Post Graduate Department of Computer Sciences,
The University of Kashmir,
Srinagar - 190006

Curriculum
for
Master of Technology in Computer Science

2020 – 2021

Structure of Curriculum for M. Tech. in Computer Science

Semester-I (24 Credit unit Semester)						
Course Code	**Course name**	**Category**	**Hours / Week**			**Credits**
			L	T	P	
CORE SUBJECTS						
CSE20511	Embedded Systems	Core	4	0	0	4
CSE20512	Lab Embedded Systems	Core	0	0	4	2
CSE20513	Graph Theory	Core	4	0	0	4
CSE20514	Lab Graph Theory	Core	0	0	4	2
CSE20515	Artificial Intelligence	Core	4	0	0	4
ELECTIVE SUBJECTS						
CSE20516x	Elective 1	Elective	4	0	0	4
CSE20516x	Elective 2	Elective	4	0	0	4

List of Elective Subjects:

i) CSE205161 Advance Data Communications
ii) CSE205162 Advanced Database Management Systems
iii) CSE205163 Engineering Mathematics
iv) CSE205164 Object Oriented ~~Analysis~~ Method & Design

Semester-II (24 Credit unit Semester)						
Course Code	**Course name**	**Category**	**Hours / Week**			**Credits**
			L	T	P	
CORE SUBJECTS						
CSE20521	Network Security and Cryptography	Core	4	0	0	4
CSE20522	Lab Network Security and Cryptography	Core	0	0	4	2
CSE20523	Image Processing	Core	4	0	0	4
CSE20524	Lab Image Processing	Core	0	0	4	2
CSE20525	Machine Learning	Core	4	0	0	4
ELECTIVE SUBJECTS						
CSE20526x	Elective 1	Elective	4	0	0	4
CSE20526x	Elective 2	Elective	4	0	0	4

List of Elective Subjects:

i) CSE205261 Advanced Algorithms
ii) CSE205262 Software Reliability Engineering
iii) CSE205263 Optimization Techniques
iv) CSE205264 Big Data

M. Tech. Syllabus –P.G. Dept. of Computer Science, University of Kashmir

Semester-III (24 Credit unit Semester)						
Course Code	Course name	Category	Hours / Week			Credits
			L	T	P	
CORE SUBJECTS						
CSE20531	Minor Project	Core	0	4	0	4
CSE20532	Parallel and Distributed ~~Algorithms~~ Computing	Core	4	0	0	4
CSE20533	Real Time Operating Systems	Core	4	0	0	4
CSE20534	Advanced Wireless and Mobile Computing	Core	4	0	0	4
ELECTIVE SUBJECTS						
CSE20535x	Elective 1	Elective	4	0	0	4
CSE20535x	Elective 2	Elective	4	0	0	4

List of Elective Subjects:

i) CSE205351 Cloud Computing
ii) CSE205352 Internet of Things
iii) CSE205353 Natural Language Processing
iv) CSE205354 Block Chain

Semester-IV (24 Credit unit Semester)						
Course Code	Course name	Category	Hours / Week			Credits
			L	T	P	
CORE SUBJECTS						
CSE20541	Major Project Problem Identification	Core	0	2	0	2
CSE20542	Major Project Problem Analysis	Core	0	4	0	4
CSE20543	Major Project Software Development	Core	0	6	0	6
CSE20544	Major Project Research Component	Core	0	6	0	6
CSE20545	Major Project Dissertation	Core	0	6	0	6

Semester – I

<table>
<tr><td colspan="8">COURSE TITLE: Embedded Systems</td></tr>
<tr><td colspan="4">Course Code:</td><td colspan="2">CSE20511</td><td colspan="2">Examination Scheme</td></tr>
<tr><td colspan="6" rowspan="2">Total number of Lecture Hours: 52</td><td>External</td><td>80</td></tr>
<tr><td>Internal</td><td>20</td></tr>
<tr><td>Lecture (L):</td><td>4</td><td>Practicals(P):</td><td>0</td><td>Tutorial (T):</td><td>0</td><td>Total Credits</td><td>4</td></tr>
</table>

Course Objectives

- Understand the fundamental concepts and design challenges of Embedded Systems and Cyber-Physical Systems, including their classification and application areas.
- Describe the architecture and memory organization of the 8051 microcontroller, including its special function registers (SFRs) and I/O ports.
- Program the 8051 microcontroller to handle interrupts, including timer interrupts, external hardware interrupts, and serial communication interrupts.
- Develop skills in interfacing the 8051 microcontroller with external devices, such as LCDs, keyboards, DACs, ADCs, and stepper motors.

Course Content	TEACHING HOURS
UNIT 1: Introduction.	**13 Hrs**
Embedded systems and Cyber Physical Systems: Definition, Characteristics, Design Challenges, Classification, Application areas. Embedded Hardware Architecture: General Purpose Processor, Microprocessor Design Options, Microcontroller, Digital Signal Processor, ASIC, PLDs, COTS; Embedded Systems Memory; Other Hardware Components: I/O Subsystem, Timers and counters, Interrupt Subsystem, UART, PWM and Analog-Digital Conversion, Sensors and Actuators. Embedded Software Architectures: Round Robin, Round Robin with Interrupts, Function Queue Scheduling, Real-time Operating System (RTOS); Programming Languages and Tools; Embedded IDE; Debugging.	
UNIT 2: The 8051 Microcontroller.	**13 Hrs**
Microcontroller: Introduction, Criteria for choosing a microcontroller; Overview of 8051 Microcontroller family: Architecture, Memory Organization of 8051, SFRs, I/O Ports, Addressing modes. Basic Assembly Language programming concepts: 8051 Instruction set, Assembler Directives, Subroutine, Stack. Time delay generations and calculations, Programming of 8051 Timers, Counter Programming,WatchDog Timer,Real Time clock.	
UNIT 3: 8051 Communication and Interrupts.	**13 Hrs**

Basics of Communication: Overview of RS-232, I^2C Bus, UART, USB; Communication with 8051: Using I/O Ports, 8051 Serial Port, 8051 connections to RS232. 8051 interrupts: Interrupt vectors and interrupt processing, Level triggered and edge triggered, Masking and priorities; Programming of 8051 Timer interrupts, Programming of External hardware interrupts, Programming of the serial communication interrupts.	
UNIT 4: 8051 Interfacing.	**13 Hrs**
Basic Concepts of Interfacing: Introduction; 8051 Interfacing to external memory and Accessing External data Memory and External Code Memory. Interfacing to LCD/Keyboard, DAC/ADC, Sensors, Stepper Motor, 8255.	

Textbooks

1. Shibu K V. *Introduction to Embedded Systems*, TMH.
2. M.A. Mazidi and J. G. Mazidi. *The 8051 Microcontroller and Embedded Systems*, PHI.
3. Raj Kamal. *Embedded Systems*, TMH.

Reference Books

COURSE OUTCOMES (CO):

CO1: Demonstrate the ability to identify and classify different types of embedded systems and cyberphysical systems, and explain their role in various application areas.

CO2: Apply knowledge of 8051 microcontroller architecture to develop basic assembly language programs, effectively utilizing its instruction set and addressing modes.

CO3: Implement communication protocols like RS-232 and I2C with the 8051 microcontroller, and manage interrupt-driven tasks to optimize system performance.

CO4: Successfully interface the 8051 microcontroller with a variety of external devices, ensuring accurate data exchange and control, thereby enhancing practical hardware integration skills.

COURSE TITLE: Lab Embedded Systems

Course Code:				**CSE20512**		**Examination Scheme**	
Total number of Lab Hours:						**External**	**40**
						Internal	**10**
Lecture (L):	**0**	**Practicals(P):**	**4**	**Tutorial (T):**	**0**	**Total Credits**	**2**

Course Objectives:

Two to three course objectives to be listed by the course instructor

- Develop proficiency in designing and simulating embedded systems hardware schematics, including interfacing LEDs, switches, relays, keypads, 7-segment displays, and LCDs with the 8051 microcontroller using Proteus.
- Gain hands-on experience in writing and debugging assembly or C code for the 8051 microcontroller, focusing on controlling peripheral devices such as LEDs, relays, keypads, 7-segment displays, and DC motors.
- Enhance problem-solving skills by implementing real-time embedded solutions, such as creating an automated irrigation system controlled by the 8051 microcontroller, integrating sensors, actuators, and LCD displays for monitoring and control.

Week 1

- Design the schematic to connect an LED to 8051 on proteus via a pullup resistor.
- Write assembly or C code to make the LED blink on a pre specified Duty Cycle.

Week 2

- Design the schematic to add a push down switch to schematic designed in **week 1** via a pulldown resistor.
- Write assembly or C code to toggle an LED on the push of the button.

Week 3

- Design the schematic to interface a relay with 8051 for controlling a bulb.
- Write assembly or C code to control the on/off of a bulb via a relay on the push of the button.

Week 4

- Design the schematic to interface a 4 x 4 key pad with 8051.
- Write assembly or C code to detect and decode a keypress from the 4 x 4 key pad with 8051.

Week 5

- Design the schematic to interface a 7-segment display with 8051.
- Write assembly or C code to detect and decode a keypress from the 4 x 4 key pad with 8051 and display it on the 7-segment display.

Week 6

- Design the schematic and write assembly or C code to blink an LED using 8051 timers.

Week 7

- Design the schematic and write assembly or C code to display the number of button presses on 3-segment displays using 8051 counters.

Week 8

- Design the schematic and write assembly or C code to control a dc motor using 8051 via an H-Bridge. Use two buttons for forward and reverse.

Week 9

- Design the schematic and write assembly or C code to read and display value from a variable resistor on a 7-segment display using an ADC.

Week 10

- Design the schematic and write assembly or C code to interface a 16 x 2 LCD with 8051 for displaying "Hello World".

Week 11

- Design the schematic and write assembly or C code to interface a 16 x 2 LCD with 8051 for displaying a real time clock.

Week 12

- Design the schematic and write assembly or C code to interface two 8051 microcontrollers via the serial port for interchanging data at 9600bps. Use 16 x 2 LCD to display the received data.

Week 13

- Design an **embedded** solution for automatically controlling the irrigation system of a green house. Your job is to control the sprinklers depending upon the temperature of the green house. The LCD should display the current temperature and the last time when the sprinklers where on.

COURSE OUTCOMES (CO):

CO1: Demonstrate the ability to integrate multiple peripheral devices with the 8051 microcontroller in a cohesive embedded system, effectively using hardware design and software programming to accomplish tasks such as controlling LEDs, relays, keypads, and displays.

