Nirma University

Institute of Technology

School of Technology

Department of Computer Science and Engineering

M Tech in Computer Science and Engineering

L	T	P	C
2	0	2	3

Course Code	3CS1206
Course Title	Advanced Computer Networks

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to -

1. demonstrate the knowledge of modern networking concepts and data center network planning
2. apply suitable methods to optimize performance of modern networks
3. design and configure networks to support a specified set of applications

Syllabus	**Teaching Hours**
Unit I **Network Concepts and Protocols:** Networking Principles, Network Elements, IPv6 addressing and interoperability with IPv4, Congestion control and TCP, QUIC, SPDY, Split TCP, Websockets	**8**
Unit II **Routing:** Router scheduling algorithms, Router architectures, Border Routing protocols BGP, MPLS	**7**
Unit III **Software Defined Networking**: Data Plane, Control Plane, Application Plane, Controller design, Virtualization, OpenFlow protocol for SDN	**4**
Unit IV **Data Center Networking:** Data center architectures, Data center congestion control, Data center network protocols, MPTCP, DCTCP, Low Latency protocols for Data center	**5**
Unit V **Case Studies and Applications:** Content delivery and video streaming networks, Content Centric Networks, Backbone of Internet, Internet exchange points and BGP, Large scale data centers, Cognitive radio networks	**6**

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 6 experiments to be incorporated.

Suggested Readings^:

1. James Kurose and Keith Ross, Computer Networking: A Top-Down Approach, Pearson
2. William Stallings, Foundations of Modern Networking (SDN, NFV, QoE, IoT and Cloud), Pearson
3. William Stallings, High-speed networks and Internets – Performance and Quality of Service, PHI
4. Hans W. Barz, Gregory A. Bassett, Multimedia Networks: Protocols, Design and Applications, Wiley
5. Rajkumar Buyya, Mukaddim Pathan and Athena Vakali, Content Delivery Networks, Springer
6. Relevant research papers for the topics

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

^this is not an exhaustive list

L	T	P	C
3	0	2	4

Course Code	3CS1112
Course Name	Advanced Database Systems

Course Learning Outcomes (CLO):
At the end of the course, students will be able to –

1. assess various storage and retrieval methods through appropriate indexing
2. design and analyze efficiency of algorithms for database operations
3. comprehend contemporary database architectures and its relevant issues

Syllabus:	**Teaching Hours:**
Unit I	**5**
Data storage: Overview of RDBMS concepts**,** Basic File Structures, File Organization & Record formats, Heap sorted & Hashed Files, Buffer management, Disk Storage, Parallel Disk access with RAID, Modern Storage Architectures	
Unit II	**8**
Indexing Structures: Single level and Multilevel Indexes, B Tree and B+ Tree Indexes, Hash and bitmap based indexing, Index Structures for Single Dimensional and Multidimensional Databases	
Unit III	**9**
Query Processing: Query Execution, Algebra for Queries, Physical-Query-Plan-Operators, Algorithms for Database Operations, Algorithms for Joins and Sorting, hash and index based algorithms, Buffer Management, Parallel Algorithms for Relational Operators	
Unit IV	**8**
Query Optimization: Algebraic Foundation for Improving Query Plans, Estimating Cost of Operations, Cost Based Plan Selection, Choosing Order of Joins, Optimization of Queries for Parallel, Distributed, Multidimensional and Text Database	
Unit V	**7**
Transactions, Concurrency control and Recovery: Transaction scheduling, serializability, Coping with System Failure, Concurrency Control techniques with locking, timestamp ordering and multiversion, Redo and Undo log based recovery, recovery in multi database systems	
Unit VI	**8**
Advances in database systems: Distributed database systems, fragmentation, replication and allocation techniques, NoSQL based systems: key-value based, document based, column based and Graph databases, Streaming SQL,Introduction to active, temporal, spatial, multimedia and deductive databases	

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

Suggested Readings^:
1. RamezElmasri, Shamkant B Navathe, Fundamentals of Database System, Pearson Education
2. Garcia Molina, Ullman, Widom, Data Base System Implementation, Pearson education
3. Raghu Ramakrishnan& Johannes Gehrke, Database Management Systems, McGraw Hill
4. Silberschatz, Korth, Sudarshan, Database System Concepts, McGraw Hill
5. M.TamerOzsu, Patrick Valduriez, S.Sridhar, Principles of Distributed Database Systems, Pearson Education

L=Lecture, T=Tutorial, P=Practical, C=Credit

^this is not an exhaustive list

L	T	P	C
3	0	2	4

Course Code	3CS1111
Course Name	Applied Machine Learning

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to –

1. comprehend statistical methods as basis of machine learning domain
2. apply and evaluate variety of machine learning algorithms
3. implement machine learning techniques to solve problems in interdisciplinary domains

Syllabus:	**Teaching Hours**
Unit I	**3**
Introduction: Motivation and Applications, Basics of Supervised and Unsupervised Learning	
Unit II	**13**
Regression Techniques: Basic Concepts and applications of Regression, Simple Linear Regression – Gradient Descent and Normal Equation Method, Multiple Linear Regression, Non-Linear Regression, Linear Regression with Regularization, Hyper-parameters tuning, Loss Functions, Decision Tree Regression, Evaluation Measures for Regression Techniques	
Unit III	**10**
Classification Techniques: Naïve Bayes Classification: Fitting Multivariate Bernoulli Distribution, Gaussian Distribution and Multinomial Distribution, K-Nearest Neighbours, Classification Trees, Linear Discriminant Analysis, Support Vector Machines: Hard Margin and Soft Margin, Kernels and Kernel Trick, Evaluation Measures for Classification Techniques	
Unit IV	**9**
Artificial Neural Networks: Biological Neurons and Biological Neural Networks, Perceptron Learning, Activation Functions, Multilayer Perceptrons, Back-propagation Neural Networks, Learning with Momentum, Winner-take-all Learning, Competitive Neural Networks, Adaptive ANN	
Unit V	**4**
Clustering: Hierarchical Agglomerative Clustering, k-means Algorithm, Self-Organizing Maps	
Unit VI	**6**
Advances in Machine Learning: Basics of Semi-Supervised and Reinforcement Learning, Introduction to Deep Learning, Best Practices for Machine Learning, Case Studies in interdisciplinary domain	

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

Suggested Readings^:

1. C. Bishop, Pattern Recognition and Machine Learning, Springer
2. R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification and Scene Analysis, Wiley
3. KishanMehrotra, Chilukuri Mohan and Sanjay Ranka, Elements of Artificial Neural Networks, Penram International
4. Tom Mitchell, Machine Learning, TMH
5. RajjanShinghal, Pattern Recognition, Techniques and Applications, OXFORD
6. AthemEalpaydin, Introduction to Machine Learning, PHI
7. Andries P. Engelbrecht, Computational Intelligence - An Introduction, Wiley Publication
8. Andrew Kelleher, Adam Kelleher, Applied Machine Learning for Data Scientist and Software engineers, Addison-Wesley Professional

L=Lecture, T=Tutorial, P=Practical, C=Credit

^this is not an exhaustive list

L	T	P	C
3	0	0	3

Course Code	3CS1113
Course Name	Applied Mathematics for Computer Science

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to –

1. comprehend the mathematical fundamentals related to sets, probability, statistics, linear algebra and mathematical optimization
2. apply the mathematical principles to solve wide range of problems in computer science
3. use the mathematical concepts as per the need of the application

