NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA, G.B. NAGAR
(AN AUTONOMOUS INSTITUTE)

Affiliated to

DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, UTTAR PRADESH, LUCKNOW

Evaluation Scheme & Syllabus

For

Bachelor of Technology

Computer Science and Engineering (Artificial Intelligence)

Second Year

(Effective from the Session: 2026-27)

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(AN AUTONOMOUS INSTITUTE)
Computer Science and Engineering (Artificial Intelligence)
Evaluation Scheme
SEMESTER-III

Sl. No.	Subject Codes	Subject Name	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1 WEEK'S COMPULSORY INDUCTION PROGRAM														
1	CCSCC0301	Employability Skill Development – I	Mandatory	2	0	0	60	40	100				100	2
2	CAS0301	Statistics and Probability	Mandatory	3	1	0	30	20	50		100		150	4
3	CCSE0303A	Operating Systems	Mandatory	2	0	0	30	20	50		50		100	2
4	CCSE0301	Data Structures and Algorithm-II	Mandatory	3	0	0	30	20	50		100		150	3
5	CCSML0301	Machine Learning	Mandatory	2	0	0	30	20	50		50		100	2
6	CCSAI0302	High Performance Computing	Mandatory	3	0	0	30	20	50		100		150	3
7	CCSE0353A	Operating Systems Lab	Mandatory	0	0	4				50		50	100	2
8	CCSE0351	Data Structures and Algorithm-II Lab	Mandatory	0	0	4				50		50	100	2
9	CCSML0351	Machine Learning Lab	Mandatory	0	0	2				25		25	50	1
10	CCSE0355	Object Oriented Techniques using Java	Mandatory	0	0	6				50		100	150	3
11	CCSE0359	Internship-I	Mandatory	0	0	2				50			50	1
12	CNC0301/ CNC0302	Artificial Intelligence and Cyber Ethics/ Environmental Science	Compulsory Audit	2	0	0	30	20	50				50	NA
13		MOOCs (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL		**17**	**1**	**18**	**210**	**140**	**350**	**225**	**400**	**225**	**1200**	**25**

List of MOOCs Based Recommended Courses for Second year B. Tech Students

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1.	CMC0027	Object Oriented Programming Using Python	Infosys Wingspan (Infosys Springboard)	46h 13m	3.5
2.	CMC0028	Probability and Statistics using Python	Infosys Wingspan (Infosys Springboard)	16h	1
3	CMC0029	Cyber Security Essential	NASSCOM	70h	4

PLEASE NOTE: -

- **A 3–4-week Internship shall be conducted during summer break after semester-II and will be assessed during semester-III.**
- **Compulsory Audit (CA) Courses (Non-Credit - CNC0301/CNC0302)**
  - All Compulsory Audit Courses (a qualifying exam) do not require any credit.
  - The total and obtained marks are not added to the grand total.

Abbreviation Used:

L: Lecture, T: Tutorial, P: Practical, CT: Class Test, TA: Teacher Assessment, PS: Practical Sessional, TE: Theory End Semester Exam.
CE: Core Elective, OE: Open Elective, DE: Departmental Elective, PE: Practical End Semester Exam, CA: Compulsory Audit,
MOOCs: Massive Open Online Courses.

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(AN AUTONOMOUS INSTITUTE)
Computer Science and Engineering (Artificial Intelligence)
Evaluation Scheme
SEMESTER-IV

Sl. No.	Subject Codes	Subject Name	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1 WEEK'S COMPULSORY INDUCTION PROGRAM														
1	CASCC0401	Employability Skill Development – II	Mandatory	2	0	0	60	40	100				100	2
2	CCSE0401	Database Management Systems	Mandatory	2	0	0	30	20	50		50		100	2
3	CCSML0402	Deep Learning	Mandatory	3	0	0	30	20	50		100		150	3
4	CCSE0403	Theory of Computation	Mandatory	3	0	0	30	20	50		100		150	3
5		Department Elective -I	Department Elective	3	0	0	30	20	50		100		150	3
6	CASL0401	Technical Communication	Mandatory	2	0	0	30	20	50		50		100	2
7	CCSE0451	Database Management Systems Lab	Mandatory	0	0	4				50		50	100	2
8	CCSML0452	Deep Learning Lab	Mandatory	0	0	2				25		25	50	1
9	CCSE0455	Web Technology	Mandatory	0	0	6				50		100	150	3
10	CCSE00459	Mini Project	Mandatory	0	0	2				50			50	1
11	CCSCC0452	Problem Solving Approaches	Mandatory	0	0	2				50			50	1
12	CNC0402/ CNC0401	Environmental Science / Artificial Intelligence and Cyber Ethics	Compulsory Audit	2	0	0			50				50	NA
13		MOOCs (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL		**17**	**0**	**16**	**210**	**140**	**350**	**225**	**400**	**175**	**1150**	**23**

List of MOOCs Based Recommended Courses for first year B. Tech Students

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1.	CMC0049	R Programming for Machine Learning	IIHT	21h 36m	1.5
2.	CMC0050	Deep Learning for Devlopers	Infosys Wingspan (Infosys Springboard)	34h 51m	2.5
3	CMC0051	An Introduction to world of AI with Machine Learning	NASSCOM	30h	2

PLEASE NOTE: -

- **Compulsory Audit (CA) Courses (Non-Credit - CNC0401/CNC0402)**
  - All Compulsory Audit Courses (a qualifying exam) do not require any credit.
  - The total and obtained marks are not added to the grand total.

Abbreviation Used:

L: Lecture, T: Tutorial, P: Practical, CT: Class Test, TA: Teacher Assessment, PS: Practical Sessional, TE: Theory End Semester Exam.
CE: Core Elective, OE: Open Elective, DE: Departmental Elective, PE: Practical End Semester Exam, CA: Compulsory Audit,
MOOCs: Massive Open Online Courses.

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

List of Departmental Electives

Sl. No.	Subject Codes	Subject Name	Types of Subjects	Bucket Name	Applicable Branch	Semester
1	CCSCY0411	Fundamental of Cyber Security	Departmental Elective - I	Cloud Computing Bucket	AI/ AIML/ DS/ M&C/ CYS	IV
2	CCSE0411	Python web development with Django	Departmental Elective – I	IoT Smart Systems Bucket	AI/ AIML/ DS/ M&C/ CYS	IV
3	CCSIOT0411	Sensors and Instrumentation	Departmental Elective – I	Cyber Security Bucket	AI/ AIML/ DS/ M&C/ CYS	IV
4	CCSAI0412	CRM Fundamentals	Departmental Elective - I	Full Stack Bucket	AI/ AIML/ DS/ M&C/ CYS	IV

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

AICTE Guidelines in Model Curriculum:

A student will be eligible to get Under Graduate degree with Honours only, if he/she completes the additional MOOCs courses such as Infosys Springboard/NASSCOM/CISCO (as applicable) certifications, or any other online courses recommended by the Institute (Equivalent to 20 credits). During Complete B.Tech. Program Guidelines for credit calculations are as follows.

1. For 6 to 12 Hours =0.5 Credit
2. For 13 to18 =1 Credit
3. For 19 to 24 =1.5 Credit
4. For 25 to 30 =2 Credit
5. For 31 to 35 =2.5 Credit
6. For 36 to 41 =3 Credit
7. For 42 to 47 =3.5 Credit
8. For 48 and above =4 Credit

For registration to MOOCs Courses, the students shall follow Infosys Springboard/NASSCOM/CISCO (as applicable) registration details as per the assigned login and password by the Institute these courses may be cleared during the B. Tech degree program (as per the list provided). After successful completion of these MOOCs courses, the students shall provide their successful completion status/certificates to the Controller of Examination (COE) of the Institute through their coordinators/Mentors only.

The students shall be awarded Honours Degree as per following criterion.

i. If he / she secures 7.50 as above CGPA.
ii. Passed each subject of that degree program in the single attempt without any grace.
iii. Successful completion of MOOCs based 20 credits

Course Code: CCSCC0301	**Course Name:** Employability Skill Development - I	**L**	**T**	**P**	**C**
Course Offered in: B.Tech III Semester		**0**	**0**	**4**	**2**

Pre-requisite: Students should have successfully completed communication skills courses in the first year.

