GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

School of Computing Science and Engineering

Program: M.Sc. (Computer Science)

Scheme: 2019 – 2021

Curriculum

Semester 1									
Sl. No	**Course Code**	**Name of the Course**					**Assessment Pattern**		
			L	**T**	**P**	**C**	**IA**	**MTE**	**ETE**
1	MSCS1110	Data Structure using C	3	0	0	3	20	50	100
2	MSCS1120	Database Management Systems	3	0	0	3	20	50	100
3	MSCS1130	Software Engineering	3	0	0	3	20	50	100
4	MSCS1140	Operating Systems	3	0	0	3	20	50	100
5	MSCS1150	Mathematical Foundations of Computer Science	3	0	0	3	20	50	100
6	MSCS1111	Data Structure using C Lab	0	0	2	1	50	-	50
7	MSCS1121	Database Management Systems Lab	0	0	2	1	50	-	50
8	MSCS1131	Software Engineering Lab	0	0	2	1	50	-	50
9	MSCS1141	Operating Systems Lab - Unix & Shell Program	0	0	2	1	50	-	50
10	MSCS1161	Python Programming Fundamentals Lab	0	0	4	2	50	-	50
		Total	**15**	**0**	**12**	**21**			

Semester II									
Sl No	**Course Code**	**Name of the Course**					**Assessment Pattern**		
			L	**T**	**P**	**C**	**IA**	**MTE**	**ETE**
1	MSCS1210	Computer Architecture	3	0	0	3	20	50	100
2	MSCS1220	OOPs Using Python	3	0	0	3	20	50	100
3	MSCS1230	Design of Analysis and Algorithms	3	0	0	3	20	50	100
4	MSCS1240	JAVA Programming	3	0	0	3	20	50	100
5	MSCS1250	Cloud Infrastructure Services & Virtualization	3	0	0	3	20	50	100
6	MSCS1280	Introduction to Data Science	3	0	0	3	20	50	100
7	MSCS1221	OOPs using Python Lab	0	0	4	2	50	-	50
8	MSCS1241	JAVA Programming Lab	0	0	4	2	50	-	50
9	MSCS1281	Introduction to Data Science Lab using Python	0	0	2	1	50	-	50
10	SLMC5012	English Proficiency and Aptitude Building - 2	0	0	4	2	50	-	50
		Total	**18**	**0**	**14**	**25**			

Semester III									
Sl No	**Course Code**	**Name of the Course**					**Assessment Pattern**		
			L	**T**	**P**	**C**	**IA**	**MTE**	**ETE**
1	MSCS2310	Cyber Security	3	0	0	3	20	50	100
2	MSCS2320	Internet Programming	3	0	0	3	20	50	100
3	MSCS2330	Artificial Intelligence and Machine Learning	3	0	0	3	20	50	100
4	MSCS2340	Computer Networks	3	0	0	3	20	50	100
5	MSCS23**	Discipline Specific Elective - I	3	0	0	3	20	50	100
6	MSCS2311	Cyber Security Lab	0	0	2	1	50	-	50
7	MSCS2321	Internet Programming Lab	0	0	2	1	50	-	50
8	MSCS2331	Artificial Intelligence and Machine Learning Lab	0	0	4	2	50	-	50
9	MSCS2351	Quantitative Aptitude II	0	0	2	1	50	-	50
10	MSCS2361	Communication Skills Lab II	0	0	2	1	50	-	50
11	MSCS2371	Comprehension and Technical Seminar	0	0	2	1	50	-	50

12	MSCS2381	Mini Project	0	0	10	5	50	-	50
		Total	**15**	**0**	**24**	**27**			

Semester IV

Sl No	**Course Code**	**Name of the Course**					**Assessment Pattern**		
			L	**T**	**P**	**C**	**IA**	**MTE**	**ETE**
1	MSCS2410	Major Project	0	0	30	15	**50**		**50**
		Total	**0**	**0**	**30**	**15**			

List of Electives

Sl No	**Course Code**	**Name of the Electives**					**Assessment Pattern**		
			L	**T**	**P**	**C**	**IA**	**MTE**	**ETE**
1	MSCD2310	Opensource Technologies	3	0	0	3	20	50	100
2	MSCD2320	Software Testing	3	0	0	3	20	50	100
3	MSCD2330	Soft Computing	3	0	0	3	20	50	100
4	MSCD2340	Compiler Design	3	0	0	3	20	50	100
5	MSCD2350	Advanced Database Technology	3	0	0	3	20	50	100
6	MSCD2360	Disruptive Technology	3	0	0	3	20	50	100
7	MSCD2370	Internet of Things	3	0	0	3	20	50	100
8	MSCD2380	Big Data Analytics	3	0	0	3	20	50	100
9	MSCD2390	Mobile Application Development	3	0	0	3	20	50	100

Detailed Syllabus

Name of The Course	Data Structure Using C				
Course Code	**MSCS1110**				
Prerequisite	Basic Algorithms and Object Oriented Paradigm.				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		3	0	0	3

Course Objectives:

1. To impart a thorough understanding of linear data structures such as stacks,queues and their applications..
2. To impart a thorough understanding of non-linear data structures such as trees,graphs and their applications.
3. To impart familiarity with various sorting, searching and hashing techniques and their performance comparison.
4. To impart a basic understanding of memory management.

Course Outcomes:

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

CO1	Compare different programming methodologies and define asymptotic notations to analyze performance of algorithms.
CO2	Use appropriate data structures like arrays, linked list, stacks and queues to solve real world problems efficiently.
CO3	Represent and manipulate data using nonlinear data structures like trees and graphs to design algorithms for various applications.
CO4	Illustrate and compare various techniques for searching and sorting.
CO5	Illustrate various hashing techniques..

Text Books:

1. GAV Pai, Data Structures and Algorithms Concepts, Techniques and Applications, Tata McGraw Hill.

Reference Books

2. Jean Paul Tremblay, Paul G. Sorenson, An Introduction to Data Structures with Applications, Tata McGraw Hill, Second Edition.
3. Sahini, “Data Structures, Algorithms and Applications in C++”, McGrawHill, 1998.
4. Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, “Fundamentals of Computer Algorithms”, Second Edition, Universities Press, 2008
5. Robert Sedgewick, PhillipeFlajolet, “An Introduction to the Analysis of Algorithms”, Addison- Wesley Publishing Company, 1996.
6. Alfred V. Aho, John E. Hocroft, Jeffrey D.Ullman, “Data Structures and Algorithms

Course Content

Unit I: Introduction **9 HOURS** Definition, Structure and Properties of algorithms – Development of an algorithm – Data Structures and algorithms – Data Structure definition and classification. Analysis of algorithms: Efficiency of algorithms –Apriori analysis – Asymptotic notations – Time complexity of an algorithm using O notation – Polynomial Vs Exponential algorithms – Average, Best and Worst case complexities –Analyzing recursive programs.
Unit II: Stacks **9 HOURS** Introduction - Stack Operations – Applications – Recursion - Evaluation of Expressions. Queues: Introduction - Operations on Queues – Circular queues – Application of a linear queue. Linked Lists: Introduction – Singly linked lists - Circularly linked lists - Doubly linked lists - Applications –polynomial addition..
Unit III: Binary Trees **9 HOURS**

Introduction – Representation of Trees – Binary Tree Traversals. Binary Search Trees: Introduction – Operations.AVL Trees: Definition -Operations. B-Trees: Introduction – m-way search trees - B trees definition and operations. Graphs: Introduction – Definitions – Representation of Graphs – Graph Traversal - Depth-First and Breadth-First Algorithms - Topological Sorting.

Unit IV: Divide and Conquer **9 HOURS**

General Method – Binary Search – Merge Sort – Quick Sort. Greedy Method: General Method –Knapsack Problem – Minimum Cost Spanning Tree – Single Source Shortest Path.

Unit V: Dynamic Programming **9 HOURS**

General Method – Multistage Graphs – All Pair Shortest Path – Travelling Salesman Problem. Backtracking: General Method – 8-Queens Problem – Sum of Subsets –Hamiltonian Cycles. Branch and Bound: The Method – 0/1 Knapsack Problem –Travelling Salesperson.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Software Engineering				
Course Code	**MSCS1130**				
Prerequisite	Fundamentals of Computers				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		3	0	0	3

Course Objectives

1. To understand Software Engineering Lifecycle Models.
2. To do project management and cost estimation.
3. To gain knowledge of the System Analysis and Design concepts.
4. To understand software testing approaches.
5. To be familiar with DevOps practices.

Course Outcomes

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

CO1	Understand the advantages of various Software Development Lifecycle Models
CO2	Understand the advantages of various Software Development Lifecycle Models.
CO3	Gain knowledge on project management approaches as well as cost and schedule estimation strategies.
CO4	Perform formal analysis on specifications using UML diagrams for analysis and design.
CO5	Able to apply architect and design using architectural styles and design patterns.
CO6	Understand software testing approaches.
CO7	Understand the advantages of DevOps practices.

Text Books

1. Bernd Bruegge, Alan H Dutoit, Object-Oriented Software Engineering, 2nd edition, Pearso Education, 2004.
2. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, 2nd edition, PHI Learning Pvt. Ltd., 2010..

Reference Books

1. Craig Larman, Applying UML and Patterns, 3rd ed, Pearson Education, 2005.
2. Len Bass, Ingo Weber and Liming Zhu, ―DevOps: A Software Architect‘s Perspectivel, Pearson Education, 2016
3. Rajib Mall, Fundamentals of Software Engineering, 3rd edition, PHI Learning Pvt. Ltd., 2009.
4. Stephen Schach, Software Engineering 7th ed, McGraw-Hill, 2007.

