RCOEM

Shri Ramdeobaba College of
Engineering and Management, Nagpur

SHRI RAMDEOBABA COLLEGE OF ENGINEERING AND MANAGEMENT, NAGPUR – 440013

PROGRAMME SCHEME & SYLLABI

2023-24

B.Tech in Computer Science and Engineering (Data Science)

B. Tech. Computer Science and Engineering (Data Science) [2023-24]

Teaching & Evaluation Scheme [B.Tech CSE(DS)]

Semester–I

Sr. No.	Category	Course Code	Course Name	Hours/week			Credits	Maximum marks			ESE Duration (Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	BSC	CHT1001	Chemistry of Smart Materials	2	0	0	2	50	50	100	2
2.	BSC	CHP1001	Chemistry of Smart Materials Lab	0	0	2	1	50	-	50	-
3.	BSC	MAT1002	Calculus	3	0	0	3	50	50	100	3
4.	ESC	CDT1001	Digital Electronics	3	0	0	3	50	50	100	3
5.	ESC	CDP1001	Digital Electronics Lab	0	0	2	1	50	-	50	-
6.	ESC	CDT1002	Programming for problem solving	3	0	0	3	50	50	100	3
7.	ESC	CDP1002	Programming for problem solving Lab	0	0	2	1	50	-	50	-
8.	VSEC	CDT1003	Computer Workshop – I	1	0	0	1	50	-	50	-
9	VSEC	CDP1003	Computer Workshop – I Lab	0	0	2	1	50	-	50	-
10	HSSM-IKS	HUT1001	Foundational Literature of Indian Civilization	2	0	0	2	50	50	100	2
11	CCA	PET1001	Sports-Yoga-Recreation	1	0	0	1	50	-	50	-
12	CCA	PEP1001	Sports-Yoga-Recreation Lab	0	0	2	1	50	-	50	-
TOTAL				**15**	**0**	**10**	**20**			**850**	**-**

Semester - II

Sr. No.	Category	Course Code	Course Name	Hours/week			Credits	Maximum marks			ESE Duration (Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	BSC	PHT2001	Introduction to Quantum Computing	2	1	0	3	50	50	100	3
2.	BSC	PHP2001	Introduction to Quantum Computing Lab	0	0	2	1	50	-	50	-
3.	BSC	MAT2002	Discrete Mathematics	3	0	0	3	50	50	100	3
4.	BSC	MAP2001	Computational Mathematics Lab	0	0	2	1	50	-	50	-
5.	BSC	CHT2007	Bioinformatics	2	0	0	2	50	50	100	2
6.	ESC	CDT2001	Object Oriented Programming	3	0	0	3	50	50	100	3
7.	ESC	CDP2001	Object Oriented Programming Lab	0	0	2	1	50	-	50	-
8.	PCC	CDT2002	Computer Architecture	2	0	0	2	50	50	100	2
9.	VSEC	CDT2003	Computer Workshop – II	1	0	0	1	50	-	50	-
10.	VSEC	CDP2003	Computer Workshop – II Lab	0	0	2	1	50	-	50	-
11.	AEC	HUT2002	English for Professional Communication	2	0	0	2	50	50	100	2
12.	AEC	HUP2002	English for Professional Communication Lab	0	0	2	1	50	-		
13.	CCA	HUP0001	Liberal/Performing Art	0	0	2	1	50	-	50	-
14.	VEC	HUT2004	Foundational Course in Universal Human Values	1	0	0	1	50	-	50	-
TOTAL				**16**	**1**	**12**	**23**			**1000**	**-**

Exit option : Award of UG Certificate in Major with 43 credits and an additional 8 credits.			
Exit Courses			
1	**Web Designer**	**Online/offline certification Course**	**8**
2	**IT Support Engineer**		**8**
3	**Certified Programmer (language learned in Sem-1 and/or Sem-2 [C,C++,Java, Python])**		**8**

Semester - III

Sr. No.	Category	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1	PCC	CDT3001	Data Structures and Algorithms	3	1	0	4	50	50	100	3
2	PCC	CDP3001	Data Structures and Algorithms Lab	0	0	2	1	50	-	50	-
3	PCC	CDT3002	Theory of Computations	3	0	0	3	50	50	100	3
5	PCC	CDP3003	Programming Language Lab	0	0	4	2	50	-	50	-
6	MDM	MAT3002	Probability and Statistics	3	0	0	3	50	50	100	3
8	VSEC	CDP3004	Software Laboratory - I	0	0	2	1	50	-	50	-
7	OE	CDT2980-1	Open Elective - I	2	0	0	2	50	50	100	2
8	AEC	HUT3001	Business Communication	2	0	0	2	50	50	100	2
9	HSSM	CDP3006	Idea Lab	0	0	4	2	50	-	50	-
10	VEC	CDT3007	Cyber Laws and Ethics in IT	2	0	0	2	50	50	100	2
TOTAL				**17**	**1**	**12**	**22**			**800**	

Semester - IV

Sr. No.	Category	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1	PCC	CDT4001	Operating Systems	3	0	0	3	50	50	100	3
2	PCC	CDP4001	Operating Systems Lab	0	0	2	1	50	-	50	-
3	PCC	CDT4002	Artificial Intelligence	3	0	0	3	50	50	100	3
4	PCC	CDP4002	Artificial Intelligence Lab	0	0	2	1	50	-	50	-
5	PCC	CDT4003	Design and Analysis of Algorithms	3	0	0	3	50	50	100	3
6	MDM	MAT4001	Linear Algebra	3	0	0	3	50	50	100	3
7	OE	CDT2990-1	Open Elective - II	3	0	0	3	50	50	100	2
8	VSEC	CDP4005	Software Laboratory - II	0	0	2	1	50	-	50	-
9	HSSM	HUT4003	Managerial Economics	2	0	0	2	50	50	100	2
10	VEC	HUT4002	Environment Education	2	0	0	2	50	50	100	2
11	CEP	CDP4006	Community Engagement Project	0	0	4	2	50	-	50	-
TOTAL				**18**	**0**	**10**	**24**	**550**	**350**	**900**	

Exit option : Award of UG Diploma in Major with 90 credits and an additional 8 credits			
Exit Courses			
1	Application Development (Android, Advanced Java)	Online/offline certification Course	8
2	Certified Software Engineer (DevOps)		8

Semester- V

Sr. No.	Category	Course Code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1	PCC	CDT5001	Database Management Systems	3	0	0	3	50	50	100	3
2	PCC	CDP5001	Database Management Systems Lab	0	0	2	1	50	-	50	-
3	PCC	CDT5002	Machine Learning	3	0	0	3	50	50	100	3
4	PCC	CDP5002	Machine Learning Lab	0	0	2	1	50	-	50	-
5	PCC	CDT5003	Computer Networks	3	0	0	3	50	50	100	3
6	PEC	CDT5004	Program Elective - I	3	0	0	3	50	50	100	3
7	MDM	CDT5005	Financial Analytics	3	0	0	3	50	50	100	3
8	MDM	CDP5005	Financial Analytics Lab	0	0	2	1	50	-	50	-
9	OE	CDT3980-1	Open Elective - III	3	0	0	3	50	50	100	3
TOTAL				**17**	**0**	**6**	**21**	**450**	**300**	**750**	

Semester- VI

Sr. No.	Category	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1	PCC	CDT6001	Data Warehousing and Business Intelligence	3	0	0	3	50	50	100	3
2	PCC	CDP6001	Data Warehousing and Business Intelligence Lab	0	0	2	1	50	-	50	-
3	PCC	CDT6002	Deep Learning - I	3	0	0	3	50	50	100	3
4	PCC	CDP6002	Deep Learning - I Lab	0	0	2	1	50	-	50	-
5	PCC	CDT6003	Data Analysis and Visualization	1	0	0	1	50	-	50	-
6	PCC	CDP6003	Data Analysis and Visualization Lab	0	0	2	1	50	-	50	-
7	PEC	CDT6004	Program Elective - II	3	0	0	3	50	50	100	3

8	PEC	CDP6004	Program Elective - II Lab	0	0	2	1	50	-	50	-
9	PEC	CDT6005	Program Elective - III	3	0	0	3	50	50	100	3
10	MDM	CDT6006	Applied Econometrics	2	0	0	2	50	50	100	2
11	VSEC	CDP6007	Mini Project	0	0	4	2	25	25	50	-
TOTAL				**15**	**0**	**12**	**22**	**525**	**275**	**800**	

Exit option : Award of UG Degree in Major with 131 credits and an additional 8 credits			
Exit Courses			
1	**Certified Database Engineer (Oracle, DB2)**	**Online/offline certification Course**	**8**
2	**Certified Cloud Engineer (AWS, AZURE)**		**8**
3	**Certified Data Science Engineer**		**8**

Semester-VII

Sr. No.	**Category**	**Course code**	**Course Name**	**Hours/ week**			**Credits**	**Maximum marks**			**ESE Duration (Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	PCC	CDT7001	Deep Learning - II	3	0	0	3	50	50	100	3
2	PCC	CDP7001	Deep Learning - II Lab	0	0	2	1	50	-	50	-
3	PCC	CDT7002	Cloud Computing	3	0	0	3	50	50	100	3
4	PCC	CDP7002	Cloud Computing Lab	0	0	2	1	50	-	50	-
5	PEC	CDT7003	Program Elective - IV	3	0	0	3	50	50	100	3
6	PEC	CDP7003	Program Elective - IV Lab	0	0	2	1	50	-	50	-
7	MDM	CDT7004	Customer Management	2	0	0	2	50	50	100	2
8	PRJ	CDP7005	Major Project - I	0	0	8	4	50	50	100	-
			TOTAL	**10**	**0**	**16**	**18**	**400**	**250**	**650**	

Semester- VIII

Sr. No.	Category	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	PEC	CDT8001	Program Elective-V	3	0	0	3	50	50	100	3
2.	PEC	CDT8002	Program Elective-VI	3	0	0	3	50	50	100	3
3.	Project	CDP8003	Major Project- II	0	0	12	6	50	50	100	-
			TOTAL	**6**	**0**	**12**	**12**	**150**	**150**	**300**	
OR											
								Continuous Evaluation	Industry Evaluation	Total	
1.	INTR	CDP8004	Industry Internship /TBI Internship / Research Internship	0	0	24	12	100	100	200	-
			TOTAL	**0**	**0**	**24**	**12**	100	100	**200**	

List of Program Electives

Elective-I	Elective-II	Elective-III	Elective-IV	Elective-V	Elective-VI
CDT5004-1 Compiler Design	CDT6004-1 Natural Language Processing	CDT6005-1 Data Science for Health Care	CDT7003-1 Image Analysis and Computer Vision	CDT8001-3 Mining Massive Datasets	CDT8002-1 Human Computer Interaction
CDT5004-2 Design Patterns	CDT6004-2 Blockchain Technology	CDT6005-2 Data Science for Genomics	CDT7003-2 Spatial Data Management	CDT8001-2 Generative AI	CDT8002-2 Optimization for Data Science
CDT5004-3 Software Engineering	CDT6004-3 Distributed and Parallel Computing	CDT6005-3 Data Science for Marketing	CDT7003-3 Information Retrieval	CDT8001-3 Information Security and Data Privacy	CDT8002-3 Reinforcement Learning

Syllabus for Semester III, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDT3001	**Category:** Program Core Course (PCC)
Course: Data Structures and Algorithms	**L:** 3Hrs**, T:** 1Hr**, P:** 0Hr**, Per Week, Credits:** 4

Course Objectives

The objective of this course is to familiarize the prospective engineers with:

1. To impart to students the basic concepts of data structures and algorithms.
2. To familiarize students on different searching and sorting techniques.
3. To prepare students to use linear (stacks, queues, linked lists) and non-linear (trees, graphs) data structures.
4. To enable students to devise algorithms for solving real-world problems.

