DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

SCHEME & SYLLABUS
FOR
BACHELOR OF TECHNOLOGY (B.Tech)

COMPUTER SCIENCE & ENGINEERING

(Data Science)

(3rd - 8th Semester)

(With effect from 2024-25 to 2028-2029)

Vision and Mission of DSU

VISION

- To be a centre of excellence in education, research & training, innovation & entrepreneurship and to produce citizens with exceptional leadership qualities to serve national and global needs.

MISSION

- To achieve our objectives in an environment that enhances creativity, innovation and scholarly pursuits while adhering to our vision.

Vision and Mission of School of Engineering

VISION

- Transform lives through excellence in engineering education, research and innovation with an emphasison sustainability, inclusive technologies and globalneeds.

MISSION

- Design and deliver contemporary engineering curricula to address regional and global needs while emphasizing ethics, values, integrity and regional relevance.
- Carryout high impact academic research, industry projectsand innovation activities with active student engagement toadvance science and engineering knowledge and state-of-the-art industry practices.
- Develop regional and national leaders to advance the society and economy.

Vision and Mission of Dept. of CSE (Data Science)

VISION

- To produce Engineers for Industry and Society in the field of Computer Science and Engineering (Data Science) by providing excellence in Education, Research and Entrepreneurship with a focus on sustainable solutions to fulfill global needs.

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

MISSION

- Impart quality technical education, critical thinking and sustainable learning practices in the domain of Computer Science and Engineering (Data Science) with ethical values and leadership qualities.
- Inculcate Interdisciplinary Research and Innovation by establishing Industry-Academia collaboration to solve critical problems.
- Prepare graduates to become Ethical Data Science practitioners to contribute in data driven global society.

Definitions / Descriptions

Definition of Credit:	
1 Hour Lecture (L) Per Week	01 Credit
1 Hour Tutorial (T) Per Week	01 Credit
1 Hour Practical (P) Per Week	0.5 Credit
1 Hour Project (J) Per Week	0.5 Credit

Course code and Definition:	
BSC	Basic Science Courses
ESC	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management Courses
IPCC	Integrated Professional Core Course
PCC	Professional Core Courses

PEC	Professional Elective Courses
OEC	Open Elective Courses
SEC	Skill Enhancement Courses
UHV	Universal Human Value Course
PROJ	Project Work
INT	Internship

Implementation of National Education Policy (NEP) 2020 for the B.Tech students of Batch 2024-2028

The implementation of Curriculum follows NEP 2020 and addresses the following features and categories of courses:

1. Student Centric flexible curriculum.
2. Interdisciplinary Courses,
3. Multi-disciplinary Courses,
4. Ability Enhancement Courses,
5. Skill Enhancement Courses,
6. Value Added Courses,
7. Product Design and Development,
8. Internship (Rural Internship, Industry Internship, Research/Development Internship), and
9. Multiple Exit and Multiple Entry
    - Certificate in Engineering after completion of first year.
    - Diploma in Engineering after completion of second year.
    - Advanced Diploma in Engineering after completion of third year.
    - Degree in Engineering after completion of fourth year
    - Degree in Engineering after completion of fourth year

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

III SEMESTER													
Sl. N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					L	T	P	J	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
1	BSC	24DS2301	Probability and Statistics	MAT	3	0	0	0	03	60	40	100	**3**
2	IPCC	24DS2302	Data Structures	CSE	3	0	2	0	05	60	40	100	**4**
3	PCC	24DS2303	Computer Organization and Architecture	DS	3	0	0	0	03	60	40	100	**3**
4	PCC	24DS2304	Discrete Mathematics and Graph Theory	CSE	3	0	0	0	03	60	40	100	**3**
5	PCC	24DS2305	Web Technologies	DS	3	0	0	0	03	60	40	100	**3**
6	IPCC	24DS2306	Database and Management System	DS	3	0	2	0	05	60	40	100	**4**
7	SEC	24DS23XX	Skill Enhancement Course – I	DS	1	0	2	0	03	100	--	100	**2**
8	CTS	24DS2307	Cognitive & Technical Skills-III	ALL	-	-	-	-	-	-	-	-	**P/F**
			Total		**19**	**0**	**06**	**0**	**25**	**460**	**240**	**700**	**22**

Sl. No	Course Code	Skill Enhancement Course – I
1	24DS2308	OOPS with JAVA
2	24DS2309	MOOC

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

IV SEMESTER													
Sl. N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				
					L	T	P	J	Duration in Hours	CIE Marks	SEE Marks	Total Marks	Credits
1	BSC	24DS2401	Transforms and Numerical Techniques	MAT	3	0	0	0	03	60	40	100	**3**
2	IPCC	24DS2402	Design and Analysis of Algorithms	CSE	3	0	2	0	05	60	40	100	**4**
3	IPCC	24DS2403	Data Science and Visualization	DS	3	0	2	0	05	60	40	100	**4**
4	PCC	24DS2404	Enterprise Data Warehousing	DS	3	0	0	0	03	60	40	100	**3**
5	IPCC	24DS2405	Machine Learning	DS	3	0	2	0	05	60	40	100	**4**
6	PROJ	24DS2406	Minor Project -1	DS	0	0	0	4	04	60	40	100	**2**
7	SEC	24DS24XX	Skill Enhancement Course – II	DS	1	0	2	0	03	100	--	100	**2**
8	CTS	24DS2407	Cognitive & Technical Skills-IV	ALL	-	-	-	-	-	-	-	-	**P/F**
			Total		**16**	**00**	**08**	**04**	**28**	**460**	**240**	**700**	**22**

Sl. No	Course Code	Skill Enhancement Course – II
1	24DS2408	Business Intelligence Tools
2	24DS2409	MOOC

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

V SEMESTER													
S.N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					L	T	P	J	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
1	PCC	24DS3501	Computer Network	DS	3	0	0	0	03	60	40	100	**3**
2	PCC	24DS3502	Software Engineering Principles	DS	3	0	0	0	03	60	40	100	**3**
3	IPCC	24DS3503	Cloud Data Engineering	DS	3	0	2	0	05	100	--	100	**4**
4	IPCC	24DS3504	Deep learning essentials	DS	3	0	2	0	05	60	40	100	**4**
5	IPCC	24DS3505	Operating System	DS	3	0	2	0	05	60	40	100	**4**
6	PEC	24DS35XX	Professional Elective Course – I	DS	3	0	0	0	03	60	40	100	**3**
7	CTS	24DS3506	Cognitive & Technical Skills- V	ALL	-	-	-	-	-	-	-	-	**P/F**
			Total		**18**	**00**	**06**	**0**	**24**	**400**	**200**	**600**	**21**

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

	VI SEMESTER												
Sl. N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					L	T	P	J	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
1	IPCC	24DS3601	Big Data Analytics	DS	3	0	2	0	05	60	40	100	**4**
2	IPCC	24DS3602	Theory of Computation	DS	3	1	0	0	05	60	40	100	**4**
3	PCC	24DS3603	Principles of DevOps & MLOps	DS	3	0	2	0	05	100	--	100	**4**
4	OEC	24OEXXXX	Open Elective – I	DS	3	0	0	0	03	60	40	100	**3**
5	PEC	24DS36XX	Professional Elective Course – II	DS	3	0	0	0	03	60	40	100	**3**
6	PEC	24DS36XX	Professional Elective Course – III	DS	3	0	0	0	03	60	40	100	**3**
7	PROJ	24DS3604	Minor Project - 2	DS	0	0	0	4	04	60	40	100	**2**
8	CTS	24DS3605	Cognitive & Technical Skills-VI	ALL	-	-	-	-	-	-	-	-	**P/F**
			Total		**18**	**0**	**06**	**04**	**28**	**460**	**240**	**700**	**23**

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

	VII SEMESTER												
Sl. No.	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					L	T	P	J	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
1	HSMC	24DS4701	Innovation and Entrepreneurship	DS	2	0	0	0	02	60	40	100	**2**
2	OEC	24OEXXXX	Open Elective – II	Any	3	0	0	0	03	60	40	100	**3**
3	PEC	24DS47XX	Professional Elective Course – IV	DS	3	0	0	0	03	60	40	100	**3**
4	PEC	24DS47XX	Professional Elective Course – V	DS	3	0	0	0	03	60	40	100	**3**
5	PROJ	24DS4702	Capstone Project Phase-I	DS	0	0	0	8	08	100	--	100	**4**
			Total		**11**	**0**	**0**	**8**	**19**	**340**	**160**	**500**	**15**

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

	VIII SEMESTER												
S. N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					L	T	P	J	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
1	PROJ	24DS4801	Capstone Project Phase-II	DS	0	0	0	28	03	60	40	100	14
2	INT	24DS4802	Internship	DS	0	0	6	0	03	--	100	100	03
			Total		**0**	**0**	**6**	**28**	**06**	**60**	**140**	**200**	**17**

I -20 II -20 III -22 IV – 22 V -21 VI -23 VII – 15 VIII – 17

NOTE: Total Credits (I-Sem to VIII Sem) = 160 credits.

NOTE: Professional elective courses Domain-wise

SN	Domain		PROFESSIONAL ELECTIVE COURSES								
			PEC-I		PEC-II		PEC-III		PEC-IV		PEC-V
			5th Semester		6th Semester				7th Semester		
		Course Code	Course Name	Course Code	Course Name	Course Code	Course Name	Course Code	Course Name	Course Code	Course Name
Domain 1	Emergent AI Technologies	24DS3507	Introduction to AI	24DS3606	Natural Language Processing	24DS3610	Generative AI Systems	24DS4703	Agentic AI	24DS4707	Artificial General Intelligence
Domain 2	Adaptive Data Science	24DS3508	Mining of Massive Dataset	24DS3607	Massive Graph Analysis	24DS3611	Data-Driven Recommendations	24DS4704	Immersive Data Science	24DS4708	Streaming Analytics
Domain 3	Domain-Driven Analytics	24DS3509	Sensor Analytics	24DS3608	Vision Analytics	24DS3612	Bio-Informatics	24DS4705	Supply Chain Logistics	24DS4709	Risk Analytics for Finance
Domain 4	Core CSE	24DS3510	Optimization Techniques	24DS3609	Compiler Design and system	24DS3613	Data Privacy and Cyber security	24DS4706	Soft Computing Techniques	24DS4710	Responsible AI

OPEN ELECTIVE COURSES			
Open Elective -I		**Open Elective-II**	
Course Code	**Course Name**	**Course Code**	**Course Name**
23OE0014	Statistical Computing for Data Science	23OE0015	Data Exploration Using R

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

Syllabus

PROBABILITY AND STATISTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hrs
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

1. This Course will enable students to: **Apply** statistical principles and probability concepts to solve complex problems in real- world scenarios involving uncertainty and randomness.
2. **Evaluate** and select appropriate probability distributions and statistical techniques to analyze and interpret data accurately in various applications.
3. **Justify** the use of estimation methods and hypothesis testing techniques for drawing meaningful inferences about population parameters.
4. **Analyze** and interpret sample test results for different statistical relationships, such as means, variances, correlation coefficients, regression coefficients, goodness of fit, and independence, to make informed decisions.
5. **Identify** sample tests using appropriate statistical procedures to investigate the significance of observed data and communicate findings effectively.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability

to evaluate, generalize, and analyze information rather than simply recall it.

7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1: Probability	**09 Hours**
Definitions of Probability, Addition Theorem, Conditional Probability, Multiplication Theorem,Bayes' Theorem of Probability	

Unit 2: Random Variables and their Properties and Probability Distributions	**09 Hours**
Discrete Random Variable, Continuous Random Variable, Joint Probability Distributions TheirProperties, Probability Distributions: Discrete Distributions: Binomial, Poisson Distributions and their Properties; Continuous Distributions: Exponential, Normal, Distributions and their Properties.	

Unit 3: Estimation and testing of hypothesis	**06 Hours**
Sample, Populations, Statistic, Parameter, Sampling Distribution, Standard Error, Un-Biasedness, Efficiency, Maximum Likelihood Estimator, Notion & Interval Estimation.	

Unit 4: Sample Tests-1	**07 Hours**
Large Sample Tests Based on Normal Distribution, Small Sample Tests: Testing Equality ofMeans, Testing Equality of Variances, Test of Correlation Coefficient	

Unit 5: Sample Tests-2	**08 Hours**
Test for Regression Coefficient; Coefficient of Association, 2 – Test for Goodness of Fit, Test forIndependence.	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Apply** the principles of probability to solve complex problemsin various real-world scenarios.	L2 & L3

2	**Solve** and **compare** different probability distributions, including discrete and continuous random variables, in order to make informed decisions and predictions.	L2 & L3
3	**Apply** statistical estimation techniques, such as maximumlikelihood estimation and interval estimation, to draw meaningful inferences about population parameters from sample data.	L3
4	**Examine** hypothesis testing methods, including large and small sample tests, to assess the significance of observed dataand draw valid conclusions.	L3
5	**Analyze** statistical relationships and perform sample tests toassess the Equality of means in different populations, Correlation coefficients between variables to determine the strength and direction of the relationship. Independence of variables using appropriate statistical tests to assess theabsence of any relationship.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2		2				1					
CO2	3	2	2		2				1					
CO3	3	2	2						1					
CO4	3	2	2		2				1					
CO5	3	2	2		2				1					

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Probability & Statistics for Engineers and Scientists, Walpole, Myers, Myers, Ye. PearsonEducation.

Reference Books:

1. Probability, Statistics and Random Processes T. Veerarajan Tata McGraw – Hill
2. Probability & Statistics with Reliability, Queuing and Computer Applications, Kishor S. Trivedi,Prentice Hall of India ,1999

Activity Based Learning:

1. Tools like Python programming, R programming can be used which helps student to develop a skill to analyze the problem and providing solution.
2. Regular Chapter wise assignments/ Activity/Case studies can help students to have critical thinking, developing

an expert mind set, problem-solving and teamwork.

Following are Assignments/ Activities Can be carried out using either R programming language or Python Programming or excel solver.

1. There are n people gathered in a room. What is the probability that at least 2 of them will have the same birthday? (Use excel solver, R Programming, Python Programming)
   a. Use simulation to estimate this for various n., and Produce Simulation Graph.
   b. Find the smallest value of n for which the probability of a match is greater than 0.5.
   c. Explore how the number of trials in the simulation affects the variability of our estimates.
2. Case Study 1: Customer Arrivals at a Coffee Shop
   a. A coffee shop wants to analyze the number of customer arrivals during its morning rush hour (7:00 AM to 9:00 AM). The shop has been recording the number of customer arrivals every 15 minutes for the past month.
   b. Data: The data consists of the number of customer arrivals recorded at the coffee shop during each 15-minute interval for the past month.
   c. Here is a sample of the data:

Time Interval	stomer Arrivals
00 AM - 7:15 AM	6
15 AM - 7:30 AM	4
30 AM - 7:45 AM	9
45 AM - 8:00 AM	7
00 AM - 8:15 AM	5
15 AM - 8:30 AM	8
30 AM - 8:45 AM	10
45 AM - 9:00 AM	6

Analyze the customer arrivals and determine the probability distribution that best
fits the data. Specifically, explore both discrete and continuous probability distributions, including the binomial, Poisson, exponential, and normal distributions.

3. Case Study 2: Comparing the Performance of Two Groups

a. Suppose you are a data analyst working for a company that manufactures a new energy drink. The marketing team conducted a promotional campaign in two different cities (City A and City B) to determine the effectiveness of the campaign in increasing sales. The sales data for a random sample of customers in each city was collected over a week. Your task is to compare the average sales between the two cities and test whether there is a significant difference in the variance of sales.

b. Data: Let's assume the following sample data for the number of energy drinks sold in each city:

City A: [30, 28, 32, 29, 31, 33, 34, 28, 30, 32]

City B: [25, 24, 26, 23, 22, 27, 29, 30, 26, 24]

perform a two-sample t-test to test the equality of means and a test for equality of variances using Python's SciPy library.

4. Case study 3: testing independence between two categorical variables.

a. Data: Sample of 100 employees, and each employee is classified as either Male or Female. They were asked to rate their job satisfaction on a scale of 1 to 5, where 1 represents low satisfaction and 5 represents high satisfaction. The data is as follows:

Employee	Gender	Job Satisfaction
1	Male	4
2	Female	3
3	Male	2
4	Female	5
...	...	...
100	Female	4

b. Test for independence between gender and job satisfaction, use the chi-squared test in R.

E Resources:

1. https://nptel.ac.in/courses/106104233
2. https://nptel.ac.in/courses/117103067
3. https://nptel.ac.in/courses/103106120
4. https://www.coursera.org/learn/probability-intro#syllabus
5. https://nptel.ac.in/courses/111104073

Laboratory Experiments: (not Mandatory)

DATA STRUCTURES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code:	**Credits**: 04
Hours /Week: 03 Hours	**Total Hours**: 39(Th)+26(P) Hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the basic approaches for analyzing and designing data structures.
2. **Introduce** dynamic memory allocation and C language concepts required for building data structures.
3. **Develop** essential skills to construct data structures to store and retrieve data quickly and **efficiently**.
4. **Utilize** different data structures that support different sets of operations which are suitable for various applications.
5. **Explore & implement** how to insert, delete, search, and modify data in any data structure- Stack, Queues, Lists, Trees
6. **Develop** applications using the available data structure as part of the course for mini project.

Teaching-Learning Process (General Instructions)

1. These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.
2. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
3. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
4. Show ***Video/animation*** films to explain functioning of various concepts.
5. Encourage ***Collaborative*** (Group Learning) Learning in the class.
6. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
7. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.

8. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
9. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

Introduction : Introduction to Data Structure, Classification, C Structure and Union, C Pointers, Array Definition, Representation, Operations (Insertion, Deletion, Search and Traversal), Two/Multidimensional Arrays, Sparse matrix. **TB1: 1.1, 2.2, 2.5 ; TB2: 1.1, 1.2,1.3.1-1.3.4; RB1: 5.1 – 5.12, 6.4**

Unit 2:	**08 Hours**

Introduction to ADT : Stack: Definition, Array Representation of Stack, Operations on Stacks.

Applications of Stack: Expression evaluation, Conversion of Infix to Postfix, Infix to Prefix Recursion, Tower of Hanoi

Queue: Definition, Representation of Queues, Operations of Queues, Circular Queue.

Applications of Queue: Job Scheduling, A Maze Problem **TB1: 3.1,3.2, 3.3,3.4,3.5 ; TB2: 2.1, 2.2, 2.3, 3.2, 3.3**

Unit 3:	**08 Hours**

Dynamic Data Structures : Linked List: Types, Representation of Linked Lists in Memory. Traversing, Searching, Insertion & Deletion from Linked List. Circular List, Doubly Linked List, Operations on Doubly Linked List (Insertion, Deletion, Traversal).

Applications: Stack & Queue Implementation using Linked Lists.

Case Study: Josephus problem. **TB2: 4.2,4.3,4.5**

Unit 4:	**08 Hours**

Trees : Basic Terminology, Binary Trees and their representation, Complete Binary Trees, Binary Search Trees, Threaded Binary Trees, Operations on Binary Trees (Insertion, Deletion, Search & Traversal).

Applications: Expression Evaluation **Case Study:** Game Tree. **TB2: 5.5.3,5.5.4,5.6**

Unit 5:	**07 Hours**

Efficient Binary Search Trees:

Optimal Binary Search Trees, AVL Trees, Red Black Trees, Splay Trees.

Case Study: B Trees. TB1: 10.1,10.2,10.3,10.4, 11.2

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Apply the concepts of pointers, arrays, structures, and unions to address real-world problems and implement the concept in C programming language.	L3
2	Utilize stacks and queue data structures to solve problems such as infix to postfix, infix to prefix conversions, the Towers of Hanoi puzzle, job scheduling and maze navigation.	L3
3	Implement and manipulate singly linked lists, doubly linked lists, and circular linked lists, executing operations such as insertion, deletion, and traversal.	L3
4	Understand the concepts of binary trees, binary search trees, and threaded binary trees, and their associated operations.	L2
5	Understand advanced binary tree structures includes optimal binary search trees, AVL trees, Red-Black trees, and Splay trees.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO2	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO3	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO4	-	1	2	-	3	-	-	-	-	-	-	-	2	2
CO5	-	1	2	-	-	-	-	-	-	-	-	-	2	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Ellis Horowitz, Susan Anderson-Freed, and Sartaj Sahni, "Fundamentals of Data structures in C", 2nd Edition, Orient Longman, 2008.
2. A.M. Tannenbaum, Y Langsam, M J Augentien "Data Structures using C", 1st Edition, Pearson, 2019.

Reference Books:

1. Brian. W. Kernighan, Dennis. M. Ritchie, "The C Programming Language", 2nd Edition, Prentice-Hall, 1988.
2. Gilbert & Forouzan, "Data Structures: A Pseudo-code approach with C", 2nd Edition, Cengage Learning,2014.

3. Jean-Paul Tremblay & Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, McGraw Hill, 2013.
4. R.L. Kruse, B.P. Learly, C.L. Tondo, "Data Structure and Program design in C", 5th Edition, PHI ,2009.

Activity Based Learning:

1. Real world problem solving using group discussion.
2. Role play E.g., Stack, Queue, etc.,
3. Demonstration of solution to a problem through programming.
4. Flip class activity E.g., arrays, pointers, dynamic memory allocation, etc.,

E Resources:

2. https://nptel.ac.in/courses/106102064
3. https://www.coursera.org/learn/data-structures?specialization=data-structures-algorithms
4. https://www.udemy.com/topic/data-structures/free/
5. https://www.mygreatlearning.com/academy/learn-for-free/courses/data-structures
6. https://cse01-iiith.vlabs.ac.in/
7. https://kremlin.cc/k&r.pdf

Laboratory Experiments: (not Mandatory)

Following are experiments to be carried out using either C programming language.

1. To Implement C programs to perform array operations.
2. To determine the validity of a 9x9 Sudoku board (application of 2-dimensional array).
3. To store, retrieve and update the elements in structures (structures and pointers to structures).
4. To implement stack using linked list.
5. To implement a queue data structure using a singly linked list.
6. To implement a singly linked list and its operations.
7. To implement a doubly linked list and its operations.
8. To create a circular queue using a circular linked list data structure
9. To implement binary tree traversal techniques.

OPEN-ENDED EXPERIMENTS

1. Design a web browser history tracker in C. Implement a stack data structure to keep track of visited URLs. Create

functions to push new URLs onto the stack as users visit websites and pop URLs when users navigate backward in their browsing history.

2. Imagine you are responsible for designing a queue-based system to manage the queue of regular customers waiting to purchase cinema tickets at a popular movie theatre. Your system should ensure fair and efficient ticket sales for all customers. When a customer's arrive at the cinema, they join the queue. Each customer is represented by his name, age (for record-keeping), and number of tickets needed. When a customer reaches the front of the queue, they are served by the ticketing agent. Implement a ticket sale process where the agent provides the customer with the requested ticket(s).

COMPUTER ORGANIZATION AND ARCHITECTURE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

This Course will enable students to:

1. Translate the elements of digital logic functions to digital system abstractions using Verilog.
2. Illustrate simplification of Boolean expressions using Karnaugh
3. Model combinational logic circuits for arithmetic operations and logical operations
4. Understand the Architecture and programming of ARM microprocessor.
5. Develop program using Arm instruction set and appreciate the advanced features provided in the ARM
6. Understand the exception handling techniques & study in detail the concept of instruction level parallelism and concepts of pipelining.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

Introduction: Number System- Binary, Hexa Decimal, Octal and its conversion. Canonical Notation - SOP & POS forms, Minimization of SOP and POS forms.

Arithmetic Circuits and Verilog Modelling

Adders: Half adder, full adder, Ripple carry adder, parallel adder /subtractor, fast adders-CLA, comparator- 2 bit. Simplification using K-Maps

Introduction to Verilog, Syntax of Verilog coding, Modelling styles in Verilog, Verilog Operators, Test bench for simulation

Text Book-1: Chapter 1: 1.2 to 1.4, Chapter 2: 2.6

Text Book-2: Chapter 5: 5.2, 5.3.3, 5.4,5.5.2, 5.5.3

Text Book-3: Chapter 1: 1.1, 1.2.2, 1.3.1, 1.3.2, 1.3.3, 1.4.2, 1.5.1.2, 1.5.2.2, 1.5.3.2, 1.5.4.2, 1.6.2

Unit 2:	**07 Hours**

Combinational Circuit Building: Multiplexers 4:1, 8:1, decoders 3:8, 2:4, demultiplexers 1:4, encoders 8:3, 4:2, code converters- B to G and G to B- Simplification using K-Maps

Verilog for combinational circuits, if else, case-casex, casez, for loop, generate.

Text Book-2: Chapter 6: 6.1, 6.2, 6.3, 6.4, 6.6

Unit 3:	**06 Hours**

An Overview of Computing Systems: History of Computers, The Computing Device.

The ARM7TDMI Programmers' Model: Introduction, Data types, Processor Modes, Registers, Program Status Registers, The vector Table.

Assembler Rules and Directives: Structure of Assembly Language UNITs, Registers, Directives and Macros.

Loads, Stores and Addressing: LODS and STORES instructions, Operand Addressing, ENDIANNES

Text Book-4: 1.1 to 1.3; 2.1 to 2.3; 4; 5.3, 5.4, 5.5

Unit 4:	**08 Hours**

Constants and Literal Pools: The ARM Rotation Scheme, Loading Constants and address into Registers

Logic and Arithmetic: Flags and their Use, Compare instructions, Data Processing Instructions

Loops and Branches: Branching, Looping, Conditional Execution, Straight-Line Coding

Subroutines and Stacks: Stack, Subroutines, Passing parameters to subroutines.

Text Book-4: 6.1 to 6.4; 7.1 to 7.4; 8.2 to 8.5; 13.1 to 13.4

Unit 5:	**10 Hours**

Exception Handling: Interrupts, Error Conditions, Processor Exception Sequence, The Vector Table, Exception Handlers, Exception Priorities, Procedures for Handling Exceptions.

