

<b>Course Code</b>	:	<b>BISA403C</b>	<b>Web Programming</b>	<b>Semester</b>	:	<b>4</b>	
<b>L:T:P</b>	:	<b>2 : 0: 2</b>		<b>Course Type</b>	:	<b>Theory Integrated</b>	
<b>Hours/Sem.</b>	<b>Teaching</b>	:		<b>56 (28L+28P)Hrs</b>	<b>CIE Marks</b>	:	<b>50</b>
	<b>Learning (TW+SL)</b>	:		<b>28Hrs</b>	<b>SEE Marks</b>	:	<b>50</b>
	<b>Exam</b>	:		<b>6(3+3)Hrs</b>	<b>Total Marks</b>	:	<b>100</b>
	<b>Total Hrs.</b>	:		<b>90Hrs</b>	<b>Credits</b>	:	<b>03</b>

### Professional Competency:

Design and implement functional, user-friendly, interactive and responsive web interfaces/pages.

### Course Outcomes:

After completion of the course, student will be able to:

CO1: Apply web fundamentals, HTTP concepts, XHTML, and CSS standards to design and develop well-structured, standards-compliant, and visually formatted web pages.

CO2: Develop client-side programs using JavaScript to handle user interactions, validate inputs, and manipulate XHTML elements through DOM and event-driven programming.

CO3: Design and implement dynamic and interactive web documents using JavaScript by analyzing user actions and modifying content, style, and element positioning in real time.

CO4: Build data-driven web applications using PHP and MySQL by integrating server-side scripting, form handling, cookies, and database connectivity.

### UNIT-I

**07 Hrs.**

**FUNDAMENTALS OF WEB, XHTML** - Internet, HTTP request and HTTP response phase, MIME, The Web Programmers Toolbox.

**XHTML:** Basic syntax; Standard XHTML document structure; Basic text markup. XHTML: Hypertext Links; Lists; Tables; Forms; Syntactic differences between HTML and XHTML. CSS: Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; CSS: Font properties; List properties; Color; Alignment of text; Background images; The <span> and <div> tags

### UNIT-II

**07 Hrs.**

**Basics of JavaScript:** General syntactic characteristics; Primitives, Screen output and keyboard input; Control statements; Arrays; Functions.

**JavaScript & XHTML Documents:** The Document Object Model, Element Access in JavaScript, Events & Event Handling, Basic Concepts of Event handling, Events, Attributes & Tags, Handling Events from Body Elements, Handling Events from Button Elements, Handling Events from Textbox & password Elements, The Focus Event

### UNIT-III

**07 Hrs.**

**Dynamic Documents with JavaScript:** Introduction, Positioning Elements, Absolute Positioning, Relative Positioning, Static Positioning, Moving Elements, Element Visibility, Changing Colors & Fonts, Changing Colors, Changing Fonts, Dynamic Contents, Stacking Elements, Locating the Mouse Cursor, Reacting to the Mouse Click, Slow Movement of Elements, Dragging & Dropping Elements.

### UNIT-IV

**07 Hrs.**

**Introduction to PHP:** Origins and Uses of PHP, Overview of PHP, General Syntactic Characteristics, Primitives, Operations and Expressions, Output, Control statements, Arrays, Functions, Form Handling, Cookies, Database access with PHP and MySQL

## Reference Books \*

1. Programming the World Wide Web - Robert W. Sebesta, 4th Edition, Pearson Education, 2008.
2. Internet & World Wide Web How to program - M. Deitel, P.J.Deitel, A. B. Goldberg, 3rd Edition, Pearson Education / PHI, 2004.
3. Web Programming Building Internet Applications - Chris Bates, 3rd Edition, Wiley India, 2006.
4. The Web Warrior Guide to Web Programming - Xue Bai et al, Thomson, 2003.
5. M. Srinivasan: Web Technology Theory and Practice, Pearson Education, 2012.
6. Jeffrey.C. Jackson: Web Technologies-A Computer Science Perspective, Pearson Education, Eleventh Impression, 2012

