# DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY LUCKNOW

**Evaalluation Scheme & Syllabus**

**For**

**B.Tech. 3rd Year**

**ARTIFICIAL INTELLIGENCE & M ACHINE LEARNING**

**on**

**AIICTE Model Curriculum**

**(Effective from the Session: 2023-24**

DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY LUCKNOW

**B.TECH.**

**ARTIFICIAL INTELLIGENCE & MACHINE LEARNING CURRICULUM STRUCTURE**

| SEMESTER- V | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Sl. No.** | **Subject Codes** | **Subject** | **Periods** | | | **Evaluation Scheme** | | | | **End Semester** | | **Total** | **Credit** |
| | | | **L** | **T** | **P** | **CT** | **TA** | **Total** | **PS** | **TE** | **PE** | | |
| 1 | KCS501 | Database Management System | 3 | 1 | 0 | 30 | 20 | 50 | | 100 | | 150 | 4 |
| 2 | KAI501 | Artificial Intelligence | 3 | 1 | 0 | 30 | 20 | 50 | | 100 | | 150 | 4 |
| 3 | KCS503 | Design and Analysis of Algorithm | 3 | 1 | 0 | 30 | 20 | 50 | | 100 | | 150 | 4 |
| 4 | Dept. Elective-I | Departmental Elective-I | 3 | 0 | 0 | 30 | 20 | 50 | | 100 | | 150 | 3 |
| 5 | Dept. Elective-II | Departmental Elective-II | 3 | 0 | 0 | 30 | 20 | 50 | | 100 | | 150 | 3 |
| 6 | KCS551 | Database Management System Lab | 0 | 0 | 2 | | | | 25 | | 25 | 50 | 1 |
| 7 | KAI551 | Artificial Intelligence Lab | 0 | 0 | 2 | | | | 25 | | 25 | 50 | 1 |
| 8 | KCS553 | Design and Analysis of Algorithm Lab | 0 | 0 | 2 | | | | 25 | | 25 | 50 | 1 |
| 9 | KCS554 | Mini Project or Internship Assessment* | 0 | 0 | 2 | | | | 50 | | | 50 | 1 |
| 10 | KNC501/ KNC502 | Constitution of India. Law and Engineering / Indian Tradition, Culture and Society | 2 | 0 | 0 | 15 | 10 | 25 | | 50 | | | |
| 11 | | MOOCs (Essential for Hons. Degree) | | | | | | | | | | | |
| | | **Total** | | | | | | | | | | **950** | **22** |

*The Mini Project or internship (4 weeks) conducted during summer break after IV semester and will be assessed during V semester.

| SEMESTER- VI | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Sl. No.** | **Subject Codes** | **Subject** | **Periods** | | | **Evaluation Scheme** | | | | **End Semester** | | **Total** | **Credit** |
| | | | **L** | **T** | **P** | **CT** | **TA** | **Total** | **PS** | **TE** | **PE** | | |
| 1 | KAI601 | Machine Learning Techniques | 3 | 1 | 0 | 30 | 20 | 50 | | 100 | | 150 | 4 |
| 2 | KCS602 | Web Technology | 3 | 1 | 0 | 30 | 20 | 50 | | 100 | | 150 | 4 |
| 3 | KCS603 | Computer Networks | 3 | 1 | 0 | 30 | 20 | 50 | | 100 | | 150 | 4 |
| 4 | Dept. Elective-III | Departmental Elective-III | 3 | 0 | 0 | 30 | 20 | 50 | | 100 | | 150 | 3 |
| 5 | | Open Elective-I [Annexure - B(iv)] | 3 | 0 | 0 | 30 | 20 | 50 | | 100 | | 150 | 3 |
| 6 | KAI651 | Machine learning Lab | 0 | 0 | 2 | | | | 25 | | 25 | 50 | 1 |
| 7 | KCS652 | Web Technology Lab | 0 | 0 | 2 | | | | 25 | | 25 | 50 | 1 |
| 8 | KCS653 | Computer Networks Lab | 0 | 0 | 2 | | | | 25 | | 25 | 50 | 1 |
| 9 | KNC601/ KNC602 | Constitution of India. Law and Engineering / Indian Tradition, Culture and Society | 2 | 0 | 0 | 15 | 10 | 25 | | 50 | | | |
| 10 | | MOOCs (Essential for Hons. Degree) | | | | | | | | | | | |
| | | **Total** | | | | | | | | | | **900** | **21** |

**Departmental Elective-I**

1. KAI051 Mathematical Foundation AI , ML and Data Science
2. KCS052 Web Designing
3. KDS051 Business Intelligence and Analytics
4. KCS054 Object Oriented System Design
5. KDS052 Distributed System

**Departmental Elective-II**

1. KML051 Cloud Computing
2. KAI052 Natural Language Processing
3. KCS056 Application of Soft Computing
4. KAI053 Intelligent Database System
5. KCS502 Compiler Design

**Departmental Elective-III**

1. KAI061 Cyber Forensic analytics
2. KDS061 Image Analytics
3. KML061 Advanced Machine Learning
4. KML062 Stream Processing and Analytics
6. KDS063 Software Engineering

**B.TECH.**

# Artificial Intelligence & Machine Learning

**FIFTH SEMESTER (DETAILED SYLLABUS)**

| **KCS 501** | **DATABASE MANAGEMENT SYSTEM** | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to understand** | | |
| CO 1 | Apply knowledge of database for real life applications. | $K_3$ |
| CO 2 | Apply query processing techniques to automate the real time problems of databases. | $K_3$, $K_4$ |
| CO 3 | Identify and solve the redundancy problem in database tables using normalization. | $K_2$, $K_3$ |
| CO 4 | Understand the concepts of transactions, their processing so they will familiar with broad range of database management issues including data integrity, security and recovery. | $K_2$, $K_4$ |
| CO 5 | Design, develop and implement a small database project using database tools. | $K_3$, $K_6$ |

| **DETAILED SYLLABUS** | | **3-1-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction:** Overview, Database System vs File System, Database System Concept and Architecture, Data Model Schema and Instances, Data Independence and Database Language and Interfaces, Data Definitions Language, DML, Overall Database Structure. Data Modeling Using the Entity Relationship Model: ER Model Concepts, Notation for ER Diagram, Mapping Constraints, Keys, Concepts of Super Key, Candidate Key, Primary Key, Generalization, Aggregation, Reduction of an ER Diagrams to Tables, Extended ER Model, Relationship of Higher Degree. | **08** |
| **II** | **Relational data Model and Language:** Relational Data Model Concepts, Integrity Constraints, Entity Integrity, Referential Integrity, Keys Constraints, Domain Constraints, Relational Algebra, Relational Calculus, Tuple and Domain Calculus. Introduction on SQL: Characteristics of SQL, Advantage of SQL. SQl Data Type and Literals. Types of SQL Commands. SQL Operators and Their Procedure. Tables, Views and Indexes. Queries and Sub Queries. Aggregate Functions. Insert, Update and Delete Operations, Joins, Unions, Intersection, Minus, Cursors, Triggers, Procedures in SQL/PL SQL | **08** |
| **III** | **Data Base Design & Normalization:** Functional dependencies, normal forms, first, second, 8 third normal forms, BCNF, inclusion dependence, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design | **08** |
| **IV** | **Transaction Processing Concept:** Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializable Schedule, Recoverability, Recovery from Transaction Failures, Log Based Recovery, Checkpoints, Deadlock Handling. Distributed Database: Distributed Data Storage, Concurrency Control, Directory System. | **08** |
| **V** | **Concurrency Control Techniques:** Concurrency Control, Locking Techniques for Concurrency Control, Time Stamping Protocols for Concurrency Control, Validation Based Protocol, Multiple Granularity, Multi Version Schemes, Recovery with Concurrent Transaction, Case Study of Oracle. | **08** |

**Text books:**


1. Korth, Silbertz, Sudarshan," Database Concepts", McGraw Hill
2. Date C J, "An Introduction to Database Systems", Addision Wesley
3. Elmasri, Navathe, " Fundamentals of Database Systems", Addision Wesley
4. O'Neil, Databases, Elsevier Pub.
5. RAMAKRISHNAN"Database Management Systems",McGraw Hill
6. Leon &Leon,"Database Management Systems", Vikas Publishing House
7. Bipin C. Desai, " An Introduction to Database Systems", Gagotia Publications
8. Majumdar& Bhattacharya, "Database Management System", TMH

| KAI501 | ARTIFICIAL INTELLIGENCE | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to understand** | | |
| CO 1 | Understand the basics of the theory and practice of Artificial Intelligence as a discipline and about intelligent agents. | $K_2$ |
| CO 2 | Understand search techniques and gaming theory. | $K_2$, $K_3$ |
| CO 3 | The student will learn to apply knowledge representation techniques and problem solving strategies to common AI applications. | $K_3$ , $K_4$ |
| CO 4 | Student should be aware of techniques used for classification and clustering. | $K_2$ , $K_3$ |
| CO 5 | Student should aware of basics of pattern recognition and steps required for it. | $K_2$ , $K_4$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **INTRODUCTION :**<br>Introduction–Definition – Future of Artificial Intelligence – Characteristics of Intelligent Agents– Typical Intelligent Agents – Problem Solving Approach to Typical AI problems. | **08** |
| **II** | **PROBLEM SOLVING METHODS**<br>Problem solving Methods – Search Strategies- Uninformed – Informed – Heuristics – Local Search Algorithms and Optimization Problems – Searching with Partial Observations – Constraint Satisfaction Problems – Constraint Propagation – Backtracking Search – Game Playing – Optimal Decisions in Games – Alpha – Beta Pruning – Stochastic Games | **08** |
| **III** | **KNOWLEDGE REPRESENTATION**<br>First Order Predicate Logic – Prolog Programming – Unification – Forward Chaining-Backward Chaining – Resolution – Knowledge Representation – Ontological Engineering-Categories and Objects – Events – Mental Events and Mental Objects – Reasoning Systems for Categories – Reasoning with Default Information | **08** |
| **IV** | **SOFTWARE AGENTS**<br>Architecture for Intelligent Agents – Agent communication – Negotiation and Bargaining – Argumentation among Agents – Trust and Reputation in Multi-agent systems. | **08** |
| **V** | **APPLICATIONS**<br>AI applications – Language Models – Information Retrieval- Information Extraction – Natural Language Processing – Machine Translation – Speech Recognition – Robot – Hardware – Perception – Planning – Moving | **08** |

