EVALUATION SCHEME

M.TECH. 1ST Year 1st Semester (Computer Science & Engineering)
Effective from session 2012-13

S. No.	COURSE CODE	SUBJECT	PERIODS			EVALUATION SCHEME				
						SESSIONAL EXAM			ESE	Subject Total
			L	T	P	CT	TA	Total		
THEORY										
1.	TCS 511	Advanced Data Structures and Algorithms	3	1	-	30	20	50	100	150
2.	TCS 512	Programming Paradigms	3	1	-	30	20	50	100	150
3.	TCS 513	Advanced Computer Architecture	3	1	-	30	20	50	100	150
4.	TCS 514	Advanced Operating Systems	3	1	-	30	20	50	100	150
5.	TCS 515	Mathematical Concepts for Computer Science	3	1	-	30	20	50	100	150
PRACTICAL										
6.	PCS 511	Data Structures and Algorithms (Pr)	-	-	3	-	-	50	50	100
7.	PCS 512	Programming Paradigms (Pr)	-	-	3	-	-	50	50	100
8.	GPP 511	General Proficiency	-	-		-	50	50	-	50
SEMESTER TOTAL			**15**	**5**	**6**	**150**	**150**	**400**	**600**	**1000**

EVALUATION SCHEME

M.TECH. 1ST Year 2nd Semester (Computer Science & Engineering)
Effective from session 2012-13

S. No.	COURSE CODE	SUBJECT	PERIODS			EVALUATION SCHEME				
						SESSIONAL EXAM			ESE	Subject Total
			L	T	P	CT	TA	Total		
THEORY										
9.	TCS 521	Advances in Database Technology	3	1	-	30	20	50	100	150
10.	TCS 522	Mobile Computing Networks	3	1	-	30	20	50	100	150
11.	TCS 523	Information Security	3	1	-	30	20	50	100	150
12.	ECS 52X	Elective-I	3	1	-	30	20	50	100	150
13.	ECS 52Y	Elective-II	3	1	-	30	20	50	100	150
PRACTICAL										
14.	PCS 521	Advances in Database Technology(Pr)	-	-	3	-	-	50	50	100
15.	PCS 522	Mobile Computing Networks (Pr)	-	-	3	-	-	50	50	100
16.	GPP 521	General Proficiency	-	-		-	50	50	-	50
SEMESTER TOTAL			**15**	**5**	**6**	**150**	**150**	**400**	**600**	**1000**

ist of Elective I		**List of Elective II**	
ECS 520	Digital Signal Processing	ECS 525	Green Computing and Communication
ECS 521	Distributed Computing	ECS 526	Reliable Computing
ECS 522	High Performance Computing	ECS 527	Modeling and Simulation
ECS 523	Soft Computing Techniques	ECS 528	Software Verification, Validation & Testing
ECS 524	Software Project Management	ECS 529	Emerging Trends & Technologies

Note: Minimum five students are required to run an elective subject.

EVALUATION SCHEME

M.TECH. 2nd Year 3rd Semester (Computer Science & Engineering)
Effective from session 2012-13

S. No.	COURSE CODE	SUBJECT	PERIODS			EVALUATION SCHEME				
						SESSIONAL EXAM			ESE	Subject Total
			L	T	P	CT	TA	Total		
THEORY										
17.	ECS 63X	Elective –III	3	1	-	30	20	50	100	150
18.	ECS 63Y	Elective –IV	3	1	-	30	20	50	100	150
PRACTICAL										
19.	PCS 631	Dissertation Phase-I	-	-	6	-	-	300	200	500
20.	PCS 632	Seminar	-	-	3	-	-	50	50	100
21.	PCS 633	Creative Problem Solving	-	-	3	-	25	25	25	50
22.	GPP 631	General Proficiency	-	-		-	50	50	-	50
SEMESTER TOTAL			**6**	**2**	**9**	**60**	**115**	**525**	**475**	**1000**

List of Elective III		**List of Elective IV**	
ECS 630	Advanced Data Modeling	ECS 635	Fault Tolerant Systems
ECS 631	Embedded Systems	ECS 636	Green Grid Computing
ECS 632	Reliable Computing	ECS 637	Network Programming
ECS 633	Robotics and Computer Vision	ECS 638	E-Commerce
ECS 634	Software Measurement and Metrics	ECS 639	Business Application

Note:-

1 Minimum five students are required to run an elective subject.

EVALUATION SCHEME

M.TECH. 2nd Year 4th Semester (Computer Science & Engineering)
Effective from session 2012-13

S. No.	COURSE CODE	SUBJECT	PERIODS			EVALUATION SCHEME				
						SESSIONAL EXAM			ESE	Subject Total
			L	T	P	CT	TA	Total		
THEORY										
23.	PCS 641	Dissertation Phase - II	-	-	20	450	-	450	500	950
24.	GPP 641	General Proficiency	-	-	-	-	50	50	-	50
		Total	**-**	**-**	**20**	**450**	**50**	**500**	**500**	**1000**

TCS 511: Advanced Data Structures and Algorithms
M.Tech. Semester –I (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**: 50 Marks**
			Exam.	**: 100 Marks**
			Total	**: 150 Marks**
			Duration of Exam	**: 3 Hrs.**

Prerequisites: Knowledge of basic C, mathematics, problem solving.

Course Outcome:

- Student will develop higher level skills of analysis, creation and evaluation.
- Student can strengthen their ability of data abstraction and problem solving using computers.
- Student can strengthen their ability of data abstraction and problem solving using computers.
- Students can enhance their ability to design and evaluate ADTs.
- Students shall be able to pursue higher end research career in various domains.

Objective: To provide basic data structure concepts and their implementation, testing and maintenance for software systems. And familiarize the students with the design strategies and bounds on the performance of different computer algorithms.

Unit 1: Review of Basic Concepts: Abstract data types, Data structures, Algorithms, Big Oh, Small Oh, Omega and Theta notations, Solving recurrence equations, Master theorems, Generating function techniques.

Unit 2: Advanced Search Structures for Dictionary ADT: Splay trees, Amortized analysis, 2-3 trees, 2-3-4 trees, Red-black trees, Skip lists, Universal hash functions.

Unit 3: Advanced Structures for Priority Queues and Their Extensions: Binomial heaps, Leftist heaps, Skewed heaps, Fibonacci heaps and its amortized analysis, Applications to minimum spanning tree algorithms

Unit 4: Graph Algorithms: DFS, BFS, Bi-connected components, cut vertices, Matching, Network flow.

Unit 5: Lower Bound Theory: Adversary arguments, information theory bounds

Books

1. Mark Allen Weiss, Data Structures and Algorithms in C++, Addison Wesley, 2003.
2. Adam Drozdek, Data Structures and Algorithms in C++, Brooks and Cole, 2001.
3. Aho, Hopcroft and Ullmann, Data structures and Algorithm, Addison Welsey, 1984.
4. A. M. Tenenbaum, Langsam, Moshe J. Augentem, Data Structures using C, PHI Pub.
5. R. B. Patel, Expert Data Structure with C, , 3rd Pub, Khanna Pub. Pvt Ltd.
6. A. V. Aho, J. E. Hopcroft and T. D. Ullman, Data Structures and Algorithms, Original edition, Addison-Wesley, 1999, Low Price Edition.
7. Ellis Horowitz & Sartaj Sahni, Fundamentals of Data Structure , Pub, 1983. AW
8. Horowitz Sahni and Rajasekaran Sanguthevar,Fundamentals of computer algorithms , University press (India) Limited.
9. Robert Kruse, Data Structure and Program design in C , PHI
10. Jean Paul Tremblay, Richard B. Bunt, Introduction to Computer Science- An algorithms approach, 2002, T.M.H.
11. William J. Collins, Data Structure and Standard Template Library, 2003, TMH

TCS 512: Programming Paradigms
M.Tech. Semester –I (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**: 50 Marks**
			Exam.	**: 100 Marks**
			Total	**: 150 Marks**
			Duration of Exam	**: 3 Hrs.**

Prerequisites: knowledge of basic programming language paradigms.

Course Outcome:

- Student will have the knowledge of programming language background: historical perspective, design considerations and evaluation criteria.
- Student will understand major paradigms (procedural, object-oriented, functional, logic, concurrent).
- Student will identify the similarities, differences, strengths and weaknesses of different programming language paradigms (procedural, object oriented, functional, logic, concurrent).
- Student will understand and apply formal methods of programming language syntax and semantics, fundamental programming language elements (data abstraction, expressions, control structures, exception handling, modules, scope, parameter passing).
- Student will be experienced while writing programs using different programming language paradigms.

