SYLLABUS FOR THE BATCH FROM THE YEAR 2023 TO YEAR 2025

Programme Code: MCS

Programme Name: M.Sc. (Computer Science)

(Semester I-IV)

Examinations: 2023-2025

P.G. Department of Computer Science & Applications

Khalsa College, Amritsar

Programme name: M.Sc. (Computer Science)
Programme code: MCS
Programme Duration :2 years

Programme Objectives

1.	To impart sound knowledge in Computer Science and to enable students to apply the acquired skills creatively in computer and related technologies in practical scenarios.
2.	To effectively utilize knowledge of computing and mathematical principles to develop sustainable solutions to the present and the prospective computing problems.
3.	To effectively initiate, create and to communicate innovation through social, legal, ethical, and cultural issues inherent in the discipline of computing concepts and solutions to bridge the gap between computing industry experts and business leaders.
4.	To exhibit computing expertise through corporate leadership and entrepreneurship;
5.	To address in the broad areas of multi-disciplinary in nature, and to keep pace with advances in computing technology.

Programme Specific Outcomes (PSOs):

PSO-1.	Students gain knowledge in the areas like Soft Computing, Web Services, Cloud Computing, Paradigm of Programming language, Design and Analysis of Algorithms, Database Technologies, Advanced Operating System, Image Processing, Software Project Management and core computing subjects.
PSO-2.	Students understand all dimensions of the concepts of software application and projects.
PSO-3.	Students become employable according to current demand of IT Industry.
PSO-4.	Work in a collaborative manner with others in a team, contributing to the management, planning and implementation of a computer system.

M.Sc. (Computer Science)
Semester I

S. N.	Course Code	Course Name	Distribution of The Marks				Lectures Per week			Credit Distribution			Total Credit L+T+P	Page No.
			Theory	Internal Assessment	Practical	Total	L	T	P	L	T	P	L+T+P	
Discipline Specific Course(DSC)														
1	MCS-411 (Major)	Advanced Data Structure	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**4-5**
2	MCS-412 (Major)	Advanced Software Engineering	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**6-7**
3	MCS-413 (Major)	Network Design and Performance Analysis	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**8-9**
4	MCS-414 (Major)	Discrete Structures	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**10-11**
5	MCS-415 (Major)	R Programming	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**12-13**
Skill Enhancement Course(SEC)														
6	MCS-416P	Programming Laboratory-I(Based on Advanced Data Structures & R Programming)	**0**	**25**	**75**	**100**	**0**	**0**	**6**	**0**	**0**	**4**	**4**	**14**
					Total Marks	**600**						**Total Credits =**	**24**	

M.Sc. (Computer Science)
SEMESTER-I
MCS-411: Advanced Data Structures
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**
Credit Hours (Per Week):4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	To provide the foundations of the practical implementation and usage of Algorithms and Data Structures.
2.	To ensure that the student evolves into a competent programmer capable of designing and analyzing implementations of algorithms and data structures for different kinds of problems.
3.	To expose the student to the algorithm analysis techniques, to the theory of reductions.

UNIT-I

Review of algorithm analysis, Binary search trees, balanced binary search trees (red-black trees), Btrees, AVL Trees, 2-3 trees, 2-3-4 trees.
Binary heaps, heap operations, specifications, implementation and applications. Advanced heap structures.

UNIT-II

Priority queue operations, and double-ended priority queues.
Dictionaries, binomial heaps, Fibonacci heaps. Data structures for disjoint sets, tables and table operations.

UNIT-III

Strings: Introduction, Operations, Memory representation, Pattern matching algorithms-Brute force, the Boyer –Moore algorithm, the Knuth-Morris-Pratt algorithm.
Amortized analysis.

UNIT-IV

Graph algorithms: DFS, BFS, Shortest path algorithm, Spanning tree, Biconnected components. External data structures - external storage, external files, external sorting searching indexing files, external hashing.

References:

1. Alfred V. Aho, Jeffrey D. Uuman, John E. Hopcroft, "Data Structures and Algorithms" Addision Wesley, 1983.
2. Dinesh P. Mehta, I. SartajSahni, "Handbook of Data Structures and Applications", Chapman & Hall/CRC, 2004.
3. Sorenson and Trembley, "An Introduction to Data Structures with Applications, McGraw Hill,\2006 Edition.

Course Outcomes:

On Completing the course, the students will be able to:

CO	Design and analyze programming problem statements.
CO	Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
CO	Understand the necessary mathematical abstraction to solve problems.
CO	Come up with analysis of efficiency and proofs of correctness
CO	Comprehend and select algorithm design approaches in a problem specific manner.

M.Sc. (Computer Science)
SEMESTER-I
MCS-412: Advanced Software Engineering
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks:100**
Credit Hours (Per Week):4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	Understand Knowledge about advanced software engineering.
2.	Understand System development life cycle.
3.	Learn Software process methodologies.
4.	Understand the principles of object-oriented software construction.
5.	To Know about the software-development process, including requirements analysis, design, programming, testing and maintenance.
6.	Able to model object-oriented software systems.
7.	Understand how to design and plans of software solutions to problems using an object-oriented strategy.

UNIT-I

Software Project Management and Requirements

Project Management: Management Activities, Project P1 Software Project Management and Requirements Project Management: Management Activities, Project Planning, Project Scheduling, Risk Management; Software Requirements: Functional and Non-Functional Requirements, User Requirements, System Requirements, Requirements Document; Requirements Engineering Process: Feasibility Studies, Requirements Elicitation and Analysis, Requirements Validation, Requirements Management.

UNIT-II

System Models, Software Prototyping and Specifications

System Models, Software Prototyping and Specifications System models: Context, Behavioural, Data, and Object models, CASE Workbenches; Software Prototyping: Prototyping in theSoftware Process, Rapid Prototyping Techniques, User Interface Prototyping; Specifications:

Formal Specification in the Software Process, Interface Specification, Behavioural Specification.

UNIT-III

Software Design

Object Oriented Design: Objects and Object Classes, Object-Oriented Design Process, Design Evolution; Real Time Software Design: Systems Design, Real-Time Executives, Monitoring and Control Systems, Data Acquisition Systems; Design with Reuse: Component-Based Development, Application Families, Design Patterns; User Interface Design: Principles, User Interaction, Information Presentation, User Support, Interface Evaluation.

Verification and Validation and Testing

Verification and Validation (V & V): Static and Dynamic V & V, V & V Goals, V & V vs. Debugging, Software Inspections / Reviews, Clean-Room Software Development; Software Testing: Defect Testing, Integration Testing, Interface Testing, Object-Oriented Testing.

UNIT-IV

Software Re-engineering: Introduction Re-engineering, Software Reengineering and its importance, goals of reengineering, Software reengineering process model, software reengineering tools and Business process reengineering: Business processes, A BPR Model.

Reverse Engineering: Need of reverse engineering, Reverse engineering process.Tools for reverse engineering. Software Re-use and Reengineering

References:

1. Software project management, Walker Royce, Pearson Education Inc 7th year of publication.
2. Software Re-engineering, Robert S. Arnold IEEE Comp. Society.
3. Object Oriented Software Metrics, Lorenz and Kidd.
4. Object-Oriented Analysis and Design, Booch 3rd edition 2007.
5. Software Engineering, Roger S. Pressman 7th edition 2019.

Course Outcomes:

At the end of this course the student shall be able to:

CO-1.	Acquire the knowledge of software-engineering.
CO-2.	Knowledge of basic Software Engineering methods and practices.
CO-3.	Understand Software Engineering appropriate applications.
CO-4.	Understanding of different measurements of object oriented.
CO-5.	A general understanding of software process models such as the water fall and evolutionary models.

M.Sc. (Computer Science)
SEMESTER-I
MCS-413: Network Design & Performance Analysis
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**
Credit Hours (Per Week):4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	Understand that global connectivity can be achieved through computer networks.
2.	Understand the function of networks and get exposure to different existing and upcoming communication technologies.
3.	Make them aware that knowledge about hardware and software requirements of networks is essential.
4.	Understanding the requirements, planning and choosing the technology for building a network.
5.	Providing the knowledge about network design, security, documentation and managing the network.

