GIRIJANANDACHOWDHURYUNIVERSITY
Hatkhowapara, Azara, Guwahati - 781017, Assam

Department of Computer Science and Engineering

Course Structure of B.Tech CSE 5th Semester

Sl. No	Course Code	Course Name	Hours per week			Credit
			L	T	P	C
1.	BCS23213T	Introduction to Database System	3	0	0	3
2.	BCS23213P	Introduction to Database System Lab	0	0	4	2
3.	BCS23214T	AI and Machine Learning	2	1	0	3
4.	BCS23214P	AI and Machine Learning Lab	0	0	4	2
5.	BCS23215T	Design and Analysis of Algorithms	3	0	0	3
6.	BCS23215P	Design and Analysis of Algorithms Lab	0	0	2	1
7.	BCS23216T	Web Technology	2	0	0	2
8.	BCS23216P	Web Technology Lab	0	0	2	1
9.	BCS23217T	Signals and Systems	3	0	0	3
10.	BLW25101T	Constitution of India	2	-	-	0
11.	BCS23218P	Internship (Academia)	-	-	4	2
TOTAL CREDIT			**15**	**1**	**16**	**22**

* 4 weeks Mandatory Academic internship needed to be done in the 4th semester summer break and the report is to be submitted and evaluated in 5th semester.

BCS23213T	Introduction to Database System	L	T	P	C
		3	0	0	3

Prerequisite:

1. Knowledge of Programming Languages

2. Basic mathematical backgrounds (elementary set theory, concepts of relations and functions etc
3. Data Structure and algorithms (Trees, B-tree, B+ tree)

Course Objectives:

1. To understand the relational database design principles, database storage structures, basic issues of transaction processing and concurrency control.
2. To provide in depth knowledge for designing and implementing database queries using Structured Query Language.

Course Outcome:

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

1. Identify the basic need of database management system, different types of databases and query languages (Bloom's L3 Applying)
2. Design entity relationship diagrams, relational DBMS.(Bloom's L6 Creating)
3. Analyze the concepts of functional dependencies, normalization and storage strategies. (Bloom's L4 Analyzing)
4. Apply the principles of database transaction management, database recovery etc. (Bloom's L3 Applying)

Module 1: Introduction	**3 hours**

Data Abstraction, Data Independence, Introduction to Data Models. Entity-relationship model, network model, relational, semi structured model etc

Module 2: Relational Databases	**4 hours**

Relational Data Model, Relational Algebra, Relational Calculus

Module 3: Interacting with database (SQL)	**9 hours**

Data Definition Language (DDL), Data Manipulation Language (DML). Insert/Delete/Update

Simple Queries (select/project/join/ aggregate queries)

Complex queries (With Clause, Nested Subqueries, Views)

Module 4: Big Data	**3 hours**

Key-value Stores and Semi- structured Data, using JSON and MongoDB, or other combinations

Module 5: Database Design	**8 hours**

Introduction to ER model, Mapping from ER to relational model. Functional Dependencies, Normalization.	
Module 6: Physical Design and Storage Strategies	**6 hours**
Overview of Physical Storage (Hard Disks, Flash/SSD/RAM), sequential vs random I/O, Reliability via RAID Storage Organization (Records, Pages and Files), Indexing, B+-Trees	
Module 7: Query Processing and Optimization	**6 hours**
Query Processing: External sort, Joins using nested loops, indexed nested loops Overview of Query Optimization: equivalent expressions, and concept of cost- based optimization	
Module 8: Transaction Processing	**6 hours**
Concept of transactions and schedules, ACID properties. Conflict-serializability. Concurrency control: locks, 2PL, Strict 2PL. Recovery using undo and redo logs.	
Total Hours	**45 hours**

