Kavayitri Bahinabai Chaudhari
North Maharashtra University, Jalgaon

School of Computer Sciences

Master of Science in Information Technology

M. Sc. (Computer Science)

Syllabus

[*under Academic Flexibility*]

Faculty of Science and Technology

With effect from July 2023-24

KAVAYITRI BAHINABAI CHAUDHARI NORTH MAHARASHTRA UNIVERSITY, JALGAON

School of Computer Sciences, M.Sc. (Computer Science) PROGRAMME (A. Y. 2023-24)

Credit distribution structure for Two Years/One Year PG Degree and Ph. D. Programme

Year (2 Yr PG)	Level	Sem (2 Yr)	Major (Core) Subjects		RM	OJT/FP	RP	Cumulative Credits	Degree
			Mandatory (MCS)	Elective (MCSE)					
I	6.0	Sem-I	MCS-411 (4) (T) MCS-412 (4) (T) MCS-413 (2) (T) MCS-414 (2) (P) MCS-415 (2) (P)	MCSE-417A (2) (T) MCSE-417B (2) (P) OR MCSE-418A (2) (T) MCSE-418B (2) (P)	MCS-416 (4)	-	-	22	PG Diploma (After 3-YrDegree)
		Sem-II	MCS-421 (4) (T) MCS-422 (4) (T) MCS-423 (4) (T) MCS-424 (2) (T) MCS-425 (2) (P) MCS-426 (2) (P)	MCSE-427 (A) (4) (T) OR MCSE-427 (B) (4) (T)	-	-	-	22	
Cum. Cr. For PG Diploma			32	8	4	-	-	44	
Exit option: PG Diploma (44 Credits) after Three Year UG Degree									
II	6.5	Sem-III	MCS-511 (4) (T) MCS-512 (4) (T) MCS-513 (4) (T) MCS-514 (2) (T) MCS-515 (2) (P) MCS-516 (2) (P)	MCSE-517A (2) (T) MCSE-517B (2) (T) OR MCSE-518A (2) (T) MCSE-518B (2) (T)	-	-	-	20	M.Sc. (CS) After 3-Yr UG Or PG Degree after4-Yr UG
		Sem-IV	MCS-521 (4) (T) (NPTEL, Swayam) MCS-522 (4) (T) (NPTEL, Swayam)	-	-	MCS-523 (14) (P)	-	22	
Cum. Cr. for 2 Yr PG Degree			24	4	-	14	-	42	
Total Cr. for 2 Yr PG Degree			56	12	4	14	-	86	
2 Years-4 Sem. PG Degree (86 credits) after Three Year UG Degree or 1 Year-2 Sem PG Degree (44 credits) after Four Year UG Degree									

Abbreviations: Yr.: Year; Sem.: Semester; OJT: On Job Training: Internship/ Apprenticeship; FP: Field projects; RM: Research Methodology; Research Project: RP; Cumulative Credits: Cum. Cr., T- Theory Course, P – Practical course, MCS- School Specific Core Course, MCSE- School Specific Elective Course

Note: The courses which do not have practical, 'P' will be treated as 'T'.

KAVAYITRI BAHINABAI CHAUDHARI NORTH MAHARASHTRA UNIVERSITY, JALGAON

School of Computer Sciences, PG DEGREE M. Sc. (Computer Science) PROGRAMME (A. Y. 2023-24)

Credit distribution structure for Two Years/ One Year PG (M.Sc.) Degree Programme

Teaching and Examination scheme, Master of Science (M. Sc)

M.Sc. (Level 6.0) Sem- I (Name of Courses for - Major, RM, OJT, RP courses)

Sr No	Course Category	Name of the course(Title of the Paper)		Total Credit	Hours/ Semester	Teaching Scheme (hrs/week)		Evaluation Scheme		
						Theory	Practical	Continuous Internal Evaluation (CIE) (CA)	End Semester Evaluation (ESE) (UA)	Duration of Examination (Hrs)
						T	P			
1	Major	MCS-411 (4) (T)	Data Structures and Algorithms	4	60	4	-	40	60	3
		MCS-412 (4) (T)	Database Management System (DBMS)	4	60	4	-	40	60	3
		MCS-413 (2) (T)	Automata Theory and Computability	2	30	2	-	20	30	2
		MCS-414 (2) (P)	LAB on Data Structures	2	60	-	4	20	30	2
		MCS-415 (2) (P)	LAB on DBMS	2	60	-	4	20	30	2
2	Elective (Any One Group)	MCSE-417 A (2) (T)	Object Oriented Programming using JAVA	2	30	2	-	20	30	2
		MCSE-417 B (2) (P)	LAB on Object Oriented Programming using JAVA	2	60	-	4	20	30	2
		MCSE-418 A (2) (T)	Object Oriented Programming using C++	2	30	2	-	20	30	2
		MCSE-418 B (2) (P)	LAB on Object Oriented Programming using C++	2	60	-	4	20	30	2
3	Research	MCS-416	Research Methodology	4	60	--	4	40	60	3
			Total	22	420	12	16	220	330	

KAVAYITRI BAHINABAI CHAUDHARI NORTH MAHARASHTRA UNIVERSITY, JALGAON

School of Computer Sciences, PG DEGREE M. Sc. (Computer Science) PROGRAMME (A. Y. 2023-24)

Credit distribution structure for Two Years/ One Year PG (M.Sc.) Degree Programme

Teaching and Examination scheme, Master of Science (M. Sc)

M.Sc. (Level 6.0) Sem- II (Name of Courses for - Major, RM, OJT, RP courses)

Sr No	Course Category	Name of the course (Title of the Paper)		Total Credit	Hours/ Semester	Teaching Scheme (hrs/week)		Evaluation Scheme		
						Theory	Practical	Continuous Internal Evaluation (CIE) (CA)	End Semester Evaluation (ESE) (UA)	Duration of Examination (Hrs)
						T	P			
1	Major	MCS-421 (4) (T)	Python Programming	4	60	4	-	40	60	3
		MCS-422 (4) (T)	Design and Analysis of Algorithms	4	60	4	-	40	60	3
		MCS-423 (4) (T)	Artificial Intelligence	4	60	4	-	40	60	3
		MCS-424 (2) (T)	Compiler Construction	2	30	-	4	20	30	2
		MCS-425 (2) (P)	LAB on Python Programming	2	60	-	4	20	30	2
		MCS-426 (2) (P)	LAB on DAA	2	60	-	4	20	30	2
2	Elective	MCSE-427 A (4) (T) **OR** MCSE-427 B (4) (T)	Mathematical Foundations of Computer Science Optimization Algorithms	4	60	4	-	40	60	3
Total				22	390	14	12	210	330	-

KAVAYITRI BAHINABAI CHAUDHARI NORTH MAHARASHTRA UNIVERSITY, JALGAON

School of Computer Sciences, PG DEGREE M. Sc. (Computer Science) PROGRAMME (A. Y. 2023-24)

Credit distribution structure for Two Years/ One Year PG (M.Sc.) Degree Programme

Teaching and Examination scheme, Master of Science (M. Sc)

M.Sc. (Level 6.5) Sem- III (Name of Courses for - Major, RM, OJT, RP courses)

Sr No	**Course Category**	**Name of the course (Title of the Paper)**		Total Credit	**Hours/ Semester**	**Teaching Scheme (hrs/week)**		**Evaluation Scheme**		
						Theory	**Practical**	Continuous Internal Evaluation (CIE) (CA)	End Semester Evaluation (ESE) (UA)	Duration of Examination (Hrs)
						T	**P**			
1	Major	MCS-511 (4) (T)	Web Application Development Technology	4	60	4	-	40	60	3
		MCS-512 (4) (T)	Machine Learning (ML)	4	60	4	-	40	60	3
		MCS-513 (4) (T)	Software Engineering	4	60	4	-	40	60	3
		MCS-514 (2) (T)	Mobile Application Development	2	30	2	-	20	30	2
		MCS-515 (2) (P)	LAB on Web Application Development Technology	2	60	-	4	20	30	2
		MCS-516 (2) (P)	LAB on ML	2	60	-	4	20	30	2
2	Elective	MCSE-517A (2) (T) MCSE-517B (2) (P) **OR** MCSE-518A (2) (T) MCSE-518B (2) (P)	Digital Image Processing (DIP) LAB on DIP OR Natural Language Processing(NLP) LAB on NLP	2 2	30 60	2 -	- 4	20 20	30 30	2 2
			Total	22	420	16	12	220	330	--

KAVAYITRI BAHINABAI CHAUDHARI NORTH MAHARASHTRA UNIVERSITY, JALGAON

School of Computer Sciences, PG DEGREE M. Sc. (Computer Science) PROGRAMME (A. Y. 2023-24)

Credit distribution structure for Two Years/ One Year PG (M.Sc.) Degree Programme

Teaching and Examination scheme, Master of Science (M. Sc)

M.Sc. (Level 6.5) Sem- IV (Name of Courses for - Major, RM, OJT, RP courses)

Sr No	Course Category	Name of the course (Title of the Paper)		Total Credit	Hours/ Semester	Teaching Scheme (hrs/week)		Evaluation Scheme		
						Theory	Practical	Continuous Internal Evaluation (CIE) (CA)	End Semester Evaluatio n (ESE) (UA)	Duration of Examination (Hrs)
						T	P			
1	Major (NPTEL Swayam)	MCS-521	Getting Started with Competitive Programming (NPTEL, Swayam)	4	60	4	-	-	100	3
		MCS-522	Object Oriented System Development Using UML, Java and Patterns (NPTEL, Swayam)	4	60	4	-	-	100	3
2	FP/OJT, RP	MCS-523	Full Time Industrial Training	14	420	-	30	100	200	3
			Total	22	420	-	30	100	400	--

<u>Note</u>: Major Courses MCS-521 and MCS-522 must be completed from NPTEL (Swayam).

Semester-I

Course Code**: MCS-411** **Data Structures and Algorithms** Clock Hours**: 60**
Total Marks**: 100**

Course Objectives:

1) To impart the basic concepts of data structures and algorithms
2) To understand basic concepts about array, stacks, queues, linked lists, trees and graphs
3) 3 To impart knowledge of advance topics like AVL Trees, BTrees, B* and B+ Trees
4) To understand concepts about searching and sorting techniques
5) Apply hashing concepts for a given problem
6) To understanding about writing algorithms and step by step approach in solving problems with the help of fundamental data structures

Unit-I **[08]** **Max Marks:06**

Introduction to Data Structures and Algorithms: Algorithmic Notation: Format Conventions, Statement and Control Structures. Time and Space Analysis: Data types and Abstract data types, Types of Data structures; Primitive, Non primitive, Linear and Nonlinear Data structures

Unit-II **[08]** **Max Marks:20**

Array: Storage representation, operations and applications (Polynomial addition and subtraction) Stack: operations and applications (infix, postfix and prefix expression handling), Queue: operations and applications, Circular Queues: operations and applications, Concept of Double ended Queue and Priority Queue, Linked representation of stack and queue.

Unit-III **[10]** **Max Marks:12**

Linked Lists: Operations and Applications of Linear linked list (Polynomial addition and subtraction), Circular linked list and Doubly linked list.