Course Objectives:

Section A: Develop understanding of computer systems, mathematical foundations, and programming concepts while applying structured problem-solving techniques to design logical solutions effectively. Enhance numerical ability and logical reasoning skills through speed mathematics, number systems, and analytical problem-solving techniques for competitive aptitude development. To develop the ability to communicate effectively and empathetically in workplace situations while strengthening active listening, adaptability, and cross-cultural awareness for navigating complex professional environments.

Course Outcome		**Bloom's Knowledge Level (KL)**
CO1	Apply fundamental concepts of computer systems, mathematical foundations, and programming constructs to analyze and solve basic computational problems.	**K3**
CO2	Design and develop structured solutions for problem statements by applying SDLC principles, logical thinking, and flowchart/pseudocode techniques.	**K6**
CO3	Apply number theory and logical reasoning concepts to solve quantitative and analytical problems efficiently.	**K3**
CO4	Apply spoken English skills to participate effectively in standard workplace routines, including daily stand-ups, reporting, and collaborative problem-solving.	**K3**
CO5	Evaluate and adapt communication styles to suit dynamic workplace communication situations and deliver context-sensitive, and logically structured verbal responses under time constraints.	**K6**

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**3**	**2**	1	-	1	-	-	-	-	-	-	**3**	**2**	1
CO2	**2**	**3**	**3**	1	1	-	-	-	1	1	-	**3**	**3**	1
CO3	**2**	**3**	1	-	-	-	-	-	-	-	-	2	**3**	-
CO4	-	1	-	-	-	-	-	**2**	**3**	**3**	1	-	-	**3**
CO5	-	2	1	-	-	-	-	**2**	**3**	**3**	**2**	-	1	**3**

Course Contents / Syllabus

UNIT-1	**Foundations of Computer Systems and Mathematical Concepts**	**8 hours**

Computer System Fundamentals: Introduction to Assembler, Compiler, Interpreter, Role of Loader and Linker in program execution.

Mathematical Foundations for Computing: Sets, Relations, and Functions: definitions and applications, Principle of Mathematical Induction and its use in proofs.

Basic Concepts of Programming Language: Memory model of C/C++ Program, Activation Record of a function, L-Value/ R-Value concept. Operators, Precedence, Order of Evaluation and Associativity. Pointers and Dynamics Memory Concept, Functions (argument passing, concept of scope), Understanding Recursion, Bitwise Operations.

UNIT 2	**Design and develop structured solutions for problem statements**	**8 hours**

Introduction to Software Development Life Cycle, Step-by-step solution to simple problems, Developing logic/flowchart/pseudocode. Stepwise refinement and Procedural Abstraction. **Case Study*:** Understand the problem statement,

identify a feasible solution, and develop the logic along with a flowchart for the proposed solution.

UNIT 3:	**Speed Math and Number System**	**16 hours**

Classification of number, Divisibility Rule, Factorization, HCF & LCM, It's Application, Unit digit (Cyclicity), Last two digit, Remainder theorem, Factorial and Number of zeroes, Highest power.

Analytical and Logical Reasoning: Direction and Sense, Blood Relation, Number Series and Letter Series, Coding Decoding

UNIT-4	**Workplace Interaction Essentials**	**12 hours**

- The Professional Shift: Navigating the transition from academic to professional tone. Using appropriate register when addressing peers vs. supervisors.
- Active Listening & The Feedback Loop: Moving beyond passive hearing. Practicing the SBI (Situation-Behaviour-Impact) model for peer feedback and clarifying multi-step technical instructions.
- Structured Micro-Speaking: The art of the concise update. Delivering 1-2 minute "Stand-up" reports (What I did, What I will do, Blockers) and explaining basic technical processes to laymen.
- Guided Interpersonal Simulations: Role-playing everyday scenarios: politely interrupting a meeting, seeking assistance on a delayed task, and confirming understanding of complex briefs using paraphrasing techniques.
- Digital & Hybrid Nuances: Verbal signalling in virtual meetings, managing "overtalk," and translating concise written chat messages into spoken professional updates.

UNIT-5	**Workplace Communication in Action**	**12 hours**

- Dynamic Problem-Based Communication: Simulating the "War Room." Explaining sudden technical failures, proposing immediate solutions, and negotiating revised deadlines with stakeholders.
- Listening Under Pressure & Global Adaptability: Interpreting fast, accented, or incomplete instructions (incorporating CEFR-level listening benchmarks). Decoding tone, urgency, and indirect cultural communication styles.
- Collaborative Conflict Resolution: Managing disagreements in team settings using "I-statements" and the "Yes, and…" technique to build consensus without escalating tension.
- The Persuasive Professional: Delivering evidence-based arguments using the PREP (Point, Reason, Example, Point) framework to justify a technical choice or resource allocation.
- Real-Time "Crisis" Simulation (Capstone): A comprehensive role-play where student groups are given a sudden project change or client issue. They must conduct an emergency meeting, delegate tasks verbally, and present a united response to the "client."

Total: 56 Hours

**A list of problem statements will be provided by the mentor; you may choose anyone.*

Note: *Students may choose an alternative problem statement with prior approval from the mentor.*

Reference Books:

S. No.	Book Title	Author
1	A Project-Based Introduction to Programming	Access Point Publishing
2	Programming: Principles and Practice Using C++	Bjarne Stroustrup
3	Effective Modern C++	Scott Meyers
4	Quantitative Aptitude	R S Aggarwal
5	Verbal & Non-Verbal Reasoning	R S Aggarwal
6	*Crucial Conversations Tools*	Kerry Patterson et al.
7	*HBR Guide to Persuasive Presentations*	Nancy Duarte

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	https://swayam-plus.swayam2.ac.in/courses/course-details?id=F_WADHW_042	**105 hours**
Video Lectures	Employability Skills Training Session - YouTube	**1 hour**
Web References	https://dde.manuu.edu.in/sites/default/files/DDE/DDE-Assignments/Diploma-Sem1-2024/Employability-Skills-1.pdf	

Course Code: CAS0301	**Course Name:** Statistics and Probability	**L**	**T**	**P**	**C**
Course Offered in: B. Tech. Second Year: Semester-III/IV AI/AIML/AI (TWIN)/AIML(TWIN)/CYS/DS/CS/CSE/CSE(TWIN)/CSE R/IT/ IT- Twin/ M. Tech. (Int.)/MCT		**3**	**1**	**0**	**4**

Pre-requisite: B. Tech. First Year Syllabus

Course Objectives:

1. The objective of this course is to familiarize the students with concepts of Probability and statistical techniques.
2. It aims to equip the students with adequate Knowledge of statistics that will enable them in formulating Problems and solving problems analytically

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply the concept of moments, skewness and kurtosis in relevant field.	K3
CO2	Apply the concept of correlation, regression and curve fitting with real world problems.	K3
CO3	Apply the concept of probability and random variable.	K3
CO4	Apply the concept of Mathematical **Expectations and Probability Distribution in real life problems.**	K3
CO5	Apply the concept of hypothesis testing and statistical quality control to create control charts.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	3	1	1	-	1	2	2	2	3	1	1
CO2	3	2	2	3	1	1	-	1	2	2	2	3	2	2
CO3	3	2	1	2	-	-	-	-	1	2	2	2	1	1
CO4	3	2	2	3	1	1	-	1	2	2	2	3	2	2
CO5	3	2	2	3	1	1	-	1	2	2	2	3	2	2

Course Contents / Syllabus

Module 1	Statistical Techniques-I	**9 hours**

Introduction: Measures of central tendency: Mean, Median, Mode, Standard deviation, Quartile deviation, Moments, Skewness, Kurtosis.

Module 2	Statistical Techniques-II	**9 hours**

Curve Fitting: Straight line, Second-degree parabola, Exponential curve by method of least squares, Correlation and Rank correlation, Linear regression, nonlinear regression and multiple linear regression.