Objective: To introduce semantics of programming languages and develop skills in describing, analyzing and using the features of programming languages.

Unit 1: Introduction to different paradigms of programming: Imperative - Object Oriented – Functional-Logic.

Unit 2: Imperative Programming - Role of Types: Static and Dynamic Type Checking - Scope rules; Grouping Data and operations.

Unit 3: Information Hiding and Abstract Data Types: Objects, Inheritance, Polymorphism, Templates.

Unit 4: Functional Programming: Expressions and Lists, Evaluation, types, type systems, values and operations, function declarations, lexical scope, lists and programming with lists, polymorphic functions, higher order and Curried functions, abstract data types.

Unit 5: Logic Programming: Review of predicate logic, clausal-form logic, logic as a programming language, Unification algorithm, Abstract interpreter for logic programs, Semantics of logic programs.

Books

1. Ravi Sethi, Programming Languages: Concepts and Constructs, 2nd Edition, Pearson Education Asia.
2. Alfred. V. Aho and Jefferey. D. Ullman, Foundations of Computer Science, Computer Science Press, 1992.
3. Stephen G. Kochan, Programming in C, Third Edition, July 2004, Pearson Education.
4. R. B. Patel, Programming in C, 1st edition Khanna Book Publishing Company Pvt. Ltd, New Delhi, 2008.
5. Kernighan and Ritchie, C Programming Language, 2nd Edition, Prentice Hall, Inc., 1988.
6. Byron S. Gottfried, Programming with C, TMH

TCS 513: Advanced Computer Architecture
M.Tech. Semester –I (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**: 50 Marks**
			Exam.	**: 100 Marks**
			Total	**: 150 Marks**
			Duration of Exam	**: 3 Hrs.**

Prerequisites: Knowledge of computer organization is essential.

Course Outcome:

- Student can develop the operations of pipelines, dynamic scheduling branch predictions, caches.
- Student can develop the operations of virtual memory.
- Student can develop architectures of RISC, Scalar, VLIW Multi core and multi CPU systems.
- Student can compare the performance of different CPU architecture.
- Develop the applications for high performance computing systems.

Objective: To expose students to advanced techniques of computer design such as pipelining, vector processing and multiprocessing.

Unit 1: Overview of von Neumann architecture: Instruction set architecture; The Arithmetic and Logic Unit, The Control Unit, Memory and I/O devices and their interfacing to the CPU; Measuring and reporting performance; CISC and RISC processors.

Unit 2: Pipelining: Basic concepts of pipelining: A Pipelined Data path, data hazards, control hazards, and structural hazards; Techniques for reducing the effects of various hazards.

Unit 3: Hierarchical Memory Technology: Inclusion, Coherence and locality properties; write policies, Cache memory organizations, Techniques for reducing cache misses; Virtual memory organization, mapping and management techniques, memory replacement policies.

Unit 4: Instruction-level parallelism: Concepts of instruction-level parallelism (ILP), Techniques for increasing ILP; Superscalar, superpipelined and VLIW processor architectures; Vector and symbolic processors.

Unit 5: Multiprocessor Architecture: Taxonomy of parallel architectures; Centralized shared-memory architecture, synchronization, memory consistency, interconnection networks; Distributed shared-memory architecture, Cluster computers.
Non von Neumann Architectures: Data flow Computers, Systolic Architectures.

Books

1. W. Stallings, Computer Organization and Architecture: Designing for performance, 4th Ed. PHI, 1996.
2. J. H. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach, 2nd Ed., Morgan Kaufmann, 1996.
3. Kai Hwang, Advanced Computer Architecture: Parallelism, Scalability and Programmability, McGraw-Hill Inc, 1993.
4. D. E. Culler, J. Pal Singh, and A. Gupta, Parallel Computer Architecture: A Hardware/Software Approach, Harcourt Asia Pvt. Ltd., 1999.
5. J. P. Hayes, Computer Architecture and Organization, McGraw Hill.
6. Harvey G. Cragon, Memory Systems and Pipelined Processors, Narosa Publication.
7. V. Rajaraman & C.S.R. Murthy, Parallel Computers, PHI.
8. R. K. Ghose, Rajan Moona & Phalguni Gupta, Foundation of Parallel Processing, Narosa Publications.
9. Kai Hwang and Zu, Scalable Parallel Computers Architecture, McGraw Hill.
10. Stallings W., Computer Organization & Architecture, PHI.

TCS 514: Advanced Operating Systems
M.Tech. Semester –I (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**: 50 Marks**
			Exam.	**: 100 Marks**
			Total	**: 150 Marks**
			Duration of Exam	**: 3 Hrs.**

Prerequisites: Knowledge of C and C++, data structure, computer organization is essential.

Course Outcome:

- Identify the functions of Operating Systems.
- Describe the concepts of process management, scheduling and synchronization.
- Predict and analyse deadlock handling, detection and avoidance.
- Describe the importance of storage management.
- Understand the basics of file systems, I/O systems and disk management.

Objective: To introduce the students about advanced features of Operating System for distributed system.

Unit 1: Review of Operating Systems principles: Synchronization mechanisms, Process deadlocks.

Unit 2: Architecture of Distributed Operating system: Motivation, System Architecture types, issues in distributed operating system, Communication primitives.

Unit 3: Inherent limitations of distributed operating systems: Event ordering. Timestamps. Distributed mutual exclusion. Token and non-token based algorithms. Comparative performance analysis.

Unit 4: Distributed deadlock detection: Deadlock handling strategies, issues in deadlock detection & reevaluation, Control Organization: Centralized distributed & Hierarchical detection algorithms.

Unit 5: Concurrency control: Shared Memory. File Systems. Agreement protocols for handling processor failures. Coordination of processes and related algorithms, Interprocess Communications, Failure handling and recovery mechanisms.

Books

1. Peterson, J.L. & Silbersehatz, A: Operating System Concepts, Addison, Wesley-Reading. . .
2. Brineh, Hansen: Operating System Principles, Prentice Hall of India.
3. Haberman, A.N: Introduction to Operating System Design Galgotia Publication, New Delhi.
4. Hansen, P.B: Architecture of Concurrent Programs, PHI.
5. Shaw, A.C: Logic Design of Operating Systems, PHI.
6. Mukesh Singhal & N.G. Shivaratri: Advanced concepts in operating systems, TMH 2001.
7. A S Tanenbaum : Modern Operating Systems, PHI.
8. A. Silberschatz, P. Galving, G. Gahne : Applied operating system concepts, Wiley.

TCS 515: Mathematical Concepts for Computer Science
M.Tech. Semester –I (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**: 50 Marks**
			Exam.	**: 100 Marks**
			Total	**: 150 Marks**
			Duration of Exam	**: 3 Hrs.**

Prerequisites: knowledge of sets & relations, graph theory, automata, concept of stochastic processes and Markov chains.

Course Outcome:

- Understand the fundamental concepts of Propositional logic and Predicate logic and use truth table method to learn and prove the Rules of Inference.
- Understand and design finite state machines as models of physical systems.
- Derive and solve recurrence relations from general behaviour of systems.
- Understand the concepts of groups, subgroups, rings and integral domain.
- Draw and analyse graphs, trees, cut sets, and transport networks.
- Understand the concept of stochastic processes and Markov chains.

Objective: To introduce the students about the fundamental concepts of mathematical foundation of computer science.

Unit 1: Sets, Relations, Functions, Logic: Propositional logic. Truth tables, Tautologies, Resolution proof system, Predicate logic.

Unit 2: Finite state machines (FSM): FSM as models of physical systems, equivalent machines, FSM as language recognizers, finite state languages & type-3 languages.

Unit 3: Recurrence relations: derivation of recurrence relations from the general behavior of Systems. Solutions of Recurrence relations. Groups, Subgroups, Cyclic Group, Rings, Integral domains, Isomorphism & Homeomorphisms.

Unit 4: Graphs & planar Graphs: Shortest path, Trees & Cut sets, Rooted Trees, Minimum Spanning Trees, and Transport Networks.

Unit 5: Random Variables: Stochastic Processes, Discrete parameter Markov Chains, Limiting state probabilities, State classification, Irreducible Markov Chains, Birth-Death processes.