SYLLABUS

UNIT I Data Structures and Algorithms Basics

Introduction: basic terminologies, elementary data organizations, data structure operations; abstract data types (ADT) and their characteristics. Algorithms: definition, characteristics, analysis of an algorithm, asymptotic notations, time and space tradeoffs. Array ADT: definition, operations and representations – row-major and column- major.

UNIT II Stacks and Queues

Stack ADT: allowable operations, algorithms and their complexity analysis, applications of stacks –expression conversion and evaluation (algorithmic analysis), multiple stacks. Queue ADT: allowable operations, algorithms and their complexity analysis for simple queue and circular queue, introduction to double-ended queues and priority queues.

UNIT III Linked Lists

Singly Linked Lists: representation in memory, algorithms of several operations: traversing, searching, insertion, deletion, reversal, ordering, etc.Doubly and Circular Linked Lists: operations and algorithmic analysis. Linked representation of stacks and queues, header node linked lists.

UNIT IV Sorting and Searching

Sorting: different approaches to sorting, properties of different sorting algorithms (insertion, Shell, quick, merge, heap, counting), performance analysis and comparison.

Searching: necessity of a robust search mechanism, searching linear lists (linear search, binary search)and complexity analysis of search methods.

UNIT V Trees

Trees: basic tree terminologies, binary tree and operations, binary search tree [BST] and operations with time analysis of algorithms, threaded binary trees. Self-balancing Search Trees: tree rotations, AVL tree and operations, B+-tree: definitions, characteristics, and operations (introductory).

UNIT VI Graphs and Hashing

Graphs: basic terminologies, representation of graphs, traversals (DFS, BFS) with complexity analysis, path finding (Dijkstra's SSSP, Floyd's APSP), and spanning tree (Prim's method) algorithms.

Hashing: hash functions and hash tables, closed and open hashing, randomization methods (division method, mid-square method, folding), collision resolution techniques.

Course Outcomes

After successful completion of this course, the student will be able to,

1. Recognize different ADTs and their operations and specify their complexities.
2. Design and realize linear data structures (stacks, queues, linked lists) and analyze their computation complexity.
3. Devise different sorting (comparison based, divide-and-conquer, distributive, and tree-based) and searching (linear, binary) methods and analyze their time and space requirements.
4. Design traversal and path finding algorithms for Trees and Graphs.

Text Books

1. Ellis Horowitz, Sartaj Sahni& Susan Anderson-Freed, Fundamentals of Data Structures in C,Second Edition, Universities Press, 2008.
2. **Robert L. Kruse and Bruce P. Leung, Data Structures and Program Design in C, Pearson Education;**
3. Mark Allen Weiss; Data Structures and Algorithm Analysis in C; Second Edition; Pearson Education; 2002.
4. G.A.V. Pai; Data Structures and Algorithms: Concepts, Techniques and Application; FirstEdition; McGraw Hill; 2008.

Reference Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein; Introduction to Algorithms; Third Edition; PHI Learning; 2009.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran; Fundamentals of Computer Algorithms; Second Edition; Universities Press; 2008.
3. A. K. Sharma; Data Structures using C, Second Edition, Pearson Education, 2013.

Syllabus for Semester III, B. Tech. Computer Science & Engineering(Data Science)

Course Code: CDP3001	**Category:** Program Core Course (PCC)
Course: Data Structures and Algorithms Lab	**L:** 0Hr**, T:** 0Hr**, P:** 2Hrs**, Per Week, Credits:** 1

Course Outcome

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

1. To enable students to employ different searching and sorting methods.
2. To prepare students to identify and apply linear (stacks, queues, linked lists) and non-Linear(trees, graphs) data structures in solving problems.
3. To encourage students to design and execute tree-based algorithms for solving real-world problems.

SYLLABUS

Experiments based on CDT3001 Syllabus in C / C++ / Java / Python

Course Outcomes

On completion of the course the student will be able to

1. Design and realize different linear data structures.
2. Identify and apply specific methods of searching and sorting to solve a problem.
3. Implement and analyze operations on binary search trees and AVL trees.
4. Implement graph traversal algorithms, find shortest paths and analyze them.

Reference Books

1. K R. Venugopal and Sudeep. R Prasad; Mastering C; Second Edition; McGraw Hill; 2015.

2. Ellis Horowitz, SartajSahni& Susan Anderson-Freed, Fundamentals of Data Structures in C,Second Edition, Universities Press, 2008.

3. Mark Allen Weiss; Data Structures and Algorithm Analysis in C; Second Edition; PearsonEducation; 2002.

Syllabus for Semester III, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDT3002	**Category:** Program Core Course (PCC)
Course: Theory of Computations	**L:** 3Hrs**, T:** 0Hr**, P:** 0Hr**, Per Week, Credits:** 3

Course Objectives

The objective of this course is to familiarize the prospective engineers with:
1. To provide students an understanding of basic concepts in the theory of computation.
2. To teach formal languages and various models of computation.
3. To exhibit fundamental concepts related with computability theory.

SYLLABUS

UNIT-I Basics of Theory of Computation

Basics of Sets and Relation, Countability and Diagonalization, Principle of mathematical induction, Pigeon-hole principle. Fundamentals of formal languages and grammars, Chomsky hierarchy of languages.

UNIT-II Finite Automata

Deterministic finite automata (DFA), Nondeterministic finite automata (NFA) and equivalence with DFA, Minimization of finite automata, NFA with Epsilon Transitions, Finite Automata with output.

UNIT-III Regular Grammar

Regular expressions and Regular languages, Regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages

UNIT-IV Context-Free Grammar

Context-free grammars (CFG) and language (CFL), parse trees, ambiguity in CFG, Reduction of CFGs, Chomsky and Greibach normal forms.

UNIT-V Push Down Automata

Deterministic pushdown automata and Non-Deterministic push down automata, Acceptance by two methods: Empty stack and Final State, Equivalence of PDA with CFG, closure properties of CFLs.

UNIT-VI Turing Machines

The basic model for Turing machines (TM), Turing recognizable recursively enumerable) and Turing-decidable (recursive) languages, variants of Turing machines, unrestricted grammars and equivalence with Turing machines, TMs as enumerators.

Course Outcomes

On successful completion of the course, students will be able to demonstrate

1. Describe the formal relationships among machines, languages and grammars.
2. Design and Optimize finite automata for given regular languages.
3. Design context free grammar for a given language.
4. Design Pushdown Automata, Turing Machine for given languages.

Text Books

1. John Martin, Introduction to Languages and The Theory of Computation, Tata McGraw Hill

Reference Books

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.
2. Harry R. Lewis and Christos H. Papadimitriou, Elements of the Theory of Computation, Pearson Education Asia.
3. Dexter C. Kozen, Automata and Computability, Undergraduate Texts in Computer Science, Springer.
4. Michael Sipser, Introduction to the Theory of Computation, PWS Publishing.

Syllabus for Semester III, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP3003	**Category:** Program Core Course (PCC)
Course: Programming Language Lab	**L:** 0Hrs**, T:** 0Hr**, P:** 4Hr**, Per Week, Credits:** 2

Course Objectives:

1. The course focuses on developing the python programming skills to do a variety of programming tasks where the students are encouraged to develop application using python.
2. To cover the basic constructs of python programming, data structures and object-oriented concepts.
3. The course also targets the coverage of important modules and libraries available in python.

Course Outcome

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

1. Design Python Programs using Different Data and Control Structures.
2. Use Functions, Python Files, Modules and Packages to handle complex python programs.
3. Develop Mathematical Models and Scientific Applications in Python using various Libraries.
4. Write Small Python Applications using Web Scrapping and PyGame Modules.

Syllabus

- Introduction to Python: Basic Building Blocks of Python, Installation and Working with Python, Keywords, Variables and Operators.
- Data Types: int , float, complex, User Input, Arithmetic Expressions ,Using Strings and Operations on Strings, List, Slicing List and Strings, Split, Tuples, Dictionary, Sets.
- Flow Control: Conditional blocks: if, else, elseif, for Loops in Python: Loops with range, Strings, List and Dictionaries, While Loop, break, continue and pass statements.
- Python Functions: Library Functions, User Defined Functions, Function Argument Types, Recursion, Returning Multiple Values, Lambda, Map, Filter, Reduce.
- Python OOPS Basics: Classes, Object, Class Variable and Instance Variable.
- Files: Reading and Writing Files in Python, File Operations and Modes.Modules and Packages: To Create and Import Module.

- Open-Source Python library- Pandas, Sci Py, NumPy, Matplotlib and Seaborn.
- Web scrapping: Web scrapping with the help of standard libraries like Requests and Beautiful Soup.
- Python Pygame (Game Development Library)

Practical based on above theory syllabus

Text Books

1. Python Programming Using Problem Solving Approach: Reema Thareja, OxfordUniversity, Press; First edition.
2. Learning Python: Powerful object-oriented programming, Mark Lutz, O'REILLY publications 5th addition.
3. Introduction to Computing & Problem Solving with Python Jeeva Jose and P SojanLal Ascher.
4. Problem Solving with Algorithms and Data Structures using Python by Brad Miller and David Ranum, 2nd addition**.**

Reference Books

1. Allen Downey, Jeffrey Elkner, Chris Meyers, Learning with Python, Dreamtech Press
2. David M. Baezly “Python Cookbook” O'Reilly Media; Third edition, 2013.

Syllabus for Semester III, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP3004	**Category:** (VSEC)
Course: Software Laboratory - I	**L:** 0Hr, **T:** 0Hr, **P:** 2Hrs, **Per Week, Credits:** 1

Course Objectives:

1. Familiarize students with the use and applications of various statistical tools and packages in data science.
2. Provide hands-on experience in statistical programming and data analysis.
3. Equip students with the skills to perform exploratory data analysis, statistical modeling, and inference.

Syllabus can include practical based on following:

1. **Data Exploration and Visualization**

Load a dataset containing information about housing prices in a city. Explore the dataset, calculate summary statistics, and visualize the distribution of housing prices using appropriate plots.

2. **Descriptive Statistics**

Given a dataset of student grades, calculate measures of central tendency (mean, median, mode) and measures of variability (variance, standard deviation). Interpret the results and discuss the implications of different measures.

3. **Probability Distributions**

Simulate a binomial distribution with given parameters (number of trials and probability of success). Plot the distribution and calculate the probability of different events occurring.

4. **Sampling Distributions and Confidence Intervals**

Generate a random sample from a population with known mean and standard deviation. Construct confidence intervals for the sample mean and interpret the results.

5. **Hypothesis Testing**

Analyze a dataset containing the weights of two different groups of individuals. Conduct a two-sample t-test to determine if there is a significant difference in mean weights between the two groups.

6. **Chi-square Test**

Given a dataset of categorical data, perform a chi-square test to determine if there is a significant association between two categorical variables.

7. **Analysis of Variance (ANOVA)**

Analyze a dataset containing crop yields from different fertilizer treatments applied to multiple plots of land. Conduct a one-way ANOVA to determine if there is a significant difference in mean crop yields among the different fertilizer treatments.

8. **Simple Linear Regression & Multiple Linear Regression**

Using a dataset of housing prices and square footage, fit a simple linear regression model to predict housing prices based on square footage. Evaluate the model's performance and interpret the regression coefficients.

Analyze a dataset containing information about cars, including factors such as engine size, fuel efficiency, and price. Fit a multiple linear regression model to predict the price of a car based on various features.

9. **Non-linear Regression**

Given a dataset of population growth over time, fit a non-linear regression model (e.g., exponential or logistic) to model the population growth pattern. Evaluate the model's fit and interpret the results.