Course Content

Unit I: Introduction **9 HOURS** Software engineering concepts – Development activities – Software lifecycle models - Classical waterfall - Iterativewaterfall – Prototyping – Evolutionary - Spiral – Software project management – Project planning – Estimation – Scheduling– Risk management – Software configuration management
Unit II: Software Requirement Specification **9 HOURS** Requirement analysis and specification – Requirements gathering and analysis – Software Requirement Specification –Formal system specification – Finite State Machines – Petrinets – Object modelling using UML – Use case Model – Classdiagrams – Interaction diagrams – Activity diagrams – State chart diagrams – Functional modelling – Data Flow Diagram.
Unit III: Architecture and Design **9 HOURS** Software design – Design process – Design concepts – Coupling – Cohesion – Functional independence – Design patterns –Model-view-controller – Publish-subscribe – Adapter – Command – Strategy – Observer – Proxy – Facade – Architecturalstyles – Layered - Clientserver - Tiered - Pipe and filter - User interface design
Unit IV: Testing **9 HOURS**

Testing – Unit testing – Black box testing– White box testing – Integration and System testing– Regression testing –Debugging - Program analysis – Symbolic execution – Model Checking
Unit V: Devops **9 HOURS** DevOps:Motivation-Cloud as a platform – Operations - Deployment Pipeline: Overall Architecture Building and Testing-Deployment- Case study: Migrating to Micro services.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Data Structure using C Lab				
Course Code	**MSCS1111**				
Prerequisite	C Language				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		0	0	2	1

LIST OF EXPERIMENTS:

1. Array implementation of Stack and Queue ADTs
2. Array implementation of List ADT
3. Linked list implementation of List, Stack and Queue ADTs
4. Applications of List, Stack and Queue ADTs
5. Implementation of Binary Trees and operations of Binary Trees
6. Implementation of Binary Search Trees
7. Implementation of AVL Trees
8. Implementation of Heaps using Priority Queues.
9. Graph representation and Traversal algorithms
10. Applications of Graphs
11. Implementation of searching and sorting algorithms
12. Hashing – any two collision techniques

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	**-**	**50**	**100**

Name of The Course	Database Management Systems Lab				
Course Code	**MSCS1121**				
Prerequisite	Basic Query Language				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		0	0	2	1

LIST OF EXPERIMENTS:

1. Data Definition Commands, Data Manipulation Commands for inserting, deleting, updating and retrieving Tables and Transaction Control statements
2. Database Querying – Simple queries, Nested queries, Sub queries and Joins
3. Views, Sequences, Synonyms
4. Database Programming: Implicit and Explicit Cursors
5. Procedures and Functions
6. Triggers
7. Exception Handling
8. Database Design using ER modeling, normalization and Implementation for any application
9. Database Connectivity with Front End Tools
10. Case Study using real life database applications

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	**-**	**50**	**100**

Name of The Course	Software Engineering Lab				
Course Code	**MSCS1131**				
Prerequisite	Basic Query Language				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		0	0	2	1

SOFTWARE REQUIRED:

Open source Tools: StarUML / UMLGraph / Topcased

Prepare the following documents for each experiment and develop the software using softwareengineering methodology.

1. **Problem Analysis and Project Planning -**Thorough study of the problem –Identify Project scope, Objectives and Infrastructure.
2. **Software Requirement Analysis -** Describe the individual Phases/modules of the projectand Identify deliverables.
3. **Data Modelling -** Use work products – data dictionary, use case diagrams and activitydiagrams, build and test class diagrams, sequence diagrams and add interface to class diagrams.
4. **Software Development and Debugging** – implement the design by coding
5. **Software Testing -** Prepare test plan, perform validation testing, coverage analysis,memory leaks, develop test case hierarchy, Site check and site monitor.

LIST OF EXPERIMENTS:

Academic domain

1. Course Registration System
2. Student marks analysing system

Railway domain

3. Online ticket reservation system
4. Platform assignment system for the trains in a railway station

Medicine domain

5. Expert system to prescribe the medicines for the given symptoms
6. Remote computer monitoring

Finance domain

7. ATM system
8. Stock maintenance

Human Resource management

9. Quiz System
10. E-mail Client system.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	**-**	**50**	**100**

Name of The Course	Operating Systems Lab - Unix & Shell Program				
Course Code	**MSCS1141**				
Prerequisite	Unix				
Corequisite					
Antirequisite					
	L	**T**	**P**	**C**	
	0	0	2	1	

LIST OF EXPERIMENTS:

1. Write shell programs using 'case', 'then' and 'if' & 'else' statements.
2. Write shell programs using while, do-while and for loop statements.
3. Write a program to create a child process using fork(), exec() system calls and use other system calls.
4. Write a program to convert upper case to lower case letters of a given ASCII file.
5. Write a program to program to search the given pattern in a file.
6. Write a program to implementation of Signals in UNIX.
7. Write a program to simulate UNIX commands like ls, grep, cp.
8. Write a program to demonstrate FCFS and SJF process schedules on the given data.
9. Write a program to demonstrate CPU Priority and Round Robin Scheduling on the given burst time and arrival times.
10. Write a program to simulate Inter Process Communication using pipes.
11. Write a program to implementing Producer and Consumer problem using Semaphores.
12. Write a program tosimulate Bankers Algorithm for Dead Lock Avoidance
13. Write a program tosimulate Bankers Algorithm Dead Lock Prevention.
14. Write a program tosimulate Paging Techniques of memory management.
15. Write a program tosimulate FIFO, LRU, LFU Page replacement algorithms.
16. Write a program tosimulate Sequential, Indexed, and Linked file allocation strategies.

SOFTWARE REQUIRED:

- Recommended to use Open Source Software like Fedora, Ubuntu, CentOS, etc...
- Recommended to write programs using C / C++ on Linux systems.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	**-**	**50**	**100**

Name of The Course	Python Programming Fundamentals Lab				
Course Code	MSCS1161				
Prerequisite	Python Language				
Corequisite					
Antirequisite					
		L	T	P	C
		0	0	4	2

S.No	List of Python Program
1	Python program to add two numbers
2	Python Program for factorial of a number
3	Python Program for simple interest
4	Python Program for compound interest
5	Python Program to check Armstrong Number
6	Python Program for Program to find area of a circle
7	Python program to print all Prime numbers in an Interval
8	Python program to check whether a number is Prime or not
9	Python Program for n-th Fibonacci number
10	Python Program for Fibonacci numbers
11	Python Program for How to check if a given number is Fibonacci number?
12	Python Program for n\'th multiple of a number in Fibonacci Series
13	Program to print ASCII Value of a character
14	Python Program for Sum of squares of first n natural numbers
15	Python Program for cube sum of first n natural numbers
16	Python Ways to find length of list
17	Python Ways to check if element exists in list
18	Python Reversing a List
19	Python Cloning or Copying a list
20	Python Count occurrences of an element in a list
21	Python program to find sum of elements in list
22	Python Multiply all numbers in the list
23	Python program to find smallest number in a list
24	Python program to find largest number in a list
25	Python program to find second largest number in a list
26	Python program to find N largest elements from a list
27	Python program to print even numbers in a list
28	Python program to print odd numbers in a List
29	Python program to print all even numbers in a range
30	Python program to print all odd numbers in a range
31	Python program to count Even and Odd numbers in a List
32	Python program to check if a string is palindrome or not
33	Python program to split and join a string
34	Python Program for Binary Search (Recursive and Iterative)
35	Python Program for Linear Search
36	Python Program for Insertion Sort
37	Python Program for Recursive Insertion Sort
38	Python Program for QuickSort
39	Python Program to Reverse a linked list
40	Python Program for Find largest prime factor of a number
41	Python Program for Efficient program to print all prime factors of a given number

42	Python Program for Product of unique prime factors of a number
43	Python Program for Find sum of odd factors of a number
44	Python Program for Check if count of divisors is even or odd
45	Python Program for Find minimum sum of factors of number
46	Python Program for Difference between sums of odd and even digits
47	Python Program for Find sum of even factors of a number
48	Python Program for Check if all digits of a number divide it
49	Python program to convert float decimal to Octal number
50	Python program to convert floating to binary

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	**-**	**50**	**100**

Name of The Course	Computer Architecture				
Course Code	MSCS1210				
Prerequisite	Data Structures				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives:

1. To explore the fundamentals of computer architecture.
2. To understand the concepts of computer architecture.
3. To identify the core concepts in the real scenario.

Course Outcomes

CO1	Understand computer abstractions and technology.
CO2	Understand concepts of arithmetic operations.
CO3	Understand concept of processor and control unit.
CO4	Understand concept of parallelism.
CO5	Identify core concepts of Memory and I/O systems.

Text Book (s)

1. David A. Patterson and John L. Hennessey, "Computer organization and design", Morgan Kauffman / Elsevier, Fifth edition, 2014.
2. V.CarlHamacher, Zvonko G. Varanesic and Safat G. Zaky, "Computer Organisation", VI th edition, Mc Graw-Hill Inc, 2012.
3. William Stallings "Computer Organization and Architecture" , Seventh Edition , Pearson Education, 2006.

Reference Book (s)

1. Vincent P. Heuring, Harry F. Jordan, "Computer System Architecture", Second Edition, Pearson Education, 2005.
2. Govindarajalu, "Computer Architecture and Organization, Design Principles and Applications", first edition, Tata McGraw Hill, New Delhi, 2005.
3. John P. Hayes, "Computer Architecture and Organization", Third Edition, Tata Mc Graw Hill, 1998.
4. http://nptel.ac.in/.

Course Content:

UNIT I OVERVIEW& INSTRUCTIONS **9 HOURS**

Eight ideas – Components of a computer system – Technology – Performance – Power wall – Uniprocessors to multiprocessors; Instructions – operations and operands – representing instructions – Logical operations – control operations – Addressing and addressing modes.

UNIT II ARITHMETIC OPERATIONS **9 HOURS**

ALU – Addition and subtraction – Multiplication – Division – Floating Point operations – Subword parallelism.

UNIT III PROCESSOR AND CONTROL UNIT **9 HOURS**

Basic MIPS implementation – Building datapath – Control Implementation scheme – Pipelining – Pipelined datapath and control – Handling Data hazards & Control hazards – Exceptions.

UNIT IV PARALLELISM **9 HOURS**

Instruction-level-parallelism – Parallel processing challenges – Flynn's classification – Hardware multithreading – Multicore processors.

UNIT V MEMORY AND I/O SYSTEMS **9 HOURS**

Memory hierarchy – Memory technologies – Cache basics – Measuring and improving cache performance – Virtual memory, TLBs – Input/output system, programmed I/O, DMA and interrupts, I/O processors.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Design of Analysis and Algorithms				
Course Code	**MSCS1230**				
Prerequisite	Data Structures				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		3	0	0	3

Course Objectives

1. To know the importance of the complexity of a given algorithm.
2. To study various algorithmic design techniques.
3. To utilize data structures and/or algorithmic design techniques in solving new problems.
4. To know and understand basic computability concepts and the complexity classes P, NP, and NP-Complete.