Books

1. C. L. Liu, Elements of Discrete Mathematics, 1985, McGraw-Hill.
2. Kishore.S.Trivedi, Statistical Models for Computer Science Applications, PHI.
3. Ronald Graham, Donald Knuth and Oren Patashik, Concrete Mathematics: A foundation for Computer Science, 1989, Addison-Wesley.
4. Judith L. Gersting, Mathematical structures for Computer Science, 1993, Computer Science Press.
5. Doerr and Levasseur, Applied discrete structures for Computer Science, (Chicago: 1985, SRA)
6. A. Chetwynd and P. Diggle, Discrete Mathematics, (Modular Mathematics series), 1995, Edward Arnold, London.
7. S. Lipshutz, Schaums Outline series: Theory and Problems of Probability 1982, McGraw-Hill Singapore.
8. B. Kolman and R. C. Busby, Discrete Mathematical Structures, 1996, PHI.
9. Trembley & Manohar, Discrete Mathematical Structures with Applications to Computers, 1995, McGraw-Hill.

PCS 511: Data Structures & Algorithms (Pr)
M.Tech. Semester –I (Computer Science & Engg.)

L	T	P
-	-	3

Class Work : 50 Marks
Exam. : 50 Marks
Total : 100 Marks
Duration of Exam : 3 Hrs.

Prerequisites: Knowledge of C/C++ Programming and basic data structures is essential.

The experiments will be based on the following Paper- TCS 511 **Advanced Data Structures and Algorithms**

Course Outcome:

- Students can learn and polish his/her basics of programming with emphasis on solving real world problems.
- Students will apply object oriented way approach for solving problems.
- Students shall deepen their skills to design and implement ADTs.
- Students shall be able to participate in various coding fests.
- Students shall be able to optimize or re-engineered the already existing solutions of various problems.

PCS 512: Programming Paradigms (Pr)
M.Tech. Semester –I (Computer Science & Engg.)

L	T	P		
-	-	3	**Class Work**	**: 50 Marks**
			Exam.	**: 50 Marks**
			Total	**: 100 Marks**
			Duration of Exam	**: 3 Hrs.**

Prerequisites: Knowledge of C/C++ Programming is essential.

The experiments will be based on the following paper: TCS 512 **Programming Paradigms**

Course Outcome:

- Student will design algorithmic strategy for solving the problem and implement it.
- Student will analyze problems and identify the computing requirements appropriate for its solution like solving quadratic equation.
- Student will develop ability to apply and implement learned algorithm design techniques and data structures to solve problems.
- Student will implement concordance problem and 8 queen problem.
- Student will learn and understand fundamentals object oriented programming using core java.

TCS 521: Advances in Database Technology
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Objective: To expose the students basic to some advanced features of database implementation technology. It also gives a direction to the students about distributed database model implementation.

Prerequisite: Knowledge of DBMS is essential.

Course Outcome:
CO 1- To make student able to understand, designing and develop different type of Databases.
CO 2- To understand the concepts of data replication and Transaction processing in databases.
CO 3- Knowledge on security, recovery and backup techniques in databases.
CO 4- To make student able to understand the concept of distributed databases.
CO 5- To understand the concepts of data warehousing and data mining.

Unit 1: Relational Data Base Design and Architecture: DBMS Architecture, Data base models, Normal Forms Based on Primary Keys, (1NF, 2NF, 3NF & BCNF), Lossless Join and Dependency Preserving Decomposition. Multivalued and Join Dependencies, Template Dependency, Inclusion and Generalized Functional Dependency.

Unit 2: SQL–A Relational Database Language, Data Definition in SQL, View and Queries in SQL, Specifying Constraints ,Indexes, Triggers in SQL.

Unit 3: Concurrency control & Recovery: Transaction Concept and State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, **Concurrency Control Techniques**, Lock-Based Protocols, Timestamp-based Protocols, Validation – based Protocols, Multiversion Schemes, Deadlock Handling, **Recovery System**, Failure Classification, Storage Structure, Recovery and Atomicity, Log-based Recovery, Shadow Paging, Recovery with Concurrent Transactions.

Unit 4: Distributed Data Bases: Distributed database concept, An over view of Client/Server Architecture, Database security, issues, GIS, Mobile databases, overview of Object oriented database & temporal databases.

Unit 5: Data Warehousing & Data mining: The Evolution of Data Warehousing, Today's Development Environment, Types of Data and their Uses, Conceptual Data Architecture, Design Techniques, Logical Architecture. Data Mining: Introduction, data mining, kind of data, Functionalities, interesting patterns, Classification of data mining systems, Major issues. Data Warehouse and OLAP Technology for Data Mining: data warehouse, operational database systems and data warehouses, Architecture, Implementation, development of data cube technology.

Books

1. Korth, Silberschatz, Database System Concepts , 4th Ed., TMH, 2000.
2. Date C. J., An Introduction to Database Systems , 7th Ed., Narosa Publishing, 2004
3. Data Mining, Adriaans, Addison-Wesley Longman.
4. Alex Berson, Stephen Smith, Korth Theorling, Data Mining, TMH.
5. Anahory, Addison-Wesley Longman, Data Warehousing in the Real World.
6. Chanchal Singh, Wiley, Data Mining and Warehousing.

TCS 522: Mobile Computing Networks
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Objective: To expose the students basic to some advanced features of different types of ad hoc networks with case studies. It also gives a direction to the students about distributed system design model & implementation.

Prerequisite: Knowledge of computer networks is essential.

Course Outcome:
CO1- Students will learn about mobile computing concepts.
CO2- Students will learn different routing techniques for the mobile networks.
CO3- To make student able to understand the concepts of location management in distributed environment.
CO4- Students will learn about various types of Mobile IP's (Information protocols) that are used in mobile computing.
CO5- Students will learn about different types of security and failure detection and recovery strategies used in mobile computing.

Unit 1: Overview of Ad Hoc Networks: Why Ad Hoc Networks?, Challenges, and benefits of Mobile Computing, Breakthrough Technology, Wireless Computing, Nomadic Computing, Mobile Computing, Ubiquitous Computing, Pervasive Computing, Invisible Computing, Applications of mobile computing, Wireless and Mobile Computing Models, LAN Protocols: IEEE 802.11/a/g/n & Bluetooth, Data Management Issues. Sensor Networks- Challenges, Architecture, and Applications.

Unit 2: Routing: Taxonomy, Applications, Challenges in Mobile Environments, Hidden and exposed terminal problems, Routing Protocols- Proactive, Reactive, and Hybrid protocols, Dynamic State Routing (DSR), Ad hoc On-Demand Distance Vector (AODV), Destination Sequenced Distance – Vector Routing (DSDV), and Cluster Based Routing Protocol (CBRP), and Temporally Ordered Routing algorithm (TORA).

Unit 3: Distributed location Management: Pointer forwarding strategies, Process communication techniques, socket programming, Remote Procedure Call (RPC), Remote Method Invocation (RMI), client/server programming.

Unit 4: Mobile IP- Problem with Mobility, Terminology, Operation, Tunneling, Data transfer to the mobile system, Transport Control Protocol (TCP) Over wireless- Indirect TCP (I-TCP), Snoop TCP, Mobile TCP (M-TCP), Case Study of Client/Server architecture.

Unit 5: Fault tolerance and Security: Mobile Agents Computing, Security- Issues and Mechanisms, Certificate, Secure Agent Transfer, Timestamp Tamper-proofing, Secure Agent Reception, Host Protection, Providing Security and Integrity to Agent Data and State, Securing Agent Itineraries, Security Architecture, fault tolerance- Issues and Mechanisms, Agent Failure Scenarios, Node (host) Failure Detection and Recovery, Agent Failure Detection and Recovery, Communication Failure Detection and Recovery, Fault Tolerant System-3-Layered Monitor System, transaction processing in Mobile computing environment. Mobile Agent Systems: Aglets, PMADE and Case Study.

