

**Gayatri Vidya Parishad College for Degree and PG Courses (AUTONOMOUS)**

**Department of Computer Science and Applications**

**MASTER OF SCIENCE IN COMPUTER SCIENCE**

*Course Structure and Scheme of Examination w.e.f 2020-21*

**I Semester**

Subject code	Title of the Paper	Periods / Week		Max Marks		Total	Credits
		Theory	Lab	Ext.	Int.		
MSCS1.1.1	Advanced Data Structures	4	-	70	30	100	4
MSCS1.1.2	Discrete Mathematical Structures	4	-	70	30	100	4
MSCS1.1.3	Computer Organization	4	-	70	30	100	4
MSCS1.1.4	Object Oriented Programming Using JAVA	4	-	70	30	100	4
MSCS1.1.5	Operating Systems	4	-	70	30	100	4
MSCS1.1.6	Advanced Data Structures Using JAVA Lab	-	3	50	50	100	1.5
MSCS1.1.7	Computer Organization Lab	-	3	50	50	100	1.5
MSCS1.1.8	Operating Systems Lab	-	3	50	50	100	1.5
<b>Total</b>		<b>20</b>	<b>9</b>	<b>500</b>	<b>300</b>	<b>800</b>	<b>24.5</b>

**II Semester**

Subject code	Title of the Paper	Periods / Week		Max Marks		Total	Credits
		Theory	Lab	Ext.	Int.		
MSCS1.2.1	Web Programming	4	-	70	30	100	4
MSCS1.2.2	Data Base Management Systems	4	-	70	30	100	4
MSCS1.2.3	Artificial Intelligence	4	-	70	30	100	4
MSCS1.2.4	Data Communications and Computer Networks	4	-	70	30	100	4
MSCS1.2.5	Formal Languages and Automata Theory	4	-	70	30	100	4
MSCS1.2.6	<b>Elective – I:</b> (i) Image Processing (ii) Information Retrieval System (iii) Embedded Systems	4	-	70	30	100	4
MSCS1.2.7	Web Programming Lab	-	3	50	50	100	1.5
MSCS1.2.8	Data Base Management Systems Lab	-	3	50	50	100	1.5
MSCS1.2.9	Computer Networks Lab	-	3	50	50	100	1.5
<b>Total</b>		<b>24</b>	<b>9</b>	<b>570</b>	<b>330</b>	<b>900</b>	<b>28.5</b>

### III Semester

Subject code	Title of the Paper	Periods / Week		Max Marks		Total	Credits
		Theory	Lab	Ext.	Int.		
MSCS2.1.1	Object Oriented Analysis and Design with UML	4	-	70	30	100	4
MSCS2.1.2	Cyber Security and Forensics	4	-	70	30	100	4
MSCS2.1.3	Data Warehousing and Data Mining	4	-	70	30	100	4
MSCS2.1.4	Python Programming	4	-	70	30	100	4
MSCS2.1.5	Internet of Things	4	-	70	30	100	4
MSCS2.1.6	<b>Elective – II:</b> (i)Data Science (ii)Machine Learning (iii)Cloud Computing	4	-	70	30	100	4
MSCS2.1.7	Python Programming Lab	-	3	50	50	100	1.5
MSCS2.1.8	Internet of Things Lab	-	3	50	50	100	1.5
<b>Total</b>		<b>24</b>	<b>6</b>	<b>520</b>	<b>280</b>	<b>800</b>	<b>27</b>

### IV Semester

Subject code	Title of the Paper	Periods / Week		Max Marks		Total	Credits
		Theory	Lab	Ext.	Int.		
MSCS2.2.1	Open Elective	2	-	70	30	100	2
MSCS2.2.2	Seminar	-	-	-	50	50	2
MSCS2.2.3	Project	-	-	50	50	100	12
<b>Total</b>		<b>2</b>	<b>-</b>	<b>120</b>	<b>130</b>	<b>250</b>	<b>16</b>

Total (Complete Course)	Periods / Week		Max Marks		Total	Credits
	Theory	Lab	Ext.	Int.		
	<b>70</b>	<b>24</b>	<b>1710</b>	<b>1040</b>	<b>2750</b>	<b>96</b>

**Syllabi**

**With effect from 2020-21 admitted batch**

**Chairman**

**Board of Studies**

**(2020-21)**

**Department of Computer Science and Applications**

**Gayatri Vidya Parishad College for Degree and PG Courses(A)**

**Affiliated to Andhra University**

**Visakhapatnam**

# MASTER OF SCIENCE IN COMPUTER SCIENCE

## Syllabi

With effect from 2020-21 admitted batch

### I YEAR I SEMESTER

Subject code	Title of the Paper	Periods / Week		Max Marks		Total	Credits
		Theory	Lab	Ext.	Int.		
MSCS1.1.1	Advanced Data Structures	4	-	70	30	100	4
MSCS1.1.2	Discrete Mathematical Structures	4	-	70	30	100	4
MSCS1.1.3	Computer Organization	4	-	70	30	100	4
MSCS1.1.4	Object Oriented Programming Using JAVA	4	-	70	30	100	4
MSCS1.1.5	Operating Systems	4	-	70	30	100	4
MSCS1.1.6	Advanced Data Structures Using JAVA Lab	-	3	50	50	100	1.5
MSCS1.1.7	Computer Organization Lab	-	3	50	50	100	1.5
MSCS1.1.8	Operating Systems Lab	-	3	50	50	100	1.5
<b>Total</b>		<b>20</b>	<b>9</b>	<b>500</b>	<b>300</b>	<b>800</b>	<b>24.5</b>

**ADVANCED DATA STRUCTURES**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

1. To discuss about stacks and queues using arrays and linked lists.
2. To develop programs for searching and sorting algorithms.
3. To explain the concepts of various trees.
4. To implement programs using graphs.

**SYLLABUS****UNIT I:**

**Introduction to Data Structures:** Abstract Data Types, Arrays, List ADT, Stack ADT and operations, implementation of Stacks using Arrays. Applications of Stacks: Infix to Postfix conversion and postfix evaluation, Recursion.

**UNIT II:**

**Queue ADT:** Operation on queues, implementation of queues using arrays, circular queues and implementation. Single Linked Lists: single linked lists and operations, implementation of single linked lists, implementation of stacks and queues using single linked lists.

**UNIT III:**

**Doubly linked lists:** Operations, implementation of doubly linked lists, dequeues. Trees: Terminology, basic properties and representation, Binary trees, traversals of a binary tree - Binary Search Trees, AVL Trees, B-Trees, B+ Trees, B\* Trees.

**UNIT IV:**

**Graphs:** Definition and representation of graphs, Graph traversal techniques, Minimal spanning trees-Prims and Kruskals algorithms, Dijkstraw's shortest path Algorithm. Hashing: Common Hash functions, Collision Resolution techniques, Dictionary ADT.

**UNIT V:**

**Sorting:** General Background, Insertion Sort, Selection Sort, Bubble Sort, Quick Sort, Merge Sort, Heapsort. **Searching:** Linear Search, Binary Search, Text Processing: Sting Operations, Brute-Force Pattern Matching, The Boyer-Moore Algorithm, The Knuth-Morris-Pratt Algorithm, The Huffman Coding Algorithm, The Longest Common Subsequence Problem (LCS), Applying Dynamic Programming to the LCS Problem.

**Outcomes:**

1. Acquires knowledge on implementation of Stacks and their applications.
2. Develop knowledge on queues and linked lists.
3. Ability to implement various tree data structures and their properties.
4. Acquires knowledge on graphs and its applications.
5. Familiarizes with various sorting and searching techniques.

**Text Book:**

Data Structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Edition.