## Lab Assignments:

1. Develop and demonstrate a XHTML document that illustrate the use of ordered list, unordered nested list, table, borders, padding, color and the <span> tag.
2. Design an XHTML web page using CSS, which has two paragraphs as follows:
  - i) First para – Arial font, 24 pt size, italic, bold, text color blue, background color yellow, underlined, aligned right
  - ii) Second para – Courier font, 40pt size, small capital letters, overlined, background color white, text color red, aligned center.
3. Develop JavaScript scripts for the following:
  - i. To model a simple calculator using ‘switch’ statement
  - ii. To print the number of prime numbers in a given range L to R using functions
  - iii. To find whether the given number is an Armstrong number using functions
  - iv. To find the number of occurrences of a character in a string using functions.
4. Develop XHTML document with Java Script to handle events as follows:
  - i) ‘Blur’ event to transform the input text to upper case.
  - ii) ‘focus’ event to change the background color of a text box.
  - iii) ‘Change’ event to display the preferred browser in an alert box when the user relocates the browser from a drop-down menu.
  - iv) ‘click’ event to copy the contents of one text into another.
5. Create and demonstrate an XHTML document using Java Script for event handling as follows: XHTML document should contain a set of radio buttons showing names of 3<sup>rd</sup> semester subjects. On clicking a particular button, event handler should be called to display a brief description about the related subject using an alert box.
6. Develop and demonstrate an XHTML document as follows:

The XHTML document must contain four paragraphs stacked on the top of each other with only enough of each showing so that the mouse cursor when placed over the exposed part of any paragraph, it should rise to the top to become completely visible.
7. Write XHTML code to create a table as follows and enter the quantity required.

Product Name	Price/item (Rs)	Quantity
		<input type="text"/>
		<input type="text"/>
		<input type="text"/>

Create a set of radio buttons to accept the payment method needed-cheque, cash or card. Develop a PHP script to display the results in a table, which should contain product name, price, quantity and total cost for each product. Below the table, display the total number of ordered items, the total cost and the payment method used.

8. Create an XHTML document to accept student data which contains student name, branch and college name. Write a PHP document to insert data into the MySQL database and retrieve the particular database on student name from the database and display.

**Table: Matrix to describe the mapping of COs with POs (considering Wks) and PSOs**

Course Outcomes (COs)	Program Outcomes and (Wks)											Program Specific Outcomes (PSOs)	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK1, WK2, WK3, WK4)	3 (WK5)	-	2 (WK6)	-	-	- (WK9)	- (WK9)	2 (WK8)	1	1	2

<b>Course Code</b> :		<b>BISA404C</b>	<b>Design and Analysis of Algorithm Using Java (Integrated)</b>	<b>Semester</b> :	<b>4</b>
<b>L:T:P</b> :		<b>3L-0T-2P</b>		<b>Course Type</b> :	<b>Integrated</b>
<b>Hours/Sem.</b>	<b>Teaching</b> :	<b>42 Hrs</b>		<b>CIE Marks</b> :	<b>50</b>
	<b>Learning (TW+SL)</b> :	<b>42+28 Hrs</b>		<b>SEE Marks</b> :	<b>50</b>
	<b>Exam</b> :	<b>08 Hrs</b>		<b>Total Marks</b> :	<b>100</b>
	<b>Total Hrs.</b> :	<b>120 Hrs</b>	<b>Credits</b> :	<b>04</b>	

<b>Professional Competency:</b>	
The overarching ability to model a real-world problem abstractly and develop a clear, step-by-step computational solution (an algorithm) to solve it.	
<b>Course Outcomes:</b>	
<b>After completion of the course student will be able to</b>	
<ol style="list-style-type: none"> <li>1. Analyze the asymptotic performance of algorithm.</li> <li>2. Apply various sorting, searching and shortest path algorithms to solve engineering problems.</li> <li>3. Implement solutions by designing algorithms for various computing problems.</li> <li>4. Analyze the limits of algorithm and coping with a limitations of algorithm.</li> </ol>	
<b>Module-I</b>	<b>10 Hrs.</b>
<b>Introduction:</b> What is an Algorithm?, Fundamentals of Algorithmic Problem Solving. <b>Fundamentals of the Analysis of Algorithm Efficiency:</b> Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Nonrecursive Algorithms, Mathematical Analysis of Recursive Algorithms.	
<b>Module-II</b>	<b>11 Hrs.</b>
<b>Brute Force:</b> Selection Sort and Bubble Sort, Sequential Search and Brute-Force String Matching, Exhaustive Search. <b>Divide-and-Conquer:</b> Mergesort, Quicksort, Binary Search, Multiplication of Large Integers and Strassen's Matrix Multiplication. <b>Decrease-and-Conquer:</b> Insertion Sort, Depth-First Search and Breadth-First Search, Topological Sorting.	
<b>Module-III</b>	<b>11 Hrs.</b>
<b>Transform-and-Conquer:</b> Balanced Search Trees, Heaps and Heapsort, Problem Reduction. <b>Space and Time Tradeoffs:</b> Sorting by Counting: Comparison Counting Sort, Input Enhancement in String Matching: Horspool's algorithm. <b>Dynamic Programming:</b> Three Basic Examples, Floyd's Algorithms, The Knapsack Problem and Memory Functions.	
<b>Module-IV</b>	<b>10Hrs.</b>
<b>Greedy Technique:</b> Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees. <b>Limitation of Algorithm Power:</b> Decision Trees, P, NP, and NP-Complete Problems. <b>Coping with the Limitations of Algorithm Power:</b> Backtracking (n-Queens Problem), Branch-and-Bound (Knapsack Problem).	
<b>Text Books:</b>	
Introduction to the Design and Analysis of Algorithms, By Anany Levitin, 3rd Edition (Indian), 2017, Pearson.	