Unit-IV **[11]** **Max Marks:24**

Trees: Binary Trees, Binary Tree: Representations, Operations (insert/delete), Traversal (inorder, preorder, postorder, level order), Threaded Binary Tree, Search Trees: AVL Tree, single and double rotations, B-Trees: insertion and deletion, Introduction to B+ and B* Trees

Unit-V **[11]** **Max Marks:18**

Graphs and Their Applications: Representation (Matrix/Adjacency) and Traversal (Depth First Search/Breadth First Search), Spanning Trees, Minimal Spanning Tree (Prim's and Kruskals's algorithm), Shortest Paths and All Pair Shortest Path, Dijkstra's, Floyd-Warshall Algorithms.

Unit-VI **[12]** **Max Marks:20**

Hash Table: Hash Function, Collision and its Resolution, Separate Chaining, Open Addressing (linear probing, quadratic probing, double hashing), Rehashing, Extendible Hashing Searching: Linear Search and Binary Search (array/binary tree). Sorting: General Background, Sorting Techniques: Bubble Sort, Insertion Sort, Selection Sort, Quicksort, Merge sort, Heapsort and Radix Sort.

References:

1. Tremblay, J. & Sorenson, P.G., (2001), An Introduction to Data Structures with Application, Mcgraw Hill India, ISBN: 978-0074624715, 0074624717

2. Langsam, Y., Augenstein, M.J. & Tenenbaum A.M., (2015), Data Structures using C School of Computer Sciences, KBCNMU, Jalgaon M.Sc. (Computer Science) Syllabus-2019-20 Page 9 and C++, 2nd Edition, Pearson Education ISBN: 978-9332549319, 9332549311

3. Balagurusamy, E., (2013), Data Structures using C, 1st Edition, Mcgraw Hill Education,

ISBN: 978-1259029547, 1259029549

4. Weiss, M.A., (2002), Data Structures and Algorithm Analysis in C, 2nd Edition, Pearson India, ISBN: 978-8177583588, 8177583581

5. Horowitz, E., Sartaj S. & Mehta, D. (2008), Fundamentals of Data Structures in C++, Universities Press ISBN: 978-8173716065, 8173716064 *

6. Lafore, R., (2003), Data Structures & Algorithms in Java, 2nd Edition, Pearson India, ISBN: 978-8131718124, 8131718123

7. Kruse, R., Tondo, C.L., Leung B., & Mogalla S, (2006), Data Structures and Program Design in C, Pearson India, ISBN: 978-8177584233.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS-411C.1	Understand the concept of Dynamic memory management, data types, algorithms, Big O notation.	2
MCS-411C.2	Understand data structures such as arrays, linked lists, stacks and queues, graphs, trees and hash tables.	2
MCS-411C.3	Solve problem involving graphs, trees and apply different sorting and searching algorithms.	3

Course Code**: MCS-412** **Database Management System (DBMS)** Clock Hours**: 60**
Total Marks**: 100**

Course Objectives:

1) The course emphasizes the understanding of the fundamentals of relational systems including data models, database architectures, and database manipulations.

2) To develop conceptual understanding of database management system

3) To understand how a real-world problem can be mapped to schemas

4) To educate students with different Database Languages.

5) Understand the needs of database processing and learn techniques for controlling the consequences of concurrent data access.

Unit-I **[05]** **Max Marks:08**

Introduction: Database system application and purpose, Characteristics of DBMS, Database Users, 1-tier, 2-tier and 3-tier architecture of DBMS along with its advantages, Levels of Database Architecture, Data Models, Data-schemas and instances, Data Independence, Role and responsibilities of DBA.

Unit-II **[10]** **Max Marks:10**

Database Design and E-R Model: Overviews of Database Design, ER Modeling concepts, ER Diagrams, Reduction to Relational Schemas, Extended ER Features, Alternative notations for Modelling, Cardinality constraints, Atomic Domains and 1NF, Decomposition using Functional Dependencies (BCNF, 3NF and 4NF).

Unit-III **[12]** **Max Marks:20**

Relational Databases: Structure of Relational Databases, Database Schemas, Keys, Schema diagrams, Relational Query Languages, Relational Operation. Overview of SQL, SQL Data Definition, Basic Structure of SQL Queries, Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Sub queries, Modification of Databases. Join Expressions, Views, Transactions, Integrity Constraints, SQL data types and Schemas, Authorization, Accessing SQL from Programming Languages, Overview of Dynamic SQL and SQL CLI. Functions and Procedures, Triggers. The relational Algebra fundamental and extended Operations. Tuple and Domain Relational Calculus.

Unit-IV **[10]** **Max Marks:22**

Transaction Management and Query Processing: Transaction Concept, Model, Storage Structure, Atomicity and Durability, Isolation, Levels of Isolation, Overview of Query Processing, Measuring Query Cost, Selection Operation, Sorting, Join Operation, Other Operations and Evaluation of Expression. Overview of Query Optimization, Transformation of Relational Expression, Choice of Evaluation Plan.

Unit-V **[10]** **Max Marks:15**

Concurrency Control and Recovery System: Lock based Protocol, Timestamp based Protocol, Validation based Protocol, Deadlock Handling, Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithms, Buffer Management, Early lock release and logical undo operations, Remote Backup Systems. Case study: ARIES

Unit-VI **[13]** **Max Marks:25**

Advanced Topics in Databases: Introduction to Object Databases: Shortcomings of Relational Data Model, The Conceptual Object Data Model, Objects in SQL:1999 and SQL:2003. Introduction to XML and Web Data: Semi structured Data, Overview of XML, XML Data Definitions, XML Schema, XML Data Manipulation: XQuery, XPath Query Languages: XPath and SQL/XML. Distributed Databases: Overview, Homogeneous and Heterogeneous Databases, Distributed Data Storage, Distributed Transactions, Commit Protocols, Concurrency Control, Cloud based Databases**.**

References:

1. Michael Kifer, Arthur Bernstein, P.M, Lewis and P.K. Panigrahi, “Database Systems: An Application Oriented Approach”, Second Edition, Pearson Education, ISBN:978-81-317 0374-8.
2. C.J.Date, A. Kannan and S. Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, ISBN:978-81-7758-556-8
3. A. Silberschatz, H.F.Korth, and S.Sudarshan, “Database System Concepts”, TMH Publications, Sixth Edition, ISBN:978-007-132522-6.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS412C.1	Apply the relational model, specify integrity constraints, and explain how to create a relational database using an ER diagram and normalization techniques.	3
MCS412C.2	Apply SQL to create, query and manipulate relational databases.	6
MCS412C.3	Determine partitioning and distribution of data across networked nodes of a DBMS and data optimization in a distributed environment.	3

Course Code**: MCS-413** **Automata Theory and Computability** Clock Hours**: 30**
Total Marks**: 50**

Course Objectives:

1) To learn fundamentals of Grammars and Languages.
2) To understand the relation between Regular Language and Finite Automata and machines.
3) To learn how to design Automata’s and machines as Acceptors, Verifiers and Translators.
4) To understand the relation between Contexts free Languages, PDA and TM.
5) To learn how to design PDA as acceptor and TM as Calculators.

6) To understand the decidability and complexity measures.

Unit-I **[06]** **Max Marks:10**

Grammars: Production systems, Chomskian Hierarchy, Right linear grammar and Finite state automata, Context free grammars, Normal forms, uvwxy theorem, Parikh mapping, Self embedding property, Subfamilies of CFL, Derivation trees and ambiguity

Unit-II **[08]** **Max Marks:12**

Finite State Automata: Nondeterministic and deterministic FSA, NFSA with ε- moves, Regular Expressions, Equivalence of regular expression and FSA, Pumping lemma, closure properties and decidability, Myhill - Nerode theorem and minimization, Finite automata with output

Unit-III **[08]** **Max Marks:14**

Pushdown Automata: Acceptance by empty store and final state, Equivalence between pushdown automata and context-free grammars, Closure properties of CFL, Deterministic pushdown automata

Unit-IV **[08]** **Max Marks:14**

Turing Machines: Techniques for Turing machine construction, Generalized and restricted versions equivalent to the basic model, Godel numbering, Universal Turing Machine, Recursively enumerable sets and recursive sets, Computable functions, time space complexity measures, context sensitive languages and linear bound automata

References:

1. K. Krithivasan and R. Rama (2009). Introduction to Formal Languages, Automata Theory and Computation: Pearson Education, ISBN 9788131723562.
2. J. E. Hopcroft, R.Motwani and J.D.Ullman (2001). Introduction to Automata Theory Languages and computation: Pearson Education Asia, ISBN 978-0321455369.
3. Peter Linz (2006). An Introduction to Formal Language and Automata 4th Edition: Narosa Publishing house, ISBN 978-1-4496-1552-9.
4. M.Sipser (1997). Introduction to the Theory of Computation; Singapore: Brooks/Cole, Thomson Learning ISBN, 978-1133187790.
5. John. C. Martin (2003). Introduction to the Languages and the Theory of Computation Third edition Tata McGraw-Hill ISBN 9780070660489.
6. http://nptel.ac.in/

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS413C.1	Differentiate between deterministic and nondeterministic automata, design finite automata, equivalence of languages described by finite automata and regular expressions.	4
MCS413C.2	Devise regular, context free grammars while recognizing the strings and tokens and design pushdown automata to recognize the language.	4
MCS413C.3	Develop an understanding of computation through Turing Machine.	6

Course Code: **MCS-414** **LAB on Data Structures** Clock Hours**: 30**
Total Marks**: 50**

Course Objectives:

1) Solve real-world problems by reasoning about data structure choices, choose appropriate implementations.
2) To make the students write various programs and ADTS for all data structures.
3) Students will learn to write, debug, and test large programs systematically.

Implementation of programs based on the following

- Arrays
- Multidimensional Arrays, Matrices
- Stacks, Polish Notation
- Queues
- Deques
- Linear Linked List, Circular Linked List, Doubly Linked List
- Polynomial Addition/Subtraction

Implementation of programs based on Trees

- Binary Search Tree
- In-order, Pre-order and Post-order Traversals
- Heap Tree

Implementation of programs based on Graphs

- Depth First Traversal
- Breadth First Traversal
- Obtaining Shortest Path (Dijkstra and Floyd-Warshall)
- Minimum spanning tree (Kruskal and Prim)

Implementation of programs for Hash Table, Searching and Sorting techniques

- Hash Table
- Linear and Binary Search (using array)
- Bubble sort
- Selection sort
- Insertion sort
- Radix sort
- Quick sort
- Merge sort
- Heap sort

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS414C.1	Develop solutions for a range of problems using procedure oriented / object-oriented programming.	6
MCS414C.2	Choose the appropriate data structure and algorithm design method for a specified application.	3

Course Code: **MCS-415** **LAB on DBMS** Clock Hours**: 30**
Total Marks**: 50**

Course Objectives:

1) The course mainly concentrates on understanding of the fundamentals of Data Definition Language and Data Manipulation Languages.

2) To develop conceptual understanding of database management system
3) To understand how a real-world schema can be implemented
4) To educate students with different Database Languages.
5) Understand the needs of database processing and learn techniques for controlling the
6) consequences of concurrent data access.

