Autonomous Program Structure of
Third Year B. Tech. Fifth Semester
(Information Technology)
Academic Year: 2023-2024 Onwards

Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Total Marks	Credit
		Lecture	Tutorial	Practical	In Sem	End Sem	Oral	Practical		
20IT 501	Theory of Computation	3	1	0	50	50	0	0	100	4
20IT 502	Design and Analysis of Algorithms	3	1	0	50	50	0	0	100	4
20IT 503	Machine Learning	3	1	0	50	50	0	0	100	4
20PEIT 501	Programme Elective-I	3	0	0	50	50	0	0	100	3
20PEIT 502	Programme Elective-II*	3	0	0	50	50	0	0	100	3
20OEHS 501	Open HS Elective –I	3	0	0	50	50	0	0	100	3
20IT 502L	Design and Analysis of Algorithms Lab	0	0	2	25	0	0	25	50	1
20IT 503L	Machine Learning Lab	0	0	2	25	0	25	0	50	1
20PEIT 501L	Programme Elective Lab- I	0	0	2	25	0	25	0	50	1
	Total	**18**	**3**	**6**	**375**	**300**	**50**	**25**	**750**	**24**
	Grand Total	**27**			**750**					

***NPTEL / Swayam Course**

Programme Elective – I 20PEIT 501 A Artificial Intelligence 20PEIT 501 B Business Intelligence 20PEIT 501 C Computer Graphics and Animation	**Programme Elective – I Lab** 20PEIT 501L A Artificial Intelligence 20PEIT 501L B Business Intelligence 20PEIT 501L C Computer Graphics and Animation
Programme Elective – II 20PEIT 502 A Blockchain Architecture Design and Use Cases 20PEIT 502 B Internet of Things	

APPROVED BY
Secretary Academic Council
MKSSS's Cummins College of Engineering
For Women, Pune-411052

APPROVED BY
Chairman Academic Council
MKSSS's Cummins College of Engineering
For Women, Pune-411052

20OEHS 501 Open Elective I (Humanities)

Sr. No.	Course Code	Course Title
1	20OEHS501A	Entrepreneurship Development
2	20OEHS501B	Intellectual Property Rights
3	20OEHS501C	Introduction to Digital Marketing
4	20OEHS501D	Law for Engineers
5	20OEHS501E	Organizational Behaviour
6	20OEHS501F	Project Management

20IT 501 Theory of Computation

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: 1 hour

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 4

Prerequisites: Discrete mathematics

Course Objectives:
Familiarize students with

1. Abstract computing models.
2. Types and applications of formal grammar.
3. Push Down Automata, Turing Machine for engineering problems
4. Application of Theory of Computation in System Programming.

Course Outcomes:
Students should be able to

1. Construct abstract computing models for engineering problems
2. Apply the concepts of formal grammars for formal language representation
3. Construct Push Down Automata, Turing Machine for engineering problems
4. Evaluate capabilities of computing models

Unit – I Fundamentals **7 Hours**

Strings, Alphabet, Language, Operations, Finite state machine, definitions, finite automaton model, acceptance of strings, and languages, deterministic finite automaton and non-deterministic finite automaton, transition diagrams and Language recognizers. NFA to DFA conversion

Unit – II Finite Automata with application **7 Hours**

NFA with ϵ transitions - Significance, acceptance of languages, Equivalence between NFA with and without ϵ transitions, minimization of FSM, equivalence between two FSM's, Finite Automata with output- Moore and Mealy machines.
Lexical analyzer as an application of Finite Automaton. Introduction to Lex tool

Unit – III Regular Expression and Grammar Formalism **7 Hours**

Regular expressions: Identity rules, Constructing finite Automata for a given regular expression, Conversion of Finite Automata to Regular expressions. Pumping lemma of regular sets, closure properties of regular sets
Introduction to Grammar: derivation trees, sentential forms. Right most and leftmost derivation of strings, Chomsky hierarchy

Unit – IV Regular Grammar with application **7 Hours**

Regular grammars-right linear and left linear grammars, equivalence between regular grammar and FA, inter conversion, Parsing techniques, Top-down parsing, Bottom-up parsing **Recursive descent parser** as an application of Regular Grammar. . Introduction to YACC tool

Unit – V Context free grammars and Push down automata 7 Hours

Context Free Grammars-Ambiguity in context free grammars. Minimization of Context Free Grammars. Normal Forms Chomsky Normal Form, Greibach Normal Form, conversion to CNF and GNF

Push down automata- definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, inter conversion,

Unit – VI Turing Machine 7 Hours

Turing Machine, definition, model, design of TM, Computable functions, recursively enumerable languages. Church's hypothesis, counter machine, types of Turing machines, Universal Turing Machine, decidability/undecidability of problems, Halting problem Correspondence problem, Turing reducibility

Modularized programming concept as an application of Turing machines

Text Books

1. Daniel I.A. Cohen, "Introduction to Computer Theory" Wiley-India, ISBN: 978-81-265-1334-5
2. Vivek Kulkarni, "Theory of Computation", Oxford University Press, ISBN-13: 978-0-19-808458-7.

