FACULTY OF ENGINEERING & TECHNOLOGY

SYLLABUS

FOR

M.TECH. (SOFTWARE SYSTEMS)
(Two Years)

(SEMESTER: I – IV)
(Under Credit Based Continuous Evaluation Grading System)

SESSION: 2014-15

GURU NANAK DEV UNIVERSITY
AMRITSAR

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER SYSTEM
(Under Credit Based Continuous Evaluation Grading System)

Scheme:

Sr.No.	Subject Code	Subject	Credits		
		Semester-I	**L**	**T**	**P**
1.	CSL570	Design & Analysis of Algorithms	4	0	0
2.	CSL571	Operating System Design	4	0	0
3.	CSL572	Software Architecture	4	0	0
4.	CSL573	Parallel Computing Architectures	4	0	0
5.	CSP574	Programming Lab-I	0	0	1
6.	CSP575	Term Paper – I	0	0	1
		Sub Total:	**16**	**0**	**2**
		Grand Total:	**18**		

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – I
(Under Credit Based Continuous Evaluation Grading System)

CSL570: DESIGN & ANALYSIS OF ALGORITHMS

Credits		
L	**T**	**P**
4	0	0

UNIT-I

Algorithm concepts: Algorithm analysis Techniques, design of efficient algorithms, complexity analysis, asymptotic notations, solving Recurrence Relations.
Design Strategies: Divide-and-conquer, Greedy Method, Sorting & Searching: Simple sorting, algorithms, Radix sorting, heap sort, quick sort, linear and binary search algorithms

UNIT-II

Data Structures: Lists, queues, graphs and trees, hash tables, binary search trees, BTrees, heaps.
Advanced Design Strategies. Dynamic Programming, Back-tracking, Local Search Algorithms, Branch-and-bound.
Algorithms on Graphs: Directed Graphs, The Single source shortest path problem, All Pair Shortest path problem, Traversals of Directed Graphs, DAG(Directed Acyclic Graphs), Undirected Graphs, Minimum cost spanning tree. Depth/Breadth First search.

UNIT-III

Advanced Algorithms: Matrix operations, FFT, string processing, simple parallel & approximation algorithms.
Non Deterministic Theory: Non Deterministic Algorithms, NP hard problems, NP completeness, Cook's Theorem

References:

1. Cormen T.H., Leiserson C.E., Rivest R.L., *Introduction to Algorithms*, PHI, 2000
2. Horowitz E., Sahni S., Rajasekaran S., *Computer Algorithms*, Galgotia Publications, 1999.
3. Aho A.V., Hopcroft J.E. Ullman J.D., *The Design and Analysis of Computer Algorithms*, Pearson Education Asia, 1998, 1974
4. Knuth D.E., *The Art of Computer Programming, Volume 1 (Fundamental Algorithms)*, Narosa Publishing House, 1973
5. Knuth D.E., *The Art of Computer Programming, Volume 3 (Sorting and Searching)*, Addison Wesley, 1973.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – I
(Under Credit Based Continuous Evaluation Grading System)

CSL571: OPERATING SYSTEM DESIGN

Credits		
L	**T**	**P**
4	0	0

Unit I

Operating System design techniques-The design process, relationship to software engineering, design problems-design skills & levels, Implementing processes – The system call interface, system initialization, process switching, system call interrupt handling, program error interrupts, disk driver system, implementing waiting, signaling and interrupts, event table managers, process implementation. Parallel systems- Parallel hardware, OS for two processor systems, race conditions with shared processes, atomic actions, multiprocessor OS, threads, Interprocess communication patterns- competing and co-operating, problems, race conditions and atomic actions, new message passing system calls, client server models. Deadlocks- Conditions for deadlock, dealing with deadlocks, two-phase locking, synchronization, semaphores.
Design techniques-some example design techniques.

Unit II

Memory management- levels of memory management, linking and loading process, memory management design, dynamic memory allocation, keeping track allocation of blocks, multiprogramming issues, memory protection, memory management system calls. Virtual memory- Fragmentation and compaction, dealing with fragmentation- paging, swapping, thrashing and load control, dealing with large page tables, sharing memory.
Design techniques-examples of multiplexing and late binding.