Text Book-4: 18.1 to 18.4; 14.1 to 14.8

Pipelining: Basic and Intermediate Concepts:

Introduction, The Major Hurdle of Pipelining, How Pipelining Implemented, what makes Pipelining hard to Implement, Extending the MIPS Pipeline to Handle Multicycle Operations, The MIPS R4000 Pipeline, Crosscutting Issues, score boarding Technique.

Text Book-5: C.1 to C.7

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Interpret Boolean Expressions of digital design in simplified form	L2
2	Build the various elements of digital logic system with Verilog and Construct Combinational logic circuits	L3
3	**Interpret** the knowledge of the internal architecture and organization of ARM microprocessors to understand their components and functionalities.	L2
4	**Apply** the instruction set of ARM Microprocessor by writing Assembly language programs.	L3
5	**Understand** the various exception handling techniques.	L2
6	**Apply** the concept of instruction-level parallelism and understand the principles of Pipelining techniques.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3		2										2	
CO2	3	1	3		1								2	
CO3	3	1	1										2	
CO4	3	1	1		1							1	2	
CO5	3	2	1		1								2	

CO6	3	2	1										2	

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. M. Morris Mano Michael D. Ciletti , "Digital Design with an Introduction to the Verilog HDL", 6th Edition, Pearson Education, 2014.
2. Stephen Brown, Zvonko Vranesic, "Fundamentals of Digital Logic with Verilog design", McGraw Hill, 2014.
3. Nazein M. Botros, "HDL programming (VHDL and Verilog)", Dreamtech Press, 2006
4. William Hohl, "ARM Assembly Language", 2nd Edition, CRC Press, 2009.
5. John L Hennessy, David A Patterson, "Computer Architecture, A Quantitative Approach", 6th Edition, Morgan Kaufmann publishers, 2019.

Reference Books:

1. John M Yarbrough, "Digital Logic Applications and Design", Thomson Learning, 2014.
2. Donald D. Givone, "Digital Principles and Design", McGraw Hill, 2015.
3. Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis", Pearson Education, 2016.
4. David A Patterson, John L Hennessy, "Computer Organization and Design", 4th Edition, Morgan Kaufmann publishers, 2010.
5. Steve Furber, "ARM System-on-chip Architecture", 2nd Edition, Pearson Publications, 2000.
6. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", 5th Edition, Tata McGraw Hill, 2002.

Activity Based Learning:

1. Design problem solving and Programming using group discussion. E.g., Traffic light controller,Digital Clock, Elevator.
2. Mini project implementation using Assembly Language Programming. Demonstration of solution to a problem through programming

E Resources:

1. https://nptel.ac.in/courses/117105080
2. https://archive.nptel.ac.in/courses/106/105/106105165/
3. https://www.udemy.com/topic/arm-cortex-m/
4. https://www.edx.org/school/armeducation
5. https://onlinecourses.nptel.ac.in/noc22_cs93/preview

Laboratory Experiments: (not Mandatory)

DISCRETE MATHEMATICS AND GRAPH THEORY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

This Course will enable students to:

1**. Learn** the set theoretic concept and its application in theory of computation.

2. **Determine** the concepts of mathematical induction, recursive relations and their application.

3. **Illustrate** the association of functions, relations, partial ordered set and lattices with problems related to theoretical computer science and network models.

4. **Discuss** the basics of graph theory and its application in computer networks. Learn the concepts of counting techniques and its application

Teaching-Learning Process (General Instructions)

1. These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.
2. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
3. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
4. Show ***Video/animation*** films to explain functioning of various concepts.
5. Encourage ***Collaborative*** (Group Learning) Learning in the class.
6. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
7. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
8. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to

solve them.

9. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
SET THEORY: Sets and subsets, Operations on Sets: Basic set operations, algebraic properties of sets, The Addition Principle **RELATIONS AND ITS PROPERTIES:** Relations and their properties, N-Ary Relations and their applications, Representing relations. **Textbook – 2: 1.1, 1.2 ; Textbook – 1: 7.1., 7.2, 7.3**	
Unit 2:	**06 Hours**
RELATIONS AND ORDER RELATIONS: Closure of relations, Equivalence Relations, Partial Orderings, Functions, The Growth of Functions. **Self-Study:** Transitive Closure and Warshall's Algorithm. **Textbook – 1: 7.4., 7.5, 7.6, 3.2**	
Unit 3:	**08 Hours**
MATHEMATICAL INDUCTION AND RECURSION: Mathematical Induction, Recurrence Relations: Rabbits and the Fibonacci Numbers, The Tower of Hanoi, Code word Enumeration, Solving Linear Recurrence Relations **Self-Study:** Basic Connectives and Truth Tables **Textbook-1: 4.1;6.1, 6.2;1.1**	
Unit 4:	**09 Hours**
GRAPH THEORY: Graphs and Graph Models. Graph Terminology and Special Types of Graphs: Basic Terminology, Some Special Simple Graphs, Bipartite Graphs, Complete Bipartite Graphs. Representing Graphs and graph isomorphism: Adjacency lists, Adjacency Matrices, Incidence Matrices, Connectivity: Paths, Connectedness in Undirected and Directed Graphs, Vertex and Edge connectivity and their applications. **Textbook-1: 8.1, 8.2, 8.3, 8.4**	
Unit 5:	**08 Hours**
GRAPHS AND ITS APPLICATIONS: Euler and Hamilton Paths and their applications, Planar Graphs and their Applications, Graph Coloring and its applications. **Textbook-1: 8.5, 8.7, 8.8**	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Identify** the membership of the Set, Relations and make use of basic Algebraic properties.	L3
2	**Examine** the steps involved in Mathematical Induction and Linear recurrence-related problems.	L3
3	**Construct** different types of graphs based on the properties and the real-time applications of graph theoretical concepts.	L3
4	**Analyze** the methods for optimizing the solution for graph coloring problem, Eulerian and Hamilton circuits/planes.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1	2					1	1	1		2	1	0
CO2	3	3	2					1	1	1		2	1	0
CO3	3	3	3					1	1	1		1	1	0
CO4	3	3	3					1	1	1		1	1	0
CO5	3	1	2					1	1	1		2	1	0

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Kenneth H. Rosen, "Discrete Mathematics and its applications", Tata McGraw Hill, 2003.
2. Bernard Kolman, Robert C. Busby, Sharon Ross, "Discrete Mathematical Structures", 3rdEdition, PHI 2001.

Reference Books:

1. Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", IV Edition, Pearson Education, Asia, 2002.
2. J. P. Tremblay, R. Manohar, "Discrete Mathematical Structures with applications to computer Science", Tata McGraw Hill, 1987.
3. J K Sharma, "Discrete Mathematics", 3rd edition, 2013, Macmillan India Ltd.

Activity Based Learning:

1. Real world problem solving and puzzles using group discussion.

2. Demonstration of solution to a problem using graph theory.

E Resources:

1. Discrete Mathematics with Algorithms by M. O. Albertson, J. P. Hutchinson – J 1988, Wiley.
2. Discrete Mathematics for Computer Science, Gary Haggard, John Schlipf, Sue Whitesides, Thomson Brooks/Cole, 2006.
3. http://ocw.mit.edu/courses/mathematics/
4. http://www.nptelvideos.in/2012/11/discrete-mathematical-structures.html
5. http://cglab.ca/~discmath/notes.html
6. https://www.cs.odu.edu/~toida/nerzic/content/web_course.html

Laboratory Experiments: (not Mandatory)

WEB TECHNOLOGIES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the fundamentals of frontend web technologies using HTML 5 and CSS3
2. **Apply** Cascading Style Sheets and XHTML to the idea of a web application.
3. **Understand** the principles of client-side programming and understand how to use JavaScript to implement them in order to create dynamic web sites. **Usage** of wide variety of testing techniques in an effective and efficient manner
4. **Implement** the principles of serverside programming using Node.js, Mongo dB
5. **Apply** the Node.js framework -Express.JS to create web applications faster and smarter

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**04 Hours**

INTRODUCTION TO HTML and CSS:
Introduction to HTML and HTML5 - HTML Syntax, Semantic Markup, Structure of HTML Documents - Formatting and Fonts -Commenting Code – Anchors – Backgrounds, Lists – Tables, Introduction HTML Forms, Form Control Elements.
Introduction to CSS - What is CSS, CSS Syntax, Location of Styles, Selectors, and The Cascade: How Styles Interact, The Box Model, CSS Text Styling, : Layout, Normal Flow, Positioning Elements

Unit 2:	**04 Hours**

PYTHON WEB INTERFACES USING FLASK, STREAMLIT, AND GUI APIS:
Introduction to Python Web Interfaces: Client-Server Model - Flask vs Streamlit vs Tkinter/PyWebIO-Installing Flask, Streamlit –**Core Flask - Routing, Templates, Forms** : - URL Routing and Views - Using render_template() - HTML + Jinja2 Templates - Handling POST/GET. Flask - Form Handling & Static Conten -Flask-WTF for form validation Handling Form Submission - Adding CSS/JS files. **Streamlit Basics:** Streamlit Architecture - Widgets: text_input, button, slider - Layouts and Interactive elements - Markdown Integration. **Streamlit -** Data Apps & Visualizations: - Uploading files - DataFrames display - Charts using line_chart, bar_chart. **GUI Programming in Python:** Tkinter - Windows, Buttons, Entries - OR PyWebIO: Input/output, form UIs

Unit 3:	**06 Hours**

JAVASCRIPT:
Client-Side Scripting, What is JavaScript and What can it do?, JavaScript Design Principles, Where does JavaScript Go?, Syntax, JavaScript Objects, The Document Object Model (DOM), JavaScript Events, Forms, Program Control, Functions in JavaScript.

Unit 4:	**06 Hours**

NODE JS:
Introduction to NodeJS, Set up Dev Environment, Node JS Modules, Node Package Manager, File System, Events, Database connectivity using Mongo DB.
ANGULAR JS: Power Features of AngularJS, MVC Architecture: Conceptual Overview, Setting up the Environment, The Anatomy of an AngularJS app, First Application.

Unit 5:	**06 Hours**

EXPRESS .JS

Introducing Express: Basics of Express, Express JS Middleware: Serving Static Pages, Listing Directory Contents, Accepting JSON Requests and HTML Form Inputs.

REACT JS Basics & Features, Setup and Hello World Application, Components and Props, Function and Class Components.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Develop proficiency in creating structured web pages using HTML3 elements, formatting, fonts, images, hyperlinks, lists, tables, and forms.	L3
2	Gain a thorough understanding of CSS3 for styling web pages, including selectors, properties, fonts, colors, and the CSS box model.	L3
3	Acquire skills in JavaScript for client-side scripting, including control statements, arrays, functions, regular expressions, and DOM events.	L2
4	Understand the fundamentals of Node.js, Angular.js including UNIT usage, file system operations, event handling, and database connectivity with MongoDB.	L2
5	Gain proficiency in building web applications with React, Express.js, including serving static pages, handling JSON requests, listing directory contents, and managing cookies.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3		2										2	
CO2	3		3										2	
CO3	3	3	1										2	
CO4	3	3	1										2	
CO5	3	3	1										2	

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Robert W. Sebesta , "Programming the World Wide Web", 7th Edition, Pearson Education, 2008.
2. Basarat Ali Syed," Beginning Node.js ",Apress ,2014

Reference Books:

1. Web Technology: A Developer's Perspective",N.P. Gopalan,J. Akilandeswari, Published October 31, 2014 by PHI

Learning, ISBN 9788120350069 (ISBN10: 8120350065).
2. "Foundation HTML5 with CSS3: A Modern Guide and Reference", Craig Cook, Jason Garber, Charles Brown October 11, 2012 by Apress, ISB 9781430238768 (ISBN10: 1430238763).
3. Head First JavaScript Programming: A Brain-Friendly Guide 1st Edition, by Eric Freeman, Elisabeth Robson, Publisher O'Reilly Media, May 6, 2014

Activity Based Learning:

1. Frontend Development
2. Database Management
3. Deployment and DevOps

E Resources:

1. https://www.geeksforgeeks.org/html/
2. https://www.geeksforgeeks.org/css/
3. https://www.geeksforgeeks.org/javascript/

Laboratory Experiments: (not Mandatory)

1. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.
2. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to 10 and outputs HTML text that displays the resulting values in an HTML table format.
3. Write a JavaScript code that displays text "TEXT-GROWING" with increasing font size in the interval of 100ms in RED COLOR, when the font size reaches 50pt it displays "TEXT-SHRINKING" in BLUE color. Then the font size decreases to 5pt.
4. Develop and demonstrate a HTML3 file that includes JavaScript script that uses functions for the following problems:
    a. Parameter: A string
    b. Output: The position in the string of the left-most vowel
    c. Parameter: A number
    d. Output: The number with its digits in the reverse order
5. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and email id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
6. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
7. Write a PHP program to display a digital clock which displays the current time of the server.
8. Write the PHP programs to do the following:
    a. Implement simple calculator operations.
    b. Find the transpose of a matrix.
    c. Multiplication of two matrices.
    d. Addition of two matrices.
9. Write a PHP program to sort the student records which are stored in the database using selection sort.

DATABASE MANAGEMENT SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code:	**Credits**: 04
Hours /Week: 03 Hours	**Total Hours**: 39 (L)+ 26(P) Hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

This Course will enable students to:

1. Acquire the concept of databases and Relational Model., Entity-Relationship Model and relational model for creating and designing databases for the real-world scenario.
2. Develop queries to extract data from the databases using a structured query language.
3. Draw ER Diagrams and Optimize the Database design using Normalization Concepts.
4. Understand the fundamental concepts of NoSQL databases and their advantages over traditional relational RDBMS
5. Learn to design, implement, and manage scalable data solutions using MongoDB

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**09 Hours**
RELATIONAL DATABASES Purpose of Database System, Views of data, Introduction to relational databases, Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, RDBMS- Overview of the SQL Query Language, SQL Data Definition, Basic Structure of SQL, Basic Operations of DDL, DML.	
UNIT 2:	**08 Hours**
RELATIONAL QUERY LANGUAGE Aggregate Functions , Nested Subqueries , Join Expressions , Views , Transactions , SQL Data Types and Schemas, Authorization, Accessing SQL from a Programming Language, Functions and Procedures ,Triggers, Cursor.	
UNIT 3:	**06 Hours**
DATABASE DESIGN Entity, Relationship model, E-R Diagrams, Complex Attribute, Mapping Cardinalities, ER-to-Relational Mapping, Features of Good Relational Designs, Decomposition Using Functional Dependencies, Non loss Decomposition – First, Second, Third Normal Forms, Dependency Preservation, Boyce/Codd Normal Form, Multi-valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.	
UNIT 4:	**09 Hours**
INTRODUCTION TO NOSQL,MONGODB Fundamentals of NoSQL, Definition and evolution of NoSQL databases, Differences between NoSQL and SQL databases, Types of NoSQL databases: Key-Value, Document, Column-Family, Graph, CAP Theorem (Consistency, Availability, Partition Tolerance), Overview of MongoDB , Introduction to MongoDB and its features, CRUD Operations in MongoDB, Creating, reading, updating, and deleting documents, Querying data with filters, projections, and sorting.	
UNIT 5:	**08 Hours**
ADVANCED MONGODB FEATURES Data Replication and Sharding-Setting up replica sets for high availability, Implementing sharding for horizontal scaling, Data distribution and shard key selection, Hands-on: Configuring replication and sharding in MongoDB, Aggregation Framework - Introduction to MongoDB Aggregation Pipeline, Aggregation stages: $match, $group, $sort, $project, $lookup.	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level

1	Understand the basic concepts of database management system and Relational Model	L2
2	Utilize SQL concepts to build and manipulate relational databases for a given schema.	L3
3	Apply normalization techniques in designing the relational database	L3
4	Demonstrate proficiency in performing CRUD operations, indexing, and basic data modelling using MongoDB	L3
5	Analyze advanced MongoDB functionalities such as aggregation pipelines, indexing strategies, replication, and sharding	L4

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**2**
CO2	**1**	**2**	**3**	**-**	**-**	**-**	**-**	**-**	**2**	**1**	**-**	**1**	**-**	**3**
CO3	**1**	**1**	**2**	**-**	**-**	**-**	**-**	**-**	**2**	**1**	**-**	**1**	**-**	**3**
CO4	**1**	**1**	**2**	**-**	**3**	**-**	**-**	**-**	**-**	**-**	**-**	**1**	**-**	**1**
CO5	**1**	**2**	**3**	**-**	**3**	**-**	**-**	**-**	**2**	**1**	**-**	**1**	**-**	**3**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Textbooks:

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, “Database System Concepts”, Seventh Edition, McGraw Hill, 2020.
2. MongoDB: The Definitive Guide" by Shannon Bradshaw, Eoin Brazil, and Kristina Chodorow, O'Reilly Media, ISBN: 978-1491954462

Reference Books:

1. Raghu Ramakrishnan and Johannes Gehrke, “Database Management Systems”, Third Edition, McGraw-Hill, 2003.
2. C.J. Date, A. Kannan, S. Swamynatham: “An Introduction to Database Systems”, Eight Edition, Pearson Education, 2012.

Activity Based Learning:

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solution to a problem through programming.

E Resources:

1. https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-what-is-database-management-system
2. **https://www.mongodb.com/resources/basics/database-management-system**
3. **https://www.oracle.com/in/database/what-is-database/**
4. **https://onlinecourses.swayam2.ac.in/cec19_cs05/preview**
5. **https://onlinecourses.nptel.ac.in/noc19_cs46/preview**

Laboratory Experiments: (not Mandatory)

1. Database Table Creation and Manipulation with key constraints - Create a database table and add constraints such as primary key, unique, check, and not null. Insert, update, and delete rows using SQL DDL and DML commands.
2. Querying with Conditions and Aggregates - Query the database tables using various WHERE clause conditions and implement aggregate functions like SUM, AVG, COUNT, etc.
3. Subqueries and Joins - Perform queries involving subqueries and simple join operations to retrieve data from multiple tables also implement natural joins, equijoins, and outer joins in your queries.
4. User Defined Functions and Stored Procedures - Write and execute userdefined functions and stored procedures in SQL for various operations.
5. Transactions and Control Commands - Execute complex transactions and understand the use of Data Control Language (DCL) and Transaction Control Language (TCL) commands.
6. SQL Triggers - Write SQL triggers for handling insert, delete, and update operations on a database table.
7. Develop a program to implement cursor
8. NoSQL Database Operations - Create and manage document based data using CRUD and Aggregation commands using MongoDB.
9. Implement a Replication and Sharding Setup using MongoDB

OOPS with JAVA

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code:	**Credits**: 02
Hours /Week: 03 Hours	**Total Hours**: 13 + 26 Hrs
L–T–P–J: 1–0-2	

Course Learning Objectives:

This Course will enable students to:

1. Introduce the basics of data mining, including statistical modeling, machine learning, and computational approaches.
2. Learn to extract relevant features from data for improved model performance.
3. Study the stream data model and techniques for sampling, filtering, and counting in data streams.
4. Understand market-basket analysis and algorithms for discovering frequent item sets.
5. Understand eigenvalues, eigenvectors, PCA, and SVD for reducing data dimensionality.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**07 Hours**
Introduction to Object-Oriented Programming (OOP) and Java **Basics of Object-Oriented Programming:** Principles of OOP: Encapsulation, Inheritance, Polymorphism, and Abstraction, Benefits of OOP over procedural programming **Introduction to Java Programming Language:** History and features of Java, Java Development Kit (JDK) and Java Runtime Environment (JRE), Writing, compiling, and executing a simple Java program **Java Basics:** Variables, data types, and operators, Control structures: if-else, switch, loops (for, while, do-while),Arrays and Strings	
Unit 2:	**06 Hours**
Advanced Object-Oriented Concepts and Java Libraries **Classes and Objects:** Creating classes and objects, Constructors and destructors, Static members and methods **Inheritance and Polymorphism:** Types of inheritance in Java, Method overriding and overloading, Abstract classes and interfaces **Exception Handling:** Types of exceptions, Try-catch block, finally clause, Custom exceptions **Collections Framework:** Introduction to Collections, List, Set, and Map interfaces, Iterating through collections using iterators and for-each loop	
Unit 3:	**06 Hours**
Lab Experiments for Object-Oriented Programming with Java **Lab 1: Basic Java Program** Write, compile, and run a simple Java program that prints "Hello, World!". Experiment with different data types and basic operations. **Lab 2: Control Structures** Implement a Java program using various control structures (if-else, switch, loops). Create a program to calculate the factorial of a number using different loop types. **Lab 3: Arrays and Strings** Write a Java program to manipulate arrays (sorting, searching, merging). Implement string operations such as concatenation, substring extraction, and string reversal. **Lab 4: Classes and Objects** Create a class representing a real-world object with attributes and methods. Instantiate objects and demonstrate the use of constructors and destructors. **Lab 5: Static Members and Methods** Write a Java program to demonstrate the use of static variables and methods. Implement a singleton class in Java. **Lab 6: Inheritance and Polymorphism** Implement a Java program demonstrating single and multilevel inheritance. Demonstrate method overriding and overloading. Create a program using abstract classes and interfaces. **Lab 7: Exception Handling**	

Write a Java program that demonstrates the use of try-catch blocks.
Implement custom exception handling.

Lab 8: File Handling

Write a Java program to read from and write to a file.
Implement a program that handles file exceptions.

Lab 9: Collections Framework

Create a program that uses ArrayList, LinkedList, HashSet, and HashMap.
Perform operations like adding, removing, and iterating through elements.

Lab 10: Mini Project

Develop a mini-project incorporating the concepts learned throughout the course.

Example project: Student Management System, Library Management System, or a simple banking application. This project should involve class design, inheritance, polymorphism, exception handling, file operations, and the use of collections.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Understand and Apply** Basic Java Programming Concepts	L2
2	Master Object-Oriented Programming Principles	L2
3	**Develop** and Use Java Collections Framework	L3
4	**Implement** and Manage Exception Handling in Java	L3
5	**Design** and Build Java Applications	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3			3									
CO2	2	3			3									
CO3	2	3			3									
CO4	2	3			3								3	3
CO5	3	3			3								3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. **Java: The Complete Reference",** Herbert Schildt**,** McGraw-Hill Education, 2021 (latest edition)
2. **"Head First Java",** Kathy Sierra and Bert Bates, O'Reilly Media, 2005 (latest edition)

Reference Books:

1. **"Head First Java",** Kathy Sierra and Bert Bates, O'Reilly Media, 2005 (latest edition)

Activity Based Learning: FLIP Classroom
E Resources: 2. https://education.oracle.com/java/java/pFamily_48
Laboratory Experiments: (not Mandatory)

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

This Course will enable students to:

1. **Apply** their knowledge of Laplace transforms and inverse Laplace transforms to proficiently solve linear ordinary differential equations with constant coefficients, facilitating the analysis and modelling of complex systems.
2. **Analyze** periodic functions using Fourier series, assessing the convergence properties and precision of the series expansion, thereby enhancing their ability to understand and manipulate periodic phenomena.
3. **Utilize** complex exponential form, Fourier transforms of basic functions, and Fourier sine and cosine transforms to solve problems involving Fourier integrals, developing proficiency in applying these techniques to various mathematical scenarios.
4. **Employ** numerical methods, including Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods, to solve differential equations and effectively analyze dynamic systems, enabling them to model real-world phenomena and make accurate predictions**.**
5. **Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to effectively solve different types of partial differential equations (PDEs) such as elliptic, parabolic, and hyperbolic equations, enhancing their problem-solving skills in the context of differential equations and their applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.

7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1: Laplace Transform and Inverse Laplace Transform	**09 Hours**

Laplace Transforms of Elementary functions (without proof), **(Text Book-1: Chapter 6: 203 to 207).**

Laplace Transforms of $e^{at}f(t)$, $t^n f(t)$ and $\frac{f(t)}{t}$, Periodic functions, Unit step function and t impulse functions **(Text Book-1: Chapter 6:208-230)**.

Inverse Laplace Transforms- By the method of Partial Fractions, Logarithmic and Trigonometric functions, Convolution Theorem, Inverse Laplace transform using Convolution Theorem **(Text Book-1: Chapter 6: 238)**.

Solution to Differential Equations by Laplace Transform. **(Text Book-1: Chapter 238-242)**.

Unit 2: Fourier Series	**09 Hours**

Periodic Functions, Trigonometric Series **(Text Book-1: Chapter 11: 495).**

Fourier series Standard function, Functions of any Period 2L, Even and Odd functions, Half-range Expansions.
(Text Book-1: Chapter 11: 483-492)

Practical Harmonic analysis (calculate average power and RMS values of periodicwaveforms)

Unit 3: Fourier Transform	**06 Hours**

Calculation of Fourier integrals using complex exponential form **(Text Book-1: Chapter 11: 510).**

Fourier transform of basic functions **(Text Book-1: Chapter 11: 510-516).**

Fourier sine and cosine transforms. **(Text Book-1: Chapter 11: 518-522).**

Unit 4: Numerical Methods for Solving Ordinary Differential Equations	**07 Hours**

Euler's Method-Basic principles of Euler's method for solving first-order ODEs **(Text Book-1:Chapter 1:10-12).**

Runge-Kutta 4th order **(Text Book-1: Chapter 21:904).**

Multistep Methods-Explanation of multistep methods (Adams-Bashforth, Adams–MoultonMethods) **(Text Book-1: Chapter 21:911-913).**

Second-Order ODE. Mass–Spring System (Euler Method, Runge–Kutta Methods)
(Text Book-1: Chapter 21:916-918).