UNIT -I

Requirements, planning, & choosing technology: System requirements, traffic sizing characteristics time & delay consideration. Traffic engineering and capacity planning: Throughput calculation traffic characteristics &source models, traditional traffic engineering, queued data & packet switched traffic modelling, designing for peaks, delay or latency Network performance modelling- Creating traffic matrix, design tools, components of design tools, types of design projects.

UNIT -II

Technology Comparisons- Generic packet switching networks characteristics, private vs. public networking, Business aspects of packet, frame and cell switching services, High speed LAN protocols comparison, Application performance needs, Throughput, burstiness, response time and delay tolerance, selecting service provider, vendor, service levels etc.

UNIT -III

Access Network Design- N/W design layers, Access N/W design, access n/w capacity, Backbone n/w design, Backbone segments, backbone capacity, topologies, Tuning the network, securing the network, Design for network security.

UNIT -IV

Documentation and network management- Documentation, network management, SNMP, RMON. Network Optimization- Network optimization theory: Goals of network optimization, measurements for network optimization, optimization tools, optimization techniques.

References:

1. James D. McCabe, Network Analysis, Architecture and Design, 2nd Edition, Morgan Kaufman Series in Networking, 2007 Edition.
2. YoueuZheng, ShakilAkhtar, Network for Computer Scientists and Engineers, Indian University, Oxford University Press, 2007 Edition.
3. Forouzan, Data Communications and Networking, Tata McGraw Hill, 2007 Edition.

Course Outcomes:

At the end of this course the student shall be able to:

CO-1.	Familiar with the concept of Network hardware and software requirements.
CO-2.	Understand the different protocols working at different layers of OSI and TCP/IP models.
CO-3.	Learn the concepts of different networking devices like router, hub, and switch.
CO-4.	Understanding the concept of traffic engineering and capacity planning.
CO-5.	Learn the concepts of switching and network performance modeling.
CO-6.	Comparing the private and the public networking.
CO-7.	Understanding the network design and network management.
CO-8.	Learning the concept of Network optimization.

M.Sc. (Computer Science)

SEMESTER-I
MCS-414: Discrete Structures
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**
Credit Hours (Per Week):4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	To construct correct direct and indirect proofs.
2.	Understand sets and perform different operations on sets.
3.	Identify fiunctions and determine their properties
4.	Apply logical reasoning to solve a variety of Problems.
5.	To use Graph Theory to solve the Problems
6.	To formulate problems and solve recurrence relations.

UNIT -I

Graph Theory: Basic terminology, Directed and undirected graphs, Eulerian chains and cycles. Hamiltonian chains and cycles Trees,Graph Representation, planar graphs, graph coloring Chromatic number Connectivity and other graphical parameter, spanning trees.

Combinatorial Mathematics: Basic counting principles Permutations and combinations Inclusion and Exclusion Principle Recurrence relations, Pigeonhole Principle, generating Function, Application.

UNIT -II

Sets and Functions : Sets relations functions operations equivalence relations, Composition of relations, Identity and Inverse relation.

Propositional Logic: logical connectives, well-formed formulas, tautologies, equivalences, Inference Theory.

UNIT -III

Monoids and Groups: Groups Semigroups and monoids Cyclic semi graphs and sub monoids, Subgroups,abelian groups, and Cosets. Congruence relations in semigroups.Morphisms.

Normal subgroups. Structure of Cyclic groups permutation groups, dihedral groups Elementary applications in coding theory.

UNIT -IV

Rings and Boolean algebra: Rings Subrings morphism of rings ideals and quotient rings. Euclidean domains Integral domains and fields Boolean Algebra direct product morphisms
Boolean sub-algebra Boolean Rings Application of Boolean algebra in logic circuits and switching functions.

References:

1. Fundamentals of Algebraic Specification I, EATCS Monographs on (Theory. Comp. Sc. Vol. 6), HartmutEhrig , Bernd MahrSpringer Verlag, Berlin 1985.
2. Mathematical Structures for Computer Science,Judith L. Gersting,W. H. Freeman & Co.,2nd Edition, New York,1987.
3. Algorithmic Graph theory,Alan Gibbons, Cambridge University Press, 1985.
4. The art of Computer Programming, Knuth, Donald Ervin, Vol. I, Fundamental Algorithms. 2nd ed. Reading, Mass, Addison Wesley 1973.
5. Kolman B. Busby R. Discrete Mathematical Structures for Computer Science, Prentice HallEnglewood Cliffs. 1987.
6. Sahni, S. Concepts in Discrete Mathematics Fridley MN., Camelot Publ. Comp., 1981.
7. Schmidt G. Strohlein T. Relations Graphs Program, EATS Monograph on Theor. Comp. Sc. Vol. 29 Berlin Spinger 1993.
8. Wheeler W. Universal Algebra for Computer Scientist EATCS Monographs on Theor.Comp. Sc. Vol. 25 Spinger-Verlag, Berlin 1991.

Course Outcomes:

The students, after the completion of the course, are expected to:

CO-1.	Identify sets, different properties of sets, set operations and set identities
CO-2.	Explain the different methods for representing the relationship between sets.
CO-3.	Evaluate Boolean functions and simplify the expressions using properties of Boolean algebra.
CO-4.	Learn the basic concepts involving functions needed in discrete structures.
CO-5.	Define and interpret the concepts of divisibility, congruence etc.
CO-6.	Express a logic sentence in terms of predicates, quantifiers and logical connectives.

M.Sc. (Computer Science)
SEMESTER-I
MCS-415: R – Programming
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**
Credit Hours (Per Week):4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	The basics of statistical computing and data analysis.
2.	How to use R for analytical programming.
3.	How to implement data structure in R.
4.	R loop functions and debugging tools.
5.	Object-oriented programming concepts in R.
6.	Data visualization in R.
7.	How to perform error handling.
8.	Writing custom R functions.

UNIT-I

Introduction to R: R character set, R words, constants, operators, precedence and associativity of the operators, R working environment as a displayer, R as a calculator, R as a data manipulator, R objects and their data types.

UNIT-II

R programming environment: Programming in R using, Sequence, Selection iteration and Case logic structures. User-defined functions in R, Recursion, Basic data structures in R (vector, factor, list, data frame, matrix, array).

UNIT-III

R factors: Understanding factors, Modifying factors, Factors in Data frames. Data frames in R: Creating data frame, Operations on data frames, Accessing data frames, Creating data frames, from various sources

UNIT-IV

Data visualization in R: Plot() function and line plot, pie chart / 3D pie chart, Scatter plot, Box plot.

Stringr package: Important functions in stringr, Regular expressions.

Dplyr package: Load data into dataframe, Viewing the data, selecting columns, selecting rows, Reordering the rows, Pipe operator, Group operations.

References:

1. The art of Programming through R by Anil BikashChowdhury
2. The art of R programming by Norman Matloff, , No Starch Press, Sanfrancisco.
3. Statistical Programming in R by Srinivasa,Siddesh,Shetty and Sowmya, Oxford Higher Education

Course Outcomes:

At the end of this course the student shall be able to

1.	Explain critical R programming concepts.
2.	Demonstrate how to install and configure RStudio.
3.	Apply OOP concepts in R programming.
4.	Explain the use of data structure and loop functions.
5.	Analyse data and generate reports based on the data.
6.	Apply various concepts to write programs in R.

M.Sc. (Computer Science)
SEMESTER-I
MCS-416 P
Programming Laboratory – I
(Based on Advanced Data Structures and R Programming)
Skill Enhancement Course (SEC)

Time: 3 Hrs **Total Marks: 100**
Credit Hours (Per Week):4 **Practical Marks: 75**
Practical Internal Assessment M: 25

Course Objectives:

1.	Understand and apply linear data structures-List, Stack and Queue.
2.	Understand the graph algorithms.
3.	Learn different algorithms analysis techniques.
4.	Apply data structures and algorithms in real time applications
5.	The basics of statistical computing and data analysis.
6.	How to use R for analytical programming.
7.	How to implement data structure in R.
8.	R loop functions and debugging tools.
9.	Object-oriented programming concepts in R.