Unit III

Introduction to I/O devices- devices and controllers, communication devices, disk devices, disk controllers, SCSI interfaces, tape devices, CD devices.
Introduction to I/O subsystems- I/O system software, unification of files and device, generalized disk device drivers, disk caching.
Introduction to File systems- File abstraction, naming.
Introduction to File system organization- file descriptors, Booting the OS, file system reliability.
Design techniques-Caching, hierarchical names and naming of objects.

References:

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

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – I
(Under Credit Based Continuous Evaluation Grading System)

CSL572: SOFTWARE ARCHITECTURE

Credits		
L	**T**	**P**
4	0	0

UNIT-I

Introduction to Software Architecture:
Software Architecture, Relationships to Other Disciplines ,Multi-Disciplinary Overview, Foundations of Software Architecture , Software architecture in the context of the overall software life cycle, Architectural Styles, CASE study of Architectures.

Software Architecture Design:
Designing, Describing, and Using Software Architecture, IS2000: The Advanced Imaging Solution, Global Analysis, Conceptual Architecture View, Module Architecture View, Styles of the Module Viewtype, Execution Architecture View, Code Architecture View.

Archetype Patterns:
Archetypes and Archetype Patterns, Model Driven Architecture with Archetype Patterns, Literate Modeling, Archetype Pattern. , Customer Relationship Management (CRM) Archetype Pattern.

UNIT-II

Domain – Specific Software Architectures (DSSA):
Components of DSSA, DSSA – Based Software Development, Domain Model, Domain Analysis, DSSA Support Tools.

Architectural Styles:
Pipes and Filters, Data Abstraction and Object-Oriented Organization, Event-based, Implicit Invocation, Layered Systems, Repositories (including Blackboards), Interpreters, Process control, Other Architectures, Heterogeneous Architectures.

Architectural Design Guidance:
Guidance for User-Interface Architectures: Design Spaces and Rules, A Design Space for User-Interface Architectures, Design Rules for User-Interface Architecture, Applying the Design Space: An example, The Qualified Design Space: Overview, Background, Qualified Design Space.

UNIT-III

Formal Models and Specifications:
Formalizing the Architecture of a Specific System, Formalizing an Architectural Style: Filters, Pipes, Pipe-and Filter System, Formalizing an Architectural Design Space

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – I
(Under Credit Based Continuous Evaluation Grading System)

Linguistic Issues:
Requirements for Architecture-Description Languages: The Linguistic Character of Architecture Description, Desiderata for Architecture-Description Languages, Problems with Existing Languages; First-Class Connectors: Problems with Current Practice, An Architectural language with First-Class Connectors; Adding Implicit Invocation to Ada

Tools for Architectural Design:
Unicon: A Universal Connector Language: Components and Connectors, Abstraction and Encapsulation, Types and Type-Checking, Accommodating Analysis Tools; Exploiting Style in Architectural Design Environments: What is Architectural Style, Automated Support for Architectural Design; Architectural Interconnection: Implementation versus Interaction, The WRIGHT Model of Architectural Description

Reference Books:

1. Essential Software Architecture, Ian Gortan, Springer; 2nd Edition. edition (May 5, 2011) ISBN-10: 3642191754, ISBN-13: 978-3642191756
2. Applied Software Architecture, Christine Hofmeister, Robert Nord, Deli Soni, Addison-Wesley Professional, 1 edition (November 14, 1999), ISBN-10: 0321643348, ISBN-13: 978-0321643346
3. Software Architecture: Foundations, Theory, and Practice, Taylor et al., John Wiley, 2010.
4. Software Architecture in Practice, Len Bass, Paul Clements, and Rick Kazman, 2nd ed, Addison-Wesley, 2003.
5. Software Design: From Programming to Architecture, Eric Braude, Wiley, 2004.
6. Software Architecture: Perspectives on an Emerging Discipline, Mary Shaw and David Garlan, Prentice-Hall, 1996.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – I
(Under Credit Based Continuous Evaluation Grading System)

