

# Jodhpur National University, Jodhpur

## M. Tech Programme (Computer Science & Engineering)

Teaching / Examination Scheme & Syllabus

Session 2009-2011

Date: 25/07/2009

### I SEMESTER

Subject Code	Subject	Hrs./ Week				Marks		
		L	T	P	Total	Theory Exam/ Viva voce	Internal Assessment	Total
1MCS01	Theory of Formal Languages	3	1	-	4	100	50	150
1MCS02	Advanced Operating Systems	3	1	-	4	100	50	150
1MCS03	Advanced Database Management Systems	3	1	-	4	100	50	150
1MCS04	Strategic Management	3	-	-	3	100	50	150
1MCS05	Advanced Database Lab	-	-	3	3	50	50	100
	<b>Total</b>	<b>12</b>	<b>3</b>	<b>3</b>	<b>18</b>	<b>450</b>	<b>250</b>	<b>700</b>

### II SEMESTER

Subject Code	Subject	Hrs./ Week				Marks		
		L	T	P	Total	Theory Exam/ Viva voce	Internal Assessment	Total
2MCS01	Advanced Data Structures	3	1	-	4	100	50	150
2MCS02	Advanced Computer Networks	3	1	-	4	100	50	150
2MCS03	Information Protection & Computer Security	3	1	-	4	100	50	150
2MCS04	Elective	3	1	-	4	100	50	150
2MCS05	Information Protection & Computer Security Lab	-	-	3	3	50	50	100
	<b>Total</b>	<b>12</b>	<b>4</b>	<b>3</b>	<b>19</b>	<b>450</b>	<b>250</b>	<b>700</b>

List of Elective subjects:

**2E01.** Information Retrieval

**2E02.** Modern Compiler Design

**2E03.** Software System Design

### III SEMESTER

Subject Code	Subject	Hrs./ Week				Marks		
		L	T	P	Total	Theory Exam/ Viva voce	Internal Assessment	Total
3MCS01	Fault Tolerant Computing	3	1	-	4	100	50	150
3MCS02	Elective:	3	1	-	4	100	50	150
3MCS03	Seminar	-	-	3	3			
	<b>Total</b>	<b>6</b>	<b>2</b>	<b>3</b>	<b>11</b>	<b>200</b>	<b>100</b>	<b>300</b>

List of Elective subjects:

**3E01.** Advanced Real Time Systems

**3E02.** Advanced Computer Graphics

**3E03.** Design of Embedded Systems

### IV SEMESTER

Subject Code	Subject	Hrs./Week				Marks		
		L	T	P	Total	Theory Exam/ Viva voce	Internal Assessment	Total
4MCS 01	Dissertation	-	-	3	3	150	150	300
	<b>Total</b>	<b>0</b>	<b>0</b>	<b>3</b>	<b>3</b>	<b>150</b>	<b>150</b>	<b>300</b>

**List of the Dissertation Subjects for session 2008-2010:**

**4E01** Cloud Computing

**4E02** Web 2.0

**4E03** Software Testing (Formal Specs. & Verification)

**4E04** VLDB – Very Large Data Bases

**4E05** Embedded System Design / Soft Computing

**4E06** HMI – Human Machine Interface

**(A new list will be given in each session)**

A group of students will be assigned, one subject each. They will study and research the subject. This study will be done in three stages:

1. Problem definition / Detailed Synopsis.
2. Width coverage. (Literature survey)
3. Depth coverage. (Research one specific aspect )

**Total Marks: 700 + 700 + 300 + 300 = 2000**

# FIRST SEMESTER

## 1MCS01-THEORY OF FORMAL LANGUAGES

3L, 1T

3 Hours, 100 Marks

Formal languages and their related automata, Turing machines, type-0 languages, linear bounded automata and CFLs. Time and tape bounded Turing machines, time and space bounds for recognizing CFLs. Turing.

Computability: number theoretic computations by Turing machines and indexing Axiomatic systems, their soundness and completeness.

Recursive function theory: primitive recursive functions and primitive recursive predicates. Ackermann's function, recursive and general recursive functions.

Computability and decidability: computable functions computable sets, decision problems. Fixpoint theory of programs functions and functional verification methods, Lambda calculus and applications.

### **Suggested reference materials:**

1. K.L.P. Mishra: Introduction to Language and Computation, Pearson Education Asia

## 1MCS02 - ADVANCED OPERATING SYSTEMS

3L, 1T

3 Hours, 100 Marks

Distributed Operating System: Issues, limitation, causal and total ordering.