10. **Logistic Regression**

Using a dataset containing customer data and purchase history, build a logistic regression model to predict whether a customer will make a purchase based on various factors (e.g., age, income, past purchases).

These problem statements cover a range of scenarios and applications, allowing students to practice data exploration, visualization, descriptive statistics, probability distributions, inferential statistics, regression analysis, and their implementation using statistical programming tools. Additionally, the case study problem statement encourages students to apply their skills to a real-world dataset and communicate their findings effectively.

Course Outcomes:

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

1. Understand and apply different statistical techniques on given data.
2. Perform and interpret various probability distributions.
3. Carry out hypothesis testing and calculate confidence intervals.
4. Create and evaluate regression models.
5. Implement statistical analysis using programming tools and packages.
6. Analyze and visualize data using statistical programming techniques.
7. Apply statistical methods and programming skills to real-world data science problems.

Reference Books:

1. An Introduction to Statistics with Python: With Applications in the Life Science by Thomas Haslwanter, Springer.
2. Practical Statistics for Data Scientists by Peter Bruce, Andrew Bruce, Peter Gedeck, Reilly Media.

Syllabus for Semester III, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP3006	**Category:** (HSSM)
Course: Idea Lab	**L:** 0Hr**, T:** 0Hr**, P:** 4Hrs**, Per Week, Credits:** 2

Course Objectives

This course provides students with a unique opportunity to engage, explore, experience, express and excel in innovative thinking. The course will accelerate the development of indigenous solutions by inculcating creative skills, problem-solving skills and entrepreneurship skills in students. The ultimate goal is to help students build his/her creative acumen to address real life challenges.

Course Outcomes:

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

1. Identify a real-world challenge or potential opportunities that demand consideration.
2. Design and propose original ideas or innovative solutions to identified real-world challenge.
3. Build a plan to develop proof of concept (POC)/ prototype/minimum viable product (MVP) for the idea.
4. Critically evaluate the feasibility and viability of proposed idea, considering its impact on various stakeholders.

Execution Plan for the Subject:

- Conduction of lectures/guest lectures to familiarize the students with concepts of principles of design thinking, innovative and business oriented solution building.
- Students will identify the potential areas or real-world problems that require innovative solution.
- Students will contribute to the creation and presentation of their solution. The developed solution can be a prototype or an idea or a business plan or a software solution, etc.
- Students will analyze and provide constructive feedback on the ideas and solutions presented by self and peers.

Syllabus for Semester III, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDT3007	**Category:** Value Education Course (VEC)
Course: Cyber Laws and Ethics in IT	**L:** 2Hrs**, T:** 0Hr**, P:** 0Hr**, Per Week, Credits:** 2

Course Objectives:

1. To explore ethical principles in the business world and IT organization
2. To understand various intellectual property issues such as copyrights, patents, Trade secrets, and trademarks.
3. To identify emerging plagiarism policies, anonymity issues, identity theft, and consumer profiling.
4. To interpret Indian Information Technology Act 2000 and explore cyber security issues in the society and business world.

Syllabus:

UNIT I

Ethics in business world & IT professional malpractices, Introduction to firewalls, IDS System, Distortion and fabrication of information

UNIT II

Ethics of IT Organization: Contingent Workers H- IB Workers, Whistle- blowing, Protection for Whistle- Blowers, Handling Whistle- blowing situation, Digital divide.

UNIT III

Intellectual Property: Copyrights, Patents, Trade Secret Laws, Key Intellectual property issues, Plagiarism, Privacy: The right of Privacy, Protection, Key Privacy and K-Anonymity issues, Identity Theft, Consumer Profiling,

UNIT IV

Cyber laws and rights in today's digital age, Emergence of Cyberspace, Cyber Jurisprudence, Cyber Crimes against Individuals, Institution and State, Hacking, Digital Forgery, Cyber Stalking/Harassment, Cyber terrorism, cyber tort, Cyber Defamation & hate speech,Competitive Intelligence, Cybersquatting, The IndianInformation Technology Act 2000 IT Act.

Course Outcomes

On successful completion, of course student will able to learn:

1. To analyze the role of ethics in IT organization.
2. To identify various cyber laws with respect to legal dilemmas in the Information Technology field.
3. To interpret various intellectual property rights, Privacy, Protection issues in Information Technology field.
4. To describe the ways of precaution and prevention of Cyber Crime as well as Human Rights.

Text Books:

1. George Reynolds, “Ethics in information Technology”,5th edition, Cengage Learning
2. Hon C Graff, Cryptography and E-Commerce - A Wiley Tech Brief, Wiley Computer Publisher, 2001.

Reference Books:

1. Michael Cross, Norris L Johnson, Tony Piltzecker, Security, Shroff Publishers and Distributors Ltd.
2. Debora Johnson,” Computer Ethic s”,3/e Pearson Education.
3. Sara Baase, “A Gift of Fire: Social, Legal and Ethical Issues, for Computing and the Internet,” PHI Public at ions.
4. Chris Reed & John Angel, Computer Law, OUP, New York, (2007).
5. Dr Pramod Kr.Singh, “Laws on Cyber Crimes [Along with IT Act and Relevant Rules]” Book Enclave Jaipur India.

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDT4001	**Category:** Program Core Course (PCC)
Course: Operating System	**L:** 3Hrs**, T:** 0Hr**, P:** 0Hr**, Per Week, Credits:** 3

Course Objectives

The objective of this course is to familiarize the prospective engineers with:

1. To learn how Operating System is Important for Computer System.
2. To make aware of different types of Operating System and their services.
3. To learn different process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
4. To know memory organization and management concepts.
5. To learn secondary memory, directory and file management

SYLLABUS

Unit I: Introduction:

Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine, Case study on different Operating Systems.

Unit II: Processes and CPU Scheduling:

Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

Threads: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads.

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SRTF, Priority, RR

Unit III: Process Synchronization:

Critical Section, Mutual Exclusion, Peterson’s solution, Hardware Solution, Software Solutions, Semaphores, Monitors, Message Passing, Classical IPC Problems: Producer-Consumer Problem, Reader-Writer Problem, Dinning Philosopher Problem etc.

Unit IV: Deadlocks:

Definition, Necessary and sufficient conditions for Deadlock, Starvation, Deadlock Prevention, Deadlock Avoidance: Banker’s algorithm, Deadlock detection and Recovery.

Unit V: Memory Management:

Basic concept, Logical and Physical address mapping, Memory allocation: Contiguous Memory allocation, Fixed and variable partition, Internal and External fragmentation and Compaction,

Paging: Page allocation, Hardware support for paging, Advantages & Disadvantages of paging. Segmentation, Virtual Memory: Basics of Virtual Memory

Demand Paging : Page Fault Mechanism, Page Replacement, Page Replacement algorithms: First in First Out (FIFO), Least Recently used (LRU), and Optimal.

Unit VI: File Management:

Concept of File, Access methods, File types, File operations, Directory structure, File System structure

Disk Management**:** Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK

Course Outcomes

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

1. Understands the different services provided by Operating System at different level.
2. Understanding the performance and design trade-offs in complex software systems
3. Understand the role of various components (process, memory, file systems, etc.) of operating system.
4. Understands the use of different process scheduling algorithm and synchronization techniques to avoid deadlock
5. Analyze and apply resource (CPU, Memory, Disk) management policies.

Text Books

1. Operating System Concepts, 8th Edition by A. Silberschatz, P.Galvin, G. Gagne,Wiley India Edition.
2. Modern Operating Systems, 2nd Edition by Andrew Tanenbaum, PHI.

Reference Books

1. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings,Prentice Hall of India.
2. Operating Systems, W. Stalling, Macmillan.
3. Operating Systems, H. M. Dietel, Addison Wesley Longman.
4. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly.
5. Linux Kernel Development, Robert Love, Third Edition, Addison-Wesley, 2010.
6. The design of Unix Operating system, Maurice J. Bach, Pearson Education, India.

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP4001	**Category:** Program Core Course (PCC)
Course: Operating System Lab	**L:** 0Hr, **T:** 0Hr, **P:** 2Hrs**, Per Week, Credits:** 1

SYLLABUS

Practical based on CDT4001 Syllabus

1. Linux System Commands
2. Execution of I/O system calls
3. CPU scheduling algorithms
4. Inter process communication
5. Process synchronization
6. Deadlock Detection
7. Memory Management
8. Disk management

Course Outcomes

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

1. Implement system commands by making use of LINUX system calls.
2. Implement processes and process schedulers.
3. Design solutions to process synchronization and deadlock handling.
4. Implement Memory management and File management solutions.

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDT4002	**Category:** Program Core Course (PCC)
Course: Artificial Intelligence	**L:** 3Hrs**, T:** 0Hr**, P:** 0Hr**, Per Week, Credits:** 3

Course Objectives

The objective of this course is to familiarize the prospective engineers with:

1. To understand challenges involved in designing intelligent systems.
2. To represent given problem using state space representation and solve it by using different search techniques.
3. To understand knowledge representation methods using logic programming.
4. To understand uncertainty theory in designing AI systems.

SYLLABUS

UNIT I:

Introduction: Basics of problem solving, problem representation (toy problems and real world problems); Structure of agent, rational agent, Specifying task environment, Properties of task environment; measuring problem-solving performance

UNIT II:

Uninformed search techniques: Depth, Breadth, Uniform Cost, Depth Limited, Iterative deepening DFS, Bidirectional Search

UNIT III:

Informed search techniques: Heuristic Based Search, Greedy Best First Search, A* Search; Local Search algorithms: Hill-climbing, Simulated Annealing, Genetic Algorithms.

UNIT IV:

Adversarial Search: Two player Games, The min-max algorithm, Alpha-Beta pruning. Constraint Satisfaction Problems: Constraint propagation, backtracking search

UNIT V:

Propositional Logic: Inference, Equivalence, Validity and satisfiability, Resolution, Forward and Backward Chaining, First Order Logic: Syntax and Semantics of FOL, Inference in FOL, Unification, Forward Chaining, Backward Chaining, and Resolution.

UNIT VI:

Uncertainty Knowledge and Reasoning: Probability and Baye's Theorem, Statistical reasoning: Bayesian networks, Naïve bayes algorithm, Fuzzy Logic, Introduction to expert system

Course Outcomes

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

1. Represent given problem using state space representation
2. Apply uninformed and informed search techniques.
3. Solve the fully informed two player games using different AI techniques.
4. Solve the AI problems by using logic programming
5. Apply uncertainty theory based on techniques like probability theory and fuzzy logic.

Text Book:

1. Stuart Russel and Peter Norvig; Artificial Intelligence: A Modern Approach; Third Edition; Pearson Education, 2009.

Reference Books:

1. E.Rich, K. Knight, S. B. Nair; Artificial Intelligence; 3rd Edition; Tata McGraw Hill, 2014.
2. Denis Rothman; Artificial Intelligence By Example: Develop machine intelligence from scratch using real artificial intelligence use cases; Kindle Edition, Packt Publishing Ltd, 2018

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP4002	**Category:** Program Core Course (PCC)
Course: Artificial Intelligence Lab	**L:** 0Hrs**, T:** 0Hr**, P:** 2Hr**, Per Week, Credits:** 1

Course Outcomes:

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

1. Implement different AI toy problems by using search techniques.
2. Design two player games using min-max algorithm with Alpha-Beta pruning.
3. Simulate AI problems using logic programming.
4. Implement algorithms to handle uncertainity.