Unit 5: Numerical Methods for Partial Differential	**08 Hours**

Equations	

Classification of PDEs (elliptic, parabolic, hyperbolic), **(Text Book-1: Chapter 21:922-923).** Finite Difference Methods (Laplace and Poisson Equations), Derivation of finite differenceapproximations **(Text Book-1: Chapter 21:923-927).**
Crank–Nicolson Method **(Text Book-1: Chapter 21:938-941).**
Method for Hyperbolic PDEs **(Text Book-1: Chapter 21:943-945).**

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Apply** Laplace transforms and inverse Laplace transforms tosolve linear ordinary differential equations with constant coefficients, demonstrating proficiency in system analysis andmodelling.	L3
2	**Analyze** periodic functions using Fourier series and evaluate the convergence properties and precision of the seriesexpansion.	L2 & L3
3	**Solve** problems involving Fourier integrals by applying complex exponential form, Fourier transforms of basicfunctions, and Fourier sine and cosine transforms.	L3
4	**Utilize** numerical methods such as Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods to solve differential equations and analyze dynamic systems	L2 & L3
5	**Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to solve various types of partial differential equations (PDEs)such as elliptic, parabolic, and hyperbolic equations.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	1					1					
CO2	3	2	2						1					
CO3	3	2	2	1					1					
CO4	3	2	2	1					1					
CO5	3	2	2	1					1					

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 2015, 10th Edition, Wiley India.

Reference Books:

1. Higher Engineering Mathematics, B.S. Grewal, 2015, 43rd Edition, Khanna Publishers.
2. Higher Engineering Mathematics, John Bird, 2017, 6 th Edition, Elsevier Limited.

Activity Based Learning:

E Resources:

1. https://nptel.ac.in/courses/111106139
2. https://nptel.ac.in/courses/111101164
3. https://nptel.ac.in/courses/111105038

Laboratory Experiments: (not Mandatory)

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code:	**Credits**: 04
Hours /Week: 03 Hours	**Total Hours**: 39(L) + 26(P) Hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the non-recursive and recursive algorithms and to represent efficiency of these algorithms in terms of the standard Asymptotic notations.
2. **Devise** the Brute Force and Divide and Conquer techniques to design the algorithms and apply these methods in designing algorithms to solve a given problem.
3. **Explain** the Decrease and Conquer, Transform and Conquer algorithm design techniques, and Time versus Space Trade-offs.
4. **Get the idea** of Greedy method and dynamic programming methods and apply these methods in designing algorithms to solve a given problem.

Describe and illustrate the idea of Backtracking and Branch and Bound algorithm design techniques to solve a given problem.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students'

understanding.

Unit 1:	08 Hours

INTRODUCTION:
What is an Algorithm? Fundamentals of Algorithmic Problem Solving. **(Text Book-1: Chapter 1: 1.1 to 1.2)**
FUNDAMENTALS OF THE ALGORITHMS EFFICIENCY:
Analysis Framework, Asymptotic Notations and Standard notations and common functions **(Text Book-2: Chapter 3: 3.1, 3.2)**, Mathematical Analysis of Non-recursive and Recursive Algorithms, **(Text Book-1: Chapter 2: 2.1, 2.3, 2.4,)**

Unit 2:	08 Hours

BRUTE FORCE:
Background, Selection Sort, Brute-Force String Matching. **(Text Book-1: Chapter 3: 3.1, 3.2)**
DIVIDE AND CONQUER:
General method, Recurrences: The recursion-tree method, The master method. **(Text Book-2: Chapter 4: 4.4, 4.5**),
Merge sort, Quick sort, Binary Search, Multiplication of large integers,
Case study: Strassen's Matrix Multiplication. **(Text Book-1: Chapter 4: 4.1 to 4.3, 4.5)**

Unit 3:	06 Hours

DECREASE & CONQUER:
General method, Insertion Sort, Graph algorithms: Depth First Search, Breadth First Search, Topological Sorting
TRANSFORM AND CONQUER:
Case study: Heaps and Heap sort.
TIME AND SPACE TRADEOFFS:
Input Enhancement in String Matching: Horspool's algorithm, Hashing: Open and Closed hashing.
(Text Book-1: Chapter 5: 5.1 to 5.3, Chapter 6: 6.3 to 6.4, Chapter 7: 7.2 to 7.3)

Unit 4:	09 Hours

GREEDY TECHNIQUE:
General method of Greedy technique, Single-Source Shortest Paths: General method, The Bellman-Ford algorithm, Single-Source Shortest Paths in DAGs, Dijkstra's Algorithm
(Text Book-2: Chapter 24: 24.1 to 24.3).
Minimum Spanning Trees: Prim's Algorithm, Optimal Tree problem: Huffman Trees;
Case study: Kruskal's Algorithm. **(Text Book-1: Chapter 9: 9.1, 9.2, 9.4).**
DYNAMIC PROGRAMMING:
General method, The Floyd-Warshall Algorithm, Johnson's algorithm for sparse graphs **(Text Book-2: Chapter 25: 25.1 to 25.3)**, The Knapsack problem **(Text Book-1: Chapter 8: 8.4).**

Unit 5:	08 Hours

LIMITATIONS OF ALGORITHMIC POWER

P, NP and NP-complete problems **(Text Book-1: Chapter 11: 11.3**)

BACKTRACKING:

General method, N-Queens problem, Subset-sum problem. **(Text Book-1: Chapter 12: 12.1**)

BRANCH AND BOUND:

General method, Travelling Salesman problem, Approximation algorithms for TSP.

Case study: Knapsack Problem. **(Text Book-1: Chapter 12: 12.2, 12.3**)

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Analyze the given recursive and non-recursive algorithms for time using step count, substitution method, recurrence tree method and mathematical analysis method. Represent the complexity of the algorithm using asymptotic notation.	L3
2	Solve sorting, searching, matrix multiplication problems based on divide and conquer design technique and implement in C programming language.	L3
3	Apply algorithms for graph-based problems (DFS, BFS and Topological sorting) using decrease and conquer design techniques. Distinguish the trade-offs between space and time complexity.	L3
4	Apply algorithms for finding the shortest path and minimum spanning tree for a given graph using greedy and dynamic programming techniques and implement the 0/1 knapsack problem in C programming language.	L3
5	Apply an efficient algorithm to solve N-Queens problem, Subset-sum problem, Knapsack and Traveling salesman problem using branch and bound and backtracking design technique. Describe the limitation of algorithmic power in terms of P, NP, NP hard and NP complete categorizations.	L3
6	Implement the graph-based algorithms including DFS, BFS, Warshall's Algorithm, Floyd's Algorithm, Kruskal's Algorithm, and Dijkstra's Algorithm in C programming language. (**Lab Experiments**)	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3	-	-	-	-	-	-	-	1	-	1	-	1
CO2	2	3	-	-	3	-	-	-	-	1	-	1	-	1
CO3	1	2	3	-	-	-	-	-	-	1	-	1	-	2

CO4	1	2	3	-	3	-	-	-	-	1	-	1	-	2
CO5	1	2	3	-	-	-	-	-	-	1	-	1	-	2
CO6	1	2	3	-	3	-	-	-	-	1	-	1	-	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Anany Levitin, "Introduction to the Design & Analysis of Algorithms", 2nd Edition, Pearson Education, 2011.
2. Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, PHI, 2014.

Reference Books:

1. Horowitz E., Sahni S., Rajasekaran S, "Computer Algorithms", Galgotia Publications, 2001.
2. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, "Introduction to the Design and Analysis of Algorithms: A Strategic Approach", Tata McGraw Hill, 2005.

Activity Based Learning:

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solutions to a problem through programming.

E Resources:

1. https://nptel.ac.in/courses/106/101/106101060/
2. http://cse01-iiith.vlabs.ac.in/
3. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
4. https://www.coursera.org/specializations/algorithms

Laboratory Experiments: (not Mandatory)

Following are experiments to be carried out using either C programming language or Object-oriented programming language:

1. Implementation of Binary Search algorithm.
2. Sort a given set of n integer elements using the Merge Sort method and compute its time complexity. Demonstrate this algorithm using the Divide-and-Conquer method.
3. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Demonstrate this algorithm using the Divide-and-Conquer method.
4. Incorporate the array data structure and demonstrate whether a given unweighted graph is connected or not using the DFS method.
5. Implement the graph traversal technique using the BFS method to print all the nodes reachable from a given starting node in an unweighted graph.
6. Compute the Transitive Closure for a given directed graph using Warshall's algorithm.

7. For a given weighted graph, construct an All-Pairs Shortest Paths problem using Floyd's algorithm and implement this algorithm to find the shortest distance and their shortest paths for every pair of vertices.
8. Implement 0/1 Knapsack problem using Dynamic Programming Memory Functions technique
9. Find Minimum Cost Spanning Tree for a given weighted graph using Prim's and Kruskal's algorithm.
10. From a given vertex in a weighted connected graph, determine the Single Source Shortest Paths using Dijkstra's algorithm.
11. Implement N-Queens problem using Backtracking technique.
12. Case Study

SEMESTER – IV	
Course Code	**Credits**: 04
Hours /Week: 03	**Total Hours**: 39(L) + 26(P) Hours
L–T–P–J:3-0-2-0	

Course Learning Objectives:

1. Understand and implement predictive modelling approaches using appropriate Python libraries.
2. Employ descriptive statistical methods to explore and prepare datasets effectively.
3. Design and apply suitable data wrangling techniques to handle and refine raw data.
4. Evaluate data from various sources to identify and develop optimal analytical solutions.
5. Apply data transformation and dimensionality reduction techniques to enhance data analysis and modelling.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
9. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
10. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**
BASIC CONCEPTS AND PYTHON PACKAGE :Basic Concepts: Predictive Modelling, Data preparation, Importance of Data preparation, Data Cleaning, Feature selection, Data Transform, Dimensionality reduction, K-fold cross validation, Data Leakage and avoidance measure. Python Packages: NumPy, Matplotlib, pandas, SciPy, scikit, Dataframe, Loading Machine Learning data.	
UNIT 2:	**08 Hours**
DESCRIPTIVE STATISTICS AND DATA PREPARATION : Descriptive Statistics: Reasons to study Statistics, Sampling, Data Analysis Process, Mean, Median, Standard Deviation, Skewness, Kurtosis, Graphical Representation-Box Plots, Pivot Table, Heat Map Correlation, Statistics–ANOVA. Data Preparation: Need for Data Pre-processing, Data Transforms, and Rescale Data Standardize Data, Normalize Data, Binarize Data, Univariate and Bivariate Data, Recursive Feature Elimination, Principal Component Analysis.	
UNIT 3:	**08 Hours**
DATA CLEANING AND FEATURE SELECTION : Data Cleaning: Basic data cleaning, Outlier Identification and Removal, how to Mark and Remove Missing Data, Statistical Imputation, KNN Imputation, Iterative Imputation. Feature Selection: Statistics for feature selection, Methods for categorical input, Methods for Numerical input, Select Features for Numerical Output, RFE for Feature Selection, Significance of feature selection.	
UNIT 4:	**08 Hours**
DATA TRANSFORM AND DIMENSIONALITY REDUCTION : Data Transforms: Scaling data source, min-max scalar and standard scaler, Scale data with outliers, encode categorical data, Make Distributions More Gaussian, Approach for Numerical Data Distributions, Deriving new input variables. Dimensionality reduction: Techniques for Dimensionality Reduction, Linear Discriminant Analysis, PCA Dimensionality Reduction, SVD Dimensionality Reduction	

UNIT 5:	**07 Hours**

OTHER TRANSFORMS : Transform numerical to categorical, Transform Numerical and Categorical Data, Transform the Target in Regression, Save and load the transformation, case studies for Binary classification, Multiclassification and Regression

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Illustrate predictive modeling techniques with necessary python packages	**L2**
2	Apply descriptive statistics concepts for data preparation	**L3**
3	Develop and examine appropriate methods for data wrangling	**L3**
4	Analyze efficient solution for the given data sources	**L4**
5	Examine the data transformation and dimension reduction techniques on the data source	**L4**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**	**3**		**2**								**3**	**3**
CO2	**3**	**2**	**2**		**2**								**3**	**3**
CO3	**3**	**2**	**2**		**2**								**3**	**3**
CO4	**3**	**2**	**2**		**2**								**3**	**3**
CO5	**3**	**2**	**2**		**2**								**3**	**3**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1) Jason Brownlee, "Data Preparation for Machine Learning" 2020.

2) Roxy Peck, Chris Olsen and Jay Devore, "Introduction to Statistics & Data Analysis "3rd Edition Thomson Higher Education,2015. ISBN: 1305445961, 9781305445963

Reference Books:

1) Andrew Park ,"Data Science For Beginners",2021. ISBN: 9781914167997

2) Norman Matloff, "Probability and Statistics for Data Science", CRC Press,2019. ISBN: 9780429687112

Activity Based Learning:

1) Flip Classroom

E Resources:

1) Data Science for Engineers:https://digimat.in/nptel/courses/video/106106179/L01.html

2) Statistics for Data Science: https://www.youtube.com/watch?v=V5fqShLVpoI

Lab Experiments

1. Reading different types of data sets (.txt, .csv) from Web and disk and writing in file in specific disk location.
   a. Reading Excel data sheet using Python Pandas.
   b. Reading XML dataset using Python Pandas.
   c. Reading JSON data using Python Pandas

   Create three datasets in different formats: employees.xlsx (Excel), departments.xml (XML), and salaries.json (JSON). Perform the following tasks:
   a. Read all three datasets into Pandas DataFrames.
   b. Merge these datasets based on a common column (e.g.,employee_id).
   c. Save the merged DataFrame to a new CSV file in a specified disk location.
2. Design a Python program to perform exploratory data analysis tasks on a sample dataset. Select an appropriate data set for your experiment and draw the following graphs.
   a. Find the data distributions using box and scatter plot.
   b. Find the outliers using plot.
   c. Plot the histogram, bar chart and pie chart on sample data.
   d. Plot the Heat Map
3. Implement K-fold cross validation techniques. Perform a cross- validation experiment on for a given set of training data examples stored in a .CSV file using 10 folds
4. Perform the following methods in order to remove outliers
   a. Standard Deviation Method
   b. Interquartile Range Method
   c. Automatic outlier Detection

   Create a statistics_metrics.csv dataset with columns quarter, revenue, expenses and profit, and perform the following tasks:

a. Identify and remove outliers using both the Standard Deviation method and the Interquartile Range (IQR) method. b. Compare the datasets before and after outlier removal using box plots and scatter plots.
5. Implement the program to avoid Data leakage with Naïve Data preparation. Select an appropriate data set for your experiment and validate the results
6. For a given set of training data examples stored in a .CSV file a. Statistical Imputation b. KNN Imputation c. Iterative Imputation
7. Implement the following Encoding methods for a given set of training data: a. Ordinal Encoding b. One Hot Encoding c. Dummy Variable Encoding
8. Implement the following Transform methods on a numerical dataset: Uniform Discretization Transform
9. Implement the following Transform methods on a numerical dataset: K-Means Discretization Transform

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	**Credits**: 03
Hours /Week: 3Hrs	**Total Hours**: 39 Hours
L–T–P–J: 3-0-0	

Course Learning Objectives:

1. Gain a comprehensive understanding of the core principles and components of traditional data warehouses.
2. Understand how to structure and organize large datasets using dimensional modelling techniques.
3. Learn how cloud-based data warehousing platforms like Snowflake operate and manage data efficiently.
4. Explore mechanisms to manage, transform, and monitor data workflows within the Snowflake ecosystem.
5. Understand how to connect Snowflake to downstream tools for business intelligence and enable collaborative data usage across teams.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**

INTRODUCTION TO DATA WAREHOUSING: Data Warehousing Concepts and Architecture-Definition and purpose of data warehousing, Differences between OLTP and OLAP system, Components of a data warehouse: operational source systems, data staging area, data presentation area, data access tools, ETL Processes- Extract, Transform, Load processes, Importance in data warehousing, Data Marts and Business Intelligence-Role of data marts in data warehousing, Introduction to business intelligence tools

UNIT 2:	**07 Hours**

DATA MODELLING AND SCHEMA DESIGN: Dimensional Modelling Techniques-Importance of dimensional modelling, Fact and dimension tables, Dimensional modelling vocabulary, Schema Design-Star schema design, Snowflake schema design, Comparison between star and snowflake schemas, Slowly Changing Dimensions (SCD)-Types 1, 2, and 3 SCDs, Implementation strategies

UNIT 3:	**08 Hours**

INTRODUCTION TO SNOWFLAKE: Snowflake Architecture and Key Features-Overview of Snowflake's architecture, Unique features of Snowflake

Setting up Snowflake Environment-Creating accounts and virtual warehouses, Understanding Snowflake's web interface, Database Objects in Snowflake, Creating databases, schemas, and table, Data types and constraints in Snowflake, Data Loading and Unloading-Methods for loading data into Snowflake, Exporting data from Snowflake.

UNIT 4:	**08 Hours**

ADVANCED SNOWFLAKE FEATURES: Time Travel and Data Cloning-Understanding Time Travel feature ,Creating zero-copy clones, Streams and Tasks-Implementing data pipelines with Streams, Automating tasks in

Snowflake, Materialized Views and Result Caching-Creating and managing materialized views, Utilizing result caching for performance, Performance Tuning and Query Optimization-Techniques for optimizing queries, Monitoring and tuning performance, Security Features-Role-Based Access Control (RBAC),Data masking and encryption.

UNIT 5:	**08 Hours**

DATA SHARING AND INTEGRATION IN SNOWFLAKE: Data Sharing and Marketplace, Sharing data securely within and outside the organization, Exploring Snowflake Marketplace, Integration with BI Tools-Connecting Snowflake to Tableau, Power BI, etc, Best practices for data visualization, Handling Semi-Structured Data-Working with JSON, Avro, Parquet formats, Parsing and querying semi-structured data, Continuous Data Ingestion with Snowpipe-Setting up Snowpipe for real-time data ingestion, Monitoring and managing data streams, Real-Time Analytics and Use Case, Implementing real-time analytics solutions, Case studies and industry applications.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Explain the architecture, components, and processes involved in traditional data warehousing systems.	**L2**
2	Design dimensional models using star and snowflake schemas by applying best practices in data modelling.	**L3**
3	Utilize Snowflake's cloud-native features for structured and semi-structured data storage and querying.	**L3**
4	Analyse and optimize Snowflake queries and data pipelines using built-in features like Time Travel, Streams, and Tasks.	**L4**

5	Develop secure and scalable data warehousing solutions integrating Snowflake with external BI tools for real-time analytics and sharing.	**L4**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**	**2**										**3**	**3**
CO2	**3**	**2**	**2**										**3**	**3**
CO3	**3**	**2**	**2**		**3**								**3**	**3**
CO4	**3**	**2**	**2**		**3**								**3**	**3**
CO5	**3**	**2**	**2**		**3**								**3**	**3**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1) Jason Brownlee, The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling**,** Ralph Kimball and Margy Ross,Wiley,2nd Edition, ISBN: 978-0471200246

2) Snowflake: The Definitive Guide, Joyce Kay Avila, O'Reilly Media,1st Edition, *ISBN:* 978-1098103828

Reference Books:

1) Building the Data Warehouse, W.H. Inmon,Wiley

2) Cloud Data Management, Divesh Srivastava, Amol Deshpande, et al.,Morgan & Claypool Publishers

Activity Based Learning:

1)Analysis of Real-World Data Warehousing Implementations

2)Performance Tuning Scenarios in Snowflake

E Resources:

1) Snowflake Official Documentation https://docs.snowflake.com/

2) Kimball Group Resources https://www.kimballgroup.com/

MACHINE LEARNING [As per Choice Based Credit System (CBCS) scheme] **SEMESTER – IV**	
Course Code	**Credits**:04
Hours /Week: 03+02 Hours	**Total Hours**: 39(L) + 26(P) Hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

1. Gain knowledge of core machine learning concepts and explore various real-world applications.
2. Implement logistic regression for solving binary and multi-class classification problems using appropriate techniques.
3. Utilize decision tree algorithms to model complex decision-making scenarios involving both categorical and numerical data.
4. Assess the performance of machine learning models using advanced evaluation metrics such as precision, recall, F1 score, and ROC-AUC.
5. Explore and compare the effectiveness of classification models built using supervised and unsupervised learning algorithms

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.

6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**
INTRODUCTION TO MACHINE LEARNING: Understanding Machine Learning: Definition and Types of Machine Learning-Application of Machine Learning- Machine Learning Algorithms: Supervised, Unsupervised, and Semi-Supervised Learning Algorithms. Machine Learning Models-**Model Evaluation Metrics:** Confusion Matrix, Precision, Recall, F1 Score - ROC Curve and AUC-ROC. **Advanced Techniques:** Feature Scaling and Normalization -Encoding Categorical Variables-Train-test Split and Cross-validation.	
UNIT 2:	**08 Hours**
SUPERVISED LEARNING ALGORITHMS : Regression: Introduction to Regression- Regression Models- Linear Regression, Logistic Regression ,Polynomial Regression-. **Decision Trees:** Introduction to Decision Trees-Tree Construction, Splitting Criteria, and Pruning-Handling Missing Values and Categorical Features- Gini Index-ID3-CART, Support Vector Machine (SVM)	
UNIT 3:	**06 Hours**
PROBABILISTIC BASED MODELS: Naive Bayes: Introduction to Naive Bayes Classifier-Bayes' Theorem and Conditional Probability-Gaussian, Multinomial, and Bernoulli Naive Bayes. Bayesian Belief Network-EM algorithm.	
UNIT 4:	**09 Hours**
UNSUPERVISED ALGORITHMS : Introduction To Unsupervised Learning, Clustering (Hard and Soft Clustering) Hierarchical Clustering: Fuzzy C-Means (FCM) Algorithm, Gaussian Mixture Models (GMM), Expectation Maximization Algorithm, Introduction to k-means and Choosing 'k' , k-Nearest Neighbors Algorithm-Distance Metrics	
UNIT 5:	**08 Hours**

TIME SERIES FORECASTING: Understanding Time Series, Components of Time series, describing vs predicting, Data preparation, Feature selection for Time series, Date Time features, Lag feature, rolling window statistics, Expanding window statistics, Time series Visualization, Line plot, Histogram, Density plot, Autocorrelation plot

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Understand the fundamental principles of Machine Learning and its applications.	L2
2	Apply logistic regression as a classification algorithm to demonstrate the proficiency in modelling binary and multi-class classification problems.	L3
3	Develop the decision tree algorithm for decision boundaries and handling categorical and numerical data.	L3
4	Analyse ML model performance (supervised and unsupervised algorithms) using advanced metrics such as precision, recall, F1 score, and ROC-AUC curve for effectiveness and robustness.	L4
5	Compare a suitable method for data preparation and ML algorithm for time series data	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	2	2								3	3
CO2	3	2	2	2	2								3	3
CO3	3	2	2	2	2								3	3
CO4	3	3	2	2	2								3	3
CO5	3	3	2	2	2								3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Tom Mitchell, "Machine Learning", McGraw Hill,1997.ISBN 9780071154673,0071154671
2. Charu C. Agarwal, "Data mining" , Springer, 2015. ISBN: 9783319141428, 3319141422

Reference Books:

1. Introduction to Machine Learning with Python: A Guide for Data Scientists" by Andreas C. Müller and Sarah Guido, 2016. ISBN: 9781449369903

Activity Based Learning:

1. Group Discussion on real-world Problems

E Resources:

1. https://www.analyticsvidhya.com/blog/2021/08/decision-tree-algorithm/
2. https://ocw.mit.edu/courses/15-097-prediction-machine-learning-and-statistics-spring 2012/eb02afbd0a9c32637dd64cdb6b76c2f1_MIT15_097S12_lec01.pdf

Lab Experiments

1.	Develop a python program to perform binary classification and evaluate performances using Confusion matrix, ROC Curve, Precision, Recall and F1 Score for any given dataset
2.	Develop models for Ridge regression and Lasso regression for the given dataset and evaluate its performance then compare above two methods
3.	Develop a multiple regression model for the given data set and evaluate its performance
4.	Write a program to demonstrate the working of the decision tree Based CART algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new Sample
5.	Develop a program for logistics regression model for the given data and compare performance with other regression model
6.	Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Calculate the accuracy, precision, and recall for your data set
7.	Apply k-means and DBSCAN Clustering algorithms to generate clusters for the given dataset and evaluate its performance and compare
8.	Write a Python program to implement the Apriori algorithm using a given transaction dataset stored as a .CSV file. Identify frequent item sets and association rules from data file
9.	Write a program to implement *k*-Nearest Neighbour algorithm to classify the data set. Print both correct and wrong predictions

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

Business Intelligence Tools

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code:	**Credits**: 02
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: 1–0-2	

Course Learning Objectives:

This Course will enable students to:

1. To introduce students to the fundamental concepts of Business Intelligence (BI) and its role in decision-making.
2. To provide knowledge of data warehousing, ETL processes, and BI architecture.
3. To develop practical skills in using BI tools for data analysis, visualization, and dashboarding.
4. To expose students to real-world data and help them create insights using popular BI platforms like Power BI and Tableau.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**07 Hours**

Introduction to Excel for Data Science: Role of Excel in data science and analytics workflows. Navigation through Excel interface, workbooks, sheets, and menus. Basic formulas and functions including SUM, AVERAGE, IF, COUNTIF, VLOOKUP, INDEX, MATCH. Understanding cell referencing: relative, absolute, and mixed. Sorting, filtering, and removing duplicates in datasets. Text manipulation functions: TRIM, LEFT, RIGHT, MID, TEXT, CONCATENATE, TEXTJOIN. Date and time handling techniques. Data validation with dropdowns and rule setting. Conditional formatting for highlighting patterns and anomalies. Error checking using IFERROR, ISNUMBER, ISBLANK. Mini project on data cleaning and summarization using Excel tools

Unit 2:	**06 Hours**

Introduction to Business Intelligence: Introduction to BI: Definition, importance, applications, BI architecture and components, Data visualization basics, BI tools overview (Power BI, Tableau, Qlik, Excel BI)