1. Creating database tables and using data types.
   - Create table
   - Modify table
   - Drop table
2. Practical Based on Data Manipulation.
   - Adding data with Insert
   - Modify data with Update
   - Deleting records with Delete
3. Practical Based on Implementing the Constraints.
   - NULL and NOT NULL
   - Primary Key Constraint
   - Foreign Key Constraint
   - Unique Constraint
   - Check Constraint
   - Default Constraint
4. Practical for Retrieving Data Using following clauses.
   - Simple select clause
   - Accessing specific data with Where
   - Ordered By
   - Distinct
   - Group By
5. Practical Based on Aggregate Functions.
   - AVG
   - COUNT
   - MAX
   - MIN
   - SUM
   - CUBE
6. Practical Based on implementing all String functions.
7. Practical Based on implementing Date and Time Functions.
8. Practical Based on implementing use of UNION, INTERSECTION, SET DIFFERENCE.
9. Implement Nested Queries & all types of JOIN operation.
10. Practical Based on performing different operations on a view.
11. Practical Based on implementing use of Procedures.
12. Practical Based on implementing use of Triggers
13. Practical Based on implementing Cursor.
14. ++++VB.NET, C#.NET, JAVA, D2K, etc.
15. Practical based on creating Data Reports.
16. Design entity relationship models for a business problem and develop a normalized database structure

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS415C.1	Design and implement a database schema for a given problem-domain.	6

MCS415C.2	Create and maintain tables using PL/SQL, Populate and query a database using SQL DML/DDL commands and programming PL/SQL including stored procedures, stored functions, cursors, triggers.	6
MCS415C.3	Application development using PL/SQL & front-end tools.	3

Course Code: **MCS-416** **Research Methodology** Clock Hours: **60**

Total Marks: **100**

Course Objectives:

1) To study and understand the research issues & challenges, research goals, scientific methods
2) To study Sampling, External Validity, Levels of Measurement, Scaling and Qualitative Measures. Data Preparation, Descriptive Statistics and Correlation; and Inferential Statistics
3) Reviewing Literature and Research Papers; Writing Research Papers, Thesis, Reports and Project Proposals Plagiarism and Copyrights.

Unit-I **[08]** **Max Marks:10**

Research Foundations: Research, Research Goals and Quality Research, Types of Research, Variables, Hypotheses and Data; Structure, Positivism and Post-Positivism; Scientific Methods, Reasoning and Arguments; Mathematical Methods of Proof and Research Fallacies.

Unit-II **[08]** **Max Marks:15**

CS Research Context: Nature of Computer Science, Scientific Methods in Computer science, Types of Research in CS, Research Methods in Computer Science, Research Paradigms in CS, Grand Challenges for CS Research.

Unit-III **[10]** **Max Marks:12**

Measurements: Sampling, External Validity, Levels of Measurement, Scaling and Qualitative Measures.

Research Design: Internal Validity, Types of Designs, Experimental Design, Probabilistic Equivalence, Hybrid Experimental Designs and Quasi-Experimental Design.

Unit-IV **[11]** **Max Marks:25**

Sampling: Concepts of Statistical Population, Sample, Sampling Frame, Sampling Error, Sample Size, Non Response. Characteristics of a good sample. Probability Sample – Simple Random Sample, Systematic Sample, Stratified Random Sample & Multi-stage sampling. Determining size of the sample – Practical considerations in sampling and sample size.

Unit-V **[11]** **Max Marks:20**

Data Analysis: Data Preparation – Univariate analysis (frequency tables, bar charts, pie charts, percentages), Bivariate analysis – Cross tabulations and Chi-square test including testing hypothesis of association.

Unit-VI **[12]** **Max Marks:18**

Research Skills: Reviewing Literature and Research Papers; Writing Research Papers, Thesis, Reports and Project Proposals; Formatting, Appendices, Citation Formats and Style; General Conventions, Issues, Plagiarism and Copyrights.

References:

1. Research Methodology: a step-by-step guide for beginners, Kumar, Pearson Education.
2. Kothari C.K. (2004) 2/e, Research Methodology – Methods and Techniques (New Age International, New Delhi).
3. Practical Research Methods, Dawson, C., UBSPD Pvt. Ltd.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS416C.1	Understand the basic concepts of research and its methodologies, identify appropriate research topics, select and define appropriate research problem and parameters	3
MCS416C.2	Prepare a project proposal	4
MCS416C.3	Organize and conduct research in a more appropriate manner, writing research report and thesis.	4

Course Code: **MCSE-417A** **Object Oriented Programming using JAVA** Clock Hours**: 30**
Total Marks**: 50**

Course Objectives:

1) To learn fundamentals of Java programming language and its constructs.
2) To understand concept of object-oriented programming concept using Java.
3) To study the concept of the Inheritance, Interfaces, Lambda Expressions, and Inner Classes.
4) To understand the concept of the Exceptions and Generic Programming
5) To learn about the Graphics Programming, Event Handling, Swing Components, and Database Programming

Unit-I [04] Max Marks:06

An Introduction to Java: Java as a Programming Platform, The Java "White Paper" Buzzwords, Java Applets and the Internet, Common Misconceptions about Java, The Java Programming Environment, Installation, A Simple Java Program, Comments, Data Types, Variables, Operators, Strings, Input and Output, Control Flow, Big Numbers, Arrays.

Unit-II [05] Max Marks:06

Objects and Classes: Introduction to Object-Oriented Programming, Using Predefined Classes, Defining Your Own Classes, Static Fields and Methods, Method Parameters, Object Construction, Packages, The Class Path, Documentation Comments

Unit-III [05] Max Marks:08

Inheritance, Interfaces, Lambda Expressions, and Inner Classes: Classes, Super classes, and Subclasses, Object: The Cosmic Superclass, Generic Array Lists, Object Wrappers and Autoboxing, Methods with a Variable Number of Parameters, Enumeration, Classes, Reflection, Interfaces, Examples of Interfaces, Lambda Expressions, Inner Classes, Proxies

Unit-IV [08] Max Marks:12

Graphics Programming, Event Handling and Swing Components: Introducing Swing, Creating a Frame, Positioning a Frame, Displaying Information in a Component, Working with 2D Shapes, Using Color, Using Special Fonts for Text, Displaying Images, Basics of Event Handling, Actions, Mouse Events, The AWT Event Hierarchy, Swing and the Model View-Controller Design Pattern, Introduction to Layout Management, Text Input, Choice Components, Menus, Sophisticated Layout Management, Dialog Boxes.

Unit-V [08] Max Marks:10

Deployment and Concurrency and Database Programming: JAR Files, Storage of Application Preferences, Service Loaders, Applets, Java Web Start, Threads, Interrupting Threads, Thread States, Thread Properties, Synchronization, Blocking Queues, Thread-Safe Collections, Callables and Futures, Executors, Synchronizers, Threads and Swing, The Design of JDBC, The Structured Query Language, JDBC Configuration, Working with JDBC

Statements, Query Execution, Scrollable and Updatable Result Sets, Row Sets, Metadata.

References:

1. Cay S. Horstmann Core Java Volume I—Fundamentals (December 2015), Tenth Edition, Prentice Hall, ISBN: 9780134177335
2. Cay S. Horstmann Core Java, Volume II—Advanced Features (December 2016), Tenth Edition, Prentice Hall, ISBN: 9780134177878
3. Herbert Schildt, Java: The Complete Reference, Ninth Edition, McGraw Hill Education, ISBN 978-0-07-180855-2

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS417AC.1	Understands the fundamentals of Java programming language and its constructs.	2
MCS417AC.2	Implement the applications using the concept of the Inheritance, Interfaces and Inner Classes.	3
MCS417AC.3	Understand how to use concept of the Graphics Programming, Event Handling, Swing Components, and JDBC in their application.	2

Course Code: **MCSE-417B** **LAB on Object Oriented Programming using JAVA** Clock Hours**: 30** Total Marks**: 50**

Course Objectives:

1) Programming using inner classes and inheritance, polymorphism and interfaces
2) Use various swing components and handle several events in the development of GUI
3) applications
4) Use JDBC and package creation

1. Write a program that demonstrate program structure of java
2. Write a program that demonstrate string operations using String and StringBuffer class.
3. Write a program that demonstrate inner class and static fields.
4. Write a program that demonstrate inheritance, polymorphism.
5. Write a program that demonstrate 2D shapes on frames.
6. Write a program that demonstrate color and fonts.
7. Write a program to illustrate use of various swing components.
8. Write a program that demonstrate use of dialog box and menus.
9. Write a program that demonstrate event handling for various types of events.
10. Write a program to illustrate multithreading.
11. Write a program to illustrate exception handling.
12. Write a program to demonstrate use of File class.
13. Write a program that demonstrate JDBC on application.
14. Write a program that demonstrate package creation and use in program.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level

MCS417BC.3	Write Java application programs using Implementing user interface: 2D shapes, events, dialog box, menu and popup menu.	3
MCS417BC.4	Develop Applets, multithreaded, generic and JDBC applications.	6

Course Code: **MCSE-418A** **Object Oriented Programming using C++** Clock Hours**: 30**
Total Marks**: 50**

Course Objectives:

1) To familiarize the Object-Oriented Programming (OOP) concepts, such as abstraction, encapsulation, instances, initializations, polymorphism, overloading, inheritance etc.
2) To write programs to solve problems using generic programming constructs such as templates and using standard template library
3) To understand and know the importance of pointers and learn file handling and exception handling in real-world problems.

Unit-I [05] Max Marks:15

Fundamentals: Object-Oriented Programming (OOP): Need, Benefits of OOP, C++ programming Basics: Data types, Enumerations, Arrays, Strings, Pointers and references, Control structures. Functions: Function prototypes, parameter lists and return values, default values, global scoping, referencing, the 'const' keyword, referencing of strings, constant pointers, inline functions, static functions, function overloading, friend functions. OOP Concepts: The 'Struct' keyword, Functions within structures, Data encapsulation and classes, 'this' pointer, Constructors and Destructors, Overloading constructors, Copy Constructor, Assignment and Copy Initialization, Methods and their return values, Objects and Memory requirements, Static Class members, friend class.

Unit-II [10] Max Marks:15

Inheritance and Polymorphism: Base Class and derived Class, access specifiers, Constructor and Destructor in Derived Class, Virtual destructor, Protected members, Overriding member functions, Public and Private Inheritance, Multiple Inheritance, Ambiguity in Multiple Inheritance, Composition, Nested Classes.

Polymorphism: Operator Overloading: concept of overloading, Overloading Unary Operators, Overloading Binary Operators, Data Conversion, Type casting (implicit and explicit), Pointers- indirect ion Operators, Memory Management: new and delete, Pointers to Objects. Virtual Functions: concept, pure virtual functions and abstract classes, arrays in polymorphism, late binding, Function pointers

Unit-III [05] Max Marks:05

Files and Streams: Data hierarchy, Classes for File Stream Operations, File Pointers, File I/O with Member Functions, Overloading the Extraction and Insertion Operators, memory as a Stream Object, Command-Line Arguments.

Unit-IV [05] Max Marks:10

Templates and Exception Handling:

Templates: Function templates, Template specialization, Class templates, Non-type parameters for templates, template, and inheritance,

Exception Handling: Other error handling techniques, Exceptions, Exception handling in C++, rethrowing an exception, exception specifications, processing unexpected exceptions, stack unwinding, exception handling in constructors, destructors.

Unit-V [05] Max Marks:05

Standard Template Library (STL): Introduction to STL: Containers, algorithms, adaptors, and iterators, Containers: Sequence container and associative containers, Adaptors: container adapters, iterator adaptors, Algorithms: basic searching and sorting algorithms, min-max algorithm, set operations, Iterators: input, output, forward, bidirectional and random access.