**Text books:**


1. S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach‖, Prentice Hall, Third Edition, 2009.
2. I. Bratko, ―Prolog: Programming for Artificial Intelligence‖, Fourth edition, Addison-Wesley Educational Publishers Inc., 2011.
3. M. Tim Jones, ―Artificial Intelligence: A Systems Approach(Computer Science)‖, Jones and Bartlett Publishers, Inc.; First Edition, 2008
4. Nils J. Nilsson, ―The Quest for Artificial Intelligence‖, Cambridge University Press, 2009.
5. William F. Clocksin and Christopher S. Mellish,‖ Programming in Prolog: Using the ISO Standard‖, Fifth Edition, Springer, 2003.
6. Gerhard Weiss, ―Multi Agent Systems‖, Second Edition, MIT Press, 2013.
7. David L. Poole and Alan K. Mackworth, ―Artificial Intelligence: Foundations of Computational Agents‖, Cambridge University Press, 2010.

| KCS 503 | DESIGN AND ANALYSIS OF ALGORITHM | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| | **At the end of course , the student will be able to understand** | |
| CO 1 | Design new algorithms, prove them correct, and analyze their asymptotic and absolute runtime and memory demands. | $K_4$, $K_6$ |
| CO 2 | Find an algorithm to solve the problem (create) and prove that the algorithm solves the problem correctly (validate). | $K_5$, $K_6$ |
| CO 3 | Understand the mathematical criterion for deciding whether an algorithm is efficient, and know many practically important problems that do not admit any efficient algorithms. | $K_2$, $K_5$ |
| CO 4 | Apply classical sorting, searching, optimization and graph algorithms. | $K_2$, $K_4$ |
| CO 5 | Understand basic techniques for designing algorithms, including the techniques of recursion, divide-and-conquer, and greedy. | $K_2$, $K_3$ |

| **DETAILED SYLLABUS** | | **3-1-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction:** Algorithms, Analyzing Algorithms, Complexity of Algorithms, Growth of Functions, Performance Measurements, Sorting and Order Statistics - Shell Sort, Quick Sort, Merge Sort, Heap Sort, Comparison of Sorting Algorithms, Sorting in Linear Time. | **08** |
| **II** | **Advanced Data Structures:** Red-Black Trees, B – Trees, Binomial Heaps, Fibonacci Heaps, Tries, Skip List | **08** |
| **III** | **Divide and Conquer** with Examples Such as Sorting, Matrix Multiplication, Convex Hull and Searching.<br>**Greedy Methods** with Examples Such as Optimal Reliability Allocation, Knapsack, Minimum Spanning Trees – Prim's and Kruskal's Algorithms, Single Source Shortest Paths - Dijkstra's and Bellman Ford Algorithms. | **08** |
| **IV** | **Dynamic Programming** with Examples Such as Knapsack. All Pair Shortest Paths – Warshal's and Floyd's Algorithms, Resource Allocation Problem. Backtracking, Branch and Bound with Examples Such as Travelling Salesman Problem, Graph Coloring, n-Queen Problem, Hamiltonian Cycles and Sum of Subsets. | **08** |
| **V** | **Selected Topics:** Algebraic Computation, Fast Fourier Transform, String Matching, Theory of NP-Completeness, Approximation Algorithms and Randomized Algorithms | **08** |

**Text books:**


1. Thomas H. Coreman, Charles E. Leiserson and Ronald L. Rivest, "Introduction to Algorithms", Printice Hall of India.
2. E. Horowitz & S Sahni, "Fundamentals of Computer Algorithms",
3. Aho, Hopcraft, Ullman, "The Design and Analysis of Computer Algorithms" Pearson Education, 2008.
4. LEE "Design & Analysis of Algorithms (POD)",McGraw Hill
5. Richard E.Neapolitan "Foundations of Algorithms" Jones & Bartlett Learning
6. Jon Kleinberg and ÉvaTardos, Algorithm Design, Pearson, 2005.
7. Michael T Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2006.
8. Harry R. Lewis and Larry Denenberg, Data Structures and Their Algorithms, Harper Collins, 1997
9. Robert Sedgewick and Kevin Wayne, Algorithms, fourth edition, Addison Wesley, 2011.
10. Harsh Bhasin,"Algorithm Design and Analysis",FirstEdition,Oxford University Press.
11. Gilles Brassard and Paul Bratley,Algorithmics:Theory and Practice,Prentice Hall,1995.

| KAI 051 | MATHEMATICAL FOUNDATION AI , ML AND DATA SCIENCE | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Understand and apply the probability distributions, random number generation and density estimations to perform analysis of various kinds of data | K2, $K_4$, $K_6$ |
| CO 2 | Understand and manipulate data, design and perform simple Monte Carlo experiments, and be able to use resampling methods | $K_5$, $K_6$ |
| CO 3 | Perform statistical analysis on variety of data | $K_2$, $K_5$ |
| CO 4 | Perform appropriate statistical tests using R and visualize the outcome | $K_2$, $K_4$ |
| CO 5 | Discuss the results obtained from their analyses after creating customized graphical and numerical summaries | $K_2$, $K_3$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Descriptive Statistics:** Diagrammatic representation of data, measures of central tendency, measures of dispersion, measures of skewness and kurtosis, correlation, inference procedure for correlation coefficient, bivariate correlation, multiple correlations, linear regression and its inference procedure, multiple regression.<br>**Probability:** Measures of probability, conditional probability, independent event, Bayes' theorem, random variable, discrete and continuous probability distributions, expectation and variance, markov inequality, chebyshev's inequality, central limit theorem. | **08** |
| **II** | **Inferential Statistics:** Sampling & Confidence Interval, Inference & Significance. Estimation and Hypothesis Testing, Goodness of fit, Test of Independence, Permutations and Randomization Test, t-test/z-test (one sample, independent, paired), ANOVA, chi-square.<br>**Linear Methods for Regression Analysis:** multiple regression analysis, orthogonalization by Householder transformations (QR); singular value decomposition (SVD); linear dimension reduction using principal component analysis (PCA). | **08** |
| **III** | **Pseudo-Random Numbers:** Random number generation, Inverse-transform, acceptance-rejection, transformations, multivariate probability calculations.<br>**Monte Carlo Integration:** Simulation and Monte Carlo integration, variance reduction, Monte Carlo hypothesis testing, antithetic variables/control variates, importance sampling, stratified sampling<br>Markov chain Monte Carlo (McMC): Markov chains; Metropolis-Hastings algorithm; Gibbs sampling; convergence | **08** |
| **IV** | **Vector Spaces-** Vector Space, Subspace , Linear Combination, Linear Independence, Basis, Dimension, Finding a Basis of a Vector Space , Coordinates, Change of Basis<br>**Inner Product Spaces-** Inner Product, Length, Orthogonal Vectors, Triangle Inequality, Cauchy-Schwarz Inequality, Orthonormal (Orthogonal) Basis, Gram-Schmidt Process | **08** |
| **V** | **Linear Transformations**- Linear Transformations and Matrices for Linear Transformation, Kernel and Range of a Linear Transformations, Change of Basis<br>**Eigenvalues and Eigenvectors**- Definition of Eigenvalue and Eigenvector, Diagonalization , Symmetric Matrices and Orthogonal Diagonalization | **08** |

**References:**


1. S.C. Gupta & V.K. Kapoor, "Fundamentals of Mathematical Statistics", Sultan Chand & Sons
2. Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", Academic Press.
3. Dudewicz, E.J., Mishra, S.N., "Modern Mathematical Statistics", Willy
4. Purohit S. G., Gore S. D., Deshmukh S. K., "Statistics using R, Narosa
5. Rizzo, M. L., "Statistical Computing with R", Boca Raton, FL: Chapman & Hall/CRC Press
6. Normal Maltoff, The Art of R programming, William
7. Dalgaard, Peter, "Introductory statistics with R", Springer Science & Business Media
8. M. D. Ugarte, A. F. Militino, A. T. Arnholt, "Probability and Statistics with R", CRC Press
9. Kundu, D. and Basu, A., "Statistical computing – existing methods and recent developments", Narosa
10. Gentle, James E., Härdle, Wolfgang Karl, Mori, Yuich, "Handbook of Computational Statistics", Springer
11. Givens and Hoeting, "Computational Statistics", Wiley Series in Prob. and Statistics
12. Elementary Linear Algebra by Ron Larson, 8th edition, Cengage Learning, 2017

| KCS 052 | WEB DESIGNING | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Understand principle of Web page design and about types of websites | $K_3$, $K_4$ |
| CO 2 | Visualize and Recognize the basic concept of HTML and application in web designing. | $K_1$, $K_2$ |
| CO 3 | Recognize and apply the elements of Creating Style Sheet (CSS). | $K_2$, $K_4$ |
| CO 4 | Understand the basic concept of Java Script and its application. | $K_2$, $K_3$ |
| CO 5 | Introduce basics concept of Web Hosting and apply the concept of SEO | $K_2$, $K_3$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction :** Basic principles involved in developing a web site, Planning process , Domains and Hosting, Responsive Web Designing , Types of Websites (Static and Dynamic Websites), Web Standards and W3C recommendations,<br>**Introduction to HTML:** What is HTML , HTML Documents, Basic structure of an HTML document , Creating an HTML document , Mark up Tags , Heading-Paragraphs , Line Breaks | **08** |
| **II** | **Elements of HTML:** HTML Tags., Working with Text , Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls | **08** |
| **III** | **Concept of CSS:** Creating Style Sheet, CSS Properties , CSS Styling(Background, Text Format, Controlling Fonts) , Working with block elements and objects , Working with Lists and Tables , CSS Id and Class, Box Model(Introduction, Border properties, Padding Properties, Margin properties) CSS Advanced(Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute sector) , CSS Color , Creating page Layout and Site Designs. | **08** |
| **IV** | Introduction to Client Side Scripting , Introduction to Java Script , Javascript Types , Variables in JS, Operators in JS , Conditions Statements , Java Script Loops, JS Popup Boxes , JS Events , JS Arrays, Working with Arrays, JS Objects ,JS Functions , Using Java Script in Real time , Validation of Forms, Related Examples | **08** |
| **V** | **Web Hosting:** Web Hosting Basics , Types of Hosting Packages, Registering domains , Defining Name Servers , Using Control Panel, Creating Emails in Cpanel , Using FTP Client, Maintaining a Website<br>**Concepts of SEO :** Basics of SEO, Importance of SEO, Onpage Optimization Basics | **08** |