**Reference Books:**

- 1.Data Structures and Algorithms in Java, Mitchell Waite, Signature Series.
- 2.Data Structures and Algorithms in C++, 2nd edition, A. Drozdek, Thomson.
- 3.The complete Reference Java2 5th Edition by Patrick Naughton and Herbert schildt. TMH

**DISCRETE MATHEMATICAL STRUCTURES**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>
<b>Course Objectives:</b>		
<ol style="list-style-type: none"> <li>1. Simplify and evaluate basic logic statements including compound statements, implications, inverses, converses, and contrapositives using truth tables and the properties of logic.</li> <li>2. Solve problems using Set Laws, Operations and properties of relations, Functions and Types of functions and Recursive Functions.</li> <li>3. Solve counting problems by applying elementary counting techniques using the product and sum rules, permutations, combinations, the pigeon-hole principle, and binomial expansion.</li> <li>4. Definition and identify different types of Graphs, Trees and Minimal spanning trees and Tree traversal Algorithms.</li> </ol>		
<b>SYLLABUS</b>		
<b>UNIT- I:</b>		
<b>Mathematical Logic:</b> Statements -connectives and Truth Tables-Tautology and contradiction - Logical Equivalence, Laws of Logic- Duality-Connectives NAND and NOR forms-Converse – Inverse and Contrapositive- Logical -Rules of Inference- Open Statements-Quantifiers.		
<b>UNIT- II:</b>		
<b>Sets and Relations:</b> Sets and Subsets-Operations on Sets and Laws of Set Theory- Cartesian Product of Sets-Relations-Operations on Relations-Properties of Relations-Equivalence Relations-Partial Orders-External elements in Posets.		
<b>UNIT- III:</b>		
<b>Functions and Combinatory:</b> Functions-Types of Functions-The Pigeonhole Principle-Recursive Functions-Definition-Fibonacci Series-Towers of Hanoi-The Rules of Sum and Product-Permutations-Combinations-Binomial and Multinomial Theorems-Combinations with Repetitions-The Principle of Inclusion-Exclusion.		
<b>UNIT- IV:</b>		
<b>Graph Theory-I:</b> Directed Graphs-Graphs-Isomorphism-Sub Graphs-Operations on Graphs-Walks and their classification-Connected and Disconnected Graphs-Euler circuits and Euler trails-Hamiltonian cycles and Hamiltonian paths.		
<b>UNIT- V:</b>		
<b>Graph Theory-II:</b> Planar and non-planar graphs-Graph Coloring-Map Coloring-Trees and their basic properties-Rooted Trees-Spanning Trees-Minimal Spanning Tree.		
<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. Ability to apply the rules and laws of propositional logic on statements.</li> <li>2. Understands the basic principles and operations on sets.</li> <li>3. Attains capability to solve recursive functions and permutations and combinations.</li> <li>4. Ability to understand graph theory and its applications.</li> <li>5. Obtains knowledge in applications of trees.</li> </ol>		
<b>Text Book:</b>		
Mathematical Foundations of Computer Science-3 <sup>rd</sup> Edition-Dr. D.S.C.-Prism Books Pvt.Ltd.		

<b>Reference Book:</b>		
Discrete and Combinatorial Mathematics-An Applied Introduction-5th Edition –Ralph. P. Grimaldi. Pearson Education.		

## COMPUTER ORGANIZATION

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

### *Course Objectives:*

1. To study about structure and functional components of a computer.
2. Understanding the hierarchical organization of a computer system which consists of instruction set of commands.
3. Learn about the architecture of a computer from a programming view.
4. To design a balance system that minimizes performance and utilization of all elements.

### SYLLABUS

#### UNIT-I:

**Data Representation:** Data Types, Complements, Fixed-Point Representation, Floating-Point Representation, Other Binary Codes, Error Detection Codes. (Chapter 3, Text 1)

**Computer Arithmetic:** Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating Point Arithmetic Operations, Decimal Arithmetic Unit- Operations. (Chapter 10, Text 1)

#### UNIT-II:

**Digital Logic Fundamentals:** Boolean Algebra, Basic Combinatorial Logic, More Complex Combinatorial Components, Combinatorial Circuit Designs, Basic Sequential Components, More Complex Sequential Components, Programmable Logic Devices. (Chapter 1, Text 2)

**Instruction Set Architectures:** Levels of Programming Languages, Assembly Language Instructions, Instruction Set Architecture Design, A Relatively Simple Instruction Set Architecture, the 8085 Microprocessor Instruction Set Architecture. (Chapter 3, Text 2)

#### UNIT-III:

**Introduction to Computer Organization:** Basic Computer Organization, CPU organization, Memory Subsystem Organization and Interfacing, I/O Subsystem Organization and Interfacing, A Relatively Simple Computer, An 8085-based Computer. (Chapter 4, Text 2)

**Register Transfer Languages:** Micro-operations and Register Transfer Language, Using RTL to Specify Digital Systems, More Complex Digital Systems and RTL, VHDL-VHSIC Hardware Description Language. (Chapter 5, Text 2)

#### UNIT-IV:

**CPU Design:** Specifying a CPU, Design a Very Simple CPU, Implementation of a Very Simple CPU Internal Architecture of the 8085 Microprocessor. (Chapter 6, Text 2)

**Micro-sequence Control Unit Design:** Basic Micro sequencer Design, Design a Very Simple Micro sequencer, Implementation of a Very Simple Micro sequencer, Micro programmed Control vs. Hardwired Control. (Chapter 7, Text 2)

#### UNIT-V:

**Input-Output Organization:** Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer,

Modes of Transfer, Priority Interrupt, Direct Memory Access (DMA). (Chapter 11, Text 1)

**Memory Organization:** Memory Hierarchy, Main Memory, Auxiliary Memory, Associate Memory, Cache Memory. (Chapter 12, Text 1)



<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. Understands about data representation and computer arithmetic.</li> <li>2. Acquires knowledge on Boolean Algebra and 8085 instruction set architecture.</li> <li>3. Understands the basics of computer organization.</li> <li>4. Ability to understand and design CPU of a computer.</li> <li>5. Ability to analyze the input and output organization of a computer.</li> </ol>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Computer System Architecture, M. Morris Mano, Third Edition, Pearson Education, 2007.</li> <li>2. Computer Systems Organization &amp; Architecture, John D. Carpinelli, Pearson Education, 2001.</li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Digital Logic and Computer Organization, V. Rajaraman and T. Radhakrishnan, PHI Publication, 2006.</li> <li>2. Computer Organization – Car Hamacher, Zvonks Vranesic, SafeaZaky, Vth Edition, McGrawHill.</li> <li>3. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int. Edition.</li> </ol>		

**OBJECT ORIENTED PROGRAMMING USING JAVA**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

- 1.The main objective is to teach the basic concepts and techniques which form the object-oriented programming paradigm.
- 2.Students completing the course should know the model of object-oriented programming: abstract data types, encapsulation, inheritance and polymorphism.
- 3.Students completing the course should know fundamental features of an object-oriented language like Java: object classes and interfaces, exceptions and libraries of object collections.
- 4.Students completing the course should know how to take the statement of a business problem and from this determine suitable logic for solving the problem; then be able to proceed to code that logic as a program written in Java.

**SYLLABUS**

**UNIT I:**

**Introduction:** Object Oriented Paradigm - Basic Concepts of OOP - Benefits of OOP - Java history - Java Features - Java Environment - Simple Java Program - Java Program Structure - Java Tokens - Implementing a Java Program – JVM - Command Line Arguments – Variables – Constants - Data Types - Declaration of Variables - Scope of Variables - Type Casting – Operators - Control Statements – Arrays – Strings - Vectors.