References:

1. Robert Lafore, Object-Oriented Programming in C++, fourth edition, Sams Publishing, ISBN:0672323087.
2. Bjarne Stroustrup, The C++ Programming language, Third edition, Pearson Education ISBN 0-201-88954-4.
3. Meeta Gandhi, Tilak Shetty, RajivShah, Vijay Mukhi's The 'C' Odyssey C++ and Graphics-The future of C, BPB publications, First Edition

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS418AC.1	Develop logic of a program for solving real time problems and isolate and fix common errors in C++ programs.	4
MCS418AC.2	Understand the object-oriented approach for the program development and make use of the OOP concepts (data abstraction, encapsulation, polymorphism, overloading, and of C++ appropriately in problem solving.	3
MCS418AC.3	Design applications using the STL library.	4

Course Code: **MCSE-418B** **LAB on Object Oriented Programming using C++** Clock Hours**: 30** Total Marks**: 50**

Course Objectives:

1) Apply object-oriented approaches to software problems in C++
2) Apply exception handling techniques to software problems in C++
3) Apply generic programming approaches using templates and efficiently use standard template library in software development

1. Write program to demonstrate class, use of constructor, constructor overloading and destructor.
2. Write program to demonstrate use of arrays, strings, pointers, constants, and references.
3. Write program to demonstrate use of operator overloading.
4. Write program(s) to demonstrate use of inheritance.
5. Write program to demonstrate use of compile time and runtime polymorphism.
6. Write program to demonstrate use of friend function and friend class.
7. Write program to demonstrate use of virtual class.
8. Write program to demonstrate use of static data member and static member function.
9. Write program to demonstrate file handling.
10. Write program to demonstrate use of function templates.
11. Write program to demonstrate use of class templates.
12. Write program to demonstrate use of exception handling.
13. Write program to demonstrate command line arguments.
14. Write program(s) to demonstrate use of STL.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level

MCS418BC.1	Understands the fundamentals of C++ programming language and its constructs.	2
MCS418BC.2	Implement the applications using the object-oriented programming concepts.	3
MCS418BC.3	Understand how to implement file handling exception handling and standard template library for C++ application development.	2

Semester-II

Course Code: **MCS-421** **Python Programming** Clock Hours: **60**
Total Marks: **100**

Course Objectives:

The student should be able to

1) understand the fundamental concepts of Python programming.
2) learn that how python programming supports some constructs of functional programming.
3) work with strings, lists, tuples, dictionaries, and files.
4) define their own classes, methods and module for solving real world problems.
5) use regular expression for searching patterns in given strings.

Unit-I **[12]** **Max Marks:20**

The Python Programming Language, Python Data, Variables, Expressions and Statements, Values and Data Types, Type conversion Functions, Operators and Operands, Input, Order of Operations, Functions, Calling Functions, Passing Functions, Formal Arguments, Variable-length Arguments, Functional Programming, Boolean Expressions, Logical operators, Precedence of Operators, Conditional Execution, Unary Selection, Nested conditionals, Chained conditionals, Boolean Functions, Iteration, The for loop, The while Statement

Unit-II **[08]** **Max Marks:15**

Strings, A Collection Data Type, Operations on Strings, Index Operator: Working with the Characters of a String, String Methods, Length, The Slice Operator, String Comparison, Lists, List Values, List Length, Accessing Elements, List Membership, Concatenation and Repetition, List Slices, Lists are Mutable, List Deletion, Objects and References, Aliasing, Cloning Lists, Repetition and References, List Methods, Append versus Concatenate Lists and for loops, Using Lists as Parameters, Nested Lists, Strings and Lists, List Type Conversion Function, Tuples, Tuple operators and built-in functions, Tuples and Mutability, Tuple Assignment, Tuples as Return Values

Unit-III **[15]** **Max Marks:20**

Dictionaries, Dictionary Operations, Dictionary Methods, Dictionary Keys, Aliasing and Copying, Sparse Matrices, Working with Data Files, Finding a File on your Disk, Reading a File, Iterating over lines in a file, Writing Text Files, Object Oriented Programming, Classes, Instances, Class method Calls, Coding Class Tree, Attributes, Building and Method Invocation, Composition, Inheritance, Operator Overloading, Encapsulation and Information Hiding, Search Algorithms, Sorting Algorithms, Hash Tables

Unit-IV **[10]** **Max Marks:20**

Regular Expressions, Exceptions, Standard Exceptions, Exceptions Syntax, The try/except/else Statement, The try/finally Statement, Unified try/except/finally, The raise Statement, The assert Statement, with/as Context Managers String-Based Exceptions, Class-Based Exceptions, General raise Statement Forms, Nesting Exception Handlers, Exception Idioms, Exception Design Tips. Catch All Exceptions, Catch A Specific Exception, Catch Multiple Specific Exceptions, Clean-up After Exceptions, GUI Programming using TKinter.

Unit-V **[15]** **Max Marks:25**

Advance Function Topics: Anonymous Function Lambda, Mapping Functions over Sequences: map, Functional Programming Tools: filter and reduce, List Comprehensions Revisited: Mappings. Modules: Python Program Architecture, Module Creation, Module usage, Module Namespaces, Reloading Modules, Module Packages. Data Hiding in Modules, Enabling Future Language Features, Mixed Usage Modes, Changing the Module Search

Path, The import as Extension, Relative Import Syntax, Module Design Concepts

References:

1. John V Guttag (2013), Introduction to Computation and Programming Using Python, Prentice Hall of India, 2013, ISBN: 9780262525008
2. R. Nageswara Rao(2016), Core Python Programming, Dreamtech Press, 2016, ISBN-13: 9789351199427
3. Wesley J. Chun(2006), Core Python Programming - Second Edition, Prentice Hall, ISBN-13: 978-0132269933, ISBN-10: 0132269937
4. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser(2013), Data Structures and Algorithms in Pyhon", Wiley, 2013, ISBN : 978-1-118-54958-2, ISBN : 978-1-118-29027-9(HardCover)
5. Kenneth A. Lambert(2011), Fundamentals of Python – First Programs, CENGAGE Publication, 2011, ISBN 1111822700, ISBN 9781111822705
6. Luke Sneeringer(2015), Professional Python, Wiley Inc.,2015, ISBN: 1119070856
7. Mark Lutz (2007), Learning Python, 3rd Edition, O'Reilly Media, Inc., 2007, ISBN-13: 978-0-596-51398- 6, ISBN-10: 0-596-51398-4

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS421C.1	Use lists, tuples, dictionaries, strings and files efficiently for solving real world problems.	3
MCS421C.2	Implement the concepts of object-oriented programming using python.	3
MCS421C.3	Develop modules, packages and GUI based programming for web.	6

Course Code**: MCS-422** **Design and Analysis of Algorithms** Clock Hours**: 60**
Total Marks**: 100**

Course Objectives:

To Understand and learn

1) Basic concepts of algorithms and analyze the performance of algorithms.
2) Algorithm design techniques for developing algorithms.
3) Searching and traversal algorithms for graphs.
4) Nondeterministic algorithms and NP class of problem

Unit-I **[10]** **Max Marks:15**

Introduction: What Is An Algorithm? Algorithm Specification, Pseudocode Conventions, Recursive Algorithms, Complexity, Asymptotic Notation, Practical Complexities And Performance Measurement

Tree And Graph Representations, Binary Trees Basics, Heaps And Heap Sort, Sets And Disjoint Set Union And Find.

Unit-II **[12]** **Max Marks:15**

Divide and Conquer: General Method, Binary Search, Finding Maximum and Minimum, Merge Sort, Quick Sort, Strassen's Matrix Multiplication.

Unit-III **[08]** **Max Marks:15**

The Greedy Method: General Method, Knapsack Problem, Huffman Code, Minimum-Cost Spanning Trees (Prim's & Kruskal's Algorithm), Optimal Storage On Tapes, Single-Source

Shortest Paths.

Unit-IV **[08]** **Max Marks:15**

Dynamic Programming: General Method, All-Pair Shortest Path, Matrix Chain Multiplication, Longest Common Sub Sequence, 0/1knapsack, Flow Shop Scheduling

Unit-V **[06]** **Max Marks:15**

Basic Search and Traversal Techniques: Breadth First Search and Traversal, Depth First Search and Traversal, Spanning Trees.

Unit-VI **[08]** **Max Marks:15**

Backtracking: General Method, Constrains, 8-Queens Problem Graph Coloring

References:

1. Horowitz E. and Sahni S. "Fundamentals of computer Algorithms" Galgotia publications. ISBN:0716783169
2. Horowitz E., Sahni S. and Rajshekaran S(),Computer Algorithms, Computer Science Press, ISBN-10: 8173716129
3. S. Dasgupta, C. H. Papadimitriou, and U. V. Vazirani(2006), Algorithms. McGraw-Hill publications, ISBN 9780073523408
4. Cormen, Leiserson and Rivest, Introduction to Algorithms, Prentice Hall of India, ISBN: 978-81-203-4007-7

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS422C.1	Analyze the asymptotic performance of algorithms and write rigorous correctness proofs for algorithms.	4
MCS422C.2	Design and analyze divide-and-conquer, greedy and dynamic-programming based algorithms.	4
MCS422C.3	Model problems using backtracking, and to classify nondeterministic polynomial time algorithms.	6

Course Code**: MCS-423** **Artificial Intelligence** Clock Hours**: 60**
Total Marks**: 100**

Course Objectives:

The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence. Emphasis will be placed on the teaching of these fundamentals, not on providing a mastery of specific software or tools programming environments.

The student should be made to:

1) Gain a historical perspective of AI and its foundations.
2) Study the concepts of Artificial Intelligence.
3) Investigate applications of AI techniques in intelligent agents
4) Learn the methods of solving problems using Artificial Intelligence.
5) Learn various peculiar search strategies for AI

Unit-I **[08]** **Max Marks:10**

Introduction: Overview and Historical Perspective, Turing test, Physical Symbol Systems and the scope of Symbolic AI, Agents.

Unit-II **[06]** **Max Marks:10**

State Space Search: Depth First Search, Breadth First Search, DFID.

Unit-III **[08]** **Max Marks:12**

Heuristic Search: Best First Search, Hill Climbing, Beam Search, Tabu Search.

Unit-IV **[08]** **Max Marks:15**

Randomized Search: Simulated Annealing, Genetic Algorithms, Ant Colony Optimization.

Unit-V **[08]** **Max Marks:12**

Problem Decomposition: Goal Trees, AO*, Rule Based Systems, Rete Net.

Unit-VI **[06]** **Max Marks:12**

Game Playing: Minimax Algorithm, AlphaBeta Algorithm, SSS*.

Unit-VII **[08]** **Max Marks:14**

Planning and Constraint Satisfaction: Domains, Forward and Backward Search, Goal Stack Planning, Plan Space Planning, Graphplan, Constraint Propagation.

Unit-VIII **[08]** **Max Marks:15**

Logic and Inferences: Propositional Logic, First Order Logic, Soundness and Completeness, Forward and Backward chaining.