Dashboarding and Data Visualization using Power BI/Tableau: Connecting to various data sources (Excel, SQL, web, etc.), Data cleaning and transformation using Power Query, creating visualizations: charts, tables, KPIs, filters, Building interactive dashboards, DAX basics (Power BI): Measures, calculated columns, Publishing and sharing dashboards

Unit 3:	**06 Hours**

Lab No.	Experiment Title	Tool
1	Data Extraction and Loading (ETL): Load and clean data from Excel and SQL using Power Query	Power BI / Tableau
2	Data Modelling: Create relationships between tables, use star schema	Power BI
3	Basic Dashboard Creation: Build a simple sales or HR dashboard	Power BI / Tableau
4	Advanced Visuals and Filters: Add slicers, time filters, and KPI cards	Power BI
5	DAX for Business Calculations: Create calculated measures like YoY, % change, rank	Power BI

6	Publish and Share Dashboard: Upload to Power BI service, create a report link, set refresh schedule	Power BI Service	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Understand the architecture and components of Business Intelligence systems	L2
2	Apply ETL techniques for preparing data suitable for business analysis	L2
3	Use BI tools to connect, transform, and visualize data from different sources	L3
4	Develop and publish interactive dashboards for real-world business scenario	L3
5	Demonstrate proficiency in writing simple analytical expressions using DAX (for Power BI)	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3			3									
CO2	2	3			3									
CO3	2	3			3									
CO4	2	3			3								3	3
CO5	3	3			3								3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books: ***Power BI for the Excel Analyst*****, Wyn Hopkins, Publisher: Holy Macro! Books, 2022**

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

COMPUTER NETWORKS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

1. **Understand** the basic principles of computer networking and how computer network hardware and software operate.
2. **Evaluate** the operation and performance of practical data link protocols using the principles of framing, error detection and correction.
3. **Apply** the principles of network layer design to the analysis and evaluation of routing algorithms, congestion control techniques, internetworking and addressing.
4. **Investigate** the basic transport layer facilities and essentials of transport. Protocol
5. **Describe** the working of various application layer protocols.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
INTRODUCTION Uses of Computer Networks, Types of Computer Networks, Network Technology, from Local to Global, Examples of Networks , Network Protocols, Reference Models , Standardization **Text Book (1.1 to 1.7 Pg. nos. -1-74)**	
Unit 2:	**08 Hours**
THE PHYSICAL LAYER Guided Transmission Media, Wireless Transmission, Using the Spectrum for Transmission, From Waveforms to Bits, Communication Satellites. **Text Book (2.1- to 2.4, 2.8 Pg. nos. - 90-130, 176 - 184)**	
Unit 3:	**08 Hours**
THE DATA LINK LAYER Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Protocols, Improving Efficiency. **THE MEDIUM ACCESS CONTROL SUB LAYER** The Channel Allocation Problem, Multiple Access Protocols. **Text Book (3.1 to 3.4 , 4.1 to 4.2.1 Pg. nos.: 202 - 238, 268 - 276)**	
Unit 4:	**08 Hours**
THE NETWORK LAYER Network Layer Design Issues, Routing Algorithms in a Single Network, The Network Layer in the Internet. **Text Book (5.1, 5.2: 5.2.1 to 5.2.6, 5.7: 5.7.1 to 5.7.3. Pg. nos.: 360 – 384, 441- 470)**	
Unit 5:	**7 Hours**
THE TRANSPORT LAYER The Transport Service, Elements of Transport Protocols, **THE APPLICATION LAYER** DNS — The Domain Name System, Electronic Mail , WWW, Streaming Audio and Video **Text Book (6.1-6.1.3, 6.2 ,7: 7.1.1 to 7.1.5, 7.2:7.2.1, 7.3:7.3.1, 7.4:7.4.1 to 7.4.2 Pg. No 501- 509, 513-536, 613-629, 632-635,650-653, 682-687)**	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	**Understanding** the basic concepts of data communications including the key aspects	L2

	of networking Network Technology, from Local to Global, Examples of Networks, Network Protocols	
2	**Explain** the concepts of Guided Transmission Media, Wireless Transmission, Using the Spectrum for Transmission	L2
3	**Illustrate** the concepts of Design Issues, Error Detection and Correction, Elementary Data Link Protocols, Improving Efficiency.	L3
4	**Explain** the Network Layer Design Issues, Routing Algorithms in a Single Network, The Network Layer in the Internet	L2
5	**Discuss** about the Transport Service, Elements of Transport Protocols, The Domain Name System, Electronic Mail , WWW, Streaming Audio and Video	L2

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	-	-	-	-	-	-	-	-	-	1	0
CO2	3	3	3	3	-	-	-	-	-	-	-	-	2	0
CO3	3	3	3	-	-	-	-	-	-	-	-	-	2	0
CO4	3	3	3	-	-	-	-	-	-	-	-	-	1	0
CO5	3	3	3	-	-	-	-	-	-	-	-	-	1	0
CO6	3	3	-	-	-	-	-	-	-	-	-	-	1	

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Andrew S. Tanenbaum, David.J. Wetherall, ―Computer Networks‖, Prentice-Hall, 6th Edition, 2022.

Reference Books:

1. Behrouz A. Forouzan,―Data Communications and Networking‖, TataMcGraw-Hill,5thEdition, 2012.
2. Chwan-Hwa Wu, Irwin, ―Introduction to Computer Networks and Cyber Security‖, CRC publications, 2014.
3. Douglas E. Comer, ―Internetworking with TCP/IP ―, Prentice-Hall, 5thEdition, 2011.

Activity Based Learning:

1. Real world problem solving using group discussion.
2. Flip class activity

E Resources:

1. http://computer.howstuffworks.com/computer-networking-channel.htm
2. https://www.geeksforgeeks.org/layers-osi-model/
3. https://www.wikilectures.eu/w/Computer_Network
4. https://technet.microsoft.com/en-us/network/default.aspx

Laboratory Experiments: (not Mandatory)

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SOFTWARE ENGINEERING PRINCIPLES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code	**Credits**:03
Hours /Week: 03	**Total Hours**:39 Hrs
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

1. Gain foundational insights into software engineering models, practices, and life cycle phases relevant to building reliable and scalable software systems.
2. Explore real-world practices in Agile development, including the use of tools like JIRA and the handling of service tickets and SLAs in IT industry environments.
3. Understand the essential components of software project planning, including scheduling, budgeting, resource allocation, and risk handling in dynamic environments.
4. Develop a working knowledge of software testing principles and quality assurance strategies, with special emphasis on applications in data science projects.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.

7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 hours**
SOFTWARE ENGINEERING PRINCIPLES: Introduction to Software Engineering-Nature and characteristics of software, Software engineering principles, Software process models: Waterfall, Incremental, Spiral, RAD, Agile, Software Development Life Cycle (SDLC)-Phases: Requirement analysis, Design, Implementation, Testing, Deployment, Maintenance, Role of software engineering in SDLC	
UNIT 2:	**08 hours**
AGILE METHODOLOGIES AND INDUSTRY PRACTICES: Agile Software Development-Principles of Agile, Scrum framework: Roles, events, artifacts, Extreme Programming (XP): Practices and benefits, Industry Tools and Practices, Introduction to JIRA for issue tracking, Ticketing systems in IT support, Understanding Service Level Agreements (SLAs),Real-world case studies of Agile implementation	
UNIT 3:	**08 hours**
SOFTWARE PROJECT MANAGEMENT: Project Management Fundamentals-Project planning and scheduling, Resource allocation and budgeting, Risk management strategies, Project Monitoring and Control-Key Performance Indicators (KPIs),Earned Value Management (EVM),Change management processes	
UNIT 4:	**08 hours**
SOFTWARE TESTING IN DATA SCIENCE PROJECTS: Testing Fundamentals-Types of testing: Unit, integration, system, acceptance, Test planning and documentation, Testing in Data Science-Challenges in testing data-driven applications, Validation of machine learning models, Tools for testing in data science (e.g., PyTest, unit test)	
UNIT 5:	**07 hours**
SOFTWARE QUALITY ASSURANCE AND MAINTENANCE: Software Quality Assurance (SQA) - Quality models: ISO 9126, CMMI,SQA activities and audits, Software Maintenance - Types of maintenance: Corrective, adaptive, perfective, preventive, Maintenance challenges in data science projects	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Understand the fundamental principles of software engineering and their application in real-world projects.	L2
2	Apply agile methodologies and utilize industry-standard tools for effective project management.	L3
3	Analyse project requirements and develop comprehensive project plans, including risk assessment and resource allocation.	L4
4	Build software quality through various testing strategies, particularly in data science applications.	L3
5	Choose the software systems post-deployment, ensuring long-term sustainability ,monitoring and maintainability	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2										2	2	3
CO2	2	2										2	2	3
CO3	2	2										2	2	3
CO4	2	2										3	2	3
CO5	2	2										3	2	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1) Software Engineering: A Practitioner's Approach by Roger S. Pressman, 7th Edition

2) Software Engineering and Project Management by Dr. Prakash Mahanwar, University of Mumbai

Reference Books:
1) Software Engineering, 10th Edition by Ian Sommerville 2) Fundamentals of Software Engineering Project Management by Dr. Johan Gouws and Mrs. Leonie Gouws
Activity Based Learning:
2) Case Studies: Analyse real-world software engineering and project management scenarios.
E Resources:
1) Atlassian Agile Coach: https://www.atlassian.com/agile 2) Scrum Guides: https://scrumguides.org/

CLOUD DATA ENGINEERING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code	**Credits**: 04
Hours /Week: 03	**Total Hours**: 39(L) + 26(P) Hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

1. Gain knowledge of Azure data engineering architecture, including key storage services and database management options, with a focus on scalability, durability, and secure data handling.
2. Design and implement end-to-end ETL and ELT data workflows using Azure Data Factory, incorporating source connectors, mapping data flows, and integration runtimes.
3. Explore how to work with large-scale datasets using Azure Synapse Analytics, including querying, data modelling, and building interactive dashboards with integrated visualization tools.
4. Work with Delta Lake features in Azure Databricks for optimized data transformations and implement machine learning workflows that integrate with Spark-based data processing.
5. Understand strategies for securing data pipelines, ensuring regulatory compliance, setting up monitoring and alerting, and optimizing data engineering solutions for performance across Azure services.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.

6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**

INTRODUCTION TO AZURE DATA ENGINEERING AND STORAGE SOLUTIONS: Overview of Azure Cloud Platform- Introduction to Azure and its core services, Understanding Azure Data Services ecosystem, Role of a Data Engineer in cloud environments, Azure portal and resource management basics, Azure Data Storage Solutions - Azure Storage Accounts: Blob, File, Table, and Queue storage, Azure Data Lake Storage (ADLS) Gen2: Features and use cases, Managing data storage with Azure CLI, PowerShell, and Azure Portal, Data ingestion strategies using Azure Storage Explorer , Introduction to Azure SQL Database - Creating and managing Azure SQL Databases, Database scaling, performance tuning, and security features, Azure SQL Data Warehouse (Synapse SQL Data Pool) basics, Hands-on: Creating and managing Azure Storage and Azure SQL Database

UNIT 2:	**08 Hours**

DATA INTEGRATION WITH AZURE DATA FACTORY (ADF): Introduction to Azure Data Factory ,Overview of ETL/ELT processes, ADF architecture and components (Pipelines, Datasets, Linked Services),Data movement and transformation capabilities, Building Data Pipelines, Creating ADF pipelines for data ingestion and transformation, Scheduling and monitoring data pipelines, Data integration with on-premises and cloud data sources, Hands-on: Building data pipelines using ADF, Data Transformation using Mapping Data Flows, Introduction to Mapping Data Flows in ADF, Data transformation activities (Join, Filter, Aggregate, etc.),Debugging and optimizing data flows, Hands-on: Data transformation using Mapping Data Flows

UNIT 3:	**08 Hours**
BIG DATA PROCESSING WITH AZURE SYNAPSE ANALYTICS: Introduction to Azure Synapse Analytics - Overview of Synapse Analytics architecture, Understanding Synapse SQL pools (Dedicated and Serverless),Synapse Studio for data integration and analytics, Data Warehousing with Azure Synapse -Designing and implementing data warehouses on Synapse, Data partitioning, indexing, and performance optimization, Integrating data from Azure Data Lake and other sources, Hands-on: Creating and managing Synapse SQL pools, Data Exploration and Analytics - Using Synapse Studio for data exploration and visualization, Integrating Synapse with Power BI for interactive reporting, Implementing security best practices in Synapse, Hands-on: Data analytics with Synapse Studio	
UNIT 4:	**08 Hours**
ADVANCED DATA ENGINEERING WITH AZURE DATABRICKS: Introduction to Azure Databricks- Overview of Databricks and Apache Spark, Setting up Azure Databricks workspace, Databricks clusters, notebooks, and jobs, Data Engineering with Spark on Databricks - Data ingestion using Spark, Data transformation and aggregation with PySpark, Working with Delta Lake for ACID transactions and data versioning, Hands-on: Data engineering workflows with Azure Databricks, Machine Learning and AI Integration-Building and training ML models using Databricks MLflow, Integrating Databricks with Azure Machine Learning, Hands-on: Building end-to-end ML pipelines on Databricks	
UNIT 5:	**07 Hours**
DATA SECURITY, MONITORING, AND OPTIMIZATION ON AZURE: Data Security and Compliance - Data encryption at rest and in transit, Implementing data governance with Azure Purview, Role-based access control (RBAC) and network security groups (NSG),Hands-on: Configuring security for Azure data services, Data Monitoring and Optimization, Monitoring data solutions using Azure Monitor and Log Analytics, Performance tuning of data pipelines and databases, Cost management and optimization strategies, Hands-on: Monitoring and optimizing Azure data workloads, End-to-End Data Engineering Project.	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	Understand the fundamentals of Azure data engineering, storage solutions, and database management for scalable and secure data storage.	L2
2	Design build and manage ETL/ELT pipelines using Azure Data Factory for data integration and transformation.	L3
3	Analyse big data using Azure Synapse Analytics for data warehousing and visualization.	L4
4	Apply the data transformations, Delta Lake, and machine learning integration in Data Bricks	L3
5	Evaluate the data security, compliance, monitoring, and performance optimization techniques to manage end-to-end data engineering projects on Azure.	L5

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**	**2**	**2**	**3**								**3**	**3**
CO2	**3**	**3**	**2**	**2**	**3**								**3**	**3**
CO3	**3**	**3**	**3**	**3**	**3**								**3**	**3**
CO4	**3**	**3**	**3**	**3**	**3**								**3**	**3**
CO5	**3**	**3**	**3**	**3**	**3**								**3**	**3**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1) Azure Data Engineering: Demystified" by Mark Beckner ,Apress, ISBN: 978-1484268834

2) Azure Data Factory Cookbook" by Dmitry Anoshin, Dmitry Foshin, Packt Publishing, ISBN: 978-1800565296

Reference Books:

1) Microsoft Azure Data Solutions: An Introduction" by Daniel A. Seara, Francesco Diaz, Microsoft Press, ISBN: 978-0136798627

Activity Based Learning:

1) Stream Real-Time Data

2) Create a Data Visualization Dashboard

E Resources:

1)Microsoft Learn: Azure Data Engineer Learning Path https://learn.microsoft.com/en-us/training/paths/azure-data-engineer/

2) Azure Architecture Center: Data Engineering on Azure https://learn.microsoft.com/en-us/azure/architecture/data-guide/technology-choices/data-engineering-overview

1. Create Storage Accounts (Blob, File, Table), work with ADLS Gen2 using Azure CLI & Portal
2. Create Azure SQL DB, configure performance settings, explore scaling and security options
3. Build ADF pipelines to ingest data from Blob storage to SQL DB using Linked Services
4. Use Join, Filter, Aggregate, Sort, and Derived Column transformations with debugging
5. Create Dedicated SQL pools, load data from ADLS, implement indexing and partitioning
6. Use Synapse Studio to query and visualize datasets; integrate Power BI dashboards
7. Use Azure Monitor, Log Analytics, and Cost Management to analyze usage and optimize
8. Use Databricks Notebooks, PySpark for ingestion/transformation, Delta Lake for versioning, MLflow integration

DEEP LEARNING ESSENTIALS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code:	**Credits**: 04
Hours /Week: 03 Hours	**Total Hours**: 39(L)Hours + 26(P) Hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

1. To understand the basic building blocks and general principles that allow one to design Deep learning algorithms
2. To become familiar with specific, widely used Deep learning networks
3. To introduce building blocks of Convolution neural network architecture
4. To learn to use deep learning tools and frameworks for solving real-life problems

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
INTRODUCTION TO DEEP NETWORKS: History of Deep Learning, Perceptron's, Perceptron Learning Algorithm and Convergence, Multilayer Perceptron's (mlps), Representation Power of mlps, batch normalization- VC Dimension and Neural Nets-Deep Vs Shallow Networks Convolutional Networks- Generative Adversarial Networks (GAN), Semi supervised Learning	
Unit 2:	**08 Hours**
COMPUTATION IN DEEP LEARNING: Forward Propagation, Backward Propagation,Computational Graphs Layers, and Blocks,Activation Functions, Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam, Ensemble methods, Dropout, Better weight initialization methods, Batch Normalization	
Unit 3:	**08 Hours**
Convolutional Neural Networks, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet, Visualizing Convolutional Neural Networks, Guided Backpropagation, Deep Dream, Deep Art, Fooling Convolutional Neural Networks, Learning Vectorial Representations Of Words.	
Unit 4:	**08 Hours**
Recurrent Neural Networks, Backpropagation Through Time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT, Gated Recurrent Units (GRUs), Long Short Term Memory (LSTM) Cells, Solving the vanidhing gradient problem with LSTMs.	
Unit 5:	**7 Hours**
Encoder Decoder Models, Attention Mechanism, Attention over images, Hierarchical Attention, Transformers: Multi-headed Self Attention, Cross Attention. **CASE STUDY AND APPLICATIONS:** Deep learning for early diagnosis of Alzheimer's disease: a contribution, Crop disease classification using deep learning approach	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	Build an Image classifier model for applying the concept of single layer and multilayer NN and analyze activation and loss function with that model.	L2
2	Apply the mathematical concept of deep learning for the manipulation and preprocessing of data.	L2

3	Evaluate deep learning models applying optimization techniques to solve real-world problems and analyse the efficiency of the models.	**L3**
4	Build an image classifier model, applying CNN and evaluating associated hyperparameters.	**L4**
5	Construct deep learning-based models for healthcare applications and compare effectivity of advanced networks.	**L4**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2				3						2	2		3
CO2	2				3						2	2		3
CO3	3			2	3						2	2		3
CO4	3	2		3	3						3	3		3
CO5	3	2		3	3						2	3		3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Aston Zhang, Zack C. Lipton, Mu Li, Alex J. Smola, “Dive into Deep Learning”, Amazon Science, 20202. Cosma Rohilla Shalizi, Advanced Data Analysis from an Elementary Point of View,2015.

Reference Books:

1. Tom Mitchell, Machine Learning, McGraw-Hill, 1997.
2. Deng & Yu, Deep Learning: Methods and Applications, Now Publishers, 2013.
3. Ian Goodfellow, YoshuaBengio, Aaron Courville, Deep Learning, MIT Press,2016. Michael Nielsen, Neural Networks and Deep Learning, Determination Press, 2015.

Activity Based Learning:

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity
3. Study projects (by very small groups of students on selected local real-time problems about the syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

Laboratory Experiments: (not Mandatory)

1. Build a simple feedforward neural network using only NumPy and implement forward

propagation.
2. Write a program to train a neural network using different optimization algorithms: SGD, Momentum, Nesterov Accelerated Gradient, AdaGrad, RMSProp, and Adam.
3. Apply batch normalization to a dataset and compare training speed and performance.
4. Implement LeNet, AlexNet, and VGGNet using a deep learning framework (e.g., PyTorch or TensorFlow).
5. Implement a basic GAN from scratch and train it on the MNIST/any other dataset.
6. Implement a basic Recurrent Neural Network (RNN) for character-level text generation.
7. Implement or fine-tune a transformer-based model (e.g., BERT or mini-transformer) for a text classification task.

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

OPERATING SYSTEM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code:	**Credits**: 04
Hours /Week: 03 Hours	**Total Hours**: 39 Hours + 26 Hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

1. **Understand** the basic concepts and functions of operating systems.
2. **Understand** Processes and Threads
3. **Analyze** Scheduling algorithms.
4. **Understand** the concept of Deadlocks.
5. **Analyze** various Memory and Virtual memory management, File system and storage techniques.
6. **Discuss** the goals and principles of protection in a modern computer system.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
OS Overview and System Structure: Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Computing environments. **Operating System Services:** User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; *Textbook – 1: 1.1 -1.10,1.12,2.1-2.8.*	
Unit 2:	**08 Hours**
Process Management: Process concept; Process scheduling; Operations on processes. Multi- threaded Programming: Overview; Multithreading models; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms. *Textbook – 1: 3.1-3.3,4.1,4.2,4.4,5.1-5.3*	
Unit 3:	**08 Hours**
Process Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors Deadlocks. **Deadlocks:** System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. ***Textbook-1: 6.1-6.7,7.1-7.7***	
Unit 4:	**08 Hours**
Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. **Virtual Memory Management**: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. ***Textbook-1: 8.1-8.6,9.1-9.6***	
Unit 5:	**7 Hours**
File System, Implementation of File System: File system: File concept; Access methods; Directory structure; File system mounting; File sharing. Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. **Mass storage structures:** Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management.	

Protection and Security:

Protection: Goals of protection, Principles of protection, System Security: The Security Problem, Program Threats, System and Network Threats.

Textbook-1: 10.1-10.6,12.1-12.6,14.1,15.1-15.3

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Illustrate** the key concepts and structures of operating systems, including process, memory, and storage management, protection, security, distributed systems, and system calls.	L2
2	**Apply** concepts of process management, including process scheduling and operations, multi-threaded programming models and issues, and process scheduling criteria and algorithms, resulting in a practical understanding of efficient process handling in operating systems.	L3
3	**Apply** concepts of process synchronization and deadlock management methods for handling, preventing, avoiding, detecting, and recovering from deadlocks.	L3
4	**Compare and contrast** various memory management techniques to enforce memory protection and manage memory allocation and deallocation efficiently.	L2
5	**Demonstrate** the various file management techniques, disk scheduling methods for efficient resource utilization and interpret the system, network, program threats and employ protection principles to safeguard the system resources.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1											2	1	0
CO2	1	1										1	1	0
CO3	1	1										1	1	0
CO4	1	1										1	1	0

CO5	1	1										1	1	0
CO6	3	2	1										2	

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2010.

Reference Books:

1. Operating Systems-Internals and Design Principles, William Stallings, 6th Edition, Pearson Education, 2009.
2. Operating Systems: A Modern Perspective, Gary J. Nutt, Addison-Wesley, 1997.

Activity Based Learning:

1. Implementing Operating System concepts by doing Mini projects.
2. Case study to compare various Operating Systems and their performance.
3. Present real-world case studies or problems related to operating systems design and performance optimization, encouraging students to apply theoretical knowledge to analyze and propose solutions.

E Resources:

1. **Modern Operating Systems** by Andrew S. Tanenbaum - Known for its comprehensive coverage of modern operating systems principles and design.
2. Operating System Fundamentals - Course (nptel.ac.in)
3. Introduction to Operating Systems - Course (nptel.ac.in)
4. Intro to Operating Systems 1: Virtualization | Coursera
5. Intro to Operating Systems 3: Concurrency | Coursera
6. Intro to Operating Systems 2: Memory Management | Coursera

Laboratory Experiments: (not Mandatory)

List of Laboratory/Practical Experiments:		
Exp. No	**Division of Experiments**	**List of Experiments**
1	System Calls	Write a C program to create a new process that exec a new program using system calls fork(), execlp() & wait()
2		Write a C program to display PID and PPID using system calls getpid () & getppid ()

3		Write a C program using I/O system calls open(), read() & write() to copy contents of one file to another file
4	Process Management	Write a C program to implement multithreaded program using pthreads
5		Write C program to simulate the Round Robin CPU scheduling algorithms
6	Process synchronization	Write a C program to simulate producer-consumer problem using semaphores
7	Deadlock	Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance.
8		
9	Memory Management	Write a C program to simulate FIFO page replacement algorithms
10		
11	I/O System	Write a C program to simulate the single level directory technique
12		Write a C program to simulate the indexed file allocation strategies.

BIG DATA ANALYTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER- VI

Course Code	**Credits**: 04
Hours /Week: 03	**Total Hours**: 39Hrs + 26 Hrs
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

1. To provide an understanding of the core concepts and properties that define Big Data, and to offer a detailed overview of the Hadoop v1 architecture along with its structural and functional challenges.
2. To explore the enhancements introduced in Hadoop v2 by analysing its architecture and to gain practical experience in data ingestion and processing through various tools in the Hadoop ecosystem.
3. To expose learners to the fundamentals of Apache Spark for scalable data processing by focusing on the creation and management of RDDs and Data Frames in distributed computing environments.
4. To enable learners to work with advanced Spark UNITs such as Spark SQL, Streaming, and MLlib, for handling real-time data flows and building analytical and machine learning applications.
5. To guide learners in the use of Hive as a data warehousing solution and to facilitate the construction and execution of structured queries for efficient large-scale data analysis.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.