**UNIT II:**

**Classes-Objects-Methods:** Defining a Class - Fields Declaration - Methods Declaration - Creating Objects - Accessing Class Members – Constructors - Methods Overloading - Static Members - Nesting of Methods.

**Inheritance:** Basics - Extending a Class - Extending a Class - Overriding Methods - Final Variables and Methods - Final Classes - Abstract Methods and Classes.

**UNIT III:**

**Interfaces and Packages:** Defining Interfaces - Extending Interfaces - Implementing Interfaces - Accessing Interface Variables - Creating Packages - Accessing a Package - Using a Package - Adding a Class to a Package - Java API Packages - Using System Packages - Naming Conventions.

**UNIT IV:**

**Multithreaded Programming:** Creating Threads - Extending the Thread Class - Life cycle of Thread - Using Thread Methods - Thread Exceptions - Thread Priority – Synchronization - Implementing the Runnable Interface - Inter-Thread Communication.

**Managing Errors and Exceptions:** Types of Errors – Exceptions - Syntax of Exception Handling Code - Multiple Catch Statements - Using Finally Statements - Throwing Our Own Exceptions - Using Exceptions for Debugging.

<b>UNIT V:</b>		
<p><b>Event Handling:</b> Events - Event Sources - Event Classes - Event Listeners - Delegation Event Model - Handling Mouse and Keyboard Events - Adapter Classes - Inner Classes.</p> <p><b>AWT:</b> The AWT Class Hierarchy - User Interface Components- Labels – Button – Canvas – Scrollbars - Text Components - Check Box - Check Box Groups – Choices - Lists Panels – Scroll Pane – Dialogs - Menu Bar – Graphics - Layout Manager – Layout Manager Types.</p> <p><b>Applets:</b> Building Applet Code – Applet Life cycle – Creating an Executable Applet – Designing Web Page – Applet Tag – Adding Applet to HTML File – Running the Applet.</p>		
<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. Understands the basics of java programming.</li> <li>2. Understands the concepts of object orientation methods and inheritance using java.</li> <li>3. Obtain the overview of interfaces and java API.</li> <li>4. Gains knowledge on multithreading and exception handling in Java.</li> <li>5. Able to design GUI using applets.</li> </ol>		
<b>Text Book:</b>		
Programming with java, E. Balagurusamy, TATA Mcgraw Hill.		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1.Head First Java 2nd edition, Kathy Sierra and Bert Bates. Orielly Publications</li> <li>2.The Complete Reference Java J2SE 5th Edition, Herbert Schildt, TMH Publishing Company Ltd, New Delhi.</li> </ol>		

**OPERATING SYSTEMS**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

- 1.To learn about fundamentals, functions, types of operating systems and processes communication.
- 2.To learn the mechanisms of OS to handle processes, Knowledge towards Processes Synchronization and gain knowledge on deadlock detection algorithms.
3. To provide the knowledge towards memory management algorithms, virtual memory and Secondary storage structure.
4. To learn about Unix advanced Commands and Bourne Shell commands.

**SYLLABUS**

**UNIT-I:**

**Introduction:** Introduction to Operating Systems-Computer-System Organization-Computer-System Architecture-Operating-System Structure-Operating-System Operations. **System Structure:** Operating- System Services-User Operating-System Interface-System Calls- Types of System Calls- System Boot. **Process Concepts:** Process Concept-Process Scheduling. Operations on Processes- Inter Process Communication.

**UNIT-II:**

**Process Scheduling:** Basic Concepts-Scheduling Criteria-Scheduling-Algorithms. **Process Synchronization:** Background-The Critical-Section Problem-Peterson’s Solution-Synchronization Hardware-Semaphores-Classic Problems of Synchronization-Monitors. **Deadlocks:** System Model-Deadlock Characterization-Methods for Handling Deadlocks-Deadlock Prevention-Deadlock Avoidance-Deadlock Detection-Recovery from Deadlock.

**UNIT-III:**

**Memory Management:** Background-Swapping-Contiguous Memory Allocation-Paging-Structure of the Page Table-Segmentation. **Virtual Memory Management:** Background-Demand Paging-Copy-on-Write-Page Replacement- Allocation of Frames-Thrashing.

**UNIT-IV:**

**File System Interface:** File Concept-Access Methods-Directory and Disk Structure. **Implementing File Systems:** File-System Structure-File-System Implementation- Allocation Methods. **Secondary Storage Structure:** Overview-Disk Structure-Disk Scheduling.

**UNIT-V:**

**Unix:** History of Unix-Features of Unix System- Structure of Unix System- Introduction to Unix File System- Organization and Structure of File Systems- Basic and Advanced Unix Commands-Visual Editor. **Bourne Shell Programming:** Features-Bourne Shell Scripting: echo, variables, expr, let, base conversion, factors, UNITS, reading input, if statement, **Loops:** for, while, until.

<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. To familiarize with the fundamentals and different types of operating systems.</li> <li>2. To learn the concepts of Process Scheduling, processes synchronization and CPU Scheduling.</li> <li>3. To acquaint knowledge about Deadlocks and memory management.</li> <li>4. To study the concept of Disk Scheduling, Disk Management and Security issues.</li> <li>5. To practice Unix File system, advanced commands and Shell programming.</li> </ol>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Operating System Concepts-8th edition-Abraham Silber schatz-Peter Galvin-Grey Gagne-John Wiley&amp; Sons.</li> <li>2. Unix and Shell Programming-B.M.Harwani-Oxford.</li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Dhamdhare (II<sup>nd</sup> Revised Edition)-System Programming and Operating Systems &amp; System Programming-Tata McGraw Hill.</li> <li>2. Unix Shell Programming BPB Yashawant kanetkar.</li> <li>3. Modern Operating Systems-A. S Tanenbaum Pearson/PHI latest Edition.</li> <li>4. Operating Systems -William Stallings-PHI latest Edition.</li> </ol>		

**ADVANCED DATA STRUCTURES USING JAVA LAB**

<b>Credits: 1.5</b>	<b>Lab: 3 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 50 Marks</b>	<b>Internal: 50 Marks</b>
<b>Course Objectives:</b>		
<ol style="list-style-type: none"> <li>1. To implement stacks and queues using arrays and linked lists.</li> <li>2. To develop programs for searching and sorting algorithms.</li> <li>3. To write programs using concepts of various trees.</li> <li>4. To implement programs using graphs.</li> </ol>		
<b>SYLLABUS</b>		
<b>Cycle I:</b>		
<ol style="list-style-type: none"> <li>1. List ADT implementation using arrays.</li> <li>2. Implementation of Stacks using Arrays.</li> <li>3. Evaluation of postfix Expression.</li> </ol>		
<b>Cycle II:</b>		
<ol style="list-style-type: none"> <li>1. Queues using arrays.</li> <li>2. Circular queue using arrays.</li> <li>3. Implementation of single Linked Lists.</li> <li>4. Stacks and Queues using Single Linked Lists.</li> <li>5. Implementation of a Dequeue.</li> </ol>		
<b>Cycle III:</b>		
<ol style="list-style-type: none"> <li>1. Binary Tree Traversals using Recursion.</li> <li>2. Binary Search Tree-insertion, deletion, and traversing.</li> <li>3. BFS and DFS Graph traversals.</li> <li>4. Implementation of AVL Trees</li> <li>5. Implementation of B-Trees</li> <li>6. Implementation of B+Trees</li> <li>7. Dictionary ADT implementation.</li> </ol>		
<b>Cycle IV:</b>		
<ol style="list-style-type: none"> <li>1. Insertion sort, Selection sort, and Bubble sort.</li> <li>2. Quick sort and Merge sort.</li> <li>3. Linear search and Binary search</li> </ol>		
<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. Practice applications of stacks and queues.</li> <li>2. Able to write programs to implement linked list.</li> <li>3. Practice implementation of various searching and sorting techniques.</li> <li>4. Implementation of TREES and GRAPHS.</li> <li>5. Exhibit applications using data structures.</li> </ol>		
<b>Text Book:</b>		
Data Structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Edition.		
<b>Reference Book:</b>		
The Complete Reference Java 2 Fifth Edition by Patrick Naughton and Herbert Schildt TMH		

**COMPUTER ORGANIZATION LAB**

<b>Credits: 1.5</b>	<b>Lab: 3 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 50 Marks</b>	<b>Internal: 50 Marks</b>

**Course Objectives:**

1. To learn the about logic gates, half adders, full adders and flip -flops.
2. To learn about the microprocessor programming.
3. To learn about the microprocessor interfacing with stepper motor, R-2R ladder.
4. To develop the skill in writing microprocessor programming.