Programs based on Advanced Data Structures using C/C++ and R Programming.
Course Outcomes (COs):

At the end of this course student will:

CO-1.	Formulate, design and analyse algorithms for problem statements.
CO-2.	Implement basic data structures and sorting algorithms.
CO-3.	Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
CO-4.	Design and develop efficient algorithms for problem.
CO-5.	Explain critical R programming concepts.
CO-6.	Demonstrate how to install and configure R Studio.
CO-7.	Apply OOP concepts in R programming.
CO-8.	Explain the use of data structure and loop functions.

M.Sc. (Computer Science)
Semester II

S N	Course Code	Course Name	Distribution of The Marks				Lectures Per week			Credit Distribution			Total Credit L+T+P	Page No.
			Theory	Internal Assessment	Practical	Total	L	T	P	L	T	P		
Discipline Specific Course(DSC)														
1	MCS-421 (Major)	Theory of Computation	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**16-17**
2	MCS-422 (Major)	Image Processing	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**18-19**
3	MCS-423 (Major)	Design and Analysis of Algorithms	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**20-21**
4	MCS-424 (Major)	Cloud Computing	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**22-24**
5	MCS-425 (Major)	Artificial Intelligence	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**25-26**
Skill Enhancement Course(SEC)														
6	MCS-426P	Programming Laboratory-II	**0**	**25**	**75**	**100**	**0**	**0**	**6**	**0**	**0**	**4**	**4**	**27**
						Total Marks600						**Total Credits =**	**24**	

M.Sc. (Computer Science)
SEMESTER-II
MCS-421: Theory of Computation
Discipline Specific Course (DSC)

Time: 3 Hrs
Credit Hours (per week):4 **Total Marks: 100**
Theory Marks: 75
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	To make students understand the basic mathematical model of computation
2.	To understand the relation between these formal languages, grammars, and machines.
3.	To understand the complexity or difficulty level of problems when solved using these machines.
4.	To compare the complexity of problems.
5.	Introduction to Recursively Enumerable Languages and Computable Functions

UNIT –I

Finite Automata Deterministic Finite Automata, non deterministic finite Automata, Transition System, Equivalence of NFA and DFA, Finite Automata with Null-moves. 2-Way Finite Automata, crossing sequences, Moore and Mealy Machine, Inter Conversion of Moore and Mealy Machine, Application of finite automata, Chomsky Hierarchy of Languages, Recursive and recursively-enumerable languages sets, Language and their relation, Languages and automata.

UNIT –II

Regular Expression and Languages: Regular expression, Equivalence of finite Automata and Regular expressions, Conversion between regular expressions and finite automata, Application of Regular Expressions.

Regular Languages and Regular sets, Pumping lemma for regular sets, Applications of pumping lemma. Closure properties of regular language.

UNIT –III

Context free Grammar and Languages: Context free Grammars, Derivation Trees, Leftmost and rightmost derivations, Ambiguity, Properties of Context free Languages- Normal forms for context free grammars (Chomsky Normal Form, Griebach Normal Form, The Kuroda Normal Form)

Pushdown Automata: Deterministic Push down Automata; Equivalence of Push Down Automata and Context free grammar. Linear Bounded Automata (LBA): Power of LBA, Closure Properties.

UNIT –IV

Turing Machine (TM): One Tape, multi tape, the notions of time and space complexity in terms of T.M. Construction of simple problems. Computational complexity.

Recursive And Recursively Enumerable Languages (REF)-: Properties of recursive and recursively enumerable languages, Universal Turing machine, The halting problem, Undecidable problem about the TMs . Context sensitive language and linear bounded Automata (LBA), Post's correspondence problem (PCP) ,undecidability of PCP

References:

1. J.E. Hopcroft, R. Motwani and J.D. Ullamn, "Introduction to Automata Theory, Languages and Computation", Pearson Education Asia, 2nd Edition.
2. John C. Martin, "Introduction to Languages and the Theory of Computation", Tata McGraw Hill Publication Company Limited, 3rd Edition.
3. K.L.P Mishra and N. Chandrasekaran," Theory of Computer Science",Prenctice-Hall of IndiaPvt.Ltd. 3rd Edition"
4. Daniel I.A. Cohen, "Introduction to Computer Theory", Wiley, Second edition.
5. B. M. Moret, "The Theory of Computation", Pearson Education Asia.
6. Formal Languages and automata theory by C K Nagpal.

Course Outcomes:

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

CO-1.	Learn about deterministic and non-deterministic finite state machines along with their designing, conversion from finite machines to regular grammar, conversion from regular grammar to finite automata, and their applications.
CO-2.	Comprehend the working and applications of pumping lemma.
CO-3.	Gain insight into the concept of Context free grammar and normal forms of context free grammar.
CO-4.	Design the pushdown automata.
CO-5.	Design the Turing Machines and will get knowledge about the notions of time and space complexity in terms of Turing Machine.
CO-6.	Apply the theoretical concepts to the practice of program design with regular expressions, parsing, and complexity analysis

M.Sc. (Computer Science)
SEMESTER-II
MCS-422: **Image Processing**
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**
Credit Hours (Per Week): 4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

The objective of this course is :

1.	To learn the Fundamental concepts of a digital image processing system.
2.	To Learn the image compression procedures.
3.	To study Compression techniques and morphological concepts.
4.	To learn Colour models and various applications of image processing
5.	To expose students to current applications in the field of digital image processing.

UNIT –I

Introduction to Image Processing: Digital Image representation, Sampling & Quantization, Steps in image Processing, Image acquisition, color image representation.

Image Transformation & Filtering: Intensity transform functions, histogram processing, Spatial filtering, Fourier transforms and its properties, frequency domain filters, colour models, Pseudo colouring, color transforms, Basics of Wavelet Transforms

UNIT –II

Image Restoration: Image degradation and restoration process, Noise Models, Noise Filters, degradation function, Inverse Filtering, Homomorphic Filtering
Image Compression: Coding redundancy, Interpixel redundancy, Psychovisual redundancy, Huffman Coding, Arithmetic coding, Lossy compression techniques, JPEG Compression.

UNIT -III

Hardware architecture for image processing: Distributed processing of image data, role of arrayprocessing, standard image processor chips (as example).

UNIT -IV

Techniques of Colour Image Processing: Colour image signal representation, colour system transformations, extension of processing techniques to colour domain.
Applications of Image Processing: Picture data archival, machine vision, medical image processing.

References:

1. Pratt, W.K. Digital Image Processing, John Wiley, N.Y./.4th Edition (2 March 2007)
2. Jain, A.K., Fundamentals of Digital Image Processing, Englewood Cliffs, Prentice Hall, Pearson (23 September 1988).
3. Chris Soloman, Stuart Gibson, Fundamentals of Digital Image Processing: A PracticalApproach using MatLab, John Wiley and Sons, 2007.
4. Digital Image Processing by Gonzalez & Wood, Addison Wesley, 2000.
5. Solomon Chris , Toby Breckon Fundamentals of Digital Image Processing :A Practical Approach with Examples in Matlab. 1st edition (January 4, 2011)

Course Outcomes:

At the end of this course the student shall be able to:

CO-1	Study the fundamental concepts of image processing
CO-2	Study the practical applications of image processing
CO-3	Contrast image segmentation and representation
CO-4	Analyse images in the frequency domain using various transforms.
CO-5	Review the fundamental concepts of a digital image processing system.

M.Sc. (Computer Science)

SEMESTER-II

MCS-423

Design & Analysis of Algorithms

Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credit Hours (Per Week): 4 **Theory Marks: 75**

Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	This course aims to introduce the classic algorithms in various domains, and techniques for designing efficient algorithms.
2.	Algorithm design and analysis provide the theoretical backbone of computer science and are a must in the daily work of the successful programmer.
3.	The goal of this course is to provide a solid background in the design and analysis of the major classes of algorithms. At the end of the course, students will be able to develop their own versions for a given computational task and to compare and contrast their performance.
4.	The goal is to provide students with a wide variety of computational problems, and to provide a thorough knowledge of the most common algorithms and data structures.

UNIT -I

Algorithm And Analysis: Concept of Algorithm, Algorithm Specification, Performance Analysis (Time and space complexities), Best, Average and Worse case performance of algorithms, Asymptotic Notations,

Divide and Conquer: General Method, Binary Search, Merge Sort, Quick Sort, Selection Sort and analysis of these Problems.

