semantics; Interpretation, time, tense and lexical semantics. Machine Translation, Natural Language Interfaces, Natural Language Generation.

Text/References:

1. Allen James, Natural Language Understanding, Second Edition, Benjamin/Cumming, 1995. Grosz, Sparck-Jones Webber
2. Readings in Natural Lnaguage Processing, Morgan Kaufmann, 1986. Winograd T.
3. Language as a Cognitive Process, Addison Wesley, 1972. Marcus M.
4. A Theory of Synactic Recognition for Natural Language, MIT Press, 1980.

CP-485 Programming Paradigms (3-0-2) 4

Introduction: History of Programming Languages, Syntax and Semantics, Paradigms.
Procedure-oriented Programming: Procedures and Modularity, Built-in and User-defined Functions, Scope and Lifetime of Variables, Structures, Recursion, Pointers and Dynamic Memory Allocation.
Case Study: **C** language.
Object-Oriented Programming: Objects and Classes, Encapsulation, Information hiding, Method and Signature, Polymorphism and Inheritance.
Case Study: **C++** language.
Aspect-Oriented Programming: Crosscutting concerns, Aspect, Joinpoint and Pointcuts, Advice and static crosscutting.
Case Study: **AspectJ** language**.**
Functional programming: Introduction, lists, User-defined data types, type classes, and arrays, Input/Ouput and Monad, Simple applications/programs.
Case Study: **Haskell** language.

Text/References:

1. Kerninghan and Ritchie: The 'C' programming language, Pearson Education.
2. Robert Lafore: C++ Programming, Sams.
3. Laddad: AspectJ in Action, Dreamtech.
4. Thompson: Haskell, The Craft of Functional Programming, Addison-Wesley.

CP-487 Wireless & Ad-hoc Networks (3-0-2) 4

Fundamentals of Wireless Communication Technology The Electromagnetic Spectrum – Radio Propagation Mechanisms Characteristics of the Wireless Channel - IEEE 802.11a,b Standard Origin Of Ad hoc: Packet Radio Networks , Technical Challenges, Driving Applications, Components of Packet Radios What Is an Ad Hoc Network? Types of Ad hoc Mobile Communications. Key definitions of ad-hoc, Advantages of ad-hoc/sensor networks, Unique constraints and challenges, Driving Applications, Media Access Control (MAC) Protocols, Issues in designing MAC protocols, Classifications of MAC protocols, MAC protocols Routing Protocol: Global State Routing (GSR), Dynamic State Routing (DSR), Fisheye State Routing (FSR), Ad hoc On-Demand Distance Vector (AODV), Destination Sequenced Distance – Vector Routing (DSDV). Transport Layer, Security Protocols :Introduction Issues in Designing a Transport Layer Protocol for Ad Hoc Wireless Networks - Design Goals of a Transport Layer Protocol for Ad Hoc Wireless Networks -Classification of Transport Layer Solutions, security in Ad Hoc Wireless Networks – Network Security Requirements - Issues and Challenges in Security Provisioning -Network Security Attacks.

Text/References:

1. C. Siva Ram Murthy and B.S. Manoj “Ad Hoc Wireless Networks: Architectures and Protocols”, Prentice Hall PTR,2004
2. C.K. Toh, Ad Hoc Mobile Wireless Networks: Protocols and Systems, Prentice Hall PTR ,2001 Charles E. Perkins, Ad Hoc Networking, Addison Wesley, 2000
3. Wireless Communications: Principles and Practice, 2nd edition, T. Rappaport, Prentice Hall, 2002
4. K. Pahlavan & P. Krishnamurthy, Principles of Wireless Networks, Prentice Hall

CP-489 Real Time Systems (3-0-2) 4

Introduction to Real-time systems, Issues in Real-time Systems, Real-time System Components, Classification of Real-time systems and Real-time tasks. Misconceptions about Real-time computing. Real-time System requirements: Speed, Predictability, reliability, adaptability. Specification of timing constraints.

Real-time scheduling: Requirements and Issues, Terminology, modeling, Introduction static and dynamic scheduling schemes, cyclic scheduling, priority driven scheduling of periodic tasks, schedulability tests, Aperiodic task scheduling: fixed priority server/non-server based scheduling algorithms. Practical factors/overheads.