**SYLLABUS**

**DIGITAL EXPERIMENTS**

1. Verification of truth tables of Logic Gates
2. TTL characteristics, Verification of Demorgan's Laws
3. Implementation of Adders and Subtractors
4. Verification of Flip-Flops (RS- JK- D- T)
5. Design of 3-to-8 Decoder, Encoder
6. Multiplexer, De-Multiplexer

**8085 ASSEMBLY LANGUAGE PROGRAMMING**

7. Clears a Memory Location
8. To Move the Contents in one memory Location to Another
9. To Exchange the contents in Memory location
10. To Find the Sum of Two Numbers
11. To Transfer a Value from one Memory to Another (Indirect Mode)
12. To Exchange the contents in two Memory Locations (Indirect Mode)
13. To Add 2-8 Bits Store Result in 16-Bit (Indirect Mode)
14. To Find largest of 2-8 Bit NO's
15. To Find Smallest of 2-8 Bit NO's
16. To Add 2-16 bits Store Result in 32-Bit
17. To Find the Sum of Series of 8-bit and result in 16-Bit
18. To Determine a NO. Is Even or Odd
19. To Count NO. Of 1's in a given Byte
20. To find 2's Complement of a given NO.
21. To Multiply Two 8-Bit No.'s
22. To Perform Division of 2-8 Bit No's
23. To Find Factorial of a Given Number
24. To add 2-8Bit Numbers with Carry
25. To Shift Left One-Bit

**Outcomes:**

1. The student understands and learns the applications of Digital logic design.
2. The student understands and learns the concept of memory design.
3. The student understands and learns the concept of data interpretation.
4. The student understands and learns the concept of data transmission.
5. The student develops the skill of writing microprocessor programming.

<b>Text Books:</b>		
1. Computer System Architecture, M. Morris Mano, Third Edition, Pearson Education, 2007.		
2. Computer Systems Organization & Architecture, John D. Carpinelli, Pearson Education, 2001.		
<b>Reference Books:</b>		
1.Digital Logic and Computer Organization, V. Rajaraman and T.Radhakrishnan, PHI Publication, 2006.		
2.Computer Organization – Car Hamacher, ZvonksVranesic, SafeaZaky, Vth Edition, McGrawHill.		
3.Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi, Springer Int.Edition.		



**OPERATING SYSTEMS LAB**

<b>Credits: 1.5</b>	<b>Lab: 3 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 50 Marks</b>	<b>Internal: 50 Marks</b>

<b>Course Objectives:</b>		
<ol style="list-style-type: none"> <li>1. To familiarize the students with the Architecture of UNIX Operating System.</li> <li>2. To learn the mechanisms of CPU Scheduling and Deadlock Detection algorithms.</li> <li>3. To learn mechanisms of Processes synchronization using semaphores.</li> <li>4. To familiarize students on UNIX System Calls and shell programming</li> </ol>		
<b>SYLLABUS</b>		
<b>Cycle-1:</b>		
<ol style="list-style-type: none"> <li>1. Write programs using the system calls of UNIX operating system.</li> <li>2. UNIX Shell Programming.</li> </ol>		
<b>Cycle -2:</b>		
<ol style="list-style-type: none"> <li>1. Programs to simulate process scheduling like FCFS- SJF and Round Robin.</li> <li>2. Programs to simulate page replacement algorithms like FIFO- Optimal and LRU.</li> <li>3. Programs to simulate deadlock detection.</li> <li>4. Implement the Producer – Consumer problem using semaphores.</li> </ol>		
<b>Cycle -3:</b>		
<ol style="list-style-type: none"> <li>1. Implement the Producer-Consumer Program using Semaphores.</li> <li>2. Implement Paging memory management scheme.</li> <li>3. Implement any file allocation technique (Linked-Indexed-Contiguous).</li> </ol>		
<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. To differentiate the command set of MS Dos and UNIX.</li> <li>2. To familiarize with shell programming and shell commands.</li> <li>3. To practice the concepts of programs using system calls.</li> <li>4. To implement CPU Scheduling and Deadlock Algorithms.</li> <li>5. To implement Page replacement algorithms.</li> </ol>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Unix Systems Programming: Communication-Concurrency and Threads- Kay Robbins- Edition- Pearson Education</li> <li>2. Unix concepts and applications-Sumitabha Das- TMH Publications.</li> <li>3. Unix programming- Stevens- Pearson Education.</li> <li>4. Shell programming-Yashwanth Kanetkar.</li> <li>5. Operating System Concepts-Silberschatz- and Peter Galvin.</li> <li>6. Beginning Android 4 Application Development by Wei-Meng Lee-Wiley India Pvt Ltd.</li> </ol>		

# MASTER OF SCIENCE IN COMPUTER SCIENCE

## Syllabi

With effect from 2020-21 admitted batch

### I YEAR II SEMESTER

Subject code	Title of the Paper	Periods / Week		Max Marks		Total	Credits
		Theory	Lab	Ext.	Int.		
MSCS1.2.1	Web Programming	4	-	70	30	100	4
MSCS1.2.2	Data Base Management Systems	4	-	70	30	100	4
MSCS1.2.3	Artificial Intelligence	4	-	70	30	100	4
MSCS1.2.4	Data Communications and Computer Networks	4	-	70	30	100	4
MSCS1.2.5	Formal Languages and Automata Theory	4	-	70	30	100	4
MSCS1.2.6	<b>Elective – I:</b> (i) Image Processing (ii) Information Retrieval System (iii) Embedded Systems	4	-	70	30	100	4
MSCS1.2.7	Web Programming Lab	-	3	50	50	100	1.5
MSCS1.2.8	Data Base Management Systems Lab	-	3	50	50	100	1.5
MSCS1.2.9	Computer Networks Lab	-	3	50	50	100	1.5
<b>Total</b>		<b>24</b>	<b>9</b>	<b>570</b>	<b>330</b>	<b>900</b>	<b>28.5</b>

**WEB PROGRAMMING**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

1. Understand the principles of Web based application development and designing dynamic content in web pages using Java Script and XML
2. Understanding the concepts of client server communication using servlets.
3. Understanding the concept of java Server Pages and design applications using them.
4. Understand the concepts of Component development and design applications by establishing connections to Databases.
5. Understand the concepts of PHP and AJAX and their applications

**SYLLABUS****UNIT-I:**

**HTML Common tags:** Introduction – List – Tables – Images – Forms – Frames - Cascading Style sheets - Introduction to Java Scripts - Objects in Java Script – Dynamic HTML with Java Script. **XML:** Introduction - Document type definition - XML Schemas - Document Object model - Presenting XML - Using XML Parsers: DOM and SAX.