Module 3	Probability and Random Variable	**9 hours**

Random Variable: Definition of a Random Variable, Discrete Random Variable, Continuous Random Variable, Probability mass function, Probability Density Function, Distribution functions.
Bivariate Random Variables: Joint probability mass function, joint density and distribution Function, Properties of Joint Distribution function, Marginal Probability mass function, Marginal density Functions, Conditional

Distribution for Bivariate Random Variables, Statistical Independence, Central Limit Theorem (Proof not expected)		
Module 4	Expectation and Probability Distributions	**9 hours**
Expectations of single Random Variable, Mean, Variance, Moment Generating Function, Binomial, Poisson, Normal, Exponential distribution.		
Module 5	Hypothesis Testing and Control Charts	**9 hours**
Testing a Hypothesis, Null hypothesis, Alternative hypothesis, Level of significance, Confidence limits, Test of significance of means by Z-test and t-test, Chi-square test, F-test, One way ANOVA. Statistical Quality Control (SQC), Control Charts, Control Charts for variables (Mean and Range Charts), Control Charts (p, np and C charts)		
	Total Lecture Hours	**45hours**

Textbook:

1. Textbook of Engineering Mathematics- IV Bali, N.P.
2. Advanced engineering mathematics Jain, R.K.
3. Higher engineering mathematics Grewal, B.S.
4. Statistical methods Gupta, S.P.
5. Advanced engineering mathematics ZILL, DENNIS G.

Reference Books:

1. Introduction to Probability Models Ross, Sheldon M
2. Probability, Random Variables and Stochastic Processes Papoulis, Athanasios
3. Advanced engineering mathematics Kreyszig, E

Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://www.futureskillsprime.in/course/probability-and-statistics/	30 hrs
Video Lectures	https://onlinecourses.nptel.ac.in/e-learning/preview/noc21_ma74 https://onlinecourses.nptel.ac.in/e-learning/preview/noc23_ma77 https://onlinecourses.nptel.ac.in/e-learning/preview/noc23_mg10	13 hrs
Web Reference	https://www.geeksforgeeks.org/maths/probability-and-statistics/	3 hrs
	https://www.geeksforgeeks.org/machine-learning/statistics-for-machine-learning/	2 hrs

Course Code: CCSE0303A	**Course Name: Operating Systems**	**L**	**T**	**P**	**C**
Course Offered in: CSE/CSE-R/IT/CS/CSE(AI)/CSE(AIML)/ IOT/DS/CYS/M.Tech. Int.		**2**	**0**	**0**	**2**

Pre-requisite: Basic knowledge of computer fundamentals, C programming, Data structures and Computer organization.

Course Objectives: The objective of the course is to provide a foundational understanding of operating system concepts, including system architecture, process and thread management, concurrency, deadlock, resource management, memory and file systems, Linux shell scripting, and an introduction to virtualization and distributed systems.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand operating system architecture and types, and use the Linux CLI for basic Operations.	K2
CO2	Implement the CPU scheduling algorithms including uses of multithreading models.	K3
CO3	Implement concurrency control, process synchronization techniques, and deadlock handling techniques.	K3
CO4	Implement memory management strategies to optimize system performance and analyze file systems and configure virtual machines in modern operating systems.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO 3
CO1	**3**	**2**	1	1	**3**	-	-	-	-	-	-	**3**	**2**	1
CO2	**3**	**3**	**2**	**2**	**2**	-	-	-	-	-	-	**3**	**3**	1
CO3	**3**	**3**	**3**	**3**	**2**	-	-	-	1	-	-	**3**	**3**	**2**
CO4	**3**	**3**	**2**	**2**	**3**	-	-	-	-	-	1	**3**	**3**	**2**

Course Contents / Syllabus

Unit 1	**Fundamentals & Shell scripting**	**8 hours**

Fundamentals of Operating Systems: Overview of Operating Systems, Generations of OS, Operating system architecture, Interrupt handling, System call and kernel, Types of Operating System: Batch OS, Multi-programming OS, Multitasking OS, Multiprocessor OS, Real time OS. **Shell Scripting in Linux:** Introduction to Linux Operating System & Architecture, Basic Command Line Interface (CLI) Operations in Linux, Shell Scripting Basics: Variables, Control Structures, Functions.
Case study: Automating system administration tasks using shell scripts in Linux.

Unit 2	**Process & Thread Management**	**8 hours**

Process Management: Process, Transition Diagram, Process Control Block (PCB), Types of Schedulers: Long Term, Mid Term, Short Term Scheduler,
CPU Scheduling: Pre-emptive and non-pre-emptive Algorithm (FCFS, SJF, SRTF, Non-Pre-emptive Priority, Pre-emptive Priority, Round Robin, Multilevel Queue Scheduling and Multilevel Feedback Queue Scheduling). **Thread:** Processes Vs Threads, Thread states, Benefits of threads, Types of threads, Multithreading Models, Concept of Hyper-Threading
Case study: Analyse and implement CPU Scheduling in Real-Time Embedded Systems and RTOS.

Unit 3	**Concurrency and Deadlock Management**	**8 hours**

Concurrency: Introduction of Concurrency, Types of Process, Race Condition, Critical Section, Inter Process Communication, Producer consumer problem.
Process Synchronization: Lock variable, Peterson's Solution, Strict alternation, Lamport Bakery Solution, Test and set lock, Semaphore- counting, binary and monitor. **Classical Problem of Synchronization:** Bound Buffer, Dinning Philosopher, Reader writer, Sleeping barber. **Deadlock:** Deadlock, Deadlock characterization, Deadlock Prevention, Deadlock Avoidance: Bankers Algorithms, Deadlock Detection, Recovery from Deadlock.
Case study: Deadlock avoidance in database transaction management systems like Oracle or MySQL.

Unit 4	**Memory Management**	**8 hours**

Memory Management: Memory Management function, Loading and linking Address Binding, Memory management techniques, Contiguous technique- Fixed Partitions, variable partitions, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Non-contiguous, Paging, Segmentation, Segmented paging, **Virtual Memory:** Virtual Memory Concepts, Demand Paging, Performance of Demand Paging, Page Replacement Algorithms: FIFO, LRU, Optimal and LFU, Belady's Anomaly, Thrashing. **File and Disk Management:** File Access Mechanism, File Allocation Method, Disk Scheduling Algorithms. **Modern Operating System:** Virtual machines – hypervisor, Introduction to GPU. Introduction of CUDA Programming. **Case study:** Virtual memory management in modern OS like Windows 10 and how paging impacts performance.

Total Lecture Hours	**32 hours**

Textbook:

S.No	**Book Title**	**Author**
1.	" Operating System Concepts Essentials" , Willey Publication,10th Edition,2018.	Abraham Silberschatz, Peter Baer Galvin and Greg Gagne
2.	"A practical guide to Linux: Commands, Editors and Shell Programming", CreateSpace Independent Publishing Platform, 4th Edition,2017.	Marks G. Sobell
3.	"LINUX for beginners", 1stEdtion,2014.	Jason Cannon

Reference Books:

S.No	**Book Title**	**Author**
1.	"Operating Systems: Internals and Design Principles", Pearson Education , 9th Edition, 2019.	William Stallings
2.	"Operating System: A Design-oriented Approach" , McGraw Hill Education,2017.	Charles Patrick Crowley
3.	"Learning Linux Shell Scripting", Packt Publishing ,2nd Edition 2018.	Ganesh Naik

