Vidya Pratishthan's
Kamalnayan Bajaj Institute of Engineering and Technology
(Autonomous Institute)

Faculty of Science and Technology

Board of Studies

Information Technology

Syllabus

Third Year B.Tech. Information Technology

(2024 Pattern)

(w.e.f. AY: 2026-27)

Institute Vision and Mission

Vision

To achieve Academic Excellence through Persistent and Synergic Collaborations amongst all Stakeholders.

Mission

- To ensure holistic development of students as lifelong learners and problem solvers through value-based quality education.
- To motivate faculty to attain the state-of-the-art knowledge and wisdom in their domain and be a facilitator towards co-creation of knowledge.
- To frame and deploy conducive and empowering policies for multifaceted growth of students, faculty and staff to make them contributors towards excellence.
- To partner with industry for mutually beneficial relations to generate employable and deployable workforce.
- To fulfill the aspirations of alumni, parents, society, region and nation at large by generating technically competent and contributing manpower.

Department Vision and Mission

Vision

To develop globally competent IT professionals through continuous learning.

Mission

- To provide graduates with the programming skills and domain knowledge.
- To collaborate with local, state, national, and international entities in education.
- To develop technically competent professionals with social values and ethics.
- To encourage faculty to acquire state-of-the art knowledge.

Program Educational Objectives	
PEO1	Graduates of the program will possess strong fundamental concepts in mathematics, science, engineering and Technology to address technological challenges.
PEO2	Possess knowledge and skills in the field of Computer Science & Engineering and Information Technology for analyzing, designing and implementing complex engineering problems of any domain with innovative approaches.
PEO3	Possess an attitude and aptitude for research, entrepreneurship and higher studies in the field of Computer Science & Engineering and Information Technology.
PEO4	Have commitment to ethical practices, societal contributions through communities and lifelong learning.
PEO5	Possess better communication, presentation, time management and team work skills leading to responsible & competent professionals and will be able to address challenges in the field of IT at global level.

Program Outcomes		
PO1	Engineering knowledge	Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.
PO2	Problem analysis	Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)
PO3	Design / Development of Solutions	Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)
PO4	Conduct Investigation of Complex Problems	Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).
PO5	Engineering Tool Usage	Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)
PO6	The Engineer and The World	Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).
PO7	Ethics	Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)
PO8	Individual and Collaborative Team work	Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
PO9	Communication	Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences
PO10	Project Management and Finance	Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.
PO11	Life-Long Learning	Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change.(WK8)

Program Specific Outcomes(PSO)	
PSO1	Students will be able to apply engineering principles and practices for the development and maintenance of software system.
PSO2	Students will be able to undertake a team project by following professional ethical practices.
PSO3	Students will be able to make successful career in IT industry meeting the requirement of industries.

SEMESTER-V

Course Type	Course Code	Course Name	Teaching Scheme			Examination Scheme and Marks							Credits			
			TH	PR	TUT	CAA	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
PCC	IT24301 TH	Theory of Computation	2	-	-	-	-	60	-	-	-	60	2	-	-	3
	IT24301 TUT		-	-	1	-	-	-	40	-	-	40	-	-	1	
PCC	IT24302 TH	Operating System	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24302 PR		-	2	-	-	-	-	-	30	-	30	-	1	-	
PCC	IT24303 TH	Deep Learning	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24303 PR		-	2	-	-	-	-	-	30	-	30	-	1	-	
PEC	IT24304 TH	Program Elective 1	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24304 PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
MDM	MD24XXX TH	MDM	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	MD24XXX PR		-	2	-	-	-	-	30	-	-	30	-	1	-	
VSEC	IT24305 PR	Advanced JAVA Programming	-	2	-	-	-	-	-	-	30	30	-	1	-	2
	IT24305 TUT		-	-	1	-	-	-	40	-	-	40	-	-	1	
AU	HS24301	Constitution of India	1	-	-	-	-	-	-	-	-	-	-	-	-	AU
Total			**15**	**10**	**2**	**40**	**120**	**300**	**110**	**60**	**60**	**690**	**14**	**5**	**2**	**21**

SEMESTER-VI

Course Type	Course Code	Course Name	Teaching Scheme			Examination Scheme and Marks							Credits			
			TH	PR	TUT	CAA	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
PCC	IT24311TH	Design and Analysis of Algorithms	3	-	-	10	30	60	-	-	-	100	3	-	-	3
PCC	IT24312TH	Computer Networks and Security	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24312PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
PEC	IT24313TH	Programme Elective 2	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24313PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
PEC	IT24314TH	Programme Elective 3	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24314PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
MDM	MD24XXXTH	Multi-disciplinary minor	2	-	-	10	-	60	-	-	-	70	2	-	-	3
	MD24XXXPR		-	2	-	-	-	-	30	-	-	30	-	1	-	
OE	OE240XX	Open Elective	3	-	-	10	30	60	-	-	-	100	3	-	-	3
AU	HS24311	Democracy, Election and Governance	1	-	-	-	-	-	-	-	-	-	-	-	-	AU
	Total		**18**	**8**	-	**60**	**150**	**360**	**30**	-	**90**	**690**	**17**	**4**	-	**21**

Prof.S.A.Takale
HOD-IT

Dr.A.H. Kolekar
CoE

28/03/26
Prof.S.M.Bhosle
Dean Academics

Prof.S.B.Lande
Principal

VPKBIET BARAMATI
Affiliated to S.P.P.U. Pune Id No./PU/PN/Engg./152/2000
PUNE (M.S.)

Principal
Vidya Pratishtham's
Kamalnayan Bajaj Institute of
Engineering & Technology, Baramati
Vidyanagari, Baramati-413133

Vidya Pratishthan's
Kamalnayan Bajaj Institute of Engineering and Technology
Board of Studies: Information Technology

Syllabus: Third Year B. Tech. Information Technology
2024 Pattern w.e.f. AY:2026-2027

SEMESTER-V

Course Type	Course Code	Course Name	Teaching Scheme			Examination Scheme and Marks							Credits			
			TH	PR	TUT	CAA	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
PCC	IT24301 TH	Theory of Computation	2	-	-	-	-	60	-	-	-	60	2	-	-	3
	IT24301 TUT		-	-	1	-	-	-	40	-	-	40	-	-	1	
PCC	IT24302 TH	Operating System	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24302 PR		-	2	-	-	-	-	-	30	-	30	-	1	-	
PCC	IT24303 TH	Deep Learning	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24303 PR		-	2	-	-	-	-	-	30	-	30	-	1	-	
PEC	IT24304 TH	Program Elective 1	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24304 PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
MDM	MD24XXX TH	MDM	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	MD24XXX PR		-	2	-	-	-	-	30	-	-	30	-	1	-	
VSEC	IT24305 PR	Advanced JAVA Programming	-	2	-	-	-	-	-	-	30	30	-	1	-	2
	IT24305 TUT		-	-	1	-	-	-	40	-	-	40	-	-	1	
AU	HS24301	Constitution of India	1	-	-	-	-	-	-	-	-	-	-	-	-	AU
Total			**15**	**10**	**2**	**40**	**120**	**300**	**110**	**60**	**60**	**690**	**14**	**5**	**2**	**21**

Sr. No	MDM Code	MDM Course
1	ET24053	IOT

Elective –I		
Sr. No	**Course Code**	**Course**
1	IT24304A	Computer Graphics
2	IT24304B	Software Testing and Automation
3	IT24304C	Python for Data Analytics

SEMESTER-VI

Course Type	Course Code	Course Name	Teaching Scheme			Examination Scheme and Marks							Credits			
			TH	PR	TUT	CAA	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
PCC	IT24311TH	Design and Analysis of Algorithms	3	-	-	10	30	60	-	-	-	100	3	-	-	3
PCC	IT24312TH	Computer Networks and Security	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24312PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
PEC	IT24313TH	Programme Elective 2	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24313PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
PEC	IT24314TH	Programme Elective 3	3	-	-	10	30	60	-	-	-	100	3	-	-	4
	IT24314PR		-	2	-	-	-	-	-	-	30	30	-	1	-	
MDM	MD24XXX TH	Multi-disciplinary minor	2	-	-	10	-	60	-	-	-	70	2	-	-	3
	MD24XXX PR		-	2	-	-	-	-	30	-	-	30	-	1	-	
OE	OE240XX	Open Elective	3	-	-	10	30	60	-	-	-	100	3	-	-	3
AU	HS24311	Democracy, Election and Governance	1	-	-	-	-	-	-	-	-	-	-	-	-	AU
	Total		**18**	**8**	**-**	**60**	**150**	**360**	**30**	**-**	**90**	**690**	**17**	**4**	**-**	**21**

Sr. No	MDM Code	MDM Course
1	ME24053	3D Printing

Elective –II			Elective -III	
Sr. No	**Course Code**	**Course**	**Course Code**	**Course**
1	IT24313A	Fullstack	IT24314A	Introduction to Internet of Things
2	IT24313B	Introduction to Artificial Intelligence	IT24314B	Cyber Security
3	IT24313C	Compiler Design	IT24314C	Computer Vision

Multidisciplinary Minor Courses			
Course Code	**3 Credit Course Name**	**Course Code**	**4 Credit Course Name**
ET24051	Introduction to Embedded system	ET24052	Drone technology
EL24051	Photovoltaic Technology & Solar Power System	AI24051	Introduction to AIML
IT24054	Data Structure	CO24053	Object Oriented Programming
CE24052	Green Building & Smart Cities	GS24052	Linear Algebra and Statistics
ME24051	Introduction to 3D Printing	AI250201	AIML
ME24052	Introduction to Robotics and Automation	ET24054	Microcontroller Architecture
		ME24053	3D Printing
		ME24054	Robotics and Automation
		ET24053	Internet of Things

Open Elective Courses			
Course Code	**2 Credit Course Name**	**Course Code**	**3 Credit Course Name**
OE23004	Industrial Management	OE23015	Design Thinking
OE23002	Professional Leadership	OE23016	Accounting and Finance
OE23003	Organizational Behavior	OE23017	Sustainability & Climate Change
OE23009	Cyber Laws	OE23018	Agriculture Technology
OE23008	Intellectual Property Rights	OE23019	Architectural Technology
OE23005	Disaster Management		
OE24001	Digital Marketing		

SEMESTER-V

Theory of Computation								
Course Code : IT24301			**Course Credits: 3**			**Course type: PCC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
2	**-**	**1**	**-**	**-**	**60**	**40**	**-**	**-**

Prerequisite Course Mapping:
1. Data Structure

Future Course Mapping:
1. Compiler Design

Course Objective:
1.To know the applicability of the model of computation to different problems.
2.To understand in detail the relationship among formal languages, formal grammars and automata.
3.To learn the design of Finite Automata, Pushdown Automata and Turing Machine for processing of formal languages.

Course Outcomes: On completion of the course, learner will be able to
1. **Construct** finite automata and its variants to solve computing problems.
2. **Develop** regular expressions for the regular languages and finite automata.
3. **Identify** types of grammar, design and simplify Context Free Grammar.
4. **Construct** Pushdown Automata machine for the CFL and Turing Machine for formal languages.