Text Book

1.	Database System Concepts, 7th Edition, Silberschatz, Korth and Sudarshan, McGraw-Hill. Indian Edition released 2021
2.	Fundamentals of Database Systems, 7th Edition, Elmasri and Navathe, Pearson Pubs, 2017
3.	Principles of Database Management, Lemahieu, Broucke and Baesens, Cambridge University Press, 2018
4.	Database Management Systems, RP Mahapatra, Khanna Publishing House, 2020.
5.	Database Management Systems, Krishnan, McGraw Hill.
6.	"Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.
7.	"Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley

Suggested reference books / Online resources:

1.	Software a. Relax Relational algebra calculator: https://dbis-uibk.github.io/relax/landing b. SQL: PostgreSQL/MySQL/MariaDB, or SQLite in browser c. B+-tree visualization:

	https://www.cs.usfca.edu/~galles/visualization/BPlusTree.html d. MongoDB e. Various DB systems playground: https://www.pdbmbook.com/playground

BCS23213P	INTRODUCTION TO DATABASE SYSTEM LAB	L	T	P	C
		0	0	4	2

Prerequisite: Basic programming skills

Course Objectives:

1. To provide students with hands-on experience to design and create databases, write SQL queries, and perform data manipulation tasks, enabling them to translate theoretical knowledge into practical skills.

2. To gain proficiency in setting up and configuring database management systems, managing user access, and implementing security measures

3. To enhance students' abilities to optimize database queries and improve system performance

Course Outcome:

After Successful completion of the course, students will be able to

1 : Make use of Data Definition Language(DDL) statement using SQL for Creating, Deleting and Modifying Relations Schemas.

2 :Make use of Data Manipulation Language(DML) statement using to SQL Retrieve, Store,Modify,Delete,Insert and Update data in Database.

3 : Apply the concept of trigger using SQL.

4 : Apply the concept of procedure using SQL.

Experiments

1. Write statements for creation/alteration/view of relational database schema along with necessary integrity constraints. Insert tuples into the created tables.
2. Write SQL statements for selection, updation and deletion of data
3. Write SQL statements for joins, grouping, nested sub query.
4. Write statements for creating and using stored procedures.
5. Write statements for creating and using triggers.

Total Lab hours	**30 hours**

Reference Books

1.	"Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.
2.	"Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.
3.	"Fundamentals of Database Systems", 5th Edition by R. Elmasri and S. Navathe, Pearson Education
4.	"Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley

	Introduction to AI\ML	L	T	P	C
		2	1	0	3

Prerequisite:

1. A course on Python (Anaconda)

Course Objectives:

1. COB1: To introduce basic concepts and techniques of Artificial Intelligence and Machine Learning.
2. COB2: To develop understanding of problem-solving, knowledge representation, and learning algorithms.
3. COB3: To implement and evaluate machine learning models using Python and relevant libraries.

Course Outcome:

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

1. CO1: Understand fundamental concepts of Artificial Intelligence and its applications.
2. CO2: Apply search and reasoning techniques to solve real-world problems.
3. CO3: Demonstrate knowledge of machine learning algorithms and evaluate their performance.
4. CO4: Implement ML algorithms using Python and interpret the results.
5. CO5: Design a simple AI/ML project that addresses a real-world use case.

Module 1: Introduction of AI & ML	**8 hours**

Definition of AI & ML, Applications of AI & ML, Types of Learning (Supervised, Unsupervised, Reinforcement), Hypothesis Space and Inductive Bias, Evaluation Metrics: Accuracy, Loss Functions, Cross-Validation Techniques, Overfitting and Underfitting

Module 2: Knowledge Representation: Introduction	**6 hours**

Knowledge Representation: Introduction, Propositional Logic, Predicate Logic, Rule-Based Systems, Inference Engine and Reasoning Mechanisms.

Module 3: AI vs ML vs Data Science	**8 hours**

Data Preprocessing: Cleaning, Encoding, Feature Handling Missing Data, Model Evaluation Metrics: Precision, Recall, F1, ROC, Introduction to pandas and scikit-learn.