CSL573: PARALLEL COMPUTING ARCHITECTURES

Credits		
L	**T**	**P**
4	0	0

UNIT – I

Introduction to Computers: Basic Computer Organization: System Buses, Instruction Cycles.
CPU Organization: Design specifications for a Simple CPU, Fetching Instructions from Memory, Decoding Instructions, Executing Instructions, Designing the Control Unit Using Hardwired Control and Microprogrammed control approach.
Parallelism in Uniprocessor Systems: Trends in parallel processing, Basic Uniprocessor Architecture, Parallel Processing Mechanism.
Parallel Computer Structures: Pipeline Computers, Array Computers, Multiprocessor Systems

UNIT – II

Architectural Classification Schemes: Multiplicity of Instruction-Data Streams, Serial versus Parallel Processing, Parallelism versus Pipelining
Pipelining : An overlapped Parallelism, Principles of Linear Pipelining, Classification of Pipeline Processors
Principles of Designing Pipelined Processors: Instruction Prefetch and Branch Handling, Data Buffering and Busing Structures, Hazard Detection and Resolution
Superscalar and Superpipeline Design: Superscalar Pipeline Design, Superpipelined Design
Structures and Algorithms for Array Processors: SIMD Array Processors, SIMD Computer Organizations, Inter-PE Communications

UNIT – III

SIMD Interconnection Networks: Static versus Dynamic Networks, Mesh-Connected Illiac Network, Cube Interconnection Networks
System Interconnect Architectures: Network Properties and Routing, Static Connection Networks, Dynamic Connection Networks
Multiprocessor Architecture: Functional Structures: Loosely Coupled Multiprocessors, Tightly Coupled Multiprocessors
Interconnection Networks: Time Shared for Common Buses, Crossbar Switch and Multiport Memories.

References:

1. *Computer Architecture and Parallel Processing*, Faye A. Briggs, McGraw-Hill International Editions.
2. *Computer Systems Organization & Architecture*, John d. Carpinelli, Addison Wesley.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – I
(Under Credit Based Continuous Evaluation Grading System)

CSP574: PROGRAMMING LAB-I

Credits		
L	**T**	**P**
0	0	1

Programming exercises based on the subjects covered in first semester.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – I
(Under Credit Based Continuous Evaluation Grading System)

CSP575: TERM PAPER – I

Credits		
L	**T**	**P**
0	0	1

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER SYSTEM
(Under Credit Based Continuous Evaluation Grading System)

Scheme:

Sr.No.	Subject Code	Subject	Credits		
		Semester-II	**L**	**T**	**P**
1.	CSL580	Distributed Operating System	4	0	0
2.	CSL581	Evolutionary and Intelligent Computing Theories and Applications	4	0	0
3.	CSL582	Advanced Database Concepts	4	0	0
4.		Elective-I	4	0	0
5.	CSP583	Programming Lab-II	0	0	1
6.	Csp588	Term Paper – II	0	0	1
		Sub Total:	**16**	**0**	**2**
		Grand Total:	**18**		
		List of Electives – I			
1.	CSL584	Advanced Software Engineering	4	0	0
2.	CSL585	Network Programming	4	0	0
3.	CSL586	Advanced Network Technologies	4	0	0
4.	CSL587	Network Security	4	0	0

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – II
(Under Credit Based Continuous Evaluation Grading System)

CSL580: DISTRIBUTED OPERATING SYSTEM

Credits		
L	**T**	**P**
4	0	0

Unit I

Distributed Operating Systems: Architectures, Theoretical Foundations, Distributed Mutual Exclusion

Unit II

Resource Management: Distributed File Systems, Distributed Shared Memory, Distributed Scheduling

Unit III

Failure Recovery: Recovery
Deadlock Management: Distributed Deadlock Detection
Communication in DS: BPC, RMI, CORBA, DCOM.

Text Book:
Singhal, Mukesh & N.G. Shivaratri, Advanced Concepts in Operating Systems, Tata McGraw-Hill, 1994.