CO2: Develop a comprehensive understanding of real-time embedded system applications by designing and implementing complex projects, such as an automated irrigation system, which involve interfacing with sensors, actuators, and display components while ensuring functional and reliable system performance.

COURSE TITLE: Graph Theory

Course Code:				CSE20513		**Examination Scheme**	
Total number of Lecture Hours: 52						**External**	**80**
						Internal	**20**
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives

Four to Six course objectives to be listed by the course instructor

- Analyze and solve problems involving graph isomorphism, shortest path algorithms, and tree structures, including the application of Cayley's Formula and Sperner's Lemma to understand basic graph properties and relationships.
- Apply concepts of connectivity, Eulerian and Hamiltonian paths, and matchings in bipartite graphs to solve practical problems such as the Chinese Postman Problem and the Travelling Salesman Problem, demonstrating understanding of algorithmic solutions for network and assignment problems.
- Utilize edge and vertex coloring principles, including Vizing's Theorem and Ramsey's Theorem, to address graph coloring problems and timetabling issues, and apply Turan's Theorem to identify and analyze independent sets and cliques within graphs.
- Examine planar graphs and their properties, including Euler's Formula and Kuratowski's Theorem, to understand graph planarities, and apply network flow concepts and the Max-Flow Min-Cut Theorem to analyze and solve flow and cut problems in various applications.

Course Content	TEACHING HOURS
UNIT I- Graph Theory Fundamentals	**13 Hrs**
Graphs and subgraphs, Adjacency and incidence matrices, isomorphism. Paths and connections, Cycles, The shortest path problem, Sperners Lemma, Trees, Cut Edges and Bonds Cut Vertices, Cayleys Formula, The connector Problem	
UNIT II- Advanced Graph Theory	**13 Hrs**
Connectivity, Blocks with example, Eulerian tours and Hamilton Cycles, The Chinese postman problem, The travelling salesman problem, Matching and Coverings in Bipartite Graphs, Perfect Matchings, The Personnel Assignment Problem, The Optimal Assignment Problem.	
UNIT III- Graph Colorings and Theorems	**13 Hrs**

EDGE COLOURINGS:- Edge Chromatic Number , Vizing's Theorem. , The Timetabling Problem, INDEPENDENT SETS AND CLIQUES, Ramsey's Theorem , Turan's Theorem. Applications, VERTEX COLOU'RINGS :- Chromatic Number , Brooks' Theorem ,Chromatic Polynomial , Girth and Chromatic Number Applications	
Unit IV- Planar Graphs and Network Flows	**13 Hrs**
Plane and Planar Graphs, Dual Graphs , Euler's Formula, Bridges, Kuratowski's Theorem , The Five- Colour Theorem and the Four-Colour Conjecture, , Non-hamiltonian Planar Graphs , Directed Graphs, Directed Paths, Directed Cycles, Applications, NETWORKS, Flows , Cuts , The MaxFlow Min-Cut Theorem Applications , Menger's Theorems , Feasible Flows , THE CYCLE SPACE AND BOND SPACE , Circulations and Potential Differences. , The Number of Spanning Trees. Applications.	

Textbooks:

1. *GRAPH THEORY WITH APPLICATIONS,* J. A. Bondy and U. S. R. Murty published by Elsevier Science Publishing Co., Inc

Reference Books:

1. Douglas B. West, Introduction to Graph Theory, Second Edition, Prentice-Hall 2. Reinhard Diestel: "Graph Theory", Electronic Edition 2010.
3. B. Bollobas: "Modern Graph Theory" Springer, 1998.
4. Deo Narsingh,"Graph Theory With Applications To Engineering And Computer Science", PHI Learning Pvt. Ltd

COURSE OUTCOMES (CO):

CO1: Students will be able to demonstrate proficiency in creating and interpreting graph models using adjacency and incidence matrices, and will be capable of solving practical problems related to shortest paths and spanning trees, including the application of Cayley's Formula.

CO2: Students will be able to effectively apply graph connectivity concepts and algorithms to real-world problems, such as optimizing routes using Eulerian and Hamiltonian paths, and solving complex assignment problems and the Chinese Postman Problem.

CO3: Students will develop the ability to analyze and solve graph coloring problems using various theorems, such as Vizing's and Ramsey's Theorem, and will be able to apply these concepts to practical scenarios like timetabling and identifying independent sets and cliques..

CO4: Students will gain the skills to analyze and work with planar graphs and network flows, including understanding and applying Euler's Formula and the Max-Flow Min-Cut Theorem to solve complex problems related to graph planarities, network flows, and circulations..

COURSE TITLE: Lab Graph Theory								
Course Code:				**CSE20514**		**Examination Scheme**		
Total number of Lab Hours:						**External**		**40**
						Internal		**10**
Lecture (L):	**0**	**Practicals(P):**	**4**	**Tutorial (T):**	**0**	**Total Credits**	**2**	

Course Objectives:

Two to three course objectives to be listed by the course instructor

- Develop and implement programs to create and manage graph representations using incidence matrices, adjacency matrices, and adjacency lists. Additionally, write programs to find the number of cycles in a graph and determine its components and girth.
- Implement algorithms to solve advanced graph problems, including finding the shortest path in a graph, determining Hamiltonian cycles, and applying Kruskal's Algorithm for minimum spanning trees. Also, demonstrate understanding by solving the Traveling Salesman Problem and applying Fleury's Algorithm to find Eulerian tours.
- Write programs to address specialized graph problems, such as proving properties of simplicial subdivisions, implementing binary search trees with traversal operations, and solving optimal assignment and edge coloring problems in bipartite graphs.

Week 1

- Write a program to implement a graph using incidence matrix, adjacency matrix and adjacency list.

Week 2

- Write a program to find the number of cycles in a graph.

Week 3

- Write a program to find the shortest path in a graph.

Week 4

- Write a program to for determining (a) the components of a graph; (b) the girth of a graph.

Week 5

- Write a program to show that every properly labelled simplicial subdivision of a triangle has an odd number of distinguished triangles.

Week 6

- Write a program to implement a binary search tree along with its traversal and insertion and deletion of a node?

Week 7

- Write a program to show that simple connected graph that has exactly two vertices which are not cut vertices is a path.

Week 8

- Write a program to implement Kruskal's Algorithm.

Week 9

- Write a program to construct the closure of a graph and finding a Hamilton cycle if the closure is complete.

Week 10

- Write a program to implement the Fleury's algorithm to find the Euler tour of a graph.

Week 11

- Write a program to implement THE TRAVELLING SALESMAN PROBLEM ?
- Write a program to implement Hall's algorithm for matching of bipartite graph ?

Week 12

- Write a program to optimal assignment problem using graphs?

Week 13

- Write a program for finding a proper edge colouring of a bipartite graph G.?

COURSE OUTCOMES (CO):

CO1: Demonstrate the ability to apply theoretical concepts of graph theory to solve real-world problems, such as the Traveling Salesman Problem and optimal assignment issues, by implementing algorithms and validating their efficiency.

CO2: Enhance critical thinking and problem-solving skills by developing algorithms that address complex graph-related challenges, including finding cycles, determining graph components and girth, and implementing solutions like Kruskal's Algorithm and Fleury's Algorithm.

COURSE TITLE: Artificial Intelligence									
Course Code:				CSE20515			**Examination Scheme**		
Total number of Lecture Hours: 52							**External**	**80**	
							Internal	**20**	
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**		**0**	**Total Credits**		**4**

Course Objectives

- Understand the structure and function of biological neural networks and their analogy to artificial neural networks (ANN), including the concepts of perceptrons and multilayer neural networks, and their applications.
- Explore various inductive learning algorithms such as ID3, AQ, and RULES, and learn how to apply these algorithms for rule extraction and solving real-world problems.
- Gain a solid understanding of fuzzy logic principles, including fuzzification, fuzzy set operations, and fuzzy inferencing techniques, and their applications in handling uncertainty in complex systems.
- Apply artificial intelligence techniques to enhance the accuracy and efficiency of biometric recognition systems, specifically in fingerprint, face, and iris recognition.

Course Content	TEACHING HOURS
UNIT 1: Introduction to Artificial Neural Networks (ANN)	**14 Hrs**
Introduction to biological neural networks. Artificial neural networks (ANN). Analogy between biological and artificial neural networks. Neuron as a basic building element of an ANN. Activation functions. Perceptron. Learning with a perceptron. Limitations of a perceptron. Multilayer neural networks. Learning with a multilayer perceptron. Backpropagation algorithm. Synergistic neural networks. Distributed neural networks. Distributed and synergistic neural networks. Applications of ANNs. .	
UNIT 2: Inductive Learning Algorithms	**13 Hrs**
Inductive learning algorithms. Categories of inductive learning algorithms. Rule extraction with inductive learning algorithms. ID3 algorithm. AQ algorithm. RULES algorithms. SAFARI algorithm. Applications of inductive learning algorithms.	
UNIT 3: Fuzzy Logic and Uncertainty	**13 Hrs**

Fuzzy logic and uncertainty. Fuzzification. Linguistic terms. Fuzzy sets. Hedges. Fuzzy Hedge Operations. Fuzzy set operations. Fuzzy vector matrix multiplication. Fuzzy Max-Min inferencing. Fuzzy Max-Product inferencing. Multiple premise fuzzy inferencing. Fuzzy multiple rule aggregation. Defuzzification. Applications of fuzzy logic.	
UNIT 4: Artificial Intelligence in Biometric Recognition	**13 Hrs**
Artificial intelligence techniques in fingerprint, face, and iris recognition	

Textbooks
4. Artificial Intelligence: A Modern Approach by Stuart Russell.
Reference Books
1. Artificial Intelligence: A Guide to Intelligent Systems by Michael Negnevitsky 2. Machine Learning by Tom Mitchell 3. Selected Journal and Conference Papers
COURSE OUTCOMES (CO): **CO1:** Apply backpropagation and other learning algorithms to train multilayer neural networks for pattern recognition tasks, improving the network's predictive accuracy. **CO2:** Evaluate and compare the effectiveness of different inductive learning algorithms like ID3 and AQ in terms of rule extraction quality and computational efficiency in various datasets. **CO3:** Design and implement fuzzy inference systems to solve complex decision-making problems in uncertain environments, enhancing system robustness and flexibility. **CO4:** Analyze and optimize AI-based biometric recognition systems to improve the accuracy and reliability of fingerprint, face, and iris recognition technologies in real-world applications.