Reference Books

1. John C. Martin, "Introduction to language and theory of computation", Tata McGraw Hill, Third edition, ISBN 0-07-049939-X
2. Hopcroft Ulman, "Introduction To Automata Theory, Languages And Computations", Pearson Education Asia, 2nd Edition
3. E V Krishnamurthy, "Introduction to Theory of Computer Science", EWP Second 2nd Edition.
4. Michael Sipser, Introduction to the Theory of Computation, esruoC Technology Inc; 3rd edition, 8877897781-879

20IT 502 Design and Analysis of Algorithms

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: 1 hour

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 4

Prerequisites: Data structures

Course Objectives:
Familiarize students with
1. Algorithmic approaches for problem solving
2. Basics of computational complexity analysis
3. Various algorithm design strategies.
4. Different classes and solutions to problems such as P, NP etc.

Course Outcomes:
Students should be able to
1. Apply algorithmic techniques to solve computational problems.
2. Analyze the time and space complexity of algorithms for performance evaluation.
3. Solve optimization problems using dynamic programming, backtracking, and branch-and-bound algorithmic techniques.
4. Perceive the given problem solution from computational complexity classes point of view and evaluate the applicability of parallel techniques

Unit – I: Introduction **7 Hours**

Analysis of Algorithm, Efficiency- Analysis framework, asymptotic notations. Proof Techniques: Proof by induction, contradiction, direct proof, contraposition and so on, Introduction to Brute Force method & Exhaustive search, Analysis of Non-recursive and recursive algorithms: Solving Recurrences

Unit – II: Divide and conquer method and Greedy strategy **7 Hours**

Divide & Conquer method: Merge sort, Quick Sort. Binary search, Finding Max-Min, Large integer Multiplication, TOH. Greedy Method: MST for graph, Single-Source Shortest Paths: Dijkstra's Algorithm, Fractional Knapsack problem, Job Sequencing.

Unit – III: Dynamic Programming **7 Hours**

General strategy, optimal substructure, 0/1 knapsack Problem, Chain matrix multiplication, Bellman-Ford Algorithm, Multistage Graph problem, Optimal Binary Search Trees, Travelling Salesman Problem.

Unit – IV: Backtracking **7Hours**

General method, Recursive backtracking algorithm, Iterative backtracking method. 8-Queen problem, Sum of subsets, Graph coloring, Hamiltonian Cycle, 0/1 Knapsack Problem.

Unit – V: Branch and bound **7 Hours**

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, Traveling sales person problem

Unit – VI: Classes of algorithms **7 Hours**

Computational Complexity: Non Deterministic algorithms, The classes: P, NP, NP Complete, NP Hard, Satisfiability problem, NP Complete Problems, Parallel Algorithms, Optimizing Parallel Algorithms, Randomized and approximation algorithms, GProf

Text Books:

1. Thomas H Cormen and Charles E.L Leiserson, Introduction to Algorithm, 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.

20IT 503 Machine Learning

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: 1 hour/week

Examination Scheme:
In-Semester: 50 Marks
End-Semester: 50 marks
Credits: 4

Prerequisites: Linear Algebra, Probability Basics

Course Objectives:

Familiarize students with

1. Concept of dataset
2. Applications of Machine Learning
3. Machine Learning algorithms
4. Evaluation metrics applicable to Machine Learning techniques

Course Outcomes:

Students will be able to:

1. Choose appropriate Machine Learning technique for solving real-world problems
2. Explain wide variety of Machine Learning algorithms and techniques
3. Solve problems using Machine Learning algorithms
4. Evaluate various Machine Learning models

Unit – I Introduction to Machine Learning 6 Hours

Introduction: What is Machine Learning, Examples of Machine Learning applications, Training versus Testing, Positive and Negative Class, Cross validation
Types of Learning: Supervised, Unsupervised and Semi-Supervised Learning, Reinforcement Learning, Incremental Learning
Dataset: Preparing dataset for Machine Learning

Unit – II Linear Models: Classification 7 Hours

Binary and Multi-class Classification: Concept, evaluating Classification models using Contingency Table/Confusion Matrix
Perceptron: Neurons, learning rate, threshold
Support Vector Machine: Hard margin, soft margin, kernel trick for non-linear data