Self-Study Contents Links:		**Duration (in hours)**
MOOCs	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013843258047987712344 43_shared/overview	**29 hours**
Video Lectures	https://www.youtube.com/watch?v=dTl9QkH4j8o&list=PL6CdojO56mZ3SeRfpzMBMObSnTziA0gfE https://www.youtube.com/watch?v=Q35sMYHaIvQ https://www.youtube.com/watch?v=rnGVincwk30 https://www.youtube.com/watch?v=akU1Ji8Vzdk&list=PLbMVogVj5nJRa3VKt_eyZdJ_DitCz1cvQ https://www.youtube.com/watch?v=pPAKs7tT8s https://www.youtube.com/watch?v=hBPYP0ZEv https://www.youtube.com/watch?v=UeVQxzlgd1 https://www.youtube.com/watch?v=9cuwgiHQyoI https://www.youtube.com/watch?v=-NONm-Jq34Y&t=25s https://www.youtube.com/watch?v=5uyG147rr0 https://www.youtube.com/watch?v=utYC2tKDdoQ https://www.youtube.com/watch?v=Cuo8GTklgoE https://www.youtube.com/watch?v=YHQxp-XduS0 https://www.youtube.com/watch?v=yB8sm-pwvZo https://www.youtube.com/watch?v=Y14b7_T3AEw https://www.youtube.com/watch?v=bK-VhQA512c https://www.youtube.com/watch?v=o2_iCzS9-ZQ https://www.youtube.com/watch?v=SnKgEuUfV4k https://www.youtube.com/watch?v=86FAWCzIe_4	**30 hours**
Web References	https://www.geeksforgeeks.org/operating-systems/operating-systems/ https://studyhubzone.com/operating-systems-notes/	**4 hours**

Course Code: CCSE0301	**Course Name: DATA STRUCTURES AND ALGORITHMS-II**	**L**	**T**	**P**	**C**
Course Offered in: CSE/CSE-R/CS/IT/CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/M.tech.Int		**3**	**0**	**0**	**3**

Pre-requisite: C, Python

Course Objectives: The objective of the course is to learn the basic concepts of algorithm analysis, along with the implementation of non-linear data structures.

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

S.No	**Course Outcome**	**Bloom's Level**
CO 1	Apply tree structures to solve specific problems using various tree operations and algorithms.	K3
CO 2	Analyse the graph data structure and evaluate the efficiency of its operations for problem solving.	K4
CO 3	Evaluate dynamic programming solutions for efficient problem-solving across diverse contexts.	K4
CO 4	Apply efficient backtracking and branch –and –bound techniques across diverse problem-solving scenarios.	K3
CO 5	Understand principles of advanced data structures, including their implementation and applications.	K2

CO-PO Mapping (Scale: 1: Low, 2: Medium, 3: High)

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO 3
CO1	3	3	2	2	2	-	-	-	-	-	-	3	3	2
CO2	3	3	3	3	2	-	-	-	-	-	-	3	3	2
CO3	3	3	3	2	2	-	-	-	-	-	1	3	3	2
CO4	3	3	3	3	2	-	-	-	1	-	1	3	3	3
CO5	3	3	2	2	3	-	-	-	-	-	-	3	3	3

Course Contents / Syllabus

Unit 1	**Design and Analysis of Algorithms: Tree**	**8 hours**

Trees: Terminology used with Trees, Binary Tree, Memory representation of Tree, Traversal Algorithms: In-order, Pre-order, and post-order. Constructing Binary Tree from given Tree Traversal, Operation of Insertion, Deletion, Searching & Modification of data in Binary Search tree, Binary Heaps, Threaded Binary trees, Traversing Threaded Binary trees, AVL Tree. **Application of Trees:** Priority Queue, Heap Sort

Unit 2	**Design and Analysis of Algorithms: Graphs**	**8 hours**

Graphs: Terminology used with Graph, Data Structure for Graph Representations: Adjacency matrices, Adjacency List. **Graph Traversal:** Depth First Search and Breadth First Search. Connected Component, Spanning Trees. **Algorithms on Graphs:** Minimum Cost Spanning. **Trees:** Prim' s and Kruskal's algorithm , Transitive Closure and Shortest Path algorithms: Dijkstra Algorithm, Bellman Ford Algorithm, Floyd Warshall's Algorithm.

Unit 3	**Dynamic Programming**	**8 hours**

Dynamic Programming: Dynamic Programming concepts 0/1 Knapsack, Longest Common Sub Sequence, Matrix Chain Multiplication, Resource Allocation Problem.

Unit 4	**Backtracking, Branch and Bound**	**8 hours**

Backtracking: Backtracking, Branch, and Bound with Examples Such as Travelling Salesman Problem, Graph Colouring, n-Queen Problem, Hamiltonian Cycles,

and Sum of Subsets.

Unit 5	Advanced- Data Structures	8 hours

Red-Black Trees, B – Trees, B+ Trees, Binomial Heaps, Fibonacci Heaps

Total Lecture Hours: 40 hours

Textbook:

S.No.	Book Details	Author
1	"Data Structures and Algorithms in Python: An Indian Adaptation", 1st Edition, 2021	Michael T. Goodrich, Roberto Tamassia
2	"Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd, 2nd Edition, 2017.	Lipschutz
3	"Introduction to Algorithms", Printice Hall of India, 4th Edition, 2022	Thomas H. Coreman, Charles E. Leiserson and Ronald L. Rivest,

Reference Books:

S.No.	Book Details	Author
1	"Data Structure Using C", Oxford University Press, 2nd Edition, 2014.	Reema Thareja
2	"Data Structure Using C", Pearson Education India, 2nd Edition,2011.	AK Sharma
3	"C and Data structure", Wiley Dreamtech Publication, 1st Edition, 2004.	P. S. Deshpandey

Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01384203240484864010470_shared/overview	18 hours
	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01309443991409459225 84_shared/overview	3 hours
Video Links	https://www.youtube.com/watch?v=zWg7U0OEAoE&list=PLBF3763AF2E1C572F https://www.youtube.com/watch?v=LQx9E2--p5c&pp=ygUMYXJyYXlzIG5wdGVs https://www.youtube.com/watch?v=K7VIKlUdo20&pp=ygUPbGluayBsaXN0IG5wdGVs https://www.youtube.com/watch?v=g1USSZVWDsY&list=PLB3CD0BBB95C1BF09&index=2&pp=iAQB https://www.youtube.com/watch?v=THMyk2_p530&pp=ygUccXVldWUgZGF0YSBzdHJ1Y3R1cmUgICBucHRlbA%3D%3D https://www.youtube.com/watch?v=_VV9v41FIq0&pp=ygUZZGl2aWRlIGFuZCBjb25xdWVyICBucHRlbA%3D%3D https://www.youtube.com/watch?v=ARvQcqJ_-NY&list=PLfFeAJ-vQopt_S5XlayyvDFL_mi2pGJE3	40 Hours
Web References	https://www.msajce-edu.in/academics/it/LectureNote/CD3291-LN.pdf	-

Course Code: CCSML0301	Course Name: Machine Learning	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)		2	0	0	2

Pre-requisite: Python

Course Objectives:

This course aims to provide students with a strong foundation in the fundamental concepts and techniques of Machine Learning. It enables learners to understand and apply data preprocessing and feature engineering methods for preparing datasets for analysis and model development. The course focuses on the implementation of regression and classification algorithms for predictive analytics and decision-making tasks. Students will also explore clustering techniques, ensemble learning methods, and other advanced machine learning approaches for discovering patterns and improving model performance.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Explain fundamental concepts of Machine Learning, types of learning, and data representation techniques	K2
CO2	Apply data preprocessing, feature engineering, and model evaluation techniques on datasets	K3
CO3	Analyze regression and classification models using appropriate algorithms	K4
CO4	Analyze clustering, probabilistic models, ensemble methods, and anomaly detection techniques for real-world problems	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	3	-	-	-	-	-	-	3	3	2
CO2	3	3	3	2	3	-	-	-	-	-	1	3	3	3
CO3	3	3	3	3	3	-	-	-	1	1	-	3	3	3
CO4	3	3	3	3	3	-	-	-	1	1	1	3	3	3

Course Contents / Syllabus

UNIT 1	Introduction, Data Handling & Feature Engineering	07 hours

Introduction to Machine Learning: Motivation, definition, applications; **Types of ML:** Supervised, Unsupervised, Reinforcement Learning; **Datasets and Features:** Types of features, **Data Preprocessing:** Handling missing data, Categorical feature encoding, Feature scaling, Outlier detection and removal; Feature Selection techniques, **Dimensionality Reduction**: Curse of Dimensionality, Principal Component Analysis (PCA) – intuition & working

UNIT 2	Regression, Optimization & Regularization	07 hours

Regression Problem formulation, Linear Regression: Simple, Multiple, Polynomial, **Error Metrics:** MSE, RMSE, MAE, **Cost Function:** Intuition & visualization, **Gradient Descent:** Algorithm, learning rate, Application to linear regression, **Model Evaluation:** R^2 and Adjusted R^2, **Regularization:** Bias–Variance tradeoff, Overfitting & Underfitting, L1 (Lasso) & L2 (Ridge), Regularized cost function.