Course Outcomes

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

CO1	Analyze the complexity of the algorithms and use technique divide and conquer to solve the problems
CO2	Identify feasible solutions for different problems through greedy method and minimize the solutions space and to solve the problems through dynamic programming.
CO3	Solve the problems through graph algorithms.
CO4	Justify that a certain problem is NP-Complete
CO5	Understand and apply linear programming concepts to real time applications

Text Books

1. Ellis Horowitz, SartajSahni and SanguthevarRajasekaran (2008). Fundamentals of Computer Algorithms, 2/e, Universities Press Private Limited, India

Reference Books

1. Ellis Horowitz and SartajSahni (2003). Fundamentals of Data Structures, Gurgaon: Galgotia Publication
2. Robert L Kruse(2008). Data Structures & Program Design, Prentice Hall, New Delhi
3. Tanenbaum A.M.(2008). Data Structures Using C, Prentice Hall of India, New Delhi

Course Content

Unit I: Introduction **9 HOURS**

Introduction: Algorithms – Analysis of algorithms – Best case and worst case complexities, Analysis of some algorithms using simple data structures, amortized time complexity. Binary search trees: Searching – Insertion and deletion of elements – Analysis.

Unit II: Algorithms for Trees **9 HOURS**

AVL trees: Definition – Height – searching – insertion and deletion of elements, AVL rotations – Analysis. Red black trees: Definition – searching – insertion and deletion of elements – algorithms and their time complexities. Splay trees: Definition – Steps in Splaying – Analysis.

Unit III: Search Trees and Graph Algorithms **9 HOURS**

Multi-way search trees: Indexed Sequential Access – m-way search trees – B-Tree – searching, insertion and deletion - B+ trees - Tries Graphs: Definition – representations, Adjacency matrix, packed adjacency list and linked adjacency list, – network representation – Graph search methods, Breadth first Search and Depth first Search

Unit IV: Divide-and-Conquer and Greedy Algorithms **9 HOURS**

Divide and conquer: The General Method – Examples – Finding the Maximum and Minimum -Merge sort - Quick sort - Binary Search. Greedy method: The General Method – Optimal Storage on Tapes – Knapsack Problem – Job Sequencing with Deadlines – Optimal Merge Patterns - Minimum cost spanning Trees – Single Source Shortest Path

Unit V: Greedy Algorithms, Amortized Analysis and Dynamic Programming 9 HOURS

Dynamic programming: The General Method – Multistage Graphs - All pairs shortest path problem –Travelling sales Person problem. Back tracking: The General Method – The Eight Queen Problem – Sum of Subset Problem – Graph Coloring – Hamiltonian Cycles.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Java Programming				
Course Code	MSCS1240				
Prerequisite	Basic knowledge of C++				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives:

1. To understand the concepts of object-oriented, event driven, and concurrent programming paradigms and develop skills in using these paradigms using Java.
2. To build software development skills using java programming for real-world applications.
3. To understand and apply the concepts of classes, packages, interfaces, arraylist, exception handling and file processing.
4. To develop applications using generic programming and event handling.

Course Outcomes

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

CO1	To understand the concept of Object Oriented Programming.
CO2	To understand the concept of Exception handling and I/O.
CO3	To understand the concept of Threading, Generics, Collection framework.
CO4	Apply knowledge for connecting java with data base.
CO5	To understand the concept of java foundation classes.

Text Books

1. C. Thomas Wu, "An introduction to Object-oriented programming with Java", Fourth Edition, Tata McGraw-Hill Publishing company Ltd., 2006.

Reference Books

1. Cay S. Horstmann and Gary Cornell, "Core Java: Volume I – Fundamentals", Eighth Edition, Sun Microsystems Press, 2008.
2. K. Arnold and J. Gosling, "The JAVA programming language", Third edition, Pearson Education, 2000.
3. Timothy Budd, "Understanding Object-oriented programming with Java", Updated Edition, Pearson Education, 2000.

Course Content

UNIT I OBJECT-ORIENTED PROGRAMMING – FUNDAMENTALS AND INHERITANCE 9 HOURS
Review of OOP - Objects and classes in Java – defining classes – methods - access specifiers – static members – constructors – finalize method – Arrays – Strings – Packages-Inheritance – class hierarchy – polymorphism – dynamic binding – final keyword – abstract classes – the Object class – Reflection – interfaces – object cloning – inner classes – wrapping classes
UNIT IIEXCEPTION HANDLING AND I/O STREAMS 9 HOURS
Exceptions – exception hierarchy – throwing and catching exceptions – Stack Trace Elements - assertions –logging Java input and output, Streams, byte streams and character streams, InputStream, OutputStream, Reader, WriterFile, FileInputStream, BufferedInputStream, FileOutputStream, BufferedOutputStream, FileReader, BufferedReader, FileWriter, BufferedWriter,InputStreamReader, OutputStreamWriter, Serialization-Object writing in file and reading
UNIT III CONCURRENT PROGRAMMING - GENERICS AND GENRIC CLASSES 8 HOURS

Multi-threaded programming – interrupting threads – thread states – thread properties – thread synchronization – thread-safe Collections – Executors – synchronizers –Generics and Genric Classes -Collection framework and collection interfaces List, Queue, Set and Map, List classes, Iterator, ListInterator,For-each method for collection and iterators,The equals method and hashcode method, Comparator and hashCode (), Collections Class

UNIT IV JDBC Connectivity **9 HOURS**

Introduction to JDBC API, Types of drivers Statement, Prepared Statement and Callable Statement, ResultSet, Performing insert, update and delete operations, Transaction management - commit and rollback

UNIT V EVENT-DRIVEN PROGRAMMING **8 HOURS**

Describe the JFC Swing technology, Identify the Swing packages, Describe the GUI building blocks: containers, components, and layout managers, Examine top-level, general-purpose, and special-purpose properties of container, Examine components, Examine layout managers

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Cloud Infrastructure Services & Virtualization				
Course Code	**MSCS1250**				
Prerequisite	Basic Knowledge of Database				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		3	0	0	3

Course Objectives

1. To understand the concept of cloud computing.
2. To appreciate the evolution of cloud from the existing technologies.
3. To have knowledge on the various issues in cloud computing.
4. To be familiar with Infrastructure Services & Virtualization in cloud.
5. To appreciate the emergence of cloud as the next generation computing paradigm.

Course Outcomes

On Completion of the course, the students should be able to:

CO1	Articulate the main concepts, key technologies, strengths and limitations of cloud computing.
CO2	Learn the key and enabling technologies that help in the development of cloud.
CO3	Develop the ability to understand and use the architecture of compute and storage cloud, service and delivery models.
CO4	Explain the core issues of cloud computing such as resource management, Infrastructure Services,Virtualization and security.

Text Books:

1. Danielle Ruest, Nelson Ruest, ―Virtualization: A Beginner‟s Guide‖, McGraw-Hill Osborne Media, 2009.

Reference Books

1. Jim Smith, Ravi Nair , "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005
2. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2010.
3. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
4. Tim Mather, Subra Kumaraswamy, and Shahed Latif ,"Cloud Security and Privacy", O‘Reilly Media, Inc.,2009.
5. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing, A Practical Approach", McGraw-Hill Osborne Media, 2009.
6. Tom White, "Hadoop: The Definitive Guide", Yahoo Press, 2012.
7. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, ―Mastering Cloud Computing, Tata Mcgraw Hill, 2013.
8. Toby Velte, Anthony Velte, Robert Elsenpeter, “Cloud Computing – A Practical Approach, Tata Mcgraw Hill, 2009.
9. George Reese, “Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice), O’Reilly, 2009.

Course Content

UNIT I INTRODUCTION **9 HOURS**

Introduction to Cloud Computing – Definition of Cloud – Evolution of Cloud Computing –Underlying Principles of Parallel and Distributed Computing – Cloud Characteristics – Elasticity in Cloud – On-demand Provisioning.

UNIT II CLOUD ENABLING TECHNOLOGIES **9 HOURS**

Service Oriented Architecture – REST and Systems of Systems – Web Services – Publish Subscribe Model – Basics of Virtualization – Types of Virtualization – Implementation Levels of Virtualization – Virtualization Structures – Tools and Mechanisms – Virtualization of CPU –Memory – I/O Devices –Virtualization Support and Disaster Recovery.

UNIT III CLOUD ARCHITECTURE, SERVICES AND STORAGE **9 HOURS**

Layered Cloud Architecture Design – NIST Cloud Computing Reference Architecture – Public, Private and Hybrid Clouds – laaS – PaaS – SaaS – Architectural Design Challenges – Cloud Storage – Storage-as-a-Service – Advantages of Cloud Storage – Cloud Storage Providers – S3.

UNIT IV VIRTUALIZATION **9 HOURS**

Basics of Virtual Machines - Process Virtual Machines – System Virtual Machines –Emulation – Interpretation – Binary Translation - Taxonomy of Virtual Machines. Virtualization –Management Virtualization — Hardware Maximization – Architectures – Virtualization Management – Storage Virtualization – Network Virtualization

UNIT V VIRTUALIZATION INFRASTRUCTURE **7 HOURS**

Comprehensive Analysis – Resource Pool – Testing Environment –Server Virtualization – Virtual Workloads – Provision Virtual Machines – Desktop Virtualization – Application Virtualization - Implementation levels of virtualization – virtualization structure – virtualization of CPU, Memory and I/O devices – virtual clusters and Resource Management – Virtualization for data center automation.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Introduction to Data Science				
Course Code	**MSCS1280**				
Prerequisite	Mathematics, Statistics				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		3	0	0	3

Course Objectives

1. To introduce the linear algebra concepts for data science
2. To introducestatistical techniques for data pre processing in data science
3. To introducedata analytics problem solving framework
4. To introduceR as a programming language for data science
5. To introduce visualization and assessment tools

Course Outcomes

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

CO1	Represent problems using linear algebra
CO2	Apply statistical techniques for data pre processing
CO3	Build predictive model for data science
CO4	Develop R code for Data science problems
CO5	Visualize and evaluate model

Text Books

1. Nina Zumel, John Mount, "Practical Data Science with R", Manning Publication, 2014
2. Mark Gardener, "Beginning R- The Statistical Programming Language", John Wiley & Sons, Inc.,2012
3. Gilbert Strang, "Introduction to linear algebra", Wellesley-Cambridge Press, 2016.