Unit – III Linear Models: Regression 8 Hours

Regression: Concept, evaluating Regression models
Univariate Regression: Linear model, constructing line of Regression
Polynomial Curve Fitting: Test-Train Curves, degree of polynomial
Theory of Generalization: Overfitting and Underfitting, Bias-Variance Dilemma, Regularization

Unit – IV Distance based Models 8 Hours

Distance based Models: Concept, Distance Measures: Euclidian, Manhattan, Minkowski, Hamming, Chebyshev.
Distance based Classification: Neighbors, exemplers, Nearest Neighbor algorithm
Distance based Clustering: Evaluating clustering algorithms, k-means, soft k-means, DBScan, Hierarchical Clustering

Unit – V Rule based and Tree based Models 8 Hours

Rule based Models: Frequent itemsets, confidence and support, Association Rule mining

Tree based Models: Building Decision Tree using impurity measures: Gini Index, Minority Class Index, Entropy. ID3.

Unit – VI Probabilistic Models 5 Hours

Generative and Discriminative Models, Bayes' Theorem, Independence assumption, Naïve Bayes Classification algorithm, Logistic Regression

Text Books

1. Etham Alpaydin, "Introduction to Machine Learning", PHI 2nd Edition – 2013.
2. Peter Flach, "Machine Learning: The Art and Science of Algorithms that Make Sense of Data", Cambridge University Press – 2012.

Reference Books

1. C. M. Bishop, "Pattern Recognition and Machine Learning", Springer 1st Edition – 2013.
2. Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, 3rd Edition – 2011.
3. Kevin Murphy, "Machine Learning – A Probabilistic Perspective", MIT Press – 2012.

Other Resources

1. UCI Machine Learning Repository https://archive.ics.uci.edu
2. WEKA Collection of datasets https://waikato.github.io/weka-wiki/datasets/
3. Kaggle datasets https://www.kaggle.com/datasets

20PEIT 501A Artificial Intelligence

Teaching Scheme:
Lectures: 3 hours/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Discrete Mathematics, Basic Probability Theory and Statistics Knowledge of Data Structures

Course Objectives:
Familiarize students with
1. The basic principles and applications of Artificial Intelligence.
2. Concepts of problem solving and knowledge representation
3. Concepts of planning and learning
4. Concepts of Uncertainty

Course Outcomes:
Students will be able to:
1. Assess underlying AI concepts and their usage.
2. Implement classical Artificial Intelligence techniques
3. Represent knowledge using logic and infer new facts from it.
4. Apply Artificial Intelligence techniques for problem solving.

Unit – I: Artificial Intelligence **7 hours**

Introduction -What is AI? The Foundations of Artificial Intelligence,
Intelligent Agents: Agents and Environments, Good Behaviour: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

Unit – II: Problem Solving by uninformed search **7 hours**

Problem Solving: Solving Problems by Searching, formulation of real world problems, Breadth first search, depth first search, Iterative deepening Depth First Search, Bidirectional Search, Comparison of Uninformed search Strategies, Searching with partial information.

Unit – III: Problem Solving by informed search **7 hours**

Generate& test, Hill Climbing, Best First Search, A* and AO* Algorithm, Constraint satisfaction, Game playing: Minimax Search, Alpha-Beta pruning, Waiting for Quiescence

Unit – IV: Knowledge Representation **7 hours**

Knowledge based agents, Wumpus world. Propositional Logic: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining. First order Logic: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining.

Unit – V: Planning and Uncertainty **7 hours**

Definition of Classical Planning, Algorithm for Planning as State-Space Search, Planning Graphs, Blocks world, STRIPS.
Quantifying Uncertainty: Acting under uncertainty, Basic probability notations, Bayesian probability, belief network, probabilistic reasoning.

Unit – VI: Artificial Neural Network **7 hours**

Introduction to Neural networks: basic, comparison of human brain and machine, biological neuron, general neuron model, activation functions, Perceptron learning rule, applications and advantages of neural networks. Brief introduction to single layer and multilayer networks.

Text Books:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall.