Syllabus:	**Teaching Hours**
Unit I	
Review of Linear Algebra: Matrices, Vectors properties, Eigenvalues and eigenvectors, Matrix factorizations, Distance measures, Projections, Notion of hyperplanes, Half-planes, Application for Linear Algebra in Computer Science	**8**
Unit II	
Probability, Statistics and Random Processes: Probability theory and axioms; Random variables; Probability distributions and density functions (univariate and multivariate), Expectations and moments, Covariance and correlation, Confidence intervals, Correlation functions, Random walks, Markov chains, Statistical inference, Applications in Regression and Classifications.	**16**
Unit III	
Optimization: Basic Concepts, Linear Programming, Duality, Constrained and unconstrained optimization, gradient decent and non-gradient techniques, Introduction to least squares optimization, optimization in Practice.	**12**
Unit IV	**9**
Advanced topics: Nonlinear dimensionality reduction methods, PCA in high dimensions and random matrix theory (Marcenko-Pastur), Linear Discriminant Analysis, Non-Negative Matrix Factorization, Hypothesis testing, Proof Techniques, Random Graphs	

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Suggested Readings^:

1. Gilbert Strang, Introduction to Linear Algebra, Cambridge Press.
2. Gilbert Strang, Linear Algebra and its applications, Harcourt, Brace, Jovanovich Publishers
3. Douglas C. Montgomery, George C. Runger, Applied Probability and Statistics for Engineers, Wiley

4. M. Mitzenmacher and E. Upfal, Probability and Computing: Randomized Algorithms and Probabilistic Analysis, Cambridge University Press
5. Sheldon Ross, A first course in Probability, Pearson
6. Cathy O'Neil and Rachel Schutt, Doing Data Science, O'Reilly Media
7. Avrim Blum, John Hopcroft, and RavindranKannan, Foundations of Data Science, e-book, Cornell University
8. Afonso S. Bandeira, Ten Lectures and Forty-Two Open Problems in the Mathematics of Data Science, e-book, MIT OCW
9. Jeff M. Phillips, Mathematical Foundations for Data Analysis, e-book, University of Utah
10. O. Paneerselvam, Operational Research, PHI

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

^this is not an exhaustive list

L	T	P	C
3	0	2	4

Course Code	3CS1109
Course Title	Complexity Theory and Algorithms

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to -

1. comprehend time & space complexity and formal aspects of algorithms
2. identify appropriate data structures and methodologies for efficient algorithm design
3. design and implement efficient algorithms using various approaches

Syllabus:	**Teaching Hours:**
Unit I	**6**
Mathematical Preliminaries of computational complexity: Asymptotic Notations, Proof of correctness, Performance analysis, Recursive Algorithms and Recurrences	
Unit II	**8**
Complexity Theory: Various complexity classes, linear reductions. Probabilistic algorithms, Approximation algorithms and complexity classes relating to Parallel algorithms	
Unit III	**6**
Data Structures: Hash tables, Binomial heaps , Fibonacci heaps, Disjoint set structures	
Unit IV	**12**
Greedy Algorithms: Making change, graphs and minimum spanning tree, Shortest path, Knapsack problem, Scheduling, etc. **Divide and Conquer**: General Template, Various algorithm implementation like Binary search, Merge Sort, Quick Sort, Convex Hull, Matrix multiplication, etc.	
Unit V	**6**
Dynamic Programming: Introduction of Dynamic Programming, Principle of Optimality, Examples like Single source shortest paths, Knapsack problem, Chained matrix multiplication, Longest Common Subsequence, etc.	
Unit VI	**7**
Graph Algorithms: Elementary algorithms, DFS, BFS, Backtracking, and Branch & Bound techniques with related examples	

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 10 experiments to be incorporated.

Suggested Readings^:

1. Gilles Brassard and Paul Bratley, Fundamentals of Algorithmics, PHI Publication.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest& Clifford Stein, Introduction to Algorithms, PHI Publication.
3. Ellis Horowitz, SartajSahni,SanguthevarRajasekaran, Fundamentals of Computer Algorithms, University Press
4. Jean-Paul Tremblay and Paul G. Sorenson, An Introduction to Data Structures with Applications, Tata McGraw Hill
5. Robert L. Kruse, Data Structures and Program Design in C, PHI

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

^this is not an exhaustive list

L	T	P	C
-	-	-	1

Course Code	3CS1105
Course Title	**Comprehensive Assessment-I**

Course Learning Outcome:

After successful completion of the course, student will be able to

- realize the collective understanding of various courses studied in the semester

Syllabus:

Student will be assessed on the basis of all the courses learned till end of the respective semester.

3CS1104 Computer Architecture [3 0 1 4]

Course Learning Outcome:

After successful completion of the course, student will be able to

- understand and relate the need of parallel computer architecture
- realize the scope for pipelining and vectorization in parallelism
- design the memory hierarchy for parallel architecture
- evaluate the parallel architecture models

Syllabus:

Introduction, Flynn's Taxonomy of Computer Architecture, Introduction to parallel processing, Parallelism in uniproc
architectural classifications, Input and output subsystem, virtual memory system, cache memories and management, I/O sub
Exploitation, Limits on Instruction-Level Parallelism, Multiprocessors and Thread-Level Parallelism, Memory Hierarchy
vector processors and vectorization methods, SIMD array processors, SIMD computer organizations and interconnecti
multiprocessor operating systems, control-flow versus data flow computers

Self Study:

The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. K. Hwang and F. A. Briggs,Computer Architecture and Parallel Processing, McGraw Hill.
2. Hesham El-Rewini, MostafaAbd-El-Barr,Advanced Computer Architecture and Parallel Processing, Wiley
3. H. Stone, Advanced Computer Architecture, Addison Wesley,
4. H. J. Siegel,Interconnection Network for Large Scale Parallel Processing, McGraw Hill
5. J. L. Hennessy and D. A. Patterson,Computer Architecture: A Quantitative Approach, Morgan Kaufmann
6. D.E. Culler, J.P. Singh, and A. Gupta,Parallel Computer Architecture - A Hardware/Software Approach, Morgan K
7. W.J. Dally and B. Towles,Principles and Practices of Interconnection Networks, Morgan Kaufmann Publishers
8. S.W. Keckler, K. Olukotun, and H.P. Hofstee,Multicore Processors and Systems, Springer.
9. Research papers from top conferences such as ISCA, HPCA, MICRO, and ASPLOS.

3CS1103 Data Structure and Algorithms [3 0 1 4]

Course Learning Outcome:

After successful completion of the course, student will be able to

- identify the appropriate data structure and algorithm design method for the given application
- evaluate various techniques for searching, sorting and recurrence
- analyze and design efficient algorithms
- calculate and conclude the associated algorithms' operations and complexity

Syllabus:

Elementary Data Structures: Arrays, stack, queues, linked list, sorting techniques, Hash Tables, Binary Search Trees, B-Trees, Binomial heaps

Mathematical Preliminaries: Algorithm analysis, Algorithm Proof Techniques, Analysis of Algorithms

Growth of Functions: Analyzing Control Structures, Using a barometer, Average case analysis, Amortized Analysis, Solving recurrences

Greedy Algorithms: Making change, graphs and minimum spanning tree, knapsack problem, Scheduling

Divide and Conquer: General Template, various algorithm implementation eg Binary search , Heap sort, Quick Sort, Finding the median, matrix multiplication

Dynamic Programming: Introduction of Dynamic Programming, Principle of Optimality, Comparison with divide and conquer, single source shortest paths, Chained matrix multiplication

Graphs: Elementary Graph Algorithms, DFS, BFS, Backtracking, The knapsack problem, Eight Queens problem, Branch and bound: The assignment problem