Syllabus

UNIT No.	Syllabus	Hrs
I	**Finite Automata** Basic Concepts: Symbols, Strings, Language, Formal Language. Finite Automata (FA): Formal definition and notations for FSM, Concept of state transition diagram and transition table for FA, Construction of DFA, NFA, NFA with epsilon moves. Conversion of NFA with epsilon moves to NFA, Conversion of NFA to DFA, and Conversion of NFA with epsilon moves to DFA, Minimization of FA, Equivalence of FAs, and Applications of FA. Finite State Machine with output: Moore and Mealy machines - Definition, Construction, Inter- Conversion.	7
II	**Regular Expressions and Languages** Regular Expressions (RE) : Definition and Identities of RE, Operators of RE, Equivalence of two regular expressions, Equivalence of regular expressions and regular languages (RL), Conversion of RE to FA using direct method, Conversion of FA to RE using Arden's theorem, Pumping lemma for RLs, Closure properties of RLs, Applications of Regular Expressions.	6
III	**Context Free Grammer and Language** Grammar: Introduction and representation, Chomsky Hierarchy, Formal definition of Regular Grammar (RG), Conversions: LRG to RLG, RLG to LRG, RG to FA, FA to RG. Context Free Grammar (CFG): Definition of CFG, Derivation tree, sentential forms, Leftmost and Rightmost derivations, Ambiguous Grammar and unambiguous grammar, Context Free Language (CFL). Grammar Simplification, Normal forms: Chomsky Normal Form.	6
IV	**Pushdown Automata** Pushdown Automata (PDA) : Introduction and formal definition of PDA, Construction of Transition diagram and Transition table for PDA, Instantaneous Description of PDA, Equivalence of Acceptance by Final State & Empty stack, Deterministic PDA and Nondeterministic PDA, Conversion of CFG to PDA. **Turing Machine** Turing Machine (TM): Formal definition of a Turing machine, Design of Turing machines, Variants of Turing Machines: Deterministic TM, Nondeterministic TM, Multi-tape TM,	7

	Universal Turing Machine, Recursive Languages and Recursively Enumerable Languages. Decidability: Decidable problems concerning regular languages, Decidable problems concerning context free languages, Un-decidability.	
Total Teaching Hours		**26**

List of Tutorials

1. Design of Regular Expression from Language
2. Design Deterministic Finite Automata
3. NFA design and NFA to DFA conversion
4. RE to NFA with null moves and NFA with null moves to NFA without null moves
5. Formal language to CFG and CFG to language conversion
6. Simplification of CFG and Chomsky Normal Form
7. Design of Push down Automata
8. Design of Turing Machine

Text Books:

1. John C. Martin, Introduction to Language and Theory of Computation, TMH, 3rd Edition, ISBN: 978-0070660489.
2. Vivek Kulkarni, Theory of Computation, Oxford University Press,ISBN- 13 : 978-0198084587.

Reference Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, Introduction to Automata Theory Languages and Computation, Addison-Wesley, ISBN 0-201-44124-1.
2. K.L.P Mishra, N. Chandrasekaran, Theory of Computer Science : Automata, Languages and Computation, Prentice Hall India, 2nd Edition.
3. Michael Sipser, Introduction to the Theory of Computation, CENGAGE Learning, 3rd Edition ISBN-13:978-81-315-2529-6.
4. Daniel Cohen, “Introduction to Computer Theory”, Wiley & Sons, ISBN 97881265133454.
5. Kavi Mahesh, “Theory of Computation: A Problem-Solving Approach”, Wiley India, ISBN-1081265331106.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc26_cs85/preview
2. https://theory.cs.princeton.edu/complexity/book.pdf
3. https://cglab.ca/~michiel/TheoryOfComputation/TheoryOfComputation.pdf

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24301	CO1	2	2	2	2	1	-	-	-	2	-	-
	CO2	2	2	2	2	1	-	-	-	2	-	-
	CO3	2	2	2	2	1	-	-	-	2	-	-
	CO4	2	3	2	2	1	-	-	-	2	-	-

Operating System								
Course Code : IT24302			**Course Credits: 04**			**Course type: PCC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**2**	**–**	**10**	**30**	**60**	**-**	**30**	**-**

Prerequisite Course Mapping:
1. Basics of Computer Programming

Future Course Mapping:
1. Ethical Hacking

Importance of Course: Students will learn what task OS is doing

Course Objectives:
1. Understand the basic concept and functions of the operating system
2. Understand the concept of Process and Thread management including scheduling synchronization, and deadlocks
3. Analyze the memory management techniques
4. Understand I/O Management and File System
5. Apply the Protection and Security to OS

Course Outcomes:
1. **Understanding** the role of Modern Operating Systems
2. **Apply** the concepts of process and thread scheduling
3. **Apply** the concept of process synchronization, mutual exclusion and the deadlock
4. **Understand** and apply the concepts of various memory management techniques
5. **Make** use of concept of I/O management and File system.
6. **Understand** Important of System software

Unit No.	Syllabus	Hrs
1	**Introduction** Operating System Objectives, The Evolution of Operating System, OS Design Considerations for Multiprocessor and Multicore OS, Architectures of Operating System: Monolithic, Microkernel, Exokernel, Introduction to Linux OS, Basic Shell Commands, Shell Scripting using BASH.	6
2	**Process Management** Process Concept, Process States, Process Control Block, Process Description Threads: Process and Threads, Basic types of threads, Multithreading, Thread Programming using thread library APIs Process Scheduling: Types of Scheduling, Scheduling Criteria, Scheduling Algorithms: First-Come First- Served, Shortest-Job-First, Priority, Round Robin, Case Study - Linux Scheduling	7
3	**Process Synchronization** Principles of Concurrency, Critical - Section Problem, Mutual Exclusion: Requirements, Operating System support - Semaphore and Mutex, Classical Synchronization Problems: Reader-Writer Problem, Producer - Consumer Problem, Real Life Problems, Inter-Process Communication: Pipes and Shared Memory Deadlock: Principles of Deadlock, Deadlock Characterization: Necessary Conditions, Resource - Allocation Graph, Methods for Handling Deadlock: Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery, Case Study: Dining Philosopher Problem	7
4	**Memory Management** Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Buddy System, Paging, Segmentation, Virtual Memory: Demand Paging, Page Replacement, Thrashing, Case Study: Linux Operating System	6

5	**Input / Output and File Management** I/O Management: I/O Devices, Organization of the I/O Function, I/O Buffering, Secondary Storage Management: Disk Structure, Disk Scheduling File Management: Overview-Files and File Systems, File structure. File Organization and Access, File Directories, File Sharing, Case Study: Linux File System, Android File System	7
6	**Protection and Security** Goals of protection, Domain of protection, Access matrix, Implementation of access matrix, Revocation of access rights, Security problems, Authentication, Program threats, System threats, Threat monitoring	6

Practical Assignments

Assign No	Title
1	Study of Basic Linux Commands: echo, ls, read, cat, touch, test, loops, arithmetic comparison, conditional loops, grep, sed find, diff, tac and Bash scripting
2	Implement the C program in which the main function accepts the integers to be sorted. Main function uses the FORK system call to create a new process called a child process. Parent process sorts the integers using a sorting algorithm and waits for the child process using WAIT system call to sort the integers using any sorting algorithm. Also demonstrate zombie and orphan states.
3	Implement the C program to simulate any 2 CPU Scheduling Algorithms FCFS,Round Robin and Priority Scehduling(1 preemptive and 1 non- preemptive) with different arrival times.
4	Implement the C program for Producer Consumer problem using counting semaphores and mutex/binary semaphore.
5	Implement the C program for Reader-Writer problem with reader priority.
6	Implement the C program for Deadlock Avoidance Algorithm: Bankers Algorithm.
7	Implement the C program for any one Page Replacement Algorithm with minimum three frames as an input.(LRU,Optimal)
8	Implement Full duplex communication between two independent processes. First process accepts sentences and writes on the first pipe to be read by the second process. Second process counts number of characters, number of words and number of lines in accepted sentences, and writes the contents on second pipe to be read by first process and displays on standard output

Text Books :

1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition,2014, ISBN-10: 0133805913 • ISBN-13: 9780133805918
2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, John Wiley & Sons ,Inc., 9th Edition,2012, ISBN 978-1-118-06333-0
3. Arnold Robbins, Nelson H. F. Beebe, Classic Shell Scripting, O'Reilly Media, Inc., 2005, ISBN 9780596005955

Reference Books:

1. Tom Adelstein and Bill Lubanovic, Linux System Administration, O'Reilly Media, ISBN-10: 0596009526, ISBN-13: 978-0596009526.
2. Harvey M. Deitel, Operating Systems, Prentice Hall, ISBN-10: 0131828274, ISBN-13: 978-0131828278.
3. Thomas W. Doeppner, Operating System in depth: Design and Programming, WILEY, ISBN: 978- 0-471-68723-8
4. Mendel Cooper, Advanced Shell Scripting, Linux Documentation Project.
5. Andrew S. Tanenbaum & Herbert Bos, Modern Operating System, Pearson, ISBN-13: 9780133592221, 4th Edition.

Online Resources:

1.NPTEL Course "**Introduction to OS**"
https://onlinecourses.nptel.ac.in/noc21_cs72/preview

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24302	CO1	3	3	2	2	1	-	-	-	2	-	-
	CO2	3	2	2	2	1	-	-	-	2	-	-
	CO3	3	3	2	2	1	-	-	-	2	-	-
	CO4	3	3	2	2	1	-	-	-	2	-	-
	CO5	2	1	2	2	1	-	-	-	1	-	-
	CO6	2	2	1	1	-	-	-	-	1	-	-

Deep Learning								
Course Code : IT24303			**Course Credits: 04**			**Course type: PCC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	2	-	10	30	60	-	30	-

Prerequisite Course Mapping: Python Programming, Machine Learning

Future Course Mapping: Reinforcement Learning

Course Objectives:

1. To introduce the theoretical foundations, algorithms, methodologies, and application of neural networks and deep learning.
2. To study and apply Convolution Neural Network for real life applications.
3. To understand the Recurrent and Recursive nets in Deep Learning.
4. Study of unsupervised deep learning algorithms like autoencoder.
5. To study and apply pretrained Deep learning models and study of transfer learning.
6. To build deep architecture to solve real world problem.