Reference Books:

1. K. Sinha, “Distributed Operating Systems” Pearson Education, 1998.
2. Distributed Operating Systems – The Logical Design by A. Goscinski, AW
3. Distributed Systems-Concepts and Design by G. Coulouris, AW
4. Distributed System Design by Jie Wu, CRC Press.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – II
(Under Credit Based Continuous Evaluation Grading System)

CSL581: EVOLUTIONARY AND INTELLIGENT COMPUTING THEORIES AND APPLICATIONS

Credits		
L	**T**	**P**
4	0	0

Unit–I

Introduction to Evolutionary Computing & Genetic Algorithms.

Introduction to Genetic Algorithms, Goals of Optimization, How Genetic Algorithms work, A Simple Genetic Algorithm's Computer Implementation highlighting Reproduction by Selection, Crossover, Mutation.

Unit–II

Advanced GA Techniques

Mapping Objective Function to Fitness Form, Fitness scaling, discretization, Different types of Selection and Crossover techniques, methods, Advanced operators in Genetic Search, Dominance, Diploidy and Abeyance. A case study of Travelling Salesman Problem using GA Techniques.

Unit–III

Advanced Computing Techniques

Introduction to Neural Networks, Introduction to Fuzzy Systems and Soft Computing. Introduction to Swarm Intelligence and Optimization. Application of advance computing in Pattern Recognition, Signal Processing & Image Retrieval.

References:

1. Genetic Algorithms in Search Optimization and Machine Learning by David E. Goldberg, Pearson Education.
2. The Simple Genetic Algorithms, Foundations and Theory by Michael D Vose, MIT Press 1999. ISBN-0-262-22058-X
3. How to Solve It: Modern Heuristics, by Zbigniew Michalewicz, David B.Fogel, second Edition Springer Verilag-2004, ISBN- 3-540-22494-7.
4. Advanced Intelligent Computing Theories and Applications, International Conference on intelligent computing, ICIC2007 Qingdao,China, proceedings edited by De-Shuarg Huang,Laurent Heutte and Marco Logo. Springer Book ISBN-978-3-540-74281-4.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – II
(Under Credit Based Continuous Evaluation Grading System)

CSL582: ADVANCED DATABASE CONCEPTS

Credits		
L	**T**	**P**
4	0	0

UNIT-I

Data Base Analysis and Design Techniques: Introduction to basic Database Concepts, Database Design Methodologies: Conceptual, Logical, Physical Designs. ER Modeling: Specialization, Generalization, Aggregation, Normalization Theory.

UNIT-II

Distributed Databases Concepts: Functions and Architecture of a DDBMS, Data Allocation, Data Fragmentation . Query Processing and Optimization Techniques .

UNIT-III

Object Oriented DBMSS Concepts and Design: Abstraction, Encapsulation, object Identity, Methods, Classification and Inheritance, Overloading, Overriding, Polymorphism . Complex Objects, storing objects in Relational Databases. Pointer swizzling techniques, Persistence schemes, versions and schema evolution, Introduction to UML, Object Relational Databases.

References:

1. Distributed Databases by Ozsu and Valduriez ,Pearson Education.

2. *Database Systems*, Thomas Conolly, Carolyn Begg, Pearson Education, Third Edition.

3. *Fundamentals of Database Systems*, Navathe and Ellmassri Pearson Education

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – II
(Under Credit Based Continuous Evaluation Grading System)

CSP583: PROGRAMMING LAB-II

Credits		
L	**T**	**P**
0	0	1

Programming exercises based on the subjects covered in second semester.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – II
(Under Credit Based Continuous Evaluation Grading System)

CSP588: TERM PAPER – II

Credits		
L	**T**	**P**
0	0	1

CSB4: M.TECH. (SOFTWARE SYSTEMS) (SEMESTER – II) ELECTIVES – I
(Under Credit Based Continuous Evaluation Grading System)

CSL584: ADVANCED SOFTWARE ENGINEERING

Credits		
L	**T**	**P**
4	0	0

UNIT-I

Software Project Management: Software Sizing: Function points, Software project models, COCOMO project plan components.