Task Synchronization: Need and priority inversion problem, Priority Inheritance protocol, priority ceiling protocol and stack-based priority ceiling protocol for fixed priority preemptive system.

Introduction to multiprocessor real-time systems, problems and issues.

An overview of a real-time operating system

Text/References:

1. J.W.S.Liu: *Real-Time Systems*, Pearson Education Asia
2. S.T.Lavi, A.K.Agrawala: *Real-time system Design*, McGraw Hill
3. P.A.Laplante: *Real-time Systems Design and Analysis, An Engineer's Handbook*, IEEE Press
4. P.D.Laurence, K.Mauch: *Real-time Microcomputer System Design, An Introduction*, McGraw Hill

Program Elective V (Semester VII)

CP-491 Cryptography (3-0-2) 4

Review of Number theory: Prime numbers, modular arithmetic, Fermat’s theorem, Euler’s theorem,

Chinese remainder theorem, Discrete logarithms, Random number generation, factoring, prime number generation.
Cryptography: Need, conventional techniques, stream ciphers, block cipher, steganography. Public v/s private key cryptography.
Stream Ciphers: Caesar Cipher, mono-alphabetic and poly-alphabetic ciphers, Playfair Cipher, Hill Cipher, Rotor machines, One time pad,.
Random Number Generation: Pseudo Random Number, PRNG, LFSR, Blum-Blum Shub generator
Private-key cryptography: Feistel structure, DES (Data encryption standard), design of S-boxes, AES, Triple DES.
Public key cryptography: Key management, Key exchange – Diffie-Hellman, El-Gamal, Merkle's Puzzle, Authentication, Signatures, Deniability, RSA.
Threshold Cryprography: Sharing Secrets.
Digital Signature: DSA and its variants, discrete logarithm based digital signatures.
One-way hash functions – MD5, SHA (Secure Hash Algorithm).
Cryptanalysis: Differential and linear cryptanalysis - cracking DES.

Text/References:

1. Stallings, Cryptography and Network Security: Principles and Practice, Pearson Education Asia. ISBN 981-403-589-0.
2. B Schneier, Applied Cryptography, Wiley. ISBN 0-471-11709-9
3. D Kahn. The Codebreakers, Sphere books. ISBN 0-7221-51497
4. P Wayner, Disappearing Cryptography, Academic Press. ISBN 0-12-738671-8
5. Cracking DES, Electronic Frontier Foundation. ISBN 1-56592-520-3
6. A.J. Menezes, P.C. van Oorschot and S.A. Vanstone, Applied Cryptography, CRC Press, ISBN 0-8493-8523-7, 1997
7. D.R. Stinson, Cryptography - Theory and practice, CRC Press, ISBN 0-8493-8521-0, 1995

CP-493 Embedded Systems (3-0-2) 4

Introduction to embedded systems., design representations, level of abstractions, design methodologies, Models and architectures, Taxonomy of models and architectures, brief descriptions of specification languages, Specification requirement for embedded systems, Spec Chart and Spec Chart Description.
Design challenges & issues, hardware and software design, co-design of software and hardware, ASIC.
Design quality estimation : Quality matrix, software and hardware estimation.
Introduction
Sample design Specification of Answering machine/ Microcontroller 8051.

Text /References:

1. Denial D. Gajski , frank Vahid: *Specification and design of embedded systems*, PH
2. Jonathan W. Valvano: *Embedded Microcomputer Systems*, Thomson Learning
3. Myke Predko: *Programming and Customizing the 8051 Micro Controller*, TMH
4. Ayala : *8051 Micro controllers*, Penram Press

CP-495 Data Mining (3-0-2) 4

Introduction : Basic Data Mining Tasks, Data Mining Issues, Data Mining Metrics, Data Mining from a Database Perspective.
Data Mining Techniques : A Statistical Perspective on Data Mining, Similarity Measures, Decision Trees, Neural Networks, Genetic Algorithms.
Classification : Statistical-Based Algorithms, Distance-Based Algorithms, Decision Tree-Based Algorithms, Neural Network-Based Algorithms, Rule-Based Algorithms, Combining Techniques.
Clustering : Similarity and Distance Measures, Hierarchical Algorithms, Partitional Algorithms, Clustering Large Databases, Clustering with Categorical Attributes.
Association Rules : Basic Algorithms, Parallel and Distributed Algorithms, Incremental Rules, Advanced Association Rule Techniques, Measuring the Quality of Rules.