Computational Complexity and NP-Completeness: The classes of P and NP, Polynomial reductions, NP-complete problems, NP completeness proofs, NP hard problems, Non-Deterministic algorithms

Self Study:

The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. Gilles Brassard and Paul Bratley, Fundamentals of Algorithmics, PHI
2. Thomas Cormen, Introduction to Algorithms, PHI
3. Trembly and Sorenson, Data Structure and Algorithms, PHI

L	T	P	C
3	0	2	4

Course Code	3CS1110
Course Name	High Performance Computing Architecture

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to –

1. comprehend various High Performance Computing (HPC) system architectures
2. identify design issues related to the architectural characteristics and performance of HPC systems
3. design and implement compute intensive applications on HPC platform

Syllabus:	**Teaching Hours**
Unit I **Parallel Computer Models:** Computing states, Multiprocessors and Multicomputers, Multivector and SIMD Computers, Conditions of parallelism, Program Partitioning and scheduling, Program flow mechanisms, System interconnect architecture	**08**
Unit II **Principles of Scalable Performance and Processor Hierarchy:** Performance Metrics and Measures, Parallel processing applications, Speedup Performance Laws, Scalability Analysis and Approaches, Advanced Processor and Memory Hierarchy Technology, Distributed Shared Memory	**10**
Unit III **Requirement and general issues of High Performance Computing:** Dependable Clustered Computing, Metacomputing: Harnessing Informal Supercomputers, Specifying Resources and Services in Metacomputing Systems,High Speed Networks, Lightweight Messaging Systems, Xpress Transport Protocol, Software RAID and Parallel File systems, Load Balancing Over Networks, Job and Resource Management Systems	**16**
Unit IV **Parallel Models and High Performance Languages:** Scheduling Parallel Jobs on Clusters, Parallel Programming Models, Parallel and High Performance programming languages, Dependence Analysis of Data arrays	**08**
Unit V **Advance Computing**: Introduction to Petascale computing, Optical Computing, Quantum computing and its issues	**03**

Self Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

Suggested Readings^:

1. Kai Hwang,Advance Computer Architecture: Parallelism, Scalability, Programmability, McGraw Hill International Editions
2. Buyya, Rajkumar, High Performance Cluster Computing : Programming and Applications, Pearson Education
3. Georg Hager and Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, CRC Press

L=Lecture, T=Tutorial, P=Practical, C=Credit

^this is not an exhaustive list

3CS1101 High Speed Networks **[3 - 1 4]**

Course Learning Outcome

After successful completion of this course, students will be able to:

- Describe and interpret the basics of high speed networking technologies
- Apply the concept learnt in this course to optimize and troubleshoot high-speed network
- Demonstrate the knowledge of network planning and optimization.
- Design and configure network that have outcome characteristics needed to support a specified set of applications

Syllabus

Introduction to Computer Networks, Networking Principles, Constant Bit Rate, Variable Bit Rate Network Services, Network Elements, Multiplexing, Switching, Error Control, Flow Control

Introduction to High Speed Networks, Analysis of Network traffic using deterministic and stochastic Models, Simulation tools, Tele-traffic engineering, Queuing Models

High Speed TCP Variants, Congestion Control in TCP/IP, ATM

High Speed LAN, Gigabit Ethernet, Distributed Queue Dual Bus (DQDB)

Protocols for QoS Support: IntServ, DiffServ, RSVP, MPLS

Optical Fiber Transmission, TCP/IP Performance over Optical Networks, Fiber Distributed Data Interface, Switched Multi-Megabit Dual Service(SMDS)

Applications demanding high speed communication, Multimedia IP broadcasting, Error resilience in Multimedia Transmission, Satellite Broadcasting

Self-Learning Component

To be decided by course coordinator at the beginning of semester, which will be a blend of one or more of the e-Learning Resources, Video Lectures, Online courses, tools, research material, web links etc. along with the related assessment component(s).

Laboratory Work

Above concepts are to be implemented and at-least 5 experiments are to be carried out.

References:

1. High-speed networks and Internets – Performance and quality of service by William Stallings
2. High Performance TCP/IP Networking: Concepts, issues and solutions: By Mahoob Hassan Raj and Jain
3. High-speed networks: TCP/IP and ATM design principles by William Stallings
4. High speed networks by Marc Boisseau, Michel Demange, Jean-Marie Munier
5. Multimedia Communications: Applications, Networks, Protocols and Standards, Fred Halsall, Addison –Wesley

3CS1106 Software Engineering [3 – 1 4]

Course Learning Outcome:

After successful completion of the course, student will be able to

- acquaint with systematic and organized approach for developing the software
- learn an individual as well as teamwork approach for project development
- develop necessary skills for designing a software system using modern tools
- understand various software quality assurance models

Syllabus:

Software process and lifecycle: Software Product, Product, Software Processes, , Study of different system models, Critical Systems, Object Oriented Software Engineering, Formal Methods, Project Management Concepts, Planning and Scheduling, Team organization and people management, RiskManagement, metric and management, software configuration, Software Cost estimation Software Requirements, Requirements Engineering Processes, Feasibility Studies, Validation and Management

Design principles: Architectural design. Distributed Systems Architecture, Object Oriented Design, Real TimeSoftware Design, User Interface Design.

Development and Testing: Rapid Software Development, Software Reuse with COTS Technology, Component Base Software Engineering, Critical systems development, Clean Room software Engineering ,System Testing

Verification and validation of Software: Software Inspections and Audit, Automates Analysis, CASE Tools andCritical systems validation

Software Quality Assurance, Quality Standards, Quality Planning and Control, Software Reliability Models,

Emerging Technologies: Security Engineering, Agile Methods, Service Oriented Software Engineering, AspectOriented Software Development, Software Engineering Aspects of Programming Languages.

Self Study:

The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated with individual as well as team work based assignments.

References:

1. Ian Sommerville,SoftwareEngineering,Addison – Wesley
2. Roger Pressman,Software Engineering A Practitioners Approach,McGraw Hill
3. RajibMall,Fundamentals of Software Engineering,PHI
4. Ivar Jacobson,Object Oriented Software Engineering A use case Approach, Pearson

3CS1E28 Artificial Intelligence [3 0 1 4]

Course Learning Outcome:
After successful completion of this course, student will be able to

- understand concept of knowledge representation and predicate logic and transform the real life information in different representation
- understand state space and its searching strategies
- analyze a system and its implementation

Syllabus:
Introduction to Artificial Intelligence Overview: What is AI, Importance, and early work in AI, AI related fields.
Knowledge: General concepts, definition and importance of knowledge, knowledge based system, representation, organization, manipulation and acquisition of knowledge.
Problems, Problem Spaces and State Space Search: The AI Problems, The Underlying Assumption, What Is An AI Techniques, The Level Of The Model, Criteria For Success, Some General References, One Final Word. Defining The Problems As a State Space Search, Production Systems, Production Characteristics, Production System Characteristics, and Issues In The Design Of Search Programs.
Knowledge Representation: Knowledge Representation Issues, Representations And Mappings, Approaches to knowledge Representation, Using Predicate Logic Representation Simple Facts in Logic, Representing Instance and ISA Relationships, Computable Functions And Predicates, Resolution. Representing Knowledge Using Rules, Procedural Versus Declarative Knowledge, Logic Programming, Forward Versus Backward Reasoning.
Weak Slot-And-Filler Structure : Semantic Nets, Frames.
Search and Control Strategies : Uninformed(Blind) and informed search, DFS, BFS, Heuristic Search Techniques : Generate-And-Test, Hill Climbing, Best-First Search, A*, AO*, Problem Reduction, Constraint Satisfaction, Means-Ends Analysis.
Reasoning : Symbolic Reasoning Under Uncertainty, Introduction to Non-monotonic Reasoning, Logics for Non-monotonic Reasoning. Statistical Reasoning , Probability And Bay's Theorem, Certainty Factors And Rule-Base Systems, Bayesian Networks, Dempster-Shafer Theory.
Game Playing: Overview and Example Domain, Min-max Search, Adding Alpha-Beta Cutoffs.
Expert System: Introduction, Architecture, and Types of Expert Systems.