Module 4: Linear Regression – Concept and Implementation	**8 hours**

Logistic Regression, Naïve Bayes, Decision Trees – ID3, Entropy, Gini, Decision Trees Day, K-Nearest Neighbors (KNN), Support Vector Machines (SVM) , K-Means Clustering , Hierarchical Clustering Confusion Matrix, Precision, Recall (Summary & Practice)

Module 5: Introduction to Neural Networks	**8 hours**

Neural Networks, Perceptron Algorithm, Activation Functions, Multilayer Networks Forward Propagation, Gradient Descent, Backpropagation Algorithm

Module 6: Model Deployment and Training	**7 hours**

Overview: Chatbots, Fraud Detection, Use Case: Sentiment Analysis, Model Training, Model Deployment.	
Total Hours	**45 hours**
Text Book	
1.	Russell, S., & Norvig, P. (2020). Artificial intelligence: A modern approach (4th ed.). Pearson.
2.	Muller, C.A & Guido, S. (2017). Introduction to Machine Learning with Python. (3rd ed.). Sebastopol, CA: O'Reilly.
3.	Raschka, S. & Mirjalili, V. (2017). Python Machine Learning (2 nd ed.). : Packt Publishing
4.	Aurelien, G. (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, (3 rd ed.). : O'Reilly.
5.	Mitchell, Tom (1997), Machine Learning. (1 st Ed.), McGraw- Hill.

	Introduction to AI\ML (Practical)	L	T	P	C
		0	0	4	4

Prerequisite:

1. A course on Python (Anaconda)

Course Objectives:

1. COB1: To introduce basic concepts and techniques of Artificial Intelligence and Machine Learning.
2. COB2: To develop understanding of problem-solving, knowledge representation, and learning algorithms.
3. COB3: To implement and evaluate machine learning models using Python and relevant libraries.

Course Outcome:

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

1. CO1: Understand fundamental concepts of Artificial Intelligence and its applications.
2. CO2: Apply search and reasoning techniques to solve real-world problems.
3. CO3: Demonstrate knowledge of machine learning algorithms and evaluate their performan ce.
4. CO4: Implement ML algorithms using Python and interpret the results.
5. CO5: Design a simple AI/ML project that addresses a real-world use case.

Module 1: Explore basic AI/ML	**8 hours**
Explore basic AI/ML using Pythons, Identify real-world AI/ML applications, Implement and visualize different types of learning in scikitlearn, Demonstrate overfitting/underfitting with polynomial degrees.	
Module 2: Implement propositional logic	**6 hours**
Propositional logic in Python using Boolean operators, Basic rule-based systems using if-else conditions	
Module 3: AI vs ML vs Data Science	**8 hours**
Compare AI, ML, Data Science model, Perform data cleaning and encoding using pandas, Apply normalization and missing value imputation , Evaluate models using precision, recall, F1-score	
Module 4: Concept of Implementation	**8 hours**
Linear Regression on a real-world dataset , Build Logistic Regression model, Naïve Bayes classifier, Decision Tree algorithm, KNN, SVM and compare model performance	
Module 5: Analysis of datasets	**6 hours**
Preprocess dataset, Train sentiment analysis model, Deploy model using Flask or Streamlit	
Total Hours	**36 hours**

Text Book

1. Russell, S., & Norvig, P. (2020). Artificial intelligence: A modern approach (4th ed.). Pearson.

2.	Muller, C.A & Guido, S. (2017). Introduction to Machine Learning with Python. (3rd ed.). Sebastopol, CA: O'Reilly.
3.	Raschka, S. & Mirjalili, V. (2017). Python Machine Learning (2 nd ed.). : Packt Publishing
4.	Aurelien, G. (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, (3 rd ed.). : O'Reilly.
5.	Mitchell, Tom (1997), Machine Learning. (1 st Ed.), McGraw- Hill.

GIRIJANANDACHOWDHURYUNIVERSITY
Hathkhowapara, Azara, Guwahati - 781017,Assam

BCS23215T	Design and Analysis of Algorithms	L	T	P	C
		3	0	0	3

Pre-requisite: A solid foundation in programming fundamentals and data structures and algorithms.