**UNIT-II:**

**Web Servers and Servlets:** Tomcat web server - Introduction to Servlets - Lifecycle of a Servlet – JSDK - The Servlet API - The javax.servlet Package - Reading Servlet parameters - Reading Initialization Parameters - The javax.servlet HTTP package - Handling Http Request & Responses - Using Cookies-Session Tracking - Security Issues.

**UNIT-III:**

**Introduction to JSP:** The Problem with Servlet - The Anatomy of JSP Page – JSP Processing - JSP Application Design with MVC Architecture - Setting Up and JSP Environment - Installing the Java Software Development Kit - Tomcat Server & Testing Tomcat - JSP Application Development - Generating Dynamic Content - Using Scripting Elements - Implicit JSP Objects.

**UNIT-IV:**

**Database Access:** Database Programming using JDBC - Studying javax.sql. \* package - Accessing a Database from a JSP Page - Application-Specific Database Actions - Deploying JAVA Beans in a JSP Page - Introduction to struts framework.

**UNIT-V:**

**PHP and AJAX:** PHP Basic Syntax, defining variable and constant, PHP Data type, Operator and Expression; **Handling Html Form with PHP:** Capturing Form Data, Dealing with Multi-value filed, Generating File uploaded form, redirecting a form after submission; Decisions and loop; Function; Strings; Arrays; **AJAX:** Introduction to AJAX, AJAX Technologies, AJAX Examples, Steps of AJAX operations, XML HttpRequest Methods, AJAX Database operations.

<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. To Construct web-based applications using Java script and XML.</li> <li>2. To learn design application using java Servlets, java Server Pages.</li> <li>3. To understand application design in JSP.</li> <li>4. To practice design applications using database connectivity.</li> <li>5. To acquire the basic concepts of PHP and AJAX.</li> </ol>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Web Programming- building internet applications- Chris Bates 2nd edition- WILEY Dreamtech.</li> <li>2. The complete Reference Java 2 Fifth Edition by Patrick Naughton and Herbert Scheldt. TMH.</li> <li>3. Java Server Pages –Hans Bergsten- SPD O’Reilly.</li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Web Technologies by Y.Ramesh Babu- Overseas Publishers Pvt.Ltd.</li> <li>2. Programming world wide web-Sebesta- Pearson.</li> <li>3. Core Servlets and Javasever Pages Volume1: Core Technologies by Marty Hall and Larry Brown Pearson.</li> <li>4. Internet and World Wide Web – How to program by Dietel and Nieto PHI/Pearson Education Asia.</li> <li>5. Jakarta Struts Cookbook- Bill Siggelkow- S P D O’Reilly for chap-8.</li> <li>6. Murach’s beginning JAVA JDK 5- Murach- SPD.</li> <li>7. Professional PHP4, Luis Argerich, WROX, SDP.</li> </ol>		

**DATABASE MANAGEMENT SYSTEMS**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

1. To understand the different issues involved in the design and implementation of a database system.
2. To study the physical and logical database designs, database modeling, relational, hierarchical, and network models.
3. To understand and use data manipulation language to query, update, and manage a database.
4. To develop an understanding of essential DBMS concepts such as: database security, integrity, and concurrency.
5. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

**SYLLABUS****UNIT-I:**

**Overview of Database systems:** History- File system verses DBMS-Advantages of DBMS-describing and storing of data-Transaction management-structure of DBMS-People who work with DBMS. **Introduction to Database Design:** Design-ER diagrams-Beyond ER Design-Entities-Attributes and Entity Sets-Relationships and Relationship sets- Additional features of ER Model-Conceptual Design with the ER Model-Conceptual Design for Large enterprises. **Relational Model:** Introduction to the Relational Model – Integrity Constraints Over Relations-Enforcing Integrity Constraints-Querying relational data- Logical data base Design-Introduction to Views – Destroying/altering Tables and Views.

**UNIT-II:**

**Relational Algebra and Calculus:** Relational Algebra – Selection and Projection-Set Operations-Renaming-Joins-Division-Examples of Algebra Queries-Relational calculus – Tuple relational Calculus – Domain relational calculus. **Form of Basic SQL Query:** Examples of Basic SQL Queries- Introduction to Nested Queries- Correlated Nested Queries-Set – Comparison Operators-Aggregate Operators-NULL values – Comparison using Null values – Logical connectives – Impact on SQL Constructs- Outer Joins-Disallowing NULL values- Complex Integrity Constraints in SQL Triggers and Active Data bases.

**UNIT-III:**

**Introduction to Schema Refinement:** Problems Caused by redundancy- Decompositions – Problem related to decomposition-Functional Dependencies - Reasoning about FDS-Normal Forms –BCNF –Properties of Decompositions- Loss less- join Decomposition-Dependency preserving Decomposition-Schema Refinement in Data base Design – Multi valued Dependencies – FOURTH Normal Form-Join Dependencies-FIFTH Normal Form-Inclusion Dependencies.

**UNIT-IV:**

**Overview of Transaction Management:** The ACID Properties-Transactions and Schedules-Concurrent Execution of Transactions – Lock Based Concurrency Control. **Concurrency Control:** Serializability and recoverability – Introduction to Lock Management – Lock Conversions-Dealing with Dead Locks-Specialized Locking Techniques – Concurrency Control without Locking.

<b>UNIT-V:</b>		
<p><b>Crash recovery:</b> Introduction to Crash recovery- Introduction to ARIES-the Log -Other Recovery related Structures- the Write-Ahead Log Protocol-Check pointing-recovering from a System Crash-Media recovery. <b>Security and Authorization:</b> Introduction to Database Security-Access Control- Discretionary Access Control.</p>		
<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. To understand various database models.</li> <li>2. To imbibe querying techniques in Entity Relation model.</li> <li>3. To practice optimization of database design with Normalization.</li> <li>4. To familiarize with the concepts of Serializability and Concurrency control.</li> <li>5. To gain knowledge on and crash recovery and the Database Security.</li> </ol>		
<b>Text Book:</b>		
<p>Database Management Systems- Raghu Ramakrishnan- Johannes Gehrke- TMH- 3rd Edition- 2003.</p>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Introduction to Database Systems-C.J. Date-Pearson Education.</li> <li>2. Data base System Concepts- A. Silberschatz-H.F. Korth- S. Sudarshan-McGraw hill- VI edition-2006.</li> <li>3. Fundamentals of Database Systems 5th edition. -Ramez Elmasri- Shamkant B.Navathe-Pearson Education-2008.</li> </ol>		