Course Outcomes:

1. **Understand** the theoretical foundations, algorithms, and methodologies of Deep Learning
2. **Apply** the concepts of Convolution Neural Networks and use of popular CNN architectures.
3. **Compare** Feed Forward Neural Network and Recurrent Neural Network and develop the application of RNN and LSTM.
4. **Elaborate and** design unsupervised deep learning algorithms like Autoencoders.
5. **Explore** Representation Learning and Transfer Learning techniques using variants of CNN architecture.
6. **Develop** application to solve real life problem

UNIT No.	Syllabus	Hrs
I	**Introduction to Neural Networks :**The Biological Neuron, The Perceptron, Multilayer Feed-Forward Networks , **Training Neural Networks** :Backpropagation and Forward propagation, **Activation Functions** : Linear ,Sigmoid, Tannh, Hard Tanh, Softmax, ReLU, Leaky ReLU, **Loss Functions** : Loss Functions for Regression , Loss Functions for Classification, **Hyperparameters** : Layer size, Learning Rate, Regularization(dropout, drop connect, L1, L2), **Learning Parameters**: Gradient Descent (GD), Stochastic and Mini Batch GD, Momentum Based GD, Nesterov Accelerated GD, Adam.	7
II	**Convolutional Neural Network:** Introduction to CNN, Convolution Operation, Parameter Sharing, Equivalent Representation, Pooling, Variants of the Basic Convolution Function, The basic Architecture of CNN, Popular CNN Architecture – ResNet.	7
III	**Recurrent Neural Networks:** Recurrent Neural Networks: Types of Recurrent Neural Networks, Feed-Forward Neural Networks vs Recurrent Neural Networks, Long Short-Term Memory Networks (LSTM), Gated Recurrent Units (GRUs), vanishing and Exploding gradient, Recursive Neural Networks	7
IV	**Autoencoders:** Encoder Decoder architectures, Undercomplete Autoencoders, Regularized Autoencoders-Sparse Autoencoders, Stochastic Encoders and Decoders, Denoising Autoencoders, Variational Autoencoder, Applications of Autoencoders.	6
V	**Representation Learning:** Greedy Layer wise Pre-training, Transfer Learning and Domain Adaption, Distributed Representation, Variants of CNN: DenseNet.	6
VI	**Advanced Trends and Applications of Deep Learning:** Attention Mechanisms: Recurrent Models of Visual Attention, Attention Mechanisms for Image Captioning, Soft Image Attention with Spatial Transformer, Attention Mechanisms for Machine Translation, Applications of Deep Learning : object detection and classification, face recognition, voice recognition, Semantic Analysis	6

List of Practical Assignments:

1. Implementing Feedforward neural networks with Keras and Tensor Flow a. Import the necessary packages b. Load the training and testing data (MNIST/CIFAR10) c. Define the network architecture using Keras d. Train the model using SGD e. Evaluate the network f. Plot the training loss and accuracy
2. Build the Image classification model by dividing the model into following 4 stages:
    1. Loading and pre-processing the image data
    2. Defining the model's architecture
    3. Training the model
    4. Estimating the model's performance
3. Use Autoencoder to implement anomaly detection. Build the model by using:
    1. Import required libraries
    2. Upload / access the dataset
    3. Encoder converts it into latent representation
    4. Decoder networks convert it back to the original input
    5. Compile the models with Optimizer, Loss, and Evaluation Metrics
4. Implement the Continuous Bag of Words (CBOW) Model. Stages can be:
    1. Data preparation
    2. Generate training data
    3. Train model
    4. Output
5. Object detection using Transfer Learning of CNN architectures
    1. Load in a pre-trained CNN model trained on a large dataset
    2. Freeze parameters (weights) in model's lower convolutional layers
    3. Add custom classifier with several layers of trainable parameters to model
    4. Train classifier layers on training data available for task
    5. Fine-tune hyper parameters and unfreeze more layers as needed
6. Mini Project:
    1. Human Face Recognition
    2. Gender and Age Detection: predict if a person is a male or female and also their age
    3. Colorizing Old B&W Images: color old black and white images to colorful images
    4. Use the Google stock prices dataset and design a time series analysis and prediction system using RNN

Text Books :

1. Goodfellow, I., Bengio, Y.,,Courville, A, "Deep Learning", MIT Press, 2016.
2. Josh Patterson & Adam Gibson, "Deep Learning"
3. Charu Agarwal, "Neural Networks and deep learning", A textbook
4. Nikhil Buduma, "Fundamentals of Deep Learning", SPD
5. Francois chollet, "Deep Learning with Python"

Reference Books:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction"
2. Seth Weidman, "Deep Learning from Scratch: Building with Python from First Principles",O'Reily
3. Francois Duval, "Deep Learning for Beginners, Practical Guide with Python and Tensorflow"

Online Resources:

http://csis.pace.edu/ctappert/cs855-18fall/DeepLearningPractitionersApproach.pdf
https://www.dkriesel.com/_media/science/neuronalenetze-en-zeta2-1col-dkrieselcom.pdf
https://www.my-mooc.com/en/categorie/deep-learning
https://nptel.ac.in/courses/106106184

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24303	CO1	3	3	2	2	3	1	-	1	2	1	-
	CO2	3	2	2	2	2	1	-	1	2	1	1
	CO3	2	2	2	1	3	1	-	1	2	1	1
	CO4	2	2	2	1	2	1	-	1	2	1	-
	CO5	2	2	2	1	2	1	-	1	2	1	1
	CO6	2	1	1	1	-	1	-	-	2	1	-

Program Elective 1: Computer Graphics								
Course Code : IT24304A			**Course Credits: 04**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	2	-	10	30	60	-	-	30

Prerequisite Course Mapping: Engineering Mathematics-I

Future Course Mapping: Computer Vision

Course Objectives:

1. To acquaint the learner with the basic concepts of Computer Graphics.
2. To learn the various algorithms for generating and rendering graphical figures.
3. To get familiar with mathematics behind the graphical transformations.
4. To understand and apply various methods and techniques regarding projections, animation, shading, illumination and lighting.

Course Outcomes:

1. **Identify** the basic terminologies of Computer Graphics and interpret the mathematical foundation of the concepts of computer graphics.
2. **Apply** mathematics to develop Computer programs for elementary graphic operations.
3. **Illustrate** the concepts of windowing and clipping and apply various algorithms to fill and clip polygons.
4. **Apply** and understand the core concepts of computer graphics, including transformation in two and three dimensions, viewing and projection.
5. **Understand** the concepts of color models, lighting, shading models and hidden surface elimination.
6. **Create** effective programs using concepts of curves, fractals, animation and gaming.

UNIT No.	Syllabus	Hrs
I	**Graphics Primitives and Scan Conversion Algorithms:** **Introduction, graphics primitives** - pixel, resolution, aspect ratio, frame buffer. Display devices, applications of computer graphics. **Introduction to OpenGL** - OpenGL architecture, primitives and attributes, simple modeling and rendering of two- and three-dimensional geometric objects, GLUT, interaction, events and call-backs picking. (Simple Interaction with the Mouse and Keyboard) **Scan conversion**: Line drawing algorithms: Digital Differential Analyzer (DDA), Bresenham. **Circle drawing algorithms**: DDA, Bresenham, and Midpoint.	7
II	**Polygon, Windowing and Clipping:** **Polygons**: Introduction to polygon, types: convex, concave and complex. Inside test. **Polygon Filling**: flood fill, seed fill, scan line fill. **Windowing and clipping**: viewing transformations, 2-D clipping: Cohen Sutherland algorithm line Clipping algorithm, Sutherland Hodgeman Polygon clipping algorithm, Weiler Atherton Polygon Clipping algorithm.	7
III	**2-D, 3-D Transformations and Projections:** **2-D transformations**: introduction, homogeneous coordinates, 2-D transformations - Translation, scaling, rotation and shear, rotation about an arbitrary point. **3-D transformations**: introduction, 3-D transformations - Translation, scaling, rotation and shear, rotation about an arbitrary axis. **Projections** : Parallel (Oblique: Cavalier, Cabinet and orthographic: isometric, diametric, trimetric) and Perspective (Vanishing Points 1 point, 2 point and 3 point)	7

IV	**Light, Colour, Shading and Hidden Surfaces:** **Colour models**: Properties of Light, CIE chromaticity Diagram, RGB, HSV, CMY. **Illumination Models**: Ambient Light, Diffuse reflection, Specular Reflection, and the Phong model, Combined diffuse and Specular reflections with multiple light sources, warn model, **Shading Algorithms**: Halftone, Gauraud and Phong Shading. **Hidden Surfaces**: Introduction, Back face detection and removal, Algorithms: Depth buffer (z), Depth sorts (Painter), Area subdivision (Warnock)	6
V	**Curves and Fractals:** **Curves**: Introduction, Interpolation and Approximation, Blending function, B-Spline curve, Bezier curve, **Fractals**: Introduction, Classification, Fractal generation: snowflake, Triadic curve, Hilbert curve, Applications.	6
VI	**Introduction to Animation and Gaming:** **Segment**: Introduction, Segment table, Segment creation, closing, deleting and renaming, Visibility. **Animation**: Introduction, Conventional and computer based animation, Design of animation sequences, Animation languages, Key- frame, Morphing, Motion specification. **Gaming**: Introduction, Gaming platform (NVIDIA, i8060), Advances in Gaming.	6

List of Practical Assignments:

1. Implement DDA and Bresenham line drawing algorithm to draw: i) Simple Line ii) Dotted Line iii) Dashed Line iv) Solid line ;using mouse interface Divide the screen in four quadrants with center as (0, 0). The line should work for all the slopes positive as well as negative.
2. Implement Bresenham circle drawing algorithm to draw any object. The object should be displayed in all the quadrants with respect to center and radius.
3. Implement the following polygon filling methods : i) Flood fill / Seed fill ii) Boundary fill ; using mouse click, keyboard interface and menu driven programming
4. Implement Cohen Sutherland polygon clipping method to clip the polygon with respect the viewport and window. Use mouse click, keyboard interface.
6. Implement following 2D transformations on the object with respect to axis :i) Scaling ii) Rotation about arbitrary point iii) Reflection
7. Generate fractal patterns using i) Bezier ii) Koch Curve
8. Implement animation principles for any object

Text Books :

1. S. Harrington, ―Computer Graphics‖, 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6.
2. D. Rogers, ―Procedural Elements for Computer Graphics‖, 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.
3. Donald D. Hearn, ―Computer Graphics with Open GL‖, 4th Edition, ISBN- 13: 9780136053583.

Reference Books:

1. J. Foley, V. Dam, S. Feiner, J. Hughes, ―Computer Graphics Principles and Practice‖, 2nd Edition, Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
2. D. Rogers, J. Adams, ―Mathematical Elements for Computer Graphics‖, 2nd Edition, Tata McGrawHill Publication, 2002, ISBN 0 – 07 – 048677 – 8.
3. Mario Zechner, Robert Green, ―Beginning Android 4 Games Development‖, Apress, ISBN: 978-81-322-0575-3.

Online Resources:

https://nptel.ac.in/courses/106106090
https://nptel.ac.in/courses/106102065

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24304A	CO1	2	1	1	-	-	-	-	-	-	-	-
	CO2	2	2	2	-	-	-	-	-	-	-	-
	CO3	1	2	-	1	-	-	-	-	-	-	-
	CO4	2	1	1	1	-	-	-	-	-	-	-
	CO5	1	-	1	-	-	-	-	-	-	-	-
	CO6	2	2	1	-	-	-	-	-	-	-	-

Program Elective 1: Software Testing and Automation								
Course Code : IT24304B			**Course Credits: 4**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**2**	**-**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping:
1. Software Engineering

Future Course Mapping:
1. Software Project Management

Course Objective:
1. Learn to apply the testing strategies and methodologies in projects.
2. To understand test management strategies and tools for testing.
3. A keen awareness of the open problems in software testing and maintenance.
4. To explain quality assurance and various automated testing tools used in quality management.
5. To learn in detail about various quality assurance models.
6. to understand the audit and assessment procedures to achieve quality.

Course Outcomes: On completion of the course, learner will be able to
1. Understand the basics of testing.
2. Understand the level of testing
3. Explore the test automation concepts, tools.
4. Apply the standard metrics to measure the software quality.
5. Apply appropriate quality assurance models and develop quality.
6. Ability to conduct formal inspections, record and evaluate results of inspection.