**Text Books:**


1. Steven M. Schafer, "HTML, XHTML, and CSS Bible, 5ed", Wiley India
2. Ian Pouncey, Richard York, "Beginning CSS: Cascading Style Sheets for Web Design", Wiley India

| KDS 051 | BUSINESS INTELLIGENCE AND ANALYTICS | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Understand the essentials of BI & data analytics and the corresponding terminologies | $K_2$ |
| CO 2 | Analyze the steps involved in the BI - Analytics process | $K_3, K_4$ |
| CO 3 | Illustrate competently on the topic of analytics | $K_2, K_3$ |
| CO 4 | Understand & Implement the K-Means Clustering with Iris Dataset | $K_2, K_3$ |
| CO 5 | Demonstrate the real time scenario (Case study) by using BI & Analytics techniques | $K_5, K_6$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **BUSINESS INTELLIGENCE – INTRODUCTION:** Introduction - History and Evolution: Effective and Timely decisions, Data Information and Knowledge, Architectural Representation, Role of mathematical Models, Real Time Business Intelligent System. | **8** |
| **II** | **BI – DATA MINING & WAREHOUSING:**Data Mining - Introduction to Data Mining, Architecture of Data Mining and How Data mining works(Process) , Functionalities & Classifications of Data Mining, Representation of Input Data, Analysis Methodologies. Data Warehousing - Introduction to Data Warehousing, Data Mart, Online Analytical Processing (OLAP) – Tools, Data Modelling, Difference between OLAP and OLTP, Schema – Star and Snowflake Schemas, ETL Process – Role of ETL | **8** |
| **III** | **BI – DATA PREPARTTION:** Data Validation - Introduction to Data Validation, Data Transformation – Standardization and Feature Extraction, Data Reduction – Sampling, Selection, PCA, Data Discretization | **8** |
| **IV** | **BI – DATA ANALYTICS PROCESS -** Introduction to analytics process, Types of Analytical Techniques in BI –Descriptive, Predictive, Perspective, Social Media Analytics, Behavioral, Iris Datasets | **8** |
| **V** | **IMPLEMENTATION OF BI –** Business Activity Monitoring, Complex Event Processing, Business Process Management, Metadata, Root Cause Analysis. | **8** |

**Text Books:**


1. Carlo-Vercellis, "Business Intelligence Data Mining and Optimization for Decision-Making", First Edition
2. Drew Bentely, "Business Intelligence and Analytics" ,@2017 Library Pres., ISBN: 978-1-9789-2136-8
3. Larissa T. Moss & Shaku Atre, "Business Intelligence Roadmap: The Complete Project Lifecycle
4. For Decision-Support Applications", First Edition, Addison-Wesley Professional,2003
5. Kimball, R., Ross, M., Thornthwaite, W., Mundy, J., and Becker, B. John, "The Data Warehouse
6. Lifecycle Toolkit: Practical Techniques for Building Data Warehouse and Business Intelligence Systems", Second Edition, Wiley & Sons, 2008.
7. Cindi Howson, "Successful Business Intelligence", Second Edition, McGraw-Hill Education, 2013.

| KCS 054 | OBJECT ORIENTED SYSTEM DESIGN | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Understand the application development and analyze the insights of object oriented programming to implement application | $K_2$, $K_4$ |
| CO 2 | Understand, analyze and apply the role of overall modeling concepts (i.e. System, structural) | $K_2$, $K_3$ |
| CO 3 | Understand, analyze and apply oops concepts (i.e. abstraction, inheritance) | $K_2$, $K_3$, $K_4$ |
| CO 4 | Understand the basic concepts of C++ to implement the object oriented concepts | $K_2$, $K_3$ |
| CO 5 | To understand the object oriented approach to implement real world problem. | $K_2$, $K_3$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction:** The meaning of Object Orientation, object identity, Encapsulation, information hiding, polymorphism, generosity, importance of modelling, principles of modelling, object oriented modelling, Introduction to UML, conceptual model of the UML, Architecture. | **08** |
| **II** | **Basic Structural Modeling:** Classes, Relationships, common Mechanisms, and diagrams. Class &Object Diagrams: Terms, concepts, modelling techniques for Class & Object Diagrams.<br>**Collaboration Diagrams:** Terms, Concepts, depicting a message, polymorphism in collaboration Diagrams, iterated messages, use of self in messages. Sequence Diagrams: Terms, concepts, depicting asynchronous messages with/without priority, call-back mechanism, broadcast messages.<br>**Basic Behavioural Modeling:** Use cases, Use case Diagrams, Activity Diagrams, State Machine , Process and thread, Event and signals, Time diagram, interaction diagram, Package diagram.<br>**Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams. | **08** |
| **III** | Object Oriented Analysis: Object oriented design, Object design, Combining three models, Designing algorithms, design optimization, Implementation of control, Adjustment of inheritance, Object representation, Physical packaging, Documenting design considerations.<br>**Structured analysis and structured design (SA/SD),** Jackson Structured Development (JSD).Mapping object oriented concepts using non-object oriented language, Translating classes into data structures, Passing arguments to methods, Implementing inheritance, associations encapsulation.<br>**Object oriented programming style:** reusability, extensibility, robustness, programming in the large. Procedural v/s OOP, Object oriented language features. Abstraction and Encapsulation. | **08** |
| **IV** | **C++ Basics :** Overview, Program structure, namespace, identifiers, variables, constants, enum, operators, typecasting, control structures<br>**C++ Functions :** Simple functions, Call and Return by reference, Inline functions, Macro Vs. Inline functions, Overloading of functions, default arguments, friend functions, virtual functions | **08** |
| **V** | **Objects and Classes:** Basics of object and class in C++, Private and public members, static data and function members, constructors and their types, destructors, operator overloading, type conversion. Inheritance : Concept of Inheritance, types of inheritance: single, multiple, multilevel, hierarchical, hybrid, protected members, overriding, virtual base class<br>**Polymorphism :** Pointers in C++, Pointes and Objects, this pointer, virtual and pure virtual functions, Implementing polymorphism | **08** |

Text Books


1. James Rumbaughet. al, "Object Oriented Modeling and Design", PHI
2. Grady Booch, James Rumbaugh, Ivar Jacobson, "The Unified Modeling Language User Guide", Pearson Education
3. Object Oriented Programming With C++, E Balagurusamy, TMH
4. C++ Programming, Black Book, Steven Holzner, dreamtech
5. Object Oriented Programming in Turbo C++, Robert Lafore, Galgotia
6. Object Oriented Programming with ANSI and Turbo C++, Ashok Kamthane, Pearson

7. The Compete Reference C++, Herbert Schlitz, TMH

| KDS 052 | DISTRIBUTED SYSTEM | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to understand** | | |
| CO 1 | To provide hardware and software issues in modern distributed systems. | K1 , K2 |
| CO 2 | To get knowledge in distributed architecture, naming, synchronization, consistency and replication, fault tolerance, security, and distributed file systems. | K2 |
| CO 3 | To analyze the current popular distributed systems such as peer-to-peer (P2P) systems will also be analyzed. | K4 |
| CO 4 | To know about Shared Memory Techniques and have Sufficient knowledge about file access | K1 |
| CO 5 | Have knowledge of Synchronization and Deadlock. | K1 |
| **DETAILED SYLLABUS** | | **3-0-0** |

| Unit | Topic | Proposed Lecture |
|---|---|---|
| **I** | **Characterization of Distributed Systems**: Introduction, Examples of distributed Systems, Resource sharing and the Web Challenges. Architectural models, Fundamental Models. Theoretical Foundation for Distributed System: Limitation of Distributed system, absence of global clock, shared memory, Logical clocks ,Lamport's& vectors logical clocks. Concepts in Message Passing Systems: causal order, total order, total causal order, Techniques for Message Ordering, Causal ordering of messages, global state, termination detection. | **08** |
| **II** | **Distributed Mutual Exclusion:** Classification of distributed mutual exclusion, requirement of mutual exclusion theorem, Token based and non token based algorithms, performance metric for distributed mutual exclusion algorithms. Distributed Deadlock Detection: system model, resource Vs communication deadlocks, deadlock prevention, avoidance, detection & resolution, centralized dead lock detection, distributed dead lock detection, path pushing algorithms, edge chasing algorithms. | **08** |
| **III** | **Agreement Protocols:** Introduction, System models, classification of Agreement Problem, Byzantine agreement problem, Consensus problem, Interactive consistency Problem, Solution to Byzantine Agreement problem, Application of Agreement problem, Atomic Commit in Distributed Database system. Distributed Resource Management: Issues in distributed File Systems, Mechanism for building distributed file systems, Design issues in Distributed Shared Memory, Algorithm for Implementation of Distributed Shared Memory. | **08** |
| **IV** | Failure Recovery in Distributed Systems: Concepts in Backward and Forward recovery, Recovery in Concurrent systems, Obtaining consistent Checkpoints, Recovery in Distributed Database Systems. Fault Tolerance: Issues in Fault Tolerance, Commit Protocols, Voting protocols, Dynamic voting protocols | **08** |
| **V** | **Transactions and Concurrency Control**: Transactions, Nested transactions, Locks, Optimistic Concurrency control, Timestamp ordering, Comparison of methods for concurrency control. Distributed Transactions: Flat and nested distributed transactions, Atomic Commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery. Replication: System model and group communication, Fault - tolerant services, highly available services, Transactions with replicated data. | **08** |

**Text books:**


1. Singhal&Shivaratri, "Advanced Concept in Operating Systems", McGraw Hill
2. Ramakrishna,Gehrke," Database Management Systems", McGraw Hill
3. Vijay K.Garg Elements of Distributed Compuitng , Wiley
4. Coulouris, Dollimore, Kindberg, "Distributed System: Concepts and Design", Pearson Education
5. Tenanuanbaum, Steen," Distributed Systems", PHI

| KML051 | CLOUD COMPUTING | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to understand** | | |
| CO 1 | Describe architecture and underlying principles of cloud computing. | $K_3$ |
| CO 2 | Explain need, types and tools of Virtualization for cloud. | $K_3$, $K_4$ |
| CO 3 | Describe Services Oriented Architecture and various types of cloud services. | $K_2$, $K_3$ |
| CO 4 | Explain Inter cloud resources management cloud storage services and their providers Assess security services and standards for cloud computing. | $K_2$, $K_4$ |
| CO 5 | Analyze advanced cloud technologies. | $K_3$, $K_6$ |

| **DETAILED SYLLABUS** | | **3-1-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction To Cloud Computing:** Definition of Cloud – Evolution of Cloud Computing – Underlying Principles of Parallel and Distributed Computing – Cloud Characteristics – Elasticity in Cloud – On-demand Provisioning. | **08** |
| **II** | **Cloud Enabling Technologies Service Oriented Architecture:** REST and Systems of Systems – Web Services – Publish, Subscribe Model – Basics of Virtualization – Types of Virtualization – Implementation Levels of Virtualization – Virtualization Structures – Tools and Mechanisms – Virtualization of CPU – Memory – I/O Devices –Virtualization Support and Disaster Recovery. | **08** |
| **III** | **Cloud Architecture, Services And Storage:** Layered Cloud Architecture Design – NIST Cloud Computing Reference Architecture – Public, Private and Hybrid Clouds – laaS – PaaS – SaaS – Architectural Design Challenges – Cloud Storage – Storage-as-a-Service – Advantages of Cloud Storage – Cloud Storage Providers – S3. | **08** |
| **IV** | **Resource Management And Security In Cloud:** Inter Cloud Resource Management – Resource Provisioning and Resource Provisioning Methods – Global Exchange of Cloud Resources – Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Virtual Machine Security – IAM – Security Standards. | **08** |
| **V** | **Cloud Technologies And Advancements Hadoop:** MapReduce – Virtual Box — Google App Engine – Programming Environment for Google App Engine —- Open Stack – Federation in the Cloud – Four Levels of Federation – Federated Services and Applications – Future of Federation. | **08** |