Reference Books:

1. E. Rich and K.Knight, Artificial Intelligence, Tata McGraw Hill
2. Charniack and D. Mcdermott, Artificial Intelligence, Addison Wesley
3. George F. Luger , "Artificial Intelligence: Structures and Strategies for Complex Problem Solving", Pearson
4. N.P. Padhy, "Artificial Intelligence And Intelligent Systems", Oxford University Publishers
5. Parag Kulkarni, Prachi Joshi: Artificial Intelligence: Building Intelligent Systems, Prentice Hall of India.
6. Toby Segaran, Programming Collective Intelligence, O"Reilly

Web References:

1. NPTEL Series: Artificial Intelligence, Prof. Anupam Basu and Prof. S. Sarkar, IIT Kharagpur

20PEIT 501B Business Intelligence

Teaching Scheme:
Lectures: 3 hours/week

Examination Scheme:
In-Semester: 50 Marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Database Systems, probability basics

Course Objectives:
Familiarize students with
1. The role of Business Intelligence in various business applications
2. Methods of data processing and modeling
3. Importance of visualization and reporting in business
4. Decision making process using Business Intelligence

Course Outcomes:
Students will be able to:
1. Solve various business problems using BI concepts
2. Choose data transformation and modeling techniques for designing data warehouse
3. Apply business analytics and visualization concepts for business reporting.
4. Explain different BI trends and their applications.

Unit – I Introduction **6 Hours**

Concepts of Data, Information, and Knowledge, Design and implementation aspect of OLTP and OLAP/Data Warehouse, Business Intelligence (BI) Concepts and definitions, BI architectural models (Top-down and bottom-Up), Business Applications of BI, Role of Data warehouse in BI, BI system components

Unit – II Dimensional Modeling And Data Warehouse Design **8 Hours**

Star schema, Snowflake schema, and Fact Constellation schema, Grain of dimensional model, transactions, Recurring Snapshots, Accumulating Snapshots, Dimensions (SCD types, conformed dimensions), Facts (additive, semi-additive, non-additive), Junk dimensions, conformed dimensions, Bridge tables

Unit – III ETL **8 Hours**

Data Quality, Data profiling, Data enrichment, data duplication, Data cleaning, ETL Architecture and what is ETL, Extraction concept and Change data capture, Transformation concept, loading concept, Initial and Incremental loading, Full loading, late arriving facts, data staging, Data marts, Smart change data capture using log-based techniques

Unit – IV Business Analytics **6 Hours**

What is business analytics (BA)? Difference between BA and BI. Types of analytics, Market-Basket Analysis, clustering, classification, regression, In-Memory Analytics and In-DB Analytics, Applications of Business Analysis

Unit – V Reporting And Data Visualization 8 Hours

Metadata Layer, Presentation Layer, Data Layer, Use of different layers and overall Reporting architecture, Various report elements such as Charts, Tables, Materialized views, Query rewrite,

Unit – VI Recent Trends 6 Hours

Introduction to Big Data, DW appliances, Types of BI: Real time BI, Operational BI, Embedded BI, Agile BI, Mobile BI, collaborative BI, BI for real world applications such as Real estate, Share market

Text Books

1. Ralph Kimball, Joe Caserta, "The Data warehouse ETL Toolkit", Publisher: Wiley
2. Jiawei Han, Micheline Kamber, Jian Pei "Data Mining: concepts and techniques", 2nd Edition, Publisher: Elsevier/Morgan Kaufmann.

Reference Books

1. Ralph Kimball, Margy Ross, "The Data Warehouse Toolkit", 3rd edition, Publisher: Wiley
2. Reema Thareja, "Data Warehouse", Publisher: Oxford University Press.
3. William Inmon, "Building the Data Warehouse", Wiley publication 4th edition.

20PEIT 501C Computer Graphics and Animation

Teaching Scheme:
Lectures: 3 hours/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Geometry and trigonometry, Vectors and Matrices

Course Objectives:
Familiarize students with
1. Basic concepts of computer graphics
2. Basic primitives and objects in computer graphics
3. Various methods and techniques used in computer graphics
4. Applications of computer graphics in animation and gaming

Course Outcomes:
Students should be able to
1. Apply appropriate algorithms for drawing Graphics primitives
2. Apply Geometric Transformations to 2D and 3D Computer Graphics objects
3. Select appropriate techniques for Image Manipulation.
4. Design Algorithmic solutions for Real-Life Applications in Computer Graphics

Unit – I Basic Concepts 7 Hours

Introduction: Basics of graphics systems, raster scan & random scan displays, basic display processor.
Display Files: display file structure, algorithms and display file interpreter. Primitive operations.
Plotting Primitives: Scan conversions, line segments, vectors, pixels and frame buffers, vector generation.
Introduction to OpenGL: Basic OpenGL syntax, display-window management using GLUT, functions.