Course Objectives:

1. To understand the fundamental characteristics and performance analysis of algorithms, including time and space complexity using asymptotic notations and recurrence relations.
2. To study and apply various algorithmic design strategies such as Divide and Conquer, Greedy, Dynamic Programming, Backtracking, and Branch & Bound to solve real-world problems efficiently.
3. To explore classical graph and tree algorithms and apply them in problem-solving using techniques like BFS, DFS, shortest path algorithms, minimum spanning tree, and network flow.
4. To analyze the theoretical foundations of computational complexity, differentiate between tractable and intractable problems, and understand NP-completeness and related concepts.
5. To examine advanced algorithmic approaches including approximation, randomized, and evolutionary algorithms, and develop insights into problems beyond NP such as PSPACE.

Course Outcomes: After successful completion of the course, the students will learn:

CO1: Understand the fundamental concepts and mathematical analysis of algorithms, including time and space complexity, and recurrence relations. → *Bloom's Level*: Understand, Analyze

CO2: Apply various algorithm design techniques such as Divide and Conquer, Greedy, Dynamic Programming, Backtracking, and Branch & Bound to solve computational problems. → *Bloom's Level*: Apply, Analyze

CO3: Develop efficient algorithms for graph and tree-based problems, and evaluate their performance using appropriate algorithmic strategies. → *Bloom's Level*: Apply, Evaluate

CO4: Analyze computational complexity classes (P, NP, NP-Complete, NP-Hard), and explore advanced approaches like approximation, randomized, and evolutionary algorithms for solving hard problems. → *Bloom's Level*: Analyze, Create

Module 1: Introduction Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds –best, average and worst case behavior; Time and space complexity, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.	12 Hours
Module 2: Fundamental Algorithmic Strategies Divide and Conquer, Brute Force, Greedy, Dynamic Programming, Branch and Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem Solving, Bin Packing, Knap Sack, TSP, Heuristics characteristics and their application domains.	12 Hours

Module 3: Graph and Tree Algorithms Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.	11 Hours
Module 4: Tractable and Intractable Problems Computability of Algorithms, Computability classes –P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP complete problems and Reduction techniques	5 Hours
Module 5: Advanced Topics Approximation algorithms, Randomized algorithms, Evolutionary algorithms, Class of problems**,** beyond NP –P SPACE	5 Hours
Total Lecture hours	45 Hours

Text Books:

1. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.

Reference Books:

1. Ellis Horowitz, Satraj Sahni, Rajasekharam, Fundamentals of Computer Algorithms, University Press, New Delhi.
2. R. C. T. Lee, S. S. Tseng, R.C. Chang and T. Tsai, Introduction to Design and Analysis of Algorithms A strategic approach, McGraw Hill, India.

Internal Member/BoS

External Member/BoS

GIRIJANANDACHOWDHURYUNIVERSITY
Hathkhowapara, Azara, Guwahati - 781017,Assam

BCS23215P	Design and Analysis of Algorithms Lab	L	T	P	C
		0	0	2	1

Pre-requisite: Prerequisite: Good understanding of data structures and basic programming in C/C++/Java/Python.

Course Objectives:

To implement and analyze various algorithms using different design strategies such as Divide and Conquer, Greedy, Dynamic Programming, Backtracking, Branch and Bound, and Graph algorithms. The students will also gain hands-on experience with problems related to NP-Completeness and advanced algorithmic approaches.

Course Outcomes: After successful completion of the course, the students will learn:

CO1: Implement and analyze algorithms for solving problems using recursion and evaluate their time complexity using asymptotic notations. (Apply, Analyze)

CO2: Design and apply algorithmic strategies such as Divide & Conquer, Greedy, Dynamic Programming, Backtracking, and Branch & Bound to solve real-world problems. (Apply, Create)

CO3: Develop and simulate graph algorithms like shortest path, MST, and traversal methods using appropriate data structures and evaluate their performance. (Apply, Evaluate)

CO4: Demonstrate the behavior of NP-complete problems and explore the use of approximation, randomized, or evolutionary algorithms for problem-solving. (Understand, Analyze, Create)

Module 1: Introduction to Algorithm Analysis — 8 Hours

Experiment 1: Write programs to compute time complexity of different operations (e.g., sorting, searching).