**Text books:**


1. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
2. Rittinghouse, John W., and James F. Ransome, —Cloud Computing: Implementation, Management and Security, CRC Press, 2017.
3. RajkumarBuyya, Christian Vecchiola, S. ThamaraiSelvi, —Mastering Cloud Computing, Tata Mcgraw Hill, 2013.
4. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing – A Practical Approach, Tata Mcgraw Hill, 2009.
5. George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice), O'Reilly, 2009.

| KAI 052 | NATURAL LANGUAGE PROCESSING | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able :** | | |
| CO 1 | To learn the fundamentals of natural language processing | $K_1$ , $K_2$ |
| CO 2 | To understand the use of CFG and PCFG in NLP | $K_1$ , $K_2$ |
| CO 3 | To understand the role of semantics of sentences and pragmatic | $K_2$ |
| CO 4 | To Introduce Speech Production And Related Parameters Of Speech. | $K_1$ , $K_2$ |
| CO 5 | To Show The Computation And Use Of Techniques Such As Short Time Fourier Transform, Linear Predictive Coefficients And Other Coefficients In The Analysis Of Speech. | $K_3$, $K_4$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **INTRODUCTION:** Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance<br>**WORD LEVEL ANALYSIS :** Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Word Tokenization,Math with words TF-IDF Vectors, Finding meaning in word count ( Semantic Analysis), Linguistic Background: Outline of English Syntax, Introduction to Semantics and Knowledge Representation,Zipf's Law | **08** |
| **II** | **SYNTACTIC ANALYSIS:** Context Free Grammars, Grammar rules for English, Grammars and Parsing: Grammars and sentence Structure, Top-Down and Bottom-Up Parsers, Transition Network Grammars, Top- Down Chart Parsing. Feature Systems and Augmented Grammars: Basic Feature system for English, Morphological Analysis and the Lexicon, Parsing with Features, Augmented Transition Networks.<br>Ambiguity Resolution: Statistical Methods, Probabilistic Language Processing, Estimating Probabilities, Part-of-Speech tagging, Obtaining Lexical Probabilities, Probabilistic Context-Free Grammars, Best First Parsing Feature structures, Unification of feature structures. | **08** |
| **III** | **SEMANTICS AND PRAGMATICS:**<br>Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods. | **08** |
| **IV** | **BASIC CONCEPTS of Speech Processing :**Speech Fundamentals: Articulatory Phonetics – Production And Classification Of Speech Sounds; Acoustic Phonetics – Acoustics Of Speech Production; Review Of Digital Signal Processing Concepts; Short-Time Fourier Transform, Filter-Bank And LPC Methods. | **08** |
| **V** | **SPEECH-ANALYSIS**: Features, Feature Extraction And Pattern Comparison Techniques: Speech Distortion Measures– Mathematical And Perceptual<br>Real World NLP Challenges-Information Extraction and Question Answering,DialogEngines,Optimization,Parallelization and batch processing | **08** |

**Text books:**


1. Daniel Jurafsky, James H. Martin―Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, Pearson Publication, 2014.
2. Steven Bird, Ewan Klein and Edward Loper, ―Natural Language Processing with Python, First Edition, OReilly Media, 2009.
3. Lawrence RabinerAndBiing-Hwang Juang, "Fundamentals Of Speech Recognition", Pearson Education, 2003.
4. Daniel JurafskyAnd James H Martin, "Speech And Language Processing – An Introduction To Natural Language Processing, Computational Linguistics, And Speech Recognition", Pearson Education, 2002.

5. Frederick Jelinek, "Statistical Methods Of Speech Recognition", MIT Press, 1997.
6. 1. Breck Baldwin, —Language Processing with Java and LingPipe Cookbook, Atlantic Publisher, 2015.
7. Richard M Reese, —Natural Language Processing with Java, OReilly Media, 2015.
8. Nitin Indurkhya and Fred J. Damerau, —Handbook of Natural Language Processing, Second Edition, Chapman and Hall/CRC Press, 2010.
9. Tanveer Siddiqui, U.S. Tiwary, —Natural Language Processing and Information Retrieval, Oxford University Press, 2008.

**KCS 056 APPLICATION OF SOFT COMPUTING**

| Course Outcome ( CO) | | Bloom's Knowledge Level (KL) |
|---|---|---|
| **At the end of course , the student will be able to :** | | |
| CO 1 | Recognize the feasibility of applying a soft computing methodology for a particular problem | $K_2$, $K_4$ |
| CO 2 | Understand the concepts and techniques of soft computing and foster their abilities in designing and implementing soft computing based solutions for real-world and engineering problems. | K2,$K_4$, $K_6$ |
| CO 3 | Apply neural networks to pattern classification and regression problems and compare solutions by various soft computing approaches for a given problem. | $K_3$, $K_5$ |
| CO 4 | Apply fuzzy logic and reasoning to handle uncertainty and solve engineering problems | $K_3$, $K_4$ |
| CO 5 | Apply genetic algorithms to combinatorial optimization problems | K3, K5 |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Neural Networks-I (Introduction & Architecture) :**Neuron, Nerve structure and synapse, Artificial Neuron and its model, activation functions, Neural network architecture: single layer and multilayer feed forward networks, recurrent networks. Various learning techniques; perception and convergence rule, Auto-associative and hetro-associative memory. | **08** |
| **II** | **Neural Networks-II (Back propagation networks):** Architecture: perceptron model, solution, single layer artificial neural network, multilayer perception model; back propagation learning methods, effect of learning rule co-efficient ;back propagation algorithm, factors affecting backpropagation training, applications. | **08** |
| **III** | **Fuzzy Logic-I (Introduction):** Basic concepts of fuzzy logic, Fuzzy sets and Crisp sets, Fuzzy set theory and operations, Properties of fuzzy sets, Fuzzy and Crisp relations, Fuzzy to Crisp conversion. | **08** |
| **IV** | **Fuzzy Logic –II (Fuzzy Membership, Rules)** : Membership functions, interference in fuzzy logic, fuzzy if-then rules, Fuzzy implications and Fuzzy algorithms, Fuzzyfications&Defuzzificataions, Fuzzy Controller, Industrial applications | **08** |
| **V** | **Genetic Algorithm(GA):** Basic concepts, working principle, procedures of GA, flow chart of GA, Genetic representations, (encoding) Initialization and selection, Genetic operators, Mutation, Generational Cycle, applications. | **08** |

**Text books:**

1. S. Rajsekaran& G.A. VijayalakshmiPai, "Neural Networks,Fuzzy Logic and Genetic Algorithm:Synthesis and Applications" Prentice Hall of India.
2. N.P.Padhy,"Artificial Intelligence and Intelligent Systems" Oxford University Press. Reference Books:
3. SimanHaykin,"Neural Networks"Prentice Hall of India
4. Saroj Kaushik, Sunita Tiwari, "Soft Computing: Fundamentals, Techniques and Applications", McGraw Hill Education
5. Timothy J. Ross, "Fuzzy Logic with Engineering Applications" Wiley India.

6. Kumar Satish, “Neural Networks” Tata Mc Graw Hill

| KAI 053 | INTELLIGENT DATABASE SYSTEM | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom’s Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Understand the concepts of Intelligent database. | $K_2$ |
| CO 2 | Make study of the Database installation then create the database with user and apply SQL. | $K_2$, $K_3$ |
| CO 3 | Understand the concepts of knowledge-based systems and apply with AI | $K_2$, $K_3$ |
| CO 4 | Design and create the small applications | $K_5$, $K_6$ |
| CO 5 | Analyze and Implement for various real-time applications in Intelligent Database System | $K_4$, $K_5$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction to IDBS-**<br>Informal definition of the domain - General characteristics of IDBSs - Data models and the relational data model - A taxonomy of intelligent database systems - Guidelines for using intelligent database systems.<br>**Practical Component:**<br>(a) Install the LAMP<br>(b) Configure and setup the Connection between back end & Front End. | **08** |
| **II** | **Semantic Data Models**<br>Nested and semantic data models – Introduction - The nested relational model - Semantic models - Hyper-semantic data models - Object-oriented approaches to semantic data modeling – Objectoriented database systems - Basic concepts of a core object-oriented data model - Comparison with other data models - Query languages and query processing - Operational aspects – Systems – The ODMG standard - The object-relational data model - Java and databases – Conclusions – Active database systems - Basic concepts – Issues – Architectures - Research relational prototypes—the Starburst Rule System - Commercial relational approaches.<br>**Practical Component:**<br>(a) Design & create the DB user in database.<br>(b) Using SQL - create sample DB for Language –DDL, DML and DCL.<br>(c) Create sample java/PHP pages with database access. | **08** |
| **III** | **Knowledge-Based Systems- AI Ccontext**<br>Characteristics and classification of the knowledge-based systems – Introduction - The resolution principle - Inference by inheritance – Conclusion - Deductive database systems - Basic concepts - DATALOG language - Deductive database systems and logic programming systems—differences - Architectural approaches - Research prototypes - Updates in deductive databases - Integration of deductive database and object database technologies - Constraint databases - Conclusions.<br>**Practical Component:**<br>1. Working on basic commands on datalog<br>2. Practice on projection and Selection in datalog<br>**3.** Write a program that uses + and - from racket/base as external queries using DATALOG language | **08** |

| | | |
|---|---|---|
| **IV** | **Advanced Knowledge-Based Systems**<br>Introduction - Architectural solutions - The 'general bridge' solution - Extending a KBS with components proper to a DBMS - The 'tight coupling' approach – Conclusion - Advanced solutions: Introduction - A 'knowledge level' approach to the interaction with an IAS- TELOS - a language for implementing very large 'integral approach' systems- The CYC project - Other projects based on a 'conceptual representation' approach - Lexical approaches to the construction of large KBs.<br>**Practical Component:**<br>Implement the techniques to manage knowledge-based systems. | **08** |
| **V** | **Applications in IDBS**<br>Introduction - Temporal databases - Basic concepts - Temporal data models - Temporal query languages – Ontologies -Ontology theoretical foundations - Environments for building ontologies - Structured, semi-structured and unstructured data - Multimedia database - Semi-structured data - Mediators – Motivation – Architecture - Application of mediators to heterogeneous systems –Proposals - Multi-Agents systems - Main issues in designing a multi-agent system - Open problems.<br>Internet indexing and retrieval - Basic indexing methods - Search engines or meta-searchers – Internet spiders - Data mining - Data mining tasks - Data mining tools - Medical and legal information systems - Medical information systems - Legal information systems – Conclusions.<br>**Practical Component:**<br>1. Implement the temporal databases.<br>2. Design and develop a project using medical information system. | **08** |