Books

1. Charles E. Perkins, Ad hoc Networks, Addison Wesley, 2008.
2. Kazem Sohraby, Daniel Minoli, Taieb Znati, Wireless Sensor Networks: Technology, Protocols, and Applications, Wiley, 2007.
3. Mazliza Othman, Principles of mobile computing and communications, Auerbach Publications, 2007.
4. Uwe Hansmann, Lothar Merk, Martin Nicklous, Thomas Stober, Principles of Mobile computing, 2nd Ed., Wiley, 2006.
5. Daniel Minoli, A Networking Approach to Grid Computing, Wiley, 2004.
6. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff UNIX Network Programming: The Sockets Networking API, Volume 1, Third Edition: Addison Wesley, 2003.
7. Lange, D.B. and Oshima, M., Programming and Deploying Java Mobile Agents with Aglets, 1st Ed., Addison Wesley, 2001.
8. William T. Cockayne, Michal Zyda, Mobile agents, Manning Publication, 2000.
9. Milojicic, D., Douglis, F. and Wheeler R., (ed.), Mobility Processes, Computers and Agents, Addison Wesley, 1999.

TCS 523: Information Security
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Objective : To impart the knowledge of internet security, Digital signature, Cryptography.

Prerequisite: Knowledge of cryptography, e-commerce is essential.

Course Outcome:

CO1: Students will learn about information systems, distributed information systems, various kinds of threats and attacks possible on information systems and principles of securing the information on systems.

CO2: Students will learn about the security threats to e-commerce, about the e-transactions concepts, and about biometric systems.

CO3: Students will learn about using cryptographic techniques, firewalls and fingerprint scanners to secure the information.

CO4: Students will learn about securing the network through various techniques.

CO5: Students will learn about the laws regarding the information security.

Unit-1: History of Information Systems and its Importance, basics, Changing Nature of Information Systems, Need of Distributed Information Systems, Role of Internet and Web Services, Information System Threats and attacks, Classification of Threats and Assessing Damages, Security in Mobile and Wireless Computing- Security Challenges in Mobile Devices, authentication Service Security, Security Implication for organizations, Laptops Security, Basic Principles of Information Security, Confidentiality, Integrity Availability and other terms in Information Security, Information Classification and their Roles.

Unit-2: Security Threats to E Commerce, Virtual Organization, Business Transactions on Web, E-Governance and EDI, Concepts in Electronics payment systems, E Cash, Credit/Debit Cards, Physical Security- Needs, Disaster and Controls, Basic Tenets of Physical Security and Physical, Entry Controls, Access Control- Biometrics, Factors in Biometrics Systems, Benefits, Criteria for selection of biometrics, Design Issues in Biometric Systems, Interoperability Issues, Economic and Social Aspects, Legal Challenges

Unit-3: Model of Cryptographic Systems, Issues in Documents Security, System of Keys, Public Key Cryptography, Digital Signature, Requirement of Digital Signature System, Finger Prints,
Firewalls, Design and Implementation Issues, Policies

Unit-4: Network Security- Basic Concepts, Dimensions, Perimeter for Network Protection, Network Attacks, Need of Intrusion Monitoring and Detection, Intrusion Detection Virtual Private Networks- Need, Use of Tunneling with VPN, Authentication Mechanisms, Types of VPNs and their Usage, Security Concerns in VPN

Unit-5: Security metrics- Classification and their benefits, Information Security & Law, IPR, Patent Law, Copyright Law, Legal Issues in Data mining, Security, Building Security into Software Life Cycle Ethics- Ethical Issues, Issues in Data and Software Privacy, Cyber Crime Types & overview of Cyber Crimes

Books

1. Schou, Shoemaker, " Information Assurance for the Enterprise", Tata McGraw Hill
2. Furnell, "Computer Insecurity", Springer

ECS 520: Digital Signal Processing
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Objective: To acquaint the students with the concepts, algorithms and applications of adaptive signal processing in wireless communication systems.

Prerequisite: Knowledge of mathematics, digital logics is essential.

Course Outcome:
CO1: To make students familiar with Digital signal and processing.
CO2: Students will learn about various signals and their classifications.
CO3: To make students able to understand various transforms.
CO4: Students will learn about the applications of various transforms.
CO5: Design of various digital filters.

Unit 1: Systems and signal processing, classifications of signals, concept of frequency in continuous time and discrete time signals. Analog to digital and digital-to-analog conversion, Discrete time signals, Discrete time systems, LTI systems, difference equations, implementation of discrete time systems.

Unit 2: Z- transform and its Applications: Z Transform, properties of Z-transform, Inversion of Z transform, applications of Z transform.

Unit 3: Discrete Fourier Transform(DFT), properties of DFT, Linear filtering methods based on the DFT, frequency analysis of signals using the DFT.

Unit 4: Fast Fourier transform and its applications: FFT algorithms (Radix 2 FFT) algorithm, Implementation of Discrete time systems, Structures for FIR systems, direct form structure, Cascade form structure, parallel form, structures for IIR systems, cascade, direct form and parallel form structures.

Unit 5: Design of Digital Filters: Design of IIR filters, Bilinear transformation and impulse invariance method, Matched Z transformation design of FIR filters with different methods. Design of FIR filters using windows.

Books

1. John G. Proakis and Dimitris G. Manolakis, Digital Signal Processing, PHI
2. Oppenheim & Schaffer, Digital Signal Processing, PHI.
3. Rabiner & Gold, Digital Signal Processing applications.
4. S.K., Mitra, Digital Signal Processing, TMH.
5. S. Salivayhan, A Vallavraj, C. Gnanapriya, Digital Signal Processing , TMH.

ECS 521: Distributed Computing
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P
3	1	-

Class Work	**:50 Marks**
Exam.	**:100 Marks**
Total	**:150 Marks**
Duration of Exam	**: 3 Hrs**

Objective: To impart knowledge of distributed Computing and Distributed Environments and case study.

Prerequisite: Knowledge of advance operating system is required.

Course Outcome:
CO1: Students will learn about the concepts of distributed computing.
CO2: Students will learn about distributed file systems, distributed processes and synchronizing them.
CO3: Students will learn about various simulation concepts of distributed computing
CO4: To give the students an overview of High performance computing , grid computing and randomization.
CO5: To make students able to understand the various types of distributed environments.

Unit 1: Fundamentals of Distributed Computing: Architectural models for distributed and mobile computing systems. Basic concepts in distributed computing such as clocks, message ordering, consistent global states, and consensus. Basic Algorithms in Message: Passing Systems, Leader Election in Rings, and Mutual Exclusion in Shared Memory, Fault-Tolerant Consensus, Causality and Time. Message Passing: PVM and MPI.

Unit 2: Distributed Operating Systems and network operating systems: Distributed File systems. Client/server model for computing, common layer application protocols (RPC, RMI, streams), distributed processes, network naming, Group Communication, distributed synchronization and distributed object-based systems.

Unit 3: Simulation: A Formal Model for Simulations, Broadcast and Multicast, Distributed Shared Memory, Fault-Tolerant Simulations of Read/Write Objects Simulating Synchrony, Improving the Fault Tolerance of Algorithms, Fault-Tolerant Clock Synchronization.

Unit 4: Advanced Topics: Randomization, Wait-Free Simulations of Arbitrary Objects, and Problems Solvable in Asynchronous Systems, Solving Consensus in Eventually Stable Systems, High Performance Computing-HPF, Distributed and mobile multimedia systems. Adaptability in Mobile Computing, Grid Computing and applications.

Unit 5: Distributed Environments: Current systems and developments (DCE, CORBA, JAVA).
Case study- Distributed information searching on the network- Mobile Agent Approach.

Books

1. Hagit Attiya, Jennifer Welch, Distributed Computing: Fundamentals, Simulations, and Advanced Topics, 2nd Edition, March 2004.
2. Mullendar S. Distributed Systems, 2nd Ed. Addison, Wesley 1994.
3. Tannenbaum, A. Distributed Operating Systems, Prentice Hall 1995.
4. Helal, Abdelsalam A. et al. Anytime, Anywhere Computing: Mobile Computing Concepts and Technology, Kluwer Academic Publishers 1999.
5. George Coulouris, Jean Dollimore and Tim Kindberg, Distributed Systems: Concepts and Design Third Edition Addison-Wesley, Pearson Education, 2001.
6. Cay S Horstmann and Gary Cornell, Java 2 Vol I and II-Sun Micro Systems-2001

ECS 522: High Performance Computing
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Objective: To expose students to advanced techniques of computer design such as Parallel Architectures, Parallel Algorithms, Parallel Programming, Compiler Transformations and Performance Evaluation.

Prerequisite: Knowledge of Computer Architecture is essential.