Logical clocks, mutual exclusion classification and algorithms, deadlock model, detection, prevention, avoidance, resolution, centralized and distributed detection algorithms. Distributed file system architecture, design issues, memory coherence, granularity, page replacement. Distributed scheduling, load distribution, stability, load sharing, task migration.

### **Suggested reference materials:**

1. Galvin / Silberschitz : Operating Systems Concepts, TMH

## 1MCS03 - ADVANCED DATABASE MANAGEMENT SYSTEM

3L, 1T

3 Hours, 100 Marks

Overview of DBMS, concurrency control , failure recovery. Introduction to distributed data base management systems, Semantic Database Models and Systems, Object-Oriented Database Systems, Relational Extensions: Design Techniques, Extension Techniques Object / Relational Systems: Open ODB, Transaction Management, Interface, OSQL, Adapter, Case Study of an ORDBMS, Related Development, Current Product Scenario. Standard For OODBMS Products and Applications: ODM – Standards, ODMG, Smalltalk Binding, SQL, User Defined ADT in SQL, Routines, ADT Subtypes and Inheritance, Tables, Procedural Facilities, Other Type Constructions, Generic ADT Packages, Language Bindings.

### **Suggested reference materials:**

1. C S R Prabhu,"Object Oriented Data Base Systems" approaches and Architectures, PHI,
2. F. H. Lochovsky, DC Tsichritzis"DBMS" NewYork Academic Press.
3. F. H. Lochovsky, DC Tsichritzis"Data Models" PHI.
4. C.J.DATE "Introduction to Data Base to Management System" Addison Wesley.
5. N. Goodman, V. Hadzilacos "Concurrency Control and Recovery in Data Base System" Addison Wesley.

## **1MCS04 – STRATEGIC MANAGEMENT**

3L

3 Hours, 100 Marks

Strategic management' could be a useful subjects of 1st semester of M-Tech courses. Strategic management' as detailed below prepares a young budding technocrat to manage assets in more efficient way and lead his/her team more effectively.

This subject would cover the following issues:-

- a) **Managing Change:-** Effective change management is the need of the hour. Situations change rather fast these days and has to prepare or mold his / her organization to face such changes and come out winningly.
- b) **Crisis Management:-** Every organization or every individual do face crisis many times. A good management or a good leader keeps himself / herself always prepare for such eventualities. Training in this area has often been neglected.
- c) **Innovation and Creativity :-** Human brain is very creative. Creative thinking is a must for the fast changing world. Creative things results in innovation and new finds. Creativity could be developed in a positive sense by training.
- d) **Entrepreneurship :-** Young budding technocrats need encouragement for creating small/medium size organizations. Sessions on this subject from industry leaders will help in it.
- e) **Work Study and Re-engineering:-** Re-engineering is term used these days in place of old terms 'Work Study' Re-engineering is needed not only for the product but also fro the processes as well as for the 'Organization'.
- f) **Managing Intangibles :-** An organization have (i) Tangibles assets like machines, material etc. and also(ii) Intangibles assets like staff, line managers etc. Managing 'people' or rather 'Leading' people needs to be taught to young engineers.
- g) **Communication Skills :-** We expect our managers to lead their terms. For this communication skills is a must. This could be covered in class room sessions along with practice sessions in groups under the supervision of a teacher.
- h) **Quality and Customers Care :-** ISO -9000 has become a hallmark for quality. This can cover pre-requisites for an organization and how to go about for getting ISO-9000 certification.
- i) **Safety and Ergonomics:-** Safety is often talked about but not cared for to that extant ergonomics is even less known. Technocrats need to know more about

**Note:-** This list is not exhaustive and could be enlarged in due course.

# SECOND SEMESTER

## 2MCS01-ADVANCED DATA STRUCTURES

3L, 1T

3 Hours, 100 Marks

Advanced data structures: self-adjustment, persistence and multidimensional trees. Randomized algorithms: Use of probabilistic inequalities in analysis & applications. Geometric algorithms: Point location, convex hulls and Voronoi diagrams, Arrangements. Graph algorithms: Matching and Flows. Approximation algorithms: Use of Linear programming and primal dual, local search heuristics. Parallel algorithms: Basic techniques for sorting, searching, merging, list ranking in PRAMs and Interconnection networks