**Text Books:**


1. Elisa Bertino, Barbara Catania, GianPieroZarri, “Intelligent Database Systems”,Collection ACM Press.
2. Ngoc ThanhNguyen, RadoslawKatarzyniak,andShyi-MingChen (Eds.), "Advances in Intelligent Information andDatabase Systems ", Springer, 2010.

| KCS 502 | COMPILER DESIGN | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Acquire knowledge of different phases and passes of the compiler and able to use the compiler tools like LEX, YACC, etc. Students will also be able to design different types of compiler tools to meet the requirements of the realistic constraints of compilers. | $K_3$, $K_6$ |
| CO 2 | Understand the parser and its types i.e. Top-Down and Bottom-up parsers and construction of LL, SLR, CLR, and LALR parsing table. | $K_2$, $K_6$ |
| CO 3 | Implement the compiler using syntax-directed translation method and get knowledge about the synthesized and inherited attributes. | $K_4$, $K_5$ |
| CO 4 | Acquire knowledge about run time data structure like symbol table organization and different techniques used in that. | $K_2$, $K_3$ |
| CO 5 | Understand the target machine's run time environment, its instruction set for code generation and techniques used for code optimization. | $K_2$, $K_4$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction to Compiler**: Phases and passes, Bootstrapping, Finite state machines and regular expressions and their applications to lexical analysis, Optimization of DFA-Based Pattern Matchers implementation of lexical analyzers, lexical-analyzer generator, LEX compiler, Formal grammars and their application to syntax analysis, BNF notation, ambiguity, YACC. The syntactic specification of programming languages: Context free grammars, derivation and parse trees, capabilities of CFG. | **08** |
| **II** | **Basic Parsing Techniques:** Parsers, Shift reduce parsing, operator precedence parsing, top down parsing, predictive parsers Automatic Construction of efficient Parsers: LR parsers, the canonical Collection of LR(0) items, constructing SLR parsing tables, constructing Canonical LR parsing tables, Constructing LALR parsing tables, using ambiguous grammars, an automatic parser generator, implementation of LR parsing tables. | **08** |
| **III** | **Syntax-directed Translation:** Syntax-directed Translation schemes, Implementation of Syntax-directed Translators, Intermediate code, postfix notation, Parse trees & syntax trees, three address code, quadruple & triples, translation of assignment statements, Boolean expressions, statements that alter the flow of control, postfix translation, translation with a top down parser. More about translation: Array references in arithmetic expressions, procedures call, declarations and case statements. | **08** |
| **IV** | **Symbol Tables**: Data structure for symbols tables, representing scope information. Run-Time Administration: Implementation of simple stack allocation scheme, storage allocation in block structured language. Error Detection & Recovery: Lexical Phase errors, syntactic phase errors semantic errors. | **08** |
| **V** | **Code Generation:** Design Issues, the Target Language. Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, Code Generator. Code optimization: Machine-Independent Optimizations, Loop optimization, DAG representation of basic blocks, value numbers and algebraic laws, Global Data-Flow analysis. | **08** |

**Text books:**


1. K. Muneeswaran,CompilerDesign,FirstEdition,Oxford University Press.
2. J.P. Bennet, "Introduction to Compiler Techniques", Second Edition, Tata McGraw-Hill,2003.
3. HenkAlblas and Albert Nymeyer, "Practice and Principles of Compiler Building with C", PHI, 2001.
4. Aho, Sethi& Ullman, "Compilers: Principles, Techniques and Tools", Pearson Education
5. V Raghvan, " Principles of Compiler Design", TMH
6. Kenneth Louden," Compiler Construction", Cengage Learning.
7. Charles Fischer and Ricard LeBlanc," Crafting a Compiler with C", Pearson Education

| KCS 551 | DATABASE MANAGEMENT SYSTEMS LAB | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Understand and apply oracle 11 g products for creating tables, views, indexes, sequences and other database objects. | $K_2$, $K_4$ |
| CO 2 | Design and implement a database schema for company data base, banking data base, library information system, payroll processing system, student information system. | K3, K5, K6 |
| CO 3 | Write and execute simple and complex queries using DDL, DML, DCL and TCL | $K_4$, $K_5$ |
| CO 4 | Write and execute PL/SQL blocks, procedure functions, packages and triggers, cursors. | $K_4$, $K_5$ |
| CO 5 | Enforce entity integrity, referential integrity, key constraints, and domain constraints on database. | $K_3$, $K_4$ |

**DETAILED SYLLABUS**

1. Installing oracle/ MYSQL
2. Creating Entity-Relationship Diagram using case tools.
3. Writing SQL statements Using ORACLE /MYSQL:
   - a)Writing basic SQL SELECT statements.
   - b) Restricting and sorting data.
   - c)Displaying data from multiple tables.
   - d)Aggregating data using group function.
   - e)Manipulating data.
   - e)Creating and managing tables.
4. Normalization
5. Creating cursor
6. Creating procedure and functions
7. Creating packages and triggers
8. Design and implementation of payroll processing system
9. Design and implementation of Library Information System
10. Design and implementation of Student Information System
11. Automatic Backup of Files and Recovery of Files
12. Mini project (Design & Development of Data and Application ) for following :
    a) Inventory Control System.
    b) Material Requirement Processing.
    c) Hospital Management System.
    d) Railway Reservation System.
    e) Personal Information System.
    f) Web Based User Identification System.
    g) Timetable Management System.
    h) h) Hotel Management System

**Note: The Instructor may add/delete/modify/tune experiments, wherever he/she feels in a justified manner**
**It is also suggested that open source tools should be preferred to conduct the lab (MySQL , SQL server , Oracle ,MongoDB ,Cubrid ,MariaDBetc)**

**Database Management Systems Lab (KCS 551): Mapping with Virtual Lab**

| Name of the Lab | Name of the Experiment |
|---|---|
| Database Management Lab (KCS-551) | Data Definition Language(DDL) Statements: (Create table, Alter table, Drop table) |
| | Data Manipulation Language(DML) Statements |
| | Data Query Language(DQL) Statements: (Select statement with operations like Where clause, Order by, Logical operators, Scalar functions and Aggregate functions) |
| | Transaction Control Language(TCL) statements: (Commit(make changes permanent), Rollback (undo) |
| | Describe statement: To view the structure of the table created |

| **KAI 551** | **ARTIFICIAL INTELLIGENCE LAB** | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to** | | |
| CO 1 | Use of python to understand the concept of AI | $K_3$ |
| CO 2 | Implementation of Different AI Techniques | $K_4, K_5$ |
| CO 3 | Application of AI techniques in practical Life | $K_4$ |
| CO 4 | Understanding of Natural Language Tool Kit. | $K_2$ |
| CO 5 | Practical Application of Natural Language Tool Kit | $K_4, K_5$ |

**DETAILED SYLLABUS**

1. Write a python program to implement Breadth First Search Traversal
2. Write a python program to implement Water Jug Problem
3. Write a python program to remove punctuations from the given string
4. Write a python program to sort the sentence in alphabetical order
5. Write a program to implement Hangman game using python.
6. Write a program to implement Tic-Tac-Toe game using python.
7. Write a python program to remove stop words for a given passage from a text file using NLTK
8. Write a python program to implement stemming for a given sentence using NLTK
9. Write a python program to POS (Parts of Speech) tagging for the give sentence using NLTK
10. Write a python program to implement Lemmatization using NLTK
11. Write a python program to for Text Classification for the give sentence using NLTK

**Note: The Instructor may add/delete/modify/tune experiments**

| KCS 553 | DESIGN AND ANALYSIS OF ALGORITHM LAB | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Implement algorithm to solve problems by iterative approach. | $K_2$, $K_4$ |
| CO 2 | Implement algorithm to solve problems by divide and conquer approach | $K_3$, $K_5$ |
| CO 3 | Implement algorithm to solve problems by Greedy algorithm approach. | $K_4$, $K_5$ |
| CO 4 | Implement algorithm to solve problems by Dynamic programming, backtracking, branch and bound approach. | $K_4$, $K_5$ |
| CO 5 | Implement algorithm to solve problems by branch and bound approach. | $K_3$, $K_4$ |

**DETAILED SYLLABUS**

1. Program for Recursive Binary & Linear Search.
2. Program for Heap Sort.
3. Program for Merge Sort.
4. Program for Selection Sort.
5. Program for Insertion Sort.
6. Program for Quick Sort.
7. Knapsack Problem using Greedy Solution
8. Perform Travelling Salesman Problem
9. Find Minimum Spanning Tree using Kruskal's Algorithm
10. Implement N Queen Problem using Backtracking
11. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n> 5000 and record the time taken to sort. Plot a graph of the time taken versus non-graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide and- conquer method works along with its time complexity analysis: worst case, average case and best case.
12. Sort a given set of n integer elements using Merge Sort method and compute its time complexity. Run the program for varied values of n> 5000, and record the time taken to sort. Plot a graph of the time taken versus non-graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate how the divide and- conquer method works along with its time complexity analysis: worst case, average case and best case.
13.6. Implement , the 0/1 Knapsack problem using
   (a) Dynamic Programming method
   (b) Greedy method.
14. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
15. Find Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm. Use Union-Find algorithms in your program.
16. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
17. Write programs to (a) Implement All-Pairs Shortest Paths problem using Floyd's algorithm.
   (b) Implement Travelling Sales Person problem using Dynamic programming.
18. Design and implement to find a subset of a given set S = {Sl, S2,.....,Sn} of n positive integers whose SUM is equal to a given positive integer d. For example, if S ={1, 2, 5, 6, 8} and d= 9, there are two solutions {1,2,6}and {1,8}. Display a suitable message, if the given problem instance doesn't have a solution.
19. Design and implement to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using backtracking principle.