UNIT –II

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum Cost Spanning Trees (Prim's Algorithm, Kruskal's Algorithm) and Single-Source Shortest Path and its analysis.

String Processing: The Boyer –Moore algorithm, Robin Karp Algorithm, Knuth-Morris-Pratt algorithm.

UNIT –III

Dynamic Programming: General Single Method, Multistage Graphs, All Pairs ShortestPaths, Single-Source Shortest Paths, Optimal Binary Search Trees.
Backtracking: General Method, 8-Queens Problem, Graph Coloring and HamiltonianCycles.
Search and Traversal Technique: Techniques for Binary Trees, Techniques for Graphs.

Unit –IV

Branch and Bound: Least cost search, LC branch and Bound, Bounding, FIFO Branch and Bound, 0/1 Knapsack and Travelling Salesman Problem.
Introduction to complexity theory: NP- Hard and NP- Complete Problem, Basic Concepts, Cook's Theorem, examples of NP- Hard problems, Approximation Algorithms.

References:

1. V. Aho, J.E. Hopcroft, J.D. Ullman, Design and Analysis of Algorithms, AddisonWesley, 1976.
2. Horowitz, S. Sahni, Fundamentals of Computer Algorithms, Galgotia Publishers,1984.
3. K. Mehlhorn, Data Structures and Algorithms, Vols. 1 and 2, Springer Verlag, 1984.
4. Purdom, Jr. and C. A. Brown, The Analysis of Algorithms, Holt RinechartandWinston, 1985.
5. D. E. Kunth, The Art of Computer Programming, Vols. I and 3, Addison Wesley,1975.
6. AnanyLevitin, Introduction to the Design & Analysis of Algorithms, Addison,Wesley, 2002.

Course Outcomes:

CO-1.	To analyse the problem and identify the computing requirements appropriate for its solution.
CO-2.	Write rigorous correctness proofs for algorithms.
CO-3.	To apply mathematical foundations, algorithmic principles, and computer science theory to the modelling and design of computer- based systems in a way that demonstrates comprehension of the trade- offs involved in design choices.
CO-4.	Synthesize efficient algorithms in common engineering design situations.

M.Sc. (Computer Science)
SEMESTER-II
MCS-424: Cloud Computing
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**
Credit Hours (Per Week): 4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

.

Course Objectives:

1.	Highlight the specific privacy and information security risks that can exists using cloud computing services.
2.	Explain the various open challenges and issues of cloud computing.
3.	Understanding of the role and usage of virtualization technologies.
4.	Clarify what cloud computing is and what are the various advantages and limitations of using cloud computing.
5.	To understand the features and usage of cloud platforms by studying the existing systems.
6.	Introduce the advance concepts such as Big Data Analytics, Federated Cloud Computing.

UNIT -I

Introduction: Definition, Vision, Reference Model, Benefits, Limitations, Terminology and Types of Cloud.

Cloud Issues and Challenges: Cloud Computing issues and challenges like Elasticity, Resource Management and Scheduling, Cost Management, Big Data, Pre-reservation and Cloud bursting.

UNIT -II

Virtualization: Definition, Type of Virtualization- CPU Virtualization, Memory Virtualization, Network Virtualization, Server Virtualization, Client Virtualization, Application Virtualization, Benefits, Limitations, Virtualization and Cloud, Virtual Appliance.

UNIT -III

Cloud Computing Architecture: Service Models- IaaS, PaaS, SaaS, Deployment Models-Automation of Cloud Deployment, Self Service Features in a Cloud Deployment, Federated Cloud Deployment, Cloud Entities, Cloud Clients, Service Level Agreement (SLA) and Quality of Service (QoS) in Cloud Computing.

Programming Models in Cloud: Thread Programming, Task Programming-High Performance Computing, High Throughput Computing, Many Task Computingand Map–Reduce Programming.

UNIT -IV

Cloud Security: Infrastructure Security, Data Security, Identity and Access Management, Privacy Management, Security as a Service on Cloud.

Advance Topic in Cloud: Energy Efficiency in cloud, Market Oriented Cloud Computing, Big–Data Analytics, Federated Cloud Computing, Advantages and Challenges of Federated Cloud Computing.

References:

1. Barrie Sosinsky, Cloud Computing Bible, Wiley India Pvt. Ltd., ISBN–13: 978–8–12–652980–3, New Delhi, India, 2011.
2. Dr.Saurabh Kumar, Cloud Computing: Insights Into New–Era Infrastructure, Wiley India Pvt. Ltd, ISBN–13: 978–8–12–652883–7, New Delhi, India, 2011.
3. Fern Halper, Hurwitz, Robin Bloor, Marcia Kaufman, Cloud Computing for Dummies, Wiley India Pvt. Ltd, ISBN–13: 978–0–47–059742–2, New Delhi, India, 2011.
4. RajkumarBuyya, Christian Vecchiola and ThamaraiSelvi, Mastering Cloud Computing: Foundation and Application Programming, Tata McGraw Hill, ISBN–13: 978–1–25–902995–0, New Delhi, India, Feb 2013.
5. Tim Mather, SubraKumaraswamy, ShahedLatif, Cloud Security and Privacy, O'Reilly,ISBN–13: 978–8–18–404815–5.
6. Puttini R. and Mahmood Z., Cloud Computing: Concepts, Technology & Architecture, Service Tech press (2013) 1st ed.
7. Buyya K, R., Broberg J. and Goscinski M. A., Cloud Computing: Principles and paradigms, MIT Press (2011) 4th ed.

Course Outcomes:

At the end of this course the student shall be able to:

CO-1.	Analyze the performance, scalability, and availability of the underlying cloud technologies and software.
CO-2.	Apply and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms.
CO-3.	Create combinatorial auctions for cloud resources and design scheduling algorithms for computing clouds.

CO-4.	Design different work flows according to requirements and apply map reduce programming model.
CO-5.	Address cloud Storage systems and Cloud security, the risks involved, its impact and develop cloud application.
CO-6.	Analyze the Cloud computing setup with its vulnerabilities and applications using different architectures.

M.Sc. (Computer Science)
SEMESTER-II
MCS-425: Artificial Intelligence
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**
Credit Hours (per week):4 **Theory Marks: 75**
Theory Internal Assessment M: 25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

.

Course Objectives:

1.	Helping student understand the world of Artificial Intelligence and its applications through games, activities and multi-sensorial learning to become AI-Ready.
2.	Introducing the student to domains of AI in an age appropriate manner.
3.	Allowing the student to construct meaning of AI through interactive participation and engaging hands-on activities.
4.	Learn the methods of solving problems using Artificial Intelligence

UNIT-I

Introduction: A Neural Network, Human Brain, Models of a Neuron, Neural Networks viewed as Directed Graphs, Network Architectures, Knowledge Representation, Artificial Intelligence and Neural Networks

Learning Process: Error Correction Learning, Memory Based Learning, Hebbian Learning, Competitive, Boltzmann Learning, Credit Assignment Problem, Memory, Adaption, Statistical Nature of the Learning Process

UNIT-II

Single Layer Perceptrons: Adaptive Filtering Problem, Unconstrained Organization Techniques, Linear Least Square Filters, Least Mean Square Algorithm, Learning Curves, Learning Rate Annealing Techniques, Perceptron –Convergence Theorem, Relation Between Perceptron and Bayes Classifier for a Gaussian Environment

UNIT-III

Multilayer Perceptron: Back Propagation Algorithm XOR Problem, Heuristics, Output Representation and Decision Rule, Computer Experiment, Feature Detection

Back Propagation: Back Propagation and Differentiation, Hessian Matrix, Generalization, Cross Validation, Network Pruning Techniques, Virtues and Limitations of Back Propagation Learning, Accelerated Convergence, Supervised Learning

UNIT-IV

Self-Organization Maps (SOM): Two Basic Feature Mapping Models, Self-Organization Map, SOM Algorithm, Properties of Feature Map, Computer Simulations, Learning Vector Quantization, Adaptive Patter Classification.