Course Outcome:

CO1: Students will acquire the knowledge of Parallelism and pipelined processors and various other processors.
CO2: Students will learn about the parallel architectures and their memory concepts.
CO3: To acquire the knowledge of various parallel algorithms for computational problems.
CO4: To make students able to understand various parallel programming concepts.
CO5: To make student able to understand the compiler transformations and metrics for evaluating the performance of various parallel and other architectures.

Unit 1: Historical progression leading to current state : types of parallelism including temporal, data and functional. Instructional level parallelism – pipelined processors – super scalar processors – VLIW processors – multithreaded processors – proposed future processors including trace, multiscalar and super flow – case studies.

Unit 2: Parallel architectures: Classification – inter connection networks – vector computers – shared memory parallel computers – cache coherence – distributed shared memory parallel computers – message passing parallel computers – cluster of workstations.

Unit 3: Parallel Algorithms: Models of parallel computation including PRAM, combinational circuits, inter connection networks, BSP-Log P – design and analysis of algorithms for a vide variety of computational problems.

Unit 4: Parallel Programming: Models including message passing – shared memory data parallel – message passing libraries – including PVM and MPI – High performance Fortran (HPF)

Unit 5: Compiler Transformations and Performance evaluation: Dependence analysis – loop transformations – transformations for parallel computers including data layouts, computational and communication optimization. Performance Metrics –performance lows – scalability – performance measurement.

Books

1. Selim G.Akl – The design and analysis of parallel algorithms – Prentice Hall International Inc, 1989.
2. Hwang K. Briggs F.A. – Computer Architecture and parallel processing – McGraw Hill – 1985.
3. Angel L.Decegama – The technology of parallel processing, parallel processing architecture and VLSI Hardware Vol I – Prentice Hall Engle wood cliffs new Jersey 1989.
4. Michael J.Quinn, Parallel computer theory and practice McGraw Hill 2nd Edition 1994.
5. V.Rajaraman and C.Siva Ram Murthy, Parallel computers architecture and programming, Prentice Hall of India, India 2000.
6. C. Siva Ram Murthy, K.N. Balasubramanya Murthy and A. Srinivas New Parallel Algorithms for Direct Solution of Linear Equations – John Wiley & Sons Inc, USA 2001.
7. K. Hwang and Z.Xu, Scalable Parallel Computing: Technology, Architecture and Programming, WCB/McGraw –Hill Inc., USA 1998.

ECS 523 : Soft Computing Techniques
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisites - Knowledge of basic understanding of problem solving, logic gates is essential.

Course Outcome:

- Implement numerical methods in soft computing.
- Explain the fuzzy set theory.
- Discuss the neural networks and supervised and unsupervised learning networks.
- Comprehend neuro fuzzy modeling.
- Apply derivative based and derivative freeoptimization.

Objectives: Soft Computing refers to a collection of computational techniques in computer science, artificial intelligence and engineering disciplines which attempt to study, model and analyze complex problems - those for which more conventional methods have not yielded low cost, analytic and complete solutions.

Unit 1: Introduction to Genetic Algorithm: Genetic Operators and Parameters, Genetic Algorithms in Problem Solving, Theoretical Foundations of Genetic Algorithms, Implementation Issues.

Unit 2: Artificial Neural Networks: Neural Model and Network Architectures, Competitive Networks, Hopfield Network, Computing with Neural Nets and applications of Neural Network.

Unit 3: Learning: Perception Learning, Supervised& Unsupervised Hebbian Learning, Backpropagation, Associative Learning.

Unit 4: Fuzzy Logics: Introduction to Fuzzy Sets, Operations on Fuzzy sets, Fuzzy Relations, Fuzzy Measures, Applications of Fuzzy Set Theory to different branches of Science and Engineering.

Unit 5: Knowledge discovery in databases: Data mining and web mining using soft computing techniques. Soft computing approaches to information systems project management.

Books

1. M. Mitchell, An Introduction to Genetic Algorithms, Prentice-Hall, 1998.
2. D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.
3. S. V. Kartalopoulos, Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications, IEEE Press - PHI, 2004.
4. S. Rajasekaran & G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications, PHI, 2003.
5. S. N. Sivanandam & S. N. Deepa, Principles of Soft Computing, Wiley - India, 2007.

ECS524: Software Project Management
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisites – Knowledge of software engineering is essential.
Course Outcome:

- Introduction about SPN & using it's concepts in project development.
- To understand the concept of software management, disciplines and control.
- Understand the software management disciplines.
- To understand the Project control and process instrumentation.
- To understand the Life cycle phases of Software management.

Objective: To introduce the concepts of software management, Disciplines and control.

Unit 1: Conventional Software management: Evolution of software economics. Improving software economics: reducing product size, software processes, team effectiveness, automation through. Software environments. Principles of modem software management.

Unit 2: Software management Process: Framework,: Life cycle phases- inception, elaboration, construction and training phase.

Unit 3: Artifacts of the process: artifact sets, management artifacts, engineering artifacts, and pragmatics artifacts. Model based software architectures. Workflows of the process. Checkpoints of the process.

Unit 4: Software Management Disciplines: Iterative process planning. Project organizations and responsibilities. Process automation.

Unit 5: Project control and process instrumentation: core metrics, management indicators, life cycle expectations. Process discriminants.

Books

1. Software Project Management, Walker Royce, Addison Wesley, 1998.
2. Project Management 2/e, Maylor.
3. Managing the Software Process, Humphrey.
4. Managing Global Software Projects, Ramesh, TMfH,2001.

ECS 525 Green Computing and Communication
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisites – Knowledge of computer system is essential

Course Outcome:

- To understand the concept ofPower Management/Control Units.
- To understand the concept of Optimization of Idle and Cooling Power.
- To understand the concept of Truths and Myths of Embedded Computing.
- To understand the concept of Energy costs in Internet-scale Systems
- To understand the concept of Energy-efficiency issues in Distributed Cyber-Physical Systems.

Objective: To impart the knowledge of how to save power when using computer and it's components

Unit 1 : Introduction to Green computing, Introductions and backgrounds of students. ACPI and power states' Nehalem, Power Management/Control Units (PMU/PCU), Thermal topics in Data Centers, The Case for Energy Proportionality, Low Power Server CPUs

Unit 2: Improving Energy Efficiency, and possibilities on how to Green the Internet.Green Server Design and the cost comparison of shipping a DVD/Blu-Ray or streaming, Coordination of power management strategies, Joint Optimization of Idle and Cooling Power,

Unit 3: Thermodynamics and Virtual Power, Truths and Myths of Embedded Computing" and Control Systems.

Unit 4: Control Systems view and coordination within data centers,vGreen: Energy Efficient Computing in Virtualized Environments & Security/Energy TradeoffEnergy costs in Internet-scale Systems, Blink and NapSAC.

Unit 5: Energy-efficiency issues in Distributed Cyber-Physical Systems - Model Based CPS design. Energy Harvesting and Variable Power Resources, Energy Neutral Systems.

Books

1. W.C.Y.Lee, "Mobile Communications Engineering: Theory and applications, Second Edition, McGraw-Hill International, 1998.
2. Stephen G. Wilson, " Digital Modulation and Coding", Pearson Education, 2003. 4. Narayana V. and Sreenivasan, N.S. Quality Management - Concepts and Tasks, New Age International 1996. 5. Zeiri. "Total Quality Management for Engineers Wood Head Publishers, 1991.
3. Principles of Mobile Communications – Gordon L. Stuber, Springer International 2nd Edition, 2007.

ECS 526:Reliable Computing

M.Tech. Semester –II (Computer Science & Engg.)

L T P		
3 1 0	**Class Work**	**:50 Marks**
	Exam.	**:100 Marks**
	Total	**:150 Marks**
	Duration of Exam	**: 3 Hrs.**

Prerequisites – Knowledge of computer architecture is essential.

Course Outcome:

- To understand the concept of Reliable System.
- To understand the concept of Fault and fault tolerance method.
- To understand the concept of Real Time systems and Performance measures for real time systems.
- To understand the concept of different Protocols for Real Time Communications.
- To understand the concept of Packages, Exception handling, Over loading and Generics, Multitasking, Task scheduling.

Objective: To acquaint the students with the reliability, Fault tolerance, and Real Time system.

Unit 1: Reliability: Definition, System reliability, Parameter values, Reliability models for hardware redundancy – Testing: Various testing methods.

Unit 2: Fault tolerance: Definition, Fault types, Detection, Redundancy, Data diversity, Reversal checks, Byzantine failures, Integrated failure handling.