**Reference Books:**

1. Computer Algorithms/C++, Ellis Horowitz, SatrajSahni and Rajasekaran, 2nd Edition, 2014, Universities
2. Press.
3. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.

Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education).

<b>Course Code</b>		<b>BISA405C</b>	<b>Introduction to Artificial Intelligence</b>	<b>Semester</b>	:	<b>04</b>	
<b>L:T:P</b>	:	<b>3L-0T-0P</b>		<b>Course Type</b>	:	<b>Theory</b>	
<b>Hours/Sem.</b>	<b>Teaching</b>	:		<b>42 Hrs</b>	<b>CIE Marks</b>	:	<b>50</b>
	<b>Learning (TW+SL)</b>	:		<b>42 Hrs</b>	<b>SEE Marks</b>	:	<b>50</b>
	<b>Exam</b>	:		<b>06 Hrs</b>	<b>Total Marks</b>	:	<b>100</b>
	<b>Total Hrs.</b>	:	<b>90 Hrs</b>	<b>Credits</b>	:	<b>03</b>	

<b>Professional Competency:</b>	
Ability to design, deploy, and maintain basic IoT solutions by integrating sensors, actuators, communication networks, data processing, and cloud platforms to monitor, analyse, and control real-world systems in entry-level industry applications.	
<b>Course Outcomes:</b>	
<ol style="list-style-type: none"> <li><b>Apply</b> machine learning algorithms to solve real world problems.</li> <li><b>Demonstrate</b> concepts of deep learning.</li> <li><b>Describe</b> natural language processing techniques.</li> <li><b>Understand</b> notions of generative AI.</li> </ol>	
<b>Module-I</b>	<b>10 Hrs.</b>
Machine Learning Overview and Basics, Data Pre-processing and Feature Engineering, Supervised Learning: Linear Regression, Supervised Learning: Logistic Regression, Decision Tree, Support Vector Machines, Ensemble Methods and Random Forest (RF), Unsupervised Learning Algorithms, Model Evaluation and Tuning.	
<b>Module-II</b>	<b>8 Hrs.</b>
Artificial Neural Networks, Recurrent Neural Networks (RNNs), Computer Vision with Convolution Neural Network, Generative Models, Deep Reinforcement Learning, Model Tuning and Optimization in Deep Learning.	
<b>Module-III</b>	<b>8 Hrs.</b>
Overview of Natural Language Processing, Text Preprocessing: The Foundation of NLP, Feature Extraction and Representation, Text Classification, Named Entity Recognition (NER), Text Summarization, Topic Modeling, Transformers and Advanced NLP Techniques.	
<b>Module-IV</b>	<b>8 Hrs.</b>
Introduction to Generative AI, Large Language Models (LLMs), Prompt Engineering, Building LLM Apps with Lang Chain, Multi-Modal Language Models, Agentic AI.	
<b>Text Books:</b>	
<a href="#">Dibyendu Banerjee</a> , <a href="#">Sourav Kairi</a> , <a href="#">Shatabdi Mondal</a> , 2025, “The Theory and Practice of Artificial Intelligence: A Handbook for Beginners to Experts”, Shroff Publishers.	
<b>Reference Books:</b>	
<ol style="list-style-type: none"> <li>C.L.Lin, “Elements of Discrete Mathematics” 2<sup>nd</sup> Editions</li> <li>Thomas Khoshy “Discrete Mathematics with applications”</li> <li>Richard Johasonbangh “Discrete Mathematics” 6<sup>th</sup> Edition</li> </ol>	