COURSE TITLE: Advanced Data Communication

Course Code:				CSE205161		**Examination Scheme**	
Total number of Lecture Hours: 52						**External**	**80**
						Internal	**20**
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives

- Understand and quantify channel capacity for both noiseless and noisy channels, applying Nyquist and Shannon's laws to real-world communication systems such as digital telephone networks.
- Explore WAN technologies, including traditional packet and circuit switching, and understand the principles of transmission media, with a focus on the advantages and applications of optical networks.
- Analyze various data encoding and modulation techniques, including NRZ, ASK, FSK, PSK, and PCM, to comprehend their applications in modern communication systems.
- Learn and apply error detection and correction techniques, such as parity checks and CRC, and understand the concepts of multiplexing and spread spectrum techniques to ensure reliable data transmission.

Course Content	TEACHING HOURS
UNIT 1: Fundamentals of Communication Systems	**15 Hrs**
Bandwidth and Channel Capacity. Quantifying Channel Capacity for noiseless channel(Nyquist Law) and noisy channel(Shannon's Law). Example of a digital telephone system to explain basic concepts of analog signals, digital signals, sampling. Data Rate versus Baud Rate. Nyquist Criterion for Sampling. Signal-to-Noise ratio. Local area network(LAN) concepts and characteristics.	
UNIT 2: Wide Area Networks and Transmission Media	**13 Hrs**
Wide area networks(WANs). WAN technologies (traditional packet and circuit switching, Frame Relay, ATM). ISDN(narrowband) concepts and services. Overview of the OSI model. Transmission media – factors affecting distance and data rate. Guided transmission media: Twisted-Pair, Co-axial Cable. Principles and advantages of optical networks. Types of optical fibers and lasers.	
UNIT 3: Data Encoding and Modulation Techniques	**13 Hrs**

Unguided transmission media: Terrestrial Microwave & Satellite Microwave systems and applications. Data encoding. Difference between modulation and encoding. NRZ-L, NRZ-I encoding. Multilevel Binary and Biphase Coding techniques and their implementations. ASK,FSK,PSK and QPSK. PCM concepts: sampling, quantization. Amplitude Modulation.	
UNIT 4: Reliable Data Transmission and Multiplexing	**13 Hrs**
Reliable transmission of data: Asynchronous and Synchronous transmission. Error detection: Parity- based, CRC-based. FCS computation. Error control and recovery techniques. Concept of ARQ standard and its versions. Concept of Multiplexing. FDM. Synchronous and Statistical TDM. Spread Spectrum Techniques: Direct Sequence and Frequency Hopping.	

Textbooks

1. William Stallings, "Data and Computer Communications", 8th Edition, Pearson Education.
2. Behrouz Fourouzan " Data Communications & Networking", 4th Edition, TMH.

Reference Books

1. Andrew Tanenbaum, "Computer Networks", Pearson Education 4/e.
2. Ulysses Black, "Principles of Data Communications ", PHI.
3. Morley, Gelber, "The Emerging Digital Future", Addison-Wesley.

COURSE OUTCOMES (CO):

CO1: Demonstrate the ability to calculate and compare the data rate and baud rate for various communication systems, applying the Nyquist sampling criterion and understanding its impact on signal transmission.

CO2: Critically evaluate different WAN technologies and their underlying principles, such as Frame Relay and ATM, in terms of their suitability for specific communication scenarios, including factors like distance and data rate.

CO3: Analyze and differentiate between various encoding techniques like NRZ, ASK, and PSK, and demonstrate their practical implementation in communication systems, understanding their advantages and limitations.

CO4: Apply error detection and correction methods such as CRC and ARQ in practical communication scenarios, ensuring reliable data transmission across different types of networks and understanding the principles of multiplexing techniques like FDM and TDM.

COURSE TITLE: Advanced Database Management Systems							
Course Code:				**CSE205162**		**Examination Scheme**	
Total number of Lecture Hours: 52						**External**	**80**
						Internal	**20**
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives

- Understand and apply object-oriented concepts, including object identity, complex data types, and type hierarchies, in the design and implementation of object-based database systems.
- Develop proficiency in modeling temporal data and relationships, applying temporal constraints, and using temporal query languages to manage and retrieve temporal data effectively.
- Analyze and implement parallel processing techniques, such as partitioning, intraoperator, and inter-operator parallelism, to optimize query execution in parallel database systems.
- Design and manage distributed databases, focusing on data fragmentation, replication, and allocation techniques, while understanding the challenges and solutions related to concurrency control, recovery, and the use of NOSQL databases.

Course Content	TEACHING HOURS
UNIT 1: Object Based Database Systems	**16 Hrs**
Object Database Concepts Overview: Object Oriented Concepts and Features, Object Identity, Complex data types, Encapsulation of Operations and Object Persistence, Type Hierarchies and Inheritance. Object Based Extensions to SQL: User-Defined Types using CREATE TYPE and Complex Objects ODMG Object Model and the Object Definition Language. .	
UNIT 2: Temporal Database Systems	**13 Hrs**
Temporal Data model: Conceptual Objects, Temporal Objects, temporal Constraints, Temporal and Non Temporal Attributes, Conceptual Relationships, Temporal Relationships and constraints among relationships. The Temporal Query Language: Temporal Projection, Temporal Selection, Temporal Version Restriction Operators, Temporal Scope Operators.	
UNIT 3: Parallel Database Systems	**13 Hrs**

I/O Parallelism: Partitioning Techniques, Managing Skew. Interquery Parallelism and Intraquery Parallelism, Intra-operator Parallelism (Parallel Sort and Parallel Join). Inter-operator Parallelism: Pipelined Parallelism and Independent Parallelism Query Optimization.	
UNIT 4: Distributed Database Systems	**13 Hrs**
Distributed Database Concepts. Data Fragmentation, Replication and Allocation Techniques For Distributed Database Design, Concurrency Control and Recovery. NOSQL Databases: Introduction, the CAP theorem, Document based NOSQL systems and MongoDB, NOSQL Key-Value Stores, Column Based NOSQL Systems, NOSQL Graph Databases and Neo4j.	

Textbooks

1. Advanced Database Systems by Nabil R. Adam and Bharat K . Bhargava, ISBN 3-54057507-3 Springer-Verlag Berlin Heidelberg New York

Reference Books

3. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 7thEdition, Pearson Education, 2017
4. ADVANCED DATABASE SYSTEMS by Dr.John Kandiri
5. Abraham Silberschatz, Henry F. Korth, S.Sudarshan, "Database System Concepts", 6th Edition, 2014

COURSE OUTCOMES (CO):

CO1: Evaluate and compare the effectiveness of object-based extensions to SQL in handling complex data structures and operations within modern database systems.

CO2: Critically analyze temporal relationships and constraints within temporal databases to ensure accurate and efficient data management over time.

CO3: Assess the impact of various parallelism strategies on query performance and scalability in large-scale database systems.

CO4: Investigate the trade-offs and challenges associated with data fragmentation, replication, and concurrency control in distributed and NOSQL database environments.

COURSE TITLE: Engineering Mathematics								
Course Code:			CSE205163			**Examination Scheme**		
Total number of Lecture Hours: 52						**External**	**80**	
						Internal	**20**	
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**	

Course Objectives

- Understand and apply concepts of matrices, including matrix operations, properties, and applications such as eigenvalues, eigenvectors, and quadratic forms.
- Develop a solid foundation in probability theory, including the concepts of random variables, probability distributions, and the relationships between multiple random variables.
- Explore Gaussian processes, multivariate Gaussian distributions, and their applications in regression, with a focus on covariance matrices and linear transformations.
- Analyze convex sets and functions, and apply principles of convex optimization, including Lagrange duality, to solve optimization problems using KKT conditions.

Course Content	**TEACHING HOURS**
UNIT 1: Linear Algebra and Matrix Operations	**17 Hrs**
Linear Algebra –Basic Concepts , Matrices , multiplication , operation and properties, Identity matrices , diagonal matrices, Transpose matrices , Symmetric matrices , Trace , Linear Independence and Rank , Inverse and Orthogonal matrices, Range and Nullspace of a matrix, Determinant, Quadratic forms and Positive SemiDefinite Matrices, Eigenvalue and Eigen Vectors, The Gradient, Hessian , Gradient and Hessian of linear and Quadratic functions. Least Squares, Gradient of the Determinant, Eigen Values as Optimization.	
UNIT 2: Probability and Random Variables	**13 Hrs**
Elements of Probability, Random Variables, Cumulative Distribution functions, Probability mass function, Probability density function, Expectation, Variance, Two random variables, Conditional distributions, Bayes Rule, Independence, Expectation and co-variance, Multiple Random variables, Random vectors.	
UNIT 3: Gaussian Processes and Multivariate Analysis	**13 Hrs**

Gaussian Processes, Multivariate Gaussian, Binary Linear Regression, The squared exponential Kernel, Gaussian Process regression, Multivariate Gaussian Distribution. The co-variance matrix, The diagonal co-variance matrix, Iso-contours, Linear Transformation interpretation.	
UNIT 4: Convex Optimization and Lagrange Duality	**13 Hrs**
Convex sets, Convex functions, Jensen's Inequality, Sublevel sets, Convex Optimization Problems, Special Cases. Lagrange Duality, Lagrangian, Primal and Dual Problems, Complementary slackness, The KKT Conditions.	

Textbooks

1. Linear Algebra and its applications by David C. Lay , Addison Wesley

Reference Books

1. Probability Theory and Stochastic Processes with applications by Oliver Knill – Overseas Press.
2. Applied Multivariate Statistical Analysis by Richard A. Johnson and Dean W. Wichern – PHI
3. Multivariate Data Analysis by Joseph F. Hair, William C. Black , babin and Anderson – Pearson
4. Convex Optimization Theory by Dimitri P. Bertsekas
5. Combinatorial Optimization Algorithms and Complexity by Papadimitrion and Kenneth Steiglitz

COURSE OUTCOMES (CO):

CO1: Students will be able to analyze and interpret the role of eigenvalues and eigenvectors in optimizing quadratic forms and other mathematical models.

CO2: Students will be capable of applying Bayes' Rule and conditional distributions to solve complex problems involving random variables and their dependencies.

CO3: Students will be able to evaluate the performance of Gaussian Process regression models in various machine learning tasks, focusing on the interpretation of covariance structures.

CO4: Students will be able to critically assess optimization problems using the KKT conditions, demonstrating the ability to determine primal and dual feasibility in real-world scenarios.