5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1: INTRODUCTION TO BIG DATA AND HADOOP V1	**08 hours**
Introduction to Big Data, Characteristics (Volume, Velocity, Variety, Veracity, Value),Traditional vs Big Data systems, Google File system, GFS architecture, Hadoop Ecosystem Overview, Core Components: HDFS, MapReduce, Hadoop v1 Architecture, HDFS: Name Node, Data Node, MapReduce Programming Model, YARN in Hadoop v1 (early concepts), Limitations of Hadoop v1	
UNIT 2: HADOOP V2 AND YARN ARCHITECTURE	**08 hours**
Introduction to Hadoop v2 , YARN Architecture, Resource Manager, Node Manager, Application Master, Comparison: Hadoop v1 vs v2, Enhanced scalability and fault tolerance in v2, HDFS Federation and High Availability, Hadoop Ecosystem tools: Pig, Sqoop, Flume (overview)	
UNIT 3: APACHE SPARK FUNDAMENTALS	**08 hours**
Spark Overview, Resilient Distributed Datasets (RDD), Spark vs MapReduce, Spark Architecture, Driver, Executors, Cluster Manager, Spark Core, Spark SQL, Spark Streaming (overview), Data Frames and Datasets, Lazy Evaluation and Transformations vs Actions	
UNIT 4: ADVANCED APACHE SPARK	**07 hours**
Spark SQL and Data Frames, Schema inference, SQL queries on Data Frames, Working with Parquet, JSON, ORC files, **Spark MLlib,** ML Pipeline components: Transformers, Estimators, Pipelines, classification & regression	
UNIT 5: APACHE HIVE AND DATA WAREHOUSING	**08 hours**

Introduction to Hive, Hive Architecture: Compiler, Execution Engine, Hive Query Language (HQL),Data types, tables (managed and external), partitions, buckets, Data loading and schema evolution, Joins, Views, and Indexes in Hive, Hive vs RDBMS

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1.	Explain the characteristics of Big Data and describe the Hadoop v1 architecture and its limitations.	**L2**
2.	Compare Hadoop v1 and v2 architectures, and demonstrate data ingestion and processing using the Hadoop ecosystem tools.	**L3**
3.	Develop distributed data processing applications using Apache Spark RDDs and DataFrames.	**L3**
4.	Build and evaluate advanced Spark applications involving SQL, Streaming, and MLlib for real-time and predictive analytics.	**L4**
5.	Build data warehousing solutions using Hive and construct queries for structured data analysis.	**L4**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**	**2**		**3**								**3**	**3**
CO2	**3**	**3**	**2**		**3**								**3**	**3**
CO3	**3**	**3**	**2**		**3**								**3**	**3**
CO4	**3**	**3**	**2**		**3**								**3**	**3**
CO5	**3**	**3**	**2**		**3**								**3**	**3**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Tom White – Hadoop: The Definitive Guide, 4th Edition, O'Reilly Media ISBN: 9781491901632

2. Jules S. Damji, Brooke Wenig, Tathagata Das, and Denny Lee – Learning Spark: Lightning-Fast Data Analytics, 2nd Edition, O'Reilly Media ISBN: 9781492050049

Reference Books:

1) Dean Wampler and Jason Rutherglen – Programming Hive, O'Reilly Media
2) Arvind Sathi – Big Data Analytics: Disruptive Technologies for Changing the Game, IBM Press

Activity Based Learning:

1. Project based learning.

E Resources:

1) Apache Hadoop Documentation:- https://hadoop.apache.org/docs/
2) Apache Spark Documentation: -https://spark.apache.org/docs/latest/
3) Apache Hive Documentation:-https://cwiki.apache.org/confluence/display/Hive/Home

Lab Experiments

1. Set up and validate a Hadoop v1 environment in pseudo-distributed mode for simulating a single-node cluster.
2. Develop and execute sample programs to explore Hadoop Distributed File System (HDFS) operations and basic MapReduce functionalities.
3. Install and configure Hadoop YARN, and execute MapReduce jobs using the YARN ResourceManager on Hadoop v2 architecture.
4. Set up Apache Hive on a Hadoop v2 cluster and validate the environment by executing basic Data Definition Language (DDL) commands.
5. Develop Hive scripts to perform basic data manipulation and retrieval using HQL (SELECT, WHERE, GROUP BY, etc.).
6. Implement a Hive program to demonstrate the use of partitioning and bucketing techniques for optimizing query performance.
7. Write and execute Spark programs to demonstrate transformations and actions using RDDs and DataFrames.
8. Implement programs using Spark SQL for structured data queries and demonstrate a simple machine learning pipeline using Spark MLlib

THEORY OF COMPUTATION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code	**Credits**: 04
Hours /Week: 04	**Total Hours**: 39(L)+26(T)
L–T–P–J: 3-1-0-0	

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the Automata Theory and Formal Languages to build efficient design of FA
2. **Identify** Regular Expression and recognize the properties that make a language regular and construct the FA of the language.
3. **Devise** the technique to minimize DFA and understand the importance of minimization in optimizing automata for efficient language recognition.
4. **Get the idea** to Interpret and design different PDA for a given language
5. **Describe** the finite automata and formal languages, equipping them with the knowledge and skills necessary to analyse and design TM for language recognition tasks.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**

INTRODUCTION TO FINITE AUTOMATA:

Study and Central Concepts of Automata Theory, Finite Automata -Yet Another Method for Defining Languages, Deterministic and Nondeterministic Finite Automata, Finite Automata with Epsilon – transitions. An Application: Text Search.

(Text Book-1: Chapter 1: 1.1, 1.7, Chapter 2: 2.2 to 2.5) ***(Text Book-2: Chapter 5: Page no: 52)***	
UNIT 2:	**08 Hours**
REGULAR EXPRESSIONS AND LANGUAGES: Regular Expressions, Finite Automata and Regular Expressions, Algebraic Laws of Regular expressions, Applications of Regular Expressions, Properties of Regular Languages - Pumping Lemma, Applications of the Pumping Lemma, Closure Properties of Regular languages, Equivalence and minimization of Automata. ***(Text Book-1: Chapter 3: 3.1 to 3.4, Chapter 4: 4.1, 4.2, 4.4)***	
UNIT 3:	**09 Hours**
CONTEXT – FREE GRAMMARS AND LANGUAGES: Context-Free Grammars, Parse Trees, Ambiguity in Grammars and Languages, Properties of Context free languages-Normal Forms of Context-Free Grammars, The Pumping Lemma for Context Free Languages, Closure Properties of Context-Free Languages. ***(Text Book-1: Chapter 5: 5.1 to 5.2.3, 5.4, Chapter 7: 7.1 to 7.3)***	
UNIT 4:	**06 Hours**
PUSHDOWN AUTOMATA: Definition of the Pushdown automation (PDA), The Language of PDA, Equivalence of PDA's and CFG's-From Grammars to Push Down Automata and PDA to Grammars, Deterministic Pushdown Automata ***(Text Book-1: Chapter 6: 6.1 to 6.4)***	
UNIT 5:	**08 Hours**
INTRODUCTION TO TURING MACHINE: Problems that Computers Cannot Solve, The Turing Machine, Programming Techniques for Turing Machine, Extensions to the Basic Turing Machine. ***(Text Book-1: Chapter 8 : 8.1, 8.4)***	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	Utilize the basic concepts of formal languages of finite automata techniques such as DFA, NFA and E-NFA	L3
2	Develop Finite Automata for different Regular Expressions and Languages and minimization of Finite Automata to Regular Expression.	L3

3	Analyze context-free grammars, ambiguity and Chomsky normal form grammars to design computer languages	L3
4	Construct context free, regular, Interpret and design different PDA for a given language	L3
5	Design Turing machine to solve problems.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3									2		3
CO2	3	3	3									2		3
CO3	1	1		3	1							1		2
CO4	2	2	3	3	1							2		3
CO5	1	2	3	3	3							3		3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. J.E. Hopcroft, R. Motwani, and J. D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edn. Pearson Education , New Delhi 2008
2. Daniel I. A. Cohen, Introduction to Computer Theory, 2nd Edition, Wiley India Student Edition, 2008.

Reference Books:

1. K.L.P. Misra and N. Chandrashekaran. Theory of Computer Science- Automata, Languages and Computation, 3rd Edn. PHI, New Delhi, 2007
2. C. Martin - Introduction to Languages and the Theory of Computation 2ndEdn, TMH, New Delhi, 2000.

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solution to a problem through programming.

E Resources:

Lab Experiments

1. Design a Program for creating machine that accepts three consecutive one.
2. Design a Program for creating machine that accepts the string always ending with 101.
3. Design a Program for Mode 3 Machine
4. Design a program for accepting decimal number divisible by 2.
5. Design a program for creating a machine which accepts string having equal no. of 1's and 0's.
6. Design a program for creating a machine which count number of 1's and 0's in a given string.
7. Design a Program to find 2's complement of a given binary number.
8. Design a Program which will increment the given binary number by 1.

9. Design a Program to convert NDFA to DFA.
10. Design a Program to create PDA machine that accept the well-formed parenthesis.
11. Design a PDA to accept WCWR where w is any string and WR is reverse of that string and C is a Special symbol.
12. Design a Turing machine that's accepts the following language an b n c n where n>0.

Principles of DevOps & MLOps

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code	**Credits**:04
Hours /Week: 03	**Total Hours**: 39(L)+26 (P)hours
L–T–P–J: 3-0-2-0	

Course Learning Objectives:

1. Understand version control concepts and master Git.
2. Learn containerization with Docker.
3. Automate CI/CD pipelines using Jenkins.
4. Orchestrate and manage containers using Kubernetes.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**

GIT & VERSION CONTROL: Version Control Systems (VCS) Overview - Types of VCS: Local, Centralized, and Distributed, Benefits of using Git, Installing Git, Git Configuration (User Profile, Aliases), Basic Git Commands: git init, git clone, git status, git log, Working with Git Repositories - Creating a Repository, Staging and Committing Changes: git add, git commit, Viewing Commit History: git log, git show, git diff, Working with Remote Repositories: git remote, git fetch, git pull, git push, Forking and Pull Requests, Git Workflows: Centralized, Feature Branch

UNIT 2:	**07 Hours**

DOCKER & CONTAINERIZATION: Introduction to Containers, Containers vs Virtual Machines, Benefits of Using Docker, Setting Up Docker, Docker Architecture, Working with Docker Images, Understanding Docker Images and Containers, Docker Hub and Image Repositories, Building Docker Images: Docker file, docker build, Managing Images: docker pull, docker push, docker rmi, Running and Managing Containers, Starting and Stopping Containers: docker run, docker stop, docker start, docker restart, Inspecting Containers: docker ps, docker inspect, docker logs, Managing Container Resources (CPU, Memory).

UNIT 3:	**08 Hours**

Introduction to MLOps , The need for MLOps in AI/ML projects , Key MLOps concepts: Continuous Integration (CI), Continuous Deployment (CD), Continuous Training (CT),Overview of MLOps tools and frameworks, Introduction to MLflow - MLflow components: Tracking, Projects, Models, and Registry , Installing and setting up MLflow , MLflow CLI and UI walkthrough, Experiment Tracking with MLflow - Logging experiments: Parameters, Metrics, and Artifacts ,Tracking multiple experiments, Organizing experiments using tags , MLflow APIs for tracking in Python.

UNIT 4:	**08 Hours**

MLFLOW PROJECTS MANAGEMENT : MLflow Projects ,Understanding MLflow Projects: Structure and configuration, Packaging ML code into reusable projects, Running MLflow Projects locally and on remote servers, Using Docker environments with MLflow Projects, MLflow Models, Model flavors supported by MLflow, Saving and loading models with MLflow, Logging models using mlflow.log_model(),Model versioning and lifecycle management, Model Registry, Registering models in the MLflow Model Registry, Managing model stages: Staging, Production, Archived, Transitioning models between stages, Model lineage and auditing.

UNIT 5:	**08 Hours**

MLFLOW DEPLOYMENT: MLflow Model Deployment, Cloud Deployment options, Serving models using MLflow Models and REST APIs, Scoring and batch inference, MLflow Advanced Features, Hyperparameter tuning with MLflow, MLflow with Spark and Databricks, End-to-End MLOps Pipeline with MLflow, Building an end-to-end ML pipeline using MLflow ,Automating model training, evaluation, and deployment, Best practices for MLflow in production environments

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Understand the basics of Git and version control systems, including repository management and fundamental Git commands.	L4
2	Apply advanced Git concepts, collaboration techniques, workflows, and automation using Git hooks.	L3
3	Build a container that includes creating, managing, and working with images and containers.	L3
4	Organise networking, volumes, and multi-container applications with Docker Compose.	L3
5	Construct CI/CD principles and implement pipelines using Jenkins, integrating with Git and Docker for automated application deployment	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3		3								3	3
CO2	3	3	3		3								3	3
CO3	3	3	3		3								3	3
CO4	3	3	3		3								3	3
CO5	3	3	3		3								3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1) 1)Pro Git Scott Chacon and Ben Straub ,Publisher: Apress,Edition: 2nd Edition,Year of Publication: 2014, ISBN: 978-1484200773

2) Docker Deep Dive, Nigel Poulton, Publisher: San Francisco: Leanpub, ISBN: 978-1521822807

3) Practical MLOps by Noah Gift, Alfredo Deza , O'Reilly Media, ISBN: 978-

Reference Books:

1) Learning Git,Anna Skoulikari,Packt Publishing

2) Docker: Up & Running (3rd Edition),Sean P. Kane, Karl Matthias,O'Reilly Media

Activity Based Learning:

3) Real Word Project based learning.

E Resources:

1) Google Cloud MLOps Guide: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

2) Play with Docker (Online Playground) https://labs.play-with-docker.com/

Laboratory Experiments:

1. Install Git, configure user profile and aliases, initialize a local repo, and practice basic commands.
2. Working with Remote Repositories and Branching
3. Implementing Feature Branch Workflow - Understand remote workflows and branching models
4. Docker Image Creation and Management - Apply a feature branch workflow in a team setting.

5. Running and Managing Docker Containers - Execute, monitor, and manage container resources
6. Experiment Tracking using MLflow - Log experiments with parameters, metrics, and artifacts
7. MLflow Projects and Model Management - Build and manage ML projects
8. Model Deployment and Lifecycle Management using MLflow

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

INNOVATION AND ENTREPRENEURSHIP

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code:	**Credits**: 02
Hours /Week: 02 Hours	**Total Hours**: 26 Hours
L–T–P–J: 2-0-0-0	

Course Learning Objectives:

1. **Identify** and **analyse** the factors that contribute to the process of successfully launching an entrepreneurial venture and managing a new business.
2. **Learn** the entrepreneurial process from idea generation to implementation.
3. **Acquainted** with special problems of starting new ventures, finding products and services, which can support new enterprises, and raising capital.
4. **Discuss** how to start your own business and also to work in or with small businesses or are involved with entrepreneurship.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.

7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**05 Hours**

OVERVIEW OF ENTREPRENEURSHIP: THE ENTREPRENEURIAL PERSPECTIVE:

Nature and Development of Entrepreneurship. Defining Manager, Entrepreneur, Entrepreneurship and Entrepreneurship. Key Elements of Entrepreneurship. Personality Characteristics of Successful Entrepreneurs. Common Myths about Entrepreneurs. Ethics and Social Responsibility of Entrepreneurs. Types of Start-Up Firms. Process of New Venture Creation. Role of Entrepreneurship in Economic Development. Emerging Trends and Issues in Entrepreneurship.

(Text Book-1: Chapter 1,2), (Text Book-2: Chapter 1,2).

Unit 2:	**05 Hours**

THE ENTREPRENEURIAL AND ENTREPRENEURIAL MIND:

The Entrepreneurial Process: Identify and Evaluate the Opportunity, develop a Business Plan, Determine the Resources Required, Manage the Enterprise. Managerial Versus Entrepreneurial Decision Making: Strategic Orientation, Commitment to Opportunity, Commitment of Resources, Control of Resources, Management Structure, Entrepreneurial Venturing inside a Corporation, Causes for Interest in Entrepreneurship, Climate for Entrepreneurship, Entrepreneurial Leadership Characteristics. **(Text Book 1: Chapter 3, 15, Text Book 2: Chapter 3).**

Unit 3:	**06 Hours**

IDENTIFY AND RECOGNIZING OPPORTUNITIES:

Observing Trends and Solving Problems. Creativity: Concept, Components and Types of Creativity, Stages of Creative Process. Sources of New Venture Ideas. Techniques for Generating Ideas. Stages of Analyzing and Selecting the Best Ideas. Protecting the Idea: Intellectual Property Rights and its Components. Linking Creativity, Innovation and Entrepreneurship. **(Text Book1: Chapter 19, Text Book 2: Chapter 5)**

Unit 4:	**05 Hours**

PREPARING THE PROPER ETHICAL AND LEGAL FOUNDATION:

Initial Ethical and Legal Issues Facing a New Firm, establishing a Strong Ethical Culture, choosing an attorney (Lawyer), Drafting a founder's agreement, avoiding legal disputes, choosing a form of business organization, obtaining business licenses and permits, Choosing a Form of Business Ownership (Sole, Proprietorship, Partnership, Corporation & Limited Liability Company) **(Text Book1: Chapter 23, Text Book 2: Chapter 6)**

Unit 5:	**05 Hours**

MANAGING EARLY GROWTH AND CHALLENGES:

Recruiting and Selecting Key Employees. Lenders and Investors. Funding Requirements: Sources of Personal Financing. Venture Capital. Commercial Banks. Sources of Debt Financing. Key Marketing Issues for New Ventures. Why marketing is critical for Entrepreneurs. Entrepreneurs face unique Marketing Challenges. Guerrilla Marketing. Business Growth: Nature of Business Growth, Planning for Growth, Reasons for Growth. Managing Growth: Knowing and Managing the Stages of Growth, Challenges of Growing a Firm. Strategies for Firms Growth: Internal and External Growth Strategies. Implications of Growth for the Firm and Entrepreneur. Entrepreneurial Skills and Strategies to Overcome Pressures On: Financial Resources (Financial Control, Managing Inventory and Maintaining Good Records). Human Resources, Management of Employees, Time Management. **(Text Book1: Chapter 25, Text Book 2: Chapter 7)**

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Demonstrate knowledge of the key elements of the entrepreneurial process	L2
2	Employ strategies to generate new ideas for startups	L2
3	Outline how to protect IP legally	L2
4	Examine different ways of generating funding	L2

5	Explain organizing managing people, finance and customers	L2

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	1	1									
CO2	3	2	2										1	1
CO3	3	2	1	1	1								1	1
CO4	3	2	1	1									1	1
CO5	3	2	1		1								1	1
CO6														**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Barringer, Ireland, “Entrepreneurship: Successfully Learning New Ventures”, Pearson, Latest Edition.
2. Hisrich, Peters, Shepherd, “Entrepreneurship”, Mc Graw Hill, Sixth Edition.

Reference Books:

1. "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses" by Eric Ries
2. "The Innovator's Dilemma: When New Technologies Cause Great Firms to Fail" by Clayton M. Christensen
3. "The Art of the Start 2.0: The Time-Tested, Battle-Hardened Guide for Anyone Starting Anything" by Guy Kawasaki

Activity Based Learning:

1. Guest Speakers and Industry Visits.
2. Role play
3. Business Plan Development.

4. Flip class activity
E Resources: 1. https://archive.nptel.ac.in/courses/110/106/110106141/ 2. https://www.udemy.com/course/diploma-in-management-and-entrepreneurship/ 3. https://www.coursera.org/mastertrack/innovation-management-entrepreneurship-hec

PROFESSIONAL ELECTIVES

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

INTRODUCTION TO ARTIFICIAL INTELLIGENCE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

1. Understand the foundational principles of artificial intelligence (AI) and its applications in data science.
2. Apply machine learning and deep learning techniques to solve real-world data problems.
3. Implement data-driven decision-making models using AI algorithms.
4. Analyze, clean, and preprocess data to feed into AI models.
5. Evaluate and interpret model performance to derive business insights

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
Introduction to Artificial Intelligence: History and evolution of AI, AI definitions and types (Narrow, General, Super AI), Applications of AI across domains, Intelligent agents and environments , Problem-solving techniques: state space, search trees	
Unit 2:	**08Hours**
Search and Problem Solving: Uninformed search strategies: BFS, DFS, UCS, Informed search strategies: Greedy, A*, Hill-climbing, Constraint satisfaction problems (CSPs) , Game playing: Minimax and Alpha-Beta Pruning	
Unit 3:	**08 Hours**
Knowledge Representation and Reasoning: Propositional and First-Order Logic (FOL), Inference in logic: forward and backward chaining, Ontologies and semantic networks, Rule-based systems and expert systems, Case-based reasoning and frames	
Unit 4:	**08 Hours**
Machine Learning Fundamentals: Introduction to supervised and unsupervised learning, Decision trees, Naïve Bayes, and k-NN Clustering techniques: k-means, hierarchical, Evaluation metrics: accuracy, precision, recall, F1-score, Bias-variance trade-off and overfitting	
Unit 5:	**07Hours**
Advanced AI Topics and Applications: Neural Networks and Deep Learning basics, Natural Language Processing (NLP) fundamentals, Computer Vision: image classification and recognition, AI in Robotics and Autonomous Systems, Ethical issues in AI: bias, privacy, accountability Overview of an Expert System, Structure of an Expert Systems, Different Types of Expert Systems: Rule Based, Model Based, Case Based and Hybrid Expert Systems, Knowledge Building System Tools, Expert System Shells and Fuzzy Expert systems	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	**Describe** the fundamental concepts and applications of Artificial Intelligence.	L1
2	**Apply** search algorithms and problem-solving techniques to AI-based scenarios.	L3
3	**Develop** logic-based models for knowledge representation and	L3

	reasoning.	
4	**Implement** basic machine learning algorithms for classification and clustering.	L3
5	**Analyze** real-world AI applications and examine the ethical implications of AI systems.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2											3	3
CO2	3	2											3	3
CO3	3	2											3	3
CO4	3	2											3	3
CO5	3	2											3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. **Stuart Russell & Peter Norvig**, *Artificial Intelligence: A Modern Approach* (3rd Edition), Pearson, 2016. ISBN: 978-0136042594
2. **Elaine Rich, Kevin Knight, Shivashankar B. Nair**, *Artificial Intelligence*, 3rd Edition, McGraw Hill, 2008. ISBN: 978-0070087705

Reference Books:

1. **Tom Mitchell**, *Machine Learning*, McGraw-Hill, 1997. ISBN: 978-0070428072
2. **Nils J. Nilsson**, *The Quest for Artificial Intelligence*, Cambridge University Press, 2010. ISBN: 978-0521122931

Activity Based Learning:

1. AI Hackathon
2. Real World Projects

E Resources:

1. CS50's Introduction to Artificial Intelligence with Python (Harvard - edX) Focuses on foundational AI concepts, search, ML, and reasoning.
2. Artificial Intelligence (NPTEL – Prof. Mausam, IIT Delhi) Government-backed course covering comprehensive AI topics with lecture videos and quizzes.

NATURAL LANGUAGE PROCESSING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. To understand the algorithms available for the processing of linguistic information and computational properties of natural languages
2. To conceive basic knowledge on various morphological, syntactic and semantic NLP task
3. To understand machine learning techniques used in NLP,
4. To write programs in Python to carry out natural language processing

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

Unit 1:	**08 Hours**

Past, present, and future of NLP; Classical problems on text processing; Necessary Math concepts for NLP; Regular expressions in NLP. Parts of Speech and Morphology, Phrase Structure, Semantics and Pragmatics, Corpus-Based Work:

Getting Set Up, Looking at Text, Marked-up Data Text processing: lemmatization, stop word, tokenization, stemming, Spelling errors corrections–Minimum edit distance, Bayesian method	
Unit 2:	**09 Hours**
Words & Sentences, N-grams: Simple unsmoothed n-grams; smoothing, backoff, spelling correction using N-grams, Metrics to evaluate N-grams. Parts of Speech tagging: Word classes, POST using Brill's Tagger and HMMs; Information Extraction: Introduction to Named Entity Recognition and Relation Extraction WordNet and WordNet-based similarity measures, Concept Mining using Latent Semantic Analysis	
Unit 3:	**09 Hours**
Sequence to sequence & Language Modelling, Word embedding: skip-gram model, CBOW, GloVe, Language Modelling: Basic ideas, smoothing techniques, Language modeling with RNN and LSTM	
Unit 4:	**08 Hours**
Case studies on Generative AIs in NLT : History of generative AI,ChatGPT technical overview, Generative pre-trained Transformer – 1, Generative pre-trained Transformer – 2, Generative pre-trained Transformer – 3.	
Unit 5:	**05 Hours**
Advanced Topics and Hands-on Practices Python libraries supporting NLP; Hands-on Data collection - from social network platforms, pdfs, word files, JSON, HTML Parsing text using regular expression; scraping data from web; Text processing: convert to lowercase, remove punctuation, remove stop words, standardizing text, tokenising, stemming, lemmatising. Applications: Spam detection, consumer complaint classification, Semantic Analyser, Dialogue processing (Chatbots), Text summarization, Text Categorization.	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	Demonstrate an understanding of fundamental NLP concepts, including text processing techniques and classical problems in NLP.	L2
2	Analyze and evaluate different NLP methods and algorithms for tasks such as part-of-speech tagging and named entity recognition.	L3
3	Apply NLP techniques to real-world problems, such as spam detection and	L3

	text summarization, using Python libraries.	
4	Compare and contrast advanced NLP models, such as language models using recurrent neural networks (RNNs) and generative pre-trained transformers (GPTs).	L3
5	Synthesize their knowledge of NLP concepts and evaluate the techniques to design and develop their own NLP applications, such as chatbots or text categorization systems.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1														
CO2														1
CO3														1
CO4														1
CO5														1
CO6														**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, "Harshit Surana, Practical Natural Language Processing: A Comprehensive Guide to Building Real-World Nlp Systems" - "O'Reilly Media, Inc.", 17 Jun 2020.
2. Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more - Denis Rothman, Packt Publishing Ltd, 2021

Reference Books:

1. Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems - Aurélien Géron,Edition 2, O'Reilly Media, 2017.
2. Deep Learning for Natural Language Processing - Palash Goyal, Sumit Pandey, Karan Jain, Apress Berkeley, CA- 2018.
3. Daniel Jurafsky and James H. Martin. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. 2nd edition. Prentice-Hall.
4. Tiwary, U. S., & Siddiqui, T. (2008). Natural language processing and information retrieval. Oxford University Press, Inc.

