S Y L L A B U S

MASTER OF TECHNOLOGY
(COMPUTER SCIENCE & ENGINEERING)

M. Tech.

M.Tech., (CSE) I Year Exam., 2014 - 2015
M.Tech., (CSE)II Year Exam., 2015 - 2016

JODHPUR NATIONAL UNIVERSITY
JODHPUR

Jodhpur National University, Jodhpur
M. Tech Programme (Computer Science & Engineering)
Teaching / Examination Scheme

I SEMESTER

Subject Code	Subject	Hrs./ Week				Marks		
		L	T	P	Total	Theory Exam/ Viva voce	Internal Assessment	Total
MCS101	Theory of Formal Languages	4	2	-	6	100	50	150
MCS102	Advanced Operating Systems	4	2	-	6	100	50	150
MCS103	Advanced Database Management Systems	4	2	-	6	100	50	150
MCS104	Software Project Management	4	2	-	6	100	50	150
MCS105	Advanced Database Lab	-	-	6	6	50	50	100
	Total	**16**	**8**	**6**	**30**	**450**	**250**	**700**

II SEMESTER

Subject Code	Subject	Hrs./ Week				Marks		
		L	T	P	Total	Theory Exam/ Viva voce	Internal Assessment	Total
MCS201	Advanced Data Structures	4	2	-	6	100	50	150
MCS202	Advanced Computer Networks	4	2	-	6	100	50	150
MCS203	Information Protection & Computer Security	4	2	-	6	100	50	150
MCS204	**Elective**	4	2	-	6	100	50	150
MCS205	Information Protection & Computer Security Lab	-	-	6	6	50	50	100
	Total	**16**	**8**	**6**	**30**	**450**	**250**	**700**

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
MCS301	Fault Tolerant Computing	4	2	-	6	100	50	150
MCS302	**Elective:**	4	2	-	6	100	50	150
MCS303	Seminar	-	-	6	6	50	50	100
	Total	**8**	**4**	**6**	**18**	**250**	**150**	**400**

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
MCS401	Dissertation	-	-	-	-	100	100	200
	Total	**-**	**-**	**-**	**-**	**100**	**100**	**200**

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

Total Marks: 700 + 700 + 400 + 200 = 2000

FIRST SEMESTER

MCS101-THEORY OF FORMAL LANGUAGES (1MCS01)

4L, 2T 3 Hours, 100 Marks

Formal languages and their related automata, Turing machines, type-0 languages, linear bounded automata and CSLs. 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 functionals, verification methods, Lambda calculus and applications.

Suggested reference materials:

1. K. L. P Mishra, N. Chandrasekaran, Theory of Computer Science, PHI
2. Peter Linz, Formal Languages and Automata, Narosa Publication.
3. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.
4. John Martin, Introduction to Languages and The Theory of Computation, Tata McGraw Hill.
5. Deniel I. A. Cohen, Introduction to Computer Theory, Wiley Publication.

MCS102 - ADVANCED OPEARTING SYSTEMS (1MCS02)

4L, 2T 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 / Silberschtz : Operating Systems Concepts, TMH
2. Distributed operating System :Tanenbaum
3. Distributed operating systems: concepts and design: Pradeep Kumar Sinha
4. Distributed systems: concepts and design: George Coulouris, Tim kindberg

MCS103 - ADVANCED DATABASE MANAGEMENT SYSTEM(1MCS03)

4L, 2T 3 Hours, 100 Marks

Overview of DBMS, Transaction Management, concurrency control , failure recovery.

Introduction to distributed data base management systems Semantic Database Models and Systems, Relational Extensions: Design Techniques, Extension Techniques.

Object / Relational Systems: Open ODB, Interface, OSQL, Adapter, Case Study of an ORDBMS, Related Development, Current Product Scenario.

Object-Oriented Database Systems : 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, GenericADT Packages, Language Bindings.

Suggested reference materials:

1. C S R Prabhu,"Object Oriented Data Base Systems" approaches and Architectures, PHI,
2. F. H. Lochousky, DC Tsichritzis"DBMS" NewYork Academic Press.
3. F. H. Lochousky, 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

MCS104 – SOFTWARE PROJECT MANAGEMENT (1MCS04)

4L, 2T 3 Hours, 100 Marks

Introduction to Software Project Management (SPM) :- Importance of SPM. Plans, methods and methodologies, setting objectives. Traditional / Modern SPM practices.

An overview of Project Planning:- Step wise project planning. Selection, identification, analysis, effort estimation, risk and resource allocation.

Activity Planning:- Objectives of activity planning, project schedules and activities. Sequencing, networking, planning models, critical path analysis.

Risk Management:- Categories of risk, framework for dealing with risk. Risk identification, assessment, planning and management.

Resource Allocation:- Nature of resources, identification and scheduling of resources, cost and scheduling sequences.