Reference Books

1. Hadley Wickham & Garrett Grolemund, "IR for Data Science", O'Reilly
2. Douglas Montgomery, "Applied Statistics and Probability",John Wiley & Sons, Inc.

Course Content:

Unit I:LINEAR ALGEBRA FOR DATA SCIENCE 8 HOURS Algebraic view - vectors, matrices, product of matrix & vector, rank, null space, solution of over-determined set of equations and pseudo-inverseGeometric view - vectors, distance, projections, eigenvalue decomposition
Unit II: STATISTICS **8 HOURS** Descriptive statistics, notion of probability, distributions, mean, variance, covariance, and covariance matrix, understanding univariate and multivariate normaldistributions, introduction to hypothesis testing, confidence interval for estimates.
Unit III: MODELLING METHODS 8 HOURS Simple linear regression and verifying assumptions used in linear regression,Multivariate linear regression, Classification using logistic regression, Classification usingkNN and k-means clustering.
Unit IV :INTRODUCTION TO R 8 HOURS Reading and getting data into R, ordered and unordered factors, arrays and matrices, list and data frames, reading data from files, probability distribution, statistical models in R, manipulating objects, data distribution
Unit V: VISUALIZATION AND ASSESSMENT 8 HOURS Introduction to graphical analysis, plot() function, displaying multivariate data, matrix plot, assessing importance of different variables, subset selection, model, model assessment.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	OOPs using Python Lab				
Course Code	MSCS1221				
Prerequisite	Python language.				
Corequisite					
Antirequisite					
		L	T	P	C
		0	0	4	2

LIST OF PROGRAMS:

1. Compute the GCD of two numbers.
2. Find the square root of a number (Newton‘s method)
3. Exponentiation (power of a number)
4. Find the maximum of a list of numbers
5. Linear search and Binary search
6. Selection sort, Insertion sort
7. Merge sort
8. First n prime numbers
9. Multiply matrices
10. Programs that take command line arguments (word count)
11. Find the most frequent words in a text read from a file
12. Simulate elliptical orbits in Pygame
13. Simulate bouncing ball using Pygame
14. Create a calculator program
15. Demonstrate use of advanced regular expressions for data validation.
16. Demonstrate the working of ‘id’ and ‘type’ functions
17. To find all prime numbers within a given range.
18. To print ‘n terms of Fibonacci series using iteration.
19. To demonstrate use of slicing in string
20. To add 'ing' at the end of a given string (length should be at least). If the given string already ends with 'ing' then add 'ly' instead. If the string length of the given string is less than 3, leave it unchanged. Sample String : 'abc' Expected Result : 'abcing'
    Sample String : 'string' Expected Result : 'stringly'
    a. To get a string from a given string where all occurrences of its first char have been changed to '$', except the first char itself.
    b. To compute the frequency of the words from the input. The output should output after sorting the key alphanumerically.
    c. Write a program that accepts a comma separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabetically.
21. Write a program that accepts a sequence of whitespace separated words as input and prints the words after removing all duplicate words and sorting them alphanumerically.
22. To demonstrate use of list & related functions
23. To demonstrate use of Dictionary& related functions
24. To demonstrate use of tuple, set& related functions
25. To implement stack using list
26. To implement queue using list
27. To read and write from a file
28. To copy a file
29. To demonstrate working of classes and objects
30. To demonstrate class method & static method
31. To demonstrate constructors 18. To demonstrate inheritance
32. To demonstrate aggregation/composition
33. To create a small GUI application for insert, update and delete in a table using Oracle as backend and front end for creating form

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	JAVA Programming Lab				
Course Code	MSCS1241				
Prerequisite	OOPs.				
Corequisite					
Antirequisite					
		L	T	P	C
		0	0	4	2

LIST OF EXPERIMENTS

1. Develop a Java application to generate Electricity bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading, type of EB connection(i.e domestic or commercial). Compute the bill amount using the following tariff. If the type of the EB connection is domestic, calculate the amount to be paid as follows:
   - First 100 units - Rs. 1 per unit
   - 101-200 units - Rs. 2.50 per unit
   - 201-500 units – Rs. 4 per unit
   - 501and above units - Rs. 6 per unit

   If the type of the EB connection is commercial, calculate the amount to be paid as follows:
   - First 100 units - Rs. 2 per unit
   - 101-200 units - Rs. 4.50 per unit
   - 201-500 units - Rs. 6 per unit
   - 501and above units - Rs. 7 per unit
2. Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa) , time converter (HOURS to minutes, seconds and vice versa) using packages.
3. Develop a java application with Employee class with Emp_name, Emp_id, Address, Mail_id, Mobile_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary.
4. Design a Java interface for ADT Stack. Implement this interface using array. Provide necessary exception handling in both the implementations.
5. Write a program to perform string operations using Array List. Write functions for the following a. Append - add at end b. Insert – add at particular index c. Search d. List all string starts with given letter.
6. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.
7. Write a Java program to implement user defined exception handling.
8. Write a Java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes.
9. Write a java program that implements a multi-threaded application that has three threads. First thread generates a random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
10. Write a java program to find the maximum value from the given type of elements using a generic function.
11. Design a calculator using event-driven programming paradigm of Java with the following options. a) Decimal manipulations b) Scientific manipulations.
12. Develop a mini project for any application using Java concepts.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	Cyber Security				
Course Code	MSCS2310				
Prerequisite	Cryptography and Network Security				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives

1. TolearnthefundamentalsofNumber theory andto and their algorithms.
2. To study and apply Cryptographic techniques.
3. To understand the Cybercrime and methods to provide authentication on cyber.
4. To understand the fundamental concepts of cyber Threats, Attacks and their Prevention.
5. To have an introductory knowledge cyber security policies and practices.

Course Outcomes

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

CO1	Understand the Number theory and different algorithms used for the same.
CO2	Implementation of Cryptographic Techniques.
CO3	Apply the authentication methods and prevent cyber crime
CO4	Apply the procedures and algorithms to stop cyber Attacks and Threats.
CO5	Understand the various cyber security policies and how to apply them.

Text Books:

1. Charles P. P fleeger, Shari Lawerance P fleeger, "Analysing Computer Security", Pearson Education India.

References:

1. V.K.Pachghare, "Cryptography and information Security", PHI Learning Private Limited, Delhi India.
2. Sarika Gupta & Gaurav Gupta, Information Security and Cyber Laws, Khanna Publishing House
3. Anshul Kaushik, Cyber Security, Khanna Publishing House

Course Content

UNIT I Introduction **9 HOURS**

Finite Fields and Number Theory: Modular arithmetic – Euclidian Algorithm – Primality Testing – Fermat's and Euler's theorem –Chinese Reminder theorem – Discrete Logarithms

UNIT II Cryptographic Techniques **9 HOURS**

Symmetric key cryptographic techniques: Introduction to Stream cipher – Block cipher: DES – AES- IDEA. Asymmetric key cryptographic techniques: principles – RSA – ElGamal - Elliptic Curve cryptography – Key distribution and Key exchange protocols.

UNIT III Authentication and Cybercrime **9 HOURS**

Hash functions – Secure Hash Algorithm (SHA) Message Authentication – Message Authentication Code (MAC) – Digital Signature Algorithm: RSA &ElGamal based Classification of cybercrimes – planning of attacks – social engineering: Human based – Computer based – Cyberstalking – Cybercafe and Cybercrimes

UNIT IV Cyber Threats, Attacks and Prevention **9 HOURS**

Phishing – Password cracking – Keyloggers and Spywares – DoS and DDoS attacks – SQL Injection. Identity Theft (ID) : Types of identity theft – Techniques of ID theft.

UNIT V Cyber Security Policies and Practices **9 HOURS**

What security policies are – determining the policy needs – writing security policies – Internet and email security policies – Compliance and Enforcement of policies- Review.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Internet Programming				
Course Code	MSCS2320				
Prerequisite	C,C++, Java, Python				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives

1. To learn the fundamentals of Basic network and SGML.
2. To study the socket programming through Java.
3. To understand the Scripting language.
4. To know about the Dynamic HTML.
5. To have an introductory knowledge about server-side programming.

Course Outcomes

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

CO1	Understand the internet protocol and concept of URL.
CO2	Understand the Socket programming through Java.
CO3	Apply the scripting language to form GUI.
CO4	Implement the concept of dynamic HTML and CSS.
CO5	Apply the knowledge of Servlets and JDBC.

Text Books:

1. Deitel, Deitel and Nieto, "Internet and World Wide Web – How to program", Pearson Education Publishers, 2000.

Reference Books:

1. Elliotte Rusty Harold, "Java Network Programming", O'Reilly Publishers, 2002
2. R. Krishnamoorthy & S. Prabhu, "Internet and Java Programming", New Age International Publishers, 2004.
3. Thomno A. Powell, "The Complete Reference HTML and XHTML", fourth edition, Tata McGraw Hill, 2003.

Course Content:

UNIT I BASIC NETWORK AND WEB CONCEPTS 9 HOURS Internet standards – TCP and UDP protocols – URLs – MIME – CGI – Introduction to SGML.
UNIT II JAVA PROGRAMMING 9 HOURS Java basics – I/O streaming – files – Looking up Internet Address - Socket programming – client/server programs – E-mail client – SMTP - POP3 programs – web page retrieval – protocol handlers – content handlers - applets – image handling - Remote Method Invocation.
UNIT III SCRIPTING LANGUAGES 9 HOURS HTML – forms – frames – tables – web page design - JavaScript introduction – control structures – functions – arrays – objects – simple web applications.
UNIT IV DYNAMIC HTML 9 HOURS Dynamic HTML – introduction – cascading style sheets – object model and collections – event model – filters and transition – data binding – data control – ActiveX control – handling of multimedia data
UNIT V SERVER SIDE PROGRAMMING 9 HOURS Servlets – deployment of simple servlets – web server (Java web server / Tomcat / Web logic) – HTTP GET and POST requests – session tracking – cookies – JDBC – simple web applications – multi-tier applications.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Artificial Intelligence and Machine Learning				
Course Code	MSCS2330				
Prerequisite	Basic knowledge of Intelligence.				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives

1. To learn the fundamentals of Artificial Intelligence.
2. To study the introduction to Machine learning.
3. To study the linear model and Regression.
4. To learn the Logic and algebraic models.
5. To have a knowledge about Probabilistic models.