### **Suggested reference materials:**

1. Motwani and Raghavan "Randomized Algorithms", Cambridge University Press
2. Preparata and Shamos "Computational Geometry", Springer Verlag
3. Mehlhorn "Data Structures and Algorithms: 1, Searching and Sorting", Springer Verlag EATCP Monograph on Theoretical Computer Science
4. Papadimitrou and Steiglitz "Combinatorial Optimization", Princeton University Press
5. Joseph Ja'Ja' "Introduction to Parallel Algorithms" Addison-Wesley.
6. Vaizirani "Approximation Algorithms", Springer

## 2MCS02-ADVANCED COMPUTER NETWORKS

3L, 1T

3 Hours, 100 Marks

Flow and Congestion Control: Window and Rate Based Schemes, Decbit, TCP, ATM ABR, hop-by-hop schemes. Quality of Service: in ATM, IETF integrated services model, Differentiated Services Model.

Flow Identification, Packet Classifiers and Filters Scheduling.

Network Management: ASN, SNMP, CMIP. Issues in the management of large network.

Multicast: IGMP, PIM, DVMRP. Mobility: IP.

### **Suggested reference materials:**

Computer Networks: a top-down approach featuring Internet, James F. Kurose & Keith W. Ross.

## 2MSC03-INFORMATION PROTECTION AND COMPUTER SECURITY

3L, 1T

3 Hours, 100 Marks

Introductions: Basic objectives of cryptography, secret-key and public-key cryptograph, one way and trapdoor one-way functions, cryptanalysis, attack models, classical cryptography.

Block ciphers: Modes of operation, DES and its variants, RC5, IDEA, SAFER, FEAL, BlowFish, AES, liner and differential cryptanalysis.

Stream ciphers: Stream ciphers based on liner feedback shift registers, SEAL, unconditional security.

Message digest: Properties of hash functions, MD2 MD5 and SHA-1, keyed hash functions, attacks on hash functions.

Public- key parameters: Modular arithmetic, gcd, primality testing , Chinese remainder theorem, modular square roots, finite fields.

Intractable problems: Integer factorization problem, RSA problem, modular square root problem, discrete logarithm problem, Diffie- Hellman problem, known algorithms for solving the intractable problems.

Public-key encryption: RSA, Rabin and ElGamal schemes, side channel attacks.

Key exchange: Diffie-Hellman and MQV algorithms.

Digital signatures: RSA, DSA and ECDSA signature schemes, blind and undeniable signatures.

Entity authentication: Passwords, challenge-response algorithms, zero-knowledge protocols.

**Suggested reference materials:**

1. Cryptography & Network Security, William Stallings

**2MSC04– INFORMATION RETRIEVAL (2E01)**

3L, 1T

3 Hours, 100 Marks

Introduction, Information retrieval V/S Retrieval, logical view of documents, retrieval process.

Basic models for IR and their formal characterization Probabilistic, Bayesian, and Dempster Shafer approaches.

Retrieval Evaluation, query languages.

Text operations – document clustering, compressions and various compression models.

Indexing and searching – brute-force, Knuth-Morris-Pratt, Boyer-Moore, pattern matching, basic idea of parallel and distributed IR.

User interfaces and visualizations, Information access process, query specifications, relevance judgments.

Searching web, search engines, browsings and met searches, searching through hyperlinks.

**Suggested reference materials:**

Modern Information Retrieval, Baeza-yates & Ribiero-neto

**2E02–MODERN COMPILER DESIGN**

3L, 1T

3 Hours, 100 Marks

Introduction to Advanced Topics of Informal Compiler Algorithm Notation (ICAN), Control-Flow Analysis,

Data-Flow Analysis, Dependence Analysis and Dependence Graphs, Alias Analysis, Introduction to

Optimization, Redundancy Elimination, Loop Optimizations, Procedure Optimizations, Case Studies of

Compilers and Future Trends.

**Suggested reference materials:**

1. Steven S. Muchnick: "Advanced Compiler Design and Implementation" Morgan Kaufmann.

1. Aho Ullaman Sethi " Compiler Construction" Addison Wesley.

2. Holob " Compiler Designing " TMH.

## **2E03–SOFTWARE SYSTEM DESIGN**

3L, 1T

3 Hours, 100 Marks

Concepts and techniques relevant to production of large software systems: Structured programming.

Requirements, specification and analysis. Top-down design and development. Information hiding, abstraction,

modularity, object-oriented techniques. Separate compilation, configuration management, program libraries.

Design patterns, UML Documentation. Validation. Quality assurance, safety. Testing and test case generation.

Software metrics. Cost analysis and estimation, manpower and time management.