**Note: The Instructor may add/delete/modify/tune experiments, wherever he/she feels in a justified manner It is also suggested that open source tools should be preferred to conduct the lab ( C, C++ etc)**

# KAI 601 MACHINE LEARNING TECHNIQUES

| Course Outcome ( CO) | | Bloom's Knowledge Level (KL) |
|---|---|---|
| **At the end of course , the student will be able:** | | |
| CO 1 | To understand the need for machine learning for various problem solving | $K_1$ , $K_2$ |
| CO 2 | To understand a wide variety of learning algorithms and how to evaluate models generated from data | $K_1$ , $K_3$ |
| CO 3 | To understand the latest trends in machine learning | $K_2$ , $K_3$ |
| CO 4 | To design appropriate machine learning algorithms and apply the algorithms to a real-world problems | $K_4$ , $K_6$ |
| CO 5 | To optimize the models learned and report on the expected accuracy that can be achieved by applying the models | $K_4$, $K_5$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **INTRODUCTION** – Learning, Types of Learning, Well defined learning problems, Designing a Learning System, History of ML, Introduction of Machine Learning Approaches – (Artificial Neural Network, Clustering, Reinforcement Learning, Decision Tree Learning, Bayesian networks, Support Vector Machine, Genetic Algorithm), Issues in Machine Learning and Data Science Vs Machine Learning; | **08** |
| **II** | **REGRESSION:** Linear Regression and Logistic Regression<br>**BAYESIAN LEARNING -** Bayes theorem, Concept learning, Bayes Optimal Classifier, Naïve Bayes classifier, Bayesian belief networks, EM algorithm.<br>**SUPPORT VECTOR MACHINE:** Introduction, Types of support vector kernel – (Linear kernel, polynomial kernel,andGaussiankernel), Hyperplane – (Decision surface), Properties of SVM, and Issues in SVM. | **08** |
| **III** | **DECISION TREE LEARNING** - Decision tree learning algorithm, Inductive bias, Inductive inference with decision trees, Entropy and information theory, Information gain, ID-3 Algorithm, Issues in Decision tree learning.<br>**INSTANCE-BASED LEARNING** – k-Nearest Neighbour Learning, Locally Weighted Regression, Radial basis function networks, Case-based learning. | **08** |
| **IV** | **ARTIFICIAL NEURAL NETWORKS** – Perceptron's, Multilayer perceptron, Gradient descent and the Delta rule, Multilayer networks, Derivation of Backpropagation Algorithm, Generalization, Unsupervised Learning – SOM Algorithm and its variant;<br>**DEEP LEARNING** - Introduction,concept of convolutional neural network , Types of layers – (Convolutional Layers , Activation function , pooling , fully connected) , Concept of Convolution (1D and 2D) layers, Training of network, Case study of CNN for eg on Diabetic Retinopathy, Building a smart speaker, Self-deriving car etc. | **08** |
| **V** | **REINFORCEMENT LEARNING**–Introduction to Reinforcement Learning , Learning Task,Example of Reinforcement Learning in Practice, Learning Models for Reinforcement – (Markov Decision process , Q Learning - Q Learning function, Q Learning Algorithm ), Application of Reinforcement Learning,Introduction to Deep Q Learning.<br>**GENETIC ALGORITHMS:** Introduction, Components, GA cycle of reproduction, Crossover, Mutation, Genetic Programming, Models of Evolution and Learning, Applications. | **08** |

**Text books:**


1. Tom M. Mitchell, ―Machine Learning, McGraw-Hill Education (India) Private Limited, 2013.
2. EthemAlpaydin, ―Introduction to Machine Learning (Adaptive Computation and Machine Learning), MIT Press
3. Stephen Marsland, ―Machine Learning: An Algorithmic Perspective, CRC Press, 2009.

4. Bishop, C., Pattern Recognition and Machine Learning. Berlin: Springer-Verlag.
5. M. Gopal, "Applied Machine Learning", McGraw Hill Education

| KCS 602 | WEB TECHNOLOGY | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to** | | |
| CO 1 | Explain web development Strategies and Protocols governing Web. | $K_1$, $K_2$ |
| CO 2 | Develop Java programs for window/web-based applications. | $K_2$, $K_3$ |
| CO 3 | Design web pages using HTML, XML, CSS and JavaScript. | $K_2$, $K_3$ |
| CO 4 | Creation of client-server environment using socket programming | $K_1$, $K_2$, |
| CO 5 | Building enterprise level applications and manipulate web databases using JDBC | $K_3$, $K_4$ |
| CO6 | Design interactive web applications using Servlets and JSP | $K_2$, $K_3$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction:** Introduction and Web Development Strategies, History of Web and Internet, Protocols Governing Web, Writing Web Projects, Connecting to Internet, Introduction to Internet services and tools, Introduction to client-server computing. Core Java: Introduction, Operator, Data type, Variable, Arrays, Methods & Classes, Inheritance, Package and Interface, Exception Handling, Multithread programming, I/O, Java Applet, String handling, Event handling, Introduction to AWT, AWT controls, Layout managers | **08** |
| **II** | **Web Page Designing:** HTML: List, Table, Images, Frames, forms, CSS, Document type definition, XML: DTD, XML schemes, Object Models, presenting and using XML, Using XML Processors: DOM and SAX, Dynamic HTML | **08** |
| **III** | **Scripting:** Java script: Introduction, documents, forms, statements, functions, objects; introduction to AJAX, **Networking :** Internet Addressing, InetAddress, Factory Methods, Instance Methods, TCP/IP Client Sockets, URL, URL Connection, TCP/IP Server Sockets, Datagram. | **08** |
| **IV** | **Enterprise Java Bean:** Preparing a Class to be a JavaBeans, Creating a JavaBeans, JavaBeans Properties, Types of beans, Stateful Session bean, Stateless Session bean, Entity bean<br>**Java Database Connectivity (JDBC):** Merging Data from Multiple Tables: Joining, Manipulating, Databases with JDBC, Prepared Statements, Transaction Processing, Stored Procedures. | **08** |
| **V** | **Servlets:** Servlet Overview and Architecture, Interface Servlet and the Servlet Life Cycle, HandlingHTTP get Requests, Handling HTTP post Requests, Redirecting Requests to Other Resources, Session Tracking, Cookies, Session Tracking with Http Session<br>**Java Server Pages (JSP):** Introduction, Java Server Pages Overview, A First Java Server Page Example, Implicit Objects, Scripting, Standard Actions, Directives, Custom Tag Libraries.. | **08** |

**Text books:**

1. Burdman, Jessica, "Collaborative Web Development" Addison Wesley
2. Xavier, C, " Web Technology and Design" , New Age International
3. Ivan Bayross," HTML, DHTML, Java Script, Perl & CGI", BPB Publication
4. Bhave, "Programming with Java", Pearson Education
5. Herbert Schieldt, "The Complete Reference:Java", TMH.
6. Hans Bergsten, "Java Server Pages", SPD O'Reilly
7. Margaret Levine Young, "The Complete Reference Internet", TMH

8. Naughton, Schildt, "The Complete Reference JAVA2", TMH
9. Balagurusamy E, "Programming in JAVA", TMH

| KCS 603 | COMPUTER NETWORKS | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| | **At the end of course , the student will be able to** | |
| CO1 | Explain basic concepts, OSI reference model, services and role of each layer of OSI model and TCP/IP, networks devices and transmission media, Analog and digital data transmission | $K_1$,$K_2$ |
| CO2 | Apply channel allocation, framing, error and flow control techniques. | $K_3$ |
| CO3 | Describe the functions of Network Layer i.e. Logical addressing, subnetting& Routing Mechanism. | $K_2$,$K_3$ |
| CO4 | Explain the different Transport Layer function i.e. Port addressing, Connection Management, Error control and Flow control mechanism. | $K_2$,$K_3$ |
| CO5 | Explain the functions offered by session and presentation layer and their Implementation. | $K_2$,$K_3$ |
| CO6 | Explain the different protocols used at application layer i.e. HTTP, SNMP, SMTP, FTP, TELNET and VPN. | $K_2$ |
| | **DETAILED SYLLABUS** | **3-0-0** |

| Unit | Topic | Proposed Lecture |
|---|---|---|
| I | **Introductory Concepts**: Goals and applications of networks, Categories of networks, Organization of the Internet, ISP, Network structure and architecture (layering principles, services, protocols and standards), The OSI reference model, TCP/IP protocol suite, Network devices and components.<br>**Physical Layer:**<br>Network topology design, Types of connections, Transmission media, Signal transmission and encoding, Network performance and transmission impairments, Switching techniques and multiplexing. | **08** |
| II | **Link layer:** Framing, Error Detection and Correction, Flow control (Elementary Data Link Protocols, Sliding Window protocols).<br>Medium Access Control and Local Area Networks: Channel allocation, Multiple access protocols, LAN standards, Link layer switches & bridges (learning bridge and spanning tree algorithms). | **08** |
| III | **Network Layer:** Point-to-point networks, Logical addressing, Basic internetworking (IP, CIDR, ARP, RARP, DHCP, ICMP), Routing, forwarding and delivery, Static and dynamic routing, Routing algorithms and protocols, Congestion control algorithms, IPv6. | **08** |
| IV | **Transport Layer:** Process-to-process delivery, Transport layer protocols (UDP and TCP), Multiplexing, Connection management, Flow control and retransmission, Window management, TCP Congestion control, Quality of service. | **08** |
| V | **Application Layer:** Domain Name System, World Wide Web and Hyper Text Transfer Protocol, Electronic mail, File Transfer Protocol, Remote login, Network management, Data compression, Cryptography – basic concepts. | **08** |

**Text books:**

**Text books and References:**

1. Behrouz Forouzan, "Data Communication and Networking", McGraw Hill
2. Andrew Tanenbaum "Computer Networks", Prentice Hall.
3. William Stallings, "Data and Computer Communication", Pearson.
4. Kurose and Ross, "Computer Networking- A Top-Down Approach", Pearson.
5. Peterson and Davie, "Computer Networks: A Systems Approach", Morgan Kaufmann
6. W. A. Shay, "Understanding Communications and Networks", Cengage Learning.
7. D. Comer, "Computer Networks and Internets", Pearson.
8. Behrouz Forouzan, "TCP/IP Protocol Suite", McGraw Hill.