Self Study:
The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. Elaine Rich And Kevin Knight, Artificial Intelligence and Expert Systems, Tata McGraw-Hill.
2. D.W.Patterson, Artificial Intelligence and Expert System, Development, W.Rolston, Mcgraw-Hill International Edition.
3. Ivan Bratko, Introduction to Prolog Programming Carl Townsend PROLOG Programming for Artificial Intelligence, Addison-Wesley
4. ClocksinAndMellish, "Programming with PROLOG", Stuart Russell.
5. Peter Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall.

L	T	P	C
3	0	2	4

Course Code	3CS12D301
Course Name	Big Data Systems

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to

1. analyse the big data analytic techniques for business applications.
2. manage big data using different tools and frameworks.
3. design efficient algorithms for mining the data from large volumes.
4. implement the HADOOP and MapReduce technologies associated with big data analytics

Syllabus	**Teaching Hours**
Unit I **Introduction to Big Data:** Introduction to Big Data Platform, Challenges of Conventional Systems, Intelligent Data Analysis, Nature of Data, Analytic Processes and Tools, Analysis vs Reporting, Modern Data Analytic Tools, Statistical Concepts: Sampling Distributions, Re-Sampling, Statistical Inference - Prediction Error	**5**
Unit II **The Big data technology landscape :** NoSQL, Types of No SQL databases, SQL Vs No SQL, why No SQL, Introduction to MongoDB, Data Types in MongoDB, CRUD, Practice examples, Apache Cassandra, Features of Cassandra, CRUD operations	**10**
Unit III **Hadoop:** History of Hadoop, The Hadoop Distributed File System, Components of Hadoop, Analysing the Data with Hadoop, Scaling Out, Hadoop Streaming, Design of HDFS, Java Interfaces to DFS Basics, Developing a Map Reduce Application, How Map Reduce Works, Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features, Hadoop ecosystem.	**10**
Unit IV **Hadoop Environment:** Setting up a Hadoop Cluster, Cluster Specification, Cluster Setup and Installation, Hadoop Configuration, Security in Hadoop, Administering Hadoop, HDFS, Monitoring, Maintenance, Hadoop benchmarks, Hadoop in the Cloud.	**10**
Unit IV	**10**

Frameworks: Applications on Big Data Using Pig and Hive, Data Processing Operators in Pig, Hive Services, HiveQL, Querying Data in Hive, Fundamentals of HBase and ZooKeeper, IBM Info Sphere Big Insights and Streams, Visualizations, Visual Data Analysis Techniques, Interaction Techniques, Systems and Applications

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 6 experiments to be incorporated.

Suggested Readings^:

1. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer
2. Tom White, Hadoop: The Definitive Guide, Third Edition, O'reilly Media
3. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw Hill Publishing
4. Anand Rajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press
5. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & sons
6. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons
7. Pete Warden, Big Data Glossary, O'Reilly
8. Jiawei Han, Micheline Kamber, Data Mining Concepts and Techniques, Second Edition, Elsevier
9. Da Ruan, Guoquing Chen, Etienne E. Kerre, Geert Wets, Intelligent Data Mining, Springer
10. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan, Harness the Power of Big Data: The IBM Big Data Platform, Tata McGraw Hill Publications
11. Michael Minelli, Michele Chambers, Ambiga Dhiraj, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley Publications
12. Zikopoulos, Paul, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, Tata McGraw Hill Publications
13. Seema Acharya and Subhashini C, Big Data and Analytics, Wiley India

L=Lecture, T=Tutorial, P=Practical, C=Credit

^this is not an exhaustive list

L	T	P	C
2	0	2	3

Course Code	3CS12D201
Course Name	Blockchain Technology

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to

1. comprehend the structure of a Blockchain networks
2. evaluate security issues relating to Blockchain and cryptocurrency
3. design and analyze the applications based on Blockchain technology

Syllabus:	**Teaching Hours**
Unit I **Introduction to Blockchain:** History, Digital Money to Distributed Ledgers, Design Primitives, Protocols, Security, Consensus, Permissions, Privacy	**3**
Unit II **Blockchain Architecture, Design and Consensus**: Basic crypto primitives: Hash, Signature, Hashchain to Blockchain, Basic consensus mechanisms, Requirements for the consensus protocols, PoW and PoS, Scalability aspects of Blockchain consensus protocols	**8**
Unit III **Permissioned and Public Blockchains**: Design goals, Consensus protocols for Permissioned Blockchains, Hyperledger Fabric, Decomposing the consensus process, Hyperledger fabric components, Smart Contracts, Chain code design, Hybrid models (PoS and PoW)	**9**
Unit IV **Blockchain cryptography**: Different techniques for Blockchain cryptography, privacy and security of Blockchain, multi-sig concept	**6**
Unit V **Recent trends and research issues in Blockchain**: Scalability, secure cryptographic protocols on Blockchain, multiparty communication, FinTech and Blockchain applicabilities	**4**

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

Suggested Readings^:

1. Narayanan, Arvind,. et al, Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press.
2. Wattenhofer, Roger, The science of the blockchain, CreateSpace Independent Publishing Platform
3. Bahga, Arshdeep, and Vijay Madisetti,. Blockchain Applications: A Hands-on Approach, VPT
4. Nakamoto, Satoshi, Bitcoin: A peer-to-peer electronic cash system, Research Paper
5. Antonopoulos, Andreas M, Mastering Bitcoin: Programming the open blockchain, O'Reilly Media, Inc
6. Diedrich, Henning, Ethereum: Blockchains, digital assets, smart contracts, decentralized autonomous organizations, Wildfire Publishing (Sydney)

L=Lecture, T=Tutorial, P=Practical, C=Credit

^this is not an exhaustive list

3CS1E13 Cloud Computing [3 0 1 4]

Course Learning Outcome:
After successful completion of this course, student will be able to

- describe the hardware and software concepts and architecture of cloud computing
- contrast the key technical and commercial issues concerning cloud computing versus traditional software models
- recognize the importance of virtualization technology in support of cloud computing
- explore the issues related to cloud computing data centres

Syllabus:
Foundations: Distributed system models and enabling technologies, computer clusters for scalable Computing, Introduction to Cloud Computing, On the Management of Virtual Machines for Cloud Infrastructures, Migrating to the cloud, virtual machines and virtualization of clusters and datacenters, Applications of Virtual Machines, Nested Virtualization
Cloud services: Infrastructure as a service, Virtual Machines Provisioning and Migration Services, Platform as a service , Enhancing Cloud Computing Environments
Using a Cluster as a Service, Software as a Service, SLA Management in Cloud Computing, Performance Prediction for HPC on Clouds, Workflow Engine for Clouds, Understanding Scientific Applications for Cloud Environments, The MapReduce Programming Model and Implementations,
Monitoring and Management of Cloud: A Service Provider's Perspective, Best PracticesinArchitecting Cloud Applications, Building Content Delivery Networks Using Clouds,Resource Cloud Mashups,Security in cloud, Governance and Case Studies, Legal Issues in Cloud Computing, Achieving Production Readiness for Cloud Services Exploring prototypes and present day clouds.