Course Outcomes

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

CO1	Understand the concept of Artificial Intelligence.
CO2	Learn the machine learning and its Classification.
CO3	Apply Linear models and understand the Regression.
CO4	Implement the Logic based and algebraic based models.
CO5	Apply probabilistic model for problem statement.

Text Books:

1. Deitel, Deitel and Nieto, "Internet and World Wide Web – How to program", Pearson Education Publishers, 2000.

Reference Books:

1. Elliotte Rusty Harold, "Java Network Programming", O'Reilly Publishers, 2002
2. R. Krishnamoorthy & S. Prabhu, "Internet and Java Programming", New Age International Publishers, 2004.
3. Thomno A. Powell, "The Complete Reference HTML and XHTML", fourth edition, Tata McGraw Hill, 2003.

Course Content

Unit I Introduction to Artificial Intelligence 9 HOURS

What is AI?, Early work in AI, AI and related fields, AI problems and Techniques Defining AI problems as a State Space Search: example, Production Systems, Search and Control Strategies, Problem Characteristics, Issues in Design of Search Programs, Additional Problems Generate-and-test, Hill Climbing, Best First Search, Problem Reduction, Constraint Satisfaction, Mean-Ends Analysis.

Unit II Introduction to machine learning and Classification 9 HOURS

Why Machine learning, Examples of Machine Learning Problems, Structure of Learning, Learning versus Designing, Training versus Testing, Characteristics of Machine learning tasks, Predictive and descriptive tasks, Machine learning Models: Geometric Models, Logical Models, Probabilistic Models. Features: Feature types, Feature Construction and Transformation, Feature Selection. Classification: Binary Classification- Assessing Classification performance, Class probability Estimation- Assessing class Probability Estimates, Multiclass Classification.

Unit III Regression and Linear models 9 HOURS

Regression: Assessing performance of Regression- Error measures, over fitting- Catalysts for over fitting, Case study of Polynomial Regression. Theory of Generalization: Effective number of hypothesis, bounding the Growth function, VC Dimensions, Regularization theory. Linear models - Multivariate Linear Regression, Regularized Regression, Using Least Square regression for Classification. Perception, Support Vector Machines, Soft Margin SVM, Obtaining probabilities from linear classifiers, Kernel methods for non Linearity

Unit IV Logic based and algebraic models 9 HOURS

Distance Based Models: Neighbors and Examples, Nearest Neighbors Classification, Rule Based Models: Rule learning for Subgroup discovery, Association rule mining. Tree Based Models: Ranking and Probability estimation Trees, Regression Trees, Clustering Trees.

Unit V Probabilistic models 9 HOURS

Normal Distribution and Its Geometric Interpretations, Discriminative learning with Maximum likelihood, Probabilistic Models with Hidden variables: Estimation-Maximization Methods, Gaussian Mixtures, and Compression based Models.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Computer Networks				
Course Code	MSCS2340				
Prerequisite	Fundamentals of Networks				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives

1. To study the physical layer.
2. To learn data link layer and Media Access.
3. To know about network layer working.
4. To learn the transportation of packets.
5. To have a knowledge about working of application layer.

Course Outcomes

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

CO1	Understand the network and physical layer working.
CO2	Understand the addressing and connection od devices over network.
CO3	Learn the formatting and forwarding of packet and switching techniques.
CO4	Understand the overall working of transport protocol and layer.
CO5	Understand how web works and their protocols.

Text Books:

1. Behrouz A. Forouzan, Data Communications and Networking, Fifth Edition TMH, 2013

Reference Books:

1. Larry L. Peterson, Bruce S. Davie, Computer Networks: A Systems Approach, Fifth Edition, Morgan Kaufmann Publishers Inc., 2012.
2. William Stallings, Data and Computer Communications, Tenth Edition, Pearson Education, 2013.
3. Nader F. Mir, Computer and Communication Networks, Second Edition, Prentice Hall, 2014.
4. Ying-Dar Lin, Ren-Hung Hwang and Fred Baker, Computer Networks: An Open Source Approach, McGraw Hill Publisher, 2011.
5. James F. Kurose, Keith W. Ross, Computer Networking, A Top-Down Approach Featuring the Internet, Sixth Edition, Pearson Education, 2013.
6.

Course Content

UNIT I INTRODUCTION AND PHYSICAL LAYER 9 HOURS

Networks – Network Types – Protocol Layering – TCP/IP Protocol suite – OSI Model – Physical Layer: Performance – Transmission media – Switching – Circuit-switched Networks – Packet Switching.

UNIT II DATA-LINK LAYER & MEDIA ACCESS 9 HOURS

Introduction – Link-Layer Addressing – DLC Services – Data-Link Layer Protocols – HDLC – PPP – Media Access Control – Wired LANs: Ethernet – Wireless LANs – Introduction – IEEE 802.11, Bluetooth – Connecting Devices.

UNIT III NETWORK LAYER 9 HOURS

Network Layer Services – Packet switching – Performance – IPV4 Addresses – Forwarding of IP Packets – Network Layer Protocols: IP, ICMP v4 – Unicast Routing Algorithms – Protocols – Multicasting Basics – IPV6 Addressing – IPV6 Protocol.

UNIT IV TRANSPORT LAYER 9 HOURS

Introduction – Transport Layer Protocols – Services – Port Numbers – User Datagram Protocol – Transmission Control Protocol – SCTP.
UNIT V APPLICATION LAYER 9 HOURS WWW and HTTP – FTP – Email –Telnet –SSH – DNS –SNMP

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Cyber Security Lab				
Course Code	**MSCS2311**				
Prerequisite	Cryptography and Network Security				
Corequisite					
Antirequisite					
		L	**T**	**P**	**C**
		0	0	2	1

LIST OF EXPERIMENTS:

1. Implement the following SUBSTITUTION & TRANSPOSITION TECHNIQUES concepts:
a) Caesar Cipher
b) Playfair Cipher
c) Hill Cipher
d) Vigenere Cipher
e) Rail fence – row & Column Transformation

2. Implement the following algorithms
a) DES
b) RSA Algorithm
c) Diffiee-Hellman
d) MD5
e) SHA-1

3. Implement the Signature Scheme - Digital Signature Standard
4. Demonstrate how to provide secure data storage, secure data transmission and for creating digital signatures (GnuPG)
5. Setup a honey pot and monitor the honeypot on network (KF Sensor)
6. Installation of rootkits and study about the variety of options
7. Perform wireless audit on an access point or a router and decrypt WEP and WPA. (NetStumbler)
8. Demonstrate intrusion detection system (ids) using any tool (snort or any other s/w)

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	**Internet Programming Lab**				
Course Code	**MSCS2321**				
Prerequisite	Any computer Procedural or Object Oriented Languages..				
Corequisite					
Antirequisite					
	L	**T**	**P**	**C**	
	0	0	2	1	

LIST OF EXPERIMENTS

1. Write programs in Java to demonstrate the use of following components Text fields, buttons, Scrollbar, Choice, List and Check box
2. Write Java programs to demonstrate the use of various Layouts like Flow Layout, Border Layout, Grid layout, Grid bag layout and card layout
3. Write programs in Java to create applets incorporating the following features:

- Create a color palette with matrix of buttons
- Set background and foreground of the control text area by selecting a color from color palette.
- In order to select Foreground or background use check box control as radio buttons
- To set background images

4. Write programs in Java to do the following.

- Set the URL of another server.
- Download the homepage of the server.
- Display the contents of home page with date, content type, and Expiration date. Last modified and length of the home page.

5. Write programs in Java using sockets to implement the following:

- HTTP request
- FTP
- SMTP
- POP3

6. Write a program in Java for creating simple chat application with datagram sockets and datagram packets.
7. Write programs in Java using Servlets:

- To invoke servlets from HTML forms
- To invoke servlets from Applets

8. Write programs in Java to create three-tier applications using servlets

- for conducting on-line examination.
- for displaying student mark list. Assume that student information is available in a database which has been stored in a database server.

9. Create a web page with the following using HTML

i) To embed a map in a web page
ii) To fix the hot spots in that map
iii) Show all the related information when the hot spots are clicked.

10. Create a web page with the following.

i) Cascading style sheets.
ii) Embedded style sheets.
iii) Inline style sheets.
iv) Use our college information for the web pages.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	Artificial Intelligence and Machine Learning Lab using Python			
Course Code	**MSCS2331**			
Prerequisite	Python			
Corequisite				
Antirequisite				
	L	**T**	**P**	**C**
	0	0	4	2

LIST OF EXPERIMENTS

1. Implement and demonstratethe FIND-Salgorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithmtooutput a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7. Write a program to construct aBayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	**Quantitative Aptitude II**
Course Code	**MSCS2351**
Prerequisite	Mathematics
Corequisite	
Antirequisite	

L	**T**	**P**	**C**
0	0	2	1

Category	Chapters
GEOMETRY	**Quadrilaterals**
	Triangle and its various kinds of centers
	Congruence and similarity of triangles
	Regular Polygon
	Right Prism & Hemispheres
	Circle and its chords, tangents
	Angles subtended by chords of a circle
	Common tangents to two or more circles
	Right Circular Cone
	Sphere
	Cylinder
	Right Circular Cylinder
	Regular Right Pyramid with Triangular base or Square base
	Rectangular Parallelepiped
MENSURATION	**Two-dimensional (2D) and Three-dimensional (3D) Mensuration**
TRIGONOMETRY	**Degree and Radian Measures**
	Trigonometric Ratios
	Complementary Angles
	Standard Identities
	Height and Distance
DATA INTERPRETATION	**Frequency Polygon**
	Histogram
	Pie-Chart
	Bar Diagram

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	**Communication Skills Lab II**				
Course Code	**MSCS2361**				
Prerequisite	English – Speaking and Listening				
Corequisite					
Antirequisite					
	L	**T**	**P**	**C**	
	0	0	2	1	

Course Objectives:

- To enable learners to develop their communicative competence.
- To facilitate them to hone their soft skills.
- To equip them with employ-ability skills to enhance their prospect of placements.