PRACTICALS BASED ON CDT4002 SYLLABUS

Reference Books

1. Stuart Russel and Peter Norvig; Artificial Intelligence: A Modern Approach; Third Edition; Pearson Education.
2. E.Rich, K. Knight, S. B. Nair; Artificial Intelligence; 3rd Edition; Tata McGraw Hill.
3. Denis Rothman; Artificial Intelligence By Example: Develop machine intelligence from scratch using real artificial intelligence use cases; Kindle Edition, Packt Publishing Ltd, 2018

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP4003	**Category:** (PCC)
Course: Design and Analysis of Algorithms	**L:** 3Hr**, T:** 0Hr**, P:** 0Hrs**, Per Week, Credits:** 3

Course Objectives:

The objective of this course is to familiarize the prospective engineers with:

1. Techniques of effective problem solving in computing.
2. Analysis of different paradigms of problem solving to solve a given problem in efficient way.

Syllabus

UNIT-I

Algorithm Design Basics and Recurrence Relations

Principles of designing algorithms and complexity calculation, Asymptotic notations for analysis of algorithms, worst case and average case analysis, amortized analysis and it's applications. Recurrence relations and their solutions using substitution, recurrence tree and master theorem methods

UNIT-II

Divide and Conquer technique

Introduction ,Basic strategy, Binary Search, finding Maximum and Minimum elements from array, Quick sort, Merge sort, Strassen's matrix multiplication, Maximum sub-array problem, Closest pair of points problem.

UNIT-III

Greedy technique

Introduction ,Basic strategy, fractional knapsack problem, Minimum cost spanning trees, Huffman Coding , activity selection problem, Scheduling problem with and without deadlines, Find maximum sum possible equal to sum of three stacks, K Centers Problem.

UNIT-IV

Dynamic Programming technique

Introduction ,Basic strategy, difference between Dynamic and Greedy approach, Bellmen ford algorithm, all pairs shortest path algorithm, multistage graphs, optimal binary search trees, traveling salesman problem, String Editing, Longest Common Subsequence problem and its variations.

UNIT-V

Backtracking and Branch and bound techniques

Introduction ,Basic strategy ,N-Queen's problem, Vertex coloring problem, Hamiltonian circuit problem, sum of subset problem, Branch and bound general technique, applications, travelling sales person problem

UNIT-VI

P and NP class

Basic concept, non-deterministic algorithms NP-hard and NP-complete problems, Cook's Theorem, decision and optimization problems, polynomial reduction, , Introduction to Approximation algorithm. ,vertex cover problem, clique cover problem

Course Outcomes

After successful completion of this course, the student will be able to,

1. Use mathematical formulation, complexity analysis and methodologies to solve the recurrence relations for algorithms.
2. Apply Greedy and Divide and Conquer algorithms and their usage in real life examples.
3. Apply Dynamic programming and Backtracking and branch and bound Paradigms to solve the real life problems.
4. Analyze P and NP class problems and formulate solutions using standard approaches.

Text Books

1. Thomas H. Cormen et.al; "Introduction to Algorithms"; 3 Edition; Prentice Hall, 2009.
2. Horowitz, Sahani and Rajasekaram; "Computer Algorithms", Silicon Press, 2008.
3. Brassard and Bratley; "Fundamentals of Algorithms", 1 Edition; Prentice Hall, 1995.
4. Richard Johnsonbaugh, "Algorithms", Pearson Publication, 2003.

Reference books

1. Parag Himanshu Dave, Balchandra Dave, "Design and Analysis of Algorithms" Pearson Education, O'relly publication
2. Richard Johnsonbaugh, "Algorithms", Pearson Publication, 2003.

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP4005	**Category:** (VSEC)
Course: Software Laboratory - II	**L:** 0Hr, **T:** 0Hr, **P:** 2Hrs, **Per Week, Credits:** 1

Course Objectives:

The objective of this course is to impart necessary and practical knowledge of recent Software strategies based on frameworks and development skills required to design real-life web based projects by using advanced object oriented programming by :

1. Designing Enterprise applications by encapsulating an application's business logic.
2. Designing and developing multi-tier web applications
3. Designing applications using existing frameworks like Spring and Hibernate.

Experiments based on:

JDBC Java Database Connectivity (JDBC): The Design of JDBC, Basics of Structured

Query Language, JDBC Configuration, Executing a basic SQL Statement.

Servlet: Handling the Client Request, Generating HTML response, Reading form Data from Servlets: Reading Three Parameters, managing a session.

Java Server Pages (JSP): Invoking Java Code with JSP Scripting Elements, JSP directives and actions, Integrating Servlets and JSP

Spring Framework: Spring Bean Life Cycle, Spring Bean Scope, Basic Bean Wiring

Hibernate: Hibernate Configuration, Hibernate Sessions, Collections Mappings, Hibernate Query Language

Course Outcomes

On successful completion of the course, the student will be able to:

1. Implement Software based database application.
2. Demonstrate server and client side programming using servlets and Java server pages.
3. Perform Dependency Injection using Spring, and create mappings in Hibernate using HQL.

Text Books:

1. M. Deitel, P. J. Deitel, S. E. Santry; Advanced Java 2 Platform HOW TO PROGRAM; Prentice Hall.
2. Cay Horstman, Gary Cornell; Core JAVA Volume-II Advanced Features; 8th Edition.

3. Craig Walls; Spring In Action; 2nd Edition
4. Marty Hall, Larry Brown; Core Servlets and Java Server Pages Volume-1: Core Technologies; 2nd Edition

Reference Books:

1. Jim Keogh; “J2EE:The Complete Reference”; McGraw Hill; Fifth Edition.
2. Spring Framework Documentation https://spring.io/
3. Hibernate Framework architecture Documentation https://hibernate.org/

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code: CDP4006	**Category:** (CEP)
Course: Community Engagement Project	**L:** 0Hrs**, T:** 0Hr**, P:** 4Hr**, Per Week, Credits:** 2

Course Objectives

The objective of Community Engagement Project is to instill a sense of social responsibility amongst the students, empowering them to apply their knowledge and skills to positively impact and contribute to the society.

Execution Plan for the Subject:

- The students will impart their knowledge and skills in the society by identifying the potential needs or identify a society need and address it by building a technical solution.

Course Outcomes

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

1. Propose a community engagement project tailored to address society needs by devising a strategy or solution to address it.
2. Apply technical knowledge or skills towards execution of the proposed solution.
3. Evaluate the effectiveness of the project in addressing community needs.
4. Demonstrate ethical principles, project management skills, team work and communication skills for project completion within the confines of a deadline.

Syllabus for Semester IV, B. Tech. Computer Science & Engineering (Data Science)

Course Code:CDT5001 Course: Database Management Systems
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week Total Credits: 3

Course Objectives

1.To understand the role of a database management system in an organization.
2.To construct simple and advanced database queries using a data language.
3.To understand and apply logical database design principles and database normalization.
4.To recognize the need for transaction management and query processing.

SYLLABUS

UNIT-I Introduction to Database System Concepts and Architecture

Databases and Database Users, Characteristics of the Database Approach, Advantages of Using the DBMS Approach, When Not to Use a DBMS, Data Models, Schemas, and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment.

UNIT-II The Relational Data Model and SQL

Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions, and Dealing with Constraint Violations, SQL Data Definition, Data Types and Constraints, Data Management in SQL, Transforming ER Model into Relational Model.

UNIT-III Database Design and Normalization

Functional Dependencies, Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decomposition, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Other Dependencies and Normal Forms.

UNIT IV Indexing and Hashing

Ordered Indices, B+-Tree Index Files and its Extensions, Static Hashing and Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Some General Issues Concerning Indexing.

UNIT V Query Processing and Optimization

Measures of Query Cost, Query Operation: Selection, Sorting and Join Operation, Transformation of Relational Expressions, Estimating Statistics of Expression Results, Choice of Evaluation Plans.

UNIT VI Transaction Processing, Concurrency Control and Recovery

Introduction to Transaction Processing, Characterizing Schedules Based on Recoverability, Characterizing Schedules Based on Serializability, Two-Phase Locking Techniques for Concurrency Control, Deadlock Handling and Multiple Granularity, Database Recovery Techniques.

Course Outcomes:

On completion of the course the student will be able to
1.Identify the basic concepts and various data model used in database design.
2.Recognize the use of normalization and functional dependency.
3.Understand the purpose of query processing and optimization.
4.Apply and relate the concept of transaction, concurrency control and recovery in database.

Text Books:

1.Abraham Silberschatz, Henry F. Korth and S. Sudarshan; "Database System Concepts" Sixth Edition, Tata McGraw Hill, 2011.

2.Ramez Elmasri and Shamkant Navathe; "Fundamentals of Database Systems", Sixth Edition, Addison Wesley 2011.

Reference Books:

Raghu Ramakrishnan and Johannes Gehrke; "Database Management Systems"; Third Edition; Tata McGraw Hill Publication, 2003.

C. J. Date; "Database in Depth – Relational Theory for Practitioners"; O`Reilly Media, 2005.

Syllabus for Semester V, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDP5001 **Course: Database Management Systems Lab**

L:0 Hrs, T:0 Hrs, P:2 Hrs Per Week Total Credits: 1

Course Objectives

1. To enable students to use DDL, DML and DCL.
2. To prepare students to conceptualize and realize database objects (tables, indexes, views and sequences) and execute SQL queries.
3. To encourage students to design and execute PL/SQL blocks and triggers.

Syllabus:

DBMS lab based on Theory syllabus

Course Outcome

On completion of the course the student will be able to

1. Demonstrate database user administration and authorizations.
2. Execute simple, nested, multiple table, and advanced queries for data retrieval.
3. Construct PL-SQL block structure and Trigger for specific application.
4. Implement various integrity constraints, views, sequences, indices and synonym on database.

Reference Books

1. James Groff, Paul Weinberg and Andy Oppel, SQL - The Complete Reference, 3rd Edition, McGraw Hill, 2017.

Syllabus for Semester V, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT5002 **Course: Machine Learning**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. To introduce the basic concepts and techniques of machine learning.
2. To understand major machine learning algorithms.
3. To identify machine learning techniques suitable for a given problem.

Syllabus

UNIT – I

Concept Learning: The concept learning task, General-to-specific ordering of hypotheses, Version spaces, Inductive bias, Decision Tree Learning, Rule Learning: Propositional and First-Order, Over-fitting, Cross Validation, Experimental Evaluation of Learning Algorithms.

UNIT – II

Instance-Based Learning: K-Nearest neighbor algorithm, Radial basis functions, Case- based learning. Computational Learning Theory: probably approximately correct (PAC) learning, Sample complexity, Computational complexity of training, Vapnik Chervonenkis dimension.

UNIT – III

Artificial Neural Networks: Linear threshold units, Perceptron, Multilayer networks and backpropagation, recurrent networks.

UNIT – IV

Probabilistic Machine Learning: Maximum Likelihood Estimation, MAP, Bayes Classifiers Naïve Bayes, Bayes optimal classifiers, Minimum description length principle.
Bayesian Networks, Inference in Bayesian Networks.

UNIT – V

Expectation Maximization algorithm, preventing over fitting, Gaussian Mixture Models, K- means and Hierarchical Clustering.
Hidden Markov Models, Reinforcement Learning, Support Vector Machines, Ensemble learning: boosting, bagging.

Course Outcomes:

On successful completion of the course, students will be able to:
1.Solve the problems related to the fundamental concepts in machine learning.
2.Apply machine learning algorithms to solve classification, regression and clustering problems.
3.Analyse the strengths and weaknesses of various machine learning approaches.
4.Apply various machine learning models to efficiently solve real-world problems.

Text Books

1. Tom Mitchell; Machine Learning- an Artificial Intelligence Approach, Volume-II; Morgan Kaufmann, 1986.
2. Christopher Bishop, Pattern Recognition and machine learning; Springer Verlag, 2006.