Activity Based Learning:

1. Better Understanding the concept of Sampling and Semantic Role Labeling Quantization of Speech and using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

E Resources:

1. https://github.com/topics/nlp-models
2. https://devopedia.org/site-map/browse-articles/natural%20language%20processing
3. https://wisdomml.in/hidden-markov-model-hmm-in-nlp-python/
4. https://spotintelligence.com/2023/06/16/activation-function/
5. https://radimrehurek.com/gensim/models/word2vec.html

GENERATIVE AI SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This course will enable students to:

1. To provide a strong foundation of fundamental concepts in Generative AI.
2. To provide a basic exposition to different types of Prompt Engineering.
3. Make use of the different Generative AI models such as GPT, attention models and transformers.
4. Make use of the different Language Models for handling text data.
5. To design the Generative AI models for various applications related to handling the texand Image data.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

Unit 1:	**08 Hours**

Introduction to Generative AI, Introduction to Generative AI, Definition and scope of Generative AI, Hierarchy of Generative AI, Overview of generative models and their applications, Importance of Generative AI in various domains, Ethical considerations and challenges.

Unit 2:	**09 Hours**
Prompt Engineering: Understanding the concept and significance of prompt engineering, Principles of Prompting, Strategies for designing effective prompts, Techniques for Prompt Engineering (Template-based prompts, Rule-based prompts, and Fine-tuning prompts), Best practices for prompt engineering in generative AI, Enhancing Model Outputs.	
Unit 3:	**09 Hours**
Generative AI Concepts: Encoder/decoder architectures as basis for Generative AI, the role of the latent space, Transformer architectures and Attention, Conditional Generative Models, Introduction to GPT and its significance, Architecture and working of GPT models.	
Unit 4:	**08 Hours**
Language Models and LLM Architectures Introduction to language models and their role in AI, how do large language models work? Difference Between Large Language Models and Generative AI, Examples of LLMs (Generative Pre-trained Transformer 3, Bidirectional Encoder Representations from Transformers, Text-to-Text Transfer Transformer, Robustly Optimized BERT Pretraining Approach), Leading language models and their real-life applications.	
Unit 5:	**05 Hours**
Case Study of Generative AI and Language Models: using ChatGPT3, BERT, T5, RoBERTa; SRGAN, ESRGAN, Cycle GAN, StyleGAN, text-2-image, GAN in Computer Vision.	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	Recall the fundamental concepts of Generative AI.	L2
2	Utilize the different types of Prompt Engineering to generate the prompts.	L3
3	Make use of the different Generative AI models such as GPT, attention models and transformers to generate text and Image data.	L4
4	Make use of the different Language Models for handling text data.	L4
5	Analyze the Generative AI models for various applications related to handling the text and Image data.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	3	3								3	2
CO2	3	3	3	3	3								3	1
CO3	3	3	3	3	3								3	1
CO4	3	3	3	3	3								3	3
CO5	3	3	3	3	3								3	3
CO6	3	3	3	3	3								3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Foster, David. *Generative deep learning*. " O'Reilly Media, Inc.", 2022.
2. Dhamani, Numa. *Introduction to Generative AI*. Simon and Schuster, 2024.

Reference Books:

1. Babcock, Joseph, and Raghav Bali. *Generative AI with Python and TensorFlow 2: Create images, text, and music with VAEs, GANs, LSTMs, Transformer models*. Packt Publishing Ltd, 2021
2. Alto, Valentina. *Modern Generative AI with ChatGPT and OpenAI Models: Leverage the capabilities of OpenAI's LLM for productivity and innovation with GPT3 and GPT4*. Packt Publishing Ltd, 2023.
3. de Albuquerque, Victor Hugo C., Pethuru Raj, and Satya Prakash Yadav, eds. *Toward Artificial General Intelligence: Deep Learning, Neural Networks, Generative AI*. Walter de Gruyter GmbH & Co KG, 2023.

Activity Based Learning:

1. Flipped Class Activity on Searching techniques.
2. Problem Solving and Discussion.
3. GPT (Generative Pre-trained Transformer) Pre-training and fine-tuning processes.
4. Mini Project

E Resources:

1. https://www.datacamp.com/blog/what-is-prompt-engineering-the-future-of-ai-communication
2. https://www.promptengineering4u.com/learning/techniques/template-based-prompting#h.2n56pv37pv0c

AGENTIC AI

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This course will enable students to:

1. Apply Agentic AI principles to real-world commercial problems.
2. Architect Agentic solutions using proven design patterns.
3. Connect Large Language Models (LLMs) to collaborate using tools, structured outputs, and memory.
4. Create autonomous Agentic applications with frameworks like CrewAI and OpenAI Agents SDK.
5. Build robust and repeatable Agentic solutions with Lang Graph and explore advanced tools like AutoGen AgentChat.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
INTRODUCTION TO AGENTIC AI : Definition and scope of Agentic AI, Differentiating Agentic AI from traditional AI and LLM-based tools, Types of agents: Reactive, Proactive, Reflective, Cooperative, Characteristics of Agentic systems (autonomy, memory, tool use, self-direction),Use cases in business, productivity, education, software development, etc.,The future of Agentic AI, commercial applications and industry adoption	
Unit 2:	**09 Hours**
ARCHITECTING AGENTIC AI: System design patterns in Agentic AI, Architectural components: Agents, tools, orchestrators, environment, Best practices in memory management and long-context prompting, Implementing structured outputs with JSON, Pedantic, OpenAI Function calling, Tool use: Function-calling, retrieval, APIs, and computation integration, Memory types: Short-term memory, long-term memory, and context management, Modular and reusable Agentic pipeline design	
Unit 3:	**09 Hours**
DEVELOPING WITH CREWAI AND OPENAI AGENTS SDK: Overview of Crew AI framework and architecture, Building agent teams using CrewAI (Roles, Goals, and Tasks),Agent lifecycle in CrewAI (initialize, collaborate, finalize),Writing agents that can generate and execute Python code, Rapid prototyping using OpenAI Agents SDK (task definition, tool binding, environment config),Integrating tools, memory, and agent chaining with OpenAI SDK, Debugging and logging Agentic workflows for performance analysis	
Unit 4:	**08 Hours**
ADVANCED AGENTIC FRAMEWORKS AND LANG GRAPH : LangGraph overview and use cases in multi-agent workflows, Building dynamic workflows using LangGraph's node and edge patterns, Error handling and retry strategies using LangGraph, Integrating LangChain tools and memory with LangGraph, Introduction to AutoGen AgentChat and AutoGen Core by Microsoft, Multi-agent conversation graphs and control flow with AutoGen	
Unit 5:	**05 Hours**
OPEN-SOURCE ECOSYSTEM AND FUTURE OF AGENTIC AI : Overview of the open-source Agentic AI ecosystem, Anthropic's Model Context Protocol (MCP) and interoperability, Open-source projects: CrewAI, LangGraph, AutoGen, Semantic Kernel, Haystack, Guidance, Licensing and deployment: Open-source vs closed-source agents, Ethical, legal, and security implications of autonomous agents, Performance benchmarking and evaluation metrics for Agentic systems	
At the end of the Course the Students will be able to	

Course	Description	Bloom's Taxonomy Level

Outcome		
1	Understand Agentic AI methodologies to analyze and solve real-world commercial problems using intelligent agent systems.	L2
2	Illustrate Agentic AI architectures using established design patterns to create scalable and modular agent-based solutions.	L2
3	Analyse LLMs with tools, structured outputs, and memory components to enable collaborative agent behavior.	L4
4	Develop autonomous AI applications using CrewAI and OpenAI Agents SDK by configuring agents with goals, tools, and reasoning logic.	L4
5	Construct resilient and repeatable Agentic workflows using LangGraph and evaluate advanced multi-agent strategies with AutoGen AgentChat.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	2	3								3	3
CO2	3	3	3	2	3								3	3
CO3	3	3	3	2	3								3	3
CO4	3	3	3	2	3								3	3
CO5	3	3	3	2	3								3	3
CO6	3	3	3	2	3								3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1) **Agentic Artificial Intelligence: Harnessing AI to Reinvent Business, Work, and Life** Tom Davenport et al. Amazon Publishing
2) Building Agentic AI Systems, Packt Publishing, 2024

Reference Books:

1) Mastering LangGraph: A Hands-On Guide to Building Complex Multi-Agent Applications, 2024

2) Crew AI in Action: A Practical Guide to Building and Managing Multi-Agent Systems, Amazon Technical Series, 2024

Activity Based Learning: 1. Flipped Class Activity
E Resources: 1) O**penAI Agents SDK Documentation** https://openai.github.io/openai-agents-python 2) M**icrosoft AutoGen Agent Chat Guide** https://microsoft.github.io/autogen/stable//user-guide/agentchat-user-guide/

ARTIFICIAL GENERAL INTELLIGENCE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This course will enable students to:

1. Understand the foundational principles and goals of AGI.
2. Explore different theoretical models and architectures proposed for AGI.
3. Analyze the cognitive, computational, and philosophical challenges in achieving AGI.
4. Evaluate the ethical, safety, and societal implications of AGI development.
5. Gain hands-on experience in AGI-related research paradigms such as symbolic reasoning, neural-symbolic integration, meta-learning, and cognitive architectures.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
Introduction to AGI: Definition and scope of AGI, History and key milestones, Narrow AI vs AGI vs Superintelligence Human-level intelligence and benchmarks	
Unit 2:	**09 Hours**
Cognitive Architectures: Overview of cognitive science models, SOAR, ACT-R, Sigma, OpenCog, Memory, perception, learning, and action systems	
Unit 3:	**09 Hours**
Approaches to AGI: Symbolic vs sub-symbolic AI, Neural-symbolic integration, Evolutionary computation, genetic programming, Probabilistic models and Bayesian reasoning, Meta-learning and self-improving systems	
Unit 4:	**08 Hours**
Reasoning, Learning, and Planning: General problem solving, Reinforcement learning and AGI, Hierarchical learning Causal reasoning and abstraction	
Unit 5:	**05 Hours**
Embodied and Situated AGI: Robotics and embodied cognition, AGI in dynamic environments, Perception-action loops, **Safety, Ethics, and Future of AGI:** Control problems and value alignment, AI safety frameworks (e.g., CIRL, IRL), AGI policy, governance, and existential risks, Long-term trajectories and societal impacts	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Understand Agentic AI methodologies to analyze and solve real-world commercial problems using intelligent agent systems.	L2
2	Illustrate Agentic AI architectures using established design patterns to create scalable and modular agent-based solutions.	L2

3	Analyse LLMs with tools, structured outputs, and memory components to enable collaborative agent behaviour.	**L3**
4	**Develop** autonomous AI applications using CrewAI and OpenAI Agents SDK by configuring agents with goals, tools, and reasoning logic.	**L3**
5	**Construct** resilient and repeatable Agentic workflows using LangGraph and evaluate advanced multi-agent strategies with AutoGen AgentChat.	**L3**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**	**3**	**2**	**3**								**2**	**2**
CO2	**3**	**3**	**3**	**2**	**3**								**2**	**2**
CO3	**3**	**3**	**3**	**2**	**3**								**2**	**2**
CO4	**3**	**3**	**3**	**2**	**3**								**2**	**2**
CO5	**3**	**3**	**3**	**2**	**3**								**2**	**2**
CO6	**3**	**3**	**3**	**2**	**3**								**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. **Goertzel, Ben**, and Pennachin, Cassio (Eds.). *Artificial General Intelligence*. Springer, 2007.
2. **Wang, Pei**. *On Artificial General Intelligence: A Philosophical and Theoretical Approach*. Springer, 2019.

Reference Books:

1. Russell, Stuart, and Norvig, Peter. *Artificial Intelligence: A Modern Approach*, 4th ed. (Chapters relevant to general intelligence).
2. Kurzweil, Ray. *The Singularity is Near*.
3. Bostrom, Nick. *Superintelligence: Paths, Dangers, Strategies*.
4. Schmidhuber, Jürgen. *Formal Theory of Creativity, Fun, and Intrinsic Motivation (2009)*.

Activity Based Learning:

1. Paper Presentation
2. Mini Project

E Resources:

1. Lecture slides and videos (hosted on LMS/Moodle or YouTube)
2. Research paper repository (IEEE, ArXiv, SpringerLink)
3. Simulation tools (e.g., NARS, OpenCog, AERA, Pyro)
4. MOOCs (e.g., AGI course by Ben Goertzel, Stanford's AI safety courses)

MINING OF MASSIVE DATASETS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

Course Learning Objectives:

This Course will enable students to:

1. Introduce the basics of data mining, including statistical modeling, machine learning, and computational approaches.
2. Learn to extract relevant features from data for improved model performance.
3. Study the stream data model and techniques for sampling, filtering, and counting in data streams.
4. Understand market-basket analysis and algorithms for discovering frequent item sets.
5. Understand eigenvalues, eigenvectors, PCA, and SVD for reducing data dimensionality.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**
DATA MINING: Introduction, Statistical Modeling, Machine Learning, Computational Approaches to Modeling, Feature Extraction, Statistical Limits on Data Mining, Hash Functions, Indexes, Natural Logarithms, Power Laws.	
Unit 2:	**09 Hours**
MAP REDUCE AND THE NEW SOFTWARE STACK: Distributed File Systems, Map Reduce, Algorithms Using MapReduce, Extensions to MapReduce, Complexity Theory for MapReduce	
Unit 3:	**09 Hours**
MINING DATA STREAMS: The Stream Data Model, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Counting Ones in a Window, Decaying Windows.	
Unit 4:	**08 Hours**
FREQUENT ITEM SETS: The Market-Basket Model, Market Baskets and the A-Priori Algorithm, Handling Larger Datasets in Main Memory, Limited-Pass Algorithms, Counting Frequent Items in a Stream	
Unit 5:	**05 Hours**
CLUSTERING: Introduction to Clustering Techniques, Hierarchical Clustering, K-means Algorithms, The CURE Algorithm, Clustering in Non-Euclidean Spaces, and Clustering for Streams and Parallelism. Dimensionality Reduction: Eigenvalues and Eigenvectors of Symmetric Matrices, Principal-Component Analysis, Singular-Value Decomposition	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Recollecting fundamentals of data mining.	L2
2	**Apply** the concept of Map reduce and data streams for storing and processing massive data sets.	L3
3	**Analyze** the issues underlying the effective applications of massive datasets	L3
4	Applying the A-Priori Algorithms for Market Basket Analysis	L3
5	**Evaluate** different clustering algorithms and analyze various decomposition	L3

	techniques	

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2											3	2
CO2	2	2											3	2
CO3	2	2											3	2
CO4	2	2											3	2
CO5	2	2											3	2
CO6	2	2											3	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Mining of Massive Datasets" by Jure Leskovec, Anand Rajaraman, and Jeffrey Ullman
2. Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei

Reference Books:

1. "The Elements of Statistical Learning: Data Mining, Inference, and Prediction" by Trevor Hastie, Robert Tibshirani, and Jerome Friedman
2. "Introduction to Information Retrieval" by Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze

Activity Based Learning:

1. Real world problem solving using group discussion.
2. Dimensionality Reduction with Johnson-Lindenstrauss Lemma
3. Random Graphs and Giant Components

E Resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs06/preview
2. https://online.stanford.edu/courses/soe-ycs0007-mining-massive-data-sets https://www.udemy.com/topic/data-structures/free/
3. https://www.udemy.com/course/information-retrieval-and-mining-massive-data-sets/?couponCode=ST16MT70224

MASSIVE GRAPH ANALYSIS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. Develop a strong foundation in graph theory and network science, including key concepts, terminologies, and fundamental algorithms.
2. Gain proficiency in handling and processing massive graphs using distributed computing frameworks and graph databases, addressing challenges associated with large-scale graph data
3. Learn and implement various graph mining techniques to discover patterns, perform link prediction, and detect anomalies within graph data
4. Understand and apply advanced graph embedding techniques and graph neural networks (GNNs) to solve complex problems in data science, such as node classification and link prediction
5. Conduct comprehensive analyses of real-world networks in various domains, such as social networks, biological networks, and knowledge graphs, to derive actionable insights and develop practical applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students'

understanding.

Unit 1:	**08 Hours**

INTRODUCTION AND APPLICATION OF LARGE SCALE GRAPHS:
Characteristics, Complex Data Sources Social Networks, Simulations, Bioinformatics; Categories-Social, Endorsement, Location, Co-occurrence graphs; Graph Data structures, Parallel, Multicore, &Multithreaded Architectural Support for Graph Processing, Mapping Graph Algorithms to Architectures.

Unit 2:	**09 Hours**

. **BASIC AND ADVANCED LARGE SCALE GRAPH ANALYSIS**:
Parallel Prefix & List Ranking, Link Analysis, Page Ranking Algorithms; Parallel BFS, Spanning Tree, Connected Components, Minimum Spanning Tree Matroid Algorithms, Social Networking Algorithms, Parallel Betweenness Centrality

Unit 3:	**09 Hours**

DYNAMIC PARALLEL ALGORITHMS:
Streaming Data Analysis- Data Structures for Streaming Data Tracking Clustering Coefficients-Tracking Connected Components-Anomaly Detection, Massive-Graphs in Computational Biology, Genome Assembly

Unit 4:	**08 Hours**

DISTRIBUTED COMPUTATION FOR MASSIVE DATA SETS:
Spectral, Modularity-based Clustering, Random Walks; Large Graph Representation and Implementation-V-Graph Representation, Map Reduce, Surfer, Graph Lab.

Unit 5:	**05 Hours**

ADVANCED TOPICS:
Power Law Distribution, Game-Theoretic Approach, Rank Aggregation and Voting Theory, Recommendation Systems, Social network analysis: case study-Facebook, LinkedIn, Google+, and Twitter.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Explain the introduction, applications, and categories of large-scale graphs, including their data structures and architectural support for processing.	L2
2	Perform basic and advanced large-scale graph analysis, including parallel algorithms for link analysis, spanning trees, and social networking.	L3

3	Describe and implement dynamic parallel algorithms for streaming data analysis and anomaly detection in massive graphs, particularly in computational biology	L3
4	Analyze and implement distributed computation methods and large graph representations using techniques such as spectral clustering, MapReduce, and Graph Lab	L3
5	Apply advanced topics in graph theory, including power law distribution, rank aggregation, and recommendation systems, to real-world social network analysis case studies.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**											**3**	**2**
CO2	**3**	**2**											**3**	**2**
CO3	**3**	**2**											**3**	**2**
CO4	**3**	**2**											**3**	**2**
CO5	**3**	**2**											**3**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. MatthewO.Jackson,"SocialandEconomicNetworks",PrincetonUniversityPress, 2010.
2. StanleyWasserman, Katherine Faust, "Social Network Analysis: Methods and Applications", (StructuralAnalysis intheSocialSciences),CambridgeUniversity Press,1995.
3. TanjaFalkowski,"CommunityAnalysisinDynamicSocialNetworks",(Dissertation), UniversityMagdeburg, 2009.

Reference Books:

1. LadislavNovak, AlanGibbons, "HybridGraphTheory andNetworkAnalysis", CambridgeTractsin Theoretical Computer Science, 2009.
2. EricD. Kolaczyk, "Statistical Analysis of Network Data Methods and Models", Springer Series in Statistics, 2009.

Activity Based Learning:

1. Graph Construction and Visualization
2. Analyse a network dataset (e.g., transaction network) to identify unusual patterns or anomalies using algorithms like LOF (Local Outlier Factor).
3. Random Graphs and Giant Components

E Resources:

1. https://www.udemy.com/course/grokking-graph-analytics-and-algorithms/?couponCode=ST16MT70224
2. https://www.coursera.org/learn/big-data-graph-analytics
3. https://onlinecourses.nptel.ac.in/noc20_cs92/preview

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

DATA DRIVEN RECOMMENDATIONS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. **Understand** the basic taxonomy and types of recommender systems (RSs).
2. **Learn** about collaborative filtering techniques in recommender systems.
3. **Gain** knowledge about content-based recommender systems and their advantages and drawbacks.
4. **Understand** knowledge-based recommendation techniques, including knowledge representation and reasoning.
5. **Explore** the applications of RSs in content media, social media, and communities.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

INTRODUCTION:

Introduction and basic taxonomy of recommender systems (RSs). Traditional and non-personalized RSs. Overview of data mining methods for recommender systems. Understanding ratings, Overview of convex and linear optimization principles. Applications of recommendation systems, Issues with recommender systems.

Unit 2:	**09 Hours**

COLLABORATIVE FILTERING:
User-based nearest neighbor recommendation, Item based nearest neighbor recommendation, Model based and pre-processing based approaches, Attacks on collaborative recommender systems.

Unit 3:	**09 Hours**

CONTENT BASED RECOMMENDER SYSTEM:
The long-tail principle. Domain-specific challenges in recommender systems. Content-based recommender systems. Advantages and drawbacks. Basic components of content-based RSs. Feature selection. Item representation Methods for learning user profiles.

Unit 4:	**08 Hours**

KNOWLEDGE BASED RECOMMENDATION:
Knowledge representation and reasoning, Constraint based recommenders, Case based recommenders.
HYBRID APPROCHES: Opportunities for hybridization, Monolithic hybridization design: Feature combination, Feature augmentation. Parallelized hybridization design: Weighted, Switching, Mixed, Pipelined hybridization design: Cascade Meta-level, Limitations of hybridization strategies.

Unit 5:	**05 Hours**

EVALUATING RECOMMENDER SYSTEM:
General properties of evaluation research, Evaluation designs, Evaluation on historical datasets, Error metrics, and decision-Support metrics. User-Centered metrics.
Case studies: Netflix, Amazon, YouTube, LinkedIn, **The Netflix data challenge.**

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Apply** the key C programming concepts such as pointers, structures and unions to build data structures. Implement arrays and perform operations such as insertion, deletion, searching, sorting, and traversing.	L3
2	**Apply** the fundamental concepts of stacks and queues to solve real-world problems.	L3

3	**Implement** Singly Linked List, Doubly Linked List, and Circular Linked Lists to address a variety of problems**.**	L3
4	**Implement** Non-linear tree data structure and perform operations on it. **Develop** critical thinking and problem-solving skills by designing and implementing efficient algorithms for tree-related tasks.	L3
5	**Apply** advanced techniques, such as balancing algorithms for AVL trees, Splay trees and Red-Black trees to maintain the balance and efficiency of binary trees.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**											**2**	**2**
CO2	**3**	**3**											**2**	**2**
CO3	**3**	**3**											**2**	**2**
CO4	**3**	**3**											**2**	**2**
CO5	**3**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Jannach D., Zanker M. and FelFering A., Recommender Systems: An Introduction, Cambridge University Press (2011), 1st ed.

Reference Books:

1. Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer(2011), 1st ed.
2. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer (2013), 1st edition.

Activity Based Learning:

E Resources:

1. https://www.iteratorshq.com/blog/an-introduction-recommender-systems-9-easy examples/#:~:text=Netflix%2C%20YouTube%2C%20Tinder%2C%20and,News%20Website
2. https://recostream.com/blog/amazon-recommendation-systems
3. https://towardsdatascience.com/deep-dive-into-netflixs-recommender-system 341806ae3b48

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

IMMERSIVE DATA SCIENCE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. To **gain** the knowledge of historical and modern overviews and perspectives on virtual reality.
2. To **learn** the fundamentals of sensation, perception, and perceptual training.
3. To have the scientific, technical, and engineering aspects of augmented and virtual reality systems.
4. To **learn** the Evaluation of virtual reality from the lens of design.

To **learn** the technology of augmented reality and implement it to have practical knowledge.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

INTRODUCTION:

Introduction to Augmented-Virtual and Mixed Reality, Taxonomy, technology and features of augmented reality, difference between AR ,VR and MR, Challenges with AR, AR systems and functionality, Augmented reality methods, visualization techniques for augmented reality.

Unit 2:	**09 Hours**

VR SYSTEM:
VR as a discipline, Basic features of VR systems, Architecture of VR systems, VR hardware: VR input hardware: tracking systems, motion capture systems, data gloves, VR output hardware: visual displays.

Unit 3:	**09 Hours**

VR SOFTWARE DEVELOPMENT:
Challenges in VR software development, Master/slave and Client/server architectures, Cluster rendering, Game Engines and available sdk to develop VR applications for different hardware (HTC VIVE, Oculus, Google VR).

Unit 4:	**08 Hours**

AR SOFTWARE DEVELOPMENT:
AR software, Camera parameters and camera calibration, Marker-based augmented reality, AR Toolkit

Unit 5:	**05 Hours**

APPLICATION:
Application of VR in Digital Entertainment: VR Technology in Film & TV Production. VR Technology in Physical Exercises and Games. Demonstration of Digital Entertainment by VR.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Identify, examine, and develop software that reflects fundamental techniques for the design and deployment of VR and AR experiences.	L2,L3
2	Describe how VR and AR systems work	L3
3	Choose, develop, explain, and defend the use of particular designs for AR and VR Experiences.	L2
4	Evaluate the benefits and drawbacks of specific AR and VR techniques on the huma n body.	L2
5	Identify and examine State-of-art AR and VR design problems and solutions from	L2, L3

	the industry and academia	

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3	2									2	3	1
CO2	2	2	3									2	2	3
CO3	2	3	2									2	3	1
CO4	1	2	3									1	2	3
CO5	2	1	3									2	2	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. George Mather, Foundations of Sensation and Perception:Psychology Press; 2 edition, 2009
2. The VR Book: Human-Centered Design for Virtual Reality, by Jason Jerald
3. Learning Virtual Reality by Tony Parisi, O' Reilly

Reference Books:

4. Burdea, G. C. and P. Coffet. Virtual Reality Technology, Second Edition.Wiley-IEEE Press, 2003/2006.

Alan B. Craig, Understanding Augmented Reality, Concepts and Applications, Morgan Kaufmann, 2013.