Unit – II Graphics Primitives for Drawing and Filling 7 Hours

Line and Circle drawing Algorithms: DDA, Bresenham's, Midpoint.
Character Generation: Stroke Principle, Starburst Principle, Bit map method, aliasing and anti-aliasing
Polygon: Polygon and its types, inside test, polygon filling methods: Seed fill, Scan Line, Flood fill and Boundary fill

Unit – III Geometric Transformations 7 Hours

2D Geometric Transformations: Translation, scaling, rotation, reflection, shearing, matrix representation and homogeneous coordinate system, composite transformations
3D Geometric Transformations: Translation, scaling, rotation, rotation about X, Y, Z and arbitrary axis reflection about XY, YZ, XZ and arbitrary plane.

Unit – IV **Segments, Windowing and Clipping** **7 Hours**

Segment: Introduction, segment table, segment creation, closing, deleting and renaming, visibility
Windowing: Concept of window and viewport, viewing transformations
Line Clipping: Cohen sutherland method, midpoint subdivision method
Polygon Clipping: Sutherland hodgman method for clipping convex and concave polygon

Unit – V **Shading and Animation** **7 Hours**

Shading: Halftoning, Gouraud and Phong Shading
Computer Animation: Design of animation sequences, general computer animation functions, computer animation languages, key-frame systems, motion specifications.

Unit – VI **Gaming** **7 Hours**

Gaming platforms: Graphics memory pipeline, block diagram of nvidia workstation and i860 introduction to opengl es
Interactive Graphics & usage of the tools of computer graphics: 3D studio and maya
2D games: Snake game

Textbooks

1. D. Hearn, M. Baker, "Computer Graphics with OpenGL", 4th Edition, Pearson Education, 2014, ISBN 978-93-325-1871-1
2. S. Harrington, "Computer Graphics", 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6.

Reference Books

1. D. Rogers, "Procedural Elements for Computer Graphics", 2nd Edition, Tata McGraw- Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.
2. J. Foley, V. Dam, S. Feiner, J. Hughes, "Computer Graphics Principles and Practice", 2nd Edition,Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
3. D. Rogers, J. Adams, "Mathematical Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2002, ISBN 0 – 07 – 048677 – 8.
4. Zhigang Xiang, Roy Plastock, "Computer Graphics", Schaum"s Series outlines.
5. F.S. Hill JR, "Computer Graphics Using Open GL", Pearson Education
6. Samuel R. Buss, "3D Computer Graphics", Cambridge University Press

20PEIT 502A Blockchain Architecture Design and Use Cases

Teaching Scheme:
Lectures: 3 hours/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Basics of programming, software engineering

Course Objectives:
Familiarize students with

1. Blockchain technology landscape
2. Bitcoin blockchain
3. Ethereum and smart contract
4. Hyperledger

Course Outcomes:
Students should be able to

1. Evaluate the security, privacy, and efficiency of Blockchain systems.
2. Apply Blockchain in real-world applications and implementation strategies.
3. Design Blockchain solutions for practical use cases.
4. Evaluate the security, privacy, and efficiency of Blockchain systems.

Description:

The widespread popularity of digital cryptocurrencies has led the foundation of Blockchain, which is fundamentally a public digital ledger to share information in a trustworthy and secure way. The concept and applications of Blockchain have now spread from cryptocurrencies to various other domains, including business process management, smart contracts, IoT and so on. This course will cover both the conceptual as well as application aspects of Blockchain. This includes the fundamental design and architectural primitives of Blockchain, the system and the security aspects, along with various use cases from different application domains. us other domains, including business process management, smart contracts, IoT and so on.

The course will cover following topics:

Introduction to Blockchain, Basic Crypto Primitives, Bitcoin Basics Distributed Consensus, Consensus in Bitcoin, Permissioned Blockchain (Basics, Consensus, RAFT Consensus, Byzantine General Problem, Practical Byzantine Fault Tolerance), Blockchain for Enterprise – Overview, Blockchain Components and Concepts, Hyperledger Fabric – Transaction Flow, Hyperledger Fabric Details, Fabric – Membership and Identity Management, Hyperledger Fabric Network Setup, Fabric Demo on IBM Blockchain Cloud, Hyperledger Composer – Application Development, Network Administration, Blockchain in Financial Service, Revolutionizing Global Trade, Blockchain in Supply Chain, Blockchain in Government Blockchain Security, Comparing Ecosystems – Ethereum development tools and Quorum

Suggested Swayam Course:

"Blockchain Architecture Design and Use Cases", by Prof. Sandip Chakraborty, IIT, Guwahati https://onlinecourses.nptel.ac.in/noc19_cs63/course

Reference Books:

1. Andreas M. Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", 1st Edition
2. Melanie Swa, "Blockchain", O'Reilly
3. Bob Dill, David Smits, "Zero to Blockchain - An IBM Redbooks course"