Reference Books

1. Soumen Chakrabarti; Mining the Web: Discovering Knowledge from Hypertext Data, Morgan Kaufmann, 2003.
2. A. K. Jain and R. C. Dubes; Algorithms for Clustering Data; Prentice Hall PTR, 1988.
3. Ethem Alpaydin, Introduction to Machine Learning, PHI.

Syllabus for Semester V, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDP5002 Course: Machine Learning Lab
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week Total Credits: 3

Course Objectives

1. To implement basic machine learning algorithm for solving problem.
2. To understand the usage of datasets in implementing machine learning problems.
3. To learn various modern tools, packages and techniques for machine learning.

Syllabus Technology: Python.

- To implement Find-S algorithm to find maximally specific hypothesis on given dataset.
- To implement Candidate Elimination algorithm on Enjoy Sports dataset and find candidate hypothesis.
- To implement linear regression algorithm on given dataset.
- To implement decision tree classifier on given dataset and display generated tree.
- To Implement KNN algorithm for classification and regression on given datasets.
- To implement the perceptron algorithm for AND, OR and NOR Boolean functions.
- To implement the backpropagation algorithm for machine learning using suitable dataset.
- To implement the Naïve Bayes algorithm on given dataset.
- To implement the support vector machine algorithm on given dataset.

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

1. Implement fundamental concepts used in machine learning algorithms.
2. Implement python programs for various learning algorithms.
3. Apply appropriate machine learning algorithms to various data sets.
4. Apply machine learning algorithms to solve real world problems.

Text Books

1. Tom Mitchell; Machine Learning- an Artificial Intelligence App
roach, Volume-II; Morgan Kaufmann, 1986.
 1. Christopher Bishop, Pattern Recognition and machine learning; Springer Verlag, 2006.

Reference Books

1. Soumen Chakrabarti; Mining the Web: Discovering Knowledge from Hypertext Data, Morgan Kaufmann, 2003.
2. A. K. Jain and R. C. Dubes; Algorithms for Clustering Data; Prentice Hall PTR, 1988
3. Ethem Alpaydin, Introduction to Machine Learning, PHI.

Syllabus for Semester V, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT5003 **Course: Computer Networks**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. To Introduce the fundamental concepts of each layer in the OSI and TCP/IP models.
2. To implement, and troubleshoot network topologies and examine network and transport Layer protocol working.
3. To learn about network security, firewalls, and intrusion detection.
4. To Investigate applications of AI in computer network.

Syllabus

UNIT – I
Data communication Components: Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Data Link Layer: Error Detection and Error Correction - Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ. IEEE 802.3.

UNIT – II
Network Layer: Internet Protocol (IP) – Logical Addressing: IPV4, IP addressing and subnetting, IPV6, ARP, RARP, BOOTP and DHCP–Delivery, Routing protocols.

UNIT – III
Transport Layer: Elements of Transport protocols: Addressing, Connection establishment, Connection release, User Datagram Protocol (UDP), Transmission Control Protocol (TCP). TCP congestion control. Traffic shaping Leaky Bucket and token bucket. Algorithm.

UNIT IV
Application Layer: Domain Name Space (DNS), File Transfer Protocol (FTP), WWW, HTTP, Network security principles and threats, Cryptographic techniques (symmetric/asymmetric encryption), SSL/TLS protocols.

UNIT – V
Firewalls, VPNs, and IPS/IDS (Intrusion Detection and Prevention Systems), Bluetooth.
Load Balancing Optimization using AI/ML, AI-based Threat Hunting in Network Security.

Course Outcomes:

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

1. Implement the computer network protocols with topologies.
2. Apply error detection and correction mechanisms related to data Link Layer and implement Routing algorithm.
3. Analyze TCP protocol related to traffic shaping and routing algorithms.
4. Implement network security mechanisms to protect data and infrastructure.

Text Books

1 "Computer Networks" by Andrew S. Tanenbaum, David J. Wetherall (5th Edition), Pearson Education

2 "Data and Computer Communications" by William Stallings (10th Edition), Pearson Education

3 Data Communication and networking by Behrouz Forouzan (4th Edition) Mc Graw Hill Publication.

4 Cryptography And Network Security by Behrouz Forouzan (3rd Edition) Mc Graw Hill Publication.

Reference Books

1. "Computer Networking: A Top-Down Approach" by James F. Kurose, Keith W. Ross (7th Edition), Pearson Education.
2. "Network Security Essentials" by William Stallings (5th Edition), Pearson Education.

Syllabus for Semester V, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT5004 **Course: Compiler Design**

L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

The main objective of this course is to introduce the fundamental concepts of compiler design and language translation. It aims to develop an understanding of the structure, function, and complexity of modern compilers. The students will learn the various phases of compilation with practical implementation using compiler writing tools.

Syllabus

UNIT I

Introduction to Compilers, Phases of Compiler, Relating Compilation Phases with Formal Systems, Lexical Analysis, tokens, pattern and lexemes, Design of Lexical analyser, Regular Expression, transition diagram, recognition of tokens, Lexical Errors.

UNIT II

Syntax Analysis- Specification of syntax of programming languages using CFG, Top-down parser, design of LL (1) parser, bottom-up parsing technique, LR parsing, Design of SLR, CLR, LALR parsers, Handling Ambiguous Grammars, Applications of the LR Parser.

UNIT III

Syntax directed translation- Study of syntax directed definitions & syntax directed translation schemes, Type and Type Checking, Implementation of SDTS, Intermediate notations, translation of Assignment Statement, controls structures, Array reference.

UNIT IV

Code optimization- machine independent Optimisation, Local optimization techniques, loop optimization- control flow analysis, data flow analysis, Loop invariant computation, Induction variable removal, other loop optimization techniques, Machine-dependent Optimization techniques. Code generation- Problems in code generation, Simple code generator, code generation using labelling algorithm, code generation using gencode algorithm

UNIT V.

Storage allocation & Error Handling**-** Run time storage administration, stack allocation, Activation of Procedures, Storage Allocation Strategies, Garbage Collection, symbol table management, Error handling, Error detection and recovery- lexical, syntactic and semantic, Error recovery in LL & LR Parser

Course Outcomes:

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

1. Exhibit role of various phases of compilation, with understanding of types of grammars and design complexity of compiler.
2. Design various types of parses and perform operations like string parsing and error handling.
3. Demonstrate syntax directed translation schemes, their implementation for different programming language constructs.
4. Implement different code optimization and code generation techniques using standard data structures.

Text Books

1. Aho, Sethi, and Ullman; Compilers Principles Techniques and Tools; Second Edition, Pearson education, 2008.
2. Alfred V. Ahoand Jeffery D. Ullman; Principles of Compiler Design; NarosaPub.House, 1977.
3. VinuV. Das; Compiler Design using Flex and Yacc; PHI Publication, 2008.
4. Manoj B Chandak, Khushboo P Khurana; Compiler Design; Universities Press, 2018.

Reference Books

1. Vinu V. Das; Compiler Design using Flex and Yacc; PHI Publication
2. V. Raghavan; Principles of Compiler Design, McGraw Hill Education (India)

Syllabus for Semester V, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT5004 **Course: Design Patterns**

L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. To learn the fundamentals of software design by referring a catalog of design patterns
2. Demonstrate how to use design patterns to address code development issues.
3. Identify the most suitable design pattern to address a given application design problem.
4. Apply design principles (e.g., open-closed, dependency inversion, substitution, etc).
5. Critique code by identifying and refactoring anti-patterns.

Syllabus

UNIT-I:

Elements of Design Pattern, Describing Design Pattern, Design Pattern Classification, Role of design patterns in software design, Selection and usage of Design Patterns, Example implementation of design pattern using UML, Case Study: Designing a Document Editor.

UNIT-II:

Creational Patterns: Introduction, Role of Creational patterns, Creational Pattern types: Factory method, Abstract Factory, Builder, Prototype, Singleton, Comparative study of creational patterns, and examples based on real life applications.

UNIT-III:

Structural Design Patterns: Introduction, Role of Structural patterns, Encapsulating complex structures to simplify interactions between components, Decoupling Components, Structural Pattern types: Adapter, Bridge, Composite, Decorator, Façade, Proxy, Comparative study of structural patterns, and examples based on real life applications.

UNIT-IV:

Behavioral Patterns-I: Introduction, Role of Behavioral pattern, Encapsulation of Behavior, Behavioral Pattern types: Chain of Responsibility, Template Method, State, Strategy, and Iterator.

UNIT-V:

Behavioral Patterns-II: Effect of single object on set of objects, Analysis of mutual behavior of classes and object's state, Reference control between objects, Behavioral Pattern types: Observer, Mediator, Memento, Interpreter, Comparative study of Behavioral patterns, and examples based on real life applications.

Course Outcomes:

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

1. Analyze the need and ability of design patterns in the software design process.
2. Implement various solutions for creation of objects, their structure and the interaction between objects.

3. Develop a loosely coupled application using design patterns.
4. Analyze the tradeoffs of applying a design pattern to a given problem.

Text Books:

1. Design Patterns: Elements of reusable object-oriented software by Gamma Erich, Helm Richard, Johnson Ralph, and Vlissides John, Pearson Education
2. Design Patterns Explained by Alan Shallowly and James Trott, Addison-Wesley

Reference Books:

1. Pattern's in JAVA Vol-I by Mark Grand, WileyDreamTech.
2. JAVA Enterprise Design Patterns Vol-III by Mark Grand, WileyDreamTech.
3. Head First Design Patterns by Eric Freeman, O'Reilly.

Syllabus for Semester V, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT5004 **Course**: **Software Engineering**

L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. To familiarize students with the fundamentals of software engineering principles and practices.
2. To aid students in designing software systems using structured and object-oriented approaches.
3. To apprise students with different software testing and debugging strategies in building a quality software.
4. To introduce students to the practice of software project management.

Syllabus

Unit-I

The Evolving Role of Software - Software Characteristics, Applications, Principles and Myths; Software Engineering as a Layered Technology; Software Process Framework.

Software Process Models - Waterfall Model, Evolutionary Models, Unified Process Model, Agile Process Models, Extreme Programming (XP), Scrum Model; Requirements Engineering.

Unit-II

Requirements Analysis, Analysis Modeling Approaches; Data Modeling, Object-Oriented Analysis, Scenario-Based Modeling, Flow-Oriented Modeling, Class-based Modeling, Behavioral Model, Design Concepts, The Design Model, Component Level Design, User Interface Design.

Unit-III

Basic concepts of Testing, Software Testing Life Cycle (STLC), Verification and Validation, Unit Testing, Integration Testing, Validation Testing, System Testing, Art of Debugging. White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing, Equivalence Partitioning, Boundary Value Analysis, Web Testing, Test case design, Building, Execution, Automated Testing.

Unit-IV

Software Project management- Plans, Methods and Methodology; Project Success and Failure, Project Evaluation, Cost-benefit evaluation technique, Project Planning & Scheduling.Software Effort Estimation- Albrecht Function Point Analysis, COSMIC Function Point, Cost Estimation, COCOMO Model, Project Scheduling. Software Quality.A Framework for Product Metrics, Metrics for Analysis & Design Models, Metrics for Source Code, Metrics for Testing & Maintenance. Metrics for process & project - Software measurement.

Unit-V

Risk management - Risk strategies, Software risks, Risk identification, Risk refinement, RMMM Risk Response development & Risk Response Control, Risk Analysis. Change Management-Software Configuration Management, SCM Repository, SCM Process, Estimation, Reengineering- Software reengineering, Reverse engineering.