Self Study:
The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. RajkumarBuyya, James Broberg,Andrzej M Goscinski, Cloud Computing: Principles and Paradigms, Wiley Publication.
2. Kai Hwang, Jack Dongarra& Geoffrey C. Fox, Distributed and Cloud Computing: Clusters, Grids, Clouds and the Future Internet, Elsevier
3. Gautam Shroff, Enterprise Cloud Computing: Technology, Architecture, and Applications, Cambridge University Pres
4. Toby Velte, Anthony Velte, Cloud Computing, A Practical Approach, McGraw-Hill Osborne Media, McGraw-Hill
5. Selected Research Papers from Various Sources.

3CS1203 Compiler Design [3 0 1 4]

Course Learning Outcome:

After successful completion of the course, student will be able to

- understand the structure and components of compiler
- apply optimization techniques for the code generated by the compiler
- construct a compiler for simple programming language

Syllabus:

Overview of Lexical analysis and Syntax analysis, Intermediate code generation, Syntax directed translation, symbol table management and error handlers, Introduction to various Intermediate representation and Run time support, Control flow analysis, Data flow analysis, Dependence analysis and dependence graphs, Introduction to optimizations, Redundancy elimination, Loop optimizations, procedure optimizations, Register allocation, introduction to code scheduling, control flow and low level optimizations

Self Study:

The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. Aho, Ulman, Sethi, Principles of Compiler Design, Pearson
2. Steven S. Muchnick, Advanced Compiler Design Implementation, Morgan Kauffman Publishers
3. Wolfe, High Performance Compilers for Parallel Computing
4. Zima and Chapman, Super-compilers for Parallel and Vector Computers
5. Utpal Banerjee, Dependence analysis for supercomputing
6. Wolfe, Optimizing Supercompilers for Supercomputers
7. Ellis.Bulldog, A Compiler for VLIW Architectures

3CS1205 Comprehensive Assessment – II [0 0 0 1]

Course Learning Outcome:

After successful completion of the course, student will be able to

- realize the collective understanding of various courses studied in the semester

Syllabus:

Student will be assessed on the basis of all the courses learned till end of the respective semester.

L	T	P	C
3	0	2	4

Course Code	**3CS12D103**
Course Name	**Data Privacy**

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to –

1. comprehend the concepts of web security and privacy, hardware and software vulnerabilities and protection mechanisms
2. realize the need for data privacy and the related technologies
3. derive and demonstrate the protection mechanisms against several data related attacks

Syllabus:

Teaching Hours

Unit I **10**

Introduction to Security: Cryptography, Web security, Hardware and software vulnerabilities

Unit II **10**

Data Privacy: Data localization issues, Managing personally identifiable or sensitive information, Hippocratic databases, Differential privacy, Privacy preserving data analysis

Unit III **5**

Basic concepts and definitions, objectives, disclosure control and inference of entities, models of protection like null map, k-map, wrong-map

Unit IV **10**

Data Explosion: Availability vs. Storage vs. Collection trade-off, barriers to distribution, mathematical models for sharing practices and policies for computing privacy and risk measurements

Unit V **10**

Demographics and Uniqueness, data linking, data profiling, data privacy attacks

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 7 experiments to be incorporated.

Suggested Readings^:

1. Stallings, W. Cryptography and Network Security. Pearson Education India.
2. Giannotti, F., &Pedreschi, D. (Eds.). Mobility, data mining and privacy: Geographic knowledge discovery. Springer Science & Business Media.
3. Bygrave, L. A. Data privacy law: an international perspective (Vol. 63). Oxford: Oxford University Press.
4. Scoble, R., Israel, S., &Benioff, M. R.. Age of context: Mobile, sensors, data and the future of privacy. USA: Patrick Brewster Press.
5. Bendat, J. S., &Piersol, A. G. Random data analysis and measurement procedures.

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

^this is not an exhaustive list

3CS1E23 Data Warehousing and Mining [3 0 1 4]

Course Learning Outcome:
After successful completion of this course, student will be able to

- identify the key processes of data mining, data warehousing and knowledge discovery process
- understand the basic principles and algorithms used in practical data mining and their strengths and weaknesses
- apply data mining techniques to solve problems in other disciplines in a mathematical way

Syllabus:
Introduction, Multidimensional Data Model, Data Warehouse & OLAP Technology for Data Mining, Architecture, Differences of Data warehouse and Data mart, Data Preprocessing, Preprocessing Techniques, Data Mining Primitive, Language & System Architecture.
Concept Description: Characterization and Comparison, Attribute Oriented Induction.
Mining Association rules in large database, Techniques for frequent pattern mining.
Classification & Prediction: Classifier, ID3, C4.5, Regression. Cluster analysis, clustering techniques.
Applications of Mining
Intelligent Functionalities of Warehouse: Querying and Reporting, Online Analytical Processing, Data mining and Executive information systems (EIS).
Case study of some available OLAP tools.

Self Study:
The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:
1. Jiahei Han &MichelineKamber, Data Mining Concepts and Techniques, Morgan Kaufmann
2. S. Nagabhushana, Data Warehousing OLAP and Data Mining, New Age publishers

L	T	P	C
3	0	2	4

Course Code	3CS12D302
Course Name	Deep Learning and Applications

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to

1. comprehend the strengths and weaknesses of deep networks
2. analyze suitability of different deep networks for variety of problems
3. design and implement deep networks for solving problems pertaining to computer science and interdisciplinary research

Syllabus:	**Teaching Hours**
Unit I **Applications of Convolutional Neural Networks:** Understanding CNN, Image Classification with Localization, Object Detection, Semantic Segmentation, Instance Segmentation	**10**
Unit II **Applications of Recurrent and Recursive Neural Networks:** Understanding Recurrent and Recursive Neural Networks, Word Embedding, Language Models, Named-Entity Recognition, Machine Translation, Parsing, Sentiment Analysis, Speech Recognition	**13**
Unit III **Specific Applications:** Image Captioning, Video Captioning, Document Classification, Prediction in Stock Markets, Recommender Systems, Dimensionality Reduction, Image Denoising, Anomaly Detection from Video, Face and Facial Expression Recognition	**13**
Unit IV **Generative Applications:** Understanding Generative Adversarial Networks, Image Inpainting, Image Super Resolution, Colorization of Black and White Images, Human Face Generation, Text2Image, Music Generation	**9**

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

Suggested Readings^:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press
2. Adam Gibson, Josh Patterson, Deep Learning, O'Reilly Media, Inc.
3. Francois Chollet, Deep Learning with Python, Manning Publications

L=Lecture, T=Tutorial, P=Practical, C=Credit

^this is not an exhaustive list

3CS1204 Distributed and Parallel Systems [3 0 1 4]

Course Learning Outcome:

After successful completion of the course, student will be able to

- understand and describe the basics principles of parallel and distributed computing system
- design and develop parallel algorithms
- explore recent advances in distributed computing systems

Syllabus:

Introduction to Parallel and Distributed Programming

Parallel Computing Architectures, Advanced Processors and Interconnects - Multicore Processors and High-bandwidth Networks, Paradigms & Issues

Scalable Multiprocessors and Multicomputers - Distributed CC-NUMA and cluster Scalability.