Syllabus

UNIT No.	Syllabus	Hrs
I	**Introduction to Software Testing** Testing as an engineering activity, Need of Software Testing, Definitions: Errors, Defects, Failures, Verification vs Validation, Objectives of Testing, Software Development Life Cycle (SDLC), Software Testing Life Cycle (STLC), Principles of Testing, Testers role in a s/w development organization, Origins of defect, Defect Life Cycle, Defect classes, Defect repository and Test Design	7
II	**Testing Techniques and Levels of Testing** Types of Testing: Static testing Vs Structural Testing, Code functional testing, Coverage and Control Flow Graph. Testing Levels: Unit Testing, Integration Testing, System Testing, Acceptance Testing, Random testing, Regression Testing, Alpha and Beta Testing, Requirements Based Testing, Decision Tables, State Based Testing, Cause Effect Graphing, Error Guessing, Configuration Testing, Compatibility Testing, Smoke and Sanity Testing	7
III	**Software Test Automation and Automation Testing** Manual Testing vs Automation Testing, Need of Automation, Automation Testing Life Cycle, Test Automation Framework, Introduction to Selenium, Selenium Components, Working of Selenium, Locators in Selenium, Handling Forms, Buttons, Alerts, TestNG Framework, JUnit Framework. Automation Tools: SoapUI, Robotic Process Automation, Tosca, Appium.	7

IV	**Quality Metrics & Software Quality Assurance** Tracking the bug, Debugging, Testing Software System Security-Six Sigma, TQM-Complexity Metrics and Models, Quality Management Metrics, Availability Metrics, Availability Metrics, Defect Removal effectiveness, FMEA, Quality Function Deployment, SQA Basics, Components of SQA system, Software quality in business context, Planning for SQA.	6
V	**Quality Assurance Model** Product Quality and Process Quality, Software Process models, 7 Quality Control tools, Models for quality assurance, ISO 9000 series, CMM, CMMI, Test maturity models, SPICE	6
VI	**Software Quality Assurance Trends** Software Process- PSP and TSP, OO Methodology, Clean Room Software Engineering, Defect Injection and Prevention, Internal Auditing and Assessment, Inspection and walkthroughs, Case tools and their effect on software quality, Mobile Application Testing, Cloud Testing	6
Total Teaching Hours		**39**

List of Practical Assignments:

1) Prepare software requirement specifications for any project or problem statement.
2) Write test scenario for the GMAIL login page.
3) Write test cases in excel sheet for Social media application, website or web application. Develop a defect report after testing.
4) Perform White Box Testing.(Statement Coverage, Branch Coverage, Path Coverage, Flow Graph Drawing)
5) Perform automation testing using selenium-Install Selenium IDE / WebDriver. Write first Selenium Script. Record and Play Test Cases. Use Locators (ID, Name, XPath, CSS). Automate Login Form
6) Perform automation testing using TestNG & its Framework. Write TestNG test scripts. Using annotations generating test reports and perform Data-Driven testing.
7) Test any existing website application(IRCTC, Online Shopping Site, Airline Reservation Site) using any automated testing tool.

Text Books:

1. Srinivasan Desikan, Golalaswamy Ramesh, "Software Testing Principles and Practices" Pearson edition.
2. Daniel Galin, "Software Quality Assurance:From theory to implementation" , Pearson Addison Wesley.

Reference Books:

1. Aditya P Mathur, "Foundations of Software Testing", Pearson.
2. Paul Ammann, Jeff Affutt, "Introduction to Software Testing", Cambridge University Press.
3. Renu Rajani, Pradeep Oak, "Software Testing-Effective Methods,Tool and Techniques", Tata McGraw Hills.
4. Stephan Kan, "Metrics and Models in Software Quality" Addison Wesley, 2nd Edition.
5. Willam Perry, " Effective methods of software testing", Wiley Publishing, 3rd Edition.
6. S.A. Kelkar, Software Quality and Testing", PHI learning, Pvt Ltd.

Online Resources:

1. **https://onlinecourses.nptel.ac.in/noc26_cs59/preview**

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24304B	CO1	3	-	3	-	-	-	-	-	-	-	-
	CO2	3	-	3	-	2	-	-	-	-	-	-
	CO3	3	3	3	-	3	-	-	-	-	-	-
	CO4	3	-	-	2	3	-	-	2	-	-	-
	CO5	-	-	3	-	-	-	-	-	-	-	-
	CO6	3	-	-	-	-	-	-	-	-	-	-

Program Elective 1: Python for Data Analytics								
Course Code : IT24304C			**Course Credits: 04**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**2**	**-**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping:
1. Database Management Systems
2. Python for Data Science

Future Course Mapping:
1. Natural Language Processing

Course Objective:
1. To know the basis of python
2. To understand data analytics process
3. To learn data preparation and data visualization techniques
4. To learn predictive analysis

Course Outcomes: On completion of the course, learner will be able to
1. **Understand** python structures for data manipulation
2. **Analyze** data using panda's library
3. **Understand** data preparation techniques and transformers
4. **Apply** data visualization techniques
5. **Analyze** time series data and data analysis
6. **Construct** data models for predictive analytics

UNIT No.	Syllabus	Hrs
I	**Introduction to Python and NumPy** Introduction to Python and IDEs, Data types, **Data structure and Sequences:** Tuple, list and Dictionary, Built in sequence functions, **Functions:** Namespace, Scope and local functions, returning multiple values, Anonymous (lambda) functions, **Numpy ndarray:** A multidimensional array, data types for ndarrays, arithmetic with ndarrays, basic indexing and slicing, reshaping arrays, Transposing arrays and swapping axes, pseudorandom Number generator, linear algebra using numpy, Introduction to Data analytics	7
II	**Data Analysis using Pandas** Introduction to pandas, Pandas Series, DataFrame, Index Objects, Indexing, Arithmetic's, **Summarizing and computing descriptive statistics:** correlation and covariance, unique values, value count. Data Loading , **Storage and File formats:** Reading writing data from files, Excel file, JSON, HTML Web scrapping, database, interacting with web API.	7
III	**Data Preprocessing** Handling missing values, Data transformation: Removing duplicates, Transforming data using function or mapping, Discretization and binning, Detecting and filtering outliers, Computing indicator / Dummy variables, Categorical Data. Data wrangling: Hierarchical indexing , Combining and merging datasets.	7
IV	**Data Visualization** Introduction to matplotlib and seaborn library, Figures and subplots, color, marker and line styles, ticks, legend, saving plot to file, **Plotting with pandas and seaborn**: Line plots, Bar plots, Histogram, scatter plots.	6
V	**Data Aggregation and Time Series** Iterating over groups, Grouping with dictionaries and series, functions, Grouping by Index levels, Time series, Date ranges, frequencies, Time zone analysis, Resampling and frequency conversions. Data analysis using database.	6

VI	**Introduction to Machine Learning using scikit-learn** Introduction to machine learning, Types of machine learning, Classification, Regression, clustering. Supervised Learning with scikit-learn, Predictive modeling.	6
	Total Teaching Hours	**39**

List of Practical Assignments:

1. Write a program in python to create list, dictionary, ndim array and perform various operations such as indexing, slicing, reshaping, arithmetic, and linear algebra.
2. Write a program in python to create a dataframe for a suitable application and perform operations such as indexing , data summarization, statistics.
3. Write a program in python to perform web scraping for ecommerce website
4. Write a program in Python to read data from different sources such as CSV, Excel, JSON, HTML, and APIs, and store it into a DataFrame.
5. Write a program in python to read a dataset and perform different types of visualization(1D,2D, Multidimensional etc) using matplotlib /seaborn library.
6. Write a program in python to perform data aggregation.
7. Develop exploratory data analysis and predictive modeling applications using machine learning (Scikit-learn) including Linear Regression, Logistic Regression, and K-Means clustering, along with evaluation of model performance.

Text Books:

1. Wes McKinney , Python for Data Analysis by, O'Reilly ,978-1-098-10403-0
2. Python: Data Analytics and Visualization, Packt Publishing Ltd.,.ISBN: 978-1-78829-009-8
3. Fabio Nelli, Python Data Analytics, Apress.

Reference Books:

1. Ken Black, Business Statistics for Contemporary Decision Making. "John Wiley & Sons, Inc

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc25_cs104/course
2. https://www.coursera.org/learn/data-analysis-with-python
3. **https://www.youtube.com/watch?v=wUSDVGivd-8**

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24304C	CO1	2	1	3	1	2	-	1	-	1	-	1
	CO2	3	-	1	2	3	1	-	-	-	-	1
	CO3	1	-	2	2	2	-	-	-	-	-	1
	CO4	2	2	-	1	3	-	1	-	-	1	1
	CO5	2	2	-	2	2	1		1		1	1
	CO6	1	1	1	-	2	-	-	-	-	-	1

Advanced Java Programming								
Course Code : IT24305			**Course Credits: 2**			**Course type: VSEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
-	**2**	**1**	**-**	**-**	**-**	**40**	**-**	**30**

Prerequisite Course Mapping:
1. Core JAVA programming

Future Course Mapping:
1. Windows, Database, and Web application development.

Course Objective:
1. To understand the development of GUI application using Applet and AWT
2. To learn the relevant of AWT , Swing component and Event handling by developing GUI applications.
3. To learn accessing database through java program using JDBC
4. To learn remote method invocation using RMI.
5. To understand client server communication programs using java networking classes and Socket programming classes.
6. to understand the development of client/ Server communication applications using Servlet JSP and Hibernet.

Course Outcomes: On completion of the course, learner will be able to
1. **Understand** the development of GUI application using Applet.
2. **Identify** AWT, Swing component for development of windows applications using event handling.
3. **Develop** the JDBC applications.
4. **Understand** the remote method invocation using RMI.
5. **Design** a client server communication programs using java networking classes and Socket programming classes.
6. **Apply the Servlet, JSP and Hibernate concepts for the** development of web applications.

Syllabus

Sr. No	List of Tutorials	Hrs
1	Applet:Introduction to applet programming, Applet skeleton, Applet lifecycle. Applet HTML tag.	1
2	Event handling:Events, Event Sources, Event classes, Event Listeners, Handling mouse event, Keyboard event, Handling action event.	2
3	AWT classes: Label, Button, TextFeild, checkbox etc, Layout managers.	2
4	Swing: Swing classes	1
5	JDBC(Java Database Connectivity): JDBC architecture, SQL package and connection classes, adding JDBC connector jar file, CURD operations in JDBC.	1
6	RMI:RMI Architecture, RMI programming model, RMI package and classes, Developing distributed application using RMI	1
7	Socket Programming:Java.net package, Connection oriented transmission- stream socket classes, TCP socket programming application. Connectionless transmission using UDP.	1

8	Web Application Development:Servlet overview, Web Server, Life cycle of servlet, JSP, servlet application,	2
9	Hibernate, Struts.	2
Total Teaching Hours		**13**

List of Practical Assignments:

1. Write a program to demonstrate status of key on an applet window such as KeyPressed, KeyReleased, KeyUp, KeyDown.
2. Write a program to create a frame using AWT, Implement mouseClicked, mouse Entered and mouseExited events. Frame should visible when the mouse enters in frame window.
3. Develop GUI based AWT application to accept student marks of five subject to find total, percentile and result. Display result in separate window.
4. Develop GUI using swing to perform CURD operations using JDBC.
5. Develop RMI application to check string palindrome by defining remote method.
6. Write a java program to find remote machine IP address. Write a java program to perform client server communication using TCP socket programming.
7. Write servlet to display the username and password accepted from the client.
8. Write a Servlet JSP program to develop a registration form

Text Books:

1. Herbert Schildt, "Java: The complete reference", 9th edition, McGraw Hill Education (India) Pvt. Ltd.
2. Jim Keogh, "Complete Reference J2EE", McGraw Hill Education (India) Pvt. Ltd

Reference Books:

1. "Java Server Programming", Java EE6(j2EE 1.6) Black book, Dreamtech.
2. M. T Savaliya, "Advanced Java Technology", Dreamtech.
3. "Java 8 Programming Black Book" Black Book, Dreamtech.