UNIT 3	Classification Algorithms & Performance Evaluation	09 hours

Classification Problem: Binary & Multiclass, Decision Boundary concept, **Logistic Regression:** Model, cost function, Regularized logistic regression, One-vs-All strategy, **Performance Metrics:** Accuracy, Precision, Recall, F1-score, Confusion Matrix, ROC-AUC Curve, **Classification Algorithms:** Decision Trees (ID3, C4.5, CART), Support Vector Machines (SVM),k-Nearest Neighbors (k-NN), **Probabilistic Models:** Bayesian Learning, Naïve Bayes Classifier, Bayesian Belief Networks.

UNIT 4	Unsupervised & Ensemble Learning	07 hours

Clustering Algorithms: Clustering problem & distance measures, K-Means (algorithm, objective, initialization, K selection), K-Medoids, K-Modes, Hierarchical Clustering (AGNES, DIANA, linkage methods), DBSCAN (density-based clustering), **Ensemble Learning:** Bagging vs Boosting, Random Forest, AdaBoost, Gradient Boosting, XGBoost (overview), **Anomaly Detection:** Concept & applications, Gaussian Distribution & GMM, Anomaly detection algorithm		
		Total Lecture Hours | **30 hours**
Textbook:		
S.No	**Book Title**	**Authors:**
1.	Machine Learning, McGraw-Hill Education,1997	Mitchell, T. M.
2.	Introduction to Machine Learning, 3rd Edition, MIT Press, 2014	Alpaydin, E.
Reference Books:		
S.no	**Book Title**	**Authors:**
1.	Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow, 2nd Edition, O'Reilly Media, 2019	Géron, A
2.	Python Machine Learning, 3rd Edition, Packt Publishing, 2019	Raschka, S., and Mirjalili, V
Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://www.coursera.org/learn/machine-learning	60 Hrs
Video Lectures	https://www.youtube.com/watch?v=_WcSO1BE7To	10 Hrs 51M
Web References	https://www.tensorflow.org/learn	10Hrs

Course Code: CCSAI0302	Course Name: High Performance Computing	L	T	P	C
Course Offered in: CSE/CS/IT/CSIT/CSE(AI)/CSE(AIML)/ECE/EEE/ELCE/ME		3	0	0	3

Pre-requisite: C Language, Fundamentals of Assembly Language

Course Objectives:

This course introduces the fundamentals of High-Performance Computing (HPC) and modern computing paradigms like cloud, grid, cluster, and quantum computing. It covers computing architectures, parallelism concepts, distributed memory models, and message-passing techniques. Students will also develop parallel applications using OpenMP and explore fault-tolerant mechanisms in cluster systems.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Level**
CO1	Identify the need and applications of High Performance Computing and emerging computing paradigms.	K2
CO2	Analyse computing architectures and parallelism concepts to assess HPC system performance.	K3
CO3	Analyse distributed memory techniques and message passing for parallel algorithm design.	K3
CO4	Analyse fault tolerance methods in cluster computing systems.	K3
CO5	Develop parallel programs using OpenMP for efficient shared memory processing.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	3	-	-	-	-	-	-	3	2	2
CO2	3	3	3	2	3	-	-	-	1	-	1	3	3	3
CO3	3	3	3	3	3	-	-	-	1	1	1	3	3	3
CO4	3	3	2	3	3	-	-	1	1	1	2	3	3	3
CO5	3	3	3	2	3	-	-	-	1	1	1	3	3	3

Course Contents / Syllabus

Unit 1	Introduction to HPC	09 hours

Computational Thinking, Computing, Why High-Performance Computing, Applications of High-Performance Computing, Parallel programming Software Platforms and its significance, Cloud computing, Grid computing, Cluster computing and Quantum Computing, Multi-core CPUs, Graphical Processing Units.

Unit 2	Computing Architectures	09 hours

SISD, SIMD, MISD, MIMD, Memory Hierarchy, Data parallelism, Task parallelism, Bit-level parallelism, Instruction-level parallelism, Concurrency, Decomposition, Mapping, PRAM, NUMA, Multithreading vs Multiprocessing, shared memory model, Open MP, distributed memory model, Message passing interface. Performance measures: Speedup, efficiency and scalability.

Unit 3	Distributed Memory	09 hours

Distributed memory and Message passing networks, Broadcast, Reduction, Parallel Prefix Sum, Scatter, Gather, Network topologies for parallel computing, Network optimization. Distributed BFS, Graphs and adjacency matrix, Matrix based BFS, CUDA programming, Parallel matrix operations, Sparse vs Dense matrices.

Unit 4	Cluster Computing	09 hours

BLAS, LAPAC, Cluster Computing, Clustering Models, Clustering Architectures, Clustering Architectures key factors, types of clusters, Mission critical Vs Business Critical Applications, Fault Detection and Masking Algorithms, Check pointing, Heartbeats, Watchdog Timers, Fault recovery through Failover and Failback Concepts.		
Unit 5	**OpenMP**	**09 hours**
OpenMP, Key concepts, Fork-join model, Goals of OpenMP, Supported platforms and compilers, OpenMP API components: directives, runtime routines, environment variables, General code structure, Core syntax and structured blocks, Compiling OpenMP programs, Parallel region construct, Creating and managing threads, Specifying the number of threads, Thread management and thread IDs, Applications of OpenMP.		
Total Lecture Hours		**45 hours**

Textbook:

1. Jack Dongarra, Geoffrey Fox, and Ken Kennedy, *"The Sourcebook of Parallel Computing"*, Morgan Kaufmann, 2003.
2. David A. Bader (Ed.), *"Petascale Computing: Algorithms and Applications"*, Chapman & Hall/CRC Computational Science Series, 2007.
3. Robert Robey and Yuliana Zamora, Parallel and High Performance Computing (1 ed.), Manning, 2021. ISBN 9781638350385, 1638350388.
4. Thomas Sterling, Maciej Brodowicz and Matthew Anderson, High Performance Computing: Modern Systems and Practices (1 ed.), Elsevier Science, 2017. ISBN 9780124202153, 0124202152 .

Reference Books:

1. Thomas Sterling, Matthew Anderson, and Maciej Brodowicz, "High Performance Computing: Modern Systems and Practices", Morgan Kaufmann, 2017.
2. Barbara Chapman, Gabriele Jost, and Ruud van der Pas, "Using OpenMP: Portable Shared Memory Parallel Programming", MIT Press, 2007.
3. Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar, "Introduction to Parallel Computing", 2nd Edition, Addison-Wesley, 2003.
4. Tim Mattson et al. The OpenMP Common Core: A hands on exploration, SC 2018.
5. Tim Mattson and Larry Meadows. A “Hands-on” Introduction to OpenMP. SC 2008.
6. Ruud van der Pas. OpenMP Tasking Explained. SC 2013. • Peter Pacheco. An Introduction to Parallel Programming.
7. Blaise Barney. OpenMP. https://computing.llnl.gov/tutorials/openMP/

NPTEL/ Youtube/ Faculty Video Link:

Unit 1	https://www.youtube.com/watch?v=zJ_D6AiGlV8
Unit 2	https://www.youtube.com/watch?v=FTXikoQr46U
Unit 3	https://www.youtube.com/watch?v=gE8KgD9D69k
Unit 4	https://www.youtube.com/watch?v=kHV6wmG35po
Unit 5	https://www.youtube.com/watch?v=JeuP0oMQ8sQ

LAB Course Code: CCSE0353A	LAB Course Name: Operating Systems Lab	L	T	P	C
Course Offered in: CSE/CSE-R/IT/CS/AI/AIML/ IOT/DS/CYS		0	0	4	2

Pre-requisite: Basic knowledge of computer fundamentals, C programming, Data structures and Computer organization.