Experiment 2: Implement recurrence relations using:

- Substitution method
- Recursion Tree method
- Master's Theorem

Module 2: Algorithm Design Strategies — 8 Hours

Experiment 3: Implement Merge Sort and Quick Sort using Divide and Conquer.

Experiment 4: Solve the Knapsack Problem.

Experiment 5: Solve the Travelling Salesman Problem.

Experiment 6: Implement a solution for the N-Queens problem using

Backtracking. **Experiment 7**: Implement Branch and Bound for: o 0/1 Knapsack o TSP	
Module 3: Graph Algorithms **Experiment 8**: Implement DFS and BFS for a graph. **Experiment 9**: Implement Dijkstra's and Bellman-Ford shortest path algorithms. **Experiment 10**: Implement Kruskal's and Prim's algorithms for Minimum Spanning Tree. **Experiment 11**: Implement Topological Sorting. **Experiment 12**: Implement Network Flow using Ford-Fulkerson algorithm.	8 Hours
Module 4: NP-Completeness and Advanced Topics **Experiment 13**: Demonstrate NP-complete behavior by solving Subset Sum problem using: • Backtracking • Dynamic Programming (Analyze time complexity for both) **Experiment 14**: Implement a basic Approximation Algorithm for Vertex Cover or TSP. **Experiment 15**: Implement a simple Randomized algorithm (e.g., Randomized Quick Sort). **Experiment 16**: (Optional) Implement a Genetic Algorithm for optimization problem.	8 Hours
Total Lecture hours	32 Hours

Syllabus	Web Technology BCS23216T	L	T	P	C
		3	0	0	3

Pre-requisite: None

Course Objectives:

- Design and develop static and responsive web pages using HTML, CSS, and JavaScript
- Build backend services using Node.js and Express
- Create modern frontend applications using React.js
- Integrate backend, frontend, and database into a working full-stack application
- Develop mini-projects such as Student and Health Information Systems

Course Outcome:

After successful completion of the course, the students will be able to

CO1 Design and develop static and responsive web pages using HTML, CSS, and JavaScript for enhanced user experience and accessibility.
CO2 Construct and manage backend services using Node.js and Express to handle server-side logic and RESTful API operations.
CO3 Develop modern, component-based frontend applications using React.js with effective state management and routing.
CO4 Integrate frontend, backend, and database layers to build and deploy full-stack web applications with seamless data communication.

Module1:Introduction — **10 hours**

Introduction to HTML: Elements, Lists, Tables, Images, Forms, Frames
Cascading Style Sheets (CSS)
Introduction to DHTML
XML Basics: DTD (Document Type Definition), XML Schemas
DOM (Document Object Model)

Module2:JavaScript — **12 hours**

JavaScript Basics: Variables, Data types, Control Structures
Functions and Arrays
Objects and Events
Dynamic HTML using JavaScript

Module3:Web Servers — **10 hours**

Web Servers: Apache (XAMPP/LAMPP), Node.js Environment
Node.js Basics:
Modules, File System, HTTP, Express.js
Handling Forms, Routing, Middleware
Session Management using Express
Cookies and Authentication
Introduction to REST API with Node.js

Module4: Frontend & Database Integration — **7 hours**

React.js: Components, Props, State, Events Forms and Hooks Axios for API Calls Database Connectivity: MySQL Integration with Node.js CRUD operations Case Study Implementations (mandatory): Student Information System Health Management System

Books

- Paul Deitel, Harvey Deitel, Abbey Deitel – Internet & World Wide Web: How to Program, 5th Edition, Pearson, 2012.
- Ethan Brown – Web Development with Node and Express, O'Reilly, Latest Edition
- Robin Wieruch – The Road to React, Leanpub