Course Outcomes:

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

1. Elucidate software engineering practices and various process models.
2. Use software design approaches for designing real-time solutions.
3. Demonstrate White Box Testing and Black Box Testing for building bug-free quality software.
4. Integrate software project management practices in software product development.

Text Books

1. Roger S. Pressman and Bruce R. Maxim; Software Engineering – A Practitioner's Approach; Eighth Edition, McGraw Hill; 2015.
2. Ian Somerville; Software Engineering; Seventh Edition; Pearson Education. 2008.

Reference Books

1. Pankaj Jalote; An Integrated Approach to Software Engineering; Third Edition, Springer, 2005.
2. Rajib Mall; Software Project Management, 5th Edition, McGrawHill.
3. David Gustafsan; Software Engineering;Schaum's Series,Tata McGraw Hill,2002.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT6001 Course: Data Warehousing and Business Intelligence
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week Total Credits: 3

Course Objectives

- To understand data warehouse concepts and conduct dimensioning modeling in building data warehouses
- To study the basics of data integration and ETL technology
- To know business intelligence architecture and technologies.
- To study enterprise reporting
- To know future trends in BI

Syllabus:

UNIT-I:

Architecture of DW, OLTP vs OLAP, Dimensional modelling : dimensions and facts, Star and Snowflake schema, pros & cons of the Star / Snowflake schema, Types of dimension and facts.

UNIT-II:

Basics of data preprocessing, concepts of data integration and data transformation, data reduction, data summarization, introduction to data quality, data profiling concepts and applications. Extraction, Transformation, and Load.

UNIT-III:

Space management in data warehouses : B-tree indexes, bitmap indexes, clusters, hash indexes, functional indexes, table partitioning.

UNIT-IV:

Introduction to BI, Leveraging Data and knowledge for BI, BI Components, BI Dimensions, Information Hierarchy, Business Intelligence and Business Analytics. BI Architecture , BI Life Cycle. Role of Data Warehousing in BI, data warehousing building blocks, Metadata in the data warehouse

UNIT-V:

Enterprise Reporting : Introduction to business metrics and KPIs, Basics of Enterprise Reporting: Introduction to enterprise reporting, concepts of dashboards, balanced scorecards, Business Activity Monitoring, Six Sigma. Memento Design pattern and its implementation

Course Outcome:

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

1. Apply the concepts of data warehousing and space management for BI problems.
2. Demonstrate knowledge of data preprocessing techniques.
3. Apply appropriate BI methods to find solutions to business problems

Text Books:

1. Business Intelligence: A Managerial Perspective on Analytics, 3rd Edition, Ramesh Sharda, Dursun Delen, Efaim Turban, Prentice Hall 2013
2. Fundamentals of Business Analytics, R N Prasad and S Acharya, Wiley India.
3. Paulraj Ponnian, - Data Warehousing Fundamentals||, John Willey

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDP6001 Course: Data Warehousing and Business Intelligence Lab
L:0 Hrs, T:0 Hrs, P:2 Hrs Per Week Total Credits: 1

Course Outcomes:

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

1. Implement the ETL process.
2. Design a data warehouse
3. Execute OLAP queries on a data warehouse.
4. Create good reports.

Practical based on syllabus of CDT6001.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT6002 **Course: Deep Learning – I**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. To introduce basic deep learning algorithms.
2. To understand real-world problems which can be solved by deep learning methods.
3. To identify deep learning techniques suitable for a real-world problem.

Syllabus:

UNIT I: Basics of Deep Learning

History of Deep Learning, McCulloch Pitts Neuron, Thresholding Logic, Perceptrons,, Perceptron Learning Algorithm and Convergence, Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons.

UNIT II: Training of Feedforward Neural Networks

Feedforward Neural Networks, Representation Power of Feedforward Neural Networks, Training of Feedforward Neural Networks, Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam

UNIT III: Optimization Algorithm

Activation Function and Initialization Methods: Sigmoid, Tanh, ReLU, Xavier and He Initialization, Regularization: Bias and Variance, Overfitting, Hyperparameters Tuning, L1 and L2 Regularization, Data Augmentation and Early Stopping, Parameter Sharing and Tying.

UNIT IV: Convolutional Neural Network (CNN)

Convolutional Neural Networks, 1D and 2D Convolution, Visualizing Convolutional Neural Networks, Guided Backpropagation.

UNIT V: Recurrent Neural Network (RNN)

Recurrent Neural Networks, Backpropagation Through Time (BPTT), Vanishing and Exploding Gradients, Long Short-Term Memory (LSTM) Cells, Gated Recurrent Units (GRUs). Variants of CNN and RNN : Encoder-Decoder Models, Attention Mechanism, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet.

Course Outcomes:

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

1. **Apply** fully connected deep neural networks to real-world problem-solving scenarios.
2. **Evaluate** the performance of various deep learning models in terms of optimization, bias-variance trade-off, overfitting, and underfitting.
3. **Analyze** the role of convolutional and recurrent neural networks in addressing different real-world problems.
4. **Create** advanced deep learning models by designing variants of CNNs and RNNs tailored to specific applications.

Text Books

1. Sandro Skansi, *Introduction to Deep Learning*, Springer
2. Charu C. Aggarwal, *Neural Networks and Deep Learning: A Textbook*, Springer, 2019
3. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, MIT Press, 2016
4. Dr. S. Lovelyn Rose, Dr. L. Ashok Kumar, Dr. D. Karthika Renuka, *Deep Learning using Python*, Wiley Publication

Reference Books:

1. Bishop, C. M., *Pattern Recognition and Machine Learning*, Springer, 2006
2. Yegnanarayana, B., *Artificial Neural Networks*, PHI Learning Pvt. Ltd., 2009
3. A. Ravindran, K. M. Ragsdell, and G. V. Reklaitis, *Engineering Optimization: Methods and Applications*, John Wiley & Sons, Inc., 2016

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDP6002 **Course**: **Deep Learning – I Lab**

L:0 Hrs, T:0 Hrs, P:2 Hrs Per Week **Total Credits: 1**

Course Objectives:

1.To solve problems in linear algebra, probability, optimization using artificial neurons.
2.To understand the usage of publically available datasets.
3.To use various python packages and tools for deep learning.

Course Syllabus:

Experiments based on CDT6002

Course Outcomes:

On completion of the course the student will be able to

1.Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains.
2.Implement deep learning models in Python using the PyTorch/Tensorflow library and train them with real-world datasets.
3.Analyze and evaluate deep learning model's.

Text Books:

1.Sandro Skansi, Introduction to Deep Learning, Springer
2.Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004.

Reference Books:

1. Francois Chollet, Deep Learning with Python, Manning Publications Co.
2. Golub, G., H., and Van Loan, C., F., Matrix Computations, JHU Press, 2013.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT6003 **Course: Data Analysis and Visualization**

L:1 Hrs, T:0 Hrs, P:0 Hrs Per Week Total Credits: 1

Course Objectives

1. To understand data analytics life cycle for solving challenging business problems.
2. To adopt appropriate statistical procedures for analysis based on goals and nature of data.
3. To employ best practices in data visualization to develop charts, maps tables and other visual representations of data.

Syllabus

UNIT - I:

Introduction and Overview: Importance of analytics and visualization, data preprocessing, Basic Analysis Techniques, Data Analytics Lifecycle and Different Phases

UNIT - II:

Association Rules and Regression Association Rules : Overview, Apriori Algorithm, Evaluation of Candidate Rules, Frequent Itemsets and Rule Generation, Validation and Testing, Diagnostics. Regression: Linear Regression, Logistic Regression, Choice of a Model.

UNIT - III:

Classification and Clustering Clustering : Overview, k-Means, k-Modes, Partitioning around Medoids (PAM), Hierarchical Agglomerative and Density Clustering Methods. Classification: Decision Trees – Overview, Detecting Significant Split, Algorithms and Evaluation; Naïve-Bayes – Bayes' Theorem, Naïve Bayes Classifier, Smoothing; Diagnostics of Classifiers.

UNIT - IV:

Time Series Analysis: Box-Jenkins Methodology, ARIMA (Auto Regressive Integrated Moving Average) Model, Choice of a Model, Overview of ARMAX, Spectral Analysis and GARCH.

UNIT - V :

Data Visualization Understanding: Understanding Data Visualization Principles, Mapping Data onto Aesthetics, Visualizing - Distributions, Proportions, Time Series, Trends and Uncertainty; Commonly used File Formats and Software.

UNIT - VI :

Creating Stories with Data :Why Planning?, Creating Interesting Stories with Data – Reader-driven Narratives, Author-driven Narratives; Perceptions and Presentation Methods, Best Practices in Visualization, Interactive Visualization, Event Listeners and Layouts, Case Studies for Visualization.

Course Outcomes:

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

1. Apply data preprocessing and basic data analysis techniques.
2. Conduct data analytics using scientific methods.
3. Analyze time series data.
4. Create presentations and visualizations.

Text Books

1. David Dietrich, Barry Heller and Beibel Yang, - Data Science and Big Data Analytics – Discovering, Analyzing, Visualizing, and Presenting Data, John Wiley and Sons [EMC Education Services], 2015.
2. Claus O. Wilke, - Fundamentals of Data Visualization – A Primer on Making Informative and Compelling Figures, O'Reilly, 2019.
3. Python: Data Analytics and Visualization, Packt Publishing, 2017.

Reference Books

1. Jiawei Han, Micheline Kamber and Jian Pei, - Data Mining Concepts and Techniques, 3rd edition; Morgan Kaufmann Publishers, 2011.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDP6003 **Course: Data Analysis and Visualization Lab**

L:0 Hrs, T:0 Hrs, P:1 Hrs Per Week **Total Credits: 1**

Course Objectives

1. To apply statistical methods for data analytics to provide business solutions.
2. To develop insights based on analytical results to facilitate better understanding of consumer attitude, perceptions and behavior.
3. To create data visualizations for effective communication to user.

Syllabus

Experiments based on Data Visualization and Analytics Theory Syllabus preferably using R, Python.

Course Outcomes:

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

1. Apply different data preparation techniques.
2. Apply various data analysis techniques.
3. Apply analytics on time series data.
4. Design effective presentations and visualizations.

Reference Books

1.David Dietrich, Barry Heller and Beibel Yang, - Data Science and Big Data Analytics – Discovering, Analyzing, Visualizing, and Presenting Data, John Wiley and Sons [EMC Education Services], 2015.
2.Kyran Dale, - Data Visualization with Python and JavaScript – Scrape, Clean and transform Your Data, O'Reilly, 2016.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDT6004-1 **Course**: **Natural Language Processing**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. To familiarize the concepts and techniques of natural language processing.
2. To learn computational techniques that enable machines to process, understand, and generate human language efficiently.
3. To apply the statistical learning methods and cutting-edge research models to solve natural language processing problems.
4. To integrate natural language processing into real-world systems to develop, evaluate, and enhance applications.

Syllabus:

UNIT I

Introduction to NLP, Definition and Scope, A Brief History, Importance, Challenges, Tasks, Significance, NLP Pipeline and Applications, Morphological Analysis and Generation using Finite State Automata, Finite State Transducer, Hidden Markov model, Viterbi Algorithm, Applications of Tagging.

UNIT II

Lexical Analysis, Part-of-Speech (POS) Tagging, Approaches for POS Tagging, Rule-Based, Stochastic, Hybrid Approach, Taggers Evaluations, Tokenization with NLTK.
Syntax and Parsing, Types of Constituents in Sentences, Context-Free Grammar (CFG), CFG Parsing, Top-Down Parser, Bottom-Up Parser, Shallow Parsing and Chunking, Thematic Roles, Conditional Random Fields, Maximum Likelihood Estimation, Lexical and Probabilistic Parsing, Probabilistic Context Free Grammars, The Probability of a String, Inside-Outside Algorithm, CKY Parsing.