UNIT-II

Software Re-engineering: Introduction Re-engineering, Restructuring and Reverse Engineering, Re-engineering existing systems, Data Re-engineering and migration, Software Reuse and Re-engineering.
Software Measurements: Introduction, Why metrics?, Classification of Software metrics, Study of Design Metrics

UNIT-III

Software Quality Assurance: Quality, Quality Models, Quality Metrices, SQA Activities.

Component Based Software Engineering: Basic Concepts, specification of software components, Component Models

References:

1. Walker Royce, *Software Project Management*, Pearson Education, ISBN: 9780201309584, 2004.
2. Robert S. Arnold, *Software Re-engineering*, IEEE Comp. Society, ISBN: 9780818632723, 2003.
3. Lorenz and Kidd, *Object Oriented Software Metrics*, Prentice Hall, ISBN 978-0131792920, 2001.
4. Booch, *Object-Oriented Analysis and Design with Applications*, Addison-Wesley Professional, 3rd Edition, ISBN: 978-0201895513, 2007.
5. Pressman, "Software Engineering" Prentice Hall, ISBN 978-0131792920, 2001.
6. IvicaCrzKovics Magnus Larsson (eds.) "Building reliable component based software systems" Artech House, 2002 ISBN – 1-58053327-2.

CSB4: M.TECH. (SOFTWARE SYSTEMS) (SEMESTER – II) ELECTIVES – I
(Under Credit Based Continuous Evaluation Grading System)

CSL585: NETWORK PROGRAMMING

Credits		
L	**T**	**P**
4	0	0

UNIT-I

Introduction : TCP/IP Architecture, TCP/IP addressing, services, FTP, SMTP, TFTP, SNMP, Network file system, domain name system, transport layer protocols, user datagram protocol, transmission control protocol.

UNIT-II

Interprocess Communications: File and record locking, pipes, FIFO's, stream and messages, message quues, samphorers.
Sockets: Sockets system cells, reserved parts, stream pipes, socket option, asynchronous I/O, Sockets and signals Transport Lay Interface : Elementary TLI functions, stream and stream pipes, asynchronous I/O I/O multiplexing

UNIT-III

Remote Procedure Calls: Remote login, remote command execution, external data representation.

Reference:

1. A. Stevens, "TCP/IP Illustrated", Vol. 1-3, Pearson Education, 2004.
2. R. Stevens, "Unix Network Programming", PHI 2002.
3. D.E. Comer, "Internetworking with TCP/IP, Vol. 1, Vol. 2, Vol. 3 Principles, Protocols, and architecture, PHI, 2000.

CSB4: M.TECH. (SOFTWARE SYSTEMS) (SEMESTER – II) ELECTIVES – I
(Under Credit Based Continuous Evaluation Grading System)

CSL586: ADVANCED NETWORK TECHNOLOGIES

Credits		
L	**T**	**P**
4	0	0

UNIT-I

Data Link Layer: Framing techniques, Flow control, Error Control, data link protocols, MAC protocols and IEEE standards.

UNIT-II

Network & Transport layer design Issues: Routing algorithms, Congestion control algorithms, Internetworking, Services and elements of Transport protocols.

Network Security , Authentication & E mail Security security attacks and their preventions

UNIT-III

IP Security: IP security overview, IP Security Architecture, Authentication Header Encapsulating Security Pay load.

Wireless LANs: Introduction, Benefits, WLANs Configurations and Standards, Security, IEEE 802.11, Wireless LAN Standard, Blue tooth.

References:

1. *Data Communication & Networking*, B.A. Forouzan.

2., *Computer Networks*, A.S. Tanenbaum, Prentice Hall, 1992, 3rd Edition.

3. *Data & Computer Communication*, William Stallings, McMillan Publishing Co.

4. *Data Networks*, Black, PHI, 1988.

5. *Data Communications*, Fred Halsall, Pearson Education.