**ARTIFICIAL INTELLIGENCE**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

<b>Course Objectives:</b>		
<p>1. The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence that includes problem characteristics, Searching Techniques.</p> <p>2. To learn about Issues and variations, advanced systems in knowledge representations.</p> <p>3. To derive the logics, reasoning and statistical analysis.</p> <p>4. To study the concepts of Expert Systems</p> <p>5. To understand the basic functions and characteristics of artificial neural networks.</p>		
<b>SYLLABUS</b>		
<b>UNIT-I:</b>		
<p><b>Introduction:</b> What is Artificial Intelligence, The AI Problem, The Underlying Assumptions, what is an AI technique, The Level of the Model, Criteria for Success? Problem, Problem Spaces and Search: Defining Problem at a State Space Search, Production Systems, Problem Characteristics, Production System Characteristics and Issues in design of search programs. Heuristic Search Techniques: Generate-and-Test, Hill Climbing, Best-First-Search, Problem Reduction, Constraint Satisfaction, Means-End-Analysis.</p>		
<b>UNIT-II:</b>		
<p><b>Knowledge Representation Issues:</b> Representations and Mappings, Approaches to Knowledge Representation, Issues in Knowledge Representation, The Frame Problem. Representing knowledge using Rules: Procedural versus Declarative knowledge, Logic Programming, forward versus backward reasoning, matching, control knowledge. Work Slot and Filler Structures: Semantic nets, Frames. Strong slot and Filler Structures: Conceptual Dependency, Scripts.</p>		
<b>UNIT-III:</b>		
<p><b>Using Predicate Logic:</b> Representing Simple Facts in Logic, Representing Instance and Isa Relationships, Computable Functions and Predicates, Resolution, Natural Deduction. Symbolic Reasoning under uncertainty: Introduction to Non-monotonic Reasoning, Logics of Non-monotonic Reasoning, Statistical Reasoning: Probability and Bay's theorem, Certainty factors and rule-based systems, Bayesian Networks, Dumpster-Shafer Theory, Introduction to Fuzzy Logic.</p>		
<b>UNIT-IV:</b>		
<p><b>Expert Systems:</b> Introduction, Rule Based Expert System Architecture, Non-Production System Architecture, Dealing with Uncertainty, Knowledge Acquisition and Validation, Knowledge System Building Rules.</p>		
<b>UNIT-V:</b>		
<p><b>Neural Networks:</b> Characteristics of Neural Networks, Historical Development of Neural Networks Principles, Artificial Neural Networks: Terminology, Models of Neuron, Topology, Basic Learning Laws, Pattern Recognition Problem.</p>		

<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. To understand the techniques and applications of Artificial Intelligence.</li> <li>2. To familiarize with key issues and advanced knowledge representations.</li> <li>3. To obtain the knowledge on the representation of the language sentences using predicate logic.</li> <li>4. To gain awareness about expert system.</li> <li>5. To develop knowledge on neural network models.</li> </ol>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Artificial Intelligence, 2nd Edition, E. Ritch and K. Knight (TMH).</li> <li>2. Introduction to Artificial Intelligence and Expert Systems – Dan Watterson – Pearson Education – L.P. E</li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Artificial Intelligence – A Modern Approach. Second Edition, Stuart Russel, Peter Norvig, PHI/Pearson Education.</li> <li>2. Artificial Neural Networks B. Yagna Narayana, PH.</li> </ol>		



**DATA COMMUNICATIONS AND COMPUTER NETWORKS**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

1. To provide a solid foundation of the basics of data communication.
2. To prepare students to know the characteristics and designs of types of computer networks and their applications
3. Learn how computer network hardware and software operate
4. Investigate the fundamental issues driving network design
5. Learn about dominant network technologies.

**SYLLABUS****UNIT-I:**

**Introduction:** Study of Data Communications-Data Communication- Networks-Protocols and Standards-Standards Organizations-**Basic Concepts:** Line Configuration Topology-Transmission mode Categories of Networks - Internetworks-**The OSI Model:** The Model-Functions of the layers-TCP/IP Protocol Suite. **Encoding and Modulating:** Digital-to-Digital conversion-Analog-to-Digital conversion Digital-to-Analog conversion Analog-to-Analog conversion.

**UNIT-II:**

**Transmission Media:** Guided Media-Unguided Media-Transmission Impairment. **Error Detection and Correction:** Types of Errors-Detection- Vertical Redundancy Check (VRC) - Longitudinal Redundancy- Check (LRC) -Cyclic Redundancy Check (CRC) -Checksum-Error Correction-**Data Link Control:** Line Discipline-Flow Control-Error control. **Local Area Networks:** Project 802-Ethernet-Other Ethernet networks-Token bus -Token Ring-**Switching:** Circuit Switching-Packet Switching-Message switching.

**UNIT-III:**

**Point-to-Point Protocol (PPP):** transition states-PPP layers- Link Control Protocol (LCP)-Authentication Network Control Protocol (NCP)-**Frame Relay:** Introduction-Frame relay operation-Frame relay layers Congestion Control-Leaky Bucket Algorithm-Traffic Control.

**UNIT-IV:**

**Networking and Internetworking Devices:** Repeaters - Bridges-Routers- Gateways -Other devices -Routing Algorithms -Distance Vector Routing-Link State Routing-**Transport Layer:** Duties of the Transport Layer-Connection-The OSI Transport Protocol. **Upper OSI Layers:** Session Layer-Presentation Layer-Application Layer-**TCP/IP Protocol Suite: Part 1:** Overview of TCP/IP-Network Layer-Addressing-Subnetting-Other protocols in the network layer-Transport Layer.

**UNIT-V:**

**TCP/IP Protocol Suite: Part 2- Application Layer:** Client-Server Model-Bootstrap Protocol (bootp) and Dynamic Host-Configuration protocol (DHCP) -Domain Name System (DNS)-Telnet-File Transfer Protocol (FTP) -Trivial File Transfer Protocol (TFTP)-Simple Mail Transfer Protocol (SMTP)-Simple Network Management Protocol (SNMP)-Hypertext Transfer Protocol (HTTP)-World Wide Web (WWW).

<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. To understand the overview of Data Communications and Networks.</li> <li>2. To analyze the study on physical and data link layers.</li> <li>3. To familiarize with frame formats of data link layer.</li> <li>4. To gain knowledge about network and transport layer functionalities.</li> <li>5. To learn practical applications of networks.</li> </ol>		
<b>Text Book:</b>		
Data Communications and Networking- Behrouz A. Forouzan- 2 <sup>nd</sup> Edition revised- Tata Mcgraw- Hill Publishing Co.		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Understanding Data Communications and Networks- William A Shay- 2nd Edition- Vikas Publishing House.</li> <li>2. Computer Networks- Andrew S. Tanenbaum- Pearson Education- Low Price- 4th Edition.</li> <li>3. Data and Computer Communications- Williams Stallings- Prentice-Hall India- Eastern Economy Edition- 7th edition.</li> <li>4. Data Communications- Computer Networks and Open Systems- Fred Halsall- Pearson Education- Low Price- Edition- 4th edition- 2001.</li> </ol>		

**FORMAL LANGUAGES AND AUTOMATA THEORY**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

1. Understand basic properties of Deterministic and Nondeterministic Finite Automata.
2. Understand basic properties of Languages, Grammars, Normal forms and difference between types of languages and types of automata.
3. Understand Pushdown Automata and Turing Machines, concepts of tractability and decidability, concepts of NP-Completeness and NP-Hard problems.
4. Understand the challenges of Theoretical computer science and its contribution to other sciences.

**SYLLABUS**

**UNIT- I:**

**Theory of Automata:** Definition and Description of an Automaton-Transition Systems-Properties and Acceptance of a String by Automaton-Deterministic Finite Automata (DFA) and Nondeterministic Finite State Automata (NFA)-Equivalence of DFA and NFA-Mealy and Moore Models-Minimization of Finite Automata.

**UNIT- II:**

**Formal Languages:** Basic Definitions and Examples-Grammar and Types of Grammar-Language and Chomsky classification of Languages- Language generated by Grammar-Grammar generated by Language-Operations on Languages-Languages and Automata. **Regular Sets and Regular Grammar:** Regular Expressions-Finite Automata and Regular Expressions-Pumping Lemma for Regular Sets-Application of Pumping Lemma-Closure properties of Regular Grammar.

**UNIT- III:**

**Context-Free Languages:** Context-Free Languages (CFL) and Derivation Trees-Ambiguity in Context-Free Grammars (CFG)-Simplification of CFG-Normal Forms for CFG-Pumping Lemma for CFL-Closure Properties of CFL-Decision Algorithms for CFL. **Pushdown Automata:** Definition-Acceptance of PDA-Context-free Languages-Parsing.

**UNIT- IV:**

**Turing Machines:** Definition and Model-Representing of Turing Machines-Language Acceptability by Turing Machine-Design and Description of Turing Machine-Techniques and Variants of Turing Machine.