UNIT III

Semantic Analysis, Lexical Vs Compositional Semantic Analysis, Word Senses and Relations, Types of Lexical Semantics, Word Sense Disambiguation, WordNet and Online Thesauri, Word Similarity and Thesaurus Methods, Text Representation, Word Embedding, TF-IDF, Bag of Words, Word2Vec, Skip-gram.Pragmatic Analysis and Discourse, Discourse Phenomena, Coherence and Coreference, Importance of Coreference Relations, Discourse Segmentation, Algorithms for Coreference Resolution.

UNIT IV

N-Gram Language Model, Language Modeling and Chain Rule, Markov Chain in N-Gram Model, Shannon's Method in N-Gram Model, Smoothing Techniques, Extrinsic Evaluation Scheme, Zero Counts Problems, Smoothing Techniques, Laplace (Add-One) Smoothing, Add-k Smoothing, Backoff and Interpolation Smoothing, Good Turing Smoothing, The Transformer, Large Language Models, Language Model Evaluation, Entropy, Perplexity, ROUGE, BLEU.

UNIT V

Major NLP Applications, Information Retrieval Systems, Social Network Analysis, Sentiment Analysis, Information Extraction, Named Entity Recognition, Text Classification, Text Summarization Systems, Machine Translation, Word Alignment, Content Recommendation System, Answering Questions, Applications in Finance, E-Commerce, Travel and Hospitality, Marketing, Insurance, Healthcare, Law, Supply Chain, Telecommunication, Education and Research.

Course Outcomes

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

1. Understand core NLP concepts and techniques.
2. Apply various POS tagging approaches and parsing techniques to analyze sentence structure and utilize probabilistic models for syntactic analysis.
3. Analyze various semantic and pragmatic analysis techniques and discourse phenomena to enhance text representation and understanding.
4. Implement N-Gram language models and Transformer-based models for effective language modeling and text generation.
5. Design and develop innovative NLP solutions to address real-world challenges across industries like finance, healthcare, e-commerce, education and research.

Textbooks

1. Daniel Jurafsky and James H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models, Third Edition, 2025, https://web.stanford.edu/~jurafsky/slp3.
2. Raymond ST. Lee, Natural Language Processing: A Textbook with Python Implementation, Springer Nature Singapore Pte Ltd. 2024, ISBN: 978-9819919987.
3. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, and Harshit Surana, Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems, O'Reilly Media, Inc., USA, First edition, 2020, ISBN: 978-1492054054.
4. Dipanjan Sarkar, Text Analytics with Python: A Practitioner's Guide to Natural Language Processing, Second Edition, Apress Media, LLC, California, 2019, ISBN: 978-1484243534.

Reference Books

1. Natural Language Processing with Python: From Basics to Advanced Projects, Second Edition, 2024, Cuantum Technologies LLC. Plano, ISBN: 979-8894968483.
2. Jyotika Singh, Natural Language Processing in the Real World: Text Processing, Analytics, and Classification, First edition, 2023, CRC Press is an imprint of Taylor & Francis Group, LLC, ISBN: 978-1003264774.
3. Gerhard Paaß and Sven Giesselbach, Foundation Models for Natural Language Processing: Pre-trained Language Models Integrating Media, Artificial Intelligence: Foundations, Theory, and Algorithms, Springer Nature Switzerland Pte Ltd. 2022, ISBN: 978-3031231896.
4. Lewis Tunstall, Leandro von Werra, and Thomas Wolf, Natural Language Processing with Transformers: Building Language Applications with Hugging Face, O'Reilly Media, Inc., USA, Revised First edition, May 2022, ISBN: 978-1098136796.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDP6004-1 **Course**: **Natural Language Processing**

L:0 Hrs, T:0 Hrs, P:2 Hrs Per Week Total Credits: 1

Course Objectives

1.To familiarize the concepts and techniques of natural language processing.
2.To learn computational techniques that enable machines to process, understand, and generate human language efficiently.
3.To apply the statistical learning methods and cutting-edge research models to solve natural language processing problems.
4.To integrate natural language processing into real-world systems to develop, evaluate, and enhance applications.

Syllabus:

1.Experiments based on the above syllabus of CDT6004 .
2.Few lab sessions shall be conducted using virtual lab platforms to enhance learning experiences and accessibility.
3.Utilise Hugging Face, Stanford, Kaggle, MIT OpenCourseWare, OpenAI Learning, Fast.ai, AllenNLP like repositories and platforms.

Textbooks

1.Daniel Jurafsky and James H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models, Third Edition, 2025, https://web.stanford.edu/~jurafsky/slp3.
2.Raymond ST. Lee, Natural Language Processing: A Textbook with Python Implementation, Springer Nature Singapore Pte Ltd. 2024, ISBN: 978-9819919987.
3.Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, and Harshit Surana, Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems, O'Reilly Media, Inc., USA, First edition, 2020, ISBN: 978-1492054054.
4.Dipanjan Sarkar, Text Analytics with Python: A Practitioner's Guide to Natural Language Processing, Second Edition, Apress Media, LLC, California, 2019, ISBN: 978-1484243534.

Reference Books

1.Natural Language Processing with Python: From Basics to Advanced Projects, Second Edition, 2024, Cuantum Technologies LLC. Plano, ISBN: 979-8894968483.
2.Jyotika Singh, Natural Language Processing in the Real World: Text Processing, Analytics, and Classification, First edition, 2023, CRC Press is an imprint of Taylor & Francis Group, LLC, ISBN: 978-1003264774.
3.Gerhard Paaß and Sven Giesselbach, Foundation Models for Natural Language Processing: Pre-trained Language Models Integrating Media, Artificial Intelligence: Foundations, Theory, and Algorithms, Springer Nature Switzerland Pte Ltd. 2022, ISBN: 978-3031231896.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)
Course Code: CDT6004-2 **Course: Blockchain Technology**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

This course aims to provide a survey on blockchain and the topics around such as history of blockchain, cryptography it uses, Bitcoin and other currencies, consensus algorithms, smart contracts, Ethereum, scalability and various use cases.

Syllabus

UNIT I

Blockchain Introduction: Blockchain Technology Mechanisms & Networks, Blockchain Origins, Blockchain Objectives, Blockchain Users & Adoption, Blockchain Challenges, P2P Systems, Hash Pointers and Data Structures, Blockchain Transactions

Unit II

Consensus Mechanism:Permissioned Blockchain, Permisionless Blockchain , Different Consensus Mechanism- Proof of Work, Proof of Stake, Proof of Activity, Proof of Burn, Proof of Elapsed Time, Proof of Authority, Proof of Importance.

Unit III

Cryptography Fundamentals:Encryption, Digital Signatures, Public-Key Cryptography, Private Key Cryptography, Distributed Denial-of-Service (DDoS) Attack, 51% Attack, Double spending problem, Merkel Tree, Security Threats to Blockchain Technology

Unit IV

Crypto currency and Wallet:Types of Wallet, Desktop Wallet, App based Wallet, Browser based wallet, Metamask, Creating a account in Metamask, Use of faucet to fund wallet, transfer of cryptocurrency in metamask.

Unit V

Smart contract and Ethereum:Overview of Ethereum, Writing Smart Contract in Solidity, Remix IDE , Different networks of ethereum, understanding blocks in blockchain, compilation and deployment of smart contracts in Remix

Course Outcomes:

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

1. Realize importance of blockchain technology and consensus mechanism
2. Identifying the security risks and challenges associated with blockchain technology
3. Implement browser based wallets and smart contracts in Remix IDE
4. Recognize the importance of blockchain security in various enterprise applications.

Text Books

1.Mastering Blockchain: Third Edition by Imran Bashier, Packt Publishing, 2020, ISBN: 9781839213199.

Reference Books

1. Blockchain: Blueprint for a New Economy by Melanie Swan, Oreilly Publication
2. Mastering Ethereum, by Andreas M. Antonopoulos, Gavin Wood

3.Bitcoin and Cryptocurrency Technologies (Princeton textbook) by Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)
Course Code: CDP6004-2 **Course: Blockchain Technology Lab**
L:0Hrs, T:0 Hrs, P:2 Hrs Per Week **Total Credits: 1**

Course Objectives

This course aims to provide hands-on experience with blockchain development tools and frameworks, and will be able to use them to build and test their own blockchain projects.

Syllabus:

Experiments based on the above syllabus CDT6004-2

Course Outcome:

On completion of the course the student will be able to

1. Realize different blockchain tools and framework
2. Implement smart contracts for the development of enterprise applications
3. Apply different wallets for the deployments of smart contracts

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)
Course Code: CDT6004-3 **Course: Distributed and Parallel Computing**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. To introduce the fundamental concepts of parallel and distributed computing, including types of parallelism, performance metrics, and architectural models.
2. To equip students with practical programming skills using parallel computing APIs (OpenMP, MPI, CUDA, Spark) for solving data- and compute-intensive problems.
3. To develop the ability to analyze, design, and optimize algorithms for multi-core, many-core, and distributed memory architectures using appropriate models and tools.

Syllabus

UNIT I:

Overview of parallel processing landscape: why and how, types of parallelism, Flynn's taxonomy and brief overview of parallel architectures, practical demonstration of CCR as an example HPC center. Basic concepts in parallel processing: formal definition of parallelism, concepts of work, speedup, efficiency, overhead, strong and weak scaling (Amdahl's law, Gustafson's law), practical considerations using parallel reduction and parallel prefix.

UNIT II:

Multi-core programming: shared memory and shared address space, data and task parallelism, summary of available APIs (OpenMP, TBB). Practical examples using OpenMP and parallel merge sort, pointer jumping, parallel BFS and basic linear algebra.

UNIT III:

Distributed memory programming: Message Passing Interface, latency + bandwidth model, distributed hashing, sample sort, parallel BFS, and basic linear algebra (matrix-vector, maxtrix-matrix products) with relation to graph algorithms.

UNIT IV:

Higher-level programming models: stateless programming in Map/Reduce, Apache Spark and fault-tolerance via Resilient Distributed Datasets. Triangle counting, connected components, single source shortest path using Spark

UNIT V:

Many-core programming: SIMD parallelism and massively parallel GPGPU accelerators. Brief overview of available APIs (OpenACC, oneAPI, OpenCL, SYCL). Programing GPUs in NVIDIA CUDA: data movement and organization, 1D/2D stencils, parallel prefix, matrix-matrix product.

Course Outcomes:

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

1. Explain the principles of parallel and distributed computing, including types of parallelism, Flynn's taxonomy, and performance laws (Amdahl's, Gustafson's).
2. Implement parallel algorithms using OpenMP, MPI, CUDA, and Spark to solve computational and data-intensive problems.
3. Analyze the scalability, efficiency, and overhead of parallel algorithms and systems.
4. Design and optimize parallel solutions for real-world data science tasks, ensuring correctness, performance, and code quality.

Text Books

1.Grama, A., Gupta, A., Karypis, G., & Kumar, V. (2003). Introduction to Parallel Computing (2nd ed.). Pearson Education. ISBN: 978-0201648652.

2.McCool, M., Reinders, J., & Robison, A. (2012). Structured Parallel Programming: Patterns for Efficient Computation. Morgan Kaufmann. ISBN: 978-0124159938.
3.Karau, H., Konwinski, A., Wendell, P., & Zaharia, M. (2020). Learning Spark: Lightning-Fast Big Data Analysis (2nd ed.). O'Reilly Media. ISBN: 978-1492050049.

Reference Books

1.Kirk, D. B., & Hwu, W. W. (2016). Programming Massively Parallel Processors: A Hands-on Approach (3rd ed.). Morgan Kaufmann. ISBN: 978-0128119860.