References:

1. Deepak Khemani (2013). A First Course in Artificial Intelligence, McGraw Hill Education (India), ISBN 9781259029981
2. Elaine Rich and Kevin Knight (1991). Artificial Intelligence, Tata McGraw Hill, ISBN 13: 9780070087705
3. Stuart Russell and Peter Norvig (2009). Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall, ISBN-13: 978-0-13-604259-4.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS423C.1	Identify problems that are amenable to solution by AI methods.	2
MCS423C.2	Identify appropriate AI methods to solve a given problem.	2
MCS423C.3	Design smart system using different informed search / uninformed search or heuristic approaches.	6

Course Code: **MCS-424** **Compiler Construction** Clock Hours: **30**
Total Marks: **50**

Course Objectives:

To cover the major topics in compiler design with emphasis on solving the problems encountered in designing a compiler regardless of the source language or the target machine.

Unit-I **[02]** Max Marks:**06**

Compiler structure: analysis-synthesis model of compilation, various phases of a compiler, tool-based approach to compiler construction.

Unit-II **[03]** Max Marks:**08**

Lexical analysis: Interface with input, parser and symbol table, token, lexeme and patterns, Difficulties in lexical analysis, Error reporting, Implementation, Regular definition, Transition diagrams, LEX.

Unit-III **[07]** Max Marks:**10**

Syntax analysis: CFGs, ambiguity, associativity, precedence, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing, bottom up parsing, operator precedence grammars, LR parsers (SLR, LALR, LR), YACC.

Syntax directed definitions: inherited and synthesized attributes, dependency graph, evaluation order, bottom up and top down evaluation of attributes, L- and S-attributed definitions.

Type checking: type system, type expressions, structural and name equivalence of types, type conversion, overloaded functions and operators, polymorphic functions.

Unit-IV **[06]** Max Marks:**08**

Run time system: storage organization, activation tree, activation record, parameter passing, symbol table, dynamic storage allocation.

Unit-V **[06]** Max Marks:**08**

Intermediate code generation: Intermediate representations, translation of declarations, assignments, control flow, Boolean expressions and procedure calls. Implementation issues.

Unit-VI **[06]** Max Marks:**10**

Code generation and instruction selection: issues, basic blocks and flow graphs, register allocation, code generation, dag representation of programs, code generation from dags, peep hole optimization, code generator generators, specifications of machine.

References:

1. Aho A.V., R. Sethi and J.D. Ullman. Compiler Principle, Techniques and Tools: Addison Wesley, ISBN 0-321-48681-1.
2. Barret, Couch. Compiler Construction Theory and Practice: Computer Science series, Asian Student Ed, ISBN 978-0574213358
3. Dhamdhere D.M. Compiler Construction Principle and Practice: McMillan India, ISBN 9780333904060
4. Gres D. Compiler Construction for Digital Computer: Wiley, ISBN 047132776X.
5. David Galles (2009). Modern Compiler Design: Pearson Education, ISBN 9788131709412

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS424C.1	Understand the basic structure of compiler, concepts and terminology in programming languages.	2
MCS424C.2	Explain lexical analysis, finite state techniques, scanner generator, parsing, kinds of parsers, designing lexical analyzer, scanner and parsers, principal ideas with intermediate code generation, optimizations.	2

MCS424C.3	Understanding of all concepts is essential to design compiler in general for programming languages.	2

Course Code**: MCS-425** **LAB on Python Programming** Clock Hours**: 30**
Total Marks**: 50**

Course Objectives:

Course Objectives:

The student should be able to

1) develop the Python programs for searching, sorting, with help of fundamental concepts like lists, dictionary.
2) understand the concepts of functions scoping, recursion, list mutability, regular expression in Python programming.
3) learn to define their own classes, methods and modules according to the requirement of the problem and use of exception handling concepts.
4) define regular expression and develop GUI programs using Tkinter.

1. Develop programs to understand the control structures of python
2. Develop programs to learn different types of structures (list, dictionary, tuples) in python
3. Develop programs to learn concept of functions scoping, recursion and list mutability.
4. Develop programs to understand object oriented programming using python.
5. Develop programs for data structure algorithms using python – searching, sorting and hash tables.
6. Develop programs to learn regular expressions using python.
7. Develop programs to learn GUI programming using Tkinter.
8. Demonstrate the concept of exception handling using try/except/else Statement, Unified try/except/finally, try/finally Statement, raise Statement, assert Statement, catch multiple specific exceptions
9. Demonstrate the concept of String-Based Exceptions, Class-Based Exceptions and Nesting Exception handlers.
10. Demonstrate implementation of the Anonymous Function Lambda.
11. Demonstrate implementation Mapping Functions over Sequences.
12. Demonstrate implementation functional programming tools such as filter and reduce
13. Demonstrate the Module Creation, Module usage, Module Namespaces, Reloading Modules, Module Packages, Data Hiding in Modules.
14. Demonstrate Mixed Usage Modes of modules, Changing the Module Search Path, The import as Extension, Relative Import Syntax, Module Design Concepts

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS425C.1	Demonstrate use and working of various data types, control structures, files, exceptional handling etc.	2
MCS425C.2	Create, configure and make use of modules.	6
MCS425C.3	Develop console based and GUI applications (both procedural/object oriented) to solve different problems using python programming.	6

Course Code: **MCS-426** **LAB on DAA** Clock Hours: **30**
Total Marks: **50**

Course Objectives:

Understand and learn

1) To convert the algorithms to code.
2) To measure the complexities at run time.
3) To modify the algorithms for efficiency.
4) To debug and test the programs.
5) To conclude using profile of outcomes.

OS: Windows/Linux, Programming Language: C++/Java/C#

1. Write a program for creating max./min. heap using
   - INSERT
   - ADJUST/HEAPIFY
2. Write a program to implement union and find operation.
3. Write a program to find minimum and maximum form a given array.
4. Write a program for searching element form given array using binary search for n=1000,2000,3000 find exact time of execution.
5. Write a program for sorting given array in ascending/descending order with n=1000,2000,3000 find exact time of execution using
   - Heap sort
   - Merge sort
   - Quick sort
6. Write a program for matrix multiplication using Strassen's matrix multiplication.
7. Write a program to find solution of Knapsack instant.
8. Write a program to find shortest path using single source shortest path.
9. Write a program to find Minimum-Cost Spanning Trees (Prim's & Kruskal's Algorithm).
10. Write a program to find shortest path using all pair path.
11. Write a program to find longest common subsequence.
12. Write a program to implement breadth first and depth first search.
13. Write a program to implement breadth first and depth first traversal.
14. Write a program to find all solutions for 8-queen problem using backtracking.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS426C.1	Construct logic for the algorithms designed using designing techniques.	3
MCS426C.2	Posterior analysis of the algorithms.	4
MCS426C.3	Debug, test and profile the algorithms, modify to improve performance of the algorithms.	4

Course Code: **MCSE-427 A** **Mathematical Foundations of Computer Science** Clock Hours: **60**
Total Marks: **100**

Course Objectives:

1) To build the foundation of computer algorithms using mathematical base.
2) To introduce the concepts of induction to prove certain property is true for non-negative integers.

3) To introduce the concepts of recursive data types, linear recurrences, dived and conquer recurrences and solving recurrences.
4) To introduce the concept of generating functions and applications of principle of inclusion and exclusion.
5) To apply statistical measures on the data and represent it graphically.
6) To relate practical examples to the probability theory and probability distributions to build the foundation for machine learning.
7) To understand stochastic processes and apply Markov chain theory to relate real time problems.
8) To understand Hidden Markov Model and Chapman-Kolmogorov equation for solving problems.

Unit-I **[15]** Max Marks:**25**

Induction and Recursion: Mathematical Induction, Strong Induction and Well Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness, The Towers of Hanoi, Merge Sort, Linear Recurrences, Divide-and-Conquer Recurrences, A Feel for Recurrences

Unit-II **[15]** Max Marks:**20**

Advance Counting Techniques: Recursive Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion-Exclusion, Applications of Inclusion-Exclusion.

Unit-III **[05]** Max Marks:**10**

Statistics: Population and sample, parameters and statistics: definition, types: Descriptive and Inferential, applications, Descriptive Statistics: Mean, median, mode and standard deviation, variance, Graphical statistics

Unit-IV **[15]** Max Marks:**20**

Probability: Making decisions under uncertainty, Classical definition of Probability, Events and their Outcomes, Rules of Probability, Probability axioms, Random variables (discrete and continuous), Joint and Conditional probability, independence and Bayes theorem, Distribution of a random vector, Probability mass function, Probability density function and distribution function. Distributions: Binomial, Poisson, Uniform, Exponential, Normal.

Unit-V **[10]** Max Marks:**15**

Stochastic Processes: Definitions and classifications of Stochastic Processes, discrete and continuous Markov models, Hidden Markov Models, Chapman-Kolmogorov equation

References:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications 6th Ed, Tata McGraw-Hill Publishing Company Limited, New Delhi, 2007 ISBN 10: 0070681880
2. Michael Baron (2014) Probability and Statistics For Computer Scientists Second Edition, CRC press. ISBN: 978-1-4822-1410-9
3. Bhat B.R, Srivenkatramana T and Rao Madhava K.S. (1996): Statistics: A Beginner's Text, Vol. I, New Age International (P) Ltd.
4. Croxton F.E, Cowden D.J and Kelin S (1973): Applied General Statistics, Prentice Hall of India ISBN 10: 0201089947 ISBN 13: 9780201089943
5. Hogg, Robert V. & Craig Allen T. (2008). Introduction to Mathematical Statistics, Pearson Education ISBN 0-02-978990-7
6. Goon A.M., Gupta M.K., Dasgupta. B. (2001), Fundamentals of Statistics, Volume I and II, World Press, Calcutta.

7. Ross, S. (2005). Introduction to Probability Models, (6th Ed. Academic Press). ISBN 978-0-12-375686-2
8. Medhi, J. (1994). Stochastic Processes, (2nd Ed. New Age Publisher) ISBN : 978-93-86286-48-2

Course Outcome:
After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS427C.1	Demonstrate their understanding of and apply methods of mathematics in computer science to subsequent courses in algorithm design and analysis.	3
MCS427C.2	Analyze the behavior of the data, model the data using statistical measures and represent it graphically on paper without using available computerized tools.	4
MCS427C.3	Apply mathematical foundations, probability theory in the modeling and design of computational systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.	3

Course Code: **MCSE-427 B** **Optimization Algorithms** Clock Hours:**60**
Total Marks: 10**0**

Course Objectives:

1) To introduce with the branch of OR and its role in decision making.
2) To list out various types of applications of operations research (OR).
3) To explain Linear Programming Problem (LPP) and practice with techniques to solve various types of LPP (transportation problem, assignment problems, special cases of duality, Integer programming problems)
4) Describe the significance, concept of game theory and algorithms to solve game theory problems.
5) Introduce critical path analysis using network problems.

Unit-I **[05]** Max Marks:**08**

Overview of operations Research: Introduction, Applications, Role of OR in Decision Making, Feasible and optimal Solutions

Unit-II **[15]** Max Marks:**20**

Linear Programming: Special Types: Transportation Problem as LPP, Initial Basic Feasible Solution, North West corner Rule, Lowest Cost Method, Vogel's Approximation Method, MoDi method for optimization, Degeneracy.

Assignment problem, Hungarian Method, Special cases of assignment problem

Unit-III **[18]** Max Marks:**24**

Linear Programming Problems: Introduction, Formulation of Mathematical model of LPP, Standard form of linear programming problems, Solving LPP using Graphical method, Infeasible LPP, Unbounded LPP, Basic feasible solutions, Simplex method for solving LPP, augmentation using Slack and artificial variables, Big M and two phase method, Degeneracy, alternative optima, Interpretation of final Simplex table, Duality: concept, applications and example.

