Department of Computer Science and Engineering

Curriculum for M Tech in Computer Science and Engineering

Semester 1

	Code	Title	L	T	P/S	C
1	CS 6101	Mathematical Foundations of Computer Science	3	0	2	4
2	CS 6111	Algorithms and Complexity	3	0	2	4
3	CS 6103	Software Systems Lab	1	0	6	4
4		Elective	3	0	2	4
5		Elective	3		0/2	3/4
		Total credits				19/20

Semester 2

	Code	Title	L	T	P/S	Cr
1	CS 6104	Term paper(optional)	0	0	6	3
2		Elective	3	0	2	4
3		Elective	3	0	2	4
4		Elective	3	0	2	4
5		Elective	3		0/2	3/4
6		Elective(Optional)	3		0/2	3/4
		Total credits				16 to 23

Semester 3

	Code	Title	L	T	P/S	Cr
1	CS 7198	Project				8
2		Elective(optional)	3		0/2	3/4
		Total credits				8/11/12

Semester 4

	Code	Title	L	T	P/S	Cr
1	CS 7199	Project	-		-	12
		Total credits				12

Minimum Requirements

1. A Candidate should have earned a total of at least 60 credits, including 20 credits from project work.
2. The number of electives credited by a student can be varied subject to minimum credit requirements for completion of the course.

Credits for elective courses may vary depending on the practical work involved.

LIST OF ELECTIVES

	Code	Title	Credit
1.	CS 6102	Compiler Design	4
2.	CS 6112	Operating System Design	4
3.	CS 6121	Computability Theory	3
4.	CS 6122	Computer Architecture	4
5.	CS 6123	Database Design	4
6	CS 6124	Topics in Programming Languages	4
7	CS 6125	Computer Networking	4
8	CS 6131	Logic and Computation	3
9	CS 6132	Topics in Algorithms	4
10.	CS 6133	Game Theory	4
11	CS 6134	Quantum Computation	3
12	CS 6135	Logic for Computer Science	4
13	CS 6136	Topics in Combinatorial Algorithms	4
14	CS 6141	Distributed Computing	4
15	CS 6142	Topics in Computer Architecture	4
16	CS 6143	Trends in Middleware Systems	4
17	CS 6144	Multicore Systems	4
18	CS 6151	Software Engineering	4
19	CS 6152	Object Oriented Modeling and Design	4
20	CS 6154	Topics in Database Design	4
21	CS 6161	Embedded Systems and Applications	4
22.	CS 6171	Natural Language Processing	4
23.	CS 6172	Computational Intelligence	4
24.	CS 6173	Image Processing	4
25.	CS 6174	Pattern Recognition	4
26.	CS 6181	Bioinformatics	4
27	CS 6201	Cryptography	4
28.	CS 6211	Formal Methods in Secure Computing	4
29	CS 6212	Network Security	4
30.	CS 6213	Foundations of Information Security	4
31.	CS 6214	Advanced Topics in Information Security	4
32.	CS 6231	Theoretical aspects of cryptographic algorithms	3
33.	CS 6232	Cryptocomplexity	4
34.	CS 6233	Information Theory and Coding	4
35	CS 6261	Perimeter Security	4
36	CS 6271	Data Compression	4
37.	CS 6282	Pragmatics of Information Security	4
38.	CS 6283	Computer Laws and Ethics	3
39.	CS 6284	Security Policies and Assurance	3
40.	CS 6285	Information Security Management	4
41.	CS 6286	Metrics for Information Security Assessment	4
42	MA 8152	Fuzzy Set Theory and Applications	3
43	MA 7156	Advanced Topics in Graph Theory	3

BRIEF SYLLABUS

CS 6101: Mathematical Foundations of Computer Science

Prerequisite: Discrete Computational Structures

L	T	P	C
3	0	2	4

Total Hours: 42 + 28

Divisibility, prime numbers, Congruences, Fermat's theorem, Chinese remainder theorem,
Groups and subgroups, Lagrange's theorem, rings, finite fields, polynomial arithmetic, quadratic residues, discrete logarithms, elliptic curve arithmetic. Fundamental principles of counting, derangements, partitions, partial order, lattices and Boolean algebra, generating functions, solution of recurrences. Graphs, Euler's formula, applications of Kuratowski's theorem, graph colouring, chromatic polynomials, weighted trees, spanning trees, max-flow.

CS 6111: Algorithms and Complexity

Prerequisite: Design and Analysis of Algorithms

L	T	P	C
3	0	2	4

Total Hours: 42 + 28

Analysis: Master's theorem and its proof - Amortized analysis - Advanced Data Structures
Graph Algorithms and complexity: Matroid Theory, All-Pairs Shortest Paths, Maximum Flow and Bipartite Matching.Randomized Algorithms **:** Finger Printing, Pattern Matching, Graph Problems, Algebraic Methods, Probabilistic Primality Testing, De-RandomizationComplexity classes - Cook's theorem. Approximation algorithms Probabilistic Complexity Classes, Probabilistic Proof Theory and Certificates.

CS 6102: Compiler Design

Prerequisite: Compiler Construction

L	T	P	C
3	0	2	4

Total Hours: 42+28

Review of compiler phases –Control Flow Analysis – Data Flow Analysis – Dependency analysis – Optimizations – Redundancy Elimination – Loop Optimizations –Value Numbering – Static Single Assignment Form: Register Allocation –Instruction Selection –Machine Code Generation

CS 6103: Software Systems Lab

Prerequisite: Programming in C/C++ or Java

L	T	P	C
1	0	6	4

Total Hours: 14 + 84

General purpose programming toolsWeb programming. Tools for good software development process. Make/gmake, source code control systems, debuggers and memory allocation debuggers. Integrated Development Environments Scripting languages Tools for text processing.Exposure to document creation tools.

CS 6104: Term Paper

Prerequisite: Nil

L	T	P	C
0	0	6	3

Total Hours: 0 + 84

The aim of this course is to introduce the student to research, and to acquaint him/her with the process of presenting his work through seminars, technical reports and research papers.

CS 6112: Operating System Design

Prerequisite: Operating Systems

L	T	P	C
3	0	2	4

Total Hours: 42+28

Operating System design techniques. Implementing processes system call interrupt handling, Parallel systems-race conditions with shared processes, atomic actions, multiprocessor OS, threads. Interprocess communication patterns IPC pattern: mutual exclusion, Deadlocks synchronization, semaphores. Memory management, dynamic memory allocation, multiprogramming issues, Virtual memory- page replacement- thrashing and load control, sharing memory, multiplexing and late binding. I/O devices, controllers, I/O subsystems generalized disk device drivers, disk caching. File systems organization. Booting the OS. integrated scheduling, real-time OS, Naming.

CS 6121: Computability Theory

Prerequisite: Theory of Computation

L	T	P	C
3	0	0	3

Total Hours: 42

Review of Induction and Diagonalization - Myhill-Nerode Theorem, Turing Acceptable, Decidable and Enumerable languages**.** Closure Properties of RE and R sets - Undecidability – Reductions – RE Completeness – Non-RE languages - Rice Theorems. Relations between deterministic and Non-Deterministic time and Space complexity classes – Hierarchy Theorems, -Savitch's Theorem - Immerman Szelepscenyi Theorem. NP – Completeness – Cook's Theorem NLCompleteness.

CS 6122: Computer Architecture

Prerequisite: Computer Organization

L	T	P	C
3	0	2	4

Total Hours: 42+28

Performance evaluation, Processor architecture, issues in pipelined processor implementation. Instruction level parallelism, out-of-order Instruction issue, speculative execution Instruction and data cache organizations, virtual memory.Multiple processor systems, Interconnection networks, shared memory system, memory models, cache coherence.

CS 6123: Database Design

Prerequisite: Database Management Systems

L	T	P	C
3	0	2	4

Total Hours: 42+28

Database System concepts and applications, Data modeling using Entity-Relationship model, Record Storage and File organization**.**The Relational Data Model, Relational constraints and the Relational Algebra, SQL, ER to Relational mapping, .Database Design Theory and Methodology- Functional Dependencies and Normalization for Relational Databases, Relational Database design algorithms, Practical Database Design and Tuning**.** Object Oriented Database concepts, Object Relational and Extended Relational Database Systems, Data Warehousing and Data Mining, Emerging Database Technologies and Applications.

CS 6124: Topics in Programming Languages

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Formal Semantics.Untyped Lambda Calculus, Simply Typed Lambda Calculus.Extensions to Simply Types Lambda Calculus: Basic Types, Derived Forms, Bindings, Pairs, Tuples,Records, Sums, Variants, General Recursion. Sub-typing, Recursive Types, Polymorphism.