Course Objectives: The course aims to provide hands-on experience with Linux and shell programming, while the lab focuses on implementing and analyzing key OS algorithms and simulating modern operating systems.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Execute basic Linux commands and shell scripts to automate file management and system administration tasks.	K3
CO2	Implement and compare various CPU scheduling algorithms, process synchronization solutions using semaphores and deadlock handling algorithms.	K4
CO3	Simulate memory allocation techniques and page replacement algorithms, disk management strategies and explore modern OS features including virtualization and distributed computing.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	1	1	1	3	–	–	2	2	–	3	2	1	**2**
CO2	3	3	3	2	2	–	–	2	2	–	2	2	2	**3**
CO3	3	2	3	2	3	–	–	2	2	–	3	2	3	**3**

List Of Practical's (Indicative & Not Limited To)

S.No	Experiment	CO
1	Installation of Linux Operating system.	CO1
2	**Implementation of Linux Commands** i. Display system information using uname, hostname, and date etc. ii. File operations using cat, touch, cp, mv, rm, and chmod ,umask etc. iii. Create, view, and navigate directories using mkdir, rmdir, cd, pwd, ls etc. iv. Disk Commands df,du,mount,unmount,mkfs,fsck etc.	CO1
3	**Implementation of Linux Commands (continued)** v. Use redirection and piping in commands. vi. File compression and archiving using tar, gzip, zip, unzip etc. vii. Process commands ps,kill, killall,nice, pgrep, top,htop etc. viii. Network commands ifconfig, ping, netstat, host,ip route etc. ix. Administrator Commands Adduser,Passwd, deluser, usermod, groupadd etc	CO1
4	**Shell Scripting Programming** i. Write a shell script to ask your name, program name and enrollment number and print it on the screen. ii. Write a shell script to find the sum, the average and the product of the four integers entered. iii. write shell script to find average of numbers given at command line	CO1

	iv. Write a shell program to exchange the values of two variables.	
	v. Write a shell program to Print Numbers 1 to 10 using while & do while loop.	
	vi. Write a shell program to Print Numbers 1 to 10 using for loop.	
5	**Shell Scripting Programming (continued)**	CO1
	vii. Write a shell script to display the digits which are in odd position in a given 5-digit number**.**	
	viii. Write a shell program to search for a given number from the list of numbers provided using binary search method.	
	ix. Write a shell program to concatenate two strings and find the length of the resultant string.	
	x. Write a shell script to find the smallest of three numbers.	
	xi. Write a shell program to count number of words, characters, white spaces and special symbols in a given text.	
6	Implement FCFS CPU Scheduling algorithm.	CO2
7	Implement the SJF CPU Scheduling algorithm (For both Pre-emptive and non-pre-emptive).	CO2
8	Implement PRIORITY CPU Scheduling Algorithm (For both Pre-emptive and non-pre-emptive).	CO2
9	Implement Round-Robin CPU Scheduling Algorithm.	CO2
10	Implement Multi-Level Queue CPU Scheduling algorithm.	CO2
11	Implement Multilevel Feedback Queue CPU Scheduling Algorithm.	CO2
12	Execute the RACE Condition of Process Synchronization.	CO3
13	Implement the Producer–consumer problem using semaphores.	CO3
14	Design a code and implement the Dinning Philosopher problem.	CO3
15	Implement Banker's algorithm of Deadlock Avoidance.	CO3
16	Execute an algorithm for Deadlock Detection.	CO3
17	Implement Contiguous memory variable size partition scheme.	CO3
18	Simulate the First-Fit contiguous memory allocation technique.	CO3
	Total Hours: 30 hrs.	

LAB Course Code: CCSE0351	**LAB Course Name:** Data structure and algorithms lab-II	**L**	**T**	**P**	**C**
Course Offered in: CSE/CS/CSR-R/M.TECH(INT) /CSE(AIML)/CSE(DS)/CSE(CS)/ /IT//CSE(AI)		**0**	**0**	**2**	**1**

Pre-requisite: C, Python

Course Objectives:. Learn to implement linear data structures

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

S.No.	Course Outcome	Bloom's Level
1.	Implementation of tree data structures for basic operations like insertion, deletion, searching and traversal	K3
2	Implementation of algorithms based on graph data structures for solving real world problems.	K3
3	Implementing Dynamic Programming, Backtracking, Branch and Bound algorithms to solve complex data efficiently and effectively.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	1	1	2	1	1	2	1	–	2	1	**2**
CO2	3	3	2	1	1	2	1	1	2	1	–	2	1	**2**
CO3	3	3	2	2	2	2	1	1	2	2	–	2	1	**3**

List Of Practical's (Indicative & Not Limited To)	Mapped CO
1. Write a program to implement an in-order traversal of a binary tree and print the nodes.	CO1
2. Write a program to implement a pre-order traversal of a binary tree and print the nodes.	CO1
3. Write a program to implement a post-order traversal of a binary tree and print the nodes.	CO1
4. Write a program to count the number of nodes in a binary tree.	CO1
5. Write a program to find the height of the tree.	CO1
6. Write a program to check if the binary tree is balanced or not.	CO1
7. Write a program to search a number in a Binary Search Tree (BST).	CO1
8. Write a program to insert a node in a Binary Search Tree (BST).	CO1
9. Write a program to delete a node from a Binary Search Tree (BST).	CO1
10. Write a program to implement a max-heap and perform heap sort on an array of integers.	CO1
11. Write a program to implement a priority queue using a max heap.	CO1
12. Write a program to create a graph using an adjacency matrix.	CO2
13. Write a program to create a graph using an adjacency list.	CO2
14. Write a program to perform Depth-First Search (DFS) on a graph.	CO2
15. Write a program to perform Breadth-First Search (BFS) on a graph.	CO2
16. Write a program to check if there is a path between two nodes in a graph using DFS.	CO2
17. Write a program to find all the vertices reachable from a given vertex in a graph.	CO2

18. Write a program to detect a cycle in an undirected graph using DFS.	CO2
19. Write a program to detect a cycle in a directed graph using DFS.	CO2
20. Write a program to find the degree of each vertex in an undirected graph.	CO2
21. Write a program to count the number of connected components in an undirected graph.	CO2
22. Write a program to implement Dijkstra's Algorithm.	CO3
23. Write a program to implement Prim's Algorithm.	CO3
24. Write a program to implement Kruskal's Algorithm.	CO3
25. Write a program to implement Floyd–Warshall's all-pairs shortest path algorithm.	CO3
26. Write a program to implement Bellman–Ford Algorithm.	CO3
27. Write a program to implement Longest Common Subsequence (LCS) using Dynamic Programming..	CO3
28. Write a program to implement 0/1 Knapsack using Dynamic Programming.	CO3
29. Write a program for Matrix Chain Multiplication using Dynamic Programming.	CO3
30. Write a program to implement Sum of Subsets problem using backtracking.	CO3
Total Hours: 30 hrs.	

LAB Course Code: CCSML0351	**LAB Course Name Machine Learning Lab**	**L**	**T**	**P**	**C**
Course Offered in: CSE(AI)/CSE(AIML)		**0**	**0**	**2**	**1**

Pre-requisite: Python Programming,

Course Objectives:

The course aims to develop practical skills in data preprocessing, feature engineering, supervised and unsupervised machine learning, model evaluation, anomaly detection, and ensemble techniques through hands-on implementation of real-world case studies using Python-based machine learning tools and algorithms.