References:

1. Gallant S.L., Neural Networks Learning & Expert Systems, MIT Press, 1993.
2. Haykin S., Neural Networks: A Comprehensive Foundation, Pearson Education Inc.,Second Edition, 2003.
3.Freeman J.A., Skapura D.M., Neural Network Algorithms, Applications and Programming Techniques, Addison-Wesley Publications, 1992.
4. Jacek M. Zurada, Introduction to Artificial Neural Systems, JAICO Publishing House Ed. 2006.
5. Li Min Fu, Neural Networks in Computer Intelligence, TMH 2003.
6. B. Vegnanarayana, Artificial Neural Networks, Prentice Hall of India P Ltd 2005

Course Outcomes:

At the end of this course the student shall be able to

1.	Explain the basic knowledge representation, problem solving, and learning methods of Artificial Intelligence.
2.	Assess the applicability, strengths, and weaknesses of the basic knowledge representation, problem solving, and learning methods in solving particular engineering problems.
3.	Develop intelligent systems by assembling solutions to concrete computational problems.
4.	Understand the role of knowledge representation, problem solving, and learning in intelligent-system engineering
5.	Develop an interest in the field sufficient to take more advanced subjects.

M.Sc. (Computer Science)
SEMESTER-II

MCS-426P
Programming Laboratory – II
Skill Enhancement Course (SEC)

Time: 3 Hrs **Total Marks:100**
Credit Hours (Per week):4 **Practical Marks: 75**
Practical Internal Assessment M: 25

Course Objectives:

1.	To develop proficiency in problem solving and programming.
2.	To be able to carry out the Analysis of various Algorithms for mainly Time and Space Complexity.
3.	To develop a base for advanced study in Computer Science.

Implementations based on Design & Analysis of Algorithms or Artificial Intelligence or Cloud Computing or Image Processing

Course Outcomes:

At the end of this course student will:

CO-1.	Get a good understanding of applications of Data Structures
CO-2.	Understand problems by applying appropriate algorithms.
CO-3.	Apply techniques of stacks and queues to solve problems.
CO-4.	Analyse the efficiency of various algorithms.
CO-5.	Solve a program in many ways using different techniques.

M.Sc. (Computer Science)
Semester III

SN	Course Code	Course Name	Distribution of The Marks				Lectures Per week			Credit Distribution			Total Credit L+T+P	Page No.
			Theory	Internal Assessment	Practical	Total	L	T	P	L	T	P		
Discipline Specific Course(DSC)														
1	MCS-531 (Major)	Advanced Computer Architecture	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**29-30**
2	MCS-532 (Major)	System Software	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**31-32**
3	MCS-533 (Major)	Data Mining and Warehousing	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**33-34**
4	MCS-534 (Major)	Concept of Core and Advanced Java	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**35-36**
5	MCS-535 (Major)	Advanced Operating System	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**37-38**
Skill Enhancement Course(SEC)														
6	MCS-536P	Programming Laboratory-III (Based on Advanced Java Programming)	**0**	**13**	**37**	**50**	**0**	**0**	**6**	**0**	**0**	**2**	**2**	**39**
												Total Credits =	**22**	

M.Sc. (Computer Science)
SEMESTER-III
MCS-531: Advanced Computer Architecture

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	To make students know about the Parallelism concepts in Uniprocessor systems.
2.	To give the students an elaborate idea about the different memory systems and buses.
3.	To introduce the advanced processor architectures to the students.
4.	To make the students know about the importance of multiprocessor and pipeline computers.
5.	To study about data flow computer architectures.

UNIT -I

Paradigms of Computing: Synchronous – Vector/Array, SIMD, Systolic
Asynchronous – MIMD, reduction Paradigm, Hardware taxanomy: Flynn's classification, Software Taxanomy: Kung's taxanomy, SPMD.
Parallel Computing Models: Combinational Circuits, Sorting Networks, PRAM models, Interconnection RAMs.

UNIT -II

Parallelism in Uniprocessor Systems: Trends in parallel processing, Basic Uniprocessor Architecture, Parallel Processing Mechanism.
Parallel Computer Structures: Pipeline Computers, Array Computers, Multiprocessor Systems

UNIT -III

Architectural Classification Schemes: Multiplicity of Instruction-Data Streams, Serial versus Parallel Processing, Parallelism versus Pipelining

UNIT -IV

Pipelining: An overlapped Parallelism, Principles of Linear Pipelining, Classification of Pipeline Processors, General Pipelines and Reservation Tables

References:

1. Computer Architecture and Parallel Processing, Faye A. Briggs, McGraw-Hill International, 2007 Edition
2. Computer Systems Organization & Architecture, John d. Carpinelli, Addison Wesley, 2007 Edition.

Course Outcomes: At the end of this course the student shall be able to:	
CO-1.	Demonstrate concepts of parallelism in hardware/software.
CO-2.	Understanding the parallel computing models.
CO-3.	Describe architectural features of advanced processors.
CO-4.	Interpret performance of different pipelined processors.
CO-5.	Understanding the parallel processing mechanism in uniprocessor systems.
CO-6.	Familiar with the concept of pipeline, array and multiprocessor systems.

M.Sc. (Computer Science)
SEMESTER-III
MCS-532: System Software
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

5. **Medium of Examination is English Language.**
6. **There will be five sections.**
7. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
8. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course objectives:

The objective of this course is to:

1.	Understand the components of system software and Familiarization with Assembly language.
2.	Detailed knowledge of Compilation process of a program.
3.	Knowledge of internal working of macro processor.
4.	Understanding the working of linker and loaders.

UNIT-I

Introduction to System Software: Evolution of System Software, components of system software, Translators, loaders, interpreters, compiler, assemblers.

Assemblers: Overview of assembly process, design of one pass and two pass assemblers.

UNIT-II

Macroprocessors: Macro definition and expansion, concatenation of macro parameters, generations of unique labels, conditional macro expansion, Recursive macro expansion.

UNIT-III

Compilers: Phases of compilation process, lexical analysis, parsing, storage management optimisation. Incremental compilers, cross compilers, P code compilers.

UNIT-IV

Loaders and Linkage Editors: Basic loader functions. Relocation, program linking, linkage, editors, dynamic linking bootstrap loaders.

Other System Software: Operating system, DBMS, text editors, Interactive debugging systems.

References:

1. Leland L. Beck: System Software, An introduction to system programming, AddisonWesley.
2. D.M. Dhamdhere: Introduction to System Software, Tata McGraw Hill.
3. D.M. Dhamdhere: System Software and Operating System, Tata McGraw Hill, 1992.
4. Madrich, Stuarte: Operating Systems, McGraw Hill, 1974.
5.Stern Nancy Assembler Language Programming for IBM and IBM compatible computers,John Wiley, 1991.

Course Outcomes:

At the end of this course the student shall be able to:

CO-1.	Study the architecture of a hypothetical machine, its assembly language, macro language.
CO-2.	Understand the structure and design of assemblers, linkers and loaders.
CO-3.	Understand the concepts and theory behind the implementation of high level programming languages.
CO-4.	Get familiarize with various software development tools.
CO-5.	Understand the fundamental principles in compiler design and to identify the relationships among different phases of compiler

M.Sc. (Computer Science)
SEMESTER-III
MCS-533: Data Mining and Warehousing
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	T	P
3	1	0

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	Understand the functionality of the various data mining and data warehousing component Knowledge.
2.	Understand and appreciate the strengths and limitations of various data mining and data warehousing models
3.	Apply, Create and Explain the analyzing techniques of various data
4.	Analyze and describe different methodologies used in data mining and data ware housing.
5.	Evaluate and Analyze and Compare different approaches of data ware housing and data mining with various technologies.

UNIT-I

Data Warehousing:

Concepts of Data Warehousing, Difference between operational database systems and Data warehousing, Need of a separate Data Warehouse. Multidimensional Data Model.

UNIT-II

Data Warehousing Architecture:

Steps for Design and Construction of Data-Warehouses, Three-Tier Data Warehouse Architecture, Characteristics of Data Warehousing Data, Data Marts, Types of OLAP Servers: ROLAP, MOLAP, HOLAP; Difference between Online Transaction Processing and Online Analytical Processing

UNIT-III

Data Warehouse Implementation:

Efficient Computation of Data Cubes, Indexing OLAP Data, Efficient Processing of OLAP Queries, Metadata Repository, Data Warehouse Back-End Tools and Utilities

UNIT-IV

Data Mining

Basic Concepts; Data Mining Techniques: Predictive Modeling, Database Segmentation, Link Analysis, Deviation Detection in details.