CSB4: M.TECH. (SOFTWARE SYSTEMS) (SEMESTER – II) ELECTIVES – I
(Under Credit Based Continuous Evaluation Grading System)

CSL587: NETWORK SECURITY

Credits		
L	**T**	**P**
4	0	0

UNIT-I

Introduction: Attacks, services, mechanisms, security attacks, security services, Model for network security, Internet standards.

Conventional encryption and message confidentiality: Conventional encryption principles, conventional encryption algorithms, cipher block modes of operations, location of encryption devices, key distribution.

UNIT-II

Public Key cryptography and authentication: Approaches to message authentication, Secure Hash Functions and HMAC, Public Key Cryptography, Principles Public Key Cryptography Algorithms, Digital signatures, Key management.

Authentication & E mail Security: Kerberos, X.509 Directory Authentication Services-PGP-S/MIME.

UNIT-III

IP Security: IP security overview, IP Security Architecture, Authentication Header, Encapsulating Security Pay load, Combing Security Associations, Key Management.

Web Security: Web Security Requirements, SSL and Transport Layer Security, SETNetwork Management Security.

System Security: Intruders, viruses related threats, Fire Design principles, Trusted Systems.

References:

1. William Stallings, *Network Security Essentials Applications and Standards*, Pearson Education Asia, New Delhi, 2006
2. Kaufman, *Network Security: Private Communication in a Public World*, Pearson Education Asia, New Delhi, 2005.
3. William Stallings, *Cryptography and Network Security*, Pearson Education Asia, New Delhi, 2005.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER SYSTEM
(Under Credit Based Continuous Evaluation Grading System)

Semester-III					
1.	CSL670	Natural Language Processing	4	0	1
		Elective-II	4	0	1
2.	CSD671	Dissertation (Part-I)	0	0	12
		Sub Total:	**8**	**0**	**14**
		Grand Total:	**22**		
		List of Electives-II			
1.	CSL672	Digital Image processing	4	0	1
2.	CSL673	Artificial Neural Networks	4	0	1
3.	CSL674	Advanced Compiler Design	4	0	1
4.	CSL675	Software Project Management	4	0	1

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – III
(Under Credit Based Continuous Evaluation Grading System)

CSL670: Natural Language Processing

Credits		
L	**T**	**P**
4	0	1*

**Credit for Term Paper*

UNIT-I

Application of Natural Language Phases of linguistic processing:: morphology, syntax, semantics.
Language processors: recognisers, transducers, parsers, generators, Language as a rule-based system, Language understanding as an inferential activity.
Resources for NLP: lexicons and knowledge bases.
Terminology Theory: alphabet, string, language, grammar, rules of grammar, symbol vocabulary,generator, recogniser, procedure.

UNIT-II

Grammars: The Chomsky Hierarchy: Context free grammars & languages, basic parsing strategies for context free grammars simplifications of context free grammars, Chomsky & Greibach normal forms Computational morphology:lemmatisation, Part-of-Speech Tagging, Finite-State Analysis. Parsing:definition of a parser; derivations, determinismand non-determinism; decidability ,data structures and algorithms for parsing ,unification based grammar formalisms.

UNIT-III

Syntactic ambiguities and heuristics, Resolution of syntactic ambiguities lexical ambiguities and selectional restrictions ,indeterminacy of reference Generation and Dialogue:Syntactic generation algorithms and reversibility, text planning, modeling dialogue agents.

Text Book :
Allen, J., *Natural language understanding*. 2nd edition. Redwood City, CA: 1994. Benjamin/Cummings.
ISBN 0805303340.

References:
Grosz, B.J., Sparck Jones, K. & Webber, B.L. (eds) *Readings in Natural Language Processing*. Los Altos, CA, 1986: Morgan Kaufmann.
Jurafsky, D. & J. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Prentice Hall.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – III
(Under Credit Based Continuous Evaluation Grading System)

CSD671: DISSERTATION (PART-I)

Credits		
L	**T**	**P**
0	0	12

The dissertation will normally contain:

1. A clear indication, at appropriate stages, of original and creative elements. The level of originality expected is likely to include the application of existing techniques to new environments, the use of original materials, the re-working of existing materials, and the use of comparative approaches to the provision of information technology;
2. A discussion of its scope and aims, and its theoretical and professional significance, including a discussion of the context in which the problem is seen as important;
3. An analysis of the topic within a critical review of the relevant literature;
4. An evaluation of methods used in the dissertation, their reliability, validity, and a comparison with alternative methods;
5. An account of the process of obtaining the data required for the dissertation and the results obtained;
6. An analysis of the results of the dissertation to include a discussion of their significance, their relationship to other research, and any methodological or theoretical implications; and
7. The relationship of the findings to existing professional understanding and, where appropriate, potential implementation difficulties.