| KAI 061 | CYBER FORENSIC ANALYTICS | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Outline the Cyber crime and its types. | $K_1$, $K_2$ |
| CO 2 | Explore the Cyber Forensics Techniques | $K_1$, $K_2$ |
| CO 3 | Use the Cyber Investigation Techniques | $K_3$, $K_4$ |
| CO 4 | Explore the Cyber Evidence Management Techniques | $K_3$, $K_4$ |
| CO 5 | Outline the Cyber Laws in India | $K_1$, $K_2$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Cyber Crime:**<br>Cyber Space – Cyber Crime – Criminal Behaviour – Jurisdictional Concerns - Jurisprudential Inconsistency – eCash Security – Prepaid Cards – Stored Values Cards – Mobile Payments – Internet Payment Services -Cyber stalking - Cyber extortion – Cyber terrorism - Cyber warfare –Cyber weapons -ATM frauds –<br>Phreaking – Internet Gambling<br>**Practical Component:**<br>1. Key logger<br>2. Email Fraud | **08** |
| **II** | **Cyber Forensics:**<br>Digital device – Hard disk –Disk characteristics - Disk imaging - Data Carving – Techniques – commercial piracy - soft lifting – Steganography – Network components - Port scans - Wireshark - pcap analysis - Trojans and Backdoors – Botnets - DoS – DDoS Attacks - Honey Pots – Malware – Virus and Worms<br>**Practical Component:**<br>1. Pcab file Analysis – Case Study<br>2. Network Port Scan – Forensics | **08** |
| **III** | **Cyber Investigation**<br>Concepts of Investigation - cyber investigation, Network Investigation - Investigating audit logs -Investigating Web attacks - Investigating Computer Intrusions - Profiling – Cyber Criminal profiling – Stylometric Techniques – Warranted searches – Warrantless searches – Undercover Techniques<br>**Practical Component:**<br>1. Investigating Audit Logs<br>2. Investigating Web attacks | **08** |
| **IV** | **Evidence Management:**<br>Evidence – Digital Evidence - Types – physical evidence – Real evidence – Circumstantial evidence –network evidence - Evidence collection – Evidence Analysis - Contextual Information –Evidence Management – pre search activities – On Scene activities – Report Preparations<br>**Practical Component:**<br>1. Digital Evidence Analysis<br>2. Network Analysis | **08** |
| **V** | **Cyber Laws and Authorities** | |

| | | |
|---|---|---|
| | Information Technology Act 2000 – Digital signature - Electronic Governance - Secure electronic records<br>- Regulation of certifying authorities – CERNTin - Electronic signature certificates - Penalties compensation - Future Trends and Emerging Concerns<br>**Practical Component:**<br>1. Digital Signature | **08** |

**Text Books:**


1. Marjie T. Britz, "Computer Forensics and Cyber Crime", Pearson, 2013.
2. Garima Tiwari, "Understanding Laws– Cyber Laws And Cyber Crimes", Lexis Nexis, 2014.
3. Chuck Easttom, Jeff Taylor, "Computer Crime, Investigation, and the Law", Course Technology, 2018.
4. Eoghan Casey, "Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet", Eoghan Casey, 2018.


| **KDS 061** | **IMAGE ANALYTICS** | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Infer the basics and fundamentals of digital image processing and Apply the various techniques for intensity transformations functions. Implement Color image Smoothing and Sharpening. | $K_1$, $K_2$ |
| CO 2 | Illustrate Morphological operation and Apply Some Basic Morphological Algorithms. | $K_2$, $K_3$ |
| CO 3 | Apply image segmentation techniques such as Optimum Global Thresholding using Otsu's Method, Active Contours: Snakes and Level Sets for various real-time applications. | $K_3$, $K_4$ |
| CO 4 | Analysis various Feature Extraction methods and Implement for various real-time applications. | $K_3$, $K_4$ |
| CO 5 | Apply and Analysis various Image Pattern Classification methods such as Minimum-Distance Classification, Optimum (Bayes) Statistical Classification, and Deep Convolutional Neural Network. | $K_3$, $K_4$ |
| **DETAILED SYLLABUS** | | **3-0-0** |
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Fundamentals:**<br>Introduction – Fundamental steps in Image Processing Systems – Image Acquisition – Sampling and Quantization – Pixel Relationships – Mathematical Tools Used in Digital Image Processing. Some Basic Intensity Transformation Functions: Image Negatives, Log Transformations, Power-Law Transformations - Histogram Processing. Color Fundamentals - Fundamentals of Spatial Filtering - Smoothing Spatial Filters - Sharpening Spatial Filters.<br>**Practical Component:** Use Python/ MATLAB<br>1. Apply various intensity transformations functions.<br>2. Computing and plotting image histograms and use standard image processing toolbox Spatial filters.<br>3. Implement color image Smoothing and Sharpening. | **08** |
| **II** | **Morphological Image Processing:**<br>Morphological Image Processing: Fundamentals - Erosion and Dilation - Opening and Closing – Hit or Miss Transform - Some Basic Morphological Algorithms – Morphological Reconstruction – Grayscale Morphology<br>**Practical Component:** Use Python/ MATLAB<br>1. Implement Morphological operations.<br>2. Implement Morphological Reconstruction.<br>3. Implement Grayscale Morphology. | **08** |

| | | |
|---|---|---|
| **III** | **Image Segmentation**<br>Introduction - Point, Line, and Edge Detection – Thresholding: Foundation, Basic Global thresholding, Optimum Global Thresholding using Otsu's Method, Multiple Thresholds, Variable Thresholding –Segmentation by Region Growing and by Region Splitting and Merging – Image Segmentation: Active Contours: Snakes and Level Sets.<br>**Practical Component:** Use Python/ MATLAB<br>1. Implement Optimum Global Thresholding using Otsu's Method.<br>2. Implement Image segmentation by Region Growing, Splitting and Merging<br>3. Implement Image Segmentation by Active Contours using anyone method Snakes and Level Sets. | **08** |
| **IV** | **Feature Extraction**<br>Background - Representation – Boundary Preprocessing – Boundary Feature Descriptors: Some Basic Boundary Descriptors, Shape Numbers, Fourier Descriptors, Statistical Moments - Regional Feature Descriptors: Some Basic Descriptors, Topological and Texture Descriptors, Moment Invariants – Principal Components as Feature Descriptors – Whole-image Features Object – Scale-Invariant Feature Transform (SIFT).<br>**Practical Component:** Use Python/ MATLAB<br>1. Implement Boundary Feature Descriptors<br>2. Implement Topological and Texture Descriptors<br>3. Implement Scale-Invariant Feature Transform (SIFT) | **08** |
| **V** | **Image Pattern Classification**<br>Background -Patterns and Pattern Classes – Pattern Classification by Prototype Matching: Minimum-Distance Classifier, Using Correlation for 2-D prototype matching, Matching SIFT Features, Matching Structural Prototypes - Optimum (Bayes) Statistical Classifiers - Neural Networks and Deep Learning: Background - The Perceptron - Multilayer Feedforward Neural Networks - Deep Convolutional Neural Networks<br>**Practical Component:** Use Python/ MATLAB<br>1. Implement Minimum-Distance Classification Algorithm.<br>2. Implement Optimum (Bayes) Statistical Classification Algorithm.<br>3. Implement Deep Convolutional Neural Network. | **08** |

**Text Books:**


1. Rafael C Gonzalez, Richard E Woods, "Digital Image Processing", 4th Edition, Pearson, 2018.
2. Kenneth R. Castleman, Digital Image Processing Pearson, 2006.
3. Anil K.Jain, "Fundamentals of Digital Image Processing", Person Education, 2003.

| KML 061 | ADVANCED MACHINE LEARNING | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Understand advanced concepts and methods of machine learning and to develop an understanding of the role of machine learning in massive scale automation. | $K_1$, $K_2$ |
| CO 2 | Apply various machine learning algorithms in a range of real-world applications. | $K_3$, $K_3$ |
| CO 3 | Integrate and apply their expertise to produce solutions for real-world problems. | $K_4$, $K_5$ |
| CO 4 | Comparative Analysis of different Machine Learning Algorithms | $K_4$ |
| CO 5 | Interpret and Analyze results with reasoning using different ML techniques. | $K_4$, $K_5$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Artificial Neural Network**<br>Introduction to ANN, Perceptron, Cost Function, Gradient Checking, multi-layer perceptron and backpropagation algorithm that is used to help learn parameters for a neural network, Random Initialization | **08** |
| **II** | **Bayesian Learning**<br>Probability theory and Bayes rule, Naive Bayes learning algorithm, Bayes nets. | **08** |
| **III** | **Decision Trees**<br>Representing concepts as decision trees, Recursive induction of decision trees, best splitting attribute: entropy and information gain. Searching for simple trees and computational complexity, Overfitting, noisy data, and pruning. | **08** |
| **IV** | **Reinforcement Learning**<br>Reinforcement earning through feedback network, function approximation. | **08** |
| **V** | **Ensemble Methods**<br>Bagging, boosting, stacking and learning with ensembles. Random Forest | **08** |

**Text Books:**


1. Tom Mitchell, Machine Learning, McGraw Hill, 1997.
2. Jeeva Jose, Introduction to Machine Learning, Khanna Book Publishing 2020.
3. Rajiv Chopra, Machine Learning, Khanna Book Publishing 2021
4. EthemApaydin, Introduction to Machine Learning, 2e. The MIT Press, 2010.
5. Kevin P. Murphy, Machine Learning: a Probabilistic Perspective, The MIT Press, 2012.