Unit 3: Real Time system: Introduction, Characterizing real time systems, Performance measures for real time systems, Estimating Program run times, Task management and Scheduling – Uni-processor, Fault tolerant scheduling.

Unit 4:Real Time Communications: Protocols, Contention based, token based, Stop and go multihop, the polled bus, hierarchical round robin, deadline based, and fault tolerance routing, Distributed delay constrained method, Dependable real time channels, recovery approach, Establishing real time channels.

Unit 5: Programming Languages and Tools: Desired Language Characteristics, Data typing, control structures, Hierarchical decomposition, Packages, Exception handling, Over loading and Generics, Multi tasking, Task scheduling, Timing specification., Flex, Euclid, Environments, Run time support.

Books

1. C. M. Krishna and K. G. Shin, 'Real time Systems', McGraw Hill International Edition, 1997.
2. C. Siva Ram Murthy and G. Manimaran, 'Resource Management in Real Time Systems and Networks', the MIT Press, 2001.
3. Phillip A. Laplante, 'Real-Time Systems Design and Analysis – An Engineers Hand book', Printice Hall India, 3rd edition, 1997.

ECS 527 Modeling and Simulation
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisites – Knowledge of embedded system is essential.

Course Outcome:

- To understand the concept of role of simulation.
- To understand the concept of Deterministic, Stochastic, Continuous and Discrete systems.
- To understand the concept of Statistical Tools and Techniques.
- To understand the concept of single and multiserver queues, network of queues.
- To understand the concept of Biological and Sociological System Simulation.

Objective: To acquaint the students with principles of modeling & simulation and also familiarize students with the simulation based system design & implementation.

Unit 1: Introduction: System models and role of simulation. Entities, Attributes, States and Activities.

Unit 2: Types of Systems: Deterministic, Stochastic, Continuous and Discrete systems. Steps in simulation studies.

Unit3: Statistical Tools and Techniques: generation of pseudorandom numbers, random variate generation for uniform, Poisson and normal distributions, sampling and estimation, maximum likelihood estimation, confidence intervals and hypothesis testing, stochastic processes and Markov models.

Unit 4: Simulation of Inventory and Queuing Systems: single and multiserver queues, network of queues. Modeling and performance evaluation of computers and computer communication networks.

Unit 5: Biological and Sociological System Simulation: Verification and validation of simulation models - input/output validation, sensitivity analysis, performance measures and their estimation. Length of simulation run, Variance reduction Techniques.

Books

1. Russell Schwartz, Biological Modeling and Simulation: A Survey of Practical Models, Algorithms and Numerical Methods, PHI Learning Pvt. Ltd. New Delhi, 2009.
2. Karian, Z.A., and Dudewicz, E. J., Modern Statistical Syatems and GPSS Simulation, 2nd Ed., CRC Press.
3. Banks, J., Carson, L.S, Nelson, B.L and Nicol, D.M., Discrete Event System Simulation, 3rd Ed., Pearson Education.
4. Law, A.M. and Kelton, W.D., Simulation, Modeling and Anlysis, 3rd Ed., Tata McGraw-Hill.

ECS 528 : Software Verification, Validation & Testing
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs.**

Prerequisites - Knowledge of software engineering is essential.

Course Outcome:

- To understand the concept of Error, Fault, Failure, Incident, Test Cases, Testing Process.
- To understand the concept of Equivalence Class Testing, Decision Table Based Testing.
- To understand the concept of Data Flow Testing, Mutation testing, Levels of Testing, Risk Analysis, Regression Testing.
- To understand the concept of Class Testing, GUI Testing, Object Oriented Integration and System Testing.
- To understand the concept of Static Testing Tools, Dynamic Testing Tools.

Objective: To impart the knowledge of software design, verify, validate and testing.

Unit 1: Introduction: What is software testing and why it is so hard?, Error, Fault, Failure, Incident, Test Cases, Testing Process, Limitations of Testing, No absolute proof of correctness, Overview of Graph Theory.

Unit 2: Functional Testing: Boundary Value Analysis, Equivalence Class Testing, Decision Table Based Testing, Cause Effect Graphing Technique.

Unit 3: Structural Testing: Path testing, DD-Paths, Cyclomatic Complexity, Graph Metrics, Data Flow Testing, Mutation testing.Testing Activities:Unit Testing, Levels of Testing, Integration Testing, System Testing, Debugging, Domain Testing.Reducing the number of test cases: Prioritization guidelines, Priority category, Scheme, Risk Analysis, Regression Testing, and Slice based testing

Unit 4: Object Oriented Testing: Issues in Object Oriented Testing, Class Testing, GUI Testing, Object Oriented Integration and System Testing.

Unit 5: Testing Tools: Static Testing Tools, Dynamic Testing Tools, and Characteristics of Modern Tools and Implementation with example. Advanced topics in software testing: web based testing, Client/Server testing, Automated test cases generation, Regular expression and FSM based testing.

Books

1. William Perry, Effective Methods for Software Testing , John Wiley & Sons, New York, 1995.
2. Cem Kaner, Jack Falk, Nguyen Quoc, Testing Computer Software , Second Edition, Van Nostrand Reinhold, New York, 1993.
3. Boris Beizer, Software Testing Techniques , Second Volume, Second Edition, Van Nostrand Reinhold, New York, 1990.
4. Louise Tamres, Software Testing , Pearson Education Asia, 2002
5. Roger S. Pressman, Software Engineering – A Practitioner's Approach , Fifth Edition, McGraw-Hill International Edition, New Delhi, 2001.
6. Boris Beizer, Black-Box Testing – Techniques for Functional Testing of Software and Systems , John Wiley & Sons Inc., New York, 1995.
7. K.K. Aggarwal & Yogesh Singh, Software Engineering , New Age International Publishers, New Delhi, 2003.
8. Marc Roper, Software Testing , McGraw-Hill Book Co., London, 1994.
9. Gordon Schulmeyer, Zero Defect Software , McGraw-Hill, New York, 1990.
10. Watts Humphrey, Managing the Software Process , Addison Wesley Pub. Co. Inc., Massachusetts, 1989.

ECS 529: Emerging Trends & Technologies
M.Tech. Semester –II (Computer Science & Engg.)

L T P		
3 1 0	**Class Work**	**:50 Marks**
	Exam.	**:100 Marks**
	Total	**:150 Marks**
	Duration of Exam	**: 3 Hrs.**

Prerequisites – Knowledge of existing technologies and their reengineering is essential.

Course Outcome:

- Overview of new related to technology management.
- Study of Intellectual property rights.
- Concepts on managerial productivity and effectiveness for organization.
- Understanding the research, development and innovation trends in modern technology.
- To understand the concept of Technology Forecasting Technology Mapping and Technology Strategy Development.

Objective: To provide adequate knowledge of technology management, management of R&D, intellectual property rights.

Unit 1

Introduction to Technology Management, Business Strategy for New Technologies: Adding value, gaining competitive advantage, timing and capability development, Technology Forecasting: Techniques of Forecasting, Technology Forecasting-Relevance, Strategic alliance and Practicality, and Technology Transfer.

Unit 2

Management of Research, Development and Innovation: Technology Mapping, Comparison of type of R&D Projects and development approaches- Radical platform and Incremental Projects, Innovation process.

Unit 3

Management of Intellectual Property Rights: Strategic value of patents, trade secrets and licensing.

Unit 4

Managing Scientists and Technologists: Identification, Recruitment, Retention, Team Work and Result orientation. Investment in Technology, Management Roles and Skills for New Technology

Unit 5

Technology for Managerial Productivity and Effectiveness, Just-in-Time, Venture Capital & Technology Development, Practice Tasks: Technology Forecasting and Technology Mapping, Technology Strategy Development ,Exercise on Just-in-timeCases on Venture Capital

Reference Books:

1. Technology and Management: By Cassell Educational Ltd., London
2. Management of High Technology Research and Development: by John Humbleton Elsevier
3. Strategic Management: by Charles W.L. Hill/Gareth R. Jones, Houghton Mifflin Co.
4. R&D Management: by S.A. Bergn, Basil Blackwell Inc.
5. Innovation and Entrepreneurship in Organisations: by Richard M. Burton & Borge Obel Elsevier.
6. Innovation and Entrepreneurship in Organisations: by Spyros Maksidakis & Steven C Wheelwright, John Wiley & Sons.
7. New Product Management: by C Marle Crawford, Irwin, USA.
8. Just-in-time: by David Hutchin, Gower, technical Press.
9. Management of Technology: Tarek M. Khalil, McGraw-Hill Book Co.
10. Managing Engineering and technology: Daniel L. Babcock, PHI.