Physical and Virtual Clusters - Server clusters, high availability, and Disaster Recovery

Processes, inter-process communication, multithreaded programming, thread synchronization, and programming parallel virtual machines (PVM and MPI), Concurrent programming primitives (semaphores, locks, monitors)

Shared Memory Models and Distributed Memory Models

Distributed Programming Issues and Algorithms

Remote procedure calls, process management, migration, mobile agents, distributed coordination, fault tolerance

Distributed File Systems, synchronization, fault tolerance, coordination and consensus, replication and sharing, security

Peer-to-Peer Computing Systems: P2P systems, Overlay networks, and Content Distribution.

Distributed Computing Tools & Technologies: P2P systems, TeraGrid, MapReduce, Clusters, Hadoop, Twister, Dryad, BigTable, GFS

Self Study:

The self study contents will be declared at the commencement of semester. Around 10% of the questions willbe asked from self study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. H. Attiya, J. Welch, Distributed Computing, Fundamentals, Simulations, and Advanced Topics, Wiley
2. Barry Wilkinson and Michael Allen, Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers, PHI
3. Calvin Lin, Larry Snyder,Principles of Parallel Programming, PHI
4. K. Hwang and Z. Xu, Scalable Parallel Computing, McGraw-Hill
5. Ian Taylor, From P2P to Web Services and Grids, Springer-Verlag
6. F. Berman, G. Fox, and T. Hey (Editors), Grid Computing, Wiley

L	T	P	C
2	0	2	3

Course Code	3CS12D202
Course Title	Human Computer Interaction

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to

1. evaluate user interfaces and detect usability problems by doing usability studies with human subjects
2. simulate how a user would understand and attempt to use an interface using an analytical method such as cognitive walkthrough
3. apply an appropriate interaction style for a given need
4. implement the HCI techniques to build multimodal GUI

Syllabus

Teaching hours

Unit 1 **5**

Introduction: Introduction to the field of HCI, HCI's and human factors engineering and user experience design. Human perception: perception, gestalt perception, information presentation: typography, color, graphic design, displays, paper, and other output devices, layout: forms design and information visualization, virtual reality, context-sensitive interfaces

Unit 2 **6**

User Interface Design and Principles: principles of HCI, ubiquity of feedback cycles, the importance of direct manipulation, and the extent of human abilities as they relate to computer interfaces, understanding of user tasks and activities, HCI heuristics. Creating good user interfaces: need-finding, prototyping potential interfaces, and evaluating those interfaces with users, research ethics underlying the design life cycle, as well as applications of this life cycle to the modern era of rapid prototyping

Unit 3 **12**

Interactive Devices, User Interfaces and Interaction styles: input devices and ergonomics: multi-touch, augmented reality, haptics, wearables, brain computer interfaces, and tangibles. Multimodal user interfaces: basic technologies for handling speech, vision, pen-based interaction, and other modalities, as well as various techniques for combining modalities. Interaction styles: metaphor, direct manipulation, widget survey, other interaction styles, and choosing among interaction styles

Unit 4 **7**

Applications: Exploration into the applications of HCI to open areas like augmented reality, education, social computing, mental health, healthcare, medical science and assistive applications for differently abled

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Students need to work on course project and implement the above mentioned learned concepts.

Suggested Readings^:

1. Ben Shneiderman, Catherine Plaisant, et al. Designing the User Interface, Addison Wesley
2. Preece, Sharp & Rogers, Interaction Design: Beyond Human-Computer Interaction, John Wiley & Sons
3. Dix A., Finlay J., Abowd G. D. and Beale R., Human Computer Interaction, Pearson Education.
4. Cooper, Reimann, Cronin, & Noessel., About Face: The Essentials of Interaction Design, Wiley
5. Preece J., Rogers Y., Sharp H., Baniyon D., Holland S. and Carey T., Human Computer Interaction, Addison-Wesley.
6. B.Shneiderman, Designing the User Interface, Addison Wesley (Indian Reprint)
7. Research Papers related to HCI applications and core research

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

^this is not an exhaustive list

L	T	P	C
3	0	2	4

Course Code	3CS12D104
Course Name	Internet of Things

Course Learning Outcomes (CLOs):

At the end of the course, students will be able to

1. comprehend the architectural components and platforms of IoT ecosystem
2. apply appropriate access technology and protocol as per the application requirement
3. identify data analytics and data visualization tools as per the problem characteristics

Syllabus:	**Teaching Hours**
Unit I Introduction, applications, need and scope of IoT, Various IoT architectures, functional stack, Processors and Operating Systems for resource constrained devices	**5**
Unit II Sensors and actuators, smart objects, Connecting objects, protocols and access technologies like IEEE802.15.4, LFNBPLC, LoRaWAN, WirelessHART, LTE-M, BLE, NB-IoT, Sigfox, White-Fi and HaLow	**12**
Unit III IoT network layer, IPv6: IPv6 structure, addressing, routing, interconnecting issues, 6LoWPAN: forwarding, addressing, header compression, neighbour discovery, Routing in LLN, RPL	**7**
Unit IV Application layer protocols, CoAP, MQTT, AMQP, XMPP, Integrating Internet Services with Interoperable data encoding with XML, JSON and CBOR, Sensor data models and representation, The Sensor Mark-up Language (SENML), lightweight web services for IoT	**9**

Unit V **5**

Data analytics for IoT, machine learning, big data analytics tools and technology like NoSQL, Hadoop

Unit VI **4**

Securing IoT, Challenges in IoT security, provisions for securing IoT network

Unit VII **3**

Case studies on IoT applications: Connected Vehicles, Autonomous Vehicles, Industrial Applications of IoT

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 6 experiments to be incorporated.

Suggested Readings^:

1. David Hanes, G. Salgueiro, IoT Fundamentals - Networking Technologies, Protocols, and Use Cases for Internet of Things, Cisco Press
2. Jean-Philippe Vasseur, Adam Dunkels, Interconnecting Smart Objects with IP: The Next Internet, Morgan Kaufmann
3. Pethuru Raj, Anupama Raman, The Internet of Things - Enabling Technologies, Platforms and Use Cases, CRC Press
4. Robert Stackowiak, Art Licht, VenuMantha and Louis Nagode, Big Data and The Internet of Things, Apress
5. Peter Waher, Learning Internet of Things, Packt Publishing Ltd
6. Daniel Kellmereit, Daniel Obodovski, The Silent Intelligence: The Internet of Things, DND Ventures
7. Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things: Key Applications and Protocols, Wiley Publications

L=Lecture, T=Tutorial, P=Practical, C=Credit

^this is not an exhaustive list

3CS1E35 Machine Learning [3 0 1 4]

Course Learning Outcomes

After successful completion of this course, student will be able to

- have exposure to machine learning concepts and range of problems that can be handled by machine learning
- compare and parameterize different learning algorithms
- apply the machine learning concepts in real life problems

Syllabus

Introduction: Motivation and Applications

Regression Techniques: Regression, Simple Linear Regression – Gradient Descent and Normal Equations Method, Multiple Linear Regression, Non-Linear Regression, Linear Regression with Regularization, Support Vector Regression, Evaluation Measures for Regression Analysis

Classification Techniques: Naïve Bayes Classification: Fitting Multivariate Bernoulli Distribution, Gaussian Distribution and Multinomial Distribution, K-Nearest Neighbours, Classification Trees, Linear Discriminant Analysis, Support Vector Machines: Hard Margin and Soft Margin, Kernels and Kernel Trick, Evaluation Measures for Classification

Artificial Neural Networks: Biological Neurons and Biological Neural Networks, Perceptron Learning, Activation Functions, Multilayer Perceptrons, Back-propagation Neural Networks, Learning with Momentum, Winner-take-all Learning, Competitive Neural Networks

Clustering: Hierarchical Agglomerative Clustering, k-means Algorithm, Self-Organizing Maps

The PAC and mistake bound learning frame work, VC dimension

Self-Learning Component

To be decided by course coordinator at the beginning of semester, which will be a blend of one or more of the e-Learning Resources, Video Lectures, Online courses, tools, research material, web links etc. along with the related assessment component(s).