Unit-IV **[06]** Max Marks:**08**

Integer Programming: Introduction, How it differs from LPP, Pure and mixed integer programming problems, Binary IPP, Techniques to solve IPP.

Unit-V **[08]** Max Marks:**15**

Network Models: Definitions, Applications, Representation of a problem in network form, Critical Path Analysis, Resource planning, Giantt Chart.

Unit-VI **[08]** Max Marks:**15**

Game Theory: Concept, Two party zero sum game, Pay off matrix, Pure and mixed strategy games, Rule of Dominance, Subgame method, Brown's Algorithm

References:

1. Hamdy Taha (2010). Operations Research: An Introduction. Pearson Education. ISBN: 978-0132555937
2. L C Jhamb. Quantitative Techniques For Managerial Decisions Vol I, Vol II. Everest Publishing House, ISBN: 8186314628
3. PanneerSelvan R (2006). Operations Research. Prentice Hall of India. ISBN: 978-8120329287

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS428C.1	Differentiate between feasible and optimal solution.	4
MCS428C.2	Apply solving techniques to all types of LPP.	3
MCS428C.3	Apply solving techniques to network problems and game theory problems as well.	3

Semester-III

Course Code: **MCS-511** **Web Application Development Technology** Clock Hours:**60** Total Marks: 10**0**

Course Objectives:

1) .Net Framework and creating ASP.Net web applications using standard .net controls.
2) Use Web Services and develop simple and complex applications using .Net framework
3) Maintain session and controls related information for user used in multi-user web applications

Unit-I **[10]** Max Marks:**15**

Desktop Computing vs. Internet Computing, Internet computing infrastructure, Client side scripting vs. Server Side Scripting technologies, Web Server basics and configuration: IIS, Apache etc., Web site hosting basics, Web Publishing, HTML, introduction to .NET framework, Features of .NET frameowork:CTS,CLS,CLR,.NET technologies, languages'C#.NET,VB.NET, basics of ASP.NET page framework, Visual studio .NET IDE, Page Life Cycle, PostBack, Viewstate, Page directives, ASP.Net page execution cycle, HTTP Pipeline, HTTP Application, HTTP Request, HTTP Response classes, HTTP Modules and HTTP Handlers, State Management, Role of Global.asax, Application configuration using web.config file

Unit-II **[15]** Max Marks:**25**

ASP.NET Control hierarchy, HTML Server Controls, Web Server Controls, User and Server controls, Validation Controls, List bound controls: dropdown lists, list boxes, Repeater, DataList, Data Grid, DataGridView, FormsView controls, Data binding to List Bound Controls, Templating and Styling of ASP.NET server controls

Unit-III **[20]** Max Marks:**25**

Web Page Designing principles, CSS anatomy, Anatomy of Master Pages, nesting master pages, Site map file, Web site Navigation controls, properties:TreeView, Sitemap Path, Menu, Other Navigation methods: Response.Redirect(), Server.Transfer(), Personalization through Profiles, Themes/Skins, Web Site security basics: authentication modes:Windows,Forms,passport, authorization, roles/Membership, access rules, login controls,Web services: working, anatomy, hosting

Unit-IV **[15]** Max Marks:**25**

Database technology: ADO.NET, Anatomy/architecture of ADO.NET, working with Connection, Command, Data Adaptor, DataReader, DataSet, DataTable objects, Editing data in Data Tables, concurrency control. Introduction to MVC, Data Reports

References:

1. Richard Anderson, Brian Francis, Alex Homer, Rob Howard, David Sussman, Karli Watson(2002), Professional ASP.NET 1.0, Special Edition, Wrox Press Ltd., 2002, ISBN 1-861007-0-3-5.
2. Chris Hart, John Kauffman, Dave Sussman, and Chris Ullman(2006), Beginning ASP.NET 2.0, Wiley Publishing, Inc., 2006, ISBN-13: 978-0-7645-8850-1, ISBN-10: 0-7645-8850-8.

3. Beginning ASP.NET 4: in C# and VB, Imar Spaanjaars, Wiley Publishing, Inc 2010., ISBN: 978-0-470-50221-1
4. Bill Evjen, Scott Hanselman, Devin Rader (2008), Professional ASP .NET 3.5 in C# and VB, Wiley Publishing Inc.,2008 ISBN:978-0-470-18757-9.
5. Dino Esposito (2008), Programming Microsoft ASP.NET 3.5, Second Edition, Microsoft Press, 2008, ISBN-10: 0735625271, ISBN-13: 978-0735625273

Auxiliary Resources:

Website URLs

- https://www.asp.net/
- http://asp.net-tutorials.com/

Video Links

- https://www.asp.net/web-forms/videos
- https://www.youtube.com/playlist?list=PL6n9fhu94yhXQS_p1i-HLIftB9Y7Vnxlo&feature=view_all

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS511C.1	Design web applications using ASP.NET.	6
MCS511C.2	Debug and deploy ASP.NET web applications.	3
MCS511C.3	Create database driven ASP.NET web applications and web services.	6

Course Code: M**CS-512** **Machine Learning** *Clock Hours:* **60**
Total Marks: **100**

Course Objectives:

1) The course gives understanding of fundamentals of Machine Learning such as its types, applications and other preliminaries.
2) Course gives fair idea about all important techniques of Machine Learning such as Classification, Regression and Clustering.
3) It also introduces Neural Network model and its applications to Machine Learning and touching Deep Learning.

[08] Max Marks:**10**

Unit-I

Introduction: Basic definitions, types of learning, hypothesis space and inductive bias, evaluation, cross-validation

Unit-II **[08]** Max Marks:**15**

Linear regression, Decision trees, overfitting

Unit-III **[09]** Max Marks:**15**

Instance based learning, Feature reduction, Collaborative filtering based recommendation

Unit-IV **[08]** Max Marks:**15**

Probability and Bayes learning

Unit-V **[09]** Max Marks:**15**

Logistic Regression, Support Vector Machine, Kernel function and Kernel SVM

Unit-VI **[09]** Max Marks:**15**

Neural network: Perceptron, multilayer network, backpropagation, introduction to deep neural network

Unit-VII **[09]** Max Marks:**15**

Clustering: k-means, adaptive hierarchical clustering, Gaussian mixture model

References:

1. Tom Mitchell (1997). Machine Learning. First Edition, McGraw- Hill, ISBN 10: 0070428077 ISBN 13: 9780070428072
2. Ethem Alpaydin (2009). Introduction to Machine Learning, Edition 2, The MIT Press. ISBN 978-0-262-01243-0

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS512C.1	Acquire in-depth knowledge of various facets of Machine Learning methods/techniques and algorithms.	2
MCS512C.2	Envisage practical application of Machine Learning to Business and Research Computational problems.	3
MCS512C.3	Use knowledge of Machine Learning for product/service development.	3

Course Code: M**CS-513** **Software Engineering** *Clock Hours:* **60**

Total Marks: **100**

Course Objectives:

1) Knowledge of basic Software Engineering methods and their application, layered technology, process framework of software process models.
2) Understanding of software requirements, project management, quality of product, data models, functional and behavioral models.
3) Understanding of modularity, coding standards, verification and validation and software testing approaches.

Unit-I **[10]** **Max Marks:10**

Introduction and Process Models: Nature of Software, Software Engineering the process, Software Myths. Process Models: Generic process model, Prescriptive process models, Specialized process models, Unified process, Personal and Team process model, Process Technology, Product and Process. Agility, cost of change, Agile process, Extreme Programming, Agile Process models: Adaptive Software development, Scrum, Dynamic system development model, Crystal, Feature Driven development, Lean Software development, Agile modelling, Agile Unified process. Tool set for Agile process

Unit-II **[10]** **Max Marks:15**

Requirement Analysis and Modelling: Requirement Engineering, Establishing Groundwork, Eliciting Requirements Developing Use cases, Building Requirement model, Negotiating and Validating requirements. Requirement analysis, Scenario based modelling, UML models that supplements use case, Data modelling concepts, class based modelling. Requirement Modelling strategy, Flow oriented modelling, Creating Behaviour model, Pattern for Requirement modelling.

Unit-III **[08]** **Max Marks:15**

Quality Assurance and Change Management: Elements of SQA, SQA Tasks, Goal and Metrics, Formal approaches to SQA, Software Reliability, ISO 9000 Quality standards, SQA Plan. Software Configuration Management, SCM Repository, SCM process

Unit-IV **[11]** **Max Marks:20**

Design Concept: Design process, Design Concept: Abstraction, Architecture, Pattern, Separation of concept, Modularity, Information hiding, Functional independence, Refinement, Aspects, Refactoring. Design Model: Data design element, Architectural design element, Interface design element, Component level design element, Deployment level design element.

Unit-V **[11]** **Max Marks:20**

Architectural and Component Level Design: Software Architectures, Architectural Genres, Architectural styles, Architectural design, Accessing alternatives Architectural design, Architectural mapping using dataflow. Introduction to component, Designing class based component, Conducting component level design, Designing traditional component, component based development.

Unit-VI **[10]** **Max Marks:20**

Software Testing: Strategic approach to software testing, Test strategies for conventional software, Validation Testing, System testing, Software testing fundamentals, Internal and external view of testing, White box testing, Basic path testing, Control structure testing, Black box testing, model based testing, Testing for specialized Environment, Architectures and applications.

References:

1. R. S. Pressman, "Software Engineering: A Practitioner's Approach", McGraw-Hill International Edition, Seventh Edition, ISBN:978-007-126782-3.
2. Pankaj Jalote, "Software Engineering: A Precise Approach", Wiley India Pvt. Limited ISBN: 978-81-265-2311-5.
3. K. K. Aggarwal and Yogesh Singh, "Software Engineering", Third Edition, New Age International Publishers, ISBN:978-81-224-2360-0.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS513C.1	Apply software engineering principles and techniques, to develop, maintain and evaluate large-scale software systems.	3
MCS513C.2	Produce efficient, reliable, robust and cost-effective software solutions.	4
MCS513C.3	Work as an effective member or leader of software engineering teams and manage time, processes and resources effectively by prioritizing competing demands to achieve personal and team goals.	3

Course Code**: MCS-514** **Mobile Application Development** Clock Hours**: 30**
Total Marks**: 50**

***Course* Objectives:**

1) Understand basics of mobile application development and get introduced Android platform and its architecture.
2) To learn activity creation and Android UI designing.
3) To be familiarized with Intent, Broadcast receivers and Internet services, SQLite Database and content providers; to integrate multimedia, camera and Location based services in Android Application.

Unit-I **[05]** **Max Marks:05**

Mobile Application Development: Introduction to handheld devices Palm, Pocket Pc, Symbian OS smart phones, MS windows based smart phones, BlackBerry, iphone etc., features of handheld devices, Device Applications Vs Desktop application, overview of application development platforms OS-Palm OS, Symbian, BlackBerry, Windows CE, OS for iphone, Android, Programming Languages C/C++, JAVA, IDE tools. Comparison of Android

with other Mobile OS. Comparative study of all versions of Android.
[Note: The unit is to be thought with respect to current scenario of Mobile Development hence above contents may change]

Unit-II **[05]** **Max Marks:08**

Hello, Android and Installations: Background, What is android and what isn't, Open Mobile Development Platform, Native Android Applications, Android SDK Features, Introducing the Open Handset Alliance, What Does Android Run On? Why Develop for Android?, Introducing the Development Framework. What Comes in the Box, Developing for Android, Developing for Mobile Devices, Android Development Tools as per current version, Installations, Emulator.