Data Mining Query Languages, Applications and Trends in Data Mining.

References:

1. Han, Kamber*"Data Mining: Concepts and Techniques"* Morgan Kaufmann.

2. RomezElmasri, ShamkantB.Navathe, *"Fundamentals of Database Systems"* Pearson Education.

3. Silberschatz, Korth, Sudershan*"Database System Concepts"* 4th Ed. McGraw Hill

4. Connolly &Begg*"Database Systems – A Practical Approach to Design, Implementation and Management*", 3rd Ed., Pearson Education.

Course Outcomes:

On Completing the course, the students will be able to:

CO-1.	Become familiarize with mathematical foundations of data mining tools.
CO-2.	Understand and implement classical models and algorithms in data warehouses and data mining.
CO-3.	Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
CO-4.	Master data mining techniques in various applications like social, scientific and environmental context.
CO-5.	Develop skills in selecting the appropriate data mining algorithm for solving practical problems.

M.Sc. (Computer Science)
SEMESTER-III
MCS-534: Concept of Core and Advanced Java
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	The objective is to equip the students with the core and advanced feature of contemporary Java which would enable them to handle complex programs relating to managing data and processes over the network.
2.	To understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
3.	To understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries etc.
4.	The major objective of this course is to provide a sound foundation to the students on the concepts, precepts and practices, in a field that is of immense concern to the industry and business.

UNIT-I

Java Fundamentals: Features, Objects Oriented Basis, Java Virtual Machine Character Set, Operators, Data Types, Control Structures.

UNIT-II

Classes & Objects: Classes, Inheritance, Polymorphism, Packages & Interfaces, Stream IO Classes, Exception Handling.

UNIT-III

Multithreading: Java Thread model, Thread Priorities, Synchronization, Interthread communication, Suspending, resuming & stopping thread.

UNIT-IV

Applet: Applet basics, Applet architecture, Display, Repaint, Parameter Passing.
Telnet, FTP, Web Server and their implementation in Java.

References:

1. Complete Reference: Java, Herbert Schidit & Pattrick Naugthon TMH Publications.
2. The java Tutorial Continued by Compione, Walrath, huml Sun Java Tutorial Team, Addison Wessley.
3. Java Black Book Steven Holzner OT Dreamtech Press.
4. Beginning Programming with Java For Dummies Barry A. Burd

Course Outcomes:

On Completing the course, the students will be able to:

CO-1.	Design and implement programs in the Java programming language that make strong use of classes and objects.
CO-2.	Learn to print formatted text to the console output and read/parse console input text using a Scanner object.
CO-3.	Study logical constructs for branching and loops as well as use iterate objects when appropriate.
CO-4.	Define classes and methods. In addition, students will learn the basics of polymorphism through use of super-classes and interfaces. Finally, students will develop an understanding of the Java language class hierarchy including the cosmic Object super class.
CO-5.	Learn to create and access arrays and array lists, including those with Books Prescribed to generalized objects types.

M.Sc. (Computer Science)
SEMESTER-III
MCS-535: Advanced Operating System
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	The aim of this course is to study, learn, and understand the main concepts of advanced operating systems.
2.	To get a comprehensive knowledge of the architecture of distributed systems.
3.	To understand the deadlock and shared memory issues and their solutions in distributed environments.
4.	To know the security issues and protection mechanisms for distributed environments.

UNIT I

Architectures of Distributed Systems: System Architecture types, issues in distributed operating systems, communication networks, communication primitives. Theoretical Foundations, inherent limitations of a distributed system, lamp ports logical clocks, vector clocks.

Casual ordering of messages: global state, cuts of a distributed computation, termination detection. Distributed Mutual Exclusion, introduction, the classification of mutual exclusion and associated algorithms a comparative performance analysis.

UNIT II

Distributed Deadlock Detection: -Introduction, deadlock handling strategies in distributed systems, issues in deadlock detection and resolution, control organizations for distributed deadlock detection, centralized and distributed deadlock detection algorithms, hierarchical deadlock detection algorithms. Agreement protocols, introduction-the system model, a classification of agreement problems, solutions to the Byzantine agreement problem, and applications of agreement algorithms.

Distributed resource management: introduction-architecture, mechanism for building distributed file systems, design issues, log structured file systems.

UNIT III

Distributed shared memory-Architecture: algorithms for implementing DSM, memory coherence and protocols, design issues. Distributed Scheduling, introduction, issues in load distributing, components of a load distributing algorithm, stability, load distributing algorithm, performance comparison – selecting a suitable load sharing algorithm – requirements for load distributing, task migration and associated issues. Failure Recovery and Fault tolerance: introduction, basic concepts, classification of failures, backward and forward error recovery, backward error recovery- recovery in concurrent systems, consistent set of check points, synchronous and asynchronous check pointing and recovery, check pointing for distributed database systems, recovery in replicated distributed databases.

Protection and security–preliminaries: the access matrix model and its implementations-safety in matrix model, advanced models of protection. Data security, cryptography: Model of cryptography, conventional cryptography, modern cryptography, private key cryptography, data encryption standard, public key cryptography, multiple encryption, authentication in distributed systems.

UNIT IV

Multiprocessor operating systems: basic multiprocessor system architectures, inter connection networks for multiprocessor systems, caching, hypercube architecture. Multiprocessor Operating System, structures of multiprocessor operating system, operating system design issues, threads-process synchronization and scheduling.

Database Operating systems: Introduction, requirements of a database operating system Concurrency control : theoretical aspects – introduction, database systems, a concurrency control model of database systems, the problem of concurrency control, serializability theory, distributed database systems, concurrency control algorithms, introduction, basic synchronization primitives, lock based algorithms-timestamp based algorithms, optimistic algorithms, concurrency control algorithms, data replication.

Course Outcomes:

At the end of this course the student shall be able to:

CO-1.	Analyze the design issues of distributed operating systems.
CO-2.	Evaluate design issues of multi-processor operating systems.
CO-3.	Identify the requirements Distributed File System and Distributed Shared Memory.
CO-4.	Formulate the solutions to schedule the real time applications.

References:

1. MukeshSinghal, NiranjanG.Shivaratri, "Advanced concepts in operating systems: Distributed, Database and multiprocessor operating systems", TMH, 2001.
2. Andrew S.Tanenbaum, "Modern operating system", PHI, 2003
3. Pradeep K.Sinha, "Distributed operating system-Concepts and design", PHI, 2003.

M.Sc. (Computer Science)
SEMESTER-III
MCS-536P
Programming Laboratory – III
(Based on Advanced Java Programming)
Skill Enhancement Course (SEC)

Time: 3 Hrs. **Total Marks: 50**

Credits		
L	T	P
0	0	2

Practical Marks: 37

Practical Internal Assessment Marks:13

Course Objectives:

1.	To introduce the object-oriented programming concepts.
2.	To understand object-oriented programming concepts and apply them in solving problems.
3.	To introduce the principals of inheritance and polymorphism and demonstrate how they relate to design of abstract class.
4.	To introduce the implementation of packages and interfaces.
5.	To introduce the concept of exception handling and multithreading.

Programming Laboratory based on Advanced Java Programming

Course Outcomes:

On Completing the course, the students will be able to:

CO-1.	Understand the concept of OOPs as well as the purpose and uses principal of inheritance, polymorphism in encapsulation and method overloading.
CO-2.	Identify classes, objects, members of a class and the relationship among them. needed for a specific problem.
CO-3.	Create a java applications programs using sound oops practices.
CO-4.	Develop programs using the java collection API as well as the java standard class library.
CO-5.	Develop and understand exception handling.