It is not intended to restrict students to a precisely defined format for the dissertation but it should follow the standard practices of dissertation writing. Although a written report will normally be expected, it should be accompanied by soft copy on CD.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – III
ELECTIVES – II
(Under Credit Based Continuous Evaluation Grading System)

CSL672: DIGITAL IMAGE PROCESSING

Credits

L	**T**	**P**
4	0	1*

**Credit for Term Paper*

UNIT-I

Introduction and Digital Image Fundamentals : The origins of Digital Image Processing, Fundamentals Steps in Image Processing, Elements of Digital Image Processing Systems, Image Sampling and Quantization, Some basic relationships like Neighbours, Connectivity, Distance Measures between pixels, Linear and Non Linear Operations.

Image Enhancement: Point Operations, Histograms, Spatial Domain methods, Frequency domain methods, Enhancement by point processing, Spatial filtering, low pass filtering, High pass filtering, Homomorphic filtering, Colour Image Processing.

UNIT-II

Image Restoration Degradation Model, Algebraic approach to Restoration, Inverse Filtering, Wiener Filter, Constrained least square restoration, Interactive restoration, Restoration in spatial domain.

Image Compression: Coding, Interpixel and Psychovisual Redundancy, Image Compression Models, Elements of Information Theory, Error free comparison Lossy Compression, Image Compression Standards.

UNIT-III

Image Segmentation: Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region Oriented Segmentation, Motion based segmentation.

Representation and Description: Representation, Boundary Descriptors, Regional Descriptors, Use of Principal Components for Description, Introduction to Morphology, Some basic Morphological Algorithms.

Object Recoginition: Patterns and Pattern Classes, Decision – Theoretic Methods, Structural Methods.

References:

1. Rafael C. Conzalez & Richard E. Woods, “Digital Image Processing”, 2nd Edition.
2. A.K. Jain, “Fundamental of Digital Image Processing”, PHI

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – III
ELECTIVES – II
(Under Credit Based Continuous Evaluation Grading System)

CSL673: ARTIFICIAL NEURAL NETWORKS

Credits		
L	**T**	**P**
4	0	1*

**Credit for Term Paper*

UNIT-I

Neural Network Technology: Evolution of ANN, Architecture of ANN, Knowledge representation.
Neural Network Learning: Basic learning rules, supervised & unsupervised learning, LMS Algorithm.

UNIT-II

Single Layer Perceptrons-I: Preceptron Model, Preceptron learning algorithms: Simple learning algorithm, pocket algorithm without and with Ratches, Linear Machines, Kessler's construction, Linear Machines Learning algorithm, Representing Boolean functions.
Single Layer Perceptrons-II: Anderson's BSB Model, Hopfield's Model, K-Means Clustering, Topology-Preserving Maps, ART1 and ART2.

UNIT-III

Multilayer Perceptrons: Back-Propagation, Applications of Back-propagation: NETtalk, Handwritten Character Recognition, Pattern Recognition.