Online Resources:

https://www.coursera.org/learn/advanced-java-certification-course

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24305	CO1	3	3	3	-	-	-	-	-	2	-	-
	CO2	2	2	2	-	2	-	-	-	2	-	-
	CO3	-	-	3	-	-	3	-	-	2	-	-
	CO4	-	-	3	3	-	3	-	-	2	-	-
	CO5	2	-	3	-	-	3	-	-	2	-	-
	CO6	-	-	3	-	-	3	-	-	2	-	-

SEMESTER-VI

Design and Analysis of Algorithms								
Course Code : IT24311			**Course Credits: 03**			**Course type: PCC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**-**	**-**	**10**	**30**	**60**	**-**	**-**	**-**

Prerequisite Course Mapping:
Data Structures, Basic Mathematics (Induction), Probability Theory

Future Course Mapping:
Distributed Computing , Advanced Algorithms

Course Objectives:
1. To understand problem-solving paradigms and classification of computational problems.
2. To analyze time and space complexity of algorithms using asymptotic notations.
3. To design efficient algorithms using various paradigms like Divide & Conquer, Greedy, and Dynamic Programming.
4. To apply algorithmic strategies to real-world problems.
5. To understand computational complexity theory including P, NP, and NP-completeness.

Course Outcomes:
1. Analyze computational complexity using asymptotic notations.
2. Apply Divide & Conquer as well as Greedy approach to design algorithms.
3. Apply Dynamic Programming to solve optimization problems.
4. Design and implement solutions using Backtracking techniques for combinatorial problems.
5. Evaluate and compare Branch and Bound strategies for solving problems.
6. Analyze and Classify P, NP, NP-complete, NP-Hard problems.

UNIT No.	Syllabus	Hrs
I	**Introduction:** Analysis of Algorithms, Efficiency- Analysis framework, asymptotic notations. Proof Techniques: Proof by induction, contradiction, direct proof, contraposition etc. Introduction to Brute Force method & Exhaustive search, Analysis of Non-recursive and recursive algorithms: Solving Recurrences.	7
II	**Divide and conquer method and Greedy strategy:** Divide & Conquer method: Merge sort, Quick sort. Binary search, Finding Minimum and Maximum, Large integer Multiplication. Greedy Method: MST for graph, Single-Source Shortest Paths: Dijkstra's Algorithm, Fractional Knapsack problem, Job Sequencing.	7
III	**Dynamic Programming:** General strategy, optimal substructure, 0/1 knapsack Problem, Bellman-Ford Algorithm, Multistage Graph problem, Optimal Binary Search Trees, Travelling Salesman Problem.	7
IV	**Backtracking:** General method, Recursive backtracking algorithm, Iterative backtracking method. 8-Queen problem, Sum of subsets, Graph coloring, Hamiltonian Cycle, 0/1 Knapsack Problem.	6
V	**Branch and bound:** The method, Control abstractions for Least Cost Search, Bounding, FIFO branch and bound, LC branch and bound, 0/1 Knapsack problem – LC branch and bound and FIFO branch and bound solution, Travelling Salesperson Problem (TSP).	6

VI	**Classes of algorithms:** Computational Complexity: Non-Deterministic algorithms, The classes: P, NP, NP-Complete, NP-Hard, Satisfiability problem, NP Complete Problems.	6
Total Teaching Hours		**39**

Text Books:

1. Thomas H Cormen and Charles E.L Leiserson, Introduction to Algorithms, PHI, ISBN: 81-203-2141-3.
2. S. Sridhar, Design and Analysis of Algorithms, Oxford, ISBN 10 : 0-19-809369-1.

Reference Books:

1. Horowitz and Sahani, Fundamentals of computer Algorithms, Galgotia, ISBN 81-7371-612-9.
2. R. C. T. Lee, SS Tseng, R C Chang, Y T Tsai, Introduction to Design and Analysis of Algorithms, A Strategic approach, Tata McGraw Hill, ISBN-13: 978-1-25-902582-2. ISBN-10: 1-25-902582-9.
3. Anany Levitin, Introduction to the Design & Analysis of Algorithm, Pearson, ISBN 81- 7758-835-4.
4. Gilles Brassard, Paul Bratle, Fundamentals of Algorithms, Pearson, ISBN 978-81-317-1244-3.

Online Resources:

https://onlinecourses.nptel.ac.in/noc20_cs27
https://onlinecourses.nptel.ac.in/noc19_cs47

Course Code	CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24311	CO1	3	3	-	2	-	3	-	-	-	-	1
	CO2	3	3	2	1	1	1	-	-	1	-	1
	CO3	3	3	3	1	1	1	-	-	1	-	1
	CO4	3	3	3	1	1	-	-	-	2	-	1
	CO5	3	3	3	2	1	1	-	-	1	-	-
	CO6	3	3	-	1	-	-	-	-	-	-	-

Computer Networks and Security								
Course Code : IT24312			**Course Credits: 03**			**Course Type: PCC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**ACTIVITY**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**2**	**–**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping:
1. Discrete Mathematics

Future Course Mapping:
1. Mobile Computing

Course Objectives:
1. To understand the fundamental concepts of networking standards, protocols and technologies.
2. To understand knowledge of data link layer architecture
3. To learn the basics of networking address and routing algorithms used in networks.
4. To learn different layer protocols in the protocol stacks
5. To understand modern network architectures with respect to design and performance

Course Outcomes:
1. Demonstrate the concepts of data communication at the physical layer and compare ISO - OSI model with TCP/IP model.
2. Elaborate the error control mechanism and medium access control sublayer of data link layer
3. Design the network using IP addressing and subnetting / supernetting schemes
4. Analyze transport layer protocols and congestion control algorithms.
5. Illustrate the role of the application layer with its protocol.
6. Describe the role of Artificial Intelligence in Networking.

UNIT No.	Syllabus	Hrs
I	**Introduction to Computer Network** Introduction: Goals and applications of networks, Categories of networks, Network structure and architecture (layering principles, services, protocols and standards), The OSI reference model, TCP/IP protocol suite, Network devices and components. Physical Layer: Network topology design, Types of connections, Transmission media, Line Coding-Manchester and Differential Manchester, Network performance, Switching techniques	7
II	**Data Link Layer and Medium Access Control Sublayer** Introduction and Design Issues, Flow and Error Control, Techniques for Error Detection and Correction, Parity Bits, Hamming Code(11/12-bits) and CRC. Simplex, Stop and Wait, Sliding Window Protocol. Random Access Protocols (ALOHA, CSMA, CSMA/CD, CSMA/CA). Introduction to Ethernet IEEE 802.3, IEEE 802.11 a/b/g/n, IEEE 802.15 Standards	7
III	**Network Layer** **Network Layer:** Point-to-point networks, IP Protocol: Classes of IP (Network Addressing), IPv4, IPv6, subnetting, Network Address Translation (NAT). Basic internetworking (CIDR, ARP, RARP, DHCP, ICMP,IGMP), Routing, forwarding and delivery, Static and dynamic routing, Routing algorithms and protocols, Congestion control algorithms. **Routing in MANET**: AODV, DSR, Mobile IP	7
IV	**Transport Layer**	

	Transport Layer: Process-to-process delivery, Transport layer protocols with Header format (UDP, TCP and SCTP), Connection management, Flow control and retransmission, Window management, TCP Congestion control, Quality of service. **Socket Programming:** TCP and UDP Connection	6
V	**Application Layer** HyperText Transport Protocol (HTTP), Cookies, Simple Mail Transport Protocol (SMTP),MIME, POP3, File Transfer Protocol (FTP). Introduction to various Types of Servers, Dynamic Host Configuration Protocol (DHCP): Header, Working, Domain Name Server (DNS): Working, Proxy Server: Need and Significance, working.	6
VI	**Security** Introduction, Security services, Need of Security, Key Principles of Security, Threats and Vulnerabilities, Types of Attacks, ITU-T X.800 Security Architecture for OSI, Security Policy and mechanisms, Operational Model of Network Security, Symmetric and Asymmetric Key Cryptography. Security in Network, Transport and Application: Introduction of IPSec, SSL, HTTPS, S/MIME, Overview of IDS and Firewalls.	6
	Total Teaching Hours	**39**

Text Books:

1. Behrouz Forouzan, "Data Communication and Networking", McGraw Hill
2. Andrew Tanenbaum "Computer Networks", Prentice Hall.

Reference Books:

1. William Stallings, "Data and Computer Communication", Pearson.
2. Kurose and Ross, "Computer Networking- A Top-Down Approach", Pearson.
3. Peterson and Davie, "Computer Networks: A Systems Approach", Morgan Kaufmann
4. W. A. Shay, "Understanding Communications and Networks", Cengage Learning.
5. D. Comer, "Computer Networks and Internets", Pearson.
6. Behrouz Forouzan, "TCP/IP Protocol Suite", McGraw Hill.

Practical Assignments

Conduct any 7 Assignments using C,C++,Java,Python, Wireshark, Packet Tracer, Juniper, Arista

1. Demonstrate the different types of topologies and types of transmission media by using a packet tracer tool.
2. Demonstrate transfer of a packet from LAN 1(wired LAN) to LAN2(wireless LAN) using switch or router and basic networking commands like ipconfig, netstat,trace, traceroute,lookup, ping.
3. Write a program for error detection and correction for 7/8 bits ASCII codes using Hamming Codes or CRC
4. Write a program to simulate Go Back N and Selective Repeat Modes of Sliding Window Protocol in Peer-to-Peer mode.
5. Write a program to demonstrate sub-netting and find subnetmask and analyze IP addressing using Wireshark tool.
6. Write a program to implement link state/ Distance vector/Krusskal's minimum spanning tree routing protocol to find a suitable path for transmission.
7. Use a packet tracer tool for configuration of 3 router network using one of the following protocol RIP/OSPF/BGP
8. Write a program using TCP socket for wired network for the following:
   a. Say Hello to Each other
   b. File Transfer

c. Calculator

9. Demonstrate configuration of DHCP Server and Firewall using packet tracer tool.
10. Demonstrate the SSL protocol by capturing the packets using Wireshark tool for any baking /commerce website

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24312	CO1	2	2	2	1	3	1	-	1	1	-	-
	CO2	3	2	3	2	2	1	-	1	1	-	1
	CO3	3	2	3	2	3	1	-	1	1	-	1
	CO4	3	2	3	2	3	1	-	1	1	-	-
	CO5	3	2	3	2	3	1	-	1	1	-	-
	CO6	2	2	3	2	1	1	2	1	1	-	1

Program Elective 2: Full Stack								
Course Code: IT24313A			**Course Credits: 04**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**ACTIVITY**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**02**	**–**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping:

1. C++, Java.

Future Course Mapping:

1. Computer Network, Distributed System.

Course Objectives:

1. To familiarize students with Web Programming basic concepts.
2. To learn and understand Web scripting languages.
3. To explore the Front end & Backend web programming skills.
4. To understand and learn Mobile web development.
5. To understand and learn Web application deployment.

Course Outcomes:

1. Understand & Develop Static and Dynamic website using technologies like HTML, CSS, Bootstrap.
2. Demonstrate the use of web scripting languages.
3. Understand the basics of Front End Technologies.
4. Develop web application with Front End & Back End Technologies.
5. Develop mobile website using JQuery Mobile.
6. Deploy web application on cloud using AWS.