CS 6125 : Computer Networking

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Computer networks and Internet, the network edge, the network core, network access, delay and loss, protocol layers and services, Application layer protocols, socket programming, content distribution. Transport layer services, UDP and TCP, congestion control, Network layer services, routing, IP, routing in Internet, router, IPV6, multicast routing, mobility. Link layer services, error detection and correction, multiple access protocols, ARP, Ethernet, hubs, bridges, switches, wireless links, PPP, ATM. Multimedia networking, streaming stored audio and video, real-time protocols, security, Cryptography, authentication, integrity, key distribution, network management.

CS 6131: Logic and Computation

Prerequisite: Logic for Computer Science

L	T	P	C
3	0	0	3

Total Hours: 42

Temporal Logic based verification, Buchi automata, Linear Temporal Logic (LTL), Modelling systems Computation Tree Logic (CTL), Symbolic model checking Tools: Spin**,** SMV Verification of infinite-state systems, Verification of real-time systems, Timed automata, Modelling real-time systems, Tools: Uppaal, Kronos Verification of pushdown systems, Verification of security protocols

CS 6132: Topics in Algorithms

Prerequisite: Design and Analysis of Algorithms

L	T	P	C
3	0	2	4

Total Hours: 42+28

Probability, Expectations, Tail Bounds, Chernoff Bound, Markov Chains and Random Walks, MartingalesFinger Printing, Pattern Matching, Graph Problems, Algebraic Methods,
Probabilistic Primality Testing, De-RandomizationProbabilistic Complexity Classes, Probabilistic Proof Theory and Certificates,Interactive and Zero Knowledge Proof Systems, Arthur Merlin Games.Definition of Randomness, Unsolvability results, Chatin's Proof for Gödel's Theorem.

CS 6133: Game Theory

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Introduction to Non Co-operative Game Theory: Extensive Form Games, Strategic Form Games, Pure Strategy Nash Equilibrium Nonco-operative Game Theory (in detail), Mixed Strategies, Existence of Nash Equilibrium, Computation of Nash Equilibrium, Bayesian Games Mechanism Design Dominant Strategy Implementation of Mechanisms, Vickrey-Clorke-Groves Mechanisms, Bayesian Implementation of Mechanisms, Revenue Equivalence Theorem, Design of Optimal Mechanisms Cooperative Game Theory, Correlated Strategies, Correlated Equilibria, The Two Person Bargaining Problem, Games in Coalitional Form, The Core Shapley Value, Other Solution Concepts for Co-operative Games

CS 6134: Quantum Computation

Prerequisites: Theory of Computation

L	T	P	C
3	0	0	3

Total Hours: 42

Finite Dimensional Hilbert Spaces – Tensor Products and Operators on Hilbert Space – Hermitian and Trace Operators - Basic Quantum Mechanics necessary for the course. Quantum Gates and operators and Measurement – Quantum Computational Model – Quantum Complexity – Schemes for Physical realization (Only peripheral treatment expected). Shor's Algorithm – Application to Integer Factorization – Grover's Algorithm – Quantum Complexity Classes and their relationship with classical complexity classes**.** Quantum Noise – Introduction to the theory of Quantum Error Correction – Quantum Hamming Bound – Coding Schemes – Calderbank-Shor-Steane codes –Stabilizer Codes.

CS 6135: Logic for Computer Science

Prerequisite: Theory of Computation

L	T	P	C
3	0	2	4

Total Hours: 42+28

Review of undecidability and Complexity classes, Formal Systems, Completeness Theorem, Compactness, NP-Completeness of Satisfiability**.** Syntax and semantics, Herbrand's Expansion Theorem, Proof Systems, Completeness and Compactness, Undecidability of Satisfiability. First order Axiomatization of number theory, Godel's Incompleteness Theorem, Limitations of first order logic. Bucchi's Theorem, Logical Characterization for Complexity classes, Fagin'sTheorem.

CS 6136: Topics in Combinatorial Algorithms

Prerequisite: Design and Analysis of Algorithms

L	T	P	C
3	0	2	4

Total Hours: 42+28

Primal dual theory of linear programming, and application to flows, cuts.Application of primal dual theoty to matching and related combinatorial optimization problems**.** Randomization techniques,Approximation algorithms and non-approximability.

CS 6141: Distributed Computing

Prerequisite: Operating Systems

L	T	P	C
3	0	2	4

Total Hours: 42+28

Characterization of Distributed Systems, System Models, Networking and Internetworking, Inter Process communication Distributed Objects and Remote Invocation, RPC, Processes and threads, Security, Logical clocks, clocks on different dimensions, Distributed File Systems Name Services and Domain Name System, Synchronizing physical clocks, Distributed Mutual Exclusion, Elections Transactions and Concurrency Control, Distributed Transactions, Distributed Deadlocks, Transaction Recovery, Fault-tolerant Services, self stabilization, Distributed Shared Memory, Drinking philosophers problem, Global state detection, Termination detection, consensus and synchrony.

CS 6142 : Topics in Computer Architecture

Prerequisite: Computer Architecture

L	T	P	C
3	0	2	4

Total Hours: 42+28

Hardware and software techniques for ILP extraction, speculative execution, studies on ILP.Concept, Studies and AnalysisIntroduction – methods, ADLs, traces, dynamic compilation. Concept, methodologies and analysis. Speculative multithreading. Multicore systems. Processor Based Security. Virtualization.

CS 6143: Trends in Middleware Systems

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Publish/Subscribe matching algorithm , Processing of XML data over Xpath expressions , Processing of XML data over XQuery queries , Regular expression processing Composite event processing, Content-based routing in distributed broker networks, Distributed Case studies- REBECA, HERMES, Gryphon.

CS 6144: Multicore Systems

Prerequisite: Compiler Design, Operating Systems and Computer Architecture

L	T	P	C
3	0	2	4

Total Hours: 42+28

Multi-core architectures, issues, Developing programs for multi-core architectures, Program optimizations techniques. Pipeline Design issues, Memory hierarchy and Memory system issues in multicore system. shared L2 vs. tiled CMP; core complexity; power/ performance; Multicore, Multiprocessor, Super speculative & VLIW architectures – a comparative study. Analysis & Optimization for multicore architectures, solving data dependence equations (integer linear programming problem); Loop optimizations; Memory hierarchy issues in code optimization; Multicore programming methodologies and measuring performance, Case studies.

CS 6151: Software Engineering

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

The Software Process: Life cycle models, from specification to maintenance. Critical systems, Requirements Engineering: Functional and non-functional requirements,. Software Design: Architectural, Object Oriented, Real-time, and Reusable Designs. UML Software Project Management: Managing Process, Reliability Formal specification and verification of programs and software. Project Delays - The symptoms, reasons, and solutions. Performance concerns

CS 6152 :Object Oriented Modeling & Design

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Structural Modeling: Object Oriented Fundamentals, Basic structural Modeling, UML Model , Class Diagrams, Object Diagrams, Packages and Interfaces, Case Studies**.** Behavioral and architectural Modeling: Use Case Diagrams, Interaction Diagrams, State Chart Diagrams, Collaborations, Design Patterns, Component Diagrams, Deployment Diagrams, Case Studies Object oriented Testing Methodologies: Implications of Inheritance on Testing, State Based Testing, Adequacy and Coverage, Scenario Based Testing, Testing Workflow, Case Studies , Object Oriented Metrics Components: Abuses of inheritance, danger of polymorphism, mix-in classes, rings of operations, class cohesion and support of states and behavior,components and objects, design of a component, lightweight and heavyweight components, advantages and disadvantages of using components.

CS 6154: Topics in Database Design

Prerequisite: Database Design

L	T	P	C
3	0	2	4

Total Hours: 42+28

Parallel and Distributed Databases: Parallel Query Optimization, Distributed DBMS Architectures, Distributed Query Processing, Distributed Concurrency Control, Distributed recovery. Internet Databases and Data Mining, Ranked Keyword searches, Data Mining, Clustering, Similarity Search over Sequences**.** Object Oriented Database Systems: User Defined ADTs, Objec Identity and Reference types, Database Design for ORDBMS, OODBMS, Spatial and Deductive Databases: Spatial and Temporal Databases, Temporal Logic, Spatial Indexes, Introduction to Recursive Queries, Introduction to Mobile Databases, Main Memory and Multimedia Databases

CS 6161: Embedded Systems and Applications

Prerequisite: Basic courses in digital hardware, algorithms, data structures, elementary calculus, and probability

L	T	P	C
3	0	2	4

Total Hours: 42+28

Introduction to embedded systems: classification, characteristics and requirements. Timing and Clocks in Embedded Systems.Task modeling and management. Real-time operating system issues. Signals: frequency spectrum, and sampling, digitization (ADC, DAC), signal conditioning and processing. Modeling and characterization of embedded computing systems. Communication strategies for embedded systems: encoding, and flow control. Fault Tolerance. Formal Verification