Advanced Techniques : Web Mining, Spatial Mining, Temporal Mining.

Text/References:

1. M. H. Dunham. Data Mining: Introductory and Advanced Topics. Pearson Education. 2001.
2. J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufman. 2001.
3. I. H. Witten and E. Frank. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann. 2000.
4. D. Hand, H. Mannila and P. Smyth. Principles of Data Mining. Prentice-Hall. 2001.

CP-497 VLSI Algorithms (3-0-2) 4

1. Introduction of VLSI Technology, VLSI design cycle, design styles, basic Layout rules and circuit abstraction, introduction to standard Cell, Gate array, FPGA
2. Overview of basic graph algorithms, Graph algorithms for physical Design
3. Partitioning: Classification of partitioning algorithms, Karnighan-Lin Algorithm, FM Algorithm, Ratio cut algorithm
4. Floor-planning: Rectangular dual graph approach of floor-planning, hierarchical tree based approach, Integer programming based floor-planning.
5. Placement: placement by simulated annealing and force directed method
6. Routing: classification of routing algorithms, Global routing: Maze routing algorithms, line probe algorithms, Steiner tree based algorithms, Detailed Routing: Single layer and two layer routing algorithms, routing in FPGAs

Text/References:

1. Naveed Shervawani, “ Algorithms for VLSI physical Design Automation “ III Ed Springer
2. Sarrafzadeh and Wong “ An introduction to VLSI Physical design “ MGH
3. Sze: VLSI Technology
4. Weste and Eshranghan, “ Introduction toVLSI Design”. Pearson Edu.
5. Sadiq M. Sait, Habib Youssef, "VLSI Physical Design Automation: Theory and Practice", World Scientific Publishing Company;
6. Cormen Leiserson, Rivest, “ Introduction to Algorithms”, Pearson Edu.

Program Elective VI (Semester VIII)

CP-470 Network Security (3-0-2) 4

Review of wired/wireless network protocols, intrusion detection systems, malicious software.
Review of cryptographic algorithms, protocols, cryptanalysis, authentication and signature protocols.
Kerberos, PKI, real-time communication security, IPSec: AH, ESP, IKE.
SSL/TLS, e-mail security, PEM and S/MIME, PGP, web security, network management security, wireless security.
Threats in networks, network security controls, firewalls, intrusion detection, administering security
Honeypots, password management, malicious software, viruses and countermeasures

Text/References:

1. C. Kaufman, R. Perlman, *Network Securit*, Prentice Hall.
2. Kurose & Ross, *Computer Networking*, Pearson Education.
3. Schiller J., *Mobile Communications*, Pearson Education.
4. W. Stallings, *Cryptography and Network Security Principles and practice*, Pearson Education.

CP-472 Biometrics (3-0-2) 4

Biometrics: Need, Conventional techniques of authentication, challenges - legal and privacy issues.
Biometrics: DNA, fingerprint, Iris, Retinal scan, Face, hand geometry, human gait, speech, ear.
Handwriting, Keystroke dynamics, Signature
Multimodal biometrics: Combining biometrics, scaling issues.
Biometric template security.