References:

1. [SG] Gallant S.L., *Neural Networks Learning & Expert Systems*, MIT Press, ISBN: 9780262071451, 1993.
2. [SH] Haykin S., *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 3rd Edition, ISBN: 9780131471399, 2007.
3. [FS] Freeman J.A., Skapura D.M., *Neural Network Algorithms, Applications and Programming Techniques*, Pearson Education, ISBN: 9780201513769, 2003.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – III
ELECTIVES – II
(Under Credit Based Continuous Evaluation Grading System)

CSL674: ADVANCED COMPILER DESIGN

Credits		
L	**T**	**P**
4	0	1*

**Credit for Term Paper*

UNIT-I

Introduction: structure and architecture of compiler,

Lexical and Syntax analysis: concept of token, Transition diagrams, LEX., CFGs, ambiguity, associativity, types of parsers, top down parsing, recursive descent parsing, , predictive parsing, bottom up parsing, operator

precedence grammars, LR parsers (SLR, LALR, LR), YACC

UNIT-II

Control flow analysis: basic blocks & loops

Data flow analysis and optimizations: bit vectors, iterative frameworks, interval analysis, reaching definitions, liveness, common subexpression elimination, constant propagation

More control flow analysis: dominators, control dependence

Static-single assignment: static-single assignment, constant propagation

Scalar optimization: loop invariant code motion, common subexpression elimination, strength reduction, dead code elimination, loop optimizations, etc.

Instruction scheduling: pipelined architectures, delayed-load architectures, list scheduling

Register allocation: coloring, allocation, live range splitting.

UNIT-III

Interprocedural analysis : side effects, flow-insensitive, flow-sensitive, constants, inlining

Data dependence analysis: dependence testing, dependence graphs

Loop transformations: interchange, tiling, fusion, distribution, splitting

Just-in-time compilation: fast global optimization

Garbage collection: automatic memory management and data locality

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – III
ELECTIVES – II
(Under Credit Based Continuous Evaluation Grading System)

References:

- Steven Muchnick, *Advanced Compiler Design Implementation,* Morgan Kaufmann.
- V. Aho, R. Sethi, and J. D. Ullman. Compilers: Principles, Techniques and Tools , PEARSON
- Education. C. Fischer and R.
- LeBlanc. Crafting a Compiler , PEARSON Education. C. Fischer and R. LeBlanc. Crafting a
- Compiler in C , PEARSON
- Education. A. C. Holub. Compiler Design in C , PEARSON Education. Appel. Modern Compiler
- Implementation in C: Basic
- Keith D Cooper and Linda Torczon, *Engineering a Compiler,* Morgan Kaufmann.
- Randy Allen & Ken Kennedy, *Optimizing Compilers for Modern Architectures,* Morgan aufmann.
- Michael Wolfe, *High Performance Compilers for Parallel Computing, ,* Addison-Wesley.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – III
ELECTIVES – II
(Under Credit Based Continuous Evaluation Grading System)

CSL675: SOFTWARE PROJECT MANAGEMENT

Credits		
L	**T**	**P**
4	0	1*

**Credit for Term Paper*

UNIT-I

Planning Fundamentals: Major issues in software project planning, Planning activities Project, master schedule, Software risk management, Risk monitoring, Risk analysis
Software cost: Major issues in estimating software cost, Cost estimation methods, Experience based model, Parameter based model, COCOMO, Versions of COCOMO, Software size estimation, Function points, Software project schedule, Rayleigh model.

UNIT-II

Functional organization: Project organization, Matrix organization, Staffing, Quality replacements, Turnover management.
Directing a software engineering project: Issues, activities, Conflict management.

UNIT-III

Issues in controlling a software project: Controlling activities, Threads of control, Work breakdown structures, Earned value tracking

References:

1. Richard Thayer, *Software Engineering Project Management*, Tata Mc Graw Hill, 2004.
2. Donald J. Reifer, *Software Management*, Pearson Education, 2003.

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER SYSTEM
(Under Credit Based Continuous Evaluation Grading System)

		Semester-IV			
1.		Inter-disciplinary subject-I	4	0	0
2.		Inter-disciplinary subject-II	4	0	0
3.	CSD680	Dissertation (Part-II)	0	0	14
		Grand Total:	**22**		

CSB4: M.TECH. (SOFTWARE SYSTEMS) SEMESTER – IV
(Under Credit Based Continuous Evaluation Grading System)

CSD680: DISSERTATION (PART-II)

Credits		
L	**T**	**P**
0	0	14

In continuation from 3rd Semester, the evaluation shall be done as per the common ordinances for courses under credit based continuous evaluation grading system.