CS 6173: Image Processing

Prerequisite: Nil

L	T	P	C
4	0	0	4

Total Hours: 56

Digital image fundamentals - elements of visual perception - sampling and quantization - image geometry - image transforms - Fourier transform – discrete Fourier transform - 2d-fourier transform (DFT)- hotelling transform Image enhancement - - frequency domain -image restoration - diagonalization of circulant and block circulant matrices - inverse filtering - least mean square filter Image compression - elements of information theory - lossy compression - image compression standards Image reconstruction from projections - parallel beam and fan beam projection - Fourier slice theorem - filtered back projection algorithms

CS 6174: Pattern Recognition

Prerequisite: Nil

L	T	P	C
4	0	0	4

Total Hours: 56

Introduction approaches, features and feature extraction, learning. Bayes Decision theory- introduction, continuous case, classifiers, discriminant functions, and decision surfaces. Error probabilities and integrals, Bayes Decision theory Discrete case. Parameter estimation and supervised learninggeneral bayesian learning. Nonparametric technique- density estimation, parzen windows nearest- neighbor rule, k-nearest neighbor rule. Linear discriminant functions- linear programming procedures. Multiplayer neural networks- Backpropagation algorithm, error surfaces, back propagation as feature mapping, Supervised learning and clustering- Mixture densities and identifiably, maximum likelihood estimates, Hierarchical clustering, low dimensional representation of multidimensional map

CS 6171: Natural Language Processing

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Introduction to Natural Language Understanding Applications of NLP, Evaluating Language Understanding Systems, Different levels of Language Analysis, Organization of Natural language Understanding Systems, Grammars and Parsing: Transition Network Grammars, Top-Down Chart Parsing. Feature Systems and Augmented Grammars Morphological Analysis and the Lexicon, Parsing with Features, Augmented Transition Networks. Grammars for Natural Language: Auxiliary Verbs and Verb Phrases, Movement Phenomenon in Language, Handling questions in Context-Free Grammars, Hold mechanisms in ATNs. Human preferences in Parsing, Encoding uncertainty, Deterministic Parser Ambiguity Resolution: Statistical Methods, Estimating Probabilities, Part-of-Speech tagging, Obtaining Lexical Probabilities, Probabilistic Context-Free Grammars, Best First Parsing. Semantics and Logical Form, Word senses and Ambiguity, Encoding Ambiguity in Logical Form.

CS 6172: Computational Intelligence

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Artificial Intelligence: Structures and Strategies for state space search- Heuristic Search- Best First Search, A* Algorithm, AO* Algorithm, Constraint Satisfaction, Using heuristics in games- Minimax Search, Alpha Beta Procedure. Knowledge representation - Propositional calculus, Predicate Calculus, Theorem proving by Resolution, AI Representational Schemes- Semantic Nets, Conceptual Dependency, Scripts, Frames, Introduction to Agent based problem solving. Machine Learning- Symbol based and Connectionist, Social and Emergent models of learning, The Genetic Algorithm- Genetic Programming, Overview of Expert System Technology- Rule based Expert Systems, Introduction to Natural Language Processing. Languages and Programming Techniques for AI

CS 6181: Bioinformatics

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: 42+28

Introduction to Molecular biology, Gene structure and information content, Molecular biology tools, Algorithms for sequence alignment, Sequence databases and tools. Molecular Phylogenetics, Phylogenetic trees, Algorithms for Phylogenetic tree construction, Introduction to Perl programming for Bioinformatics. Introduction to Protein structure, Algorithms for Protein structure prediction, Gene expression analysis, Micro Arrays, Pathway analysis. Pattern Matching algorithms, Bio-data analysis, Data Mining in Bioinformatics, Algorithms and data structures for efficient analysis of biological data.

MA 8152: Fuzzy Set Theory and Applications

Prerequisite: Nil

L	T	P	C
3	0	0	3

Total Hours: 42

The notion of fuzzy sets –basic concepts of fuzzy sets – membership functions – methods of generating membership functions – defuzzification methods- operations on fuzzy sets – combinations of operations – General aggregation operations, Fuzzy numbers- arithmetic operations on intervals- arithmetic operations on fuzzy numbers- fuzzy equations- crisp and fuzzy relations – binary relations – equivalence and similarity relations – compatibility or tolerance relations, Fuzzy measures – belief and plausibility measures – probability measures – possibility and necessity measures – possibility distribution - relationship among classes of fuzzy measures, Fuzzy Logic and Applications **,** Classical logic : an overview – fuzzy logic – approximate reasoning - other forms of implication operations - other forms of the composition operations – fuzzy decision making – fuzzy logic in database and information systems - fuzzy pattern recognition – fuzzy control systems.

MA 7156: Advanced Topics in Graph Theory

Prerequisite: Nil

L	T	P	C
3	0	0	3

Total Hours: 42

Review of basics in graphs - -Trees- Blocks- Matrices-Operations on graphs,Vertex Connectivity and edge connectivity – n-connected graphs- Menger's Theorem, Euler graphs-Hamiltonian Graphs-Planar and Nonplanar graphs, Metric in graph: Centre, Median, eccentric vertex, Eccentric graph, boundary vertex, complete vertex, interior vertex, Convexity: Closure Invariants-gin(G) –gn (G)-Hull number- Geodetic Graphs-Distance Hereditary Graphs, Graphs and groups-- Symmetric Graphs - Distance Symmetry-Distance transitive graphs-distance regular graphs, Degree sequence, Eccentric Sequence - Distance Sequences - The Distance Distribution, Mean distance, Matchings :Maximum matching-Perfect matching-Matching in bipartite graphs, Factorization :Coverings and independence-1-fcatorization-2-factorization-Arboricity, Dominating set-Domination number-total dominating set –total domination number, Digraphs and connectedness-Tournaments- directed trees-binary trees- weighted trees and prefix codes , Networks: Flows-cuts- The Max-Flow Min-Cut Theorem, Graph Algorithms: Polynomial Algorithms and NP completeness ,Complexity, Search algorithms**,** Shortest path algorithms.

Detailed Syllabus

CS 6101: Mathematical Foundations of Computer Science

Prerequisite: Discrete Computational Structures

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (10 Hours)
Divisibility, gcd, prime numbers, fundamental theorem of arithmetic, Congruences, Fermat's theorem, Euler function, primality testing, solution of congruences, Chinese remainder theorem, Wilson's theorem.

Module 2: (10 Hours)
Groups and subgroups, homomorphism theorems, cosets and normal subgroups, Lagrange's theorem, rings, finite fields, polynomial arithmetic, quadratic residues, reciprocity, discrete logarithms, elliptic curve arithmetic.

Module 3: (10 Hours)
Fundamental principles of counting, pigeonhole principle, countable and uncountable sets, principle of inclusion and exclusion, derangements, equivalence relations and partitions, partial order, lattices and Boolean algebra, generating functions, recurrence relations, solution of recurrences.

Module 4: (12 Hours)
Graphs, Euler tours, planar graphs, Hamiltonian graphs, Euler's formula, applications of Kuratowski's theorem, graph colouring, chromatic polynomials, trees, weighted trees, shortest path algorithms, spanning trees, the max-flow min-cut theorem.

References

1. Niven, H.S. Zuckerman and Montgomery, An Introduction to the Theory of Numbers, 3/e, John Wiley and Sons, New York, 1992.
2. R. P. Grimaldi, Discrete and Combinatorial Mathematics: An Applied Introduction,3/e, Addison-Wesley, New Delhi, 1994.
3. B. Kolman and R.C. Busby, Discrete Mathematical Structures for Computer Science, PHI, New Delhi, 1994.
4. J. Clark and D. A. Holton, A First Look at Graph Theory, Allied Publishers (World Scientific), New Delhi, 1991.
5. C. L. Liu, Elements of Discrete Mathematics, McGraw Hill, 2/e, Singapore, 1985.

CS 6102: Compiler Design

Prerequisite: Compiler Construction

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (12 Hours)
Review of compiler phases –Symbol Table Structure – Intermediate Representations. Control Flow Analysis: Basic Blocks and CFG, Dominators and Loops.

Module 2: (12 Hours)
Data Flow Analysis: Reaching Definitions, Available Expressions, and Live Variable Analysis. Optimizations: Redundancy Elimination – Loop Optimizations –Value Numbering.

Module 3: (10 Hours)
Static Single Assignment Form (SSA): SSA Construction – Optimizations on SSA Form. Register Allocation – Graph Colouring Algorithm.

Module 4: (10 Hours)
Machine Code Generation: Instruction Selection - Maximal munch and Dynamic programming Algorithm. Code Generation – Target Machine – Code Generation for Run- time Stage Management. Code Generation Algorithms.