**UNIT- V:**

**Decidability and Recursively Enumerable Languages:** Decidability-Decidable and Undecidable Languages-Halting Problem of TM-Post Correspondence Problem-Godelization. **Computability and Complexity:** Basic Concepts-Primitive Recursive Functions-Recursive Functions-Partial Recursive Functions.

<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. Familiarizes with various types of Finite Automata.</li> <li>2. Understand the types of Grammar and Regular expressions.</li> <li>3. Learn the concepts of Context Free Language, Normal Forms and Pushdown Automata.</li> <li>4. Ability to construct Turing machines and apply on its applications.</li> <li>5. Optimize computability using Recursive functions.</li> </ol>		
<b>Text Book:</b>		
Theory of Computer Science, K.L.P. Mishra & N.Chandrasekaran, Third Edition, Prentice Hall of India Private Limited.		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Elements of Theory of Computation, Harry R Lewis &amp; Cistos H. Papadimitriou, Pearson Education/Prentice Hall of India Privated Limited.</li> <li>2. Introduction to Automata theory, Languages&amp; Computation, Hopcroft.J. E and J.D.Ullman, Addision-Wesely, Mass 1979.</li> </ol>		

**IMAGE PROCESSING**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

**Course Objectives:**

1. To explain fundamentals of Image processing concepts.
2. To provide mathematical foundation of image enhancement, image compression and image segmentation.
3. To explain the students about Morphology and its applications in image processing.
4. To explain various methods and techniques for image transformation.

**SYLLABUS**

**UNIT I:**

**Fundamental steps in Image Processing**-Elements of Digital image processing systems-Digital image fundamentals. Uniform and Non-uniform sampling and quantization. Some basic relationships between pixels-Imaging Geometry.

**UNIT II:**

**Image enhancement (in spatial domain and frequency domain):** Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters. Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters.

**UNIT III:**

**Image compression:** Definition- A brief discussion on – Run length encoding, contour coding, Huffman code, compression due to change in domain, compression due to quantization Compression at the time of image transmission. Brief discussion on Image Compression standards.

**UNIT IV:**

**Image Segmentation:** Definition, characteristics of segmentation. Detection of Discontinuities, Thresholding Pixel based segmentation method. Region based segmentation methods – segmentation by pixel aggregation, segmentation by sub region aggregation, histogram-based segmentation, spilt and merge technique. Use of motion in segmentation (spatial domain technique only). **Morphology:** Preliminaries, Dilation and Erosion, Opening and Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms.

**UNIT V:**

**Image Transforms** – Introduction to Fourier Transform-DFT- Some properties of 2-D Fourier Transform-FFT, Properties. Walsh transform, Hadamard Transform, Discrete cosine Transform, Haar transform.

**Outcomes:**

1. Understands the fundamentals of Image processing concepts.
2. Ability to understand various image enhancement techniques.
3. Acquires mathematical foundation on compression techniques image enhancement.
4. Familiarizes with various methods of image segmentation and morphology.
5. Gains knowledge on image transformation techniques.

<b>Text Book:</b>		
Digital Image processing – R.C. Gonzalez & R.E. Woods, Addison Wesley/ Pearson education, 2 <sup>nd</sup> Edition, 2002.		
<b>Reference Books:</b>		
1. Fundamentals of Digital Image processing – A.K. Jain, PHI. 2. Digital Image processing using MATLAB – Rafael C. Gonzalez, Richard E Woods and Steven L Edition, PEA, 2004. 3. Digital Image Processing – William K. Pratt, John Wiley, 3rd Edition, 2004.		

**INFORMATION RETRIEVAL SYSTEM**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>
<b>Course Objectives:</b>		
<ol style="list-style-type: none"> <li>1. To explain fundamentals of Domain Analysis of IR systems concepts.</li> <li>2. To Know the Techniques for Inverted Files.</li> <li>3. To explain the New Indices for Text.</li> <li>4. To explain various methods and techniques for String Searching Algorithms.</li> </ol>		
<b>SYLLABUS</b>		
<b>UNIT I:</b>		
<b>Introduction to Information storage and retrieval systems:</b> Domain Analysis of IR systems, IR and other types of Information Systems, IR System Evaluation. <b>Introduction to Data structures and algorithms related to Information Retrieval:</b> Basic Concepts, Data structures, Algorithms.		
<b>UNIT II:</b>		
<b>Inverted Files:</b> Introduction, Structures used in Inverted Files, Building an Inverted file using a sorted array, Modifications to the Basic Techniques. <b>Signature Files:</b> Introduction, Concepts of Signature files, Compression, Vertical Partitioning, Horizontal Partitioning.		
<b>UNIT III:</b>		
<b>New Indices for Text:</b> PAT Trees and PAT Arrays: Introduction, PAT Tree structure, Algorithms on the PAT Trees, Building PAT Trees as PATRICA Trees, PAT representation as Arrays. <b>Lexical Analysis and Stoplists:</b> Introduction, Lexical Analysis, Stoplists.		
<b>UNIT IV:</b>		
<b>Stemming Algorithms:</b> Introduction, Types of Stemming algorithms, Experimental Evaluations of Stemming, Stemming to Compress Inverted Files. <b>Thesaurus Construction:</b> Introduction, Features of Thesauri, Thesaurus Construction, Thesaurus construction from Texts, Merging existing Thesauri.		
<b>UNIT V:</b>		
<b>String Searching Algorithms:</b> Introduction, Preliminaries, The Naive Algorithm, The Knutt-Morris-Pratt Algorithm, The Boyer-Moore Algorithm, The Shift-Or Algorithm, The Karp-Rabin Algorithm.		
<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. Understands the fundamentals of Information storage and retrieval systems.</li> <li>2. Ability to understand various concepts of Signature files.</li> <li>3. Gains knowledge about PAT Trees and Lexical Analysis and Stoplists.</li> <li>4. Acquires knowledge on Thesaurus Construction.</li> <li>5. Familiarizes with various String Searching Algorithms.</li> </ol>		

<b>Reference Books:</b>		
-------------------------	--	--

1. Modern Information Retrieval, Ricardo Baeza-Yates, Neto, PEA,2007.
2. Information Storage and Retrieval Systems: Theory and Implementation, Kowalski, Gerald, Mark Academic Press, 2000.
3. Information Retrieval: Algorithms and Heuristics, Grossman, Ophir Frieder, 2/e, Springer, 2004.
4. Information Retrieval Data Structures and Algorithms, Frakes, Ricardo Baeza-Yates, PEA
5. Information Storage and Retieval, Robert Korfhage, John Wiley & Sons.
6. Introduction to Information Retrieval, Manning, Raghavan, Cambridge University Press.