Texts/References:

1. Julian D. M. Ashbourn, Biometrics: Advanced Identify Verification: The Complete Guide
2. Davide Maltoni (Editor), et al, Handbook of Fingerprint Recognition
3. L.C. Jain (Editor) et al, Intelligent Biometric Techniques in Fingerprint and Face Recognition
4. John Chirillo, Scott Blaul, Implementing Biometric Security

CP-474 High Level Synthesis of Digital Systems (3-0-2) 3

Overview. Design methodologies. Abstractions and views.
Review of basic concepts in algorithms and graph theory
Design representation and modeling, Modeling languages, Abstract models
Synthesis at higher levels of abstraction
Scheduling, Resource sharing
Structural synthesis: Module selection. Pipeline. Control
Synthesis at lower levels of abstraction, Logic synthesis

Text/Reference:

1. G. D. Micheli. *Synthesis and optimization of digital systems*.
2. N.D. Dutt, D. D. Gajski. *High level synthesis*, Kluwer, 2000.
3. T. H. Cormen, C. E. Leiserson and R. L. Rivest, "*Introduction to Algorithms*," McGraw-Hill, 1990.
4. Recent papers from journals and conferences.

CP-476 Selected Topics in Cryptography (3-0-2) 4

Elliptic Curve Cryptography
Secret Sharing, Threhold cryptography – Robust ElGamal system
Visual Cryptography.
Interactive zero knowledge proofs, witness hiding protocols.
Group encryption, decryption. Group signatures, ring signatures.
EVoting: requirements, issues and challenges, existing solutions, write-in ballots.
Pair based cryptography – Weil and Tate pairing.
Contemporary Issues.

Text/References:

1. Oded Goldrich. Foundations of Cryptography – Vol I & Vol II.
2. Selected paper and online reference material.

CP-478 Digital Image Processing (3-0-2) 4

Digital Image Fundamentals: Image Model, Sampling, Quantization, Neighborhood, connectivity of pixels, Labelling of connected components, Distance measures
Image Transforms: Fourier Transform, Discrete Fourier Transform, Properties of 2D Discrete Fourier Transform, The fast Fourier Transform and its algorithm, number of operations, the inverse FFT. Discrete Cosine Transform and its applications, KL Transform, Convolution and correlation
Image Enhancement: Enhancement by point processing, spatial filtering, enhancement in frequency domain, generation of spatial masks from frequency domain specifications
Image Segmentation: Detection of discontinuities, edge linking and boundary detection, thresholding, region oriented segmentation

Representation and Description: Representation schemes, boundary descriptors, regional descriptors. Morphology: Dilation, erosion, opening, closing, Hit-or-Miss Transform, some basic morphological algorithms like pruning, thinning and thickening

Text/References:

1. Gonzalez and Woods. Digital Image Processing, Addison Wesley.
2. Castleman. Digital Image Processing. Prentice Hall.
3. Duda and Hart. Pattern Classification. John Wiley.

Program Elective VII (Semester VIII)

CP-480 Evolving Architectures (3-0-2) 4

Special, and emerging advanced topics in different areas of Computer Engineering will be covered under this course.

Text and References:

1. Research reports and papers from journals.

CP-482 Topics in Advanced Computing (3-0-2) 4

Fault Tolerant Computing: Fault Tolerance Strategies - Fault detection, masking, containment, location, reconfiguration, and recovery. Fault Tolerant Design Techniques - Hardware redundancy, software redundancy, time redundancy, and information redundancy.
Parallel and Distributed Computing: Concepts and issues in parallel and distributed computing.
Concepts and issues in quantum computing, Trusted Computing, Grid Computing
Any other contemporary and relevant issues. GPGPU and Multicore computing.

Text and References:

1. P. Jalote, *Fault Tolerance in Distributed Systems*, Prentice-Hall Inc., 1994
2. D. K. Pradhan (editor), Fault-Tolerant Computing, Theory and Techniques, Prentice-Hall, 1998.
3. Los Alamitos, CA, "Fault-tolerant Software Systems: Techniques and Applications", IEEE Computer Society Press, 1992.
4. Design and Analysis of Fault Tolerant Digital Systems, Barry W. Johnson, Addison Wesley, 1989 (Chapters 1-5).
5. A.K. Somani and N.H. Vaidya, ``Understanding fault-tolerance and reliability," IEEE Computer, vol.30, no.4, pp.45-50, Apr. 1997.
6. Research papers and internet resources.

CP-484 Software Testing and Validation (3-0-2) 4

Basic software testing principles - Software Quality, Software testing, test generation and test management.
Acceptance Testing: User acceptance testing, alpha and beta testing. Verification and Validation, Functional and Non-functional system testing.
Static and dynamic testing, Black-box or functional testing, structural, white box or glass box testing.
Integration testing, component testing.
Software testing tools.