UNIT I ENGLISH FOR NATIONAL AND INTERNATIONAL EXAMINATIONS AND PLACEMENTS

International English Language Testing System (IELTS) – Test of English as a Foreign Language (TOEFL) – Graduate Record Examination (GRE) – Civil Service (Language related) – Verbal ability.

UNIT II SOFT SKILLS

Motivation – self image – goal setting – managing changes – time management – stress management – leadership traits – team work – career and life planning.

TEACHING METHODS:

1. To be totally learner-centric with minimum teacher intervention as the course revolves around practice.

2. Suitable audio/video samples from Podcast/YouTube to be used for illustrative purposes.

3. Portfolio approach for writing to be followed. Learners are to be encouraged to blog, tweet, text and email employing appropriate language.

4. GD/Interview/Role Play/Debate could be conducted off the laboratory (in a regular classroom) but learners are to be exposed to telephonic interview and video conferencing.

5. Learners are to be assigned to read/write/listen/view materials outside the classroom as well for gaining proficiency and better participation in the class.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	**Comprehension and Technical Seminar**				
Course Code	**MSCS2371**				
Prerequisite	Basic English Knowledge				
Corequisite					
Antirequisite					
	L	**T**	**P**	**C**	
	0	0	2	1	

Course Objectives:

- To encourage the students to comprehend the knowledge acquired from the first Semester to third Semester of Degree Course through periodic exercise.
- To encourage the students to study advanced topics.
- To prepare and present technical reports.
- To encourage the students to use various teaching aids such as overhead projectors, power point presentation and demonstrative models.

Course Outcomes:

- Ability to understand and comprehend any given problem related to computer Science field.

METHOD OF EVALUATION:

- The students will be assessed 100% internally through test with objective type questions on all the subject related topics.
- During the seminar session each student is expected to prepare and present a topic on engineering/ technology, for a duration of about 8 to 10 minutes. In a session of two periods per week, students are expected to present the seminar. Each student is expected to present at least twice during the semester and the student is evaluated based on that.

At the end of the semester, he / she can submit a report on his / her topic of seminar and marks are given based on the report. A Faculty guide is to be allotted and he / she will guide and monitor the progress of the student and maintain attendance also. Evaluation is 100% internal.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Name of The Course	**Mini Project**				
Course Code	**MSCS2381**				
Prerequisite	Must be specialized in Computer Lang, Or design, or Query Lang..				
Corequisite					
Antirequisite					
	L	**T**	**P**	**C**	
	0	0	10	5	

GUIDELINES FOR THE MINI PROJECT:

- Every student is required to carry out Mini Project work under the supervision of a Mentor provided by the MSC Programme Coordinator.
- The mentor shall monitor progress of the student continuously. A candidate is required to present the progress of the Mini Project work during the semester as per the schedule provided by the MSC Programme Coordinator.

OBJECTIVES

- Application of knowledge and techniques learnt in theoretical classes for developing the s/w for real problems.
- Gives an insight into the working of the real organizations/companies.
- Gaining deeper understanding in specific functional areas.
- Helps in exploring career opportunities in their areas of interest.

The course MSC Mini Project is one that involves requirement analysis, feasibility analysis, Database design, coding, testing, implementation and maintenance

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
50	-	50	100

Discipline Specific Elective - I

Name of The Course	Open source Technologies			
Course Code	**MSCD2310**			
Prerequisite	Know Basic of OSS.			
Corequisite				
Antirequisite				
	L	**T**	**P**	**C**
	3	0	0	3

Course Objectives

1. To learn the fundamentals of PHP.
2. To study the advance PHP and MySQL.
3. To study the connection of PHP with AJAX, SEO.
4. To learn PERL.
5. To have a knowledge about advance PERL.

Course Outcomes

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

CO1	Understand the concept and logics of PHP.
CO2	Understand the advance PHP and their connection with Database
CO3	Understand the connection of PHP with AJAX, SEO and CMS.
CO4	Implement the Logic of basic PERL.
CO5	Implement the concept of Advance PERL.

Text Books:

1. Mehdi AcHOURS, Friedhelm, Betz Antony Dovgal, Nuno Lopes, Hannes Magnusson, Georg Richter, Damien Seguy, Jakub VranaAnd several others, (1997-2011), "PHP Manual (Download the manual from PHP official website www.php.net)", The PHP Documentation Group. (For Units I to III)
2. Lee Babin,(2007), "Beginning Ajax with PHP From Novice to Professional", Apres.,(For Units IV to V)

Reference Books:

1. Jaimie Sirovich and Cristian Darie, (2007), "Professional Search Engine Optimization with PHP A Developer's Guide to SEO", Wiley Publishing, Inc., Indianapolis, Indiana.
2. Randal L. Schwartz, Tom Phoenix, brian d foy, "Learning Perl, Fifth Edition.

Course Content:

UNIT I BASIC PHP **9 HOURS** Web Server-Apache-PHP-Data Types-User defined Variables-Constants-Operators-Control Structures-User defined Functions-Directory Functions-File system Functions-Arrays-String Functions-Date and Time Functions-Mathematical Functions-Miscellaneous Functions
UNIT II ADVANCED PHP with MySQL **9 HOURS** Exceptions handling-Error Handling Functions-Predefined Variables-Cookies-Sessions-COM-DOM-CURL-SOAP-Classes and Objects-Mail Function-URL Functions. PHP with MySQL: PHP MySQL Functions-Database driven application.
UNIT III ADVANCED PHP with AJAX, SEO and CMS **9 HOURS** PHP with AJAX: Introducing Ajax-Ajax Basics-PHP and Ajax-Database Driven Ajax. PHP with SEO: Basic SEO-Provocative SE Friendly URLs-Duplicate Content- CMS: Word press Creating an SE-Friendly Blog
UNIT IV BASIC PERL **9 HOURS** Introduction-Scalar Data- Lists and Arrays-Subroutines-Input and Output- Hashes-Regular Expressions-Control Structures-Perl Modules-File Tests
UNIT V ADVANCED PERL **9 HOURS** Directory Operations-Strings and Sorting-Smart Matching-Process Management- Advanced Perl Techniques.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Software Testing				
Course Code	MSCD2320				
Prerequisite	Software Engineering.				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives

1. To learn the introduction of Software Testing.
2. To study the Test Cases and Design.
3. To study the level of testing.
4. To learn how to manage Test Cases.
5. To have a knowledge about Test Automation.

Course Outcomes

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

CO1	Understand the concept of Software Testing.
CO2	Implement the Test case concept.
CO3	Implement the level of Testing.
CO4	Understand the Management of Test Case.
CO5	Implement the Test automation.

Text Books :

1. Srinivasan Desikan and Gopalaswamy Ramesh, "Software Testing – Principles and Practices", Pearson Education, 2006.

Reference Books:

1. Ron Patton, "Software Testing", Second Edition, Sams Publishing, Pearson Education, 2007.

Course Content

UNIT I – INTRODUCTION 9 HOURS

Testing as an Engineering Activity – Testing as a Process – Testing axioms – Basic definitions – Software Testing Principles – The Tester's Role in a Software Development Organization – Origins of Defects – Cost of defects – Defect Classes – The Defect Repository and Test Design – Defect Examples – Developer/Tester Support of Developing a Defect Repository – Defect Prevention strategies.

UNIT II – TEST CASE DESIGN 10 HOURS

Test case Design Strategies – Using Black Bod Approach to Test Case Design – Random Testing – Requirements based testing – Boundary Value Analysis – Equivalence Class Partitioning – Statebased testing – Cause-effect graphing – Compatibility testing – user documentation testing – domain testing – Using White Box Approach to Test design – Test Adequacy Criteria – static testing vs. structural testing – code functional testing – Coverage and Control Flow Graphs – Covering Code Logic – Paths – code complexity testing – Evaluating Test Adequacy Criteria.

UNIT III – LEVELS OF TESTING 10 HOURS

The need for Levers of Testing – Unit Test – Unit Test Planning – Designing the Unit Tests – The Test Harness – Running the Unit tests and Recording results – Integration tests – Designing Integration Tests – Integration Test Planning – Scenario testing – Defect bash elimination System Testing – Acceptance testing – Performance testing – Regression Testing – Internationalization testing – Ad-hoc testing – Alpha, Beta Tests – Testing OO systems – Usability and Accessibility testing – Configuration testing – Compatibility testing – Testing the documentation – Website testing.

UNIT IV – TEST MANAGEMENT 9 HOURS

People and organizational issues in testing – Organization structures for testing teams – testing services – Test Planning – Test Plan Components – Test Plan Attachments – Locating Test Items – test management – test process – Reporting Test Results – The role of three groups in Test Planning and Policy Development – Introducing the test specialist – Skills needed by a test specialist – Building a Testing Group.

UNIT V – TEST AUTOMATION 9 HOURS

Software test automation – skill needed for automation – scope of automation – design and architecture for automation – requirements for a test tool – challenges in automation – Test metrics and measurements – project, progress and productivity metrics.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	SOFT COMPUTING			
Course Code	MSCD2330			
Prerequisite	Computer Algorithms			
Corequisite				
Antirequisite				
	L	T	P	C
	3	0	0	3

COURSE OBJECTIVES

1. To learn the introduction of soft computing.
2. To study the Artificial Neural Network.
3. To understand the Fuzzy Systems.
4. To learn Genetic Algorithms.
5. To Learn the hybrid System.

COURSE OUTCOMES

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

CO1	Understand the Soft computing concept.
CO2	Understand neural Network and algorithms.
CO3	Implement the fuzzy algorithm.
CO4	Understand the genetic algorithm for problem.
CO5	Implement the algorithms to solve hybrid system problem.