Course Outcomes	Programme Outcomes (POs)											Programme Specific Outcomes (PSOs)		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
<b>CO1</b>	1	1	2									1		
<b>CO2</b>	1	1	2		1							1	1	
<b>CO3</b>	1	1	2									1	1	
<b>CO4</b>	1	1	2		1						1	1	1	1

<b>Course Code</b>	:	<b>BISA406C</b>	<b>Discrete Mathematical Structures</b>	<b>Semester</b>	:	<b>4</b>	
<b>L:T:P</b>	:	<b>2L-2T-0P</b>		<b>Course Type</b>	:	<b>Theory</b>	
<b>Hours/Sem.</b>	<b>Teaching</b>	:		<b>28 Hrs.</b>	<b>CIE Marks</b>	:	<b>50</b>
	<b>Learning (TW+SL)</b>	:		<b>28+28 Hrs.</b>	<b>SEE Marks</b>	:	<b>50</b>
	<b>Exam</b>	:		<b>06 Hrs.</b>	<b>Total Marks</b>	:	<b>100</b>
	<b>Total Hrs.</b>	:		<b>90 Hrs.</b>	<b>Credits</b>	:	<b>03</b>

**Professional Competency:**

Students will be able to apply discrete mathematical foundations to model, analyze, and solve computational problems effectively in entry-level software and IT roles.

**Course Outcomes:**

1. Apply Fundamentals of computational theory and basic terminology to solve real world problems.
2. Model and solve problems on DFA, Minimization of DFA, NFA, Epsilon-NFA and conversion between them.
3. Prove the properties of regular languages using regular expressions.
4. Design context-free grammars (CFGs) and pushdown automata (PDAs) for formal languages.

Design Turing machines to solve the computational problems.

**Module-I**

**10 Hrs.**

**Fundamentals Principles of Counting:** The Rules of sum and product, permutations, combinations: the binomial theorem, combinations with repetition, mathematical induction.

**Recurrence relation:** first order linear recurrence relation, the second order linear homogeneous recurrence relation with constant coefficient.

**Module-II**

**8 Hrs.**

**Fundamentals of Logic:** Basic connectives and truth tables, Logical equivalence: the laws of logic, logical implication: rules of inference, the use of quantifiers, definitions and the proofs of theorems.

**Set Theory:** Sets and subsets, set operations and the laws of set theory.

**Module-III**

**8 Hrs.**

**Relations and Functions:** Cartesian products and relations, functions: plain and one to one, on to functions: sterling numbers of the second kind, special functions, the pigeonhole principle, function composition and inverse functions, properties of relations, computer recognition: zero one matrices and directed graphs, partial order: Hasse diagram, equivalence relations and partitions, lattices.

**Semi-groups and Groups:** Definition, example and elementary properties, Homomorphism, Isomorphism.

**Module-IV**

**8 Hrs.**

**An introduction to graph theory:** Definitions and examples, sub graphs, complement and graph isomorphism, vertex degree: Euler trails and circuits.

**Trees:** Definitions, properties and examples, rooted trees, trees and sorting weighted trees and prefix codes.

**Text Books:**

4. Discrete and Combinatorial Mathematics-"An Applied Introduction", Ralph P Grimaldi, Pearson Education, 4<sup>th</sup> and 5<sup>th</sup> Edition

**Reference Books:**

5. C.L.Lin, "Elements of Discrete Mathematics" 2<sup>nd</sup> Editions
6. Thomas Khoshy "Discrete Mathematics with applications"
7. Richard Johasonbangh "Discrete Mathematics" 6<sup>th</sup> Edition

<b>Course Code</b>		: <b>BBTA402C</b>	<b>Biology for Engineers</b>	<b>Semester</b>	: <b>4</b>
<b>L:T:P</b>		: <b>3L-0T-0P</b>		<b>Course Type</b>	: <b>Theory</b>
<b>Hours/Sem.</b>	<b>Teaching</b>	: <b>42 Hrs</b>		<b>CIE Marks</b>	: <b>50</b>
	<b>Learning (TW+SL)</b>	: <b>42 Hrs</b>		<b>SEE Marks</b>	: <b>50</b>
	<b>Exam</b>	: <b>06 Hrs</b>		<b>Total Marks</b>	: <b>100</b>
	<b>Total Hrs.</b>	: <b>90 Hrs</b>	<b>Credits</b>	: <b>03</b>	

### Professional Competency:

Understanding fundamentals of biological processes and apply to solve engineering problems and bridging technical engineering with living systems.