Laboratory Work

Above concepts are to be implemented and at least 5 experiments are to be carried out.

References

1. C. Bishop, Pattern Recognition and Machine Learning, Springer
2. R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification and Scene Analysis, Wiley
3. Kishan Mehrotra, Chilukuri Mohan and Sanjay Ranka, Elements of Artificial Neural Networks, Penram International
4. Tom Mitchell, Machine Learning, TMH
5. Rajjan Shinghal, Pattern Recognition, Techniques and Applications, OXFORD
6. Athem Ealpaydin, Introduction to Machine Learning, PHI
7. Andries P. Engelbrecht, Computational Intelligence - An Introduction, Wiley Publication

L	T	P	C
-	-	10	5

Course Code	**3CS1207**
Course Title	**Minor Project**

Course Outcomes (COs):
At the end of the course, students will be able to -

1. identify the issues related with the recent trends in the field of computer science and its applications
2. formulate the problem definition, analyze and do functional simulation of the same
3. design, implement, test and verify the proposed solution related to problem definition
4. compile, comprehend and present the work carried out

A student is required to carry out project work in the relevant area of post-graduate study. The project may include design / simulation / synthesis / development of a system, etc. At the end of the semester, a student has to submit a detailed report incorporating literature survey, problem formulation, clear problem statement, research methods, result analysis, conclusion, etc. It is expected that the student should defend his/her work before the jury / panel of examiners.

L = Lecture, T = Tutorial, P = Practical, C = Credit

3CS1E07 **Modern Databases** **[3 0 1 4]**

Course Learning Outcome:
After successful completion of this course, student will be able to

- understand query processing and optimization for various centralized / distributed databases
- understand new-generation data models used for highly distributed databases
- design and deploy various column oriented, document oriented, key-value stores and graph databases
- apply tools and techniques for applications related to social networks and similar systems

Syllabus:
Principles of query processing, Indexing techniques, Query execution plans and operators, Query optimization Next-generation data models, Aggregate Data Models ,declarative query language, Distribution Models

Databases Vs. File Systems: GFS, HDFS, Big Data, Bigtable/HBASE Row-oriented Vs. column-oriented storage Advanced indexing methods, Scalable data processing , Parallel query plans and operators, massively parallel joins

Systems based on MapReduce: Hadoop, Hive

Scalable Key-Value Stores: Amazon Dynamo, Cassandra No-SQL Databases, Document Store database

Distributed Data: Parallel Computing with Monad algebra, NESL, DryadLINQ, PigLatin Graph Databases, Grid & Cloud Database Solutions Large-Scale Graph Processing: Pregel

Self Study:
The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. Martin Fowler, Pramod J. Sadalage,NoSQL Distilled, Pearson.
2. Elmasri and Navathe,Fundamentals of Database Systems, 6th Ed Addison Wesley.
3. Garcia-Molina, Ullman and Widom, Database Systems: The Complete Book, Pearson.
4. Connolly and Begg, Database Systems, Addison Wesley.
5. J. Hellerstein& M. Stonebraker,THE RED BOOK: Readings in Database Systems, MIT Press.
6. Selected research papers relevant to the course topics

3CS1202 Real Time Systems [3 0 1 4]

Course Learning Outcome:
After successful completion of the course, student will be able to

- understand software and system design process for microcontroller based real time systems
- comprehend intricacies of real time databases
- apply programming skills to develop real time systems
- assess and design fault tolerant systems

Syllabus:
Introduction: Issues in Real Time Computing, Structure of a Real Time System. Task Classes, Performance Measures for Real Time Systems, and Estimating Program Run times. Task Assignment and Scheduling: Classical Uniprocessor scheduling algorithms, UniProcessor scheduling of IRIS Tasks, Task Assignment, Mode Changes, and Fault Tolerant Scheduling
Programming Language and Tools: Desired Language characteristics, Data Typing, Control structures, Facilitating Hierarchical Decomposition, Packages, Run time (Exception) Error handling, Overloading and Generics, Multitasking, Low Level programming, Task scheduling, Timing Specifications, Programming Environments, Run time Support
Real Time Databases: Basic Definition, Real time Vs General Purpose Databases, Main Memory Databases, Transaction priorities, Transaction Aborts, Concurrency Control Issues, Disk Scheduling Algorithms, Two-phase Approach to improve Predictability, Maintaining Serialization Consistency, Databases for Hard Real Time systems
Real Time Communication: Communications Media, Network Topologies Protocols, Fault Tolerant Routing.
Fault Tolerance Techniques: Fault Types, Fault Detection. Fault Error containment Redundancy, Data Diversity, Reversal Checks, Integrated Failure handling
Reliability Evaluation Techniques: Obtaining Parameter Values, Reliability Models for Hardware Redundancy, Software Error models
Clock Synchronization: Clock, A Nonfault Tolerant Synchronization Algorithm, Impact of Faults, Fault Tolerant Synchronization in Hardware, Fault Tolerant Synchronization in Software

Self Study:
The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. C.M. Krishna, Kang G. Shin, Real Time Systems, McGraw Hill International Editions.
2. Stuart Bennett, Real Time Computer Control - An Introduction,PHI.
3. Peter D. Lawrence, Real time Micro Computer System Design – An Introduction, McGraw Hill.
4. S.T. Allworth and R.N. Zobel, Introduction to real time software design, Macmillan.
5. R.J.A Buhur, D.L. Bailey, An Introduction to Real Time Systems, PHI.
6. Philip. A. Laplante, Real Time System Design and Analysis, PHI.

3CS1201 Research Seminar [0 0 1 1]

Course Learning Outcome:

After successful completion of the course, student will be able to

- compare the work done by various researchers in the area of interest using literature survey
- identify the research gaps in the specific area of interest
- demonstrate practical aspect of research contribution
- comprehend and document the study and contribution made towards research

Syllabus:

Candidates have to select any Research Topic as Self Study for their Research Seminar. They will be required to present the progress of their Study in front of the Reviewing Panel at Regular intervals. During the final review, student are required to submit the report of Seminar.

3CS1E25 Semantic Web [3 0 1 4]

Course Learning Outcomes

After successful completion of this course, student will be able to

- understand the principles behind semantic web and Ontology Engineering
- model and design ontologies using Resource Description Framework (RDF) and Web Ontology Language (OWL)
- query ontologies using SPARQL
- apply semantic web technologies to real world applications

Syllabus

History, Semantic Web Layers, Semantic Web technologies, Semantics in Semantic Web

XML: Structuring, Namespaces, Addressing, Querying, Processing, Metadata, Traditional Search vs. Semantic Search

RDF and Semantic Web: RDF Specification , RDF Syntax, XML and Non-XML, RDF elements, RDF Relationship, Reification, Container and Collaboration , RDF Schema, Editing, Parsing, and Browsing RDF/XML, RQL, RDQL, SPARQL, Web Data Management,

Ontology: Ontology Movement, OWL, OWL Specification, OWL Elements, OWL Constructs, Simple and Complex, Ontology Engineering, Constructing Ontologies, Reusing ontologies, Ontology Reasoners, On To Knowledge Semantic Web architecture, Ontology Mapping, Alignment and Merging Tools, Ontology Editor

Logic and Inference: Logic, Description Logics, Rules, Monotonic Rules, Syntax, Semantics and Examples, Non Monotonic Rules, Motivation, Syntax, and Examples, Rule Markup in XML, Monotonic Rules and Non Monotonic Rules

Applications of Semantic Web Technologies: Commercial and Non Commercial Use of RDF, Sample Ontology, Web Services, Web Mining, Horizontal information, Data Integration, Future of Semantic Web

Self-Learning Component

To be decided by course coordinator at the beginning of semester, which will be a blend of one or more of the e-Learning Resources, Video Lectures, Online courses, tools, research material, web links etc. along with the related assessment component(s).