COURSE TITLE: Object Oriented Methods & Design							
Course Code:				CSE205164		**Examination Scheme**	
Total number of Lecture Hours: 52						**External**	**80**
						Internal	**20**
Lecture (L):	**4**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives

- Understand and apply the foundational concepts of Object-Oriented Analysis and Design (OOAD), including the Unified Process (UP) framework and iterative development methodologies, to initiate and manage software development projects.
- Develop comprehensive use case models by identifying primary actors, goals, and writing use cases in a UI-free style to capture functional requirements, ensuring alignment with the project's objectives and scope.
- Create and interpret System Sequence Diagrams (SSDs) and Domain Models to accurately represent system behavior, interactions, and conceptual classes, enhancing the design and analysis phases of software development.
- Apply GRASP (General Responsibility Assignment Software Patterns) principles and GoF (Gang of Four) Design Patterns in the creation of interaction diagrams, sequence diagrams, and class diagrams, ensuring a robust and scalable software architecture.

Course Content	**TEACHING HOURS**
UNIT 1: Introduction to OOAD and UML	**18 Hrs**
OOAD – Introduction, Applying UML and Patterns in OOAD, Assigning Responsibilities, what is analysis and Design, An Example, The UML, Iterative Development–a Unified Process idea, Additional UP Best Practices and Concepts, The UP Phases and Schedule oriented Terms, The UP disciplines. Process Customization and the development case. The Agile UP. The Sequential Waterfall Lifecycle. Inception. Artifacts that may start in inception, Understanding requirements, types of requirements. .	

UNIT 2: Use Case Modeling and Requirement Analysis	**13 Hrs**
Use –case Model, Writing requirements in context, goals and stories, background, use cases and adding value, use cases and functional requirements, use case types and formats. Goal and scope of a use case, Finding primary actors, goals and use cases, writing use cases in an essential UI-free style, Actors, Use Case Diagrams, Use Cases writing the UP, Case Study. Identifying other requirements. From inception to elaboration.	

UNIT 3: System Sequence Diagrams and Domain Modeling	**13 Hrs**
Use Case Model: Drawing System Sequence Diagrams. Example of an SSD. Inter System SSDs, SSDs and Use Cases, System Events and the System Boundary, Name System Events and Operations, Showing Use Case Text, SSDs within the UP. Domain Model: Visualizing Concepts, Domain Models, Conceptual Class Identification, Candidate Conceptual classes, Adding Associations, The UML association notation, NextGen POS Domain Model Associations, NextGen POS Domain Model, Adding Attributes, Non Primitive Data Type Classes, Adding Detail with Operation Contracts, Contract Sections, Post Conditions, Contracts, Operations and the UML. Operation Contracts within the UP.	
UNIT 4: Transitioning from Requirements to Design with GRASP and Design Patterns	**13 Hrs**
From Requirements to Design, Interaction Diagram Notation, Sequence and Collaboration Diagrams, GRASP, Responsibilities and methods, interactions diagrams, Patterns, GRASP: Pattern of General Principles in Assigning Responsibilities, Information Expert, creator, Low Coupling, High Cohesion, Controller, Object Design and CRC Cards, Design Model: Use Case Realization with GRASP Patterns, Determining Visibility, Creating Design Class Diagrams, Mapping Design to Code. GRASP: More Patterns , Polymorphism , Pure Fabrication , Indirection , Protected Variations , GoF Design Patterns : Adapter , Factory , Singleton , Strategy , Façade , Observer / Publish-Subscribe / Delegation Event Model ,Relating Use Cases , Modeling Generalization , Refining the Domain Model , Adding New SSDs and Contracts , Modeling Behaviour in Statechart Diagrams.	

Textbooks

1. Craig Larman,"Applying UML and Patterns", PHI

Reference Books

1. James Rumbaugh, "Object Oriented Models and Design" Pearson Education 2/e Harrington."
2. C & Object Oriented Paradigm" John Viley & sons Publication
3. Ali Bahrani "Object Oriented Systems Development" McGraw -Hill 1999 4. Lafore Robert, "Object Oriented Programming in C++", Galgotia Publications.

5. Balagurusami, E, "Object Oriented with C++", Tata McGraw-Hill.

COURSE OUTCOMES (CO):

CO1: Students will be able to critically analyze and compare different software development lifecycle models, including iterative and agile methodologies, to determine their suitability for various project requirements.

CO2: Students will demonstrate the ability to effectively communicate functional requirements through well-structured use case models, ensuring clarity in software design documentation.

CO3: Students will be able to interpret and construct system sequence diagrams and domain models, identifying key system events and operations essential for accurate system design.

CO4: Students will gain the ability to apply GRASP principles and design patterns to solve complex object-oriented design problems, enhancing their skills in creating scalable and maintainable software architectures.

Semester – II

COURSE TITLE: Network Security and Cryptography								
Course Code:				**CSE20521**		**Examination Scheme**		
Total number of Lecture Hours: 46						**External**	**80**	
						Internal	**20**	
Lecture (L):	**46**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**	

Course Objectives:

- To understand the OSI Security Architecture and identify common types of security threats, vulnerabilities, and controls.
- To learn about core security services such as confidentiality, integrity, availability, authentication, access control, and non-repudiation.
- To introduce fundamental concepts in cryptography, including number theory, encryption techniques, and cryptographic algorithms.
- To explore advanced topics in network security such as IP security, intrusion detection systems, and defense mechanisms against DDoS attacks.

Course Content	**TEACHING HOURS**
UNIT 1: Unit Heading	**12 Hrs**
Unit 1: Part 1: The OSI Security Architecture, Security Attack – Threats, Vulnerabilities, and Controls, Types of Threats (Attacks) Part 2: Security Services – Confidentiality, Integrity, Availability, Authentication, Access Control and Non repudiation; Security Mechanism. Part 3: Introduction to Number Theory: Prime Number Generation and Testing for Primality, Fermat's and Euler's Theorems, Modular Arithmetic, Euclidean and Extended Euclidean Algorithm, Euler's Phi Function.	
UNIT 2: Unit Heading	**12 Hrs**
Part 1: Introduction to Cryptology. Types of Encryption Systems – Based on Key, Based on Block; Confusion and Diffusion; One-time pad, Block Ciphers and Data Encryption Standard. Part 2: Block Cipher Modes of operation, Advanced Encryption Standard. Stream Ciphers, Random Number Generation. Shift Register based stream Ciphers, RC4. Part 3: Public-Key Cryptography. RSA Cryptosystem	
UNIT 3: Unit Heading	**12 Hrs**
Part 1: Double and Triple Encryption. Key Management, Diffie-Hellman Key ExchangePart 2: Digital Signatures, The RSA signature scheme, Hash Functions, The Secure Hash Algorithm SHA-1. Part 3: Message Authentication Codes, HMAC and CBC-MAC, Message Digest	
UNIT 4: Unit Heading	**12Hrs**
Part 1: IP Security, Authentication Header, Encapsulating Security Payload, Electronic Mail Security.Part 2: Network intrusion Detection system using machine learning: Supervised and Unsupervised.General IDS model and Taxonomy. IDS Signatures. Part 2: DDoS Attacks. Specification and rate based DDoS. Defending against DoS attacks in scout: signature based solutions.	

Textbooks
• Paar, Christof, and Jan Pelzl. Understanding cryptography: a textbook for students and practitioners.Springer Science & Business Media, 2009. • William, S., and Cryptography Stalling. "Network Security, 4/E." Prentice Hall. (2006).
Reference Books
• Forouzan, Behrouz A., and Debdeep Mukhopadhyay. Cryptography and network security (Sie). McGraw-Hill Education, 2011. • Endorf, C., Schultz E and Mellander J, "Intrusion Detection and prevention". McGraw Hill. 2003
COURSE OUTCOMES (CO): • Students will be able to identify and analyze various security threats and vulnerabilities in a system. • Students will demonstrate knowledge of different encryption systems and apply cryptographic algorithms for securing data. • Students will gain proficiency in key management techniques, digital signatures, and message authentication codes. • Students will be able to design and implement solutions for network security, including intrusion detection systems and defense strategies against cyber-attacks.

COURSE TITLE: Lab Network Security and Cryptography								
Course Code:				**CSE20522**		**Examination Scheme**		
Total number of Lab Hours:26 hrs						**External**	**40**	
						Internal	**10**	
Lecture (L):	**-**	**Practicals(P):**	**4**	**Tutorial (T):**	**-**	**Total Credits**	**2**	

Course Objectives:

- To gain practical knowledge of network security tools and techniques for monitoring and protecting computer networks.
- To learn how to analyze network traffic and detect potential threats using tools like Wireshark and SNORT.
- To develop skills in performing cryptographic operations using classical encryption techniques and modern algorithms.
- To understand and implement various methods of network penetration testing and vulnerability assessments.

- **Week 1 :** Experiment 1: Using Wireshark, Demonstrate Packet Sniffing for Router Traffic
- **Week 2:** Experiment 2: Demonstrate Intrusion Detection System using SNORT
- **Week 3:** Experiment 3: Perform Wireless Audit of an Access Point and Decrypt WEP and WPA
- **Week 4:** Experiment 4: Using KF Sensor, Setup a Honey Pot and Monitor the Honeypot on Network.
- **Week 5:** Experiment 5: Using NMAP,
  Find Open Ports on a system Machine that are Active Version of operating System
- **Week 6:** Experiment 6: Implement Ceaser Cipher Encryption Decryption
- **Week 7:** Experiment 7: Implement Hill Cipher Encryption Decryption
- **Week 8:** Experiment 8: Implement Playfair Cipher Encryption Decryption
- **Week 9:** Experiment 9: Implement Vigenere Cipher Experiment
- **Week 10:** Experiment10: Implement Rail Fence (Row Column Transformation) **Week 11:** Experiment 1: Implement RSA Algorithm.

COURSE OUTCOMES (CO):

- Students will be able to use packet sniffing tools to monitor and analyze network traffic and detect suspicious activities.
- Students will demonstrate the ability to configure and deploy intrusion detection systems and honeypots for network security.
- Students will acquire hands-on experience in implementing various cryptographic algorithms, including classical ciphers and public-key encryption.
- Students will be able to perform network scans, identify open ports, and assess vulnerabilities in a networked environment using tools like NMAP.

COURSE TITLE: Image Processing

Course Code:				**CSE20523**		**Examination Scheme**	
Total number of Lecture Hours: 40						**External**	**80**
						Internal	**20**
Lecture (L):	**40**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives:

- To understand the fundamental concepts and components of digital image processing (DIP).
- To explore techniques for image enhancement using spatial and frequency-domain filters.
- To introduce methods for image restoration and noise reduction in digital images.
- To learn the principles and techniques of image compression, including both lossless and lossy compression models.