Activity Based Learning:

1. AR/VR Prototyping.
2. AR/VR Immersive Experiences

E Resources:

1. http://msl.cs.uiuc.edu/vr/
2. Unity Learn: https://learn.unity.com/
3. Coursera: https://www.coursera.org/
4. Oculus Developer Center: https://developer.oculus.com/

STREAMING ANALYTICS [As per Choice Based Credit System (CBCS) scheme] **SEMESTER – VII**	
Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. Comprehend the processing and application of very large datasets.
2. Develop the knowledge and abilities to handle very large datasets and continuous streaming data that require real-time processing.
3. Gain practical experience with technologies that facilitate the ingestion and management of Big Data and real-time data.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

INTRODUCTION TO STREAM COMPUTING :

Streaming Data – Sources – Difference between Streaming Data and Static Data. Overview of Large Scale Stream Processing Engines – Issues in Stream Processing.

Unit 2:	**09 Hours**

STREAMING ANALYTICS ARCHITECTURE:
Phases in Streaming Analytics Architecture - Vital Attributes - High Availability – Low Latency – Horizontal Scalability-Fault Tolerance - Service Configuration and Management - Apache ZooKeeper

Unit 3:	**09 Hours**

DATA FLOW MANAGEMENT :
Distributed Data Flows – At Least One Delivery – Apache Kafka – Apache Flume – Zero MQ - Messages, Events, Tasks

Unit 4:	**08 Hours**

PROCESSING & STORING STREAMING DATA :
Distributed Stream Data Processing: Co-ordination, Partition and Merges, Transactions. Duplication Detection using Bloom Filters - Apache Spark Streaming Examples Choosing a storage system – NoSQL Storage Systems

Unit 5:	**05 Hours**

DELIVERING STREAMING METRICS:
Visualizing Data – Mobile Streaming Apps –Times Counting and Summation - Stochastic Optimization.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Explain the need for stream computing	L3
2	Comprehend the architecture of stream analytics	L3
3	Build data flow management pipelines for streams.	L3
4	Process the streaming data	L3
5	Deliver the results of streaming analytics	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3											2	2
CO2	3	3											2	2

CO3	**3**	**3**											**2**	**2**
CO4	**3**	**3**											**2**	**2**
CO5	**3**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Byron Ellis, "Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data", Wiley, 1st edition, 2014.
2. SherifSakr, "Large Scale and Big Data: Processing and Management", CRC Press, 2014. 2014.

Reference Books:

1. Bill Franks, "Taming The Big Data Tidal Wave Finding Opportunities In Huge Data Streams With Advanced Analytics", Wiley, 2012.

Activity Based Learning:

1. LiveData Analysis Case Study.
2. WEB log Data Analysis Case Study

E Resources:

1. kafka.apache.org
2. flume.apache.org
3. zookeeper.apache.org
4. spark.apache.org

SENSOR ANALYTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. **Understand** the benefits and shortcomings of various sensing systems used for automotive applications.
2. Ability to **apply** appropriate data fusion techniques to problems in automotive applications.
3. Ability to **analyse** the intelligent fusion algorithms for automotive applications.
4. Ability to **create** fusion models for state estimation and localization for automotive applications

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.

1. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the

students' understanding.

Unit 1:	**08 Hours**

INTRODUCTION TO IOT:

What is IoT, Genesis of IoT, IoT and Digitization, IoT Impact, Convergence of IT and IoT, IoT Challenges, A Simplified IoT Architecture, The Core IoT Functional Stack, IoT Data Management and Compute Stack.

Smart Objects: The "Things" in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, Communications Criteria, IoT Access Technologies

Unit 2:	**09 Hours**

IOT PHYSICAL DEVICES AND ENDPOINTS:

Introduction to Arduino and Raspberry Pi- Installation, Interfaces (serial, SPI, I2C) Controlling Hardware- Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, unipolar and bipolar Stepper motors

Unit 3:	**09 Hours**

SENSORS: Types of sensors, Sensor Technologies: Basics of Camera, LIDAR, RADAR sensors – Sensor Positioning – Sensor Calibration

Example — simulation of Point Cloud Data Sensing Algorithms – Automated Driving Systems – Mapping – Connectivity – Use of Artificial Intelligence for Autonomous Driving.

Example — Comparing typical autonomous vehicle sensor sets including Tesla, Uber and Mercedes.

Unit 4:	**08 Hours**

DATA FUSION MODELS:

Configurations and architectures – Probabilistic Data Fusion- Dempster-Shafer Method- Maximum Likelihood – Least-squares method, Maximum Entropy methods – Recursive Bayesian methods.

Unit 5:	**05 Hours**

STATE ESTIMATION:

Estimation and Localization for Self-driving cars: Use of Kalman filter variants – Information filtering – H∞ filtering. **GNSS/INS sensing** for position and orientation estimation – Basics of LIDAR sensing – Fusion of sensor data for an autonomous Vehicle State Estimator.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Develop a comprehensive understanding of IoT, including its impact. Challenges and core functional stack	L3
2	Acquire knowledge of different sensor types, technologies and their role in automated driving systems	L3
3	Gain an understanding of data fusion models and their application in probabilistic data fusion	L3
4	Learn state estimation and localization techniques, including the use of Kalman filter variants and fusion of GNSS/INS and LIDAR sensing	L3
5	Understand the process of sensor data fusion for accurate estimation of autonomous vehicle states	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**											**2**	**2**
CO2	**3**	**3**											**2**	**2**
CO3	**3**	**3**											**2**	**2**
CO4	**3**	**3**											**2**	**2**

CO5	**3**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry,"IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1st Edition, Pearson Education (Cisco Press Indian Reprint). (ISBN: 978-9386873743), 2017.
2. David L. Hall, Sonya A.H. McMullen, Mathematical Techniques in Multisensor Data Fusion, Second Edition, Artech House, Boston, 2004.

Reference Books:

1. Computer Vision: Algorithms & Applications, R. Szeleski, Springer, 2nd Edition 2022.
2. Gonzalez, R. C. and Woods, R. E., "Digital Image Processing", Prentice Hall, 3rdEd, 2009.
3. Trucco, E. and Verri, A., "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998.

Activity Based Learning:

1. Image Filtering Experiment.
2. Image Segmentation Challenge.
3. Object Detection Project

E Resources:

1. https://archive.nptel.ac.in/courses/106/105/106105216/
2. https://opencv.org/
3. https://www.tensorflow.org/
4. https://pytorch.org/

VISION ANALYTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

2. **Develop** an understanding of image processing techniques and their applications in computer vision.
3. **Explore** the fundamentals of image formation, including geometric primitives, photometric image formation, and the functioning of digital cameras.
4. **Gain** knowledge of 3D vision concepts such as feature detection, segmentation, pose estimation, and 3D reconstruction.
5. **Understand** the principles of video analytics and its various applications, including real-time security, user insights, and in-store performance improvement.
6. **Explore** the wide range of applications of computer vision, including image processing, machine learning, information retrieval, neuroscience, robotics, speech, cognitive sciences, graphics, algorithms, systems, theory, and pattern recognition.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

2. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
3. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
4. Show ***Video/animation*** films to explain functioning of various concepts.

5. Encourage ***Collaborative*** (Group Learning) Learning in the class.
6. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
7. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
8. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
9. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

COMPUTER VISION FOUNDATIONS:

Image Processing - Colour - Linear Algebra Primer - Pixels and Filters - Edge Detection - Features and Fitting - Feature Descriptors - Image Resizing - Segmentation - Semantic Segmentation - Clustering - Object recognition - Dimensionality Reduction: Face Identification, Visual Bag of Words - Object Detection from Deformable Parts - Semantic Hierarchies and Fine Grained Recognition - Motion - Tracking.

Unit 2:	**09 Hours**

IMAGE FORMATION:

Geometric primitives and transformations – Photometric image formation – The digital camera – Point operators – Linear Filtering – More neighbourhood operators – Fourier transforms – Pyramids and wavelets – Geometric transformations – Global optimization.

Unit 3:	**09 Hours**

3D VISION:

Feature detection and matching – Segmentation – Edge detection - 2D and 3D feature based alignment – Pose estimation – Geometric intrinsic calibration – Triangulation – Two-Frame Structure from motion. Methods for 3D

Vision - 3D reconstruction – Image based rendering, Image Recognition – Object Detection – Space, Instance

Unit 4:	**08 Hours**

VIDEO ANALYSIS- Video Processing – use cases of video analysis-Vanishing Gradient and exploding gradient problem RestNet architecture-RestNet and skip connections-Inception Network-GoogleNet architecture Improvement in Inception v2-Video analytics-RestNet

Unit 5:	**05 Hours**

VIDEO-BASED RENDERING AND RECOGNITION

Video based rendering-Object detection - Face recognition - Instance recognition - Category recognition - Context and scene understanding- Recognition databases and test sets

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Apply linear algebra principles to solve computer vision problems	L3
2	Analyze and evaluate the components and working principles of a digital camera	L3
3	Apply segmentation algorithms to partition images into meaningful regions	L3
4	Analyze and evaluate different parameters used in video analytics	L3.L3
5	To design and develop innovative video processing and computer vision applications	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**											**2**	**2**

CO2	**3**	**3**											**2**	**2**
CO3	**3**	**3**											**2**	**2**
CO4	**3**	**3**											**2**	**2**
CO5	**3**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Forsyth, D. A. and Ponce, J., "Computer Vision: A Modern Approach", Prentice Hall, 2nd Ed. 2011.
2. Szeliki, R., "Computer Vision: Algorithms and Applications", Springer, 2011.
3. Hartley, R. and Zisserman, A., "Multiple View Geometry in Computer Vision", Cambridge University Press, 2003.

Reference Books:

4. Computer Vision: Algorithms & Applications, R. Szeleski, Springer, 2nd Edition 2022.
5. Gonzalez, R. C. and Woods, R. E., "Digital Image Processing", Prentice Hall, 3rdEd, 2009.
6. Trucco, E. and Verri, A., "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998.

Activity Based Learning:

4. Image Filtering Experiment.
5. Image Segmentation Challenge.
6. Object Detection Project

E Resources:

5. https://archive.nptel.ac.in/courses/106/105/106105216/
6. https://opencv.org/
7. https://www.tensorflow.org/
8. https://pytorch.org/

BIO INFORMATICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. Equip students with the ability to navigate and utilize major biological databases for retrieving and analyzing genetic and protein data.
2. Develop proficiency in using essential bioinformatics tools and software for sequence alignment, genome analysis, and data interpretation.Apply various distributed algorithms for mutual exclusion, deadlock detection, consensus, and fault tolerance.
3. Teach students to implement and apply computational algorithms and statistical methods for sequence alignment, phylogenetic analysis, and data mining.
4. Enable students to understand and analyze gene expression data, including techniques for gene expression profiling and functional genomics.
5. Foster awareness and understanding of the ethical, legal, and social implications of bioinformatics research, emphasizing data privacy, security, and ethical conduct

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.

Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students'

understanding.	
Unit 1:	**08 Hours**
INTRODUCTION TO BIOINFORMATICS: Overview of Bioinformatics**:** Definition and history**,** Applications in various fields**.** Biological Databases**:** Types of databases: Nucleotide, Protein, Structure**.** Key databases: GenBank, EMBL, DDBJ, UniProt, PDB**.** Data Retrieval Systems**:** NCBI, EBI, ExPASy**,** Sequence retrieval methods	
Unit 2:	**09 Hours**
MOLECULAR BIOLOGY FOR BIOINFORMATICS: DNA, RNA, and Protein Structure**,** Nucleotides and nucleic acids**,** Protein structure: primary, secondary, tertiary, quaternary**.** Gene Expression and Regulation**:** Transcription and translation**,** Regulatory elements and gene control. Techniques in Molecular Biology: PCR, Gel electrophoresis, DNA sequencing methods Genomics and Proteomics: Genome projects, Techniques in proteomic	
Unit 3:	**09 Hours**
COMPUTATIONAL METHODS IN BIOINFORMATICS: Sequence Alignment**:** Pairwise alignment: Needleman-Wunsch, Smith-Waterman algorithms, Multiple sequence alignment: Clustal W, MUSCLE. Phylogenetics: Evolutionary trees, Tree-building methods: UPGMA, Neighbor-Joining, Structural Bioinformatics: Protein structure prediction, Homology modeling	
Unit 4:	**08 Hours**
GENOMICS AND FUNCTIONAL GENOMICS: Genome Sequencing Technologies**:** Next-generation sequencing (NGS) ,Whole-genome sequencing. Functional Genomics: Gene function prediction, Gene expression profiling: microarrays, RNA-Seq. Comparative Genomics: Comparative analysis of genomes, Evolutionary insights Systems Biology: Network biology, Pathway analysis	
Unit 5:	**05 Hours**
BIOINFORMATICS TOOLS AND APPLICATIONS: Software Tools and Resources**,** Bioinformatics software: Bioconductor, Cytoscape**,** Online tools and resources**.** Applications in Medicine: Personalized medicine, Drug discovery and development. Ethical, Legal, and Social Issues	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level

1	Proficiently navigate and use key biological databases to retrieve and analyze genetic and protein data	L2
2	Effectively use bioinformatics tools like BLAST and FASTA for sequence alignment and analysis.	L3
3	**Implement** and utilize algorithms for sequence alignment and phylogenetic analysis.	L3
4	**Analyze** gene expression data and understand gene regulation processes.	L3
5	**Understand** and address ethical, legal, and social issues in bioinformatics, including data privacy and the implications of genetic research.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**											**2**	**2**
CO2	**3**	**3**											**2**	**2**
CO3	**3**	**3**											**2**	**2**
CO4	**3**	**3**											**2**	**2**
CO5	**3**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Bioinformatics: Sequence and Genome Analysis by David W. Mount, Cold Spring Harbor Laboratory Press, Second Edition, 2004, ISBN: 978-0879697129.
2. Bioinformatics: Principles and Applications by Zhumur Ghosh and Bibekanand Mallick, Oxford University Press, First Edition, 2008, ISBN: 978-0195692303.

Reference Books:

1. Essential Bioinformatics by Jin Xiong, Cambridge University Press, First Edition, 2006, ISBN: 978-0521600828.

Activity Based Learning:

1. Phylogenetic Tree Construction
2. Database Exploration and Annotation
3. Gene Expression Analysis

E Resources:

1. https://www.udemy.com/topic/bioinformatics/
2. https://www.coursera.org/courses?query=bioinformatics

3. https://onlinecourses.nptel.ac.in/noc21_bt06/preview

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SUPPLY CHAIN LOGISTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. Learn about Supply Chain, Operations, Channels of Distribution fit in to various types of Business.
2. Understand the management components of supply chain management
3. Develop the various management inventories and network design techniques useful in implementing supply chain management

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.

Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

Introduction to Supply Chain Management, Concept of SCM Building, Components of Supply Chain, a strategic framework to analyze supply chains. Understanding the supply chain, Supply chain performance: Achieving strategic fit and scope, Supply chain Drivers and Obstacles, Value Chain Management.

Unit 2:	**09 Hours**
SUPPLY CHAIN MANAGEMENT: Customer Focus in SCM, Planning demand and supply in a supply chain. Demand forecasting in a supply chain, Aggregate planning in a supply chain, and Planning supply and demand in a supply chain. Managing predictable variability.	
Unit 3:	**09 Hours**
MANAGING INVENTORIES: Planning and managing inventories in a supply chain, Managing Economies of scale in a supply chain: Cycle inventory, Managing uncertainty in a supply chain. Safety inventory, Determining optimal level of product availability.	
Unit 4:	**08 Hours**
NETWORK DESIGN: Transportation, Network design and information technology in a supply chain, Facilities decisions. Network Design in a supply chain, Information Technology in a supply chain	
Unit 5:	**05 Hours**
RECENT TRENDS IN SUPPLY CHAIN: Coordinating a supply chain and the role of E-Business, Coordination in a supply chain E-business and the supply chain, financial evaluation of supply chain decisions-Best Practice in Supply Chain.	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Explain** the fundamental concepts and components of supply chain management (SCM).	L3
2	**Develop** demand and supply planning strategies within a supply chain context.	L3
3	**Evaluate** methods for managing uncertainty and maintaining safety inventory.	L3
4	**Design** effective supply chain networks considering cost and service factors.	L3
5	**Examine** the principles and practices of supply chain coordination and e-business.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**											**2**	**2**

CO2	**3**	**3**											**2**	**2**
CO3	**3**	**3**											**2**	**2**
CO4	**3**	**3**											**2**	**2**
CO5	**3**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. "Supply Chain Management: Strategy, Planning, and Operation" by Sunil Chopra, Peter Meindl, and D.V. Kalra, Edition: 8th Edition (2022), ISBN: 978-0136520986
2. "Supply Chain Management: From Vision to Implementation" by Stanley E. Fawcett, Lisa M. Ellram, and Jeffrey A. Ogden, Edition: 2nd Edition (2019), ISBN: 978-0134133515

Reference Books:

1. "Inventory Management: Principles, Concepts, and Techniques" by John Toomey, Edition: 2nd Edition (2020), ISBN: 978-3319816371
2. "Logistics & Supply Chain Management" by Martin Christopher, Edition: 5th Edition (2016), ISBN: 978-1292083797

Activity Based Learning:

1. Real world problem solving using group discussion.
2. Flip class activity

E Resources:

1. https://archive.nptel.ac.in/courses/110/106/110106045/
2. https://www.coursera.org/specializations/supply-chain-management

SEMESTER – VII	
Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

This Course will enable students to:

1. Understand Fundamental Concepts of Financial Risk.
2. Develop Analytical Skills for Risk Measurement.
3. Integrate Risk Management with Business Practices.
4. Understand Regulatory and Compliance Issues.
5. Promote Ethical Considerations in Risk Management.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

Unit 1:	**08 Hours**

Introduction to Risk- Understanding Risk- Nature of Risk, Source of Risk, Need for risk management, Benefits of Risk Management, Risk Management approaches. Risk Classification-credit risk, market risk, operational risk and other risk

Unit 2:	**09 Hours**
Risk Measurements-Measurement of Risk–credit risk measurement, market risk measurement, interest rate risk measurement, Asset liability management, measurement of operational risk	
Unit 3:	**09 Hours**
Risk Management-Risk Management-Managing credit risk, managing operational risk, managing market risk, insurance	
Unit 4:	**08 Hours**
Risk in Instruments-Tools for risk management–Derivatives, combinations of derivative instruments, Neutral and volatile strategies, credit derivatives, credit ratings, swaps	
Unit 5:	**05 Hours**
Regulation and Other Issues: Other issues in risk management–Regulatory framework, Basel committee, legal issues, accounting issues, tax issues, MIS and reporting, 32 integrated risk management	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Explain the nature and sources of risk, and articulate the need and benefits of risk management.	L3
2	Demonstrate the ability to measure various types of risks, including credit, market, interest rate, and operational risks, and understand asset liability management.	L2
3	Develop strategies to manage different types of risks such as credit risk, operational risk, and market risk, including the use of insurance	L3
4	Evaluate and apply various risk management tools and instruments, including derivatives, credit derivatives, and swaps, and understand their strategies.	L3
5	Analyze the regulatory framework, including Basel committee guidelines, and address legal, accounting, tax issues, and the importance of MIS and integrated risk management	L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**3**											**2**	**2**

CO2	**3**	**3**											**2**	**2**
CO3	**3**	**3**											**2**	**2**
CO4	**3**	**3**											**2**	**2**
CO5	**3**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. "Financial Risk Management: Applications in Market, Credit, Asset, and Liability Management, and Firmwide Risk, Jimmy Skoglund and Wei Chen,Edition: 1st Edition (2015), Publisher: Wiley, ISBN: 978-1119135513
2. Market Risk Analysis, Volume I-IV,Carol Alexander, Edition: 1st Edition (2008), Publisher: Wiley, ISBN: 978-0470997996 (Volume I)

Reference Books:

1. Credit Risk Modeling: Theory and Applications, David Landon, 1st Edition (2004, with latest reprint in 2020), **Publisher**: Princeton University Press, **ISBN**: 978-0691089294
2. Operational Risk Management: Best Practices, J.R. Bell, **Edition**: 1st Edition (2016), **Publisher**: Apress, **ISBN**: 978-1484222607

Activity Based Learning:

1. Simulated Trading and Risk Management.
2. Credit Rating Analysis
3. ERM Framework Development

E Resources:

1. https://archive.nptel.ac.in/courses/110/107/110107128/
2. https://www.coursera.org/courses?query=risk%20management

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

OPTIMIZATION TECHNIQUES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39 Hours
L–T–P–J: .3-0-0-0	

Course Learning Objectives:

1. Introduces various optimization techniques i.e classical, linear programming, simplex algorithm, and constrained and unconstrained optimization techniques.
2. Apply different optimization techniques for solving and optimizing engineering problems in real-world situations.
3. Illustrate population-based optimization techniques.
4. To provide a clear understanding of heuristic optimization techniques and multi-objective optimization techniques.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.

7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**

Introduction And Classical Optimization Techniques: Statement of an Optimization problem, design vector, design constraints, constraint surface, objective function, Classification of Optimization problems Single variable Optimization, multi variable Optimization without constraints, multivariable Optimization with equality constraints, Solution by the method of Lagrange multipliers, multivariable Optimization with inequality constraints, Kuhn– Tucker conditions.

UNIT 2:	**07 Hours**

Linear Programming: Standard form of a linear programming problem, Solution of a system of linear simultaneous equations, pivotal reduction of a general system of equations, simplex algorithm.

UNIT 3:	**08 Hours**

Transportation Problem & Unconstrained Optimization: Finding initial basic feasible solution by north-west corner rule, least cost method, Vogel's approximation method, testing for optimality of balanced transportation problems Constrained Nonlinear Programming: Characteristics of a constrained problem, Classification, Basic approach of Penalty Function method, Basic approaches of Interior and Exterior penalty function 67 methods.

UNIT 4:	**08 Hours**

Genetic algorithms: Genetic Algorithm versus Conventional Optimization Techniques, Genetic representations and selection mechanisms, Genetic operators, different types of crossover and mutation operators. Particle Swarm

Optimization: anatomy of a particle, equations based on velocity and positions, PSO topologies, control parameters, Fly Optimization Technique

UNIT 5:	**08 Hours**

Ant Colony Optimization: Biological ant colony system, Pheromone updating, local-global, Pheromone evaporation. Artificial Bee Colony Algorithms: Task partitioning in honey bees, Balancing foragers and receivers, Artificial bee colony (ABC) algorithms, binary ABC algorithms. Multi-Objective Optimization: Application to multi-modal function optimization. Introduction to multi-objective optimization, Concept of Pareto optimality. Evolutionary Computing: Evolutionary Computing, Simulated Annealing, Random Search, Downhill Simplex Search.

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Make use of the classical optimization techniques, linear programming, and simplex algorithm to real-world problems and applications.	**L3**
2	Analyse unconstrained optimization, constrained non-linear programming, and dynamic programming techniques.	**L3**
3	Distinguish the different types of optimization techniques	**L3**
4	Apply heuristic optimization techniques and multi-objective optimization techniques to complex problems.	**L3**
5	Evaluate the concepts of population-based optimization techniques	**L3**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**2**												**3**	**2**

CO2	**3**	**2**											**3**	**2**
CO3	**3**	**3**											**3**	**2**
CO4	**3**	**3**											**3**	**2**
CO5	**3**	**3**											**3**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Operations Research: An Introduction, H.A.Taha, ISBN-10 :9352865278, Pearson Pvt. Ltd, 2019.
2. Bio-Inspired Artificial Intelligence - Theories, Methods, and Technologies By Dario Floreano and Claudio Mattiussi, ISBN: 9780262547734, MIT Press, 2023.
3. Xin-She Yang, "Recent Advances in Swarm Intelligence and Evolutionary Computation, Springer International Publishing, Switzerland, 2015.
4. Kalyanmoy Deb, Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, ISBN: 978-0-471-87339-6, 2001

Reference Books:

1. Engineering optimization: Theory and practice", S. S.Rao, New Age International (P) Limited, John Wiley & Son, ISBN 978-0-470-18352-6, 2019.
2. Optimization Methods in Operations Research and Systems Analysis, K.V. Mittal and C. Mohan, New Age International (P) Limited ISBN-10: 9388818334, 2020.

Activity Based Learning:

1. Student seminars (on topics of the syllabus and related aspects (individual activity)
2. Study projects (by very small groups of students on selected local real-time problems pertaining to the syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity)).

COMPILER DESIGN AND SYSTEM SOFTWARE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39Hours
L–T–P–J: 3-0-0	

Course Learning Objectives:

1. **Explain** the basic assembler design and functionality to identify the process of translating assembly language into machine code.
2. **Differentiate** between various types of loaders and identify the purpose and functionality of linkers and loaders in program execution.
3. **Outline** the purpose and function of lexical analysis and syntax analysis in the compilation process.
4. **Analyze** the principles and mechanics of different top down and bottom up parsing methods.
5. **Apply** various local optimizations (e.g., constant folding, dead code elimination) and global optimizations (e.g., loop transformations, data flow analysis) and its role in efficient code generation.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the

students' understanding.