Unit-III **[08]** **Max Marks:10**

Creating Applications, activities and User Interfaces: What Makes an Android Application?, Introducing the Application Manifest. Using the Manifest Editor. The Android Application Life Cycle. Understanding Application Priority and Process States. Externalizing Resources. A Closer Look at Android Activities. Fundamental Android UI Design. Introducing Views. Introducing Layouts and fragments, Using Adapters, Creating New Views.

Unit-IV **[04]** **Max Marks:06**

Intents, Broadcast Receivers, and the Internet: Introducing Intents, Creating Intent Filters and Broadcast Receivers, Using Internet Resources. Introducing Dialogs and Action Bars, Creating and Using Menus.

Unit-V **[03]** **Max Marks:08**

Working in the Background: Introducing Services, Using Background Worker Threads, Using Toast, Introducing Notifications, Using Alarms.

Unit-VI **[05]** **Max Marks:08**

Data Storage, Retrieval, and Sharing: FILES, SAVING STATE, AND PREFERENCES: Saving Application Data, Creating and Saving Shared Preferences, Retrieving Shared Preferences, Introducing the Preference Framework and the Preference Activity, Persisting the Application Instance State, Including Static Files as Resources, Working with the File System.

DATABASES AND CONTENT PROVIDERS: Introducing Android Databases, SQLite Databases, Content Providers, Introducing SQLite, Content Values and Cursors, Working with SQLite Databases, Creating Content Providers.

References:

1. Reto Meier. Professional Android Application Development, Wrox Publications ISBN: 978-0-470-34471-2.
2. Rick Rogers, John Lombardo, Zigurd Mednieks, G. Blake Meike. Android Application Development: Programming with the Google SDK. O'Reilly ISBN 10: 0596521472 / ISBN 13: 9780596521479.

Auxiliary Resources:

a) https://developer.android.com/index.html

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS514C.1	Compare android with other smartphone OS and desktop OS; Able to understand software stack of android OS.	4
MCS514C.2	Understand Activity lifecycle, UI management, use Intent, Broadcast receivers and Internet services.	2

MCS514C.3	Effectively use SQLite Database and content providers, multimedia, camera and Location based services in Android Application.	3

Course Code: M**CS- 515** **LAB on Web Application Development Technology** Clock Hours**: 30** Total Marks: **50**

Course Objectives:

1) Understand the working of Internet, Types of Web Sites/applications, basics of Web hosting and working of IIS web server.
2) Configure and create an ASP.Net application, .net controls based and data driven web application with session and controls used in multi-user web applications
3) Basic hands on the C#.Net/VB.NET programming language.

1. Demonstrate followings in IIS:
   a. Creation of Virtual Directory, Home directory, Home page, hosting of website
2. Demonstrate Page Life Cycle of ASP.NET. Use important page events for your demonstration.
3. Write VB.Net/C# console applications to demonstrate: OO concepts: polymorphism, encapsulation, inheritance, interface inheritance, abstract classes/methods, overloading, overriding, collection classes, properties
4. Demonstrate concept of postback and viewstate using web form server controls of ASP.NET
5. Demonstrate various Web form server controls using sample data entry screen form for registering for a service on website. Also use validation controls to validate input data.
6. Demonstrate DropDown List box, CheckButtonList, RadioButtonList controls.
7. Demonstrate Databinding using Hashtable, ArraryList, DataTable data sources.
8. Demonstrate Repeater control with the help of various templates.
9. Demonstrate paging, sorting, filtering of data in asp:DataGrid/DataGridView.
10. Demonstrate editing process in DataGrid and DataList controls. Make use of necessary templates for proper visual appearance.
11. Demonstrate State Management features of ASP.NET using sample shopping cart application.
12. Create sample website for demonstrating use of Profiles/Themes using skin files.
13. Demonstrate Master Pages and website navigation controls(sitemap path, treeview, menu) using SiteMap file.
14. Demonstrate Properties of website navigation controls.
15. Demonstrate Authorization/Authentication using Login controls and Roles/Membership/AccessRules
16. Demonstrate creation of simple/complex DataReader/DataSet Objects.
17. Demosntrate editing in DataTable objects.
18. Demonstrate Web Service hosting, access in ASP.NET

Course Outcomes:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS515C.1	Develop applications using C#.Net/VB.NET.	3
MCS515C.2	Design web applications using ASP.NET, use ASP.NET controls, debug and deploy ASP.NET web applications.	3
MCS515C.3	Create database driven ASP.NET web applications and web services.	3

Course Code: M**CS- 516** **LAB on Machine Learning** Clock Hours**: 30**
Total Marks: **50**

Course Objectives:

1. Make use of Data sets in implementing the machine learning algorithms
2. Implement various ML algorithms for Classification clustering, regression using Python.
3. Implement the machine learning concepts and algorithms

1. Perform Data Pre-processing, Feature Engineering and Exploratory Data Analysis using Python:
   a) Data Pre-processing: Import Python Libraries (Numpy, Pandas, Matplotlib, Seaborn), Reading Dataset, Data Cleaning/Wrangling, Missing values
   b) Feature Engineering: Data Reduction, Feature Engineering, Creating Features, Encoding and one-hot-encoding, Feature Scaling: Normalization (MinMaxScaler) and Standardization (StandardScaler), Binning
   c) EDA Exploratory Data Analysis: Statistics Summary, Multivariate Analysis, EDA Univariate Analysis, Data Transformation,EDA Bivariate Analysis,EDA Multivariate Analysis, Impute Missing values, Outlier detection and removal
2. Implement dimensionality reduction using:
   a) PCA algorithm
   b) LDA algorithm
3. Implement a program for computing the Bias, Variance and Cross-validation.
4. Implementation of various evaluation metrics using sklearn: Accuracy, Precision, Recall and Confusion Matrix.
5. Implement Simple Linear Regression algorithm using the Gradient Descent Algorithm. (Do not make use of ML libraries like Sklearn)
6. Implement Linear Regression using sklearn library. Use an appropriate data set and calculate the accuracy of your model.
7. Write program to calculate popular attribute selection measures (ASM) like Information Entropy, Information Gain, and Gini Index etc. for decision tree.
8. Using the sklearn library build a decision tree-based classifier (train the classifier using ID3 algorithm). Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
9. Using the sklearn library build a classifier using the k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions. Use the Python ML library classes can be used for this problem.
10. Using the sklearn library build a 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.
11. Using the sklearn library build a logistic regression classifier for the Iris data set stored as a .CSV file. Display the performance of the model in terms of accuracy, precision, recall, F1 Score, AUC and also display the confusion matrix.
12. Write a program to construct a Bayesian network.
13. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets
14. Implement K-Means Clustering on iris dataset using the scikit-learn (sklearn) library
15. Demonstrates the effect of different metrics on the hierarchical clustering with Agglomerative clustering with different metrics
16. Write a python code for Agglomerative clustering, compute the ward linkage using euclidean distance, and visualize it using a dendrogram

Course Outcomes:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS516C.1	Understand the implementation procedures for the machine learning algorithms.	2
MCS516C.2	Design Python programs for various ML algorithms.	6
MCS516C.3	Apply appropriate data sets to the Machine Learning algorithms.	3
MCS516C.4	Identify and apply Machine Learning algorithms to solve real world problems	3

Course Code: **MCSE-517A** **Digital Image Processing (DIP)** *Clock Hours:* **30**
Total Marks: **50**

Course Objectives:

1) The fundamental knowledge and basic technical competence in the field of Computer Graphics and Digital Image Processing.
2) Give in-depth knowledge about 2D and 3D transformation algorithms.
3) Provide awareness about current technologies and issues specific to Digital Image Enhancement, Restoration, Segmentation, Color Image Processing, and Morphological Image Processing.

Unit-I **[04] Max Marks:06**

Introduction to Computer Graphics and Output Primitives: Overview of Computer Graphics, Its Applications and Software, Representation in Graphics, Vector Graphic Display, Raster Graphics Display, I/O Devices, Representing Image, Straight Line, Line Drawing Algorithm, DDA, Bresenham's Line Algorithm, Circle-generating algorithm, Ellipse-generating Algorithm, Polygon Filling Algorithm.

Unit-II **[03] Max Marks:06**

Two-Dimensional Transformation: Matrix and transformation, 2D Basic transformation, Homogeneous coordinates, Translation, Scaling and Rotation of straight line or polygon, Combined Transformation, Rotation about an arbitrary point/line, Reflection and Shearing Transformation, Viewing Transformation, Clipping, Cohen-Sutherland line clipping.

Unit-III **[03] Max Marks:06**

Three-Dimensional Transformation: Introduction, Matrix representation of 3D Transformation, 3D Translation, Scaling, Rotation, Composition of 3D Transformation, Projection, Orthographic, Isometric, Oblique Projection, Perspective Projection, One-Two-Three point perspective Projection.

Unit-IV **[04] Max Marks:06**

Introduction to Digital Image Processing & Applications: Digital Image Processing. Applications of Digital Image Processing, Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Image Sensing and Acquisition. Image Sampling and Quantization.

Unit-V **[05] Max Marks:06**

Image Enhancement: Background, Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods,

Unit-VI **[05] Max Marks:06**

Image Restoration and Color Image Processing: A Model of the Image Degradation/Restoration Process, Noise Models, Restoration in the Presence of Noise Only-Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Function, Inverse Filtering, Minimum Mean Square Error (Wiener) Filtering, Color Fundamentals, Color Models, Pseudocolor

Unit VII **[03] Max Marks: 06**

Morphological Image Processing & Segmentation: Detection of Discontinuities, Edge linking & Boundary Detection, Thresholding, Region based segmentation Laplacian of Gaussian, Derivative of Gaussian, Canny Edge Detection, Morphological operation: Dilation erosion, Opening & Closing, Basic Morphological Algorithm, Image representation schemes.

Unit VIII **[03] Max Marks: 04**

MATLAB Image processing toolbox: Introduction to MATLAB, Matrix Operations, Introduction to Image Processing Tool Box, Image Read & Write, Filters (spatial and frequency domain), Image Restoration and Reconstruction, Morphological Operations, Edge Detection and linking, Segmentation.

References:

1. Amarendra N Sinha, Arun D. Udai, (2008). Computer Graphics, TMH publication ISBN- 13 : 978-0-07-063437-4.
2. D. Hearn and M. Pauline Baker, Computer Graphics (C Version), Pearson Education, 2nd Edition ISBN-13: 978-0135309247
3. D. F. Rogers and J. A. Adams, Mathematical Elements for Computer Graphics, 2nd Edition, McGraw-Hill International Edition, ISBN-13:978-0-07-0486775
4. R.C.Gonzalez & R.E.Woods, Digital Image Processing, Pearson Education, 3rd edition, ISBN. 13:978-0131687288
5. S. Jayaraman Digital Image Processing TMH (McGraw Hill) publication, ISBN-13:978-0-07-0144798
6. Gonzalez, Woods & Steven, Digital Image Processing using MATLAB, Pearson Education, ISBN-13:978-0130085191

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCS517AC.1	Develop scientific and strategic approach to solve complex problems in the domain of Computer Graphics and Digital Image Processing; expose students to MATLAB Image Processing Toolbox.	6
MCS517AC.2	Demonstrate various algorithms for scan conversion and filling of basic primitives objects and their comparative analysis and applied 2-D and 3-D geometric transformations, viewing and clipping on graphical objects.	3
MCS517AC.3	Use the Mathematics for digital image representation, image acquisition, image transformation, image enhancement and restoration.	3

Course Code: **MCSE-517B** **LAB on Digital Image Processing** Clock Hours:**30**
Total Marks: **50**

Course Objectives:

1) The fundamental knowledge and basic technical competence in the field of Computer Graphics and Digital Image Processing.
2) Give an in-depth knowledge about 2D and 3D transformation algorithms.
3) Provide awareness about the current technologies and issues specific to Digital Image Enhancement, Restoration, Segmentation, Color Image Processing, and Morphological Image Processing.