Unit No.	Syllabus	Hrs
I	**INTRODUCTION TO WEB TECHNOLOGIES** **HTML:** Getting started with HTML, Why HTML, Tags and Elements, Attributes, Properties, Headings list, Links, Tables, Images, HTML Form, Media (Audio, Video), Semantic HTML5 Elements. **CSS:** Why CSS, Types of CSS, how to use CSS, Properties, Classes, Child-Class (Nested CSS), Colors, Text, Background, Border, Margin, Padding, Positioning (flex, grid, inline, block), Animation, Transition. **BOOTSTRAP:** Why Bootstrap, CSS over Bootstrap, how to Use Bootstrap, Bootstrap Grid System, Bootstrap Responsive, Bootstrap Classes, Bootstrap Components (i.e., Button, Table, List, etc.), Bootstrap as a Cross Platform. **W3C:** What is W3C, How W3C handles/Supports Web Technologies.	7
II	**WEB SCRIPTING LANGUAGES** **JavaScript:** Introduction to Scripting languages, Introduction to JavaScript (JS), JS Variables and Constants, JS Variable Scopes, JS Data Types, JS Functions, JS Array, JS Object, JS Events. **Advanced JavaScript:** JSON - JSON Create, Key-Value Pair, JSON Access, JSON Array, JS Arrow Functions, JS Callback Functions, JS Promises, JS Async-Await Functions, JS Error Handling. **AJAX:** Why AJAX, Call HTTP Methods Using AJAX, Data Sending, Data Receiving, AJAX Error Handling. **JQUERY :**Why JQuery, How to Use, DOM Manipulation with JQuery, Dynamic Content Change with JQuery, UI Design Using JQuery	7
III	**FRONT END TECHNOLOGIES** **Front-End Frameworks:** What is web framework? Why Web Framework? Web Framework Types. **MVC:** What is MVC, MVC Architecture, MVC in Practical, MVC in Web	6

	Frameworks. **TypeScript:** Introduction to TypeScript (TS), Variables and Constants, Modules in TS. **AngularVersion 10+:** Angular CLI, Angular Architecture, Angular Project Structure, Angular Lifecycle, Angular Modules, Angular Components, Angular Data Binding, Directives and Pipes, Angular Services and Dependency Injections (DI), Angular Routers, Angular Forms. **ReactJS:** Introduction to ReactJS, React Components, Inter Components Communication, Components Styling, Routing, Redux- Architecture, Hooks- Basic hooks, useState() hook, useEffect() hook useContext() hook.	
IV	**BACK END TECHNOLOGIES** **Node.JS:** Introduction to Node.JS, Environment Setup, Node.JS Events, Node.JS Functions, Node.JS Built- in Modules, File System, NPM, Install External Modules, Handling Data I/O in Node.JS, Create HTTP Server, Create Socket Server, Microservices- PM2. **ExpressJS:** Introduction to ExpressJS, Configure Routes, Template Engines, ExpressJS as Middleware, Serving Static Files, REST HTTP Method APIs, Applying Basic HTTP Authentication, Implement Session Authentication. **MongoDB:** NoSQL and MongoDB Basics, MongoDB-Node.JS Communication, CRUD Operations using Node.JS, Mongoose ODM for Middleware, Advanced MongoDB.	7
V	**MOBILE WEB DEVELOPMENT** **Mobile-First:** What is Mobile-First? What is Mobile Web? Understanding Mobile Devices and Desktop. **JQuery Mobile:** Introduction to the jQuery Mobile Framework, Set-up jQuery Mobile, Pages, Icons, Transitions, Layouts Widgets, Events, Forms, Themes, Formatting Lists, Header and Footer, CSS Classes, Data Attributes, Building a Simple Mobile Webpage.	6
VI	**WEB APPLICATION DEPLOYMENT** **Cloud:** AWS Cloud, AWS Elastic Compute, AWS Elastic Load Balancer and its types, AWS VPC and Component of VPC, AWS storage, Deploy Website or Web Application on AWS, Launch an Application with AWS Elastic Beanstalk.	6

List of Practical Assignments:

1. Create a responsive web page which shows the ecommerce/college/exam admin dashboard with sidebar and statistics in cards using HTML, CSS and Bootstrap.
2. Write a JavaScript Program to get the user registration data and push to array/local storage with AJAX POST method and data list in new page.
3. Create an Angular application which will do following actions: Register User, Login User, Show User Data on Profile Component.
4. Create a Node.js application that serves a static website and implements four CRUD APIs using Express.js and MongoDB.
5. Create a simple Mobile Website using jQuery Mobile.
6. Deploy/Host Your web application on AWS VPC or AWS Elastic Beanstalk.

Text Books:

1. Kogent Learning Solutions Inc, Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, XML and AJAX, Blackbook, Dreamtech Press, Second Edition, ISBN: 9788177228496.
2. Raymond Camden, Andy Matthews, JQuery Mobile Web Development Essentials, Packet Publishing, Second Edition, 9781782167891.

Reference Books:

1. Steven M. Schafer, "HTML, XHTML and CSS", Wiley India Edition, Fourth Edition,978- 81- 265-1635-3.
2. Steven M. Schafer, "HTML, XHTML and CSS", Wiley India Edition, Fourth Edition,978- 81-265-1635-3.
3. Ivan Bayross, "Web Enabled Commercial Application Development Using HTML, JavaScript,

DHTML and PHP, BPB Publications,4th Edition, ISBN:978-8183330084.
4. Adam Bretz& Colin J Ihrig, Full Stack Javascript Development with MEAN, SPD, First Edition, ISBN:978-0992461256.
5. JavaScript: The Definitive Guide - Master The World's Most-Used Programming Language, Seventh Edition.
6. Programming Typescript: Making Your JavaScript Applications Scale, Boris Cherny.

Online Resources:

NPTEL Course:

1. https://archive.nptel.ac.in/courses/106/106/106106222/
2. https://archive.nptel.ac.in/courses/106/106/106106156/

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
IT24313A	CO1	2	2	3	-	3	-	-	-	-	-	-	-
	CO2	2	2	2	-	2	-	-	-	1	-	-	-
	CO3	2	1	2	-	1	1	-	-	1	-	-	-
	CO4	2	2	3	1	3	-	-	-	-	1	1	1
	CO5	2	1	2	-	2	-	1	-	-	-	-	-
	CO6	3	2	3	1	3	-	-	1	1	1	1	1

Program Elective 2: Introduction to Artificial Intelligence								
Course Code : IT24313B			**Course Credits: 04**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
03	**02**	**–**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping: Data Structures

Future Course Mapping: Deep Learning, Computer Vision.

Course Objectives:

1. To understand Fundamental concepts of Artificial Intelligence
2. To study application of different search strategies for problem solving.
3. To study local search algorithms and constraint satisfaction problem
4. To understand Fundamentals of Game Theory.
5. To explore Various knowledge representations and reasoning schemes.
6. To understand uncertainty in AI and Bayesian Networks

Course Outcomes:

1. To **Study** the fundamental concepts of Artificial Intelligence
2. **Implement** smart system using different informed search / uninformed search or heuristic approaches
3. To **Understand** and formulate local search algorithms and Constraint satisfaction problems
4. **Application** of adversarial search techniques and **designing** of games.
5. To **Illustrate** knowledge reasoning and knowledge representation methods.
6. To **Analyze** Uncertainty in AI and **Study** Bayesian Networks

UNIT No.	Syllabus	Hrs
I	**Introduction:** Introduction to Artificial Intelligence, Foundations of Artificial Intelligence, History of Artificial Intelligence, State of the Art, Risks and Benefits of AI, Intelligent Agents, Types of Agents, Agents and Environments, Good Behaviour: Concept of Rationality, Nature of Environments, Structure of Agents.	6
II	**Problem Solving:** Search Strategies: Problem spaces (states, goals and operators), problem solving by search, **Uninformed search**: breadth-first, depth-first, depth limited search, depth first with iterative deepening, uniform cost search **Heuristic Search Techniques:** Generate-and-Test; Hill Climbing; Properties of A* algorithm, Best-first Search; Problem Reduction. Local Search algorithms.	7
III	**Local Search and Constraint Satisfaction** **Local Search:** Local Search Algorithms and Optimization Problems: Hill-climbing search, Simulated annealing, Local beam search, Genetic algorithms, Searching with Nondeterministic Actions: AND–OR search trees **Constraint Satisfaction Problems**: Defining Constraint Satisfaction Problems (CSP), Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs, Local Search for CSPs.	7
IV	**Adversarial Search and Games:** Game Theory, Optimal Decisions in Games, MiniMax, Heuristic Alpha–Beta Tree Search, Monte Carlo Tree Search, Stochastic Games, Partially Observable Games, Limitations of Game Search Algorithms,	7

V	**Knowledge Representation:** Logic, Propositional Logic, First Order Logic, Inference in First Order Logic: Unification, Forward Chaining, Backward Chaining, resolution, Introduction to NLP.	6
VI	**Uncertainty in AI and Bayesian Networks:** Basic Probability Notation, Inference Using Full Joint Distributions, Independence, Bayes' Rule And Its Use, Representing Knowledge In An Uncertain Domain, The Semantics of Bayesian Networks	

List of Practical Assignments:

1. Implement A star (A*) Algorithm for any game search problem.
2. Implement n-queens problem using Hill-climbing / simulated annealing
3. Write a program for the Information Retrieval System using appropriate NLP tools a. Text tokenization b. Count word frequency c. Remove stop words d. POS tagging.
4. Implement Greedy search algorithm for any of the following application:

a. Single-Source Shortest Path Problem
b. Job Scheduling Problem

5. Write a program for the Tic-Tac-Toe game.
6. Develop an elementary Chabot for any suitable customer interaction application.(Group Project)

Text Books :

1. Stuart Russel, Peter Norvig, "AI – A Modern Approach", Third Edition, Pearson Education, 2009
2. Elaine Rich, Kevin Knight and Shivashankar B Nair", Artificial Intelligence ", Tata McGraw Hill Edition 3rd Edition, 2009

Reference Books:

1. Algorithmic Game theory Edited by N Nishan, T Roughgarden; Cambridge University Press
2. Allen B. Downey, "Think Stats", Second Edition, O'Reilly Media, ISBN: 978-1-491-90733-7
3. D Fudenberg & J Tirole " Game Theory", MIT Press
4. K. Boyer, L. Stark, H. Bunke, "Applications of AI, Machine Vision and Robotics", World Scientific PubCo, 1995

Online Resources:

1. https://nptel.ac.in/courses/106102220
2. http://onlinestatbook.com/Online_Statistics_Education.pdf
3. https://london.ac.uk/sites/default/files/study-guides/introduction-tonatural-language-processing.pdf
4. https://www.deeplearningbook.org/contents/TOC.html
5. https://cvlesalfabegues.com/search/natural-language-understanding-2nd--dition/ww.ebookphp.com/data-structures-and-algorithms-professional-edition-beginners-guide-epubpdf/

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24313B	CO1	1	2	2	1	-	-	1	1	-	2	-
	CO2	1	2	2	2	2	1	-	1	1	2	-
	CO3	2	2	2	2	2	1	-	1	1	2	-
	CO4	2	2	2	2	2	1	-	1	1	2	-
	CO5	2	2	2	2	1	1	1	-	-	2	-
	CO6	2	2	2	1	-	-	-	-	-	2	-

Program Elective 2: Compiler Design								
Course Code : IT24313C			**Course Credits: 04**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
03	**02**	**–**	**10**	**30**	**60**	-	-	**30**

Prerequisite Course Mapping : Theory of Computation, Operating System

Course Objectives:

1. To Understand the language translation theories and compiler design stages
2. To Illustrate and Study various parser configurations
3. To Exemplify the use of syntax directed translation in intermediate code
4. To Understand Storage Management and Control Structure Environment.
5. To Learn to develop a Code generator
6. To Demonstrate the numerous machine independent optimization methods

Course Outcomes:

1. **Design** and **implement** a lexical analyzer using LEX tools
2. **Design** and **implement** a syntax analyzer using YACC tools
3. **Understand** syntax-directed translation and **Generate** intermediate codes
4. **Understand** run-time environment
5. **Construct** algorithms to produce computer code.
6. **Analyze** and **transform** programs to improve time and memory efficiency

UNIT No.	Content	Hrs
I	**Introduction:** Introduction to compiler, translators, interpreter, Structure of Compiler **Lexical Analyzer:** Role of lexical analyzer, specification of tokens, recognition of tokens, regular expression, finite automata, regular expression to finite automata transition diagrams, tool for lexical analyzer LEX	7
II	**Syntax Analysis:** Introduction to parsing techniques, Context free Grammar, bottom-up parsing and top down parsing. top down parsing , recursive descent parsing, predictive parsing ,bottom up parsing : operator precedence parsing, LR parsers, construction of SLR, canonical LR and LALR parsing tables, construction of SLR parse tables for ambiguous grammar, the parser generator tools – YACC, error recovery in top down and bottom up parsing.	7
III	**Syntax-Directed Translation:** Syntax-Directed Definitions, Evaluation Orders for SDD's **Intermediate-Code Generation:** Variants of Syntax Trees, Three-Address Code,Types and Declarations, Type Checking, Control Flow, Intermediate Code for Procedures.	7
IV	**Run-Time Environments:** Storage Organization, Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection .	6
V	**Code Generation:** Issues in the Design of a Code Generator ,The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization, Register Allocation and Assignment, Dynamic Programming Code-Generation.	6
VI	**Code Optimization:** The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, Loops in Flow Graphs.	6

List of Practical Assignments:

1. Design and implement a lexical analyzer for given language using C and the lexical analyzer should ignore redundant spaces, tabs and newlines.
2. Implement Lexical Analyzer using Lex Tool.
3. Implement Calculator using LEX and YACC.
4. Write a program for construction of DFA for given Regular Expression.
5. Implement the front end of a compiler that generates the three address code for a simple language. (Group Project).
6. Implement Local and Global Code Optimizations such as Common Sub-expression Elimination, Copy Propagation, Dead-Code Elimination, Loop and Basic-Block Optimizations. (Group Project).

Text Books :

1. V. Aho, R. Sethi, J. D. Ullman, “Compilers: Principles, Techniques, and Tools", Pearson Edition, ISBN 81-7758-590-8.

Reference Books:

1. Dick Grune, Bal, Jacobs, Langendoen, “ Modern Compiler Design”, Wiley, ISBN 81-265-0418-8.
2. Anthony J. Dos Reis, “Compiler Construction Using Java, JavaCC and Yacc” Wiley, ISBN 978-0-470-94959-7.
3. K Muneeswaran, “Compiler Design", Oxford University press, ISBN 0-19-806664-3 3. J R Levin.
4. T Mason, D Brown, “Lex and Yacc", O'Reilly, 2000 ISBN 81-7366-061-X.

Online Resources:

1. NPTEL Compiler Design Course : https://archive.nptel.ac.in/courses/106/105/106105190/

Course Code	Course Outcome	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24313C	CO1	2	2	2	2	2	-	-	-	1	1	-
	CO2	2	2	2	2	1	-	-	-	1	1	-
	CO3	1	2	2	2	1	-	-	-	1	1	-
	CO4	2	1	1	1	1	-	-	-	1	1	-
	CO5	1	2	2	2	1	-	-	-	1	1	1
	CO6	2	2	2	2	1	-	-	-	1	1	1

Program Elective 3: Introduction to Internet of Things								
Course Code : IT24314A			**Course Credits: 4**			**Course type: PCE**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
03	**02**	**-**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping:

1) Basics of Computer Network
2) Processor Architecture

Future Course Mapping:

Robotics, Sensor Programming and IOT Applications

Course Objectives:

1. To provide an understanding of the foundational concepts, frameworks, and architecture of IoT, including its enabling technologies and deployment models.
2. To explore the role of sensors, actuators, and communication protocols in IoT, focusing on the integration of smart objects into IoT systems
3. To provide hands-on experience with IoT platforms like Arduino, NodeMCU, and Raspberry Pi, and enable students to program and interface sensors and actuators.
4. To understand and evaluate various communication protocols used in IoT, including both non-IP and IP-based protocols, and the challenges related to IoT deployment.
5. To explore the integration of cloud computing in IoT systems and address security concerns in IoT environments.
6. To examine and implement IoT applications across various domains such as smart homes, healthcare, agriculture, and industry, showcasing the versatility of IoT technologies.

Course Outcomes:

On completion of the course, students will be able to:

1. Demonstrate a comprehensive understanding of IoT definitions, characteristics, frameworks, architecture, and enabling technologies, and apply this knowledge to explain IoT deployment levels.
2. Identify and describe the components of smart objects, including sensors, actuators, and communication protocols, and explain how they connect and operate within an IoT system.
3. Demonstrate proficiency in programming and interfacing IoT platforms such as Arduino, NodeMCU, and Raspberry Pi to control sensors and actuators for practical IoT applications
4. Explain and compare communication protocols in IoT systems (such as MQTT, IPv6, ZigBee) and identify the challenges related to wireless communication, node discovery, and routing in IoT networks.
5. Understand the role of cloud services in IoT, implement cloud-based IoT applications, and identify and address key security challenges in IoT systems
6. Design and implement domain-specific IoT applications, demonstrating the practical benefits of IoT in improving automation, monitoring, and efficiency in various sectors

UNIT No.	Syllabus	Hrs
I	**Introduction to Internet of Things** Definition and Characteristics of IoT , Working Definition, IoT Frameworks, Basic Nodal Capabilities, IoT Architecture, IoT Devices, Physical Design of IoT: IoT Protocols, Logical Design of IoT: Functional block, communication Model, Communication API's, IoT Enabling Technologies: WSN, cloud computing, Big data Analytics, communication Protocols, Embedded systems, IoT levels and Deployment templates: Level 1 to Level 5.	7
II	**Things in IoT** Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects:	6

	Communications Criteria, IoT Access Technologies: IEEE 802.15.4, IEEE 802.15.4g and 802.15.4e, IEEE 1901.2a, LoRaWAN, Electronic Communication Protocols (Device Interfacing) Protocols: I2C, SPI, UART, USRT, CAN.	
III	**IOT Platforms and its programming** **Introduction to Arduino, NODEMCU and Raspberry Pi :** Installation, Introduction to Python program with Raspberry Pi with focus on interfacing external gadgets (Bluetooth Speaker, Camera), controlling output, and reading input from pins. Introduction to Arduino Programming, Basics of NODEMCU interfacing with sensors, actuators and cloud, Integration of Sensors and Actuators with Arduino.	6
IV	**Communication Protocols and IOT Challenges** Introduction to Non-IP Based Protocol (BlueTooth, ZigBee), IP Based Protocol (IPV4, IPV6, 6LoWPAN), Application Layer Protocols (MQTT, AMQP) Wireless medium access issues, MAC protocol, routing protocols, Sensor deployment & Node discovery, Data aggregation & dissemination.	6
V	**Cloud Offerings and IOT Security** **Introduction to Cloud Storage models (SaaS, Paas, IaaS) and communication APIs** **Web server –** Web server for IoT, Cloud for IoT (Thing Speak), Python web application framework, designing a RESTful web API. **IoT Security**: Vulnerabilities of IoT, Security Requirements, Challenges for Secure IoT, Threat Modelling, Key elements of IoT Security: Identity establishment, Access control, Data and message security, Non repudiation and availability, Security model for IoT.	7
VI	**IOT Applications** Home Automation - Smart Appliances, Intrusion Detection, Smoke/Gas Detector, Smart City –Smart Parking, Smart Road, Structural Health Monitoring, Surveillance applications, Health - Fitness and Health Monitoring, Wearable Electronics, Agriculture - Smart Irrigation, Greenhouse Control, Environment -Weather Monitoring, Noise Pollution Monitoring, Logistic - Root Generation and Scheduling, Shipment Monitoring, Retail Management - Inventory Management, Smart Payments, Industry Applications - Machine Diagnosis and Prognosis, Indoor Air Quality Monitoring.	7

List of Practical Assignments:

1. Create a simple traffic light system using 3 LEDs (Red, Yellow, Green) controlled by a timer
2. Use an ultrasonic sensor to measure the distance of an object and display the result on the serial monitor
3. Create a system that regulates the room temperature by turning on/off a fan or heater based on sensor input.
4. Use IR sensor to detect the obstacle and control the LED based on Input received (LED should be ON if obstacle found, display the status on serial monitor).
5. Use a gas sensor (MQ-2) to detect gas leaks and trigger an alarm.
6. Build an automated plant watering system using a soil moisture sensor, a relay, and a water pump.
7. Implement a door lock system using a servo motor and an RFID reader to unlock doors upon identification.
8. Use a PIR motion sensor to detect movement and trigger an alarm
9. Control an LED using a web interface hosted on the NodeMCU over Wi-Fi. This introduces basic web server functionality with NodeMCU
10. Use a DHT11 temperature sensor with NodeMCU to read temperature and humidity values and display them on a simple web page hosted on the NodeMCU.

Text Books :

1. Vijay Madisetti, ArshdeepBahga, “Internet of Things: A Hands-On Approach” , 2014, Universities Press(India) Pvt Ltd., ISBN: 9788173719547
2. Matt Richardson & Shawn Wallac, “Getting Started with Raspberry Pi”, 2014, O'Reilly (SPD), ISBN:9789350239759
3. Pethuru Raj and Anupama C Raman, “The Internet of Things: Enabling Technologies, Platforms and Use Cases”, 2017, CRC Press, ISBN: 13:978-1-4987-6128-4.
4. Rushi Gajjar, “Raspberry Pi Sensors”, 2015, Packt Publishing, ISBN : 978-1-78439-361-8
5. Robert H. Bishop, “The Mechatronics Handbook”, 2002, CRC Press , ISBN: 0-8493-0066-5/02

Reference Books:

1. Peter Waher, “Learning Internet of Things”, 2015, Packt Publishing, ISBN: 978-1-78355-353-2
2. Peter Friess, “Internet of Things – From Research and Innovation to Market Deployment”, 2014, River Publishers, ISBN: 978-87-93102-94-1
3. Waltenegus Dargie, Christian Poellabauer, "Fundamentals of Wireless Sensor Networks: Theory and Practice", 2010, Wiley Publication, ISBN: 978-0-470-99765-9
4. Simon Monk, “Raspberry Pi Cookbook, Software and Hardware Problems and solutions”, 2019, O'Reilly, ISBN 9781492043225

Online Resources:

1. Introduction to Arduino and its Setup: https://www.arduino.cc/en/software
2. Introduction to Raspberry Pi and its OS (Raspbian Lit): https://www.raspberrypi.org/software/operating-systems/
3. Cloud for IoT– ThingSpeak: https://thingspeak.com/
4. Cloud for IoT - Ubidots: https://ubidots.com/stem/

Overall IoT Course Contents: https://onlinecourses.nptel.ac.in/noc21_cs17/preview

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24314A	CO1	3	2	-	2	2	-	1	-	-	2	-
	CO2	3	3	-	2	2	-	1	-	1	2	-
	CO3	2	2	3	2	3	1	-	1	2	2	-
	CO4	2	3	-	3	3	1	-	-	1	2	-
	CO5	2	3	-	2	3	1	1	-	1	2	-
	CO6	2	2	3	2	3	1	1	1	2	2	1

Program Elective 3: Cyber Security								
Course Code: IT24314B			**Course Credits: 04**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**2**	**-**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping:
1.Basics of Computer Network

Future Course Mapping:
1. Quantum Cryptography

Course Objective:
1. To understand the basic approaches in cyber security and Information Security.
2. To analyze the cryptography and networking terms.
3. To understand about different types of hacking and cyber-crimes.
4. Analyze and secure web applications using secure coding and penetration testing practices.
5. Apply digital forensics techniques and incident response methodologies.
6. Explore emerging trends such as Cloud Security, IoT Security, and AI in Cyber Security.