References:

1. Aho A.V., Lam M.S., Sethi R., and Ullman J.D., Compilers: Principles, Techniques, and Tools. Pearson Education, 2007.

2. Steven Muchnick., Advanced Compiler Implementation. Morgan Kauffman Publishers, 1997.

3. Appel A.W and Palsberg J., Modern Compiler Implementation in Java. Cambridge University Press, 2002.

4. Srikant Y.N and Shankar P., The Compiler Design Handbook: Optimization and Machine Code Generation. CRC Press, 2003.

CS 6103: Software Systems lab

Prerequisite: Programming in C/C++ or Java

L	T	P	C
1	0	6	4

Total Hours (14+84)

Module 1: (4 hours)
General purpose programming tools(e.g. Java, C++, use of GUI tools like QT), Web programming tools(e.g. HTML, Java with applets/servelets/JSP/J2EE, CGI, Perl)

Module 2: (3 hours)
Tools for good software development process. Make/gmake, source code control systems, debuggers and memory allocation debuggers, Introduction to Integrated Development Environments (e.g. Visual Studio or Netbeans)

Module 3: (4 hours)
Scripting languages (e.g. Python, Perl). Tools for text processing (e.g. AWK, Python, Lex, Yacc)

Module 4: (3 hours)
Exposure to document creation tools (e.g. Latex, dia, xfig), plotting tools (e.g. gnuplot, Excel),

The contents may be adapted to software practices and trends at the time of offering the course. Hence the contents in parenthesis are simply examples and not strict requirements.

References

1. Larry Wall,Tom Christiansen & Randal L. Schwartz . Programming Perl. O Reilly Media, Third Edition, 2000.
2. Scott Guelich, Shishir Gundavaram and Gunther Birznieks, CGI Programming with Perl. O'Reilly Media, Third Edition, June 2000.
3. Mark Summerfield, Programming in Python 3. Addison Wesley Professional, Second Edition, November 2009.
4. Mark Summerfield, Rapid GUI Programming with Python and Qt**,** Prentice Hall, 2009.
5. GNU Operating System. links from http://www.gnu.org/software/ last accessed 28/3/2010
6. Wikibook Contributors LaTeX, Wikibooks, 2006 available at http://upload.wikimedia.org/wikibooks/en/2/2d/LaTeX.pdf last accessed 28/4/2010
7. John Levine, flex & bison, O'Reilly Media, 1st Edition, 2009.
8. Bruce Eckel, *Thinking in Java*, 3/ed, Prentice Hall, 2002, Available online at www.bruceeckel.com last accessed 28/3/2010
9. Bruce Eckel, *Thinking in C++*, 2/ed. Vol I and II, Prentice Hall. 2003, Available online at www.bruceeckel.com last accessed 28/3/2010

CS 6104: Term Paper

Prerequisite: Nil

L	T	P	C
0	0	6	3

Total Hours: (0+84)

The aim of this course is to introduce the student to research, and to acquaint him with the process of presenting his work through seminars and technical reports.

The student is expected to do an extensive literature survey and analysis in an area related to computer science, chosen by him, under the supervision of a faculty member from the department. The study should preferably result in design ideas, designs, algorithms, theoretical contributions in the form of theorems and proofs, new methods of proof, new techniques or heuristics with analytical studies, implementations and analysis of results.

The student should give two seminars on his work, one in the middle of the semester and the other at the end of the semester, and submit a technical report.

References

Articles from ACM/IEEE Journals/Conference Proceedings and equivalent documents, standard textbooks and web based material, approved by the supervisor.

CS 6111: Algorithms and Complexity

Prerequisite: Design and Analysis of Algorithms

L	T	P	C
3	0	2	4

Total Hours (42+28)

Module 1: (11 hours)
Analysis: RAM model – Notations, Recurrence analysis - Master's theorem and its proof - Amortized analysis - Advanced Data Structures: B-Trees, Binomial Heaps, Fibonacci Heaps, Disjoint Sets, Union by Rank and Path Compression

Module 2: (11 hours)
Graph Algorithms and complexity: Matroid Theory, All-Pairs Shortest Paths, Maximum Flow and Bipartite Matching.

Module 3: (10 hours)
Randomized Algorithms **:** Finger Printing, Pattern Matching, Graph Problems, Algebraic Methods, Probabilistic Primality Testing, De-Randomization

Module 4: (10 hrs)
Complexity classes - NP-Hard and NP-complete Problems - Cook's theorem NP completeness reductions. Approximation algorithms – Polynomial Time and Fully Polynomial time Approximation Schemes.
Probabilistic Complexity Classes, Probabilistic Proof Theory and Certificates.

References

1. Dexter Kozen, The Design and Analysis of Algorithms, Springer, 1992.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, Introduction to Algorithms, Prentice Hall India, 1990.
3. S. Basse, Computer Algorithms: Introduction to Design and Analysis, Addison Wesley, 1998.
4. U. Manber, Introduction to Algorithms: A creative approach, Addison Wesley, 1989.
5. V. Aho, J. E. Hopcraft, J. D. Ullman, The design and Analysis of Computer Algorithms, Addison Wesley, 1974.
6. R. Motwani and P. Raghavan, Randomized Algorithms, Cambrdige University Press, 1995.
7. C. H. Papadimitriou, Computational Complexity, Addison Wesley, 1994
8. Leonard Adleman, Two theorems on random polynomial time. In Proceedings of the 19th IEEE Symposium on Foundations of Computer Science, pages 75–83, 1978.
9. J. Gill, Computational complexity of probabilistic Turing machines. SIAM Journal of Computing, 6:675–695, 1977.
10. C. Lautemann, BPP and the Polynomial Hierarchy. Information Processing Letters, 17:215–217, 1983.
11. M. Sipser, A complexity theoretic appraoch to randomness. In Proceedings of the 15th ACM Symposium on Theory of Computing, pages 330–335, 1983.
12. L.G. Valiant and V.V. Vazirani, NP is as easy as detecting unique solutions. Theoretical Computer Science, 47:85–93, 1986.

CS 6112: Operating System Design

Prerequisite: Operating Systems

L	T	P	C
3	0	2	4

Total Hours (42+28)

Module1: (11 Hours)

Introduction- Introduction, Hardware interface, Operating system interface. design problems, Operating System design techniques. Implementing processes – The system call interface, system initialization, process switching, system call interrupt handling, program error interrupts, disk driver system, implementing waiting, flow of control through OS, signaling and interrupts, event table managers, process implementation. Parallel systems- Parallel hardware, OS for two processor systems, race conditions with shared processes, atomic actions, multiprocessor OS, threads.

Module 2: (11 Hours)

Interprocess communication patterns- competing and co-operating, problems, race conditions and atomic actions, new message passing system calls, IPC pattern: mutual exclusion, signalling and rendezvous models, producer-consumer and client server models. Deadlocks- Conditions for deadlock, dealing with deadlocks, two-phase locking, message variations, synchronization, semaphores. Design techniques- some example design techniques. Memory management- levels of memory management, linking and loading process, memory management design, dynamic memory allocation, keeping track allocation of blocks, multiprogramming issues, memory protection, memory management system calls.

Module 3: (10 Hours)

Virtual memory- Fragmentation and compaction, dealing with fragmentation- paging, swapping, overlay, page replacement- global and local page replacement algorithms, thrashing and load control, dealing with large page tables, sharing memory. Design techniques- examples of multiplexing and late binding.

I/O devices- devices and controllers, terminal devices, communication devices, disk devices, disk controllers, SCSI interfaces, tape devices, CD devices. I/O subsystems- I/O system software, disk device driver access strategies, modelling disks, unification of files and device, generalized disk device drivers, disk caching. File systems- File abstraction, naming, filesystem objects and operations.

Module 4: (10 Hours)

File system organization- organization, file descriptors, locating file blockson disks, implementation of logical to physical block mapping, file sizes, Booting the OS, file system reliability, file security and protection. Resource management and protection- resources in an OS, resource management issues, types of resources, integrated scheduling, queuing models of scheduling, real-time OS, protection of resources, user authentication, mechanism for protecting hardware resources, representation of protection information, mechanisms for software protection, Design techniques- Caching, hierarchical names and naming of objects.

References

1. Charles Crowley, Operating systems - A Design Oriented Approach, Tata McGraw-Hill Edition, New Delhi, 1998.
2. Silberschatz and Galvin, Operating System Concepts, Addison Wesley, 1998.
3. Tanenbaum Andrew S, Modern Operating System , Eaglewood Cliffs, NJ: Prentice Hall, 1992
4. Gary J.Nutt, Operating systems- A Modern Perspective, 2nd edition, Addison Wesley, 2000.
5. Stallings William, Operating systems- Internals and Design Principles, 4th Edn, PHI, 2002

CS 6121: Computability Theory

Prerequisite: Theory of Computation

L	T	P	C
3	0	0	3

Total Hours: (42)

)Module 1: (10 Hours)
Automata Theory**:** Review of Induction and Diagonalization - Finite Automata – Myhill-Nerode Theorem, Pumping Lemma. Turing Machines – Turing Acceptable, Decidable and Enumerable languages**.**

Module 2: (10 hours)
Computability:Closure Properties of RE and R sets - Undecidability – Reductions – RE Completeness – Non-RE languages - Rice Theorems.

Module 3: (11 hours)
Introduction to Complexity: Time and Space complexity classes – Relations between deterministic and Non-Deterministic time and Space complexity classes – Hierarchy Theorems, - Savitch's Theorem - Immerman Szelepscenyi Theorem.