Organization and management

of large software design projects; use of CASE tools.

### **Suggested reference materials:**

1. Sommerville, "Software Engineering", Addison-Wesley, 1999.
2. Peters and Pedrycz, "Software Engineering: an Engineering Approach", Wiley, 1999.
3. Pressman "Software Engg", PHI

# THIRD SEMESTER

## **3MCS01–FAULT TOLERANT COMPUTING**

3L, 1T

3 Hours, 100 Marks

Dependability concepts: dependable system, techniques for achieving dependability, dependability measure, fault, error, failure, and classification of faults and failures Fault Tolerance Strategies: Fault detection, masking, containment, location, reconfiguration, and recovery; Fault Tolerant Design Techniques: Hardware redundancy, software redundancy, time redundancy, and information redundancy. Dependable communication: Dependable channels, survivable networks, fault-tolerant routing. Fault recovery, stable storage and RAID architectures, and data replication and resiliency. Tolerance in Distributed System: Byzantine General Problem, consensus protocols, check pointing and Fault Tolerance interconnection networks: Analysis of fault tolerant hardware and software architectures. Case studies of fault tolerant multiprocessor and distributed systems.

### **Reference Books:-**

1. P. Jalote, "Fault Tolerance in Distributed Systems" Prentice-Hall Inc. 1994,
2. D. K. Pradhan "Fault-Tolerant Computing, Theory and Techniques", Prentice-Hall, 1998,
3. Los Alamitos, CA, "fault-Tolerant Computing, Theory and Techniques and application", IEEE Computer Society Press.
4. Barry W. Johnson, "Design and Analysis of Fault-Tolerant Digital System", Addison Wesley, 1989

## **3E01. ADVANCED REAL TIME SYSTEMS**

3L, 1T

3 Hours, 100 Marks

Advance Real-Time Systems: Multiprocessor scheduling, load sharing techniques, performance metric in Real-Time Systems. Resource management and resource reclaiming in uniprocessor and multiprocessor systems. Scheduling flexible computations and tasks with temporal distance constraints. Practical factors and overheads in scheduling, task synchronization, fault tolerance in multi processor systems, Real-Time communication. Introduction to object oriented approaches; case study of distributed Real-Time Systems.

### **Suggested reference materials:**

1. J.W.S. Liu: "Real-Time system", Pearson Education Asia.
2. S.T. Lavi, A.K. Agarwal: "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.

### **3E02. ADVANCED COMPUTER GRAPHICS**

3L, 1T

3 Hours, 100 Marks

Rendering: Ray tracing, Radiosity methods, Global illumination models, Shadow generation, Mapping, Anti-aliasing, Volume rendering, Geometrical Modeling: Parametric surfaces, Implicit surfaces, Meshes, Animation: spline driven, quaternion, articulated structures (forward and inverse kinematics), deformation purely geometric, physically-based.

#### **Suggested reference materials:**

1. Alan Watt and Mark Watt: "Advanced Animation and Rendering Techniques, Theory and Practice", Addison Wesley.

### **3E03. DESIGN OF EMBEDDED SYSTEMS**

3L, 1T

3 Hours, 100 Marks

Embedded Computing Requirements: Characteristics and applications of embedded systems; Components of Embedded Systems; challenges in Embedded System Design and design process; Formalism for system design.

Embedded Processors: RISC vs. CISC architectures; ARM processor – processor architecture and memory organization, instruction set, data operations and flow control; SHARC processor – memory organization, data operations and flow control, parallelism within instructions; Input and output devices, supervisor mode, exception and traps; Memory system, pipelining and superscalar execution. Embedded Computing Platform: CPU Bus – Bus protocols, DMA, system bus configurations, ARM bus; Timers and counters, A/D and D/A converters, Keyboards, LEDs, displays and touch screens; Design examples. Embedded Software Analysis and Design: Software design pattern for Embedded Systems; Model programs – data flow graphs and control/data flow graphs; Assembly and linking; Compilation techniques; Analysis and optimization of execution time, energy, power and program size. Embedded System Accelerators: Processor accelerators, accelerated system design

#### **Suggested reference materials:**

1. Computer as Components by Wayne Wolf published by Elsevier Inc
2. ARM System Developer's Guide by Andrew S. Loss published by Elsevier Inc
3. Embedded System Design by Steve Heath published by Elsevier Inc
4. Embedded System design: A unified hardware/software Introduction by Frank Vahid & Tony Givagi published by John Wiley & Sons Inc.