Text Books:

1. N.P.Padhy, S.P.Simon, "Soft Computing with MATLAB Programming", Oxford University Press, 2015.

Reference Books:

1. S.N.Sivanandam , S.N.Deepa, "Principles of Soft Computing", Wiley India Pvt. Ltd., 2nd Edition, 2011.
2. S.Rajasekaran, G.A.Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm, Synthesis and Applications ", PHI Learning Pvt. Ltd., 2017.
3. Jyh-Shing Roger Jang, Chuen-Tsai Sun, EijiMizutani, ―Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2002.
4. Kwang H.Lee, ―First course on Fuzzy Theory and Applications, Springer, 2005.
5. George J. Klir and Bo Yuan, ―Fuzzy Sets and Fuzzy Logic-Theory and Applications, Prentice Hall, 1996

COURSE CONTENT

UNIT I INTRODUCTION TO SOFT COMPUTING **10 HOURS**

Introduction-Artificial Intelligence-Artificial Neural Networks-Fuzzy Systems-Genetic Algorithm and Evolutionary Programming-Swarm Intelligent Systems-Classification of ANNs-McCulloch and Pitts Neuron Model-Learning Rules: Hebbian and Delta- Perceptron Network-Adaline Network-Madaline Network.

UNIT II ARTIFICIAL NEURAL NETWORKS 10 HOURS

Back propagation Neural Networks – Kohonen Neural Network -Learning Vector Quantization -Hamming Neural Network – Hopfield Neural Network- Bi-directional Associative Memory -Adaptive Resonance Theory Neural Networks- Support Vector Machines – Spike Neuron Models.

UNIT III FUZZY SYSTEMS 10 HOURS

Introduction to Fuzzy Logic, Classical Sets and Fuzzy Sets – Classical Relations and Fuzzy Relations -Membership Functions - Defuzzification – Fuzzy Arithmetic and Fuzzy Measures -Fuzzy Rule Base and Approximate Reasoning – Introduction to Fuzzy Decision Making.

UNIT IV GENETIC ALGORITHMS 9 HOURS Basic Concepts- Working Principles -Encoding- Fitness Function – Reproduction -Inheritance Operators – Cross Over – Inversion and Deletion -Mutation Operator – Bit-wise Operators -Convergence of Genetic Algorithm.
UNIT V HYBRID SYSTEMS 9 HOURS Hybrid Systems -Neural Networks, Fuzzy Logic and Genetic -GA Based Weight Determination – LR-Type Fuzzy Numbers – Fuzzy Neuron – Fuzzy BP Architecture – Learning in Fuzzy BP- Inference by Fuzzy BP – Fuzzy ArtMap: A Brief Introduction – Soft Computing Tools – GA in Fuzzy Logic Controller Design – Fuzzy Logic Controller.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Compiler Design				
Course Code	MSCD2340				
Prerequisite	Program Structure and Analysis				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

1. Introduction of Compilers.
2. To study the Lexical Analysis of compiler.
3. To understand the syntax Analysis.
4. To learn the translation and run time environment.
5. Introduction to Code optimization.

COURSE OUTCOMES

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

CO1	Understand the Compiler design concept.
CO2	Understand the working of lexical analyzer.
CO3	Understand the working of syntax analyzer.
CO4	Learn the syntax directed translation and concepts of run time environment.
CO5	Learn how to optimize code and generation of code.

Text Books:

1.Alfred V Aho, Monica S. Lam, Ravi Sethi and Jeffrey D Ullman, "Compilers – Principles,Techniques and Tools", 2nd Edition, Pearson Education, 2007.

Reference Books:

1. Randy Allen, Ken Kennedy, "Optimizing Compilers for Modern Architectures: ADependence-based Approach", Morgan Kaufmann Publishers, 2002.
2. Steven S. Muchnick, "Advanced Compiler Design and Implementation", Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint 2003.
3. .Keith D Cooper and Linda Torczon, "Engineering a Compiler", Morgan Kaufmann Publishers Elsevier Science, 2004.
4. Charles N. Fischer, Richard. J. LeBlanc, "Crafting a Compiler with C", Pearson Education, 2008.

COURSE CONTENT:

UNIT I INTRODUCTION TO COMPILERS 5 HOURS

Translators-Compilation and Interpretation-Language processors -The Phases of Compiler-Errors Encountered in Different Phases-The Grouping of Phases-Compiler Construction Tools - Programming Language basics.

UNIT II LEXICAL ANALYSIS 9 HOURS

Need and Role of Lexical Analyzer-Lexical Errors-Expressing Tokens by Regular Expressions- Converting Regular Expression to DFA- Minimization of DFA-Language for Specifying Lexical Analyzers-LEX-Design of Lexical Analyzer for a sample Language.

UNIT III SYNTAX ANALYSIS 10 HOURS

Need and Role of the Parser-Context Free Grammars -Top Down Parsing -General Strategies- Recursive Descent Parser Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser-LR (0)Item- Construction of SLR Parsing Table -Introduction to LALR Parser - Error Handling and Recovery in Syntax Analyzer-YACC-Design of a syntax Analyzer for a Sample Language .

UNIT IV SYNTAX DIRECTED TRANSLATION & RUN TIME ENVIRONMENT 12 HOURS

Syntax directed Definitions-Construction of Syntax Tree-Bottom-up Evaluation of S-Attribute Definitions- Design of predictive translator - Type Systems-Specification of a simple type checker- Equivalence of Type Expressions-Type Conversions.

RUN-TIME ENVIRONMENT: Source Language Issues-Storage Organization-Storage Allocation- Parameter Passing-Symbol Tables-Dynamic Storage Allocation-Storage Allocation in FORTAN.
UNIT V CODE OPTIMIZATION AND CODE GENERATION 9 HOURS Principal Sources of Optimization-DAG- Optimization of Basic Blocks-Global Data Flow Analysis- Efficient Data Flow Algorithms-Issues in Design of a Code Generator - A Simple Code Generator Algorithm.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Advanced Database Technology				
Course Code	MSCD2350				
Prerequisite	DBMS				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives

1. Introduction to parallel databases.
2. To study the intelligent database.
3. Use of XML database.
4. To learn the mobile database.
5. Introduction to multimedia database.

Course Outcomes

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

CO1	Understand the concept of distributed database.
CO2	Understand the working of intelligent database.
CO3	Understand the concept of XML database.
CO4	Implement the use of mobile database.
CO5	Learn how to use multimedia with database.

Text Books:

1. C.J.Date, A.Kannan, S.Swamynathan, ―An Introduction to Database Systems‖ Eighth Edition, Pearson Education, 2006.

Reference Books:

1. Carlo Zaniolo, Stefano Ceri, Christos Faloutsos, Richard T.Snodgrass, V.S.Subrahmanian, Roberto Zicari, ―Advanced Database Systems‖, Morgan Kaufmann publishers,2006.
2. Henry F Korth, Abraham Silberschatz, S. Sudharshan, ―Database System Concepts‖, Sixth Edition, McGraw Hill, 2011.

Course Content:

UNIT I PARALLEL AND DISTRIBUTED DATABASES 9 HOURS Database System Architectures: Centralized and Client-Server Architectures – Server System Architectures – Parallel Systems- Distributed Systems – Parallel Databases: I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Design of Parallel Systems Distributed Database Concepts - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing – Case Studies
UNIT II INTELLIGENT DATABASES 9 HOURS Active Databases: Syntax and Semantics (Starburst, Oracle, DB2)- Taxonomy- Applications- Design Principles for Active Rules- Temporal Databases: Overview of Temporal Databases TSQL2- Deductive Databases-Recursive Queries in SQL-Spatial Databases- Spatial Data Types - Spatial Relationships- Spatial Data Structures-Spatial Access Methods- Spatial DB Implementation.
UNIT III XML DATABASES 9 HOURS XML Databases: XML Data Model – DTD – XML Schema – XML Querying – Web Databases – Open Database Connectivity.
UNIT IV MOBILE DATABASES 9 HOURS Mobile Databases: Location and Handoff Management - Effect of Mobility on Data Management - Location Dependent Data Distribution - Mobile Transaction Models -Concurrency Control - Transaction Commit Protocols
UNIT V MULTIMEDIA DATABASES 9 HOURS Multidimensional Data Structures – Image Databases – Text / Document Databases – Video Databases – Audio Databases – Multimedia Database Design.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Disruptive Technology				
Course Code	MSCD2360				
Prerequisite	Basic Program Processing Techniques.				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

Course Objectives

1. Introduction to Enterprise innovation.
2. To study the services.
3. Study about real time computing.
4. To learn the mobile business and security.
5. Future trends.

Course Outcomes

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

CO1	Understand the concept of Enterprise Innovation.
CO2	Understand services provided for web and peer.
CO3	Apply the real time computing and business management.
CO4	Understand Mobile business and enterprise security.
CO5	Learn what are the future trends.

Text Books:

1. Carol Moran, 'Business Innovation and Disruptive Technology', Pearson Education, Inc. 2003

Reference Books:

1. https://richtopia.com/emerging-technologies/11-disruptive-technology-examples
2. https://www.cognizant.com/whitepapers/the-futur e-of-it-infrastructure-codex2946.pdf

COURSE CONTENT:

Unit I: Introduction &Enterprise Innovation 9 HOURS Introduction - Business and IT Trends - Enterprise Software Trends- Key Emerging Technology Vendors - Key Applications- ITIS Innovations - Industry 4.0
Unit II: Web Services & Peer Services 9 HOURS Web services Market (Technology, Business Strategy) - Peer Services Market (Technology, Business Strategy) – Web 2.0- Motion UI and Progressive Web Apps (PWA) - Hybrid Cloud – Containers (Docker,Warden, Garden)
Unit III: Real-Time Computing & Business Process Management 9 HOURS Real-Time Computing (Technology, Business Strategy) -Prescriptive Analytics - Edge Computing - Business Process Management (Technology- Business Strategy) - Cyber Physical Systems.
Unit IV: Mobile Business & Enterprise Security 9 HOURS Wireless Infrastructure Management- Touch commerce and Personalized Shopping - Location-Based Services- Telematics- Electronic Tagging - Enterprise Security Prevention- Detection- Reaction- Estimating Results
Unit V: Future Trends 9 HOURS AR/VR- Digital currencies and Blockchain Technology- Intelligent Computing AI and Autonomous Robots– Data Science and Deep learning- Computer Vision – Industrial IoT.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Internet of Things			
Course Code	MSCD2370			
Prerequisite	Basic Application Software.			
Corequisite				
Antirequisite				
	L	T	P	C
	3	0	0	3

COURSE OBJECTIVES

1. Introduction to IOT.
2. To study the IOT protocols.
3. Study the things on Web.
4. To learn the integrated techniques.
5. Understand the Applications of IOT.