Course Content	TEACHING HOURS
UNIT 1:	**10 Hrs**
Introduction Digital Image processing, Origins of DIP, Examples, Fundamental steps in DIP, Components of DIP. Fundamentals Elements of visual perception, Light and the electromagnetic spectrum, Image Sensing and acquisition, Image sampling and quantization, basic relationships between pixels	
UNIT 2:	**10 Hrs**
Image Enhancement Background, some basic gray level transformation, Histogram processing, enhancement using arithmetic /Logic operation, Basics of Spatial filtering, smoothing spatial filters, sharpening spatial filters	
UNIT 3:	**10 Hrs**
Image enhancement Background , Introduction to the Fourier transform and the frequency domain, smoothing frequency- domain filters, sharpening frequency domain filters, homomorphic filters & implementation	
UNIT 4:	**10Hrs**
Image restoration Noise models, restoration in the presence of noise only – spatial filtering, Periodic noise reduction by frequency domain filtering. Inverse filtering Image compression Fundamentals. Image compression models, error free compression,lossy compression	

Textbooks

- Digital Image Processing by Woods & Gonzalez
- Digital Image Processing, Kenneth R Castleman, Pearson Education,1995.

Reference Books
• Digital Image Procesing, S. Jayaraman, S. Esakkirajan, T. Veerakumar, McGraw Hill Education, 2009. Pvt Ltd, NewDelhi • Fundamentals of Digital image Processing, Anil Jain.K, Prentice Hall of India, 1989. Image Processing, Sid Ahmed, McGraw Hill, New York, 1995
COURSE OUTCOMES (CO): • Students will be able to explain the basic principles of digital image processing and its applications. • Students will demonstrate the ability to apply various image enhancement techniques using gray-level transformations, spatial filters, and frequency-domain operations. • Students will acquire skills in restoring images affected by noise and will be able to implement filtering techniques to improve image quality. • Students will gain knowledge of different image compression techniques and be able to apply appropriate methods for efficient image storage and transmission.

COURSE TITLE: Lab Image Processing

Course Code:			**CSE20524**		**Examination Scheme**	
Total number of Lab Hours:26 hrs					**External**	**40**
					Internal	**10**
Lecture (L):	**-**	**Practicals(P):**	**3**	**Tutorial (T):** **-**	**Total Credits**	**2**

Course Objectives:

- To introduce the basic concepts of image processing, including pixel operations, image transformations, and filtering techniques.
- To develop a deep understanding of advanced image processing methods like Fourier transforms, image enhancement, and segmentation.
- To explore image compression, color image processing, and morphological operations for improved image analysis.
- To gain practical skills in implementing image processing algorithms using MATLAB for applications like edge detection, object recognition, and image restoration.

Experiments/ Labs

- Basics of an Image Processing (reading an image to mat lab, display pixel operations, flipping and cropping).
- Viewing digital images, bits and bytes, raster scan format, quantization, Scaling, translation and rotation, sums and differences
- Histograms and stretches, convolutional filters
- Fourier transforms and the frequency domain, filters
- FFTs, Image filtering: smoothing and sharpening
- 2D convolution and correlation
- Creating multiple image sequences for the project Image enhancement.
- Image compression
- Color image processing
- Image segmentation
- Image Morphology
- Image Restoration
- Edge detection in an Image
- Blurring 8 bit color versus monochrome
- Object Reorganization like circles and triangle.

COURSE OUTCOMES (CO):

- Students will be able to read, manipulate, and perform basic operations on images using MATLAB, such as flipping, cropping, and pixel-level transformations.
- Students will acquire knowledge in applying image filtering techniques, including convolution, correlation, and frequency-domain filtering, to enhance image quality.
- Students will be able to implement advanced image processing tasks like segmentation, morphological operations, and image restoration for various applications.
- Students will demonstrate the ability to recognize and classify objects in images using techniques like edge detection, shape analysis, and feature extraction.

COURSE TITLE: Machine Learning

Course Code:				**CSE20525**		**Examination Scheme**	
Total number of Lecture Hours: 40						**External**	**80**
						Internal	**20**
Lecture (L):	**40**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives:

- To provide a comprehensive understanding of clustering algorithms and techniques for evaluating cluster validity.
- To explore the principles and applications of Support Vector Machines (SVMs) for solving linear and non-linear classification problems.
- To introduce dimensionality reduction techniques such as Principal Component Analysis (PCA) and Fisher Linear Discriminant for feature extraction and data analysis.
- To study advanced topics in neural networks, including Convolutional Neural Networks (CNNs), and their applications in image recognition and other fields.

Course Content	TEACHING HOURS
UNIT 1:	**10 Hrs**
Clustering Algorithms, Euclidean and Mahalanobis Distances, Basic Sequential Algorithm Scheme, K-Means Algorithm, Fuzzy C-Means Clustering, Clustering with Gaussian Probability Density Function. Cluster Validity index. Compactness Cluster Measure, Distinctness Cluster Measure, Validity Index Using Standard Deviation, Point Density Based Validity Index, Validity index using Local and Global Data Spread,	
UNIT 2:	**10 Hrs**
Support Vector Machines. Binary Linear Support Vector Machines, Optimal Hyperplane, Canonical Form, Kernel Functions, Solving Non-linear Classification problems with Linear Classifier. Multiclass Suport Vector Machines, Directed Acyclic Graph Support Vector Machines. Application of Support Vector Machines.	
UNIT 3:	**10 Hrs**
Dimensionality Reduction, Principal Component Analysis, Fisher Linear Discriminant, Multiple Discriminant Analysis. Watershed Based Clustering. SubSpace Grid Based Approach. Coarse and Fie Rule Extraction using Sub-Space Grid Based Approach for Clustering.	
UNIT 4:	**10Hrs**
Convolutional Neural Network Architectures and applications.	

Textbooks

- Machine Learning by Tom M. Mitchel, McGraw-Hill publication Pattern Classification by Duda and Hart. John Wiley publication
- Introduction to Machine Learning by EthemAlpaydin, The MIT Press.
- Machine Learning: An Algorithmic Perspective by Stephen Marsland, Chapman and Hall/CRC.

Reference Books

- Advances in Deep Learning, M. Arif Wani,
- The Elements of Statistical Learning: Data Mining, Inference, and Prediction by Trevor Hastie, Robert Tibshirani, Jerome Friedman, Springer.
- Learning From Data, Yaser S. Abu-Mostafa, Hsuan-Tien Lin, Malik Magdon-Ismail, AML Book.

COURSE OUTCOMES (CO):

- Students will gain proficiency in implementing clustering algorithms like K-Means and Fuzzy C-Means and evaluate clustering performance using various validity indices.
- Students will be able to apply Support Vector Machines to binary and multi-class classification problems and understand how kernel functions can solve non-linear classification challenges.
- Students will acquire knowledge of dimensionality reduction techniques and apply methods like PCA and Fisher Linear Discriminant for feature selection and data compression.
- Students will develop practical skills in designing and applying Convolutional Neural Networks for solving real-world problems, especially in image processing and pattern recognition.

COURSE TITLE: Advanced Algorithms								
Course Code:			**CSE205261**			**Examination Scheme**		
Total number of Lecture Hours: 46						**External**	**80**	
						Internal	**20**	
Lecture (L):	**46**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**	

Course Objectives:

- To understand the fundamental concepts of algorithm design, analysis, and growth functions, including asymptotic notations.
- To explore advanced data structures like Red-Black Trees, B-Trees, and their applications in efficient algorithm implementation.
- To introduce various algorithmic paradigms, including dynamic programming, multithreaded algorithms, and network flow algorithms.
- To study specialized algorithms for string matching and computational geometry, focusing on solving real-world problems.

Course Content	**TEACHING HOURS**
UNIT 1:	**12 Hrs**
Introduction to Algorithms, Analysis of algorithms, Designing Algorithms, Growth of Functions, Asymptotic notations (5L) Recurrences, The Master Method, (2L) Probabilistic Analysis. The hiring problem, Indicator random variables, randomized algorithms (5L)	
UNIT 2:	**12 Hrs**
Advanced Data structures: Red-Black Trees, B Trees, Binomial Heap, Augmenting Data Structures, Interval trees (6L) Network Flow Algorithm: Flow networks, Ford_fulkerson method, Pre-flow push algorithm, push relabel algorithms (6L).	
UNIT 3:	**12 Hrs**
Dynamic Programming: elements of dynamic programming, Rod cutting (2L) Multithreaded Algorithms: Basics of dynamic multithreading, Multithreaded matrix multiplication, Multithreaded merge sort (6L) Matrix operations: Strassens multiplication algorithm, inverting matrices (4L).	
UNIT 4:	**12Hrs**
String Matching: The naive string-matching algorithm, The Rabin-Karp algorithm, String matching with finite automata, The Knuth-Morris-Pratt algorithm (6L) Computational Geometry: Line-segment properties, determining whether any pair of segments intersects, Finding the convex hull, Finding the closest pair of points (6L)	

Textbooks

- Coremen, Leiserson, Rivest,Stein, "Introduction to Algorithms", Third edition, PHI
- Horowitz, Sahni, Rajasekaran "Fundamentals of Computer Algorithms", Galgotia Publications

Reference Books

Michael T. Goodrich, Roberto Tamassia "Algorithm Design and Applications" , Wiley

COURSE OUTCOMES (CO):

- Students will gain proficiency in analyzing and designing algorithms using various methods such as recurrences, the Master Method, and probabilistic analysis.
- Students will develop the ability to implement advanced data structures and apply them to optimize algorithmic solutions for complex problems.
- Students will acquire practical knowledge in using dynamic programming, multithreaded algorithms, and matrix operations to solve computational problems efficiently.
- Students will demonstrate skills in applying algorithms for string matching and computational geometry to address challenges like pattern matching, convex hull determination, and point proximity detection.

COURSE TITLE: Software Reliability Engineering

Course Code:			**CSE205262**			**Examination Scheme**	
Total number of Lecture Hours: 46						**External**	**80**
						Internal	**20**
Lecture (L):	**46**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives:

- To introduce fundamental reliability concepts and measures, including reliability analysis techniques like block diagrams, fault tree analysis, and Monte Carlo simulation.
- To explore advanced reliability models such as Nonhomogeneous Poisson Processes (NHPP) and their application in software reliability modeling.
- To study various execution time and debugging models, focusing on reliability in software systems, including imperfect debugging and modular software models.
- To analyze different S-shaped NHPP models and other advanced reliability models to assess fault complexity, error removal, and system performance.