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

CO1	Apply data preprocessing and feature engineering techniques on datasets	K3
CO2	Analyze and evaluate supervised learning models	K4
CO3	Investigate clustering, probabilistic models, and ensemble techniques for data analysis	K4

CO-PO Mapping (Scale 1 : Low, 2: Medium, 3: High)

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	1	2	2	–	–	–	–	1	1	3	1	**2**
CO2	3	3	2	2	3	–	–	–	1	1	2	3	3	**3**
CO3	2	3	2	3	3	–	–	–	1	2	2	3	3	**3**

List Of Practical's (Indicative & Not Limited To)

S. No	Experiment (Case Study Based Question)	CO Mapping
1	Load a dataset using Python and perform basic data exploration. Identify number of rows, columns, data types, and summary statistics.	CO1
2	Given a dataset with missing values, implement techniques to handle missing data.	CO1
3	Convert categorical features into numerical format using appropriate encoding techniques.	CO1
4	Apply normalization and standardization techniques on dataset features and compare the results.	CO1
5	Detect and remove outliers from a dataset using statistical methods and visualize the results.	CO1
6	Select important features from a dataset using suitable feature selection techniques.	CO1
7	Reduce dimensionality of a dataset using Principal Component Analysis (PCA) and analyze the variance retained.	CO1
8	Implement simple linear regression on a dataset and predict the output variable.	CO2
9	Build a multiple regression model using multiple input features and evaluate its performance.	CO2

10	Implement binary classification using logistic regression and evaluate model accuracy.	CO2
11	Evaluate the performance of a classification model using confusion matrix, precision, recall, and F1-score.	CO2
12	Implement k-Nearest Neighbors (k-NN) algorithm for classification and analyze the effect of different values of K.	CO2
13	Build and visualize a decision tree classifier for a given dataset.	CO2
14	Apply Naïve Bayes classifier on a dataset and evaluate its performance.	CO2
15	Implement Support Vector Machine (SVM) classifier and compare its performance with other classifiers.	CO2
16	Cluster a dataset into groups using K-Means algorithm and visualize the clusters.	CO3
17	Perform hierarchical clustering on a dataset and represent the results using a dendrogram.	CO3
18	Implement DBSCAN clustering algorithm and identify noise/outlier points.	CO3
19	Implement AdaBoost algorithm and analyze its performance improvement over base learners.	CO4
20	Detect anomalies in a dataset using Gaussian-based anomaly detection techniques.	CO4

\

Course Code:CCSE0351	Course Name: Object Oriented Techniques using Java	L	T	P	C
Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AIML)/CSE(IOT)/CSE(AI)/CSE(DS)/CSE-R/M.Tech int		0	0	6	3

Pre-requisite: Basic Programming Knowledge,Knowledge of any programming language (e.g., C, C++, Python)

Course Objectives:

The objective of this course is to understand the object-oriented methodology, and its techniques to design stand alone and GUI applications using hands-on engaging activities.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the concepts of object-oriented programming and relationships among them needed in modeling.	K2
CO2	Demonstrate the Java programs using OOP principles and also implement the concepts of lambda expressions	K3
CO3	Analyze packages with different protection level resolving namespace collision and implement the error handling concepts for uninterrupted execution of Java program.	K4
CO4	Analyze Concurrency control, I/O Streams and Java Socket Programming Concepts.	K4
CO5	Design and develop the GUI based application, Generics and Collections in Java programming language to solve the real-world problem	K6

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3	2	2	1	–	1	–	2	3	3	**2**
CO2	3	3	3	3	2	2	1	–	1	1	2	3	2	**3**
CO3	3	3	3	3	3	2	2	–	2	1	2	3	3	**3**
CO4	3	3	3	3	3	2	2	1	2	1	2	3	2	**3**
CO5	3	3	3	3	3	2	2	1	2	1	2	3	3	**3**

Course Contents / Syllabus

Unit 1	**Basics of Java Programming**	**hours**

Object Oriented Programming: Introduction and Pillars of OOP with real life example, jvm architecture and its components
Modelling Concepts: Introduction, Class Diagram and Object Diagram, UML concepts: Association, Composition, aggregation, realization, and Generalization.
Control Statements: Decision Making, Looping and Branching, Argument Passing Mechanism: Command Line Argument, Console Input.

Class and Object: Object Reference, Constructor, Abstraction: Abstract Class, Interface and its uses, Defining Methods, Use of "this" and "super" keyword, Garbage Collection and finalize () Method etc.

Unit 2	**OOPs features, arrays and lambda expressions**	**16 hours**

Inheritance: Introduction and Types of Inheritance in Java, Access Modifiers, Constructors and super constructor in Inheritance.

Polymorphism: Introduction and Types, Overloading and Overriding. **Lambda expression:** Introduction and Working with Lambda Variables. **Arrays:** Introduction and its Types. Jagged Array with example

Unit 3	**Packages, Exception Handling and String Handling**	**16 hours**

Packages: Introduction and Types, Access Protection in Packages, Import and Execution of Packages. E**xception Handling, Assertions and Localizations:** Introduction and Types, Exceptions vs. Errors, Handling of Exception. Finally, Throws and Throw keyword, Multiple Catch Block, Nested Try and Finally Block, Customized Exceptions, Tokenizer. Assertions and Localizations Concepts and its working. **String Handling:** Introduction and Types, Operations, Immutable String, Method of String class, String Buffer and String Builder class.		
Unit 4	**Concurrency in Java and I/O Stream**	**16 hours**
Threads: Introduction and Types, Creating Threads, Thread Life-Cycle, Thread Priorities, Daemon Thread, Runnable Class, Synchronizing Threads etc. **I/O Stream:** Introduction and Types, Common I/O Stream Operations, Interaction with I/O Streams Classes. character and byte oriented stream classes with example **Java Socket Programming**: Introduction and types(TCP, UDP), java socket program with server-side and client-side by using connection.		
Unit 5	**GUI Programming, Generics and Collections**	**16 hours**
GUI Programming: Introduction and Types, Swing, AWT, Components and Containers, Layout Managers and User-Defined Layout and Event Handling. **Generics:** Introduction to Generic Classes, types of generic defined in brief, bounded type parameter(Upper and Lower bound), Initializing a Generic Object, Classes, Methods and Interfaces Use enumerated type. **Collections:** Introduction, main interfaces of collections(Collection, List Set, Map, Queue), classes of collections(ArrayList,Linked list, HashSet, HashMap and TreeSet) and methods(List, Set Map) Collection using Iterators		
	Total Lecture Hours	**80 hours**
Textbook:		
S.NO	**Book Title**	**Author**
1.	” Java: A Beginner’s Guide”, McGraw-Hill Education 2nd edition	Herbert Schildt
2.	“Programming with Java A Primer”, TMH, 4th edition.	E Balagurusamy,
Reference Books:		
S.NO	**Book Title**	**Author**
1.	“Core Java Volume I – Fundamentals”, Prentice Hall	Cay S. Horstmann
2.	” Effective Java”, Addison Wesley	Joshua Bloch
3.	Java - The Complete Reference”, McGraw Hill Education 12th edition	Herbert Schildt
Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012540972274925568 1063_shared/overview	46h 13m
Video Lectures	https://www.youtube.com/watch?v=iw1Xf_33YM0&list=PLQEaRBV9gAFujcBWJhBT2XXsuMl IfETBy	30 h
Web Referen	1. https://www.geeksforgeeks.org/dsa/introduction-of-object-oriented-programming/	20 m

ces	2. https://en.wikipedia.org/wiki/Object-oriented_programming	58 m

Course Code: CNC0302/CNC0402	Course Name: Environmental Science	L	T	P	C
Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AIML)/CSE(IOT)/CSE(AI)/CSE(DS)/CSE-R/M.Tech (Int) / MCT / CSE (CYS)		2	0	0	NA

Pre-requisite: Basic knowledge of biology, chemistry, ecology, geology, mathematics, and understanding of human impacts on natural systems.