PCS 521: Advances in Databases Technology (Pr)
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
-	-	3	**Class Work**	**:50 Marks**
			Exam.	**:50 Marks**
			Total	**:100 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisites - Knowledge of SQL is essential.

Course Outcome:

- Understand the basic database creation in SQL.
- Understand the insert, delete, modify, createqueries in SQL.
- To understand the queries about Join Dependencies.
- To understand the Constraints,Indexes in SQL.
- To understand the View, Triggers in SQL.

The experiments will be based on the following paper: TCS 521 **Advances in Databases Technology**

PCS522: Mobile Computing Networks (Pr)
M.Tech. Semester –II (Computer Science & Engg.)

L	T	P		
-	-	3	**Class Work**	**:50 Marks**
			Exam.	**:50 Marks**
			Total	**:100 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisites - Knowledge of java and socket programming is essential.
Course Outcome:

- Create theRemote Procedure Call programs in java programming.
- Create the Method Invocation programs in java programming.
- Create thesocket programming in java programming.
- Create theclient/server programming in java programming.
- Create the TCP/IP in java programming.

The experiments will be based on the following paper:TCS 522 **Mobile Computing Networks**

ECS 631: Embedded Systems
M.Tech. Semester –III(Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisite: Knowledge of assembly language, Artificial intelligence is essential.

Course Outcome:

- Student will learn various micro-controller architecture.
- Student will get knowledge about the technical aspects of embedded systems.
- Student will learn about the advanced features of CPU architecture.
- Student will learn about the hardware and software aspects of Motorola MC-68 HII family architecture and ARM processor.
- Student will gain knowledge about various features of real time operating systems

Objective: To impart the knowledge of micro computer systems.

Unit 1: The Concepts of Embedded System Design: Embedded microcontroller cores, embedded memories, examples of embedded systems.

Unit 2: Technological Aspects of Embedded System: interfacing between analog and digital blocks, signal conditioning, Digital signal processing, subsystem interfacing, interfacing with external systems, user interfacing, Design tradeoffs due to process compatibility, Thermal consideration etc.
Software aspects of embedded systems: real time programming languages and operating systems.

Unit 3: CPU Architecture: registers, instruction sets addressing modes Loop timing, timers, Interrupts; Interrupt timing, I/O Expansion, I2C Bus Operation Serial EEPROM, Analog to digital converter, UART Baud Rate-Data Handling-Initialisation, Special Features - serial Programming-Parallel Slave Port.

Unit 4: Motorola MC68H11 Family Architecture: Registers, Addressing modes Programs. Interfacing methods parallel I/O interface, Parallel Port interfaces, Memory Interfacing, High Speed I/o Interfacing, Interrupts-interrupt service routine-features of interrupts-Interrupt vector and Priority, timing generation and measurements, Input capture, Output compare, Frequency Measurement, Serial I/o devices RS.232, RS.485. Analog Interfacing, Applications. ARM processors.

Unit 5: Real Time Operating System: Task and Task States, tasks and data, semaphores and shared Data Operating system Services-Message queues-Timer Function-Events-Memory Management, Interrupt Routines in an RTOS environment, basic design Using RTOS.

Books

1. David E Simon, An embedded software primer , Pearson education Asia, 2001.
2. John B Peat man Design with Microcontroller , Pearson education Asia, 1998.
3. Jonartthan W. Valvano Brooks/cole Embedded Micro computer Systems. Real time Interfacing , Thomson learning 2001.
4. Burns, Alan and Wellings, Andy, Real-Time Systems and Programming Languages , Second Edition. Harlow: Addison-Wesley-Longman, 1997.
5. Raymond J.A. Bhur and Donald L.Bialey, An Introduction to real time systems: Design to networking withC/C++,PrenticeHallInc.NewJersey,1999.

ECS 632: Reliable Computing
M.Tech. Semester –III(Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisite: Knowledge of basic computing devices, concepts operating systems and programming languages is essential.

Course Outcome:

- Student will get knowledge about the technical aspects of reliability in computing devices.
- Student will learn about the fault detection and correction techniques.
- Student will learn about the aspects of real time systems.
- Student will gain knowledge about various features of real time operating systems.

Objective: To acquaint the students with the reliability, Fault tolerance, and Real Time system.

Unit 1: Reliability: Definition, System reliability, Parameter values, Reliability models for hardware redundancy Testing: Various testing methods.

Unit 2: Fault tolerance: Definition, Fault types, Detection, Redundancy, Data diversity, Reversal checks, Byzantine failures, Integrated failure handling.

Unit 3: Real Time system: Introduction, Characterizing real time systems, Performance measures for real time systems, Estimating Program run times, Task management and Scheduling – Uni-processor, Fault tolerant scheduling.

Unit 4: Real Time Communications: Protocols, Contention based, token based, Stop and go multihop, the polled bus, hierarchical round robin, deadline based, and fault tolerance routing, Distributed delay constrained method, Dependable real time channels, recovery approach, Establishing real time channels.

Unit 5: Programming Languages and Tools: Desired Language Characteristics, Data typing, control structures, Hierarchical decomposition, Packages, Exception handling, Over loading and Generics, Multi tasking, Task scheduling, Timing specification., Flex, Euclid, Environments, Run time support.

Books :

1. C. M. Krishna and K. G. Shin, ‘Real time Systems’, McGraw Hill International Edition, 1997.
2. C. Siva Ram Murthy and G. Manimaran, ‘Resource Management in Real Time Systems and Networks’, The MIT Press, 2001.
3. Phillip A. Laplante, ‘Real-Time Systems Design and Analysis – An Engineers Hand book’, Printice Hall India,IIIedition,1997.

ECS 633: Robotics and Computer Vision
M.Tech. Semester –III (Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisite: Knowledge of calculus, dynamics, kinematics and linear controls is essential.

Course Outcome:

- Student will get knowledge about Industrial robots: Structure and applications.
- Student will get knowledge about Robot kinematics, coordinate frames and Jacobian matrices.
- Student will get knowledge about Robot dynamics.
- Student will get knowledge about Control systems for motion control and control of interaction forces.
- Student will get knowledge about Robot vision.

Objective: The course introduces the fundamentals of robotics dynamics, its features and performance, controller techniques, and image analysis for obstacle avoidance.

Unit 1: Robot Anatomy Arm Geometry: Direct & Inverse Kinematics Problem. Arm Dynamics Alembert Equations of Motion, Synthesis of elements with movalulity constraints, manipulations-trajectory planning, joint interpolated trajectories.

Unit 2: Control of Robot Manipulation: computed torque technique sequencing & adaptive control, resolved motion control Moluie Robots.

Unit 3: Robot sensing: Range & Proximity & Higher-Level vision, illumination techniques, Imaging Geometry, Segmentation Recognition & Interpretation.

Unit 4: Robot Programming Language: Characteristics of Robot Level & Task Level languages.

Unit 5: Robot intelligence: State Space search, Robot learning, Robot Task Planning, Knowledge Engineering.

Books

1. K.S Fu R.C . CSG Lee-Robotics Control, Sensing, Vision & Intelligence, McGraw-Hill.
2. M.P. Groover, M.Weins, R.N. Nagel, N.C. Odrey –Industrial Robotics,McGraw Hill
3. Andrew C.Straugard-Robotics & AI,PHI
4. S. Sitharama Iyengar, Alberto Elefes-Autonomous Mobile Robots Control, Planning & Architecture, IEEE Computer Society Press

ECS 634: Software Measurement and Metrics

M.Tech. Semester –III(Computer Science & Engg.)

L	T	P		
3	1	0	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisite: Knowledge of software engineering is essential.

Course Outcome:

- Student will get knowledge about mathematics, science, and engineering.
- Student will get the ability to design and conduct experiments, as well as to analyze and interpret data.
- Student will get the ability to function on multi-disciplinary teams.
- Student will get the ability to identify, formulate, and solve engineering problems.
- Student will get the ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

Objective : To impart the knowledge of software measurement for different kind of systems.