20PEIT 502B Internet of Things

Teaching Scheme:
Lectures: 3 hours/week
Tutorial:-

Examination Scheme:
In-Semester: 50 Marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Basic programming knowledge, Network Fundamentals

Course Objectives:

Familiarize students with

1. Core concepts of Internet of Things (IoT)
2. Communication protocols and different types of networks in IoT
3. Programming of various exemplary devices like Arduino, Raspberry Pi
4. State of art IoT technologies and application areas

Course Outcomes:

Students will be able to:

1. Explain core concepts of iot
2. Compare different communication protocols and networks
3. Program exemplary devices like Arduino, raspberry Pi
4. Design iot applications with iot technologies

Description:

Internet of Things (IoT) is presently a hot technology worldwide. Government, academia, and industry are involved in different aspects of research, implementation, and business with IoT. IoT cuts across different application domain verticals ranging from civilian to defence sectors. These domains include agriculture, space, healthcare, manufacturing, construction, water, and mining, which are presently transitioning their legacy infrastructure to support IoT. Today it is possible to envision pervasive connectivity, storage, and computation, which, in turn, gives rise to building different IoT solutions. IoT-based applications such as innovative shopping system, infrastructure management in both urban and rural areas, remote health monitoring and emergency notification systems, and transportation systems, are gradually relying on IoT based systems. Therefore, it is very important to learn the fundamentals of this emerging technology.

The course will cover following topics:

Introduction to IoT, Basics of Networking, Communication Protocols, Sensor Networks, Interoperability in IoT, Introduction to Arduino Programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi, SDN for IoT, Data Handling and Analytics, Cloud Computing, Fog Computing, Smart Cities and Smart Homes, Connected Vehicles, Smart Grid, Industrial IoT

Suggested Swayam Course:

"Introduction to Internet of Things", By Prof. Sudip Misra, IIT, Kharagpur
https://onlinecourses.nptel.ac.in/noc21_cs17/preview

Reference Books

1. S. Misra, A. Mukherjee, and A. Roy, 2020. Introduction to IoT. Cambridge University Press.
2. S. Misra, C. Roy, and A. Mukherjee, 2020. Introduction to Industrial Internet of Things and Industry 4.0. CRC Press.

20IT 502L Design and Analysis of Algorithm Laboratory

Teaching Scheme:
Practical: 2 hours/week
Marks

Examination Scheme:
In Semester: 25

Practical: 25 marks
Credits: 1

Prerequisites: Data Structures

Course Objectives:
Familiarize students with

1. Basics of computational complexities.
2. The space and time requirements of the algorithms.
3. The various algorithmic design techniques.
4. The categorization of the given problem for finding an appropriate solution.

Course Outcomes:
Students should be able to

1. Implement algorithmic solutions to solve computational problems using appropriate strategies.
2. Develop program by applying specific algorithmic paradigms.
3. Analyze the time and space complexity of implemented algorithms..
4. Test and validate algorithms with different inputs and edge cases..

List of Laboratory Assignments (Minimum 4 assignments**)**

1. Write a program to implement an algorithm using Brute Force method or Exhaustive search approach.(For e.g Sorting techniques or Password cracking)
2. Write a program to implement a program using the Divide and Conquer approach (for e.g, Quick, Merge sort, Binary search, Strassen's method).
3. Write a program to implement an algorithm using Greedy method. (for e.g Prims, kruskals, knapsack problem).
4. Write a program to implement an algorithm using Dynamic Programming also verify the complexity. (for e.g Chain matrix multiplication, Bellman-Ford Algorithm, Multistage Graph problem, Optimal Binary Search Trees, Travelling Salesman Problem)
5. Write a recursive program to find the solution using Backtracking approach. (n queens, Graph coloring, Hamiltonian Cycle, 0/1 Knapsack Problem).
6. Write a program to find the solution using Branch and Bound approach (0/1 Knapsack problem – LC branch and bound and FIFO branch and bound solution, Traveling sales person problem, Job scheduling Problem).

Text Books:

1. Thomas H Cormen and Charles E.L Leiserson, Introduction to Algorithm, 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.

20IT 503L Machine Learning Laboratory

Teaching Scheme:
Practical: 2 hours/week

Examination Scheme:
In-Semester: 25 Marks
Oral: 25 marks
Credits: 1

Prerequisites: Python Programming

Course Objectives:
Familiarize students with

1. Programming of Machine Learning algorithms
2. Libraries for Machine Learning
3. Usage of large datasets
4. Evaluation metrics for Machine Learning techniques

Course Outcomes:
Students will be able to:

1. Implement Machine Learning algorithms
2. Compare the performance of various Machine Learning algorithms
3. Apply Machine Learning algorithms to large datasets
4. Evaluate different Machine Learning models

Implement the following assignments using Python.