Course Objectives:

To help the students in realizing the inter-relationship between man and environment and help the students in acquiring basic knowledge about environment.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand the basic principles of ecology and the environment. Ecosystem: Basic concepts, components of ecosystems, food chains and food webs. Ecological pyramids	**K2**
CO2	Understand the different types of natural recourses like food, forest, Minerals and energy and their conservation	**K2**
CO3	Understand the different types of pollution, pollutants, their sources, effects, and their control methods.	**K2**
CO4	Understand the basic concepts of sustainable development, Environmental Impact Assessment (EIA) and different acts related to environment	**K2**

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO MAPPING	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	-	3	3	2	2	-	2	2	2
CO2	3	3	2	2	-	3	3	2	2	-	2	2	2
CO3	3	3	2	2	-	3	3	2	2	-	2	2	2
CO4	3	3	2	2	-	3	3	3	2	-	2	2	2

Course Contents / Syllabus

UNIT 1	Basic Principle of Ecology and Biodiversity	5 HOURS

Definition, Scope and basic principles of ecology and environment. Ecosystem: Basic concepts, components of ecosystem. Food chains and food. Webs. Ecological pyramids, Energy flow in ecological systems, Characteristics of different ecosystems. Biogeochemical Cycles: Importance, gaseous and sedimentary cycles. Carbon, Nitrogen, Phosphorus and Sulphur Cycles. Biodiversity and their importance, Threats to biodiversity, major causes, extinction's, vulnerability of species to extinction, IUCN threat categories, Red data book. Strategies for biodiversity conservation, principles of biodiversity conservation in-situ and ex-situ conservation strategies Mega diversity zones and Hot spots, concepts, distribution and importance.

UNIT 2	Natural Resources and Ecological succession	5 HOURS

Natural resources and associated problems. Forest resources: Use and over- exploitation, deforestation. Timber extraction, mining, dams and their effects on forest and tribal people. Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources. Food resources: World food problems, changes caused by agriculture and over- grazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, and salinity. Land resources: Land as a resource, land degradation, man induced landslides. Equitable use of resources for sustainable lifestyles. Non-Renewable Energy Resources: Fossil fuels and their reserves, Nuclear energy, types, uses and effects, Renewable Energy Resources: hydropower, Solar energy, geothermal, tidal and wind energy, Biomass energy, biogas and its advantages. Ecological succession-Types, stages, examples of ecological succession

UNIT 3	Pollution and Waste Management	5 HOURS

Air pollution: sources of air pollution, Primary and secondary air pollutants. Origin and effects of SOX, NOX, Cox, CFC, Hydrocarbon, control of air pollution. Water pollution: sources and types of water pollution, Effects of water pollution, Eutrophication, Soil pollution: Causes of soil pollution, Effects of soil pollution, Major sources of and effects of noise pollution on health, Radioactive and thermal pollution sources and their effects on surrounding environment. Solid waste disposal and its effects on surrounding environment, Introduction to E- Waste, Types and classification of E- Waste, Impacts of E- Waste on environment and human health,E-Waste management and recycling., Climate change, global warming, acid rain, ozone layer depletion.

UNIT 4	Environmental Assessment and Legislation	5 HOURS

Women education, Role of NGOs regarding environmental protection, Bio indicators and their role, Natural disasters and disasters management, Aims and objectives of Environmental Impact Assessment (EIA). Salient features of following Acts: Environmental Protection Act, 1986, Wildlife (Protection) Act, 1972.Water (Prevention and control of pollution) Act, 1974. Forest (Conserving) Act, 1980. Definition and concept of sustainability, impacted areas of sustainable development, Global initiative, and issues on sustainable development UN SDGs, System Thinking and Sustainability.		
	Total Lecture Hours	**20 HOURS**
Textbook:		
S.No	**Book Title with publication agency & year**	**Author**
1	Brady, N.C. 1990. The nature and properties of Soils, Tenth Edition. Mac Millan Publishing Co., New York	Brady, N.C
2	Sodhi G.S. 2005, Fundamentals of Environmental Chemistry: Narosa Publishing House, New Delhi.	Sodhi G.S
3	Dash, M.C. (1994), Fundamentals of Ecology, Tata Mc Graw Hill, New Delhi.	Dash, M.C
Reference Books:		
1	Rao M.N. and H.V.N. Rao, 1989 : Air Pollution, Tata McGraw Hill Publishing Co. Ltd., New Delhi	Rao M.N. and H.V.N. Rao
2	A Text Book of environmental Science By Shashi Chawla	Shashi Chawla
Self-Study Content Links:		
MOOC'S		
VIDEO LECTURES	https://www.youtube.com/watch?v=T21OO0sBBfc https://www.youtube.com/watch?v=qt8AMjKKPDo https://www.youtube.com/watch?v=brF0RWJyx9w https://www.youtube.com/watch?v=GK_vRtHJZu4	**7 min 42 sec** **15 min** **3 min 25 sec** **4 min 25 sec**
WEB REFERENCES	https://www.youtube.com/watch?v=yqev1G2iy20, https://www.youtube.com/watch?v=_74S3z3IO_I, https://www.youtube.com/watch?v=jXVw6M6m2g0	**28 min** **31 min 35 sec** **11 min**

SEMESTER-IV
SYLLABUS

Course Code: CASCC0401	**Course Name:** Employability Skill Development - II	**L**	**T**	**P**	**C**
Course Offered in: IV Semester **Note:** Section C - offered in B.Tech. III/IV SEM		**0**	**0**	**4**	**2**

Pre-requisite: Students should have successfully completed communication skills courses in the first year.

Course Objectives

Develop proficiency in C++ programming, data management, and project development by applying concepts, teamwork, and problem-solving to build practical applications. Enhance quantitative aptitude skills through business mathematics concepts, enabling accurate problem-solving in financial calculations, ratios, and real-world numerical applications. To develop the ability to communicate effectively and empathetically in workplace situations while strengthening active listening, adaptability, and cross-cultural awareness for navigating complex professional environments.

Course Outcome		Bloom's Knowledge Level (KL)
CO1	Apply C++ programming concepts, file handling, and STL techniques to develop efficient programs for data storage and manipulation.	**K3**
CO2	Design and implement a C++ -based project by applying planning, teamwork, and problem-solving skills.	**K4**
CO3	Solve real-life business math problems involving percentages, profit and loss, discounts, interest, averages, mixtures, and ratios using appropriate mathematical methods.	**K3**
CO4	Apply spoken English skills to participate effectively in standard workplace routines, including daily stand-ups, reporting, and collaborative problem-solving.	**K3**
CO5	Implement and adapt communication styles to suit dynamic workplace communication situations and deliver context-sensitive, and logically structured verbal responses under time constraints.	**K4**

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	–	2	3	1	–	3	2	2	2	–	–	**2**
CO2	3	3	3	3	3	2	1	3	2	2	2	–	–	**3**
CO3	3	3	2	1	–	–	–	1	2	2	2	–	–	**2**
CO4	–	–	–	–	1	2	–	3	2	2	–	–	–	**2**
CO5	–	2	1	–	1	2	–	3	2	2	1	–	–	**2**

Course Contents / Syllabus

UNIT 1:	**FOUNDATIONS OF C++ PROGRAMMING AND DATA MANAGEMENT TECHNIQUES**	**8 hours**

Core Fundamentals of C++, Functions & Program Structure, Arrays & Strings, File **Handling:** File streams (ifstream, ofstream, fstream), Reading/writing text files, saving user data (scores, records, logs) **STL Basics:** vector, array, map, unordered_map set Algorithms (sort, find)

UNIT 2	**PROJECT/GAME DEVELOPMENT**	**8 hours**

Project Planning & Development (Teams, roles, idea pitching and implement in C++), Mini Project, Project Demonstration and Review

UNIT 3	**BUSINESS MATH I**	**16 hours**

Percentage, Profit and Loss, Discount, Simple Interest and Compound Interest, Average, Ratio & Proportion, Partnership, Mixture & Allegation, Clock , Calendar.

UNIT-4	WORKPLACE INTERACTION ESSENTIALS	12 hours

- The Professional Shift: Navigating the transition from academic to professional tone. Using appropriate register when addressing peers vs. supervisors.
- Active Listening & The Feedback Loop: Moving beyond passive hearing. Practicing the SBI (Situation-Behaviour-Impact) model for peer feedback and clarifying multi-step technical instructions.
- Structured Micro-Speaking: The art of the concise update. Delivering 1-2 minute "Stand-up" reports (What I did, What I will do, Blockers) and explaining basic technical processes to laymen.
- Guided Interpersonal Simulations: Role-playing everyday scenarios: politely interrupting a meeting, seeking assistance on a delayed task, and confirming understanding of complex briefs using paraphrasing techniques.
- Digital & Hybrid Nuances: Verbal signalling in virtual meetings, managing