Syllabus	Web Technology LAB BCS23216P	L	T	P	C
		0	0	2	1

Pre-requisite: Basic knowledge of Programming

CourseObjectives:

- Design and develop static and responsive web pages using HTML, CSS, and JavaScript
- Build backend services using Node.js and Express
- Create modern frontend applications using React.js
- Integrate backend, frontend, and database into a working full-stack application
- Develop mini-projects such as Student and Health Information Systems

CourseOutcome:

CO1 Design and develop static and responsive web pages using HTML, CSS, and JavaScript for enhanced user experience and accessibility.
CO2 Construct and manage backend services using Node.js and Express to handle server-side logic and RESTful API operations.
CO3 Develop modern, component-based frontend applications using React.js with effective state management and routing.
CO4 Integrate frontend, backend, and database layers to build and deploy full-stack web applications with seamless data communication.
CO5 Design and implement mini-projects (e.g., Student Information System, Health Information System) demonstrating full-stack development practices.

Suggested Practical Lists:

1. Create a basic static web page using HTML with proper structure and semantic elements.
2 Design a responsive web layout using CSS Grid and Flexbox.
3 Implement a form with validation using HTML5 and JavaScript (e.g., registration form).
4 Develop an image gallery or portfolio using CSS transitions and animations.
5 Create a JavaScript program to manipulate the DOM dynamically (add, edit, remove elements).
6 Build a To-Do List application using HTML, CSS, and JavaScript with local storage.
7 Develop a RESTful API using Node.js and Express to perform CRUD operations on student data.
8 Connect the backend API with a MySQL database for data persistence.
9 Create a React.js frontend to consume RESTful APIs and display data dynamically.
10 Implement user authentication (login/register) using Express sessions or JWT.
11 Integrate React frontend with Node.js backend and MySQL database to form a full-stack mini-project.
12 Develop and present a mini project (e.g., Student Management / Health Record System) demonstrating full-stack integration.

BCS23217T	SIGNALS AND SYSTEMS	L	T	P	C
		3	0	0	3

Pre-requisite: Basics of Mathematics

Course Objectives:

1. To introduce the fundamental concepts of signals and systems and their classifications in both continuous and discrete time domains.
2. To develop analytical skills required to analyze Linear Time-Invariant (LTI) systems using convolution, differential and difference equations.
3. To provide a comprehensive understanding of signal transformation techniques including Fourier, Laplace, and Z-transforms, along with the implications of signal sampling.

Course Outcome:

After successful completion of the course, the students will learn

1. CO1 : Explain the characteristics of different types of signals and systems including linearity, causality, and stability.
2. CO2 : Apply convolution and system equations to determine the output of continuous and discrete time LTI systems.
3. CO3 : Analyze signals using Fourier, Laplace, and Z-transform techniques for both time and frequency domain representations.
4. CO4 : Evaluate the effects of sampling, aliasing, and reconstruction in signal processing systems.

MODULE 1 — **15hours**

Signals and systems as seen in everyday life and in various branches of engineering and science, Signal properties: periodicity, determinism and stochastic character. Standard signals- Step, Ramp, Pulse, Impulse, Real and complex exponentials and Sinusoids_ Classification of signals — Continuous time (CT) and Discrete Time (DT) signals, Periodic & Aperiodic signals, Deterministic & Random signals, Energy & Power signals — Classification of systems- CT systems and DT systems- — Linear & Nonlinear, Time-variant & Time-invariant, Causal & Non-causal, Stable & Unstable.

Module 2 — **10hours**

Impulse response and step response, convolution, cascade interconnections, Characterization of causality and stability of LTI systems, System representation through differential equations and difference equations. Periodic inputs to an LTI system, the notion of a frequency response and its relation to the impulse response.