Module 4: (11 hours)
Machine Independent Complexity Theory NP – Completeness – Cook's Theorem – Reductions – PSPACE completeness, NLCompleteness.

References

1. Sipser, M. Introduction to the Theory of Computation, PWS Publishing Company, 1997
2. Papadimitriou, C. H. Computational Complexity, Addison Wesley,
1994
3. Hopcroft, J. E. and Ullman, J. D. Introduction to Automata Theory, Languages and Computation, Addison Wesley, 1979
4. Jon Barwise and John Etchemendy, Turing's World 3.0, Center for the study of Language and Information, 1995.

CS 6122: Computer Architecture

Prerequisite: Computer Organization

L	T	P	C
3	0	2	4

Total Hours (42+28)

Module 1: (9 Hours)
Performance evaluation, Processor architecture, pipelining, pipeline hazards, issues in pipelined processor implementation.

Module 2: (12 Hours)
Instruction level parallelism, hardware and compiler support for branch prediction, out-of-order Instruction issue, speculative execution and other techniques for high-performance

Module 3: (9 Hours)
Instruction and data cache organizations, multilevel caches, parallel memory systems, Support for virtual memory.

Module 4: (12 Hours)
Multiple processor systems, Interconnection networks, shared memory system, memory models, cache coherence.

References

1. Hennessy J. L., D. Patterson, Computer Architecture – A quantitative Approach, Morgan Koffman (4/e), 2007
2. John Paul Shen, Mikko Lipasti. Modern Processor Design – Fundamentals of Superscalar Processors. McGraw Hill International Edition, 2005.
3. Dezso Sima, Terence Fountain, Peter Kacsuk. Advanced Computer Architecture – A Design Space Approach, Addison Wesley, 2000.
Current Litrature

CS 6123: Database Design

Prerequisite: Database Management Systems

L	T	P	C
3	0	2	4

Total Hours (42+28)

Module 1 (11 hours)
Database System concepts and applications, Data modeling using Entity-Relationship model, Record Storage and File organization**.**

Module 2 (11 hours)
The Relational Data Model, Relational constraints and the Relational Algebra, SQL, ER to Relational mapping, Examples of RDBMS.

Module 3 (11 hours)
Database Design Theory and Methodology- Functional Dependencies and Normalization for Relational Databases, Relational Database design algorithms, Practical Database Design and Tuning**.**

Module 4 (10 hours)
Object Oriented Database concepts, Object Relational and Extended Relational Database Systems, Data Warehousing and Data Mining, Emerging Database Technologies and Applications.

References

1. Elmasri, Navathe. Fundamentals of Database Systems, Third Edition, Pearson Education, 2000.
2. T. Connolly, C. Begg, Database Systems, 3/e, Pearson Education, 2003.
3. Silberschatz A., Korth H. F., & Sudarshan S., Database System Concepts, Tata McGraw Hill, 2003
4. Ullman J. D., Principles of Database Systems, Galgotia Publications,1996.

CS 6124: Topics in Programming Languages

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (8 hours)
Introduction to Programming Languages. Untyped Arithmetic Expressions: Syntax and Semantics - Properties of the language of Untyped Arithmetic Expressions.

Module 2: (12 hours)
Untyped Lambda Calculus: Syntax, Operational Semantics, Evaluation strategies – Programming in Lambda Calculus. Typed arithmetic Expressions: Typing relation – Type safety.

Module 3: (10 hours)
Simply Typed lambda Calculus: Typing relation – Properties of the Language – Type safety. Extensions: Basic Types, Derived Forms, Let Bindings.

Module 4: (12 hours)
Extensions to Lambda Calculus: Pairs, Tuples, Records, Sums, Variants, References, Exceptions. Subtyping, Recursive Types, Polymorphism.

References:

1. Benjamin C Pierce. Types and Programming Languages. MIT Press, 2002.
2. Luca Cardelli. Type Systems. In Allen B Tucker (Ed.), Handbook of Computer Science and Engineering. CRC Press, 1996.
3. Michael L Scott. Programming Languages Pragmatics Elsevier, 2004.

CS 6125 : Computer Networking

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1 (9 Hrs)
Computer networks and Internet, the network edge, the network core, network access, delay and loss, protocol layers and services, Application layer protocols, socket programming, content distribution.

Module 2 (13 Hrs)
Transport layer services, UDP and TCP, congestion control, Network layer services, routing, IP, routing in Internet, router, IPV6, multicast routing, mobility.

Module 3 (9 Hrs)
Link layer services, error detection and correction, multiple access protocols, ARP, Ethernet, hubs, bridges, switches, wireless links, PPP, ATM.

Module 4 (11Hrs)
Multimedia networking, streaming stored audio and video, real-time protocols, security, Cryptography, authentication, integrity, key distribution, network management.

References

1. J. F. Kurose and K . W. Ross, Computer Networking: A Top-Down Approach Featuring Internet, 3/e, Perason Education, 2005.
2. Peterson L.L. & Davie B.S., Computer Networks, A systems approach, 3/E, Harcourt Asia, 2003.
3. Keshav S., An Engineering Approach to Computer Networking, Pearson Education, 2000.
4. Andrew S. Tanenbaum, Computer Networks, 3/E, PHI, 1996.
5. IEEE/ACM Trans on Networking

CS 6131: Logic and Computation

Prerequisite: Logic for Computer Science

L	T	P	C
3	0	0	3

Total Hours: (42)

Module 1:(10 hrs)
Temporal Logic based verification, Buchi automata, Linear Temporal Logic (LTL), Relational product , Modelling systems

Module 2:(9hrs)
Computation Tree Logic (CTL), Concept of fairness, Symbolic model checking Tools: Spin**,** SMV

Module 3:(13hrs)
Verification of infinite-state systems, Verification of real-time systems, Timed automata, Modelling real-time systems, RTCTL. Complexity Consideration, Tools: Uppaal, Kronos

Module 4:(10hrs)
Verification of pushdown systems, Verification of security protocols, Formal Methods, Abstract Interpretation Model

References

1. E. M. Clarke, O. Grumberg, and D. Peled. Model Checking. MIT Press, 1999.
2. B. Berard, M. Bidoit, A. Finkel, F. Laroussinie, A. Petit, L. Petrucci, P. Schnoebelen, and P.McKenzie. Systems and Software Verification: Model-Checking Techniques and Tools. Springer Verlag, 2001.
3. Jean A. Gallier, Logic for Computer Science Foundations of Automatic Theorem Proving, Wiley, 1986

CS 6132: Topics in Algorithms

Prerequisite: Design and Analysis of Algorithms

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1 :(10 Hours)
Discrete Probability Probability, Expectations, Tail Bounds, Chernoff Bound, Markov Chains and Random Walks, Martingales

Module 2: (11 hours)
Randomized Algorithms Finger Printing, Pattern Matching, Graph Problems, Algebraic Methods, Probabilistic Primality Testing, De-Randomization

Module 3: (10 hours)
Complexity Probabilistic Complexity Classes, Probabilistic Proof Theory and Certificates, Interactive and Zero Knowledge Proof Systems, Arthur Merlin Games.

Module 4: (11 hours)
Kolmogorv Complexity Definition of Randomness, Unsolvability results, Chatin's Proof for Gödel's Theorem.

References

1. R. Motwani and P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.
2. C. H. Papadimitriou, Computational Complexity, Addison Wesley, 1994
3. Journal of Algorithms, Elsevier.

CS 6133: Game Theory

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1:(7hrs)
Introduction to Non Co-operative Game Theory: Extensive Form Games, Strategic Form Games, Pure Strategy Nash Equilibrium

Module 2:(10hrs)
Nonco-operative Game Theory (in detail), Mixed Strategies, Existence of Nash Equilibrium, Computation of Nash Equilibrium, Two Player Zero-Sum Games, Bayesian Games

Module 3:(13hrs)
Mechanism Design : An Introduction, Dominant Strategy Implementation of Mechanisms, Vickrey-Clorke-Groves Mechanisms, Bayesian Implementation of Mechanisms, Revenue Equivalence Theorem, Design of Optimal Mechanisms

Module 4:(13hrs)
Cooperative Game Theory, Correlated Strategies, Correlated Equilibria, The Two Person Bargaining Problem, Games in Coalitional Form, The Core Shapley Value, Other Solution Concepts for Co-operative Games

References

1. Roger B. Myerson. Game Theory: Analysis of Conflict. Harvard University Press, September 1997.
2. Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green. Microeconomic Theory. Oxford University Press, New York, 1995.
3. Martin J. Osborne, Ariel Rubinstein. A Course in Game Theory. The MIT Press, August 1994.
4. Philip D. Straffin, Jr. Game Theory and Strategy. The Mathematical Association of America, January 1993.
5. Ken Binmore, Fun and Games : A Text On Game Theory, D. C. Heath & Company, 1992.
6. Paul Klemperer, Auctions: Theory and Practice, The Toulouse Lectures in Economics, Princeton University Press, 2004.