Course Content	TEACHING HOURS
UNIT 1:	**12 Hrs**
BASIC RELIABILITY CONCEPTS: Reliability Measures (Definition of reliability, Mean time to failure (MTTF), Failure rate function, Maintainability and availability), Common Techniques in Reliability Analysis (Reliability block diagram, Network diagram, Fault tree analysis, Monte Carlo simulation), Markov Process Fundamentals (Stochastic processes, Standard Markov models, General procedure of Markov modelling	
UNIT 2:	**12 Hrs**
Nonhomogeneous Poisson Process (NHPP) Models (General formulation, Reliability measures and properties, Parameter estimation); MODELS FOR SOFTWARE RELIABILITY: Basic Markov Model (Model description, Parameter estimation). Execution Time models: Basic execution time model, logarithmic Poisson model;	
UNIT 3:	**12 Hrs**
: Imperfect debugging models (Monotonous death process, Birth-death process, Imperfect debugging model considering multi-type failure), Modular Software Systems: The Littlewood semiMarkov model; Software NHPP Models: Calender time models: Goel-Okumoto (GO) model, Hyperexponential model, exponential fault categorization model;	
UNIT 4:	**12Hrs**
S-shaped NHPP models: Delayed S-shaped NHPP model, Inflected S-shaped NHPP model; Failure rate dependent flexible model, SRGM for error removal phenomenon, SRGM defining Complexity of faults, generalized SRGM(Erlang model), Incorporating fault complexity considering learning phenomenon; Some other NHPP models: Duane model-Log-power model, Musa-Okumoto model	

Textbooks

- Musa, Iannino, Okumoto, “Software Reliability: Measurement, Prediction, Application”, McGrawHill, 1987.
- Min Xie Yuan-Shun Dai and Kim-Leng Poh, “Computing System Reliability: Models and Analysis “ KLUWER ACADEMIC PUBLISHERS, 2004

Reference Books

- P. K. Kapur, H. Pham, A. Gupta, P. C. Jha,“Software Reliability Assessment with OR Applications”, Springer-Verlag London Limited 2011
- Hoang Pham, “system software reliability”, Springer, 2006
- Michael R. Lyu, “Handbook of software reliability engineering-IEEE Computer Society Press_
- McGraw Hill (1996)”.
- M. Lyu, ed. ”Handbook of Software Reliability Engineering”, McGraw-Hill and IEEE Computer Society Press, 1996 7. Pham, H. (2000). ‘Software Reliability’, Springer-Verlag, Singapore.

COURSE OUTCOMES (CO):

- Students will be able to apply reliability measures and analysis techniques, such as Markov processes and fault tree analysis, to evaluate system reliability.
- Students will gain knowledge of NHPP models and their application in software reliability, including the estimation of parameters and prediction of system performance.
- Students will acquire skills in modeling software reliability using various debugging models, execution time models, and modular software approaches.
- Students will demonstrate the ability to analyze and implement advanced NHPP models for software reliability, incorporating fault complexity, error removal, and learning phenomena in real-world systems.

<table>
<tr><td colspan="8">COURSE TITLE: Optimization Techniques</td></tr>
<tr><td colspan="4">Course Code:</td><td colspan="2">CSE205263</td><td colspan="2">Examination Scheme</td></tr>
<tr><td colspan="6" rowspan="2">Total number of Lecture Hours: 46</td><td>External</td><td>80</td></tr>
<tr><td>Internal</td><td>20</td></tr>
<tr><td>Lecture (L):</td><td>46</td><td>Practicals(P):</td><td>0</td><td>Tutorial (T):</td><td>0</td><td>Total Credits</td><td>4</td></tr>
</table>

Course Objectives:

- To introduce linear programming and optimization techniques, covering both graphical and simplex methods for solving linear problems.
- To explore unconstrained one-dimensional optimization techniques, including search methods and interpolation techniques for finding optimal solutions.
- To study unconstrained multi-dimensional optimization methods, such as random search, pattern search, and descent algorithms like steepest descent and quasi-Newton methods.
- To understand constrained optimization techniques, including conditions for optimality, Kuhn-Tucker conditions, and methods like gradient projection, cutting plane, and penalty function.

Course Content	TEACHING HOURS
UNIT 1:	**12 Hrs**
Linear programming –formulation-Graphical and simplex methods-Big-M method Two phase method-Dual simplex method-Primal Dual problems.	
UNIT 2:	**12 Hrs**
Unconstrained one dimensional optimization techniques -Necessary and sufficient conditions – Unrestricted search methods-Fibonacci and golden section method Quadratic Interpolation methods, cubic interpolation and direct root methods.	
UNIT 3:	**12 Hrs**
Unconstrained n dimensional optimization techniques – direct search methods – Random search – pattern search and Rosen brooch’s hill claiming method- Descent methods-Steepest descent, conjugate gradient, quasi -Newton method.	
UNIT 4:	**12Hrs**
Constrained optimization Techniques- Necessary and sufficient conditions – Equality and inequality, constraints-Kuhn-Tucker conditions-Gradient projection method-cutting plane method- penalty function method .	

Textbooks
Ashok D. Belegundu, Tirupathi R. Chandrupatla, “Optimization Concepts and Applications in Engineering”, Cambridge University Press.
Reference Books
• Rao,S.S.,‟Optimization :Theory and Application‟ Wiley Eastern Press, 2nd edition 1984. • Taha,H.A., Operations Research –An Introduction,Prentice Hall of India,2003. Fox, R.L., „Optimization methods for Engineering Design‟, Addition Welsey, 1971., Singapore.
COURSE OUTCOMES (CO): • Students will be able to formulate and solve linear programming problems using methods like the simplex method, Big-M method, and dual simplex method. • Students will acquire skills in applying one-dimensional optimization techniques, including Fibonacci, golden section, and interpolation methods for function optimization. • Students will demonstrate proficiency in using multi-dimensional optimization techniques, including random search, pattern search, and descent algorithms for solving complex problems. • Students will understand and apply constrained optimization techniques, including the Kuhn-Tucker conditions, gradient projection, and penalty function methods to engineering and real-world problems.

<table>
<tr><td colspan="8">COURSE TITLE: Big Data</td></tr>
<tr><td colspan="4">Course Code:</td><td colspan="2">CSE205264</td><td colspan="2">Examination Scheme</td></tr>
<tr><td colspan="6" rowspan="2">Total number of Lecture Hours: 46</td><td>External</td><td>80</td></tr>
<tr><td>Internal</td><td>20</td></tr>
<tr><td>Lecture (L):</td><td>46</td><td>Practicals(P):</td><td>0</td><td>Tutorial (T):</td><td>0</td><td>Total Credits</td><td>4</td></tr>
</table>

Course Objectives:

- To provide an understanding of the Big Data platform, challenges of conventional systems, and the tools and techniques used for modern data analysis.
- To explore data stream mining concepts, including stream data models, real-time analytics platforms, and applications in sentiment analysis and stock market predictions.
- To introduce Hadoop and its ecosystem, covering HDFS, MapReduce, and the development of big data applications using these tools.
- To gain hands-on experience with Big Data frameworks like Pig, Hive, and HBase, and to learn predictive analytics techniques such as regression analysis and data visualization.

Course Content	TEACHING HOURS
UNIT 1:	**12 Hrs**
Introduction to BigData Platform – Challenges of Conventional Systems - Intelligent data analysis – Nature of Data - Analytic Processes and Tools - Analysis vs Reporting - Modern Data Analytic Tools - Statistical Concepts: Sampling Distributions - Re-Sampling - Statistical Inference - Prediction Error	
UNIT 2:	**12 Hrs**
Mining data streams : 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 3:	**12 Hrs**
Hadoop: History of Hadoop- the Hadoop Distributed File System – Components of Hadoop Analysing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design of HDFS-Java interfaces to HDFS Basics- Developing a Map Reduce Application-How Map Reduce Works-Anatomy of a Map Reduce Job run-Failures-Job Scheduling-Shuffle and Sort – Task execution - Map Reduce Types and FormatsMap Reduce Features, Hadoop environment.	
UNIT 4:	**12Hrs**
Frameworks: Applications on Big Data Using Pig and Hive – Data processing operators in Pig – Hive services – HiveQL – Querying Data in Hive - fundamentals of HBase and ZooKeeper - IBM InfoSphere Big Insights and Streams. Predictive Analytics- Simple linear regression- Multiple linear regression- Interpretation 5 of regression coefficients. Visualizations - Visual data analysis techniquesinteraction techniques - Systems and applications.	

Textbooks

- Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.
- Tom White "Hadoop: The Definitive Guide" Third Edition, O'reilly Media, 2012.
- Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data", McGrawHill Publishing, 2012.

Reference Books

- Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", CUP, 2012.
- Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley& sons, 2012.
- Glenn J. Myatt, "Making Sense of Data", John Wiley & Sons, 2007.
- Pete Warden, "Big Data Glossary", O'Reilly, 2011.

COURSE OUTCOMES (CO):

- Students will be able to comprehend the challenges of Big Data and differentiate between conventional data analysis and modern tools for handling large datasets.
- Students will acquire knowledge in mining data streams, analyzing real-time data, and applying these techniques to real-world scenarios like sentiment analysis and financial predictions.
- Students will develop practical skills in using Hadoop, including HDFS and MapReduce, and will be able to design and execute large-scale data processing tasks.
- Students will demonstrate proficiency in using Big Data frameworks like Pig, Hive, and HBase for data processing and will be able to apply predictive analytics and data visualization techniques to analyze complex datasets.

Semester - III

COURSE TITLE: Minor Project	
Course Code:	**CSE20531** **Examination Scheme**

Minor project to be completed under the supervision of assigned faculty member on a topic to be selected in consultation with the supervisor.