### Course Outcomes:

#### After completion of the course student will be able to

To familiarize the students with the basic biological aspects.

To enable the students to apply biological concepts for engineering applications.

To show the students how nature and biological systems inspire building sustainable solutions and technologies.

To motivate the students to develop the interdisciplinary vision of biological engineering..

### Unit-I

**10 Hrs.**

#### Introduction to Biology:

The cell: Structure, and functions of a cell. Biomolecules: Properties and functions of Carbohydrates, Nucleic acids, Proteins and Lipids. Importance of special biomolecules: Enzymes, vitamins and hormones -properties and functions.

#### Biomolecules and their Applications:

Carbohydrates in cellulose-based water filters production, PHA and PLA in bioplastics production, Nucleic acids in vaccines and diagnosis, Proteins in food production, Lipids in biodiesel, Enzymes in biosensor fabrication, food processing, detergent formulation and textile processing.

### Unit-II

**10 Hrs.**

#### Bio Inspiration Models Used In Engineering:

Bio inspiration - Introduction, Alliance between Engineering and Biology, Biomimicry - Science mimicking nature. Bird flying (GPS and aircrafts), Lotus leaf effect (Super hydrophobic and self-cleaning surfaces), Gecko Feet, Plant burrs (Velcro), Shark skin (Friction reducing swimsuits), Kingfisher beak (Bullet train), Fire fly LED.

#### Nature Bioinspired Materials And Mechanisms:

BioEcholocation (ultrasonography, sonars), Photosynthesis (photovoltaic cells, bionic leaf), Respiration (MFCs) Human Blood substitutes-hemoglobin based oxygen carriers (HBOCs) and perfluorocarbons (PFCs).

### Unit-III

**10 Hrs.**

#### Human Organ Systems And Bio Designs

**Brain** as a CPU system (architecture, CNS and Peripheral Nervous System, signal transmission, EEG, Robotic arms for prosthetics. Engineering solutions for Parkinson's disease).

**Heart** as a pump system (architecture, electrical signalling - ECG monitoring and heart related issues, reasons for blockages of blood vessels, design of stents, pace makers, defibrillators).

**Lungs** as purification system gas exchange mechanisms, spirometry, Ventilators, Heart-lung machine).

**Eye** as a Camera system, bionic eye. **Kidney** as a filtration system - dialysis systems.

### Unit-IV

**10 Hrs.**

#### Trends In Bioengineering

Bioprinting techniques and materials, 3D printing of ear, bone and skin, electrical tongue and electrical nose in food science, Self-healing bioconcrete (based on bacillus spores, calcium lactate

nutrients and biomineralization processes), Biomining via microbial surface adsorption. Artificial Intelligence for disease diagnosis. Biochips & their applications. Biosensors & their applications.

### Text Books:

Biology for Engineers, Rajendra Singh C and Rathnakar Rao N, Rajendra Singh C and Rathnakar Rao N Publishing, Bengaluru, 2023.

Biology for Engineers, Thyagarajan S., Selvamurugan N., Rajesh M.P., Nazeer R.A., Thilagaraj W., Barathi S., and Jaganthan M.K., Tata McGraw-Hill, New Delhi, 2012.

Biology for Engineers, Arthur T. Johnson, CRC Press, Taylor and Francis, 2011

Biomedical Instrumentation, Leslie Cromwell, Prentice Hall 2011.

Biology for Engineers, Sohini Singh and Tanu Allen, Vayu Education of India, New Delhi, 2014.

Biomimetics: Nature-Based Innovation, Yoseph Bar-Cohen, 1st edition, 2012, CRC Press.

3D Bioprinting: Fundamentals, Principles and Applications by Ibrahim Ozbolat, Academic Press, 2016.

Electronic Noses and Tongues in Food Science, Maria Rodriguez Mende, Academic Press, 2016

### Reference Books:

<https://nptel.ac.in/courses/121106008>

<https://freevidelectures.com/course/4877/nptel-biology-engineers-other-non-biologists>

<https://ocw.mit.edu/courses/20-020-introduction-to-biological-engineering-design-spring-2009>

<https://ocw.mit.edu/courses/20-010j-introduction-to-bioengineering-be-010j-spring-2006>

<https://www.coursera.org/courses?query=biology>

[https://onlinecourses.nptel.ac.in/noc19\\_ge31/preview](https://onlinecourses.nptel.ac.in/noc19_ge31/preview)