COURSE OUTCOMES

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

CO1	Understand the concept of IOT.
CO2	Understand the protocols needed for IOT.
CO3	Apply the logic and perform things on web.
CO4	Implement the Integrated techniques over IOT.
CO5	Learn what are the Application of IOT.

Text Books:

1. Honbo Zhou, ―The Internet of Things in the Cloud: A Middleware Perspective‖, CRC Press, 2012.
2. Dieter Uckelmann, Mark Harrison, Florian Michahelles, ―Architecting the Internet of Things‖, Springer, 2011.

Reference Books:

1. David Easley and Jon Kleinberg, ―Networks, Crowds, and Markets: Reasoning About a Highly Connected World‖, Cambridge University Press, 2010.
2. Olivier Hersent, Omar Elloumi and David Boswarthick, ―The Internet of Things: Applications to the Smart Grid and Building Automation‖, Wiley, 2012.
3. Olivier Hersent, David Boswarthick, Omar Elloumi, ―The Internet of Things – Key applications and Protocols‖, Wiley, 2012. Other Reference: 1. http://nptel.ac.in/courses/106105081/

COURSE CONTENT:

Unit – I INTRODUCTION 10 HOURS

Definitions and Functional Requirements –Motivation – Architecture - Web 3.0 View of IoT– Ubiquitous IoT Applications – Four Pillars of IoT – DNA of IoT - The Toolkit Approach for End-user Participation in the Internet of Things. Middleware for IoT: Overview – Communication middleware for IoT –IoT Information Security.

Unit – II IOT PROTOCOLS 8 HOURS

Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Issues with IoT Standardization – Unified Data Standards – Protocols – IEEE 802.15.4 – BACNet Protocol – Modbus – KNX – Zigbee Architecture – Network layer – APS layer – Security.

Unit – III WEB OF THINGS 10 HOURS

Web of Things versus Internet of Things – Two Pillars of the Web – Architecture standardization for WoT– Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals and Business Intelligence. Cloud of Things: Grid/SOA and Cloud Computing–Cloud Middleware – Cloud Standards – Cloud Providers and Systems – Mobile Cloud Computing – The Cloud of Things Architecture.

Unit – IV INTEGRATED 9 HOURS

Integrated Billing Solutions in the Internet of Things Business Models for the Internet of Things - Network Dynamics: Population Models – Information Cascades - Network Effects – Network Dynamics: Structural Models - Cascading Behavior in Networks - The Small-World Phenomenon.

Unit – V APPLICATIONS 8 HOURS

The Role of the Internet of Things for Increased Autonomy and Agility in Collaborative Production Environments - Resource Management in the Internet of Things: Clustering, Synchronisation and Software Agents. Applications - Smart Grid – Electrical Vehicle Charging.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Big Data Analytics				
Course Code	MSCD2380				
Prerequisite	Big Data.				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

COURSE OBJECTIVES:

1. Introduction to BIG DATA.
2. To study the clustering.
3. Study the recommendation system.
4. To learn the graph theory and stream.
5. Understand the NOSQL data management.

COURSE OUTCOMES

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

CO1	Understand the concept BIG DATA.
CO2	Apply the clustering methods and classify.
CO3	Apply the logic to create recommendation system.
CO4	Understand the graph theory and stream memory concept.
CO5	Understand the big data Visualization through NOSQL Data.

Text Books:

1. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012.

Reference Books:

1. David Loshin, "Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph", 2013. 3. EMC Education Services, "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", Wiley publishers, 2015.
2. Bart Baesens, "Analytics in a Big Data World: The Essential Guide to Data Science and its Applications", Wiley Publishers, 2015.
3. Dietmar Jannach and Markus Zanker, "Recommender Systems: An Introduction", Cambridge University Press, 2010.
4. Kim H. Pries and Robert Dunnigan, "Big Data Analytics: A Practical Guide for Managers " CRC Press, 2015.
5. Jimmy Lin and Chris Dyer, "Data-Intensive Text Processing with MapReduce", Synthesis Lectures on Human Language Technologies, Vol. 3, No. 1, Pages 1-177, Morgan Claypool publishers, 2010.

COURSE CONTENT:

UNIT I INTRODUCTION TO BIG DATA 9 HOURS

Evolution of Big data - Best Practices for Big data Analytics - Big data characteristics - Validating - The Promotion of the Value of Big Data - Big Data Use Cases- Characteristics of Big Data Applications - Perception and Quantification of Value -Understanding Big Data Storage - A General Overview of High-Performance Architecture - HDFS - MapReduce and YARN - Map Reduce Programming Model.

UNIT II CLUSTERING AND CLASSIFICATION 9 HOURS

Advanced Analytical Theory and Methods: Overview of Clustering - K-means - Use Cases - Overview of the Method - Determining the Number of Clusters - Diagnostics - Reasons to Choose and Cautions .- Classification: Decision Trees - Overview of a Decision Tree - The General Algorithm - Decision Tree Algorithms - Evaluating a Decision Tree - Decision Trees in R - Naïve Bayes - Bayes‘ Theorem - Naïve Bayes Classifier.

UNIT III ASSOCIATION AND RECOMMENDATION SYSTEM **9** HOURS

Advanced Analytical Theory and Methods: Association Rules - Overview - Apriori Algorithm - Evaluation of Candidate Rules - Applications of Association Rules - Finding Association& finding similarity - Recommendation System: Collaborative Recommendation- Content Based Recommendation - Knowledge Based Recommendation- Hybrid Recommendation Approaches.

UNIT IV GRAPH MEMORY AND STREAM MEMORY **9 HOURS**

Using Graph Analytics for Big Data: Graph Analytics - The Graph Model - Representation as Triples - Graphs and Network Organization - Choosing Graph Analytics - Graph Analytics Use Cases - Graph Analytics Algorithms and Solution Approaches - Technical Complexity of Analyzing Graphs- Features of a Graph Analytics Platform - Considerations: Dedicated Appliances for Graph - Introduction to Streams Concepts – Stream Data Model and Architecture - Stream Computing, Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating moments – Counting oneness in a Window – Decaying Window – Real time Analytics Platform(RTAP) applications - Case Studies - Real Time Sentiment Analysis, Stock Market Predictions.

UNIT V NOSQL DATA MANAGEMENT FOR BIG DATA AND VISUALIZATION 9 HOURS

NoSQL Databases : Schema-less Models‖: Increasing Flexibility for Data Manipulation-Key Value Stores- Document Stores - Tabular Stores - Object Data Stores - Graph Databases Hive - Sharding –- Hbase – Analyzing big data with twitter - Big data for E-Commerce Big data for blogs - Review of Basic Data Analytic Methods using R.

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100

Name of The Course	Mobile Application Development				
Course Code	MSCD239				
Prerequisite	HTML, XML, Java or Python Language.				
Corequisite					
Antirequisite					
		L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

1. Understand the wireless communication.
2. To study mobile short-range networks.
3. Study the mobile IP and transport Layer.
4. To learn the mobile App development.
5. Understand the concept of Android.

COURSE OUTCOMES

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

CO1	Understand the concept of wireless communication and Architecture.
CO2	Apply the concept to maintain wireless short-range network in mobile.
CO3	Understand how IP layer and Transport Layer works on Mobile.
CO4	Implement the Mobile application techniques through android.
CO5	Implement the Mobile application techniques and storage of data in it.

Text Books:

1. Asoke K Talukder, Hasan Ahmed,Roopa R Yavagal "Mobile Computing", Tata McGraw Hill Pub ,2nd Edition Aug – 2010.

Reference Books:

1. Barry A. Burd ,"Android Application Development For Dummies All in One", Wiley, 2015.
2. Ed Burnette,"Hello, Android: Introducing Google"s Mobile Development Platform" third edition" Pragmatic Programmers,2012.
3. Jochen Schillar "Mobile Communications" Pearson Education second Edition.
4. Jerome(J.F) DiMarzio "Android A programmer"s Guide" Tata McGraw-Hill 2010 Edition.
5. Maritn Sauter, ―From GSM to LTE: An Introduction to Mobile Networks and Mobile Broadband‖, John Wiley and Sons, 2011 .
6. Raj Kamal "Mobile Computing" Oxford Higher Education, Second Edition, 2012.
7. RetoMeier,Professional Android 2 Application Development, Wrox"s Programmer to Programmer series.

COURSE CONTENT:

UNIT I WIRELESS COMMUNICATION FUNDAMENTALS, ARCHITECTURE 9 HOURS

Frequency Spectrum- Multiplexing- Spread spectrum-GSM vs CDMA - -Comparison of 2G , 3 G, 4G - GSM Architecture-Entities-Call Routing- Address and identifiers- GSM Protocol architecture-Mobility Management-Frequency Allocation- Security –GPRS Architecture (entity and Protocol).

UNIT II MOBILE WIRELESS SHORT RANGE NETWORKS 9 HOURS

Introduction-WLAN Equipment-WLAN Topologies-WLAN Technologies-IEEE 802.11 Architecture-WLAN MAC-Security of WLAN, Power Management-Standards- WAP Architecture- Bluetooth enabled Devices Network-Layers in Bluetooth Protocol-Security in Bluetooth- IrDA- ZigBee.

UNIT III MOBILE IP NETWORK LAYER, TRANSPORT LAYER 9 HOURS

IP and Mobile IP Network Layer- Packet delivery and Handover Management-Location Management-Registration- Tunneling and Encapsulation-Route Optimization- Mobile Transport Layer-Conventional TCP/IP Transport Layer Protocol-Indirect, Snooping, Mobile TCP.

UNIT IV MOBILE APPLICATION DEVELOPMENT USING ANDROID 9 HOURS Mobile Applications Development - Understanding the Android Software Stack – Android Application Architecture –The Android Application Life Cycle – The Activity Life CycleCreating Android Activity -Views-Layout -Creating User Interfaces with basic views- linking activities with Intents.
UNIT V MOBILE APPLICATION DEVELOPMENT USING ANDROID 9 HOURS Services-Broadcast Receivers – Adapters – Data Storage, Retrieval and Sharing.-Location based services-Development of simple mobile applications .

Continuous Assessment Pattern

Internal Assessment (IA)	Mid Term Exam (MTE)	End Term Exam (ETE)	Total Marks
20	30	50	100