CS 6134: Quantum Computation

Prerequisites: Theory of Computation

L	T	P	C
3	0	0	3

Total Hours: (42)

Module 1: (12 Hours)
Foundations:Finite Dimensional Hilbert Spaces – Tensor Products and Operators on Hilbert Space – Hermitian and Trace Operators - Basic Quantum Mechanics necessary for the course.

Module 2: (9 hours)
Model of Computation:Quantum Gates and operators and Measurement – Quantum Computational Model – Quantum Complexity – Schemes for Physical realization (Only peripheral treatment expected).

Module 3: (10 hours)
Algorithms and Complexity Shor's Algorithm – Application to Integer Factorization – Grover's Algorithm – Quantum Complexity Classes and their relationship with classical complexity classes**.**

Module 4: (11 hours)
Coding Theory Quantum Noise – Introduction to the theory of Quantum Error Correction –Quantum Hamming Bound – Coding Schemes – Calderbank-Shor-Steane codes –Stabilizer Codes.

References

1. Nielsen M. A. and I. L. Chauang, Quantum Computation and Quantum Information, Cambridge University Press, 2002.
2. Gruska J. Quantum Computing, McGraw Hill, 1999.
3. Halmos, P. R. Finite Dimensional Vector Spaces, Van Nostrand, 1958.
4. Julian Brown, Minds, Machines and the Multiverse: The Quest for the Quantum Computer by Julian Brown. Simon and Schuster, 2000.

CS 6135: Logic for Computer Science

Prerequisite: Theory of Computation

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (6 Hours)
Propositional Logic**:** Review of undecidability and Complexity classes, Formal Systems, Syntax and Semantics of Prepositional Calculus, Completeness Theorem, Compactness, NP-Completeness of Satisfiability**.**

Module 2: (12 hours)
Predicate Logic**:** Syntax and semantics, Herbrand's Expansion Theorem, Proof Systems, Completeness and Compactness, Undecidability of Satisfiability.

Module 3: (12 hours)
Incompleteness:First order Axiomatization of number theory, Godel's Incompleteness Theorem, Limitations of first order logic.

Module 4: (12 hours)
Logic and Computation: Bucchi's Theorem, Logical Characterization for Complexity classes, Fagin'sTheorem.

References

1. C. H. Papadimitriou, Computational Complexity, Addison Wieley, 1994.
2. Jean A. Gallier, Logic for Computer Science Foundations of Automatic Theorem Proving, Wiley, 1986
3. E. M. Clarke, O. Grumberg, and D. Peled. Model Checking. MIT Press, 1999.

CS 6136: Topics in Combinatorial Algorithms

Prerequisite: Design and Analysis of Algorithms

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (10 hrs)
Primal dual theory of linear programming, and application to max flow, Min cuts, Dijkstra's algorithm and Floyd Warshall algorithms.

Module 2: (10 hrs)
Application of primal dual theoty to matching, bipartite matching, non bipartite matching, spanning trees and matroids and other related combinatorial optimization problems**.**

Module 3: (10hrs)
Randomization techniques, The probabilistic method. Existence proofs.

Module 4: (12 hrs)
Approximation algorithms. Node cover and travelling salesman problems. Non-approximability. Examples of non approximable problems.

References:

1. C. H. Papadimitirou and K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, Dover, 1998.
2. G. Ausiello et.al., Complexity and Approximation: Combinatorial Algorithms and their Approximability Properties, Springer, 2002.
3. Journal of Algorithms. Elsevier.

CS 6141: Distributed Computing

Prerequisite: Operating Systems

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1 : (10 hours)
Characterization of Distributed Systems, System Models, Networking and Internetworking, Inter Process communication

Module 2 : (10 hours)
Logical clocks, verifying clock algorithms, Mutual Exclusion, Mutual exclusion using timestamps, tokens ad Quorums.

Module 3 : (10 hours)
Name Services and Domain Name System, Directory and Discovery Systems, Drinking philosophers problem, leader elections, Global state, Termination Detection

Module 4: (12 hours)
Transactions and Concurrency Control, Distributed Transactions, Distributed Deadlocks, Transaction Recovery, Fault-tolerant Services, Distributed Shared Memory, Distributed consensus.

References

1. Vijay K. Garg. Elements of Distributed Computing, Wiley Interscience, 2002
2. Nancy Lynch, Distributed Algorithms, Morgan Kaufmann Publishers Inc., 1996.
3. Coulouris G., Dollimore J. & Kindberg T., Distributed Systems Concepts And Design, 3/e, Addison Wesley 2004
4. Tanenbaum S, Maarten V.S., Distributed Systems Principles and Paradigms, Pearson Education 2004
5. Chow R. & Johnson T., Distributed Operating Systems and Algorithms, Addison Wesley 2003
6. Tanenbaum S., Distributed Operating Systems, Pearson Education 2005

CS 6142 : Topics in Computer Architecture

Prerequisite: Computer Architecture

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (10 Hours)

Advanced ILP Exploitation Techniques**:** Hardware and software techniques for ILP extraction, speculative execution, studies on ILP.

Module 2: (10 hours)

Multithreaded processors and Multicore processors Concept, methodologies and analysis. Speculative multithreading.Multicore processor design and compilation issues, scheduling. **CMPs and Polymorphic processors** Concept, Studies and Analysis

Module 3: (12 hours)

Simulators in Computer Architecture Introduction – methods, ADLs, traces, dynamic compilation. Multicore simulators. Functional and performance Simulators.

Module 4: (10 hours)

Processor Based Security. Virtualization, Virtual Machines. Hypervisors. Security Issues.

References

1. ACM SIGARCH Computer Architecture News.
2. The WWW Computer Architecture page. http://www.cs.wisc.edu/arch/www/ last accessed 28/3/2010
3. Hennessy J. L., D. Patterson, Computer Architecture – A quantitative Approach, Morgan Kuffman (3/e), 2003

CS 6143: Trends in Middleware Systems

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (12 hours)
Publish/Subscribe matching algorithm **,** event based systems , notification filtering mechanisms, Composite event processing, content based routing, content based models and matching, matching algorithms, distributed hash tables (DHT)

Module 2: (10 hours)
Distributed notification routing, content based routing algorithms, engineering event based systems, Accessing publish/subscribe functionality using APIs. Scoping, event based systems with scopes, notification mappings, transmission policies, implementation strategies for scoping.

Module 3: (10 hours)
Composite event detection, detection architectures, security, fault tolerance, congestion control, mobility, existing notification standards- JMS, DDS, HLA.

Module 4:(10 hours)
Topic based systems, Overlays, P2P systems, overlay routing, Case studies- REBECA, HERMES, Gryphon. Commercial systems- IBM Websphere MQ, TIBCO Rendezvous.

References

1. Gero Muhl, Ludger Fiege, Peter R. Pietzuch. Distributed Event-Based Systems. Springer, 2006
2. Chris Britton, Peter Bye. IT Architectures and Middleware. Pearson Education second Edition, 2005
3. Yanlei Diao, and Michael J. Franklin. Query Processing for High-Volume XML Message Brokering. VLDB 2003.
4. 2. Chee-Yong Chan, Minos Garofalakis, Rajeev Rastogi . RE-Tree: An Efficient Index Structure for Regular Expressions, VLDB 2002.
5. Peter R. Pietzuch, Brian Shand, Jean Bacon. A Framework for Event Composition in Distributed Systems, Proc. of the 4th Int. Conf. on Middleware (MW'03)
6. A. Carzaniga and A.L. Wolf. Forwarding in a Content-Based Network. Proceedings of ACM SIGCOMM 2003. p. 163-174. Karlsruhe, Germany. August, 2003.

CS 6144 – Multicore Systems

Prerequisite: Compiler Design, Operating Systems and Computer Architecture

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1: (12 Hours)
Introduction to multi-core architectures, issues involved in writing code for multi-core architectures, Challenges in developing programs for multi-core architectures, Introduction Program optimizations techniques and about building these techniques in compilers. Pipeline Design issues, Memory hierarchy and Memory system issues in multicore system. Memory consistency models. Synchronization primitives; Performance implications in shared memory programs;

Module 2: (10 Hours)
Chip multiprocessors: Why CMP (Moore's law, wire delay); shared L2 vs. tiled CMP; core complexity; power/performance; Multicore, Multiprocessor, Super speculative & VLIW architectures – a comparative study. Chip multiprocessor case studies: Intel Montecito and dual-core Pentium4, IBM Power4

Module 3: (8 Hours)
Optimization: Analysis & Optimization for multicore architectures, Dataflow analysis, Pointer analysis, alias analysis; Data dependence analysis, solving data dependence equations (integer linear programming problem); Loop optimizations; Memory hierarchy issues in code optimization.