Unit 1:	08 Hours

INTRODUCTION TO SYSTEM SOFTWARE:
Machine Architecture of SIC and SIC/XE.
(Text Book-1: Chapter 1: 1.1, 1.2, 1.3)
ASSEMBLERS:
Basic assembler functions: A simple assembler, Assembler algorithm and data structures.
(Text Book-1: Chapter 2: 2.1)
MACHINE DEPENDENT ASSEMBLER FEATURES:
Instruction formats and addressing modes – Program relocation.
(Text Book-1: Chapter 2: 2.2)
MACHINE INDEPENDENT ASSEMBLER FEATURES:
Literals, Symbol-defining statements, Expressions, Program blocks.
(Text Book-1: Chapter 2: 2.3)

Unit 2:	08 Hours

LOADERS:
Basic loader functions: Design of an Absolute Loader, A Simple Bootstrap Loader
(Text Book-1: Chapter 3: 3.1)
MACHINE DEPENDENT LOADER FEATURES:
Relocation, Program Linking, Algorithm and Data Structures for Linking Loader
(Text Book-1: Chapter 3: 3.2)

MACHINE-INDEPENDENT LOADER FEATURES:
Automatic Library Search, Loader Options, Loader design options: Linkage Editors, Dynamic Linking. **(Text Book-1: Chapter 3: 3.3)**

Unit 3:	08 Hours

INTRODUCTION TO COMPILERS:
Language Processors, Structure of compiler, The science of building a compiler, Applications of compiler technology.
(Text Book-2: Chapter 1: 1.1, 1.2, 1.4,1.5)

LEXICAL AND SYNTAX ANALYSIS:
Role of lexical Analyzer, Specification of Tokens, Lexical Analyzer generator Lex.
(Text Book-2: Chapter 3: 3.1)

SYNTAX ANALYSIS I:

Role of Parser, Syntax error handling, Error recovery strategies, Writing a grammar: Lexical vs Syntactic Analysis, Eliminating ambiguity, Left recursion, Left factoring.
(Text Book-2: Chapter 4: 4.1, 4.3)

Unit 4:	**08 Hours**

TOP DOWN PARSING:
Recursive Descent Parsing, First and follow, LL (1)
(Text Book-2: Chapter 4: 4.4)

BOTTOM UP PARSING:
Shift Reduce Parsing, Introduction to LR parsing Simple LR: Why LR Parsers, Items and LR0 Automaton.
(Text Book-2: Chapter 4: 4.5, 4.6)
SYNTAX-DIRECTED TRANSLATION:
Syntax-Directed Definitions: Inherited and Synthesized Attributes.
(Text Book-2: Chapter 5: 5.1)

EVALUATION ORDERS FOR SDDS:
Dependency graphs, Ordering the evaluation of Attributes, S-Attributed Definition, L-Attributed Definition.
(Text Book-2: Chapter 5:5.2)

APPLICATION:
Construction of Syntax Trees.
(Text Book-2: Chapter 5: 5.3)

Unit 5:	**7 Hours**

Three Address Code:
Addresses and Instructions, Quadruples, Triples, indirect triples.
(Text Book-2: Chapter 6: 6.2)

CODE GENERATION:
Issues in the design of code generator
(Text Book-2: Chapter 8: 8.1)

Basic Blocks:
Optimization of Basic Blocks, The Code Generation Algorithm, Peephole optimization.
(Text Book-2: Chapter 8:8.4,8.5 8.6,8.7)

MACHINE INDEPENDENT OPTIMIZATION:
The Principal Sources of Optimization
(Text Book-2: Chapter 9: 9.1)

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	**Identify** the data structures, algorithm, machine dependent Assembler features and build the object code for Simplified Instructional Computer program	L2
2	**Infer** how linker and loader builds an executable program from an object UNIT generated by assembler	L2
3	**Interpret** the major phases of compilation and to apply the knowledge of Lex tool & YACC tool to build the appropriate parsing application	L3
4	**Compare and Contrast** various top down and bottom up parsing techniques to analyze grammatical structures involved in compiler construction and **use** formal attributed grammars for specifying the syntax and semantics of programming languages.	L2
5	**Select** various optimization techniques used for dataflow analysis and build machine code from the source code of a novel language.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2		2									2	1	2
CO2				2								1	1	
CO3	2		2		2							2	2	2
CO4	2	3	2	2								2	3	2
CO5	2	3	1	2								2	2	2
CO6	2		2									2	1	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. Leland L. Beck, "System Software – An Introduction to Systems Programming", 3rd Edition, Pearson Education Asia, 2006.
2. Alfred V Aho, Monica S. Lam, Ravi Sethi and Jeffrey D Ullman, "Compilers – Principles, Techniques and Tools", 2nd Edition, Pearson Education, 2007.

Reference Books:

1. V. Raghavan, Principles of Compiler Design‖, Tata McGraw Hill Education Publishers, 2010.
2. D.M.Dhamdhere, Systems Programming and operating systems, Second Revised edition, Tata McGraw Hill.

Activity Based Learning:

1. Using Software Simulators for Parsing techniques.
2. Use interactive tools like online compiler visualizers such as LLVM IR viewer, Godbolt Compiler Explorer. Perform live demonstrations of code compilation. Encourage students to experiment with their code and observe the generated output at each stage.

E Resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs07/preview
2. https://www.cs.princeton.edu/courses/archive/spring20/cos320
3. https://ocw.mit.edu/courses/6-004-computation-structures-spring-2017/pages/c11/

Following are experiments to be carried out using lex and yacc.
1 a. Program to count the number of characters, words, spaces and lines in a given input file.
b. Program to recognize and count the number of identifiers in a file.
2 a. Program to count the numbers of comment lines in a given C program. Also eliminate them and copy the resulting program into separate file.
b. Program to recognize whether a given sentence is simple or compound.
3 a. Program to count no of: i.+ve and –ve integers ii. +ve and –ve fractions
b. Program to count the number of "scanf" and "printf" statements in a C program. Replace them with "readf" and "writef" statements respectively.
4.Program to evaluate arithmetic expression involving operators +,-,*,/
5. Program to recognize a valid variable which starts with a letter, followed by any number of letters or digits.
6. Program to recognize the strings using the grammar (a^nb^n ;n>=0)
Following are experiments to be carried out using C language.
7. C Program to implement Pass1 of Assembler
8. C Program to implement Absolute Loader
9. C program to find the FIRST in context free grammar.
10.C Program to implement Shift Reduce Parser for the given grammar E →E+E E→E*E E→(E) E → id
11.Write a LR parser program in C. Define the data structure for the parsing table in such a way that it can be initialized easily (manually) for a given grammar. Take a simple grammar, eg., expression grammar, compute the parsing table entries by hand using the steps discussed

in the class, and initialize the table in your program with these values. Try to parse input expressions scanned by a lexical analyzer (which can be easily created using flex). The output of the parser should be SUCCESS or FAILURE depending on the input. In case of FAILURE the parser should indicate the incorrect token in the input.
12. Implement a C program to generate three address code for the given expression z = a + b + c – d
13.Write a C program to perform the recursive descent parsing for the given grammar S->AA A->aB/ϵ B->b
14. Write a program to illustrate the loop based code optimization technique.

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

DATA PRIVACY & CYBER SECURITY [As per Choice Based Credit System (CBCS) scheme] **SEMESTER – VI**	
Course Code	**Credits:**03
Hours /Week: 03 Hours	**Total Hours:**39 Hours
L–T–P–J : 3-0-0-0	

Course Learning Objectives:

1. Foundational understanding of confidentiality, integrity, and authentication techniques.
2. Grasp of state-of-the-art privacy frameworks including differential privacy and federated learning.
3. Familiarity with encryption practices for storage, transmission, and machine-learning models.
4. Awareness of prevalent cyber-threat landscapes and enterprise defence strategies.
5. Knowledge of global compliance mandates and policy implementation processes.

Teaching-Learning Process (General Instructions)

1. These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.
2. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
3. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
4. Show ***Video/animation*** films to explain functioning of various concepts.
5. Encourage ***Collaborative*** (Group Learning) Learning in the class.
6. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
7. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.

8. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
9. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**

FOUNDATIONS OF CRYPTOGRAPHIC CONFIDENTIALITY: Confidential communication- Security goals (CIA triad) ,Shannon's notions of perfect secrecy ,Computational vs. information-theoretic security , Adversary capabilities & threat modelling, Symmetric-key encryption- Feistel vs. SP-networks, Modes of operation (ECB, CBC, CTR, GCM) ,Key sizes & brute-force limits ,Padding schemes & oracle attacks, Public-key encryption- – RSA key generation & math preliminaries ,Diffie–Hellman key exchange flow, Elliptic-curve variants ,Hybrid encryption (KEM + DEM)

UNIT 2:	**08 Hours**

ENSURING INTEGRITY AND AUTHENTICITY: Hashing & MACs- – Collision resistance, pre-image, 2-nd pre-image ,Merkle–Damgård vs. sponge constructions ,HMAC design & security proofs ,Authenticated encryption (AEAD) overview, Digital signatures- – RSA-PSS vs. RSA-PKCS#1 v1.5 ,ECDSA workflow and parameters ,Certificate authorities & PKI basics ,Revocation lists and OCSP, Key-exchange & TLS- – Forward secrecy with Ephemeral DH / ECDHE ,TLS 1.3 handshake message flow ,Record-layer encryption and cipher suites ,Certificate pinning & HSTS

UNIT 3:	**08 Hours**

PRIVACY-ENHANCING TECHNIQUES: Differential privacy- Formal ε, δ definitions ,Laplace, Gaussian, and Exponential mechanisms ,Composition and privacy budget ,Local vs. global DP settings, Federated learning - Parameter server architecture ,Secure aggregation protocols ,Data heterogeneity & personalization ,Privacy attacks (model inversion, membership inference), Encrypted computation- Partially vs. fully homomorphic encryption ,Secure multi-party computation basics ,Trusted execution environments (SGX) ,Performance trade-offs & use cases

UNIT 4:	**08 Hours**
DATA IN TRANSIT AND STORAGE PROTECTION: Data-at-rest encryption- Full-disk vs. file-level encryption ,Key derivation (PBKDF2, Argon2) ,Hardware Security UNITs (HSMs) ,Backup encryption and key escrow, Data-in-motion security- – TLS cipher negotiation & downgrade resistance ,VPN tunnelling protocols (IPsec, WireGuard) ,Perfect forward secrecy in transit ,Certificate lifecycle automation (ACME), Big-data model protection- Model extraction & poisoning threats ,Encryption of model parameters ,Secure model serving pipelines ,Access-control patterns for ML APIs	
UNIT 5:	**07 Hours**
GOVERNANCE, THREATS & COMPLIANCE: Attack model-Insider threat taxonomy ,Advanced persistent threats (APT) ,Side-channel and timing attacks ,Social-engineering vectors, Policy & enterprise protection- Security policy frameworks (ISO 27001) ,Risk assessment matrices ,Incident response planning ,Auditing & continuous monitoring, Legal & compliance-GDPR principles and fines ,HIPAA Privacy & Security Rules ,Data-transfer impact assessments ,Emerging AI-specific regulations	
At the end of the Course the Students will be able to	

Course Outcome	Description	Bloom's Taxonomy Level
1	Explain core cryptographic and privacy mechanisms for securing data.	**L2**
2	Apply differential-privacy and encrypted-computation methods to protect datasets.	**L3**
3	Configure data-at-rest and data-in-motion security using industry protocols.	**L3**
4	Analyse sophisticated attack vectors and recommend mitigations.	**L4**
5	Evaluate organisational compliance with major data-protection regulations.	**L4**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2		2									2	1	2
CO2												1	1	
CO3	2		2									2	2	2
CO4	2	3	2									2	3	2
CO5	2	3	1									2	2	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

2

Text Books:

1) Daniele Venturi et al., *Data Privacy and Security* (Springer, 2013)

2) Julia Lane, Victoria Stodden et al., Privacy, Big Data, and the Public Good: Frameworks for Engagement, Cambridge University Press.

Reference Books:

1) William Stallings, Cryptography and Network Security: Principles and Practice (7th Edition) Pearson publication

2) Benjamin C.M. Fung et al.,Privacy-Preserving Data Publishing: An Overview, Morgan & Claypool Publishers

Activity Based Learning:

1) Group Discussion about GDPR audit case study

2) Debate on the significance of the compliance

E Resources:

1) NIST Privacy Framework: https://www.nist.gov/privacy-framework 2) Statistics for 2)

2) Mozilla Data Privacy Hub: https://foundation.mozilla.org/en/data-privacy/

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SOFT COMPUTING TECHNIQUES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

This Course will enable students to:

1. Learn the various soft computing frameworks
2. Be familiar with various neural network frameworks
3. Learn Genetic programming
4. Be exposed to hybrid systems

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.

7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**

INTRODUCTION:

Scope of soft computing, various components, description of Artificial neural networks, overview fuzzy logic, theory of genetic algorithms, theory of hybrid systems.

UNIT 2:	**08 Hours**

NEURAL NETWORKS:

Fundamentals of neural network, basic models of ANN, learning and activation functions, basic fundamental McCulloch- Pitts neuron model

Learning Models: Supervised learning networks, Adaline, Back propagation, Unsupervised learning network, Korhonen self-organizing feature maps networks

UNIT 3:	**09 Hours**

FUZZY SETS AND RELATIONS:

Uncertainty and Imprecision - Chance vs ambiguity - Classical Sets and Fuzzy Sets - Classical Relations and Fuzzy Relations - Membership functions - Properties of Membership functions - Fuzzification and Defuzzification

UNIT 4:	**06 Hours**

FUZZY LOGIC:

Introduction to fuzzy logic, classical sets and Fuzzy sets, Classical relations and Fuzzy relations, Membership functions, Fuzzy arithmetic and Fuzzy measures, fuzzy decision making.

UNIT 5:	**08 Hours**

HYBRID SOFT COMPUTING TECHNIQUES:

Introduction to hybrid systems – Integration of neural networks, fuzzy logic, and genetic algorithms – Neuro-fuzzy systems – Fuzzy-genetic systems – Neural-genetic systems – Case studies and applications of hybrid approaches in real-world problem solving

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Explain the principal components of soft computing that include ANN and genetic Algorithm	L3
2	Apply a suitable method of soft computing to solve a particular problem	L3
3	Understanding the significance of neural network in soft computing	L3
4	Analyse the Gentic algorithms and fuzzy logic for solving the optimization problems	L3
5	Implementing the soft computing techniques in real life applications	L3, L3

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2									2	3	3
CO2	3	2	3									2	2	3
CO3	2	2	2									2	3	2
CO4	3	1	3									1	2	3

CO5	1	1	1									2	1	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Text Books:

1. S.N Sivanandam and S N Deepa,"Principles of Soft Computing", Wiley India Pvt Ltd, 2011
2. J.S.R. Jang, CT Sun and E Mizutani, "Neuro Fuzzy and Soft Computing", PHI/Pearson Education 2004
3. T.J.Ross, "Fuzzy Logic with Engineering Applications", McGraw Hill, 1995.
4. S. Rajasekaran and G.A.V. Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications", PHI, 2007

Reference Books:

1. George J Klir, Ute St Clair, Bo Yuan,"Fuzzy Set Theory: Foundations and Application" Prentice HAll, 1997
2. David E Goldberg,"Genetic Algorithm in Search Optimization and Machine Learning "Pearson Education India 2013

Activity Based Learning (Suggested Activities in Class)

1. Interactive workshops on Fuzzy Logic
2. Neural networks construction kits
3. Genetic algorithm simulation games

E Resources:

- https://archive.nptel.ac.in/courses/106/105/106105173/
- https://www.udemy.com/topic/fuzzy-logic/
- https://www.coursera.org/courses?query=neural%20networks

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

RESPONSIBLE AI [As per Choice Based Credit System (CBCS) scheme] **SEMESTER – VII**	
Course Code:	**Credits**: 03
Hours /Week: 03 Hours	**Total Hours**: 39Hours
L–T–P–J: 3-0-0-0	

Course Learning Objectives:

1. To introduce the principles and motivations behind developing AI systems responsibly in alignment with societal needs.
2. To present common ethical challenges encountered in AI systems, including algorithmic bias, surveillance, and misinformation.
3. To provide an understanding of the importance of human-centric design and moral reasoning in the development of AI applications.
4. To offer insight into the current legal, policy, and governance frameworks that influence the deployment of AI technologies.
5. To explore the broader implications of AI on employment, environment, social structures, and the future of human-machine interaction.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.

2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT 1:	**08 Hours**
INTRODUCTION TO AI ETHICS AND RESPONSIBILITY: Understanding the societal impact of AI, necessity of ethical considerations in AI development, Key principles: fairness, accountability, transparency, and privacy, Stakeholders in the AI ecosystem, Overview of AI narratives and public perception	
UNIT 2:	**07 Hours**
FAIRNESS AND BIAS IN AI: Types and sources of bias, Discrimination and inequality in AI, Methods to detect and mitigate bias, Fairness metrics, Inclusiveness and accessibility in AI system	
UNIT 3:	**08 Hours**
TRANSPARENCY, EXPLAINABILITY, AND ACCOUNTABILITY: Need for explainability in AI, Black-box vs interpretable models, Explainable AI (XAI) techniques: SHAP, LIME, Accountability in AI systems, Human oversight and decision-making	

UNIT 4:	**08 Hours**
PRIVACY, SECURITY, AND REGULATION: Data privacy challenges in AI, GDPR, HIPAA, SOX, ISO/IEC 27001, PCI DSS and data protection laws, Differential privacy and federated learning, AI misuse and adversarial attacks, National and global policy frameworks	
UNIT 5:	**08 Hours**
GOVERNANCE, SOCIETY, AND THE FUTURE OF AI: AI ethics boards and governance models , Ethical AI frameworks (IEEE, EU, OECD) , Responsible innovation and sustainability , AI and employment, surveillance, misinformation, Designing human-centric and trustworthy AI	

At the end of the Course the Students will be able to

Course Outcome	Description	Bloom's Taxonomy Level
1	Explain the foundational concepts of ethical AI, including fairness, accountability, transparency, and privacy.	**L2**
2	Analyse real-world case studies to identify biases in data and evaluate the societal implications of algorithmic decision-making.	**L4**
3	Apply ethical frameworks and tools to design AI systems that promote human values and reduce harm.	**L3**
4	Examine AI governance models, policies, and regulatory frameworks for responsible deployment of AI systems.	**L4**
5	Develop ethical guidelines and strategies for AI implementation considering social, legal, and environmental implications.	**L3**

Table: Mapping Levels of COs to POs / PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2

CO1	**2**												**1**	**2**
CO2													**1**	
CO3	**2**												**2**	**2**
CO4	**2**	**3**											**3**	**2**
CO5	**2**	**3**											**2**	**2**

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low) **2**

Text Books:

1.Virginia Dignum's Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way (Springer, 2019)

2.Mark Coeckelbergh's AI Ethics MIT Press, Essential Knowledge Series, 2020

Reference Books:

1. Cathy O'Neil – Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy Publisher: Crown Publishing, 2016

2. Shalini Sharma – Artificial Intelligence and Ethics Publisher: Wiley India, 2021

Activity Based Learning:

1.Case Study Analysis: AI Gone Wrong.

2.Role-Playing Debate: AI Ethics Council.

E Resources:

1. Policies and data science of AI: https://oecd.ai/en/

2. AI Ethics guidelines: https://algorithmwatch.org/en/ai-ethics-guidelines-global-inventory

List of Open Electives offered from Data Science

STATISTICAL COMPUTING FOR DATA SCIENCE
[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Subject Code	:	22OE0042	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P	:	3–0-0			

Prerequisites:

Proficiency in a Python and R programming language.

Course Learning Objectives:

This Course will enable students to:

1. **Familiarize** with Python libraries, specifically for data science tasks, using Pycharm as the development environment
2. **Gain** knowledge of parametric tests, such as the Z-test, one-sample T-test, paired T-test, independent sample T-test, ANOV MANOVA, and their significance levels and power values. Also, **learn** about non-parametric tests, including the chi-squ test, Fisher's test, Mann-Whitney U test, Kruskal-Wallis rank test, and Wilcoxon sign rank test.
3. **Study** classification models, including logistic regression, discriminant regression analysis, support vector machines (SVM naive Bayes, random forests, CHAID analysis, decision trees, k-nearest neighbors, and neural networks. **Understand** th principles, strength of associations, maximum likelihood estimation, and the use of confusion matrices.
4. **Explore** unsupervised learning techniques, such as principal component analysis (PCA), reliability tests, KMO tests, eigenval interpretation, and clustering methods like K-means clustering and agglomerative clustering.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adop to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listeni formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evalua generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the studer understanding.	
UNIT – I	**08 Hours**
INTRODUCTION: Overview of the Data science process. Different types of data Data Pre-processing: Data Cleaning- Missing values, Noisy da Data cleaning as a process. Data Reduction: principal component analysis. Data Transformation: Strategies overview. D transformation by normalization. Discretization by binning. (Text Book-1: Chapter1) (Text Book-2: Chapter 3)	
UNIT – II	**09 Hours**
EXPLORATORY DATA ANALYSIS AND HYPOTHESIS TESTING: Exploratory Data Analysis: Central Tendency (Text Book-1: Chapter 3-3.1), Dispersions, Five number Distributions, Cr Tabulations. Data Visualization: Histogram, Box Plot, Correlation Plot, Scatter Plot, Line Chart, Bar Chart, Pie Chart, Bubble Cha Decision Tree, Cluster Charts.(Text Book-1:Chapter 2.3,2.4,2.5) Hypothesis Testing: Confidence Intervals (Text Book-1:Chapter 8), Constructing a hypothesis, Null Hypothesis; Alternati Hypothesis, Type I and Type II errors, Power Value(Text Book-1: Chapter 9)	
UNIT – III	**08 Hours**
PARAMETRIC AND NON-PARAMETRIC TESTS: Parametric test: Z test, One Sample T-TEST, Paired T-TEST, Independent Sample T-TEST, ANOVA, MANOVA, Level significance, Power of a test.(Text Book-1:Chapter 9, Chapter 11) Non parametric test: Chi Square Test, Kruskal-Wallis Rank Test, Wilcoxon sign rank.(Text Book-1: Chapter 12)	
UNIT – IV	**07 Hours**
CLASSIFICATION MODELS: Classification Models: Logistic Regression, Test of Associations, Maximum likelihood estimation, Confusion matrix, Supp Vector Machines (SVM), Naive Bayes, Random Forests: Bagging & Boosting, Decision trees, k-Nearest Neighbors, Neu Network	
UNIT – V	**07 Hours**
UNSUPERVISED LEARNING: Unsupervised Learning: Principal component analysis, Reliability Test, Rotation and Extraction steps, Clustering Methods: Means clustering, Agglomerative Clustering	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		

1	To Summarize the data using visual & amp; summary analytics and comm probability distributions	L3
2	To make inference about a sample & population using hypothesis tests.	L3
3	To fit, interpret, and assess regression models and classification with one or m predictors.	L3
4	To apply statistical analysis using R Programming	L3
5	Understanding and analyzing the Data using R packages	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	2	3										2	
CO2	3	3	3										2	
CO3	3	3	3										2	
CO4	3	3	3										2	
CO5	2	3	3										2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, EMC Education Services, Wiley

2. Tilman M.Davies,“THE BOOK OF R - A FIRST PROGRAMMING AND STATISTICS” Library of Congress Cataloging-in-Publication Data,2016

REFERENCE BOOKS:

1. Data Mining in excel: Lecture Notes and cases by Galit Shmueli, Publisher: Wiley
2. Hastie, Tibshirani, Friedman, “The Elements of Statistical Learning” (2nd ed)., Springer, 2008

E-Resources:

1. https://www.simplilearn.com/pgp-data-science-certification-bootcamp-program

****

DATA EXPLORATION USING R

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VII

Subject Code	:	22OE0043	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P	:	3–0-0			

Prerequisites:

Proficiency in a Python and R programming language.

Course Learning Objectives:

This Course will enable students to:

1. Understand and Apply R Programming Fundamentals
2. Perform Data Analysis and Visualization
3. Conduct Exploratory Data Analysis (EDA)
4. Implement and Evaluate Statistical and Machine Learning Models

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adop to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listeni formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evalua generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the studer understanding.

UNIT – I	**08 Hours**

Fundamentals of R Programming: Introduction to R Studio, R vs Python, Comments in R, Constants, R Data Typ R Operators, R Conditional constructs, Looping constructs, Unconditional constructs, R Functions, Data structures us in R, Packages in R.

UNIT – II	**09 Hours**

Introduction to Data Analysis: Overview of data analysis, working with directory in R, Loading and handling data R, Data Visualization with ggplot2, Data Transformation with dplyr.

UNIT – III	**08 Hours**
Exploratory Data Analysis: Exploring a new dataset, Anomalies in numerical data, visualizing relations betwe variables, Assumptions of Linear Regression, Validating Linear Assumption, Missing Values, Covariation, Patterns a Models, gglot2 Calls.	
UNIT – IV	**07 Hours**
Regression Analysis: Introduction, Types of Regression Analysis Models, Linear Regression, Simple Line Regression, Non-Linear Regression, Regression Analysis with Multiple Variables, Cross Validation, Princi Component Analysis, Factor Analysis.	
UNIT – V	**07 Hours**
Classification: Introduction, Different types of Classification, Logistic Regression, Support Vector Machines, K-Near Neighbours, Naïve Bayes Classifier, Decision Tree Classification, Random Forest Classification, Evaluation.	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Apply basic R programming constructs, including data types, operators, a functions, to develop and debug simple R scripts for data manipulation.	**L3**
2	Utilize ggplot2 and dplyr to create visualizations and perform da transformations, effectively preparing data for analysis.	**L4**
3	Conduct exploratory data analysis to identify patterns, anomalies, a relationships in datasets, using visualization tools like ggplot2.	**L3**
4	Apply various regression models (e.g., linear, non-linear, and multi regression) to analyze data and assess model performance using cro validation.	**L4.L5**

5	Implement and evaluate classification algorithms (such as logistic regressic decision trees, and random forests) to classify data and measure moc accuracy.	**L5**

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	2	3										2	
CO2	3	3	3										2	
CO3	3	3	3										2	
CO4	3	3	3										2	
CO5	2	3	3										2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXTBOOKS

1. Hadley Wickham and Garrett Grolemund, “R for Data Science”, O’reilly, 2017.

REFERENCE BOOKS

1. Dr. Bharati Motwani, “Data Analytics using R”, Wiley, 2019.

****