2.Quinn, M. J. (2003). Parallel Programming in C with MPI and OpenMP. McGraw-Hill Education. ISBN: 978-0072822564.d Edition, O'Reilly Media, 2021.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)
Course Code: CDP6004-3 Course: Distributed and Parallel Computing Lab
L:0 Hrs, T:0 Hrs, P:2 Hrs Per Week Total Credits: 1

Course Objectives

1. To introduce the fundamental concepts of parallel and distributed computing, including types of parallelism, performance metrics, and architectural models.
2. To equip students with practical programming skills using parallel computing APIs (OpenMP, MPI, CUDA, Spark) for solving data- and compute-intensive problems.
3. To develop the ability to analyze, design, and optimize algorithms for multi-core, many-core, and distributed memory architectures using appropriate models and tools.

Syllabus

Experiments based on the above syllabus of CDT6004

Text Books

1.Grama, A., Gupta, A., Karypis, G., & Kumar, V. (2003). Introduction to Parallel Computing (2nd ed.). Pearson Education. ISBN: 978-0201648652.

2.McCool, M., Reinders, J., & Robison, A. (2012). Structured Parallel Programming: Patterns for Efficient Computation. Morgan Kaufmann. ISBN: 978-0124159938.

3.Karau, H., Konwinski, A., Wendell, P., & Zaharia, M. (2020). Learning Spark: Lightning-Fast Big Data Analysis (2nd ed.). O'Reilly Media. ISBN: 978-1492050049.

Reference Books

1.Kirk, D. B., & Hwu, W. W. (2016). Programming Massively Parallel Processors: A Hands-on Approach (3rd ed.). Morgan Kaufmann. ISBN: 978-0128119860.

2.Quinn, M. J. (2003). Parallel Programming in C with MPI and OpenMP. McGraw-Hill Education. ISBN: 978-0072822564.d Edition, O'Reilly Media, 2021.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)
Course Code: CDT6005-1 **Course: Data Science for Health Care**
L:0 Hrs, T:0 Hrs, P:2 Hrs Per Week **Total Credits: 1**

Course Objectives

1.Introduce healthcare data sources (EHR, medical imaging, biomedical signals, and genomics) and their challenges.
2.Learn machine learning and deep learning techniques for healthcare applications.
3.Develop skills in medical image analysis, signal processing, genomic data analysis, and clinical NLP and address bias, fairness, and ethical considerations in AI-driven healthcare.

Syllabus

Unit I:

Fundamentals of AI in Healthcare & EHR Analytics :Overview of AI in Healthcare, Introduction to Machine Learning & Deep Learning in Healthcare, AI Applications: Disease Diagnosis, Prognosis, Decision Support Systems, Electronic Health Records (EHR) Analytics: Components, Benefits, Barriers to Adoption, Coding Systems & Phenotyping Algorithms.

Unit II:

Image Analysis in Healthcare:Biomedical Imaging Modalities (X-ray, CT, MRI, Ultrasound), Image Processing Techniques: Object Detection, Image Segmentation, Image Registration, Feature Extraction for Disease Classification.

Unit III:

Biomedical Signal Analysis**:**Types of Biomedical Signals (ECG, EEG, EMG), ECG Signal Analysis: Signal Processing: Denoising, Feature Extraction, Multivariate Biomedical Signal Analysis, Cross-Correlation Analysis

Unit IV:

Genomic Data Analysis:Genomic Data Generation & Quality Control, Analysis Techniques : Normalization, Differential Expression Detection, Clustering & Classification in Genomics, Genome Sequencing Analysis & Personalized Medicine, Public Tools for Genomic Data Analysis (e.g., NCBI, Ensembl)

Unit V:

NLP for Clinical Text & Ethical Considerations in AI:Clinical Text Mining & Report Analysis, Core NLP Components in Healthcare, Challenges in Processing Clinical Reports, Applications of NLP in Healthcare (ICD Coding, Chatbots, Summarization) , Bias & Ethics in Healthcare AI, Algorithmic Bias in Medical AI Models, Explainability & Interpretability in Healthcare AI

Course Outcomes:

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

1. Analyze and preprocess healthcare data from various sources for AI applications.
2. Develop machine learning models for disease prediction, risk assessment, and diagnostics.
3. Apply AI techniques to medical imaging, signal processing, genomics, and NLP.
4. Evaluate AI models in healthcare for interpretability, fairness, and ethical concerns.

Text Books and Reference Books

1. Chandan K. Reddy, Charu C. Aggarwal, "Healthcare Data Analytics", CRC Press
2. Sergio Consoli, Diego Reforgiato Recupero, Milan Petković; Data Science for Healthcare: Methodologies and Applications; Springer, 2019.
3. Eric Topol MD, Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)
Course Code: CDT6005-2 **Course: Data Science for Genomics**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. Develop a foundational understanding of genomic data science, including bioinformatics tools, machine learning, and deep learning techniques for analyzing genomic datasets.
2. Explore the integration of computational methods with genomics to enable applications in functional genomics, personalized medicine, and genome-wide association studies (GWAS).

Syllabus

Unit I:

Introduction to Genomics & Bioinformatics:Introduction to DNA, RNA, and Proteins, Genome Organization and Functional Genomics, Next-Generation Sequencing (NGS) Technologies, Bioinformatics Databases: NCBI, Ensembl, UCSC Genome Browser, KEGG, Genome Annotation and Comparative Genomic

Unit II:

Genomic Data Processing & Analysis:Data Formats in Bioinformatics: FASTA, FASTQ, VCF, BAM, GFF, Sequence Alignment Algorithms: BLAST, Needleman-Wunsch, Smith-Waterman, Genomic Variant Calling: GATK, SAMtools, BCFtools, Quality Control & Preprocessing of NGS Data (FASTQC, Trimmomatic), Cloud Computing for Genomics (Google Genomics, AWS, Terra.bio)

Unit III:

Statistical & Machine Learning Approaches in Genomics:

Statistical Foundations: Bayesian Inference, Maximum Likelihood Estimation (MLE), Dimensionality Reduction: PCA, t-SNE, UMAP for single-cell RNA sequencing (scRNA-seq), Supervised Learning: Gene Expression Classification (SVM, Random Forest), Unsupervised Learning: Clustering of genes (K-Means, Hierarchical Clustering), Feature Selection & Biomarker Discovery in Genomics

Unit IV:

Deep Learning for Genomic Data:

Introduction to Deep Learning: ANN, CNN, RNN, CNNs for DNA Sequence Analysis & Motif Detection, Transformer Models in Genomics (e.g., DNABERT), Autoencoders & Variational Autoencoders (VAE) for DNA Data Augmentation, Hands-on Implementation using TensorFlow/PyTorch

Unit V:

Genome-Wide Association Studies (GWAS) & Variant Analysis:

Understanding GWAS: Identifying SNPs associated with traits, Population Genetics & Linkage Disequilibrium (LD), Statistical Methods in GWAS: Logistic Regression, Chi-square test, Disease Risk Prediction & Polygenic Risk Scores (PRS), Case Study: GWAS in Cancer Research & Personalized Medicine.

Course Outcomes:

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

1. Demonstrate the ability to process, analyze, and interpret genomic sequences using computational techniques.
2. Apply machine learning and deep learning models for tasks such as gene expression analysis, mutation detection, and disease risk prediction.
3. Develop and implement high-throughput genomic data analysis using tools like GATK, SAMtools, BCFtools.
4. Perform statistical analysis and visualization of genomic data, including dimensionality reduction, clustering, and GWAS studies.

Text Books

1. Bioinformatics and Functional Genomics (3rd Edition), Jonathan Pevsner, Cambridge University Press, 2015.

2. Deep Learning for the Life Sciences, Bharath Ramsundar, Peter Eastman, Patrick Walters, Vijay Pande, O'Reilly Media, 2019.

Reference Books

1. Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, Richard Durbin, Sean R. Eddy, Anders Krogh, Graeme Mitchison, Cambridge University Press, 1998.

2. Bioinformatics Algorithms: An Active Learning Approach (2nd Edition, Vol. I), Phillip Compeau, Pavel Pevzner, Active Learning Publishers, 2015.

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)
Course Code: CDT6005-3 **Course: Data Science for Marketing**
L:3 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 3**

Course Objectives

1. Introduce marketing data sources (CRM, web analytics, social media, customer transactions) and their challenges.
2. Learn machine learning and deep learning techniques for marketing applications.
3. Develop skills in customer segmentation, demand forecasting, sentiment analysis, and marketing campaign optimization while addressing bias, fairness, and ethical considerations in AI-driven marketing.

Syllabus

Unit I: Fundamentals of AI in Marketing & CRM Analytics

Overview of AI in Marketing, Introduction to Machine Learning & Deep Learning in Marketing, AI Applications: Customer Insights, Personalization, Recommendation Systems, Customer Relationship Management (CRM) Analytics: Components, Benefits, Challenges, Data Collection & Preprocessing in Marketing: Customer Behavior Data, Clickstream Data, Transactional Data

Unit II: Customer Segmentation & Sentiment Analysis

Customer Lifetime Value (CLV) and Churn Prediction, Market Segmentation Techniques: RFM Analysis, K-Means Clustering, Text Analytics for Sentiment Analysis, Social Media & Review Data Processing

Unit III: Demand Forecasting & Recommendation Systems

Regression Models for Sales Forecasting, Time Series Forecasting Methods (ARIMA, Prophet), AI-Powered Recommendation Systems: Collaborative vs. Content-Based Filtering, Personalization Strategies in Digital Marketing, Case Study: AI-Driven Pricing Optimization

Unit IV: Marketing Campaign Analytics & A/B Testing

Key Marketing KPIs (CAC, ROAS, Conversion Rate, CTR), A/B Testing & Multivariate Testing for Campaign Optimization, Attribution Models (First-Touch, Last-Touch, Multi-Touch), Web and Social Media Analytics Tools (Google Analytics, Facebook Insights), Case Study: Measuring the ROI of AI-Driven Marketing

Unit V: NLP in Marketing & Ethical Considerations in AI

NLP for Customer Reviews & Chatbots, Text Summarization & Topic Modeling in Marketing, AI Bias in Marketing: Fairness in Ad Targeting & Personalization, Ethical Issues in AI-Driven Consumer Behavior Analytics, Explainability & Interpretability of AI Models in Marketing

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

1. Analyze and preprocess marketing data from various sources for AI applications.
2. Develop machine learning models for customer segmentation, demand forecasting, and recommendation systems.
3. Apply AI techniques to campaign optimization, sentiment analysis, and marketing strategy development.
4. Evaluate AI models in marketing for interpretability, fairness, and ethical concerns.

Text Books and Reference Books

1. Foster Provost & Tom Fawcett, "Data Science for Business", O'Reilly Media
2. Eric Siegel, "Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die", Wiley
3. Stephen Sorger, "Marketing Analytics: Strategic Models and Metrics", Pearson

Syllabus for Semester VI, B. Tech. Computer Science and Engineering (Data Science)

Course Code: CDP6007 **Course**: **MINI PROJECT**
L:4 Hrs, T:0 Hrs, P:0 Hrs Per Week **Total Credits: 2**

Course Objectives

The objective of the Mini Project is to let the students map and utilize the technical knowledge acquired in the previous semesters to solve a real-world problem through team effort.

Course Outcomes:

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

1. Identify and finalize the problem statement by investigating various domains and society needs.
2. Perform requirement analysis and design methodology for solving the identified problem.
3. Apply programming techniques and modern tools for the development of the solution.
4. Apply ethical principles, project management skills and demonstrate the ability to work in teams for project development within the confines of a deadline.
5. Communicate technical information employing written reports and presentations.