**EMBEDDED SYSTEMS**

<b>Credits: 4</b>	<b>Theory: 4 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 70 Marks</b>	<b>Internal: 30 Marks</b>

<b>Course Objectives:</b>		
<ol style="list-style-type: none"> <li>1. To study the basics of embedded systems and its examples.</li> <li>2. To study the 8051 Microcontroller architecture and its instruction set.</li> <li>3. To discuss various software architectures in embedded systems.</li> <li>4. To discuss Inter Task Communication procedures in RTOS and design issues of RTOS.</li> <li>5. To study various embedded software development tools and debugging techniques.</li> </ol>		
<b>SYLLABUS</b>		
<b>Unit I:</b>		
<b>Introduction-</b> Definition & Examples of Embedded Systems (Ch 1 of Text 2) Microprocessors and Micro controllers- The 8051 Architecture (Ch 1- Ch 3 of Text 1)		
<b>Unit II:</b>		
<b>8051 Assembly Language Programming:</b> Moving Data- Arithmetic and Logical Operations- Jump and Call Instructions (Ch5- Ch6- Ch7- Ch8of Text 1)		
<b>Unit III:</b>		
Interrupts and Survey of Software Architectures (Ch 4- Ch 5 of Text 2)		
<b>Unit IV:</b>		
Introduction to Real-Time Operating Systems (Ch 6 of Text 2) More Operating System Services (Ch 7 of Text 2)		
<b>Unit V:</b>		
Embedded Software Development Tools (Ch 9 of Text 2) Debugging Techniques (Ch 10 of Text 2)		
<b>Outcomes:</b>		
<ol style="list-style-type: none"> <li>1. Understands the basics of Embedded systems, Microprocessors and Microcontrollers.</li> <li>2. Develops ability to write programs using 8051 Assembly Language instructions.</li> <li>3. Learns about various Interrupts and Software Architecture.</li> <li>4. Analyzes various design issues of RTOS.</li> <li>5. Familiarizes with embedded software development tools and debugging techniques.</li> </ol>		
<b>Text Book:</b>		
<ol style="list-style-type: none"> <li>1. The 8051 Micro controllers- Architecture- Programming- &amp; Applications- by Kenneth J. Ayala- Penram International Publishing (India)- Second Edition 1996.</li> <li>2. An Embedded Software Primer- David E. Simon- Pearson Education Inc.- 1999.</li> </ol>		
<b>References:</b>		
<ol style="list-style-type: none"> <li>1. Embedded Systems- Architecture- Programming and Design- by Raj Kamal TMH- 2003.</li> <li>2. Embedded Real Time Systems Programming- by Sriram V Iyer and Pankaj Gupta-TMH- 2004.</li> </ol>		

**WEB PROGRAMMING LAB**

<b>Credits: 1.5</b>	<b>Lab: 3 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 50 Marks</b>	<b>Internal: 50 Marks</b>

**Course Objectives:**

1. Understand the principles of Web based application development.
2. Design dynamic content in Web Pages using JavaScript and XML.
3. Understanding the concepts of java Servlets, java Server Pages and design applications using them.
4. Understand the concepts of Component development and design applications by establishing connections to Databases.
5. Understand the concepts of PHP and design applications.

**SYLLABUS**

1. Introduction:
  - Introduction to HTML
  - Introduction to Java Script
  - Introduction to XML
2. Feedback Form using HTML tags
3. Develop Web-site using only HTML Tags and Cascading style sheets
4. DHTML Programs
  - 4.1 Color Picker
  - 4.2 Rollover Buttons
5. Java Script Programs
6. XML Programs
7. Installation and Running of Tomcat Server
8. Servlet Programs
  - 8.1 Displaying simple text
  - 8.2 Validating user's login information by parameter passing
  - 8.3 Handling http request and response
  - 8.4 Handling cookies
  - 8.5 Session tracking
9. JSP Programs
  - 9.1 Displaying simple text
  - 9.2 Validating user's information by conditional processing
  - 9.3 Session tracking
10. PHP Programs
  - 10.1 Displaying simple text
  - 10.2 Validating user's information by conditional processing

**Outcomes:**

1. To develop web-based applications using Java script and XML.
2. To design applications using java Servlets, java Server Pages.
3. To learn the connectivity of java programs to different databases.
4. To structure applications using PHP.

**DATABASE MANAGEMENT SYSTEMS LAB**

<b>Credits: 1.5</b>	<b>Lab: 3 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 50 Marks</b>	<b>Internal: 50 Marks</b>

**Course Objectives:**

1. The major objective of this lab is to provide a strong formal foundation in database concepts, technology and practice to the participants to groom them into well- Informed database application developers.
2. To present SQL and procedural interfaces to SQL comprehensively.
3. To present the concepts and techniques relating to query processing by SQL Engines.
4. To understand and use data manipulation language to query, update, and manage a Database.
5. To present the concepts and techniques relating to ODBC and its Implementations.
6. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

**SYLLABUS****Cycle-I:**

1. Creation- altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub-Queries) using ANY- ALL- IN- EXISTS- NOTEXISTS- UNION- INTERSET- Constraints. Example: - Select the roll number and name of the student who secured fourth rank in the class. Queries using Aggregate functions (COUNT- SUM- AVG- MAX and MIN)- GROUP BY- HAVING and Creation and dropping of Views.
3. Queries using Conversion functions (to\_char- to\_number and to\_date)- string functions (Concatenation- lpad- rpad- ltrim- rtrim- lower- upper- initcap- length- substr and instr)- date functions (Sysdate- next\_day- add\_months- last\_day- months\_between- least- greatest- trunc- round- to\_char- to\_date).

**Cycle-II:**

1. Creation of simple PL/SQL program which includes declaration section- executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found).
2. Insert data into student table and use COMMIT- ROLLBACK and SAVEPOINT in PL/SQL block.
3. Develop a program that includes the features NESTED IF- CASE and CASE expression.

**Cycle-III:**

1. Program development using WHILE LOOPS- numeric FOR LOOPS- nested loops using ERROR Handling- BUILT-IN Exceptions-USE defined Exceptions-RAISE-APPLICATION ERROR.
2. Program's development using creation of procedures- passing parameters IN and OUT of PROCEDURES.
3. Program development using creation of stored functions- invoke functions in SQL Statements and write complex functions.

<b>Cycle-IV:</b>		
1. Develop programs using features parameters in a CURSOR- FOR UPDATE CURSOR- WHERE CURRENT of clause and CURSOR variables. 2. Develop Programs using BEFORE and AFTER Triggers- Row and Statement Triggers and INSTEAD OF Triggers.		
<b>Mini Project</b>		
<b>Outcomes:</b>		
1. To practice DDL, DML, DCL commands. 2. To design and implement a database schema for a given problem-domain and normalize a database. 3. To declare and enforce integrity constraints on a database using a state-of-the-art RDBMS. 4. To learn PL/SQL programming. 5. To familiarize with database connectivity.		

**COMPUTER NETWORKS LAB**

<b>Credits: 1.5</b>	<b>Lab: 3 Hours</b>	<b>Tutorials: -</b>
<b>Max Marks: 100</b>	<b>External: 50 Marks</b>	<b>Internal: 50 Marks</b>

**Course Objectives:**

1. Build an understanding of the fundamental concepts of computer networking.
2. Preparing the student for Advanced courses in computer networking.
3. Allow the students to gain expertise in some specific tools of Networking.
4. Allow the students to gain expertise in some specific Networking Simulators.

**SYLLABUS**

**Week 1:** Study of Network Devices in Detail.

Study of Different Types of Network Cables - **Cross Cable, Straight Cable**

**Week 2:** Study of Basic Network Oriented Commands

Study Classification of IP Addresses

**Week 3:** Study of **Subnetting**

**Week 4:** Configure a Network Topology Using **CISCO Packet Tracer**

**Week 5:** Study of Routing Protocols **RIP, OSPF**

**Week 6:** Study of Protocols **IPV6, DHCP**

**Week 7:** Study of **Client Server Configuration** using Packet Tracer

**Week 8:** Socket Programming **TCP Sockets**

**Week 9:** Protocol Analysis of TCP using **TTCP tool.**

**Week 10:** Protocol Analysis of TCP using **Wire Shark Tool**

**Outcomes:**

1. To understand the difference between serial communication and parallel communication with direct cable software component.
2. To learn the importance of Dialup networking and HyperTerminal.
3. To understand the concepts of different network configurations using star Topology.
4. To acquire knowledge on Cisco-packet tracer simulator by configuring the different applications.
5. To practice socket programming using TCP and UDP.