Laboratory Work

Above concepts are to be implemented and at least 5 experiments are to be carried out.

References

1. Grigorous Antoniou and Van Hermelen, A Semantic Web Primer, The MIT Press.
2. Dieter Fensel, Jim Hendler, Henry Lieberman and Wolfgang Wahister, Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential, The MIT Press.
3. Shelley Powers, Practical RDF, O'Reilly Publishers.
4. Manish Joshi, Harold Boley, Rajendra Akerkar, Advances in Semantic Computing, Techno Mathematics Research Foundation
5. Elliotte Rusty Harold, The XML 1.1 Bible, Wiley.

3CS1E04 Software Testing and Quality Assurance [3 0 1 4]

Course Learning Outcome:
After successful completion of this course, student will be able to

- learn and develop project documentations and soft skills for effective project presentation
- develop practical skills related to software quality assurance
- apply software testing techniques for information systems development

Syllabus:
Software Quality, Role of testing, verification and validation, White-Box and Black-Box Testing , Test Planning and Design, Monitoring and Measuring Test Execution, Test Tools and Automation, Test Team Organization and Management.
Unit Testing: Concept of Unit Testing, Static Unit Testing, Defect Prevention, Mutation Testing, Debugging, Unit Testing in eXtreme Programming
Control Flow Testing: Control Flow Graph, Paths in a Control Flow Graph, All-Path Coverage Criterion, Statement Coverage Criterion, Branch Coverage Criterion, Examples of Test Data Selection.
Data Flow Testing: Data Flow Anomaly, Data Flow Graph, Data Flow Testing Criteria, Feasible Paths and Test Selection Criteria, Comparison of Testing Techniques.
System Integration Testing: Types of Interfaces and Interface Errors, System Integration Techniques, Software and Hardware Integration, Off-the-Shelf Component Testing, Built-in Testing
System Test Categories: Basic Tests, Functionality Tests, Robustness Tests, Interoperability Tests, Performance Tests, Scalability Tests, Stress Tests, Load and Stability Tests, Reliability Tests, Regression Tests, Documentation Tests.
Functional Testing: Equivalence Class Partitioning, Boundary Value Analysis, Decision Tables, Random Testing, Error Guessing, Category Partition.
System Test Design: Test Design Factors, Requirement Identification, Characteristics of Testable Requirements, Test Design Preparedness Metrics, Test Case Design Effectiveness
System Test Planning and Automation: Structure of a System Test Plan, System Test Automation
System Test Execution: Metrics for Tracking System Test, Beta Testing, System Test Report, Product Sustaining, Measuring Test Effectiveness.
Acceptance Testing: Types of Acceptance Testing, Selection of Acceptance Criteria, Acceptance Test Execution, Acceptance Testing in eXtreme Programming.
Software Quality: Five Views of Software Quality, McCall's Quality Factors and Criteria, Quality Factors Quality Criteria, Relationship between Quality Factors and Criteria, Quality Metrics, ISO 9126 Quality Characteristics, ISO 9000:2000 Software Quality Standard ISO 9000:2000 Fundamentals, ISO 9001:2000 Requirements

Self Study:
The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References:

1. SagarNaik University of Waterloo, PiyuTripathy, Software Testing and Quality Assurance: Theory and Practice, Wiley.

2. William Perry, Effective Methods for Software Testing, Wiley.
3. Paul C. Jorgensen,Software Testing - A Craftsman's Approach, CRC Press.
4. Srinivasan Desikan and GopalaswamyRamesh,Software Testing, Pearson Education.
5. Louis Tamres,Introducing to Software Testing, Addison Wesley Publications.
6. Ron Patton, SAMS Techmedia Indian Edition,Software Testing, Pearson Education.
7. Glenford J. Myers,The Art of Software Testing, John Wiley & Sons.
8. Robert V. Binder, Testing Object-Oriented Systems: Models Patterns and Tools, Addison Wesley.
9. Daniel Galin, Software Quality Assurance, Pearson Education.

3CS1E24 Web Search and Mining [3 0 1 4]

Course Learning Outcome:
After successful completion of this course, student will be able to

- identify and differentiate between application areas for web content mining, web structure mining and web usage mining
- understand how web search engines crawl, index, and rank web content
- understand the functionality of the various web search and web mining methods and components

Syllabus:
Introduction to web mining, Taxonomy of web mining, Architecture of a general purpose web search engine such as google or alta vista Web Content Mining: document indexing and retrieval in the web environment - Boolean and vector retrieval models, latent semantic indexing (LSI), results ordering, meta-search, Indexing and Retrieval, Standard and extended query languages for web data. Indexing and retrieving text files by words. Database structure. Measuring the relevance of a text to a query, web documents categorization and clustering
Natural Language Processing Methods Used For Web Information Retrieval: lemmatization, part-of-speech tagging, disambiguation, shallow syntactic parsing etc.
Web Structure Mining: primary web browsing (crawling, spidering),focused crowling, link topology analysis, PageRank, HITS methods ,Global analysis of the Web
Social Networks Analysis Web Usage Mining: mining for user behavior on the web, internet marketing, Information Extraction as a specific type of web content mining: wrapper-based vs. token activated extraction
Specific Applications: opinion mining vs. fact mining, web spam analysis, comparative shopping, etc. Web information integration, mapping schemas usage
Web Mining And Its Relation To The Semantic Web: automatic semantic annotation, ontology learning, Semantic Web search The invisible web and specialized search engines.

Self Study:
The self study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self study contents.

Laboratory Work:
Laboratory work will be based on above syllabus with minimum 5 experiments to be incorporated.

References

1. SoumenCharkabarti, Mining The Web: Discovering Knowledge from Hypertext Data, Morgan Kaufmann.
2. RichardoBaeza-Yates and BerthierRibierno-Neto, Modern Information RetrievalAddison Wesley.
3. Alfred and Emily Glossbrenner, Search Engines for The World Wide Web, Peachpit Press.

3CS1301 Project Part I [00 30 13]

Course Learning Outcome:

After successful completion of the course, student will be able to

- compare the work done by various researchers in the area of interest using literature survey
- identify specific research gaps
- correlate other facets of research through innovative thinking and ideas
- demonstrate effective communication skills
- apply optimization and enhancement skillsforeffective research
- employ ethical practicesforresearch

Syllabus:

The student will carry out a project with significant technical contribution either in the institute, any R&D organization or Industry. At the end of the semester III, student will submit a report on the progress of his work.

3CS1401 Project Part II [00 30 14]

Course Learning Outcome:

After successful completion of the course, student will be able to

- compare the work done by various researchers in the area of interest using literature survey
- identify specific research gaps
- correlate other facets of research through innovative thinking and ideas
- demonstrate effective communication skills
- apply optimization and enhancement skills for effective research
- employ ethical practices for research

Syllabus:

The student will continue the project work started in semester III and complete the work defined and submit final dissertation for evaluation.