Course Outcomes: On completion of the course, students will be able to
1. Understand core concepts of Information Security, CIA triad, threats, attacks, and cryptographic mechanisms.
2. Analyze network models, protocols, devices, and implement basic cryptographic techniques.
3. Identify cyber-crimes and evaluate ethical hacking methodologies and attack techniques.
4. Detect and mitigate web application vulnerabilities using secure coding and penetration testing practices.
5. Apply digital forensics techniques and incident response methodologies to investigate cyber incidents.
6. Evaluate advanced security challenges in Cloud, IoT, Mobile, and AI-based cybersecurity systems.

Unit No.	Syllabus	Hrs
I	**Introduction to Information Security & Cyber Security** Confidentiality, Integrity & Availability (CIA), The challenges of Security, Threats, Attacks, Operational Model of Security, Cryptography, Symmetric and Asymmetric Cryptography, Brute-Force Attack. **Cyber Security Basics:** Introduction to Cyber Security, Need, importance and challenges in Cyber Security, Cyberspace, Cyber threats, Cyber-warfare, Cyber Terrorism.	7
II	**Network Fundamentals & Cryptography Basics** OSI, TCP/IP Model, Types of Networks, Wi-Fi, Network Protocols (TCP/IP, DNS, HTTP), Network Devices (Routers, Switches, Firewalls), Proxy Servers, VPN, Network Sniffing, **Cryptography Basics:** Encryption, Decryption, Types of Encryptions (Symmetric Cipher-DES, Asymmetric Cipher-RSA, Diffie Hellman, Cryptographic Hash functions and Message Authentication codes)	7
III	**Cyber Crimes & Ethical Hacking** Cyber stalking, forgery, software piracy, cyber terrorism, phishing, computer hacking, creating and distributing viruses over the internet, spamming, cross site scripting, cybersquatting, logic bombs, web jacking, internet time thefts, DoS attack, Man in the middle attack, salami attack, data diddling, email spoofing. Ethical Hacking, Types of Hacking	7

IV	**Web Application Security** Web Architecture & HTTP-Client -Server Architecture, HTTP methods, Cookies & sessions, SQL Injection, Authentical flaws, Secure Coding Practices, Input Validation, Secure Session management, Common Coding Mistakes, Penetration testing.	6
V	**Digital Forensics & Incident Response** Introduction, Forensics lifecycle, Evidence handling, Chain of Custody, Disk & file System Forensics, Deleted file recovery, file signature, Memory & Malware Forensics- RAM Forensics, Network Forensics-Log analysis, Traffic reconstruction.	6
VI	**Advanced Topics, Trends & Capstone** Cloud Security- Cloud Service Models, Cloud Threads. IOT & Mobile Security- IOT attack Surface, Mobile OS Security. AI in Cyber Security- AI based attacks, Machine Learning for intrusion detection	6
Total Teaching Hours		**39**

List of Practical Assignments:

1. Perform port scanning with nmap.
2. Install Wireshark for monitoring Network Communication with sniffers.
3. Install Jscript/Cryptool tool (or any other equivalent) and demonstrate Asymmetric, Symmetric crypto algorithm, Hash and Digital/PK! signatures.
4. Use Open SSL command to generate min 10 passwords of length 12 characters.
5. Perform email analysis using the Autopsy tool.
6. Identify the vulnerabilities in IoT device.

Text Books:

1. Computer and Information Security Handbook – John R. Vacca
   Comprehensive coverage of information security, risk management, forensics, and network security fundamentals.
2. Cryptography and Network Security: Principles and Practice – William Stallings
   Classical and widely used text for cryptography fundamentals.
3. Cyber Security and Digital Forensics – Mangesh M. Ghonge.
4. Hacking: The Art of Exploitation – Jon "Smibbs" Erickson
   Great for understanding hands-on exploitation techniques and low-level security concepts.

Reference Books:

1. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC Press.
2. Introduction to Cyber Security, Chwan-Hwa(john) Wu,J. David Irwin, CRC Press T&F Group.
3. Handbook of Digital Forensics and Investigation, E. Casey, Academic Press, 2010.
4. Applied Cryptography: Protocols, Algorithms, and Source Code in C – Bruce Schneier
   Gold standard for cryptography in practice.
5. Computer Forensics and Cyber Crime: An Introduction (e.g., by Marjie T. Britz or others)

Online Resources:

1. https://www.nptelprep.in/courses/106105162/materials
2. https://onlinecourses.nptel.ac.in/noc26_lw01/preview
3. https://onlinecourses-archive.nptel.ac.in/noc17_cs36/course
4. https://onlinecourses.nptel.ac.in/noc25_cs16/preview

Course Code	Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
IT24314B	CO1	3	2	-	-	1	-	-	-	-	-	-
	CO2	3	3	1	2	2	-	-	-	-	-	-
	CO3	2	3	1	2	2	2	-	3	-	-	-
	CO4	2	3	3	2	3	-	-	2	-	-	-
	CO5	2	3	2	3	3	2	-	2	-	-	-
	CO6	2	3	2	2	2	3	2	2	-	-	-

Program Elective 3: Computer Vision								
Course Code : IT24314C			**Course Credits: 4**			**Course type: PEC**		
Teaching Scheme			**Evaluation Scheme**					
TH	**PR**	**TUT**	**CAA**	**ISE**	**ESE**	**TW**	**PR**	**OR**
3	**2**	**-**	**10**	**30**	**60**	**-**	**-**	**30**

Prerequisite Course Mapping:

1. Vectors, Linear algebra
2. Python, Matlab

Future Course Mapping:

1. Robotics
2. Applied AI

Course Objectives:

1. To review image processing techniques for computer vision
2. To understand basic image intensity transforms, shape and region analysis.
3. To understand Feature extraction techniques.
4. To understand segmentation techniques.
5. To understand Motion estimation and Object detection as well as pattern recognition
6. To study some applications of computer vision algorithms.

Course Outcomes:

On completion of the course, students will be able to:

1. **Understand** mathematical modeling methods for low, intermediate and high- level image processing tasks.
2. **Apply** and **Evaluate** basic image enhancement and filtering techniques.
3. **Implement and Analyze** feature detection, extraction and matching techniques.
4. **Apply** image segmentation and deep learning based methods.
5. **Demonstrate** understanding of motion analysis and 3D vision techniques.
6. **Design and Develop computer** vision applications to solve a computer vision problem.

UNIT No.	Syllabus	Hrs
I	**Fundamentals of digital image processing** What is Computer vision? Image Formation: Geometric primitives and transformations, Photometric image formation, the digital camera, Image Processing: Introduction, Applications and examples of digital image processing, Fundamental steps in digital image processing, components of digital image processing system, Digital image, basic relationship between pixels. Binocular Stereopsis: Camera calibration and Epipolar Geometry, Homography, Rectification.	7
II	**Intensity transformation, Spatial Filtering and Shape** Basic intensity transformation functions, Histogram equalization, Histogram matching, Local Histogram Processing, Using histogram statistics for image enhancement, Contrast Limited Adaptive Histogram Equalization (CLAHE), Spatial filtering, smoothing and sharpening spatial filters. Color Image Processing: Color models, Shape and Regions: Object labelling and counting.	7
III	**Feature Detection and Matching** Points and Patches: Feature detectors, Feature descriptors, Feature matching, Edges - Canny, LOG, DOG; Line detectors (Hough Transform), Corners - Harris corner detection, Orientation Histogram, SIFT, SURF, HOG, ORB, ScaleSpace Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters, Image Matching, Principal Component Analysis (PCA)	7

IV	**Image Segmentation:** Introduction, Segmentation techniques: Region Growing, Edge Based approaches to segmentation, Texture Segmentation, Object detection, Object segmentation, Active contours Deep learning models for segmentation, Metrics: IoU, mAP, precision/recall**,** Dice**,** High-level vision, Deep learning for object detection, Transfer learning for CV models	6
V	**Motion Analysis:** Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo, Motion parameter estimation. **Shape from shading:** Light at Surfaces, Reflectance Map, Albedo estimation. Photometric Stereo, Use of Surface Smoothness Constraint	6
VI	**Applications of Image Processing and Computer vision:** Real-world applications of Computer Vision for various industries: Manufacturing, Healthcare, Agriculture, Transportation, Sports using Gesture Recognition, Motion Estimation and Object Tracking, Face Recognition, Facial Expression Recognition, Optical Character Recognition, Automated Video Surveillance.	6

List of Practical Assignments:

Tools: Python, OpenCV, TensorFlow / PyTorch

1) Implement various grey level transformations (Log, gamma, inversion).
2) Apply Histogram Equalization and CLAHE on grayscale and color images.
3) Perform smoothing and sharpening using spatial filters.
4) Implement SIFT, ORB and Histogram of Oriented Gradient (HOG) feature descriptors and perform key point matching between two images using SIFT/ORB.
5) Implement region growing and edge-based segmentation technique
6) Apply a pre-trained U-Net / Mask R-CNN for semantic segmentation using Tensorflow or Pytorh
7) Implement object detection using Haar cascades or YOLO
8) Implement background subtraction for detecting moving objects in a video.
9) Track moving objects using Kalman filter or DeepSORT.
10) Implement Principal Component Analysis to reduce the dimensionality.
11) Mini Project: Develop a computer vision application in a group of 2 - 3 students.

Text Books :

1. Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer
2. Digital Image Processing- Refael C. Gonzalez and Richard E. Woods, Wesley

Reference Books:

1. R. Davies, “Computer & Machine Vision”, Fourth Edition, Academic Press, 2012.
2. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, 2nd Edition, Cambridge University Press, March 2004
3. Computer Vision - A modern approach, by D. Forsyth and J. Ponce, Prentice Hall Robot Vision, by B. K. P. Horn, McGraw-Hill.

Online Resources:

1) Modern Computer Vision by Prof. A.N. Rajagopalan, IIT Madras
2) https://onlinecourses.nptel.ac.in/noc25_ee51/preview
3) Computer Vision And Image Processing - Fundamentals And Applications By Prof. M. K. Bhuyan | IIT Guwahati https://onlinecourses.nptel.ac.in/noc25_ee13/preview
4) https://www.cse.iitd.ac.in/~suban/vision/index.html
5) https://www.sci.utah.edu/~gerig/CS6320-S2015/CS6320_3D_Computer_Vision.html
6) https://www.cse.iitb.ac.in/~ajitvr/CS763_Spring2017/
7) https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic3.htm

Course Code	Course Outcome	PO1	PO 2	PO3	PO4	PO5	PO6	PO7	PO 8	PO9	PO10	PO11
IT24314C	CO1	3	2	-	2	1	-	-	-	-	-	-
	CO2	3	2	2	2	2	-	-	-	-	-	-
	CO3	3	3	2	2	2	-	-	-	-	-	-
	CO4	3	3	3	2	3	-	-	1	-	-	-
	CO5	3	3	2	2	2	-	-	1	-	-	-
	CO6	3	2	3	2	3	-	-	1	2	2	2