Module 4: (10 Hours)
Multicore programming methodologies and measuring performance, Case studies from Applications such as Digital Signal Processing, Image processing, Speech processing.

References

1. Michael J. Quinn. Parallel Programming in C with MPI and Open MP. McGraw Hill. 2003.
2. David E. Culler and Jaswinder Pal Singh, with Anoop Gupta. Parallel Computer Architecture: A Hardware/Software Approach. 1998.
3. Hennesy J. L. & Pattersen D. A., Computer Architecture: A Quantitative approach, 3/e, Harcourt Asia Pte Ltd. (Morgan Kaufman), Singapore.

CS 6151: Software Engineering

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1:(9 Hours)
The Software Process: Life cycle models, from specification to maintenance. Critical systems, Requirements Engineering: Functional and non-functional requirements, preparing and validating feasibility studies

Module 2:(11 Hours)
Software Design: Architectural, Object Oriented, Real-time, and Reusable Designs. Object Oriented design with UML

Module 3: (10 Hours)
Software Project Management: Managing Process, People, Product and Quality, Reliability

Module 4:(12 Hours)
Formal specification and verification of programs and software. Project Delays - The symptoms, reasons, and solutions. Performance concerns - Improving design methods and programming styles for producing better software

References:
1. Ian Sommerville, Software Engineering, 6/e, Pearson Education Asia, 2001.
2. Oestereich, Developing Software with UML, Addison Wesley.
3. Steve McConnell, Code Complete, Microsoft Press, 1993.
4. F P Brooks, Jr., Mythical Man-Month, 2/E,Addison Wesley.
5. J L Bentley, Programming Pearls, 2/E, Addison Wesley, 1999.

CS 6152 :Object Oriented Modeling & Design

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1 : (10 hours)
Structural Modeling: Object Oriented Fundamentals, Basic structural Modeling, UML Model , Class Diagrams, Object Diagrams, Packages and Interfaces, Case Studies**.**

Module 2 : (12 hours)
Behavioral and architectural Modeling: Use Case Diagrams, Interaction Diagrams, State Chart Diagrams, Collaborations, Design Patterns, Component Diagrams, Deployment Diagrams, Case Studies

Module 3 : (9 hours)
Object oriented Testing Methodologies: Implications of Inheritance on Testing, State Based Testing, Adequacy and Coverage, Scenario Based Testing, Testing Workflow, Case Studies , Object Oriented Metrics

Module 4: (11 hours)
Components: Abuses of inheritance, danger of polymorphism, mix-in classes,
rings of operations, class cohesion and support of states and behavior,
components and objects, design of a component, lightweight and heavyweight components, advantages and disadvantages of using components.

References
1. Page Jones M., Fundamentals of Object Oriented Design in UML, Pearson Education, 2002
2. Booch G., Rumbaugh J. & Jacobsons I., The Unified Modeling Language User Guide, Addison Wesley, 2002.
3. Bahrami A., Object Oriented System Development, McGraw Hill, 2003.
4. Baugh J., Jacobson I. & Booch G., The unified Modeling Language Reference Manual, Addison Wesley, 1999.
5. Man C., Applying UML & Patterns: An Introduction to Object – Oriented Analysis & Design, Addison Wesley, 2002.
6. Ooley R. & Stevens P., Using UML: Software Engineering with Objects & Components, Addison Wesley, 2000.

CS 6154: Topics in Database Design

Prerequisite: Database Design

L	T	P	C
3	0	2	4

Total Hours(42+28)

Module 1 : (11 hours)
Parallel and Distributed Databases: Architecture of Parallel Databases, Parallel Query Optimization, Distributed DBMS Architectures, Distributed Query Processing, Distributed Concurrency Control, Distributed recovery.

Module 2 : (10 hours)
Internet Databases and Data Mining : XML –QL, Ranked Keyword searches on the Web, Data Mining, Clustering, Similarity Search over Sequences**.**

Module 3 : (10 hours)
Object Oriented Database Systems: User Defined ADTs, Objects, Object Identity and Reference types, Database Design for ORDBMS, OODBMS, Comparison of RDBMS with OODBMS and ORDBMS**.**

Module 4: (12 hours)
Spatial and Deductive Databases: Spatial and Temporal Databases, Temporal Logic, Spatial Indexes, Introduction to Recursive Queries, Introduction to Mobile Databases, Main Memory and Multimedia Databases

References
1. Elmasri & Navathe, Fundamentals of Database Systems, 3rd Edition, Addison Wesley
2. O'neil P. & O'neil E., Database Principles, Programming, and Performance, 2/e, Harcourt Asia, Morgan Kaufman
3. Silberschatz A., Korth H. F., & Sudarshan S., Database System
Concepts, Tata McGraw Hill, 2003
4. Ullman J. D., Principles of Database Systems, Galgotia Publications,1996.
5. Date C. J., An Introduction to Database Systems, Addison Wesley, 2000.
6. Ramakrishnan R. & Gehrke J., Database Management Systems, 3/e, McGraw Hill, 2004

CS 6171: Natural Language Processing

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours(42+28)

Module 1 (11 hours)
Introduction to Natural Language Understanding: The study of Language, Applications of NLP, Evaluating Language Understanding Systems, Different levels of Language Analysis, Representations and Understanding, Organization of Natural language Understanding Systems, Linguistic Background: An outline of English syntax.

Module 2 (11 hours)
Grammars and Parsing: Grammars and sentence Structure, Top-Down and Bottom-Up Parsers, Transition Network Grammars, Top-Down Chart Parsing. Feature Systems and Augmented Grammars: Basic Feature system for English, Morphological Analysis and the Lexicon, Parsing with Features, Augmented Transition Networks

Module 3 (11 hours)
Grammars for Natural Language: Auxiliary Verbs and Verb Phrases, Movement Phenomenon in Language, Handling questions in Context-Free Grammars, Hold mechanisms in ATNs. Human preferences in Parsing, Encoding uncertainty, Deterministic Parser

Module 4 (9 hours)
Ambiguity Resolution: Statistical Methods, Estimating Probabilities, Part-of-Speech tagging, Obtaining Lexical Probabilities, Probabilistic Context-Free Grammars, Best First Parsing. Semantics and Logical Form, Word senses and Ambiguity, Encoding Ambiguity in Logical Form.

References

1. James Allen, Natural Language Understanding, 2/e, Pearson Education, 2003.
2. D. Jurafsky, J. H. Martin, Speech and Language Processing, Pearson Education, 2002.
3. Christpher G. Manning, Hinrich Schütze, Foundations of Statistical Natural Language Processing, The MIT Press, Cambridge, Massachusetts.1999.

CS 6172: Computational Intelligence

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours(42+28)

Module 1: (10 hours)
Artificial Intelligence: History and Applications, Production Systems,Structures and Strategies for state space search- Data driven and goal driven search, Depth First and Breadth First Search, DFS with Iterative Deepening, Heuristic Search- Best First Search, A* Algorithm, AO* Algorithm, Constraint Satisfaction, Using heuristics in games- Minimax Search, Alpha Beta Procedure.

Module 2: (11 hours)
Knowledge representation - Propositional calculus, Predicate Calculus,Theorem proving by Resolution, Answer Extraction, AI Representational Schemes- Semantic Nets, Conceptual Dependency, Scripts, Frames, Introduction to Agent based problem solving.

Module 3: (10 hours)
Machine Learning- Symbol based and Connectionist, Social and Emergent models of learning, The Genetic Algorithm- Genetic Programming, Overview of Expert System Technology- Rule based Expert Systems, Introduction to Natural Language Processing.

Module 4 :(11 hours)
Languages and Programming Techniques for AI- Introduction to PROLOG and LISP, Search strategies and Logic Programming in LISP, Production System examples in PROLOG.

References

1. George.F.Luger, Artificial Intelligence- Structures and Strategies for Complex Problem Solving, 4/e, 2002, Pearson Education.
2. E. Rich, K.Khight, Artificial Intelligence, 2/e, Tata McGraw Hill
3. Winston. P. H, LISP, Addison Wesley
4. Ivan Bratko, Prolog Programming for Artificial Intelligence, 3/e, Addison Wesley, 2000

CS 6173: Image Processing

Prerequisite: Nil

L	T	P	C
4	0	0	4

Total Hours (56)

Module 1 (21 hours)
Introduction - digital image representation - fundamental steps in image processing - elements of digital image processing systems - digital image fundamentals - elements of visual perception - a simple image model - sampling and quantization - basic relationship between pixels - imagegeometry - image transforms - introduction to Fourier transform – discrete Fourier transform - some properties of 2d-fourier transform (DFT)- other separable image transforms - hotelling transform

Module 2 (13 hours)
Image enhancement - point processing - spatial filtering - frequency domain -
image restoration - degradation model - diagonalization of circulant and block circulant matrices - inverse filtering - least mean square filter

Module 3 (11 hours)
Image compression - image compression models - elements of information theory - error-free compression - lossy compression - image compression standards

Module 4 (11 hours)
Image reconstruction from projections - basics of projection - parallel beam and fan beam projection - method of generating projections - Fourier slice theorem - filtered back projection algorithms - testing back projection algorithms

Reference books

1. Rafael C., Gonzalez & Woods R.E., Digital Image Processing, Addison Wesley, 1999.
2. Rosenfeld A. & Kak A.C., Digital Picture Processing, Academic Press, 1998
3. Jain A.K, Fundamentals of Digital Image Processing, Prentice Hall, Englewood Cliffs, 2002.
4. Schalkoff R. J., Digital Image Processing and Computer Vision, John Wiley, 2004.
5. Pratt W.K., Digital Image Processing, John Wiley, 2002.