<table>
<tr><td colspan="9">COURSE TITLE: Parallel and Distributed Computing</td></tr>
<tr><td colspan="5">Course Code:</td><td colspan="2">CSE20532</td><td colspan="2">Examination Scheme</td></tr>
<tr><td colspan="7" rowspan="2">Total number of Lecture Hours: 46</td><td>External</td><td>80</td></tr>
<tr><td>Internal</td><td>20</td></tr>
<tr><td>Lecture (L):</td><td>46</td><td>Practicals(P):</td><td>0</td><td colspan="2">Tutorial (T):</td><td>0</td><td>Total Credits</td><td>4</td></tr>
</table>

Course Objectives:

- Develop a solid understanding of the core principles of parallel and distributed computing, including the architecture and performance evaluation of various parallel processing systems.
- Analyze and compare different interconnection topologies and routing methods used in parallel processing systems, focusing on real-world examples and high-performance computing (HPC) systems.
- Gain hands-on experience in shared memory and distributed memory parallel programming using tools such as Pthreads, OpenMP, and MPI, with a focus on performance optimization.
- Explore advanced parallel programming models and technologies, including OpenCL, CUDA C, and distributed object computing, to understand emerging trends in highperformance and cloud computing.

Course Content	TEACHING HOURS
UNIT 1:	**12 Hrs**
Introduction to Parallel and Distributed Computing, Flynn’s Taxonomy of Parallel Architectures: Parallel/Vector Computers, Shared Memory Multiprocessors (UMA, NUMA, COMA), Distributed Memory Multiprocessors, Multivector and SIMD computers, Data Parallel Pipelined and Systolic Architectures, Instruction set Architectures (CISC, RISC, VLIW, superpipelined, vector processors), Performance Evaluation of Computer Systems, PRAM Model of Parallel Computation, PRAM Algorithms: Parallel Reduction, List Ranking, Preorder tree traversal.	
UNIT 2:	**12 Hrs**
Interconnection Topologies and Routing for Parallel Processing Systems: Categorization of Topologies, On-Chip Interconnection Topologies, Supercomputer Interconnection Topologies: Blue Waters, Blue Gene/Q, A case Study of HPC, Topology detection, Comparison of Topologies: The Moore Bound, Routing in Static Networks: Topology independent Routing (Point-to-Point routing, Broadcasting, Gossiping), Topology dependent routing.	
UNIT 3:	**12 Hrs**
Shared Memory Programming with Pthreads (Critical Sections, Busy Waiting, Mutexes, Barriers And Condition Variables, Read-Write Locks), Shared Memory Programming with OpenMP: Cover OpenMP basics, Distributed Memory Parallel Programming: Cover MPI programming basics with simple programs and most useful directives, Collective Communication, Parallel Sorting Algorithm, Performance Evaluation of MPI programs.	
UNIT 4:	**12Hrs**

Advnced Topics: Introduction to OpenCL, Parallel programming with OpenACC, Introduction to Data Parrallelism and CUDA C, Distribuited Object Computing Tools: Basic Models (RMI, CORBA, DCOM), Trends and Visions (Cloud and Grid Computing, P2P computing, Autonomic Computing).	

Textbooks

- Advanced computer architectures, Dezso Sima.
- Advanced computer architecture, Kai Hwang & Naresh Jotwani.
- Parallel Programming for Multicore and Cluster systems, Thomas Rauber Gudula Runger.
- An introduction to parallel programming, Peter S.Pacheco.
- Tools and Environment for Parallel and Distributed Computing, Salim Hariri Manish Parashar.
- Programming Massively Parallel Processors, David Kirk.

Reference Books

- Interconnection Topologies and Routing for Parallel Processing Systems: Gabriele Kotsis, Technical Report Series, ACPC/TR 92-19,1992.
- Topology and Routing Aware Mapping on Parallel Processors, Thesis, Dept. of Mathematics & computer sciences, Sri satya sai institute of high learning.

COURSE OUTCOMES (CO):

- Demonstrate a thorough understanding of parallel and distributed computing architectures, including the ability to classify and evaluate different types of parallel computers.
- Effectively compare and contrast interconnection topologies, and develop efficient routing strategies for parallel processing systems, based on specific application requirements.
- Write and optimize parallel programs using Pthreads, OpenMP, and MPI, and evaluate their performance on shared and distributed memory systems.
- Apply advanced parallel programming techniques using OpenCL, CUDA C, and distributed object computing tools, and critically assess their applicability to modern computing challenges such as cloud and grid computing.

<table>
<tr><td colspan="8">COURSE TITLE: Real-Time Operating Systems</td></tr>
<tr><td colspan="4">Course Code:</td><td colspan="2">CSE20533</td><td colspan="2">Examination Scheme</td></tr>
<tr><td colspan="6" rowspan="2">Total number of Lecture Hours: 46</td><td>External</td><td>80</td></tr>
<tr><td>Internal</td><td>20</td></tr>
<tr><td>Lecture (L):</td><td>46</td><td>Practicals(P):</td><td>0</td><td>Tutorial (T):</td><td>0</td><td>Total Credits</td><td>4</td></tr>
<tr><td colspan="8">Course Objectives:
• Develop a comprehensive understanding of real-time systems, including their fundamental principles, architectures, and the distinctions between hard and soft real-time applications.
• Explore and classify real-time scheduling algorithms, focusing on both clock-driven and priority-driven approaches, to ensure timely and predictable task execution in real-time environments.
• Understand the challenges of resource sharing and synchronization in real-time systems, and study protocols to manage priority inversion and ensure task coordination.
• Investigate the features, standards, and performance benchmarks of real-time operating systems (RTOSs), including case studies and practical applications in various domains.</td></tr>
<tr><td colspan="7">Course Content</td><td>TEACHING HOURS</td></tr>
<tr><td colspan="7">UNIT 1: Introduction</td><td>12 Hrs</td></tr>
<tr><td colspan="7">Basic OS Principles and Structures review; Real-Time Systems – Basic Model, Characteristics, Hard vs. Soft, Applications; Real-Time Reference Model – Tasks and Types; Software Architectures – Petri nets, RTOS Architecture, Real-Time Kernels.</td><td></td></tr>
<tr><td colspan="7">UNIT 2: Real Time Task Scheduling</td><td>12 Hrs</td></tr>
<tr><td colspan="7">Classification of Real-Time Scheduling Algorithms; Common Approaches; Clock Driven; Priority Driven – Earliest Deadline First, Rate Monotonic, Deadline Monotonic; Overview of Real-Time Multiprocessor Scheduling.</td><td></td></tr>
<tr><td colspan="7">UNIT 3: Real-Time Resource Sharing/Synchronization</td><td>12 Hrs</td></tr>
<tr><td colspan="7">Resource Sharing among Real-Time Tasks – Contention and Control; Priority Inversion; Priority Inheritance Protocol; Highest Locker Protocol; Priority Ceiling Protocol.</td><td></td></tr>
<tr><td colspan="7">UNIT 4: Real World RTOSs</td><td>12Hrs</td></tr>
<tr><td colspan="7">Features of RTOSs; UNIX and Windows as RTOSs – Pros and; POSIX Standard; Survey of Contemporary RTOSs – Case Study of any one, Porting to a Target; RTOS Benchmarking; RTOS Application Domains.</td><td></td></tr>
<tr><td colspan="8">Textbooks</td></tr>
<tr><td colspan="8">- Andrew S. Tanenbaum, Modern Operating Systems (Third Edition), Pearson Education.
- David E. Simon, An Embedded Software Primer, Pearson Education.</td></tr>
</table>

Reference Books
• Laplante, P., Real-Time Systems Design and Analysis (Third Edition), IEEE/Wiley Interscience. • Rajib Mall, Real-Time Systems: Theory and Practice (Second Edition), Pearson Education. • Jane W.S. Liu, Real-Time Systems (Sixth Edition), Pearson Education. • Raj Kamal, Embedded Systems: Architecture, Programming and Design (Third Edition), Tata McGraw-Hill Education
COURSE OUTCOMES (CO): • Demonstrate a solid understanding of real-time system characteristics, models, and architectures, and apply this knowledge to analyze and design real-time systems. • Implement and evaluate real-time scheduling algorithms, ensuring the ability to select and apply appropriate scheduling techniques for different real-time system requirements. • Develop strategies for resource sharing and synchronization among real-time tasks, effectively managing issues such as priority inversion using appropriate protocols. • Analyze and compare different RTOSs, including their standards and performance metrics, and apply this understanding to practical scenarios, including RTOS porting and benchmarking.

COURSE TITLE: Advanced Wireless & Mobile Computing

Course Code:				**CSE20534**		**Examination Scheme**	
Total number of Lecture Hours: 46						**External**	**80**
						Internal	**20**
Lecture (L):	**46**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives:

- Develop a foundational understanding of various types of wireless telephones, including their classification and evolution across different generations of mobile phone technologies.
- Learn the essential principles of cellular networks, including cell structures, frequency reuse, and wireless transmission techniques, with a focus on signal propagation and multiple access methods.
- Gain in-depth knowledge of CDMA technology, including forward and reverse channels, Walsh codes, and the operation of IS-95 CDMA, to understand the fundamentals of codedivision multiple access.
- Study the GSM reference architecture, including key network components and their functions, and explore the basics of advanced mobile technologies such as OFDM and LTE.

Course Content	**TEACHING HOURS**
UNIT 1:	**12 Hrs**
Classification and types of Wireless telephones. Introduction to Cordless, Fixed Wireless(WLL), Wireless with limited mobility(WLL-M) and (Fully)Mobile Wireless phones. Introduction to various generations of mobile phone technologies and future trends. Wireline vs. Wireless portion of mobile communication networks. Mobile-Originated vs. Mobile-Terminated calls. Mobile-Phone numbers vs. FixedPhone numbers.	
UNIT 2:	**12 Hrs**
Concept of cells, sectorization, coverage area, frequency reuse, cellular networks & handoffs. Wireless Transmission concepts; types of antennas; concepts of signal propagation, blocking, reflection, scattering & multipath propagation. Comparison of multiple access techniques FDM, TDM and CDM. Concept and use of chipsequences.	
UNIT 3:	**12 Hrs**
Concept of Forward and Reverse CDMA channel for a cell/sector. Concept/derivation of Walsh codes & Code Channels within a CDMA Channel. Simplified illustration of IS-95 CDMA using chip sequences. Purpose of Pilot, Sync, Paging, Forward Traffic Channels. Purpose of Access & Reverse TCs.	

UNIT 4:	12Hrs
GSM reference architecture and components of Mobile Networks: MS, BTS, BSC, MSC; their basic functions and characteristics. Use of HLR and VLR in mobile networks. Handoff scenarios in GSM. Basic Concept of OFDM and LTE technology for mobile networks.	

Textbooks

- K.Pahlavan, P.Krishnamurthy, “Principles of Wireless Networks”, PHI.
- T. Rappaport, “Wireless Communications, Principles and Practice (2nd Edition)”,Pearson.