| KML 062 / KDS 053 | STREAM PROCESSING AND ANALYTICS | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to:** | | |
| CO 1 | Explain the need for stream processing | $K_1$, $K_2$ |
| CO 2 | Comprehend the architectures of stream processing. | $K_2$, $K_3$ |
| CO 3 | Explain and run Distributed Processing and Resilience Model | $K_1$, $K_2$ |
| CO 4 | Design effective streaming solutions using Structured Streaming | $K_5$, $K_6$ |
| CO 5 | Design effective streaming solutions using Spark Streaming | $K_5$, $K_6$ |

| **DETAILED SYLLABUS** | | **3-0-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Fundamentals of Stream Processing:** What Is Stream Processing? Examples of Stream Processing- Scaling<br>Up Data Processing- Distributed Stream Processing- Introducing Apache Spark.<br>Stream-Processing Model: Sources and Sinks- Immutable Streams Defined from One Another- Transformations and Aggregations- Window Aggregations - Stateless and Stateful Processing- The Effect of<br>Time.<br>**Practical Component:**<br>a. Installing and configuring Apache Spark<br>b. Installing and configuring the Scala IDE<br>c. Installing and configuring JDK | **08** |
| **II** | **Components of a Data Platform-** Architectural Models- The Use of a Batch-Processing Component in a<br>Streaming Application- Referential Streaming Architectures- Streaming Versus Batch Algorithms.<br>Apache Spark as a Stream-Processing Engine: Spark's Memory Usage- Understanding Latency-Throughput-<br>Oriented Processing- Fast Implementation of Data Analysis.<br>**Practical Component:**<br>a. Write your own Spark Streaming program, to count the number of words in text data received from<br>a data server listening on a TCP socket<br>b. Write a simple Spark Streaming program that prints a sample of the tweets it receives from Twitter every second. | **08** |
| **III** | **Spark's Distributed Processing Model:** Running Apache Spark with a Cluster Manager- Spark's Own Cluster<br>Manager - Resilience and Fault Tolerance in a Distributed System- Data Delivery Semantics- Microbatching and One-Element-at-a-Time - Bringing Microbatch and One-Record-at a- Time Closer Together- Dynamic Batch Interval- Structured Streaming Processing Model. Spark's Resilience Model: Resilient Distributed<br>Datasets in Spark - Spark Components - Spark's Fault-Tolerance Guarantees.<br>**Practical Component:**<br>a. Create Spark RDD using parallelize with spark Context Parallelize() method and using Spark shell<br>b. Write a scripts in Spark to Read all text files from a directory into a single RDD<br>c. Write a spark program to load a CSV file into Spark RDD using a Scala<br>d. Write a Spark Streaming program for adding 1 to the stream of integers in a reliable, fault tolerant manner, and then visualize them. | **08** |

| | | |
|---|---|---|
| **IV** | **Introducing Structured Streaming-** The Structured Streaming Programming Model – Structured Streaming in Action – Structured Streaming Sources – Structured Streaming Sinks - Event Time–Based Stream Processing.<br>**Practical Component:**<br>a. Develop a streaming application by- Connecting to a Stream, Preparing the Data in the Stream, Performing Operations on Streaming Dataset, creating a Query, Starting the Stream Processing and Exploring the data.<br>b. Create a Structured streaming job by Initializing Spark, acquiring streaming data from sources, declaring the operations we want to apply to the streaming data and outputting the resulting data using Sinks.<br>c. Create a small but complete Internet of Things (IoT)-inspired streaming program.<br>d. Define the schema in Structured Streaming to handle the data at different levels.<br>e. Create custom sinks to write data to systems not supported by the default implementations | **08** |
| **V** | **Introducing Spark Streaming** - The Spark Streaming Programming Model - The Spark Streaming Execution<br>Model - Spark Streaming Sources - Spark Streaming Sinks - Time-Based Stream Processing- Working with<br>Spark SQL – Checkpointing - Monitoring Spark Streaming- Performance Tuning.<br>**Practical Component:**<br>(i) Develop any Spark Streaming application and do the following :<br>a) Create a Spark Streaming Context,<br>b) Define one or several DStreams from data sources or other DStreams<br>c) Define one or more output operations to materialize the results of these | **08** |

**Text Books:**


1. Gerard Maas and Francois Garillot , "Stream Processing with Apache Spark: Mastering Structured Streaming and Spark Streaming", O'Reilly, 2019.
2. Henrique C. M. Andrade, BuğraGedik and Deepak S. Turaga, "Fundamentals of Stream Processing: Application Design, Systems, and Analytics", Cambridge University Press, 2014.
3. Bryon Ellis, "Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data", Wiley, 1st edition, 2014.
4. AninditaBasak, Krishna Venkataraman, Ryan Murphy, Manpreet Singh, "Stream Analytics with Microsoft Azure", Packt Publishing, December 2017.

| KDS 063 | SOFTWARE ENGINEERING | |
|---|---|---|
| | **Course Outcome ( CO)** | **Bloom's Knowledge Level (KL)** |
| | **At the end of course, the student will be able to** | |
| CO 1 | Explain various software characteristics and analyze different software Development Models. | $K_1$, $K_2$ |
| CO 2 | Demonstrate the contents of a SRS and apply basic software quality assurance practices to ensure that design, development meet or exceed applicable standards. | $K_1$, $K_2$ |
| CO 3 | Compare and contrast various methods for software design | $K_2$, $K_3$ |
| CO 4 | Formulate testing strategy for software systems, employ techniques such as unit testing, Test driven development and functional testing. | $K_3$ |
| CO 5 | Manage software development process independently as well as in teams and make use of Various software management tools for development, maintenance and analysis. | $K_5$ |

| **DETAILED SYLLABUS** | | **3-1-0** |
|---|---|---|
| **Unit** | **Topic** | **Proposed Lecture** |
| **I** | **Introduction:** Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Software Quality Attributes. Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models. | **08** |
| **II** | **Software Requirement Specifications (SRS):** Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modelling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model. | **08** |
| **III** | **Software Design:** Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design. Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs. | **08** |
| **IV** | **Software Testing:** Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, TopDown and Bottom-Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards. | **08** |
| **V** | **Software Maintenance and Software Project Management:** Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re- Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management. | **08** |

**Text books:**


1.RS Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
2. Pankaj Jalote, Software Engineering, Wiley
3. Rajib Mall, Fundamentals of Software Engineering, PHI Publication.
4. KK Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.
5. Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication.

6. Ian Sommerville, Software Engineering, Addison Wesley.
7. Kassem Saleh, "Software Engineering", Cengage Learning.
8. P fleeger, Software Engineering, Macmillan Publication

| KAI 651 | MACHINE LEARNING LAB | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to** | | |
| CO 1 | Understand complexity of Machine Learning algorithms and their limitations; | $K_5$, $K_6$ |
| CO 2 | Understand modern notions in data analysis-oriented computing; | $K_5$, $K_6$ |
| CO 3 | Be capable of performing experiments in Machine Learning using real-world data. | $K_5$, $K_6$ |
| CO 4 | Be capable of confidently applying common Machine Learning algorithms in practice and implementing their own; | $K_5$, $K_6$ |

**DETAILED SYLLABUS**

Implementation of following machine learning algorithms in various projects using Python:

## Lab Experiments:

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

4. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.

5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

6. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.

8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

9. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and

wrong predictions. Java/Python ML library classes can be used for this problem.

10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

**Note: The Instructor may add/delete/modify/tune experiments**

| **KCS 652** | **WEB TECHNOLOGY LAB** | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to** | | |
| CO 1 | Develop static web pages using HTML | $K_2$, $K_3$ |
| CO 2 | Develop Java programs for window/web-based applications. | $K_2$, $K_3$ |
| CO 3 | Design dynamic web pages using Javascript and XML. | $K_3$, $K_4$ |
| CO 4 | Design dynamic web page using server site programming Ex. ASP/JSP/PHP | $K_3$, $K_4$ |
| CO 5 | Design server site applications using JDDC,ODBC and section tracking API | $K_3$, $K_4$ |

**DETAILED SYLLABUS**

This lab is based on the Web Technologies. Some examples are as follows:

1. Write HTML/Java scripts to display your CV in navigator, your Institute website, Department Website and Tutorial website for specific subject
2. Write an HTML program to design an entry form of student details and send it to store at database server like SQL, Oracle or MS Access.
3. Write programs using Java script for Web Page to display browsers information.
5. Write a Java applet to display the Application Program screen i.e. calculator and other.
6. Writing program in XML for creation of DTD, which specifies set of rules. Create a style sheet in CSS/ XSL & display the document in internet explorer.
7. Program to illustrate JDBC connectivity. Program for maintaining database by sending queries. Design and implement a simple servlet book query with the help of JDBC & SQL. Create MS Access Database, Create on ODBC link, Compile & execute JAVA JDVC Socket.
8. Install TOMCAT web server and APACHE. Access the above developed static web pages for books web site, using these servers by putting the web pages developed.
9. Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a servlet for doing the following. Create a Cookie and add these four user id's and passwords to this Cookie. 2. Read the user id and passwords entered in the Login form and authenticate with the values available in the cookies.
10. Install a database (Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page.
11. Write a JSP which insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the user name and password from the database
12. Design and implement a simple shopping cart example with session tracking API.

**Note: The Instructor may add/delete/modify/tune experiments, wherever he/she feels in a justified manner**
**It is also suggested that open source tools should be preferred to conduct the lab ( Java , JSP , Bootstrap Firebug , WampServer , MongoDB, etc)**

| KCS 653 | COMPUTER NETWORKS LAB | |
|---|---|---|
| **Course Outcome ( CO)** | | **Bloom's Knowledge Level (KL)** |
| **At the end of course , the student will be able to** | | |
| CO 1 | Simulate different network topologies. | $K_3$, $K_4$ |
| CO 2 | Implement various framing methods of Data Link Layer. | $K_3$, $K_4$ |
| CO 3 | Implement various Error and flow control techniques. | $K_3$, $K_4$ |
| CO 4 | Implement network routing and addressing techniques. | $K_3$, $K_4$ |
| CO 5 | Implement transport and security mechanisms | $K_3$, $K_4$ |

**DETAILED SYLLABUS**

1. Implementation of Stop and Wait Protocol and Sliding Window Protocol.
2. Study of Socket Programming and Client – Server model
3. Write a code simulating ARP /RARP protocols.
4. Write a code simulating PING and TRACEROUTE commands
5. Create a socket for HTTP for web page upload and download.
6. Write a program to implement RPC (Remote Procedure Call)
7. Implementation of Subnetting .
8. Applications using TCP Sockets like
   a. Echo client and echo server b. Chat c. File Transfer
9. Applications using TCP and UDP Sockets like d. DNS e. SNMP f. File Transfer
10. Study of Network simulator (NS).and Simulation of Congestion Control Algorithms using NS
11. Perform a case study about the different routing algorithms to select the network path with its optimum and economical during data transfer. i. Link State routing ii. Flooding iii. Distance vector
12. To learn handling and configuration of networking hardware like RJ-45 connector, CAT-6 cable, crimping tool, etc.
13. Configuration of router, hub, switch etc. (using real devices or simulators)
14. Running and using services/commands like ping, traceroute, nslookup, arp, telnet, ftp, etc.
15. Network packet analysis using tools like Wireshark, tcpdump, etc.
16. Network simulation using tools like Cisco Packet Tracer, NetSim, OMNeT++, NS2, NS3, etc.
17. Socket programming using UDP and TCP (e.g., simple DNS, data & time client/server, echo client/server, iterative & concurrent servers)

**Note: The Instructor may add/delete/modify/tune experiments, wherever he/she feels in a justified manner**
**It is also suggested that open source tools should be preferred to conduct the lab ( C , C++ , Java , NS3, Mininet, Opnet, TCP Dump, Wireshark etc.**

.