Unit 1: Basics of Measurement: Measurement in everyday life, measurement in software engineering, scope of software metrics, representational theory of measurement, measurement and models, measurement scales, meaningfulness in measurement, goal-based framework for software measurement, classifying software measures, determining what to measure, software measurement validation, empirical investigation, types of investigation, planning and conducting investigations.

Unit 2: Software: Metrics data collection and analysis: What is good data, how to define the data, how to collect the data, how to store and extract data, analyzing software-measurement data, frequency distributions, various statistical techniques.

Unit 3: Measuring internal product attributes: Measuring size, aspects of software size, length, functionality and complexity, measuring structure, types of structural measures, control-flow structure, modularity and information flow attributes, data structures. Measuring external product attributes: Modeling software quality, measuring aspects of software quality, software reliability, basics of software reliability, software reliability problem, parametric reliability growth models, predictive accuracy, recalibration of software-reliability growth predictions, importance of operational environment, wider aspects of software reliability.

Unit 4: Metrics for object-oriented systems: The intent of object-oriented metrics, distinguishing characteristics of object-oriented metrics, various object-oriented metric suites LK suite, CK suite and MOOD metrics. Dynamic Metrics: Runtime Software Metrics, Extent of Class Usage, Dynamic Coupling, Dynamic Cohesion, and Data Structure Metrics.

Unit 5: Metrics for component-based systems: The intent of component-based metrics, distinguishing characteristics of component-based metrics, various component-based metrics. Resource measurement: Measuring productivity, teams, tools, and methods.

Books

1. Software Metrics: A rigorous and Practical Approach by Norman E. Fenton and Shari Lawrence Pfleeger, International Thomson Computer Press (1997) 2nd ed.
2. Applied Software Measurement by Capers Jones, McGraw Hill (2008).
3. Object-Oriented Software Metrics by Mark Lorenz, Jeff Kidd, Prentice Hall (1994).
4. Practical Software Metrics For Project Management And Process Improvement by Robert B Grady, Hewlett Packard Professional Books (2004) 1st ed.

ECS 635: Fault Tolerant Systems
M.Tech. Semester –III(Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisite: Knowledge of distributed system, mathematics is essential.

Course Outcome:

- Student will get knowledge about the understanding of the fundamental concepts of fault-tolerance
- Student will get knowledge about basic techniques for achieving fault-tolerance in electronic, communication and software systems.
- Student will develop skills in modeling and evaluating fault-tolerant architectures in terms of reliability, availability and safety.
- Student will get knowledge about the sources of faults and means for their prevention and forecasting.
- Student will understand merits and limitations of fault-tolerant design.

Objective: To impart of Fault Tolerant Systems, Tools and Techniques.

Unit 1: Fundamental concepts in the theory of reliable computer systems design.

Unit 2: Introduction to redundancy theory, limit theorems; decision theory in redundant systems.

Unit 3: Hardware fault tolerance, redundancy techniques, detection of faults, replication and compression techniques, self-repairing techniques, concentrated and distributed voters, models of fault tolerant computing systems, Case studies.

Unit 4: Software fault tolerance: fault tolerance versus fault intolerance, errors and their management strategies. Implementation techniques: software defense, protective redundancy, architectural support.

Unit 5: Fault recovery techniques & Coding theory: application to fault tolerant system design. Fault-tolerance and reliability of multicomputer networks (direct and indirect) including fault-tolerant routing and sparing techniques. Yield and reliability enhancement techniques for VLSI/WSI array processors.

Books

1. Israel Koren, C. Mani Krishna Fault Tolerant Systems.
   Course developed by professor

ECS 636: Green Grid Computing
M.Tech. Semester –III(Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisite: Knowledge of network protocols and experiences with object-oriented programming is essential.

Course Outcome:

- Students will understand the key concepts of Grid computing.
- Students will understand the Grid computing standards and its toolkits.
- Students will be able to understand about Grid computing history, evolution of Grid and its Security issues
- Students will gain a basic knowledge of Data management and transfer in Grid environments and Resource management.

Objective : To impart the knowledge of Green and Grid computing.

Unit 1: Introduction : The Grid - Past, Present, Future, A New Infrastructure for 21st Century Science, - The Evolution of the Grid - Grids and Grid Technologies, Programming models - A Look at a Grid Enabled Server and Parallelization Techniques – Grid applications

Unit 2: The Anatomy Of The Grid: The concept of virtual organizations – Grid architecture – Grid architecture and relationship to other Distributed Technologies – computational and data Grids, semantic grids. **The Open Grid Services Architecture:** Grid Management systems, security, Grid Grid-Enabling software and Gridenabling network services, Data Grid - Virtualization Services for Data Grids, Peer-to-Peer Grids - Peer-to-Peer Grid Databases for Web Service Discovery.

Unit 3: The Open Grid Services Infrastructure : Technical details of OSGI specification, service data concepts, Naming and Change Management Recommendations – OGSA basic services

Unit 4: Green Grid: The development of standards, measurement methods, processes and new technologies to improve data center performance against the newly defined criteria. The promotion of energy efficient standards, processes, measurements and technology. Green Grid Metrics- Data Center Power Efficiency, Power Usage Effectiveness (PUE) and Data Center Infrastructure Efficiency (DCiE)

Unit 5: Application Case Study: Molecular Modeling for Drug Design and Brain Activity Analysis, Resource management and scheduling, Setting up Grid, deployment of Grid software and tools, and application execution.

Books

1. Fran Bermn, Geoffrey Fox, Anthony HeyJ.G., "Grid Computing: Making the Global Infrastructure a Reality", Wiley, USA, 2003
2. Joshy Joseph, Craig Fallenstein, "Grid Computing", Pearson Education, New Delhi, 2004,
3. Ian Foster, Carl Kesselman, "The Grid2: Blueprint for a New Computing Infrastructure". Morgan Kaufman, New Delhi, 2004
4. Ahmar Abbas, "Grid Computing: Practical Guide to Technology and Applications", Delmar Thomson Learning, USA, 2004

ECS 637: Network Programming
M.Tech. Semester –III(Computer Science & Engg.)

L	T	P		
3	1	-	**Class Work**	**:50 Marks**
			Exam.	**:100 Marks**
			Total	**:150 Marks**
			Duration of Exam	**: 3 Hrs**

Prerequisite: Knowledge of network protocols and experiences with java programming is essential.

Course Outcome:

- Student will get knowledge about security requirements of a networked programming environment and identify the issues to be solved.
- Student will get knowledge about the conceptual solutions to those issues.
- Student will be able to implement a programming solution;
- Student will be able to understand the key protocols that support the Internet;
- Student will be familiar with several common programming interfaces for network communication;
- Student will have a detailed knowledge of the TCP/UDP Sockets
- Student will learn advanced programming techniques such as Broadcasting, Multicasting

Objective: To impart the knowledge socket programming web programming & client server programming.

Unit 1: Socket Programming: Creating sockets, Posix data type, Socket addresses, Assigning address to a socket, Java socket programming, Thread programming, Berkeley Sockets: Overview, socket address structures, byte manipulation & address conversion functions, elementary socket system calls – socket, connect, bind, listen, accept, fork, exec, close, TCP ports (ephemeral, reserved), Berkeley Sockets: I/O asynchronous & multiplexing models, select & poll functions, signal & fcntl functions, socket implementation (client & server programs), UNIX domain protocols
.

Unit 2: APIs & Winsock Programming: Windows socket API, window socket & blocking I/O model, blocking sockets, blocking functions, timeouts for blocking I/O, API overview, Different APIs & their programming technique, DLL & new API's, DLL issues, Java Beans.

Unit 3: Web Programming: Java network programming, packages, RMI, Overview of Javascript, WAP architecture & WAP services, Web databases, Component technology, CORBA concept, CORBA architecture, CGI programming.

Unit 4: Firewall & security techniques Cryptography, Digital Signature.

Unit 5: Client/Server Programming: Client side programming: Creating sockets, Implementing generic network client, Parsing data using string Tokenizer, Retrieving file from an HTTP server, Retrieving web documents by using the URL class. Server side programming: Steps for creating server, Accepting connection from browsers, creating an HTTP server, Adding multithreading to an HTTP server.

Books:

1. Steven.W.R: UNIX Network Programming, PHI (VOL I& II)
2. Window Socket Programming by Bobb Quinn and Dave Schutes
3. Davis.R: Windows Network Programming, Addison Wesley
4. NETWORK PROGRAMMING With Windows Socket By Baner .P., PH New Jersey