Module 3 — **15hours**

Fourier series for periodic signals, Calculation of Fourier Coefficients, Fourier Transform — properties- Laplace Transforms and properties Impulse response — Difference equations-Convolution sum- The Discrete-Time Fourier Transform (DTFT) and the Discrete Fourier Transform (DFT) and Z Transform Analysis of Recursive & Non-Recursive systems-DT systems connected in series and parallel.

Module 4	**5 hours**
The Sampling Theorem and its implications, Spectra of sampled signals, Aliasing and its effects, Relation between continuous and discrete time systems.	
Total Lecture hours	**45 hours**

Text Book

1. A. V. Oppenheim, A. S. Willsky and S. H. Nawab, “Signals and systems”, Prentice Hall India,1997.
2. A Nagoor Kani , “SIGNALS AND SYSTEMS”,Tata McGraw-Hill
3. B.P. Lathi "Principles of signal processing and linear systems" , Publication- New Delhi Oxford University Press c2009

Reference Books

1. J. G. Proakis and D. G. Manolakis, “Digital Signal Processing: Principles, Algorithms, and Applications”, Pearson, 2006.

2. H. P. Hsu, “Signals and systems”, Schaum’s series, McGraw Hill Education, 2010.

3.S. Haykin and B. V. Veen, “Signals and Systems”, John Wiley and Sons, 2007.

4.A. V. Oppenheim and R. W. Schafer, “Discrete-Time Signal Processing”, Prentice Hall, 2009.

5.M. J. Robert “Fundamentals of Signals and Systems”, McGraw Hill Education, 2007.7.B. P. Lathi, “Linear Systems and Signals”, Oxford University Press, 2009.

6. Ramesh Babu & Anandh Natarajan, "Signals and Systems", Publisher: Scitech Publications.

7. Simon Haykin and Barry Van Veen, "Signals and Systems", Publisher: Wiley India

Internal Member/BoS External Member/BoS

Course Code	Course Title	Hours per week L-T-P	Credit C
	Constitution of India	2-0-0	0 (PP/NP)

Course Objectives: Students will be able to:

1. Understand the importance of Constitution in a country also understand the nature of the Constitution of India.
2. Understand the provisions of equality, liberty and freedom under the Constitution of India.
3. To address the importance of DPSP and its relation with Fundamental Rights.
4. To address the provisions of emergency and amendment.

Course Outcomes: Students will be able to:

1. Discuss the characteristics and nature of the Constitution of India .
2. Discuss about the relation among FR, DPSP and Fundamental Duties.
3. Explain the power, function of the different organs of the Constitution.
4. Discuss various provisions of Emergency and its effect.

Module 1. Introduction :

a) Definition and Classification of Constitution
b) Salient features of the Constitution of India
c) Nature of Constitution of India- Whether federal or unitary
d) Preamble of the Constitution of India
e) Article 1
f) Citizenship – Acquirement and Termination (Article 5---11)

Module 2: Fundamental Rights:

a) What is Fundamental Rights?
b) Definition of Law, State (Article 12, 13)

c) Part III of the Constitution of India (Article 14, 15, 16, 19, 20, 21, 21-A, 23, 24, 29 30, 32)

Module 3: Directive Principles of State Policy and Fundamental Duties

a) Nature and Significance of DPSP

b) Differences between FR and DPSP

c) Part IV – (Articles 38, 29, 39A, 41, 42, 44, 45, 47, 48A)

d) Fundamental Duties

Module 4: **Organs of Government under the Constitution of India**

a) Parliament- Composition

b) Executive- Centre and State

c) Judiciary- Supreme Court and High Court

Module 5: Emergency provisions and Amendment of the Constitution of India

a) Different types of emergency

b) Effect of emergency

c) Amendment- meaning, importance, procedures and limitation

Text books/ references

1. **The Constitution of India, 1950 (Bare Act)**
2. **Constitution of India, J. N. Pandey**
3. **Constitutional Law of India, Narender Kumar**
4. **Constitution of India, S. N. Myneni**