M.Sc. (Computer Science)
Semester IV

SN	Course Code	Course Name	Distribution of The Marks				Lectures Per week			Credit Distribution			Total Credit L+T+P	Page No.
			Theory	Internal Assessment	Practical	Total	L	T	P	L	T	P		
Discipline Specific Course(DSC)														
1	MCS-541 (Major)	Advanced Web Technologies using ASP.NET	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**41-42**
2	MCS-542 (Major)	Microprocessors and its Applications	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**43-44**
3	MCS-543 (Major)	Optional Paper Options (i) : Object Oriented Modelling, Analysis and Design (ii): Big Data Analytics Option (iii): Natural Language Processing	**75**	**25**	**0**	**100**	**5**	**1**	**0**	**3**	**1**	**0**	**4**	**45-46** **47-48** **49-50**
Skill Enhancement Course(SEC)														
5	MCS-544P	Programming Laboratory-IV (Based on Advanced Web Technologies using ASP.NET)	**0**	**13**	**37**	**50**	**0**	**0**	**6**	**0**	**0**	**2**	**2**	**51**
6	MCS-545P	Project Work	**0**	**50**	**150**	**200**	**0**	**0**	**24**	**0**	**0**	**8**	**8**	**52**
							Total Credits =			**22**				

M.Sc. (Computer Science)
SEMESTER-IV
MCS-541: Advanced Web Technologies using ASP.NET
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

1.	The students will Study the architecture of Dot Net framework.
2.	Understand the basic principles of website development using IDE.
3.	Learn advanced windows and web development techniques using dot NET.

UNIT-I

Introduction to .Net Framework

Developing console applications, C# Type Conversion Methods, boxing and unboxing, compiling & building projects, using command line argument, compiling a C# program using commandLine utility CSC.EXE

Introduction to Web Applications:

Standard Controls: Display information, accepting user input, submitting form data, displaying images, Using the panel control, Using the hyperlink control.

Validation Controls: Using the required field validator control, Using the range validator control using the compare validator control, Using the regular expression validator control, Using the custom validator control, Using the validation summary controls.

UNIT-II

Rich Controls: Accepting file uploads, displaying a calendar, displaying advertisement, displaying different page views, Displaying a wizard.

Designing Website with Master Pages: Creating master pages, modifying master page content, Loading master page dynamically.

SQL Data Source Control: Creating database connections, executing database commands, Using ASP.NET parameters with the SQL data source controls, programmatically executing SQL data source commands, Caching database data with the SQL data Source controls.

UNIT-III

List Controls: Dropdown list control, Radio button list controls, list box controls, bulleted list controls, custom list controls.

Grid View Controls: Grid view control fundamentals, using field with the grid view control, working with grid view control events extending the grid view control.

UNIT-IV

Building Data Access Components with ADO.NET: Connected the data access, Disconnected data access, executing a synchronous database commands, Building data base objects with the .NET framework.

Maintaining ApplicationState: Using browser cookies, using session state, Using profiles.

Caching Application Pages and Data: page output caching, partial page caching, data source caching, data caching, SQL cache dependences.

References:

1. ASP.NET 3.5: Stephen Walther, Pearson Education, 2005
2. ASP.NET 4.0: In Simple Steps by Kogent Learning Solutions Inc.
3. ASP.NET 4.5: Black Book by Kogent Learning Solution Inc.

Course Outcomes: On Completing the course, the students will be able to:

CO-1.	Evaluate C# and the .NET framework namespace contents.
CO-2.	Develop the console and GUI applications using C# .Net.
CO-3.	Set up various navigation techniques for integrating web pages within the site.
CO-4.	Create the dynamic web page using ASP.NET Controls which interact with databases.
CO-5.	Manage cookies and sessions as state management techniques.

M.Sc. (Computer Science)
SEMESTER-IV
MCS-542: Microprocessors and its Applications
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives:

The objective of this course is to:

1.	Learn detailed software and hardware architecture of 8088/8086 microprocessor.
2.	Understand the concept of Interrupts and their significance in 8088/8086 microprocessor.
3.	Analyse how different peripherals are interfaced with microprocessor.
4.	The student will be able to understand the future trends in microprocessor.

UNIT-I

Introduction to Microprocessor, General Architecture of Microcomputer System, Microarchitecture of 8088/8086 microprocessor , bus interface unit, execution unit concept of pipelining, Memory Address space and data organization , Register organization , Memory segmentation , Physical address generation, addressing modes of 8088/8086

UNIT-II

Description of various pins of 8088/8086 microprocessor , Minimum mode and maximum mode configurations, system clock, Bus cycle

Memory control signals, Read and write Bus cycles, hardware organization of memory address space, memory bus status code, program and data storage memory

UNIT-III

I/O interface, I/O address space and data transfer, I/O Control Signals, memory mapped I/O, Isolated I/O, I/O instructions, I/O bus cycles, Output ports, 8255A Programmable Peripheral Interface (PPI), Serial communication interface (USART and UART) – the RS- 232 C interface.

UNIT-IV

Interrupt Interface of 8086/8088 Microprocessor, Types of Interrupt, Interrupt Vector Table (IVT), Enabling and Disabling Interrupts, interrupt service routine

Features of Pentium processor, internal architecture of Pentium processor, operating modes. software architecture of Pentium processor, Register Sets , enhancement to instruction set.

References:

1. BARRY B. BREY, " THE INTEL MICROPROCESSORS-Architecture, Programming and Interfacing", Pearson Education India. Eighth Edition(2008)
2. Walter Triebel, Avtar Singh: The 8088 and 8086 Microprocessor – Architecture, Software and Interfacing Techniques, PHI, Delhi. Fourth Edition.
3. Douglas V. Hall: Microprocessors and Interfacing – Programming and Hardware, Tata McGraw Hill Publishing Company Ltd., New Delhi. Second Edition
4. Microcomputer Systems: The 8086/8088 Family - Yu Cheng Liu and Glen A Gibson – PHI. Second Edition

Course Outcomes:

At the end of this course the student shall be able:

CO-1.	To illustrate the architecture of 16 bit 8088/8086 microprocessor.
CO-2.	To use the various addressing modes of the microprocessor.
CO-3.	To understand higher processor architecture and embedded systems.
CO-4.	To understand the working of Programmable Peripheral interface.
CO-5.	To introduce the operations and interfacing techniques of 8088/8086 microprocessor

M.Sc. (Computer Science)
SEMESTER-IV
MCS-543
Option (i)
Object Oriented Modelling, Analysis and Design
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course Objectives: At the end of this course, the students will know:

1.	The importance of modeling in the software development life cycle.
2.	The UML notation and symbols.
3.	The object-oriented approach to analyzing and designing systems and software solutions.
4.	How to Employ the UML notation to create effective and efficient system designs.
5.	To inculcate necessary skills to handle complexity in software design.

UNIT-I

Object Orientation, OMT Methodology, Object and Class, Link and Association Generalization, Aggregation Multiple Inheritance, Packages.

UNIT-II

Object Meta Modeling, Metadata and Metamodels, Functional ModelingPseudocode with the Object navigation Notation, ONN Constructs, Combining ONN Constructs.

UNIT-III

Analysis**:** Object Model, Data Dictionary, Dynamic Model, Functional Model.

UNIT-IV

System Design :- Devising an Architecture, Database Management Paradigm, Object Model, Elaborating the functional Model, Evaluating the Quality of Design Model.

References:

1. Object Oriented Modeling and Design By Michael Blaha, William Premerlani, and Prentice Hall
2. Object-Oriented Analysis and Design with Applications by Booch, Pearson
3. Head First Object-Oriented Analysis and Design: A Brain Friendly Guide to OOA&D by Brett D. McLaughlin, David West and Gary Pollice
4. Object - Oriented Analysis and Design Using UML: Introduction to Unified Process and Design Patterns by Matha

Course Outcomes: On Completing the course, the students will be able to:

CO-1.	Select the basic elements of modeling such as Things, Relationships and Diagrams depending on the views of UML Architecture and SDLC.
CO-2.	Apply basic and Advanced Structural Modeling Concepts for designing real time applications.
CO-3.	Design Class and Object Diagrams that represent Static Aspects of a Software System.
CO-4.	Analyse Dynamic Aspects of a Software System using Use Case, Interaction and Activity Diagrams.
CO-5.	Apply techniques of State Chart Diagrams and Implementation Diagrams to model behavioral aspects and Runtime environment of Software Systems.

M.Sc. (Computer Science)
SEMESTER-IV
MCS-543
Option (ii)
Big Data Analytics
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course objectives: The objective of this course is:

1.	To study the basic technologies that forms the foundations of Big Data.
2.	To study the programming aspects of cloud computing with a view to rapid prototyping of complex applications.
3.	To understand the specialized aspects of big data including big data application, and big data analytics on Structured, Unstructured Data.
4.	To study different types Case studies on the current research and applications of the Hadoop and big data in industry.