Software Quality:- Software quality in project planning and its importance. Product & process metrics, quality management systems. Process capability models, techniques for enhancing software quality.

Managing Contracts:- Types of contracts, stages in contract placement, terms of contracts and its management, acceptance.

Overview of PRINCE 2 (Project in Controlled Environments) & Project Management Tools.

Suggested reference materials:

1. Bob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", Tata Mc Graw Hill. (5th Edition)

2. Roger Pressman, Software Engineering A Practitioner's Approach, Tata Mc Graw-Hill Series in Computer Science (6th Edition)

3. Pankaj Jalote, An Integrated Approach to Software Engineering, (Third Edition), Narosa Publication.

SECOND SEMESTER
MCS201-ADVANCED DATA STRUCTURES (2MCS01)

4L, 2T 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. Cormen, Leiserson, Rivest, Stein “Intoduction to Algorithms”, McGraw Hill.
2. Michael T. Goodrich, Roberto Tamassia, Algorithm Design, Wiley Student Edition
3. Motwani and Raghavan "Randomized Algorithms", Cambridge University Press
4. Aho, Hopcraft, Ullman, “The Design and Analysis of Computer Algorithms”, Addison Wesley
5. Horowitz, Sahni, “Fundamentals of Computer Algorithm”, Galgotia
6. Mehlhorn "Data Structures and Algorithms: 1, Searching and Sorting", Springer

MCS202 – ADVANCED COMPUTER NETWORKS (2MCS02)

4L, 2T 3 Hours, 100 Marks

Flow and Congestion Control-Flow and Congestion Control: Window and Rate Based Schemes, Decbit, TCP, ATM ABR, hopby-hop schemes.

Quality of Service :-Quality of Service: in ATM, IETF integrated services model, Differentiated Services Model.

Flow Identification:-Flow Identification: Packet Classifiers and Filters Scheduling.

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

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

Suggested reference materials:

1. Computer Networks: a top-down approach featuring Internet, J F. Kurose & Keith W. Ross
2. Heuring – Computer system Design and Architecture, Pearson Education.
3. Computer Networks: Frozen

MCS203–INFORMATION PROTECTION AND COMPUTER SECURITY (2MCS03)

4L, 2T 3 Hours, 100 Marks

-

Introduction:-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 and Stream ciphers: -Block ciphers: Modes of operation, DFS and its variants, RCS, IDEA, SAFER, FEAL, Blowfish, AES, liner and differential cryptanalysis.
Stream ciphers: Stream ciphers based on liner feedback shift registers, SEAL, unconditional security.

Public- key parameters and Intractable problems:--Public- key parameters: Modular arithmetic, god, 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 and Key exchange :-Public-key encryption: RSA, Rabin and EIGamal schemes, side channel attacks.Key exchange: Diffie-Hellman and MQV algorithms.

Digital signatures and Entity authentication-Digital signatures: RSA, DAD and NR signature schemes, blind and undeniable signatures.

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

Suggested reference materials:

1. **Cryptography & Network Security, William Stallings**
2. **Data Communications & Network Atul Kahate**
3. **Cryptography & Network Security 2E Forouzan**

MCS204– INFORMATION RETRIEVAL (2E01)

4L, 2T 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 IR.

basic idea of parallel and distributed, Searching web, search engines, browsings and met searches, searching through hyperlinks.

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:

1. Modern Information Retrieval, Baeza-yates & Riberio-neto

MODERN COMPILER DESIGN (2E02)

4L, 2T 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: "Ädvanced Compiler Design and Implementation" Margan Kaufmann.
2. Aho Ullaman Sethi " Compiler Construction" Addition Wesley.
3. Holob " Compiler Designing " TMH.

SOFTWARE SYSTEM DESIGN (2E03)

4L, 2T 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

MCS301–FAULT TOLERANT COMPUTING (3MCS01)

4L, 2T 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 .**Fault Tolerance interconnection networks:** Analysis of fault tolerant hardware and software architectures. Case studies of fault tolerant multiprocessor and distributed systems

Suggested reference materials:

1. Israel Koren, C. Mani. Krishna, Fault Tolerant Systems, Elsevier.
2. P. Jalote, “Fault Tolerance in Distributed Systems” Prentice-Hall Inc. 1994,
3. D. K. Pradhan “,Fault-Tolerant Computing, Theory and Techniques”, Prentice-Hall, 1998,
4. Los Alamitos, CA, “Fault-Tolerant Computing, Theory and Techniques and application”, IEEE Computer Society Press.

5. Barry W. Johnson, “Design and Analysis of Fault-Tolerant Digital System”, Addison Wesley.

ADVANCED REAL TIME SYSTEMS (3E01)

4L, 2T 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.

ADVANCED COMPUTER GRAPHICS (3E02)

4L, 2T 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.

DESIGN OF EMBEDDED SYSTEMS (3E03)

4L, 2T 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