CS 6174: Pattern Recognition

Prerequisite: Nil

L	T	P	C
4	0	0	4

Total Hours: (56)

Module 1: (16 hrs)
Introduction- Introduction to statistical, syntactic and descriptive approaches, features and feature extraction, learning.Bayes Decision theory- introduction, continuous case, 2-categoryclassification, minimum error rate classification, classifiers, discriminant functions, and decision surfaces. Error probabilities and integrals, normal density**,** discriminant functions for normal density, Bayes Decision theory Discrete case.

Module 2: (14 hrs)
Parameter estimation and supervised learning- Maximum likelihood estimation, the Bayes classifier, learning the mean of a normal density,general bayesian learning.Nonparametric technique- density estimation, parzen windows, k-nearest Neighbor estimation, estimation of posterior probabilities, kn nearest neighbor rule, nearest-neighbor rule, k-nearest neighbor rule.

Module 3: (13 hrs)
Linear discriminant functions- linear discriminant functions and decision surfaces, generalized linear discriminant functions, 2-category linearly separable case, non-separable behavior, linear programming procedures.

Multiplayer neural networks- Feed forward operation and classification, Backpropagation algorithm, error surfaces, back propagation as feature mapping,practical techniques for improving back propagation**.**

Module 4: (13 hrs)
Supervised learning and clustering- Mixture densities and identifiably, maximum likelihood estimates, application to normal mixtures, unsupervised Bayesian learning, data description and clustering, Hierarchical clustering,low dimensional representation of multidimensional map

References

1. Duda and Hart P.E, Pattern classification and scene analysis, John Wiley and sons, NY, 1973.
2. Earl Gose, Richard Johnsonbaugh, and Steve Jost; Pattern Recognition and Image Analysis, PHI Pvte. Ltd., NewDelhi-1, 1999.
3. Fu K.S., Syntactic Pattern recognition and applications, Prentice Hall, Eaglewood cliffs, N.J., 1982
4. Rochard O. Duda and Hart P.E, and David G Stork, Pattern classification , 2nd Edn., John Wiley & Sons Inc., 2001.

CS 6181: Bioinformatics

Prerequisite: Nil

L	T	P	C
3	0	2	4

Total Hours: (42+28)

Module 1 :(10 Hrs)
Introduction to Molecular biology, Gene structure and information content, Molecular biology tools, Algorithms for sequence alignment, Sequence databases and tools.

Module 2 :(10 hrs)
Molecular Phylogenetics, Phylogenetic trees, Algorithms for Phylogenetic tree construction, Introduction to Perl programming for Bioinformatics.

Module 3: (11 Hrs)
Introduction to Protein structure, Algorithms for Protein structure prediction, Gene expression analysis, Micro Arrays, Pathway analysis.

Module 4: (11 Hrs)
Pattern Matching algorithms, Bio-data analysis, Data Mining in Bioinformatics, Algorithms and data structures for efficient analysis of biological data.

References

1. D. E. Krane and M. L. Raymer, Fundamental Concepts of Bioinformatics, Pearson Education, 2003.
2. T. K. Attwood and D. J. Parry-Smith, Introduction to Bioinformatics, Pearson Education, 2003.
3. Arthur M Lesk, Introduction to Bioinformatics, Oxford University Press, 2002
4. Jean Michel Claverie and Cedric Notredame, Bioinformatics – A Beginner's guide, Wiley-Dreamtech India Pvt. Ltd., 2003
5. Neil C Jones and Pavel A Pevzner, An Introduction to Bioinformatics Algorithms, MIT Press, 2004
6. Current literature.

MA 8152 Fuzzy Set Theory and Applications

Prerequisite: Nil

L	T	P	C
3	0	0	3

Total Hours (42)

Module I: (11 hours)

Crisp sets and Fuzzy sets **:** Introduction – crisp sets an overview – the notion of fuzzy sets –basic concepts of fuzzy sets – membership functions – methods of generating membership functions – defuzzification methods- operations on fuzzy sets - fuzzy complement – fuzzy union – fuzzy intersection – combinations of operations – General aggregation operations.

Module 2: (11 hours)

Fuzzy arithmetic and Fuzzy relations: Fuzzy numbers- arithmetic operations on intervals- arithmetic operations on fuzzy numbers- fuzzy equations- crisp and fuzzy relations – binary relations – binary relations on a single set – equivalence and similarity relations – compatibility or tolerance relations.

Module 3 : (10 hours)

Fuzzy measures – belief and plausibility measures – probability measures – possibility and necessity measures – possibility distribution - relationship among classes of fuzzy measures.

Module 4: (10 hours)

Fuzzy Logic and Applications **:** Classical logic : an overview – fuzzy logic – approximate reasoning - other forms of implication operations - other forms of the composition operations – fuzzy decision making –fuzzy logic in database and information systems - fuzzy pattern recognition – fuzzy control systems.

References:

1. George J Klir and Tina A Folger , Fuzzy sets, Uncertainty and Information, Prentice Hall of India, 1988.
2. H.J. Zimmerman, Fuzzy Set theory and its Applications, 4th Edition, Kluwer Academic Publishers, 2001.
3. Goerge J Klir and Bo Yuan , Fuzzy sets and Fuzzy logic: Theory and Applications. Prentice Hall of India, 1997.
4. Hung T Nguyen and Elbert A Walker, First Course in Fuzzy Logic, 2nd Edition , Chapman & Hall/CRC, 1999.
5. Jerry M Mendel, Uncertain Rule – Based Fuzzy Logic Systems ; Introduction and New Directions, PH PTR, 2000.

6. John Yen and Reza Langari, Fuzzy Logic : Intelligence Control and Information, Pearson Education, 1999.
7. Timothy J Ross, Fuzzy Logic with Engineering Applications, McGraw Hill International Editions, 1997.

MA7156 Advanced Topics in Graph Theory

Prerequisite: Nil

L	T	P	C
3	0	0	3

Total Hours (42)

Module 1: (8 hours)

Graphs, Connectivity and Traversability Graphs: review of basics in graphs - -Trees- Blocks- Matrices- Operations on graphs. Connectivity: Vertex Connectivity and edge connectivity – n- connected graphs- Menger's Theorem. Traversability: Euler graphs-Hamiltonian Graphs-Planar and Nonplanar graphs.

Module 2: (14 hours)

Metric in graph, Distance Sequences ,Convexity and Symmetry. Metric in graph: Centre, Median, eccentric vertex, Eccentric graph, boundary vertex, complete vertex, interior vertex. Convexity: Closure Invariants- gin(G) –gn(G)-Hull number- Geodetic Graphs- Distance Hereditary Graphs Symmetry: Graphs and groups-- Symmetric Graphs - Distance Symmetry-Distance transitive graphs-distance regular graphs Distance Sequences **:**Degree sequence, Eccentric Sequence - Distance Sequences - The Distance Distribution, Mean distance.

Module 3 : (10 hours)

Matchings, Factorization and Domination Matchings :Maximum matching-Perfect matching-Matching in bipartite graphs Factorization :Coverings and independence-1-fcatorization-2-factorization-Arboricity Domination: Dominating set-Domination number-total dominating set –total domination number.

Module 4: (10 hours)

Digraphs, Networks and Algorithms Digraphs: Digraphs and connectedness- Tournaments- directed trees- binary trees- weighted trees and prefix codes Networks: Flows-cuts- The Max- Flow Min-Cut Theorem Graph Algorithms: Polynomial Algorithms and NP completeness ,Complexity, Search algorithms, Shortest path algorithms.

References

1. Gary Chartrand, Ping Zhang, Introduction to Graph Theory, McGraw Hill International Edition, 2005.
2. J.A.Bondy, U.S.R.Murty, Graph Theory, Springer, 2008.
3. Distance in Graphs, Fred Buckley and Frank Harary, Addison - Wesley (1990).
4. C. R. Foulds: Graph Theory Applications, Narosa Publishing House, 1994.
5. Harary F: Graph Theory, Addison-Wesley pub. 1972.
6. R. P. Grimaldi, Discrete and Combinatorial Mathematics: An Applied Introduction, Addison Wesley, 1994.
7. C. Vasudev, Graph Theory with Applications, New Age international publishers, 2006.