UNIT-I

An Overview of Big Data and Big Data Analytics. Understanding Hadoop Ecosystem (Hadoop Distributed File System, MapReduce, Hadoop YARN, HBase, Combining HBaseand HDFS, Hive, Pig, Sqoop, ZooKeeper, Flume, Oozie). MapReduce Framework, Techniques to Optimize MapReduce Jobs, Role of HBase in Big Data Processing.

UNIT-II

Developing Simple MapReduce Application, Points to Consider while Designing MapReduce. Controlling MapReduce Execution with InputFormat, Reading Data with Custom RecordReader, Organizing Output Data with OutputFormats, Customizing Data with RecordWriter, Optimizing MapReduce Execution with Combiner, Controlling Reducer Execution with Partitioners.

UNIT-III

YARN Architecture, Working of YARN, YARN Schedulers, Backward Compatibility withYARN, YARN Configurations, YARN Commands, YARN Containers. Introduction to NoSQL. Types of NoSQL Data Models, Schema-Less Databases, Materialized Views, Distribution Models.

UNIT-IV

Analytical Approaches, Introducing to various Analytical Tools, Installing R, Handling Basic Expressions in R, Variables in R, Working with Vectors, Storing and Calculating Values in R, Creating and Using Objects, Interacting with Users, Handling Data in R Workspace, Executing Scripts, Reading Datasets and Exporting Data from R, Manipulating and Processing Data in R, Working with Functions and Packages in R, Performing Graphical Analysis in R, Techniques Used for Visual Data Representation, Types of Data Visualization

References:

1. Big Data, Black Book by DT Editorial Services, Dreamtech Press.
2. Big Data Computing and Communications edited by Yu Wang, HuiXiong, ShlomoArgamon, XiangYang Li, Jian Zhong Li Springer
3. Big Data Analytics Beyond Hadoop by Vijay Srinivas Agneeswaran, FT Press.

Course Outcomes:On Completing the course, the students will be able to:

CO-1.	Identify Big Data and its Business Implications.
CO-2.	Get knowledge about HDFS Concepts and Interfacing with HDFS. And learns R language.
CO-3.	List the components of Hadoop and Hadoop Eco-System.
CO-4.	Access and Process Data on Distributed File System.
CO-5.	Manage Job Execution in Hadoop Environment and Develop Big Data Solutions using Hadoop Eco System.

M.Sc. (Computer Science)
SEMESTER-IV
MCS-543
Option (iii)
Natural Language Processing
Discipline Specific Course (DSC)

Time: 3 Hrs. **Total Marks: 100**

Credits		
L	**T**	**P**
3	**1**	**0**

Theory Marks: 75

Theory Internal Assessment Marks:25

Note for paper setter and students:

1. **Medium of Examination is English Language.**
2. **There will be five sections.**
3. **Section A is compulsory and will be of 15 marks consisting of 8 short answer type questions carrying 2.5 mark each covering the whole syllabus. The answer should not exceed 50 words. The students will have to attempt any 6 questions in this section.**
4. **Sections B, C, D and E will be set from units I, II, III & IV respectively and will consist of two questions of 15 marks each from the respective unit. The students are required to attempt one question from each of these sections.**

Course objectives: The objective of this course is to:

1.	Study the basic technologies that forms the foundations of NLP.
2.	Study the significance of natural language processing in solving real-world problems.
3.	Study to map the appropriate processing technique to a problem and implement the technique.

UNIT-I

Introduction: Natural Language Processing (NLP), Challenges of NLP, NLP Applications, Processing of Indian Languages. Words and Word Forms: Morphology fundamentals; Morphological Diversity of Indian Languages; Morphology Paradigms; Finite State Machine Based Morphology; Automatic Morphology Learning; Shallow Parsing; Named Entities; Maximum Entropy Models; Random Fields, Scope Ambiguity and Attachment Ambiguity resolution.

UNIT-II

Structures : Theories of Parsing, Parsing Algorithms; Robust and Scalable Parsing on Noisy Text as in Web documents; Hybrid of Rule Based and Probabilistic Parsing; Scope Ambiguity and Attachment Ambiguity resolution.

UNIT-III

Machine Translation: Need of MT, Problems of Machine Translation, MT Approaches, Direct Machine Translations, Rule-Based Machine Translation, Knowledge Based MT System, Statistical Machine Translation, UNL Based Machine Translation, Translation involving Indian Languages.

UNIT-IV

Meaning: Lexical Knowledge Networks, WorldNet Theory; Indian Language Word Nets and Multilingual Dictionaries; Semantic Roles; Word Sense Disambiguation; WSD and Multilinguality; Metaphors.

Speech Recognition: Signal processing and analysis method, Articulation and acoustics, Phonology and phonetic transcription, Word Boundary Detection; Argmax based computations; HMM and Speech Recognition.

References:

1. Allen J., Natural Language understanding, Benjamin/Cunnings, (1987).
2. Siddiqui and Tiwary U.S., Natural Language Processing and Information Retrieval, Oxford University Press (2008).
3. Jensen K., Heidorn G.E., Richardson S.D., Natural Language Processing: The PLNLP Approach, Springer (2013).
4. Roach P., Phonetics, Oxford University Press (2012).
5. Jurafsky, Dab and Martin, James, Speechand Language Processing, Second Edition, Prentice Hall, 2008.

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

CO-1.	Describe the fundamental concepts and techniques of natural language processing.
CO-2	Distinguish among the various techniques, taking into account the assumptions, strengths, and weaknesses of each.
CO-3.	Use appropriate descriptions, visualizations, and statistics to communicate the problems and their solutions.
CO-4.	Analyze large volume text data generated from a range of real-world applications.

M.Sc. (Computer Science)
SEMESTER-IV
MCS-544 P
Programming Laboratory – IV
(Based on Advanced Web Technologies using ASP.NET)
Skill Enhancement Course (SEC)

Time: 3 Hrs. **Total Marks: 50**

Credits		
L	**T**	**P**
0	**0**	**2**

Practical Marks: 37

Practical Internal Assessment Marks:13

Course Objectives:

1.	To study the architecture of Dot Net framework and implement it using C# programming and visual studio.
2.	Understand the basic principles of website development using IDE and implement it.
3.	Learn advanced web development techniques using cookies and session, database connectivity and user profiles.

Programming Laboratory based on Advanced Web Technologies using ASP.NET
Course Outcomes:On Completing the course, the students will be able to:

CO-1.	Implement C# and the .NET framework namespace contents.
CO-2.	Develop the console and GUI applications using C# .Net.
CO-3.	Set up various navigation techniques for integrating web pages within the site.
CO-4.	Create the dynamic web page using ASP.NET Controls which interact with databases.
CO-5.	Manage cookies and sessions as state management techniques.

M.Sc. (Computer Science)
SEMESTER-IV
MCS-545P
Project Work
Skill Enhancement Course (SEC)

Time: 3 Hrs. **Total Marks: 200**

Credits		
L	**T**	**P**
0	**0**	**8**

Practical Marks: 150

Practical Internal Assessment Marks:50

Course Objectives:

1.	Develop skills in presentation and discussion of research topics in a public forum.
2.	Exposure to a variety of research projects and activities in order to enrich their academic experience
3.	It makes the student confident in designing an Online Project with advanced technologies on their choice
4.	Students are trained to meet the requirements of the industry.

The Project is to be prepared based on some current problems from industry / business / academic domain using some currently available technology / platform.

Note:

1. The end semester project work evaluation is to be conducted by following panel of examiners:-
   a. Internal Examiner
   b. External Examiner
2. The Project are to be submitted as per the common ordinances for P.G. courses under semester system.

Course Outcomes: On Completing the course, the students will be able to:

CO-1.	Carry out time planning for the project.
CO-2.	Follow correct grounding and shielding practices
CO-3.	Do effective trouble-shooting of the mini project.
CO-4.	Demonstrate a through and systematic understanding of project contents.
CO-5.	Understand methodologies and professional way of documentation and communication.
CO-6.	Know the key stages in development of the project.