<https://www.classcentral.com/subject/biology>

<https://www.futurelearn.com/courses/biology-basic-concepts>

Course Outcomes	Programme Outcomes											Programme Specific Outcomes		
	1	2	3	4	5	6	7	8	9	10	11	PSO1	PSO2	PSO3
CO 1	3	3	2	2	2						3	3	2	2
CO 2	3	3	2	2	3						3	3	2	2
CO 3	3	3	2	2	3						3	3	2	2
CO 4	3	3	2	2	3						3	3	2	2

<b>Course Code</b>	:	<b>BISA407C</b>	<b>Theory of Computations</b>	<b>Semester</b>	:	<b>4</b>	
<b>L:T:P</b>	:	<b>3L-0T-0P</b>		<b>Course Type</b>	:	<b>Theory</b>	
<b>Hours/Sem.</b>	<b>Teaching</b>	:		<b>42 Hrs</b>	<b>CIE Marks</b>	:	<b>50</b>
	<b>Learning (TW+SL)</b>	:		<b>42 Hrs</b>	<b>SEE Marks</b>	:	<b>50</b>
	<b>Exam</b>	:		<b>06 Hrs</b>	<b>Total Marks</b>	:	<b>100</b>
	<b>Total Hrs.</b>	:		<b>90 Hrs</b>	<b>Credits</b>	:	<b>03</b>

<b>Professional Competency:</b>	
Ability to model, analyze, and design computational systems using automata, formal languages, and Turing machines to solve fundamental problems in computer science	
<b>Course Outcomes:</b>	
<ol style="list-style-type: none"> <li>1. Apply Fundamentals of computational theory and basic terminology to solve real world problems.</li> <li>2. Model and solve problems on DFA, Minimization of DFA, NFA, Epsilon-NFA and conversion between them.</li> <li>3. Prove the properties of regular languages using regular expressions.</li> <li>4. Design context-free grammars (CFGs) and pushdown automata (PDAs) for formal languages.</li> </ol> Design Turing machines to solve the computational problems.	
<b>Module-I</b>	10 Hrs.
Automata: Introduction to Finite Automata, The central concepts of Automata theory. Finite Automata: Deterministic Finite automata, Non-Deterministic Finite Automata. An application of Finite Automata, and Finite Automata with Epsilon-transitions, Regular Expressions: Regular expressions, Finite Automata and Regular Expressions, and Applications of Regular Expressions	
<b>Module-II</b>	8 Hrs.
Properties of Regular Languages: Proving languages not to be regular languages, Closure properties of regular languages, Decision properties of regular languages, and Equivalence and Minimization of Automata. Context Free Grammars and Languages: Context Free Grammars, Parse trees, Applications of Context Free Grammars, Ambiguity in Grammars and Languages.	
<b>Module-III</b>	8 Hrs.
Properties of Context-Free Languages: Normal forms for Context Free Grammars. Pushdown Automata: Definition of the Pushdown Automaton, The languages of a PDA, Deterministic Pushdown Automata.	
<b>Module-IV</b>	8 Hrs.
Introduction To Turing Machine: The Turing Machine, Programming Techniques for Turing Machines, Extensions to the basic Turing Machines, Turing Machine and Computers.	
<b>Text Books:</b>	
Sudip Misra, Anandarup Mukherjee, Arijit Roy, "Introduction to IoT", Cambridge University Press 2021.	
<b>Reference Books:</b>	
<ol style="list-style-type: none"> <li>1. John. E., Hopcroft, Rajeev. Motwani, Jeffrey. D., Ullman, 2007, "Introduction to Automata Theory, Languages and Computation", 3<sup>rd</sup> Edition, Pearson Education.</li> <li>2. Peter. Linz, "An Introduction to Formal Languages and Automata", Third Edition, Fifth printing.</li> <li>3. John, E., Hopcroft, Jeffrey. D. Ullman, "Introduction to Automata Theory, Languages and Computation", Narosa Publication.</li> </ol>	

Course Outcomes	Programme Outcomes (POs)											Program Specific Outcomes (PSOs)		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
<b>CO1</b>	2	3	3	3	-	-	-	-	-	-	1	2	1	2
<b>CO1</b>	3	2	3	2	2	-	-	-	-	-	1	3	3	2
<b>CO2</b>	3	2	2	3	-	-	-	-	-	-	1	3	2	-
<b>CO3</b>	3	2	3	2	2	-	-	-	-	-	1	3	3	2
<b>CO4</b>	3	2	3	2	2	-	-	-	-	-	1	3	3	2