1. Select a suitable dataset from UCI/Kaggle/Weka for classification. Statistically summarize this dataset. The summaries could be –
    a. dimensions of the dataset,
    b. top and last 5 instances,
    c. mean, count, standard deviation, min, max value for each attribute
    d. class distribution

    AND

    Split the dataset in 1 into training and test datasets. Classify the instances in the test dataset using any two classification algorithms listed below. Compare the results and conclude.

    a. Any one linear model (Perceptron or SVM)
    b. Distance based model (kNN)
    c. Tree based model (ID3)
    d. Probabilistic model (Naïve Bayes)

2. Select a suitable dataset from UCI/Kaggle/Weka for linear regression. Statistically summarize this dataset. Split the dataset into training and test datasets. Predict the values for target attribute. Compare both the methods below.
    a. Use linear regression library and predict values for test instances.

    AND

    b. Use library methods for mean, covariance and variance and predict values for test instances.

3. Select a suitable dataset from UCI/Kaggle/Weka for clustering. Statistically summarize this dataset.

AND

Use any two clustering algorithms listed below and cluster the instances for dataset in 3. Compare the results.

a. Distance based model (k-means and hierarchical clustering)
b. Density based model (DBScan)

Text Books

1. Andrea Muller and Sarah Guido, "Introduction to Machine Learning with Python", O'Reily – 2017.
2. Michael Bowles, "Machine Learning in Python", Wiley – 2018.

Reference Books

1. Ian H. Witten, Eibe Frank, Mark A Hall, "Data Mining: Practical Machine Learning Tools and Techniques", Elsevier 3rd Edition.

Other Resources

1. UCI Machine Learning Repository https://archive.ics.uci.edu
2. WEKA Collection of datasets https://waikato.github.io/weka-wiki/datasets/
3. Kaggle datasets https://www.kaggle.com/datasets

20PEIT 501L A Artificial Intelligence Laboratory

Teaching Scheme:

Practical: 2 hours/week

Examination Scheme:

In-Semester: 25 marks

Oral: 25 marks

Credits: 1

Prerequisites: Discrete mathematics, basic probability theory and statistics, Knowledge of data structures

Course Objectives:

Familiarize students with

1. Basics of Artificial Intelligence
2. Basic implementation of AI algorithms.
3. Intelligence searches and knowledge Representation
4. AI techniques used for application development.

Course Outcomes:

Students should be able to

1. Implement AI core concepts using AI algorithms.
2. Apply AI principles towards problem solving, knowledge representation and learning.
3. Explain the fundamentals of AI applications in intelligent and expert systems, artificial neural networks and other machine learning techniques.
4. Develop AI application using appropriate AI technique through team work.

Suggested List of Laboratory Assignments (Any 5)

1. Implement A* algorithm for any of the following problems: a) 8 puzzle b) Missionaries and Cannibals c) Blocks World Problem
2. Solve 8-queens problem using backtracking.
3. Implement a program to solve constraint satisfaction problem using any searching technique.
4. Implement minimax algorithm using alpha-beta pruning.
5. Implement the code for decision tree learning.
6. Implement Truth Maintenance System.
7. Implement Neural network to understand back propagation.

Group Assignment

1. Develop application such as but not limited to
   . Chatbot
   a. Interactive Sudoku solver
   b. Stock market predictor (offline past data)
   c. Face Recognition
   d. Captcha breakers
   e. Auto tagging of friends on social media
   f. Pac-Man

Text Books:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall.

Reference Books:

1. E. Rich and K.Knight, Artificial Intelligence, Tata McGraw Hill
2. Charniack and D. Mcdermott, Artificial Intelligence, Addison Wesley
3. George F. Luger , “Artificial Intelligence: Structures and Strategies for Complex Problem Solving”, Pearson
4. N.P. Padhy, “Artificial Intelligence And Intelligent Systems”, Oxford University Publishers
5. Parag Kulkarni, Prachi Joshi: Artificial Intelligence: Building Intelligent Systems, Prentice Hall of India.
6. Toby Segaran, Programming Collective Intelligence, O‟Reilly

Web References:

.. NPTEL Series: Artificial Intelligence, Prof. Anupam Basu and Prof. S. Sarkar, IIT Kharagpur

20PEIT 501L B Business Intelligence Laboratory

Teaching Scheme:

Practical: 2 hours/week

Examination Scheme:

In-Semester: 25 Marks
Oral: 25 marks
Credits: 1

Prerequisites: Database Systems, probability basics

Course Objectives:
Familiarize students with

1. Implementation of different Business Intelligence (BI) techniques
2. Methods of data processing and modeling
3. Importance of visualization and reporting in business
4. Various library functions to develop BI applications

Course Outcomes:
Students will be able to:

1. Apply various library functions to develop BI applications
2. Implement data transformation and modeling techniques for building data warehouse
3. Apply business analytics and visualization concepts for business reporting
4. Develop BI system project for real world business application in team

Suggested list of laboratory assignments:
Choose a Business Problem as Case Study to design and build BI solution using BI concepts:

1. Execute ETL process for building data warehouse
2. Perform dimension modeling
3. Implement OLAP operations on given data set
4. Visualize data using various charts using data visualization tool
5. Perform business analytics for the chosen application
6. Demonstrate complete BI application

Text Books

1. Big Data, Black Book, DT Editorial services, 2015 edition
2. Jiawei Han, Micheline Kamber, Jian Pei "Data Mining: concepts and techniques", 2nd Edition, Publisher: Elsevier/Morgan Kaufmann.

Reference Books

1. Ralph Kimball, Margy Ross, "The Data Warehouse Toolkit", 3rd edition, Publisher: Wiley
2. Reema Thareja, "Data Warehouse", Publisher: Oxford University Press.
3. William Inmon, "Building the Data Warehouse", Wiley publication 4th edition.

20PEIT 501L C Computer Graphics Laboratory

Teaching Scheme:

Practical: 2 hours/week

Examination Scheme:

In-semester: 25 Marks
End-semester: 25marks
Credits: 1

Prerequisites: Basics of Programming, Data Structures, Algorithms, Geometry, Trigonometry, Vectors and Matrices

Course Objectives:

Familiarize students with

1. Various methods and techniques used in computer graphics
2. Applications of computer graphics in animation and gaming
3. Functions and Libraries of OpenGL
4. Applications and implementation of computer graphics.

Course Outcomes:

Students should be able to

1. Apply fundamental OpenGL functions to create graphical environments.
2. Implement various algorithms for drawing geometric shapes and filling regions.
3. Perform 2D transformations and clipping techniques for graphical objects.
4. Develop animated scenes and basic games using core computer graphics techniques.

Sr. No.	List of Assignments (minimum 8 out of 10)
1	Get Familiar with basic OpenGL environment, display-window management using GLUT, OpenGL functions.
2	Write a function in OpenGL on Linux Platform to draw a Line using DDA/ Bresenham's Line Drawing Algorithm. Call the Function to draw any pattern consisting of at least 10 function calls.
3	Write a function in OpenGL on Linux Platform to draw a circle using Midpoint Circle Drawing Algorithm. Call this function at least 6 times to draw any pattern. Users should only give center coordinates and radius. Rest should be drawn automatically
4	Write a program in OpenGL on Linux Platform to draw chess board using any Line drawing algorithm and fill alternate blocks using flood fill algorithm
5	Write a program in OpenGL on Linux Platform to draw a flag using any Line drawing algorithm and fill it using scanline polygon filling algorithm.
6	Write a program in OpenGL on Linux Platform to draw a polygon and perform following 2DTransformations on Triangle. Translation, Scaling, Rotation
7	Write a program in OpenGL on Linux Platform to clip a Line using Cohen Sutherland Outcode Method.
8	Write a program in OpenGL on Linux Platform to clip a Polygon using Sutherland Hodgman Polygon Clipping.
9	Write a program in OpenGL on Linux Platform to animate a scene like "Moving Car", "kite flying" etc.

10 Write a program to design a game using computer graphics basic techniques and OpenGL

Textbooks

1. D. Hearn, M. Baker, "Computer Graphics with OpenGL", 4th Edition, Pearson Education, 2014,ISBN 978-93-325-1871-1
2. S. Harrington, "Computer Graphics", 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6.

Reference Books

1. D. Rogers, "Procedural Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill publication, 2001, ISBN 0 – 07 – 047371 – 4.
2. J. Foley, V. Dam, S. Feiner, J. Hughes, "Computer Graphics Principles and Practice", 2nd Edition,Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
3. D. Rogers, J. Adams, "Mathematical Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2002, ISBN 0 – 07 – 048677 – 8.
4. Zhigang Xiang, Roy Plastock, "Computer Graphics", Schaum"s Series outlines
5. F.S. Hill JR, "Computer Graphics Using Open GL", Pearson Education
6. Samuel R. Buss, "3D Computer Graphics", Cambridge University Press