Books/References:

1. Recent papers from conferences and journals.
2. A. P. Mathur, Fundamentals of software testing.

CP-486 Machine Learning (3-0-2) 4

Introduction: Definition of learning systems. Goals and applications of machine learning. Aspects of developing a learning system: training data, concept representation, function approximation.
Inductive Classification: The concept learning task. Concept learning as search through a hypothesis space. General-to-specific ordering of hypotheses. Finding maximally specific hypotheses. Version spaces and the candidate elimination algorithm. Learning conjunctive concepts. The importance of inductive bias.
Decision Tree Learning: Representing concepts as decision trees. Recursive induction of decision trees. Overfitting, noisy data, and pruning.
Ensemble Learning Using committees of multiple hypotheses. Bagging, boosting, and DECORATE. Active learning with ensembles.
Experimental Evaluation of Learning Algorithms: Measuring the accuracy of learned hypotheses. Comparing learning algorithms: cross-validation, learning curves, and statistical hypothesis testing.
Rule Learning: Propositional and First-Order: Translating decision trees into rules. Heuristic rule induction using separate and conquer and information gain. First-order Horn-clause induction (Inductive Logic Programming) and Foil. Learning recursive rules. Inverse resolution.
Artificial Neural Networks: Neurons and biological motivation. Linear threshold units. Perceptrons: representational limitation and gradient descent training. Multilayer networks and back propagation. Hidden layers and constructing intermediate, distributed representations. Overfitting, learning network structure, recurrent networks.
Bayesian Learning: Probability theory and Bayes rule. Naive Bayes learning algorithm. Parameter smoothing. Generative vs. discriminative training. Logistic regression. Bayes nets and Markov nets for representing dependencies.
Instance-Based Learning: Constructing explicit generalizations versus comparing to past specific examples. k-Nearest-neighbor algorithm. Case-based learning.
Clustering and Unsupervised Learning: Learning from unclassified data. Clustering. Hierarchical Agglomerative Clustering. k-means partitional clustering. Expectation maximization (EM) for soft clustering. Semi-supervised learning with EM using labeled and unlabeled data.

Text/References:

1. Bishop, C. (2006) Mitchell, T. M. (1997) Machine Learning. McGraw-Hill
2. Pattern Recognition and Machine Learning. Berlin: Springer-Verlag.
3. Richard O. Duda, Peter E. Hart and David G. Stork. Pattern Classi_cation. Wiley-Interscience,second edition, 2001.
4. Thomas Mitchell. Machine Learning. McGraw Hill Higher Education, First edition, 1997.
5. Stuart Russell and Peter Norvig. Arti_cial Intelligence: A Modern Approach. Prentice Hall,second edition, 2003. (Machine-learning related chapters.)
6. Information Theory, Inference and Learning Algorithms by David MacKay.

CP-488 Modelling and Simulation (3-0-2) 4

Analytical v/s simulation modeling, performance measurement and benchmarking, Workload modeling, random variables, commonly used distributions, Stochastic Processes, Performance evaluation methods, Evaluation Metrics’
Markov chains, Birth and Death Processes, Markov chain models of Computer systems, Steady-state and transient analysis
Queuing models, M/M systems and their steady state analysis, Single server and multi-server queues, open and closed queuing networks
Petri Net based Performance Modeling : Classical Petri Nets, Timed Petri Nets, Discrete Petri Nets,

Modeling multiprocessor systems
Discrete event simulation – Simulation languages, random number generation and testing, model verification and validation, analysis of simulation results, confidence intervals, variance reduction techniques, Case studies of analytical and simulation studies of computer systems

Text/References :

1. Law and Kelton, Simulation Modeling and Analysis, Mcgraw Hill
2. Raj Jain, The Art of Computer System Performance Analysis, John Wiley
3. K.S.Trivedi, Probability and Statistics with Reliability, Queuing and Computer Science Applications, PHI
4. Kant, Introduction to Computer System Performance Evaluation, Mcgraw Hill