Reference Books

- Andy Dornan, “The Essential Guide to Wireless Communications Applications”, Pearson.
- Jochen Schiller, “Mobile Communications”, Pearson.

COURSE OUTCOMES (CO):

- Demonstrate a thorough understanding of the types and classifications of wireless telephones, including the differences between wireline and wireless networks, and the significance of mobile phone technologies.
- Apply cellular network concepts such as sectorization, frequency reuse, and multiple access techniques (FDM, TDM, CDM) to real-world scenarios, ensuring efficient wireless communication.
- Utilize knowledge of CDMA channels, Walsh codes, and IS-95 CDMA operations to analyze and troubleshoot CDMA-based wireless systems.
- Analyze the GSM architecture and its components, understand handoff scenarios, and apply basic concepts of OFDM and LTE technologies in the context of modern mobile networks.

COURSE TITLE: Cloud Computing

Course Code:				**CSE205351**		**Examination Scheme**	
Total number of Lecture Hours: 46						**External**	**80**
						Internal	**20**
Lecture (L):	**46**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives:

- To understand the fundamental concepts of cloud computing, including different cloud deployment models (public, private, hybrid) and service models (IaaS, PaaS, SaaS).
- To explore the technologies and processes involved in deploying and managing cloudbased applications and web services.
- To analyze the management aspects of cloud services, including reliability, availability, scalability, and the economic factors influencing cloud platform choices.
- To gain practical knowledge in cloud-based application development and service creation environments, focusing on the benefits and challenges of cloud architecture.

Course Content	TEACHING HOURS
UNIT 1:	**12 Hrs**
CLOUD COMPUTING FUNDAMENTALS (8 hours) Cloud Computing definition; , private, public and hybrid cloud. Cloud types; IaaS, PaaS, SaaS. Benefits and challenges of cloud computing, public vs private clouds, role of virtualization in enabling the cloud; Business Agility: Benefits and challenges to Cloud architecture. Application availability, performance, security and disaster recovery; next generation Cloud Applications.	
UNIT 2:	**12 Hrs**
CLOUD APPLICATIONS (6 hours) Technologies and the processes required when deploying web services; Deploying a web service from inside and outside a cloud architecture, advantages and disadvantages.	
UNIT 3:	**12 Hrs**
MANAGEMENT OF CLOUD SERVICES (12 hours) Reliability, availability and security of services deployed from the cloud. Performance and scalability of services, tools and technologies used to manage cloud services deployment; Cloud Economics: Cloud Computing infrastructures available for implementing cloud based services. Economics of choosing a Cloud platform for an organization, based on application requirements, economic constraints and business needs (e.g Amazon, Microsoft and Google, Salesforce.com, Ubuntu and Redhat)	
UNIT 4:	**12Hrs**
APPLICATION DEVELOPMENT (10 hours) Service creation environments to develop cloud based applications. Development environments for service development; Amazon, Azure, Google App.	

Textbooks

- Gautam Shroff, “Enterprise Cloud Computing Technology Architecture Applications”, Cambridge University Press; 1 edition, [ISBN: 9780521137355], 2010.
- Toby Velte, Anthony Velte, Robert Elsenpeter, “Cloud Computing, A Practical Approach”

Reference Books

- McGraw-Hill Osborne Media; 1 edition [ISBN: 0071626948], 2009.
- Dimitris N. Chorafas, “Cloud Computing Strategies” CRC Press; 1 edition [ISBN: □ 1439834539],2010.

COURSE OUTCOMES (CO):

- Students will be able to describe the key concepts of cloud computing, distinguish between various cloud models, and understand the role of virtualization in cloud architecture.
- Students will develop skills in deploying and managing cloud-based applications, both from within and outside a cloud architecture, while evaluating the advantages and disadvantages.
- Students will acquire the ability to assess the reliability, availability, and security of cloud services and make informed decisions on cloud platform selection based on economic and business needs.
- Students will demonstrate the capability to develop cloud-based applications, utilizing service creation environments and understanding the performance, security, and disaster recovery aspects.

COURSE TITLE: Internet of Things

Course Code:			CSE205352			**Examination Scheme**	
Total number of Lecture Hours: 46						**External**	**80**
						Internal	**20**
Lecture (L):	**46**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives:

- To understand the fundamental concepts of cloud computing, including different cloud deployment models (public, private, hybrid) and service models (IaaS, PaaS, SaaS).
- To explore the technologies and processes involved in deploying and managing cloudbased applications and web services.
- To analyze the management aspects of cloud services, including reliability, availability, scalability, and the economic factors influencing cloud platform choices.
- To gain practical knowledge in cloud-based application development and service creation environments, focusing on the benefits and challenges of cloud architecture.

Course Content	TEACHING HOURS
UNIT 1:	**12 Hrs**
Definition & Characteristics of Iot, Physical Design of Iot, Things in Iot, Iot Protocols; Logical Design Of Iot: Iot Functional Blocks, Iot Communication Models, Iot Communication APIs; IoT Levels and Templates [8 Lectures] Domain Specific IoTs – Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle [4 Lectures]	
UNIT 2:	**12 Hrs**
Wireless Sensor Networks, Cloud Computing, Big Data Analytic, Communication Protocols, Machine to Machine, Difference between IoT and M2M, Software define Network, Embedded Systems [6 Lectures] Design challenges, Development challenges, Security challenges, Other challenges [6 Lectures].	
UNIT 3:	**12 Hrs**
Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. [4 Lectures] Real-World Design Constraints- Introduction, Technical Design constraints-hardware is popular again [2 Lectures] Data representation and visualization, Interaction and remote control. Industrial Automation- Serviceoriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things [6 Lectures]	
UNIT 4:	**12Hrs**
Setting up the Arduino development environment: Options for Internet connectivity, Interacting with basic sensors, Interacting with basic actuators, Configuring Arduino for the IoT [4 Lectures] Grabbing the content from a web page, Sending data to the cloud, Monitoring sensor data from a cloud dashboard, Monitoring several Arduino boards at, Storing data on Google Drive [4 Lectures] Basic local M2M interactions, Cloud M2M with IFTTT; Case Study: IoT based Flood Monitoring and Alert System [4 Lectures]	

Textbooks

- Vijay Madisetti and Arshdeep Bahga, “Internet of Things (A Hands-on-Approach)”, 1stEdition, VPT, 2014. (ISBN-13: 978-8173719547)
- Schwartz, Marco. “Internet of Things with Arduino Cookbook”. Packt Publishing Ltd, 2016.

Reference Books

- Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David
  Boyle, “From Machine-to-Machine to the Internet of Things: Introduction to a New Age of
- Intelligence”, 1st Edition, Academic Press, 2014. (ISBN-13: 978-0124076846)
- The Internet of Things: How Smart TVs, Smart Cars, Smart Homes, and Smart Cities Are Changing the World, Michael Miller

COURSE OUTCOMES (CO):

- Students will be able to describe and differentiate between key IoT enabling technologies, such as Wireless Sensor Networks, Cloud Computing, and M2M communication, while understanding the role of software-defined networks and embedded systems.
- Students will gain knowledge of the challenges in IoT design and development, including security and technical constraints, and propose solutions to overcome these issues.
- Students will acquire skills in designing and analyzing IoT architecture, considering various views like functional, information, and deployment, and apply these concepts to industrial automation and real-world applications.
- Students will develop practical skills in using Arduino for IoT projects, including setting up the environment, connecting to sensors and actuators, and implementing cloud-based data monitoring and M2M interactions.

COURSE TITLE: Natural Language Processing

Course Code:				**CSE205353**		**Examination Scheme**	
Total number of Lecture Hours: 46						**External**	**80**
						Internal	**20**
Lecture (L):	**46**	**Practicals(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**

Course Objectives:

- To provide a comprehensive understanding of Natural Language Processing (NLP), its applications, and the fundamental levels of language analysis, including grammar and sentence structure.
- To explore various parsing techniques and grammar models used in NLP, including deterministic parsers, probabilistic grammars, and part-of-speech tagging.
- To introduce semantic analysis concepts such as word sense disambiguation, speech acts, semantic interpretation, and the use of feature systems for lexical and grammatical representation.
- To examine advanced semantic filtering techniques, statistical methods for word sense disambiguation, and the integration of multiple approaches for semantic analysis and interpretation.

Course Content	**TEACHING HOURS**
UNIT 1:	**12 Hrs**
Introduction to Natural Language Processing, Applications of NLP, Different levels of Language Analysis, Representation and Understanding, Linguistic Background, Grammar and sentence structure, Top down parser, Bottom up chart parser, Transition Network Grammars, Finite state Models and Morphological Processing. Feature Systems and Augmented Grammars, Morphological Analysis and Lexicon.	
UNIT 2:	**12 Hrs**
Grammars for Natural Language, Encoding uncertainty : Shift Reduce Parsers, A deterministic parser, Partial Parsing, Ambiguity resolution , Part of speech tagging, Probabilistic Context free grammars, Best first parsing	
UNIT 3:	**12 Hrs**
Semantics and logical form, word sense and ambiguity, Speech acts and embedded sentences, defining semantic structure Semantic Interpretation an compositionality, A simple grammar and lexicon with semantic interpretation, Lexicalized semantic interpretation and semantic roles, Semantic interpretation using feature unification.	
UNIT 4:	**12Hrs**
Selectional restrictions, Semantic filtering, semantic networks, statistical word sense disambiguation, statistical semantic preferences, Combining approaches to disambiguation. Grammatical relations, Semantic grammars, template matching, semantically driven parsing techniques, scooping phenomenon, co-reference and binding constraints.	

Textbooks

- Allen, James, Natural Language Understanding, Second Edition, Benjamin/Cumming. Charniack, Eugene, Statistical Language Learning, MIT Press,.
- Jurafsky, Dan and Martin, James, Speech and Language Processing, Second Edition, Prentice Hall, 2008.
-

Reference Books

- Manning, Christopher and Heinrich, Schutze, Foundations of Statistical Natural Language Processing, MIT Press.

COURSE OUTCOMES (CO):

- Students will be able to explain the key concepts of NLP, including language representation, grammar models, and different parsing strategies like top-down and bottom-up parsers.
- Students will gain skills in implementing various parsing techniques and part-of-speech tagging, along with understanding ambiguity resolution in natural language processing.
- Students will acquire knowledge of semantic interpretation, including word sense disambiguation, compositionality, and the role of semantic roles in lexicalized interpretation.
- Students will demonstrate proficiency in applying semantic filtering techniques, statistical approaches to word sense disambiguation, and the combination of various methods for effective natural language understanding.