1. Line drawing algorithm (DDA and Bresenham's Line Algorithm)
2. Circle drawing algorithm
3. Ellipse drawing algorithm
4. Polygon filling algorithm
5. Windowing and clipping algorithm (Point, line and polygon clipping)
6. Composite 2-D transformation, (rotation, scaling & reflection)
7. 3-D geometric transformation (rotation, scaling & reflection)
8. Introduction to Image Processing Toolbox
9. Read an 8 bit image and then apply different image enhancement techniques:
   - Brightness improvement
   - Brightness reduction
   - Thresholding
   - Negative of an image
   - Log transformation
   - Power Law transformation.
10. Implement different interpolation techniques using MATLAB/ SciLab
11. Read an image, plot its histogram then do histogram equalization. Comment about the result.
12. Read an image and apply
    - Implement Gray level slicing (intensity level slicing) in to read cameraman image.
    - Read an 8 bit image and to see the effect of each bit on the image.
    - Read an image and to extract 8 different planes i.e. 'bit plane slicing."
13. Implement various Smoothing spatial filter.
14. Read an image and apply
    1. Gaussian 3x3 mask for burring
    2. High pass filter mask with different masks
    3. Laplacian operator with centre value positive and negative
    4. High boost filtering.
15. Write a program to implement various low pass filters and high pass filter in frequency domain.
16. Write a program for erosion and dilation, opening & closing using inbuilt and without inbuilt function.
17. Implement and study the effect of Different Mask (Sobel, Prewitt and Roberts)
18. Implement various noise models and their Histogram
19. Implement inverse filter and wiener filter over image and comment on them

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCSE517B C.1	Develop scientific and strategic approaches to solve complex problems in the domain of Computer Graphics and Digital Image Processing;	6
MCSE517B C.2	Implement various algorithms for scan conversion, filling objects, 2-D and 3-D geometric transformations, viewing and clipping on graphical objects;	5
MCSE517B C.3	Make use of MATLAB and Image Processing Toolbox to implement image transformation, image enhancement in spatial and frequency domain.	3

Course Code: **MCSE-518A** **Natural Language Processing** Clock Hours:**30**
Total Marks: **50**

Course Objectives:

1) The prime objective of this course is to introduce the students to the field of Language Computing and its applications ranging from classical era to modern context.
2) Course also aims to provide understanding of various NLP tasks and NLP abstractions such as Morphological analysis, POS tagging, concept of syntactic parsing, semantic analysis etc.

Course provide knowledge of different approaches/algorithms for carrying out NLP tasks; it also discusses concepts of Language grammar and grammar representation in Computational Linguistics.

Unit-I **[04]** **Max Marks:05**

Introduction to NLP, brief history, NLP applications: Speech to Text(STT), Text to Speech(TTS), Story Understanding, NL Generation, QA system, Machine Translation, Text Summarization, Text classification, Sentiment Analysis, Grammar/Spell Checkers etc., challenges/Open Problems, NLP abstraction levels, Natural Language (NL) Characteristics and NL computing approaches/techniques and steps, NL tasks: Segmentation, Chunking, tagging, NER, Parsing, Word Sense Disambiguation, NL Generation.

Unit-II **[05]** **Max Marks:09**

Text Processing Challenges, Overview of Language Scripts and their representation on Machines using Character Sets, Language, Corpus and Application Dependence issues, Segmentation: word level (Tokenization), Sentence level. Regular Expression and Automata Morphology, Types, Survey of English and Indian Languages Morphology, Morphological parsing FSA and FST, Porter stemmer, Rule based, and Paradigm based Morphology, Human Morphological Processing, Machine Learning approaches.

Unit-III **[05]** **Max Marks:09**

Word Classes ad Part-of-Speech tagging (POS), survey of POS tagsets, Rule based approaches (ENGTOWL), Stochastic approaches(Probabilistic, N-gram and HMM), TBL morphology, unknown word handling, evaluation metrics: Precision/Recall/F-measure, error analysis

Unit-IV **[08]** **Max Marks:11**

NL parsing basics, approaches: Top Down, Bottom Up, Overview of Grammar Formalisms: constituency and dependency school, Grammar notations CFG, LFG, PCFG, LTAG, Feature-Unification, overview of English CFG, Indian Language Parsing in Paninian Karaka Theory, CFG parsing using Earley's and CYK algorithms, Probabilistic parsing.

Unit-V **[08]** **Max Marks:11**

Concepts and issues in NL, Theories and approaches for Semantic Analysis, Meaning Representation, word similarity, Lexical Semantics, word senses and relationships, WordNet (English and IndoWordnet), Word Sense Disambiguation: Lesk Algorithm Walker's algorithm, Coreferences Resolution: Anaphora, Cataphora

References:

1. Indurkhya, N., & Damerau, F. J. (Eds.). (2010). *Handbook of Natural Language Processing, 2nd Edition.* New York: CRC Press Taylor and Francis Group, Boca Raton London, New York. ISBN-10: 1420085921, ISBN-13: 978-1420085921
2. Martin, J. H., & Jurafsky, D.(2013), *Speech and Language Processing,* Pearson Education India; 2 edition, ISBN-10: 9332518416, ISBN-13: 978-9332518414
3. Manning, Christopher and Heinrich, Schutze(1999), Foundations of Statistical Natural Language Processing", MIT Press, ISBN-10: 0262133601, ISBN-13: 978-0262133609.
4. Akshar Bharati, Chaitanya, V., Kulkarni, A., & Sangal, R. (July 1997). *Machine translation in Stages* (Vol. 10 no. 3). Mumbai: NCST, Mumbai.
5. Bharati, A., Chaitanya, V., & Sangal, R. (1995). *Natural Language Processing: A Paninian Perspective,* New Delhi: Prentice Hall of India, ISBN 10: 8120309219, ISBN 13: 9788120309210.
6. Steven Bird, Edward Loper (2016),Natural Language Processing With Python, Ed. 2, O'Reilly Media,ISBN 1491913428, 9781491913420

Auxiliary Resources:

a. Web Links
   1. https://see.stanford.edu/Course/CS224N
   2. https://web.stanford.edu/~jurafsky/NLPCourseraSlides.html

b. Video Links
   1. http://www.nptelvideos.in/2012/11/natural-language-processing.html
   https://www.youtube.com/playlist?list=PL6397E4B26D00A269

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCSE518AC.1	Understand issues and challenges in Natural Language Processing and NLP applications and their relevance in the classical and modern context.	2
MCSE518AC.2	Understand Computational techniques and approaches for solving NLP problems and develop modules for NLP tasks and tools.	2
MCSE518AC.3	Understand various grammar formalisms, which they can apply in different fields of study.	2

Course Code: **MCSE-518B** **LAB on Natural Language Processing** Clock Hours:**30**
Total Marks: **50**

Course Objectives:

1) To understand natural language processing and to learn how to apply basic algorithms in this field.
2) To get acquainted with the basic concepts and algorithmic description of the language levels: morphology, syntax, semantics and pragmatics
3) To design and implement applications based on natural language processing

Assignments:

1. Implement tokenizer which considers a sentence and splits it into words/tokens.
2. Implement a program to find synonyms, antonyms, and hypernyms for given words using WordNet.
3. Write a program to remove stop words from the given paragraph.
4. Write a program to find the stem of the words from the given paragraph.
5. Implement a program that gives frequency distribution that tells the frequency of each vocabulary item in the text.
6. Implement morphological analyser for a given text using regular expressions.
7. Implement a part-of-speech tagger using rule-based or probabilistic approaches.
8. Implement data scraping from a website (e.g., product reviews or news articles) and perform sentiment analysis on the collected text using natural language processing techniques.
9. Implement a context-free grammar parser using the CYK algorithm for a simple grammar. Evaluate the tagger's accuracy and performance on sample sentences.
10. Implement a program for Word Sense Disambiguation using Lesk Algorithm.

Course Outcome:

After completion of this course students shall be able to-

CO No.	CO	Cognitive level
MCSE518BC.1	Understand the fundamental concept of natural language text processing and implement basic commands of text processing using any linguistic tool.	3
MCSE518BC.2	Apply morphological analysis on natural language text.	5
MCSE518BC.3	Analyze syntactic structure of a language using syntax analysis techniques.	6

Semester-IV

Course Code: **MCS-521**	Getting Started with Competitive Programming (NPTEL, Swayam)	Clock Hours: **60** Total Marks: **100**

The school (Admission Committee with School Swayam Coordinator) should initiate registration by informing students about the MS-521 course on the NPTEL (Swayam) portal in advance, preferably in the month of November-December. The course should be selected based on availability. If the course is not available in the January-June period, then a similar (Programming Skills) course should be selected (decided by the Academic Committee) from the available NPTEL courses.

If the Course on NPTEL is 3 Credit Course, then the marks obtained in the course will be converted to 4 Credit Course.

Course Code: **MCS-522**	Object Oriented System Development Using UML, Java and Patterns (NPTEL, Swayam)	Clock Hours: **60** Total Marks: **100**

The school (Admission Committee with School Swayam Coordinator) should initiate registration by informing students about the MS-521 course on the NPTEL (Swayam) portal in advance, preferably in the month of November-December. The course should be selected based on availability. If the course is not available in the January-June period, then a similar (Programming Skills) course should be selected (decided by the Academic Committee) from the available NPTEL courses.

If the Course on NPTEL is 3 Credit Course, then the marks obtained in the course will be converted to 4 Credit Course.

Course Code**: MCS-523** **Full Time Industrial Training** Clock Hours**: 420**
Total Marks**: 300**

Course Objectives:

1) To provide comprehensive learning platform to students where they can enhance their employ ability skills and become job ready along with real corporate exposure.

2) To enhance students' knowledge in a particular technology and to Increase self-confidence of students and helps in finding their own proficiency.

3) To cultivate student's leadership ability and responsibility to perform or execute the given task.

Fourteen Credits shall be awarded to the Industrial Training/Project course, which will commence in the IVth Semester and the final work and report will be completed at the end of IVth Semester of M.Sc. The student is expected to work on a software development/Research project. The project work should have a coding part. Students will have to submit the bound project report in university prescribed format at the end of the semester. Student will be evaluated with continuous internal evaluation as well as he/she will have to appear for External Project Viva-voce and the marks and the credits will be allotted at the end of IVth semester of the M.Sc. Programme.

Course Outcomes:

After completion of this course students will:

1) Handle specialized technology and update themselves with latest changes in technological world with ability to communicate effectively.

2) Be multi-skilled IT professional with good technical knowledge, management, leadership and entrepreneurship skills.

3) Be able to identify, formulate and model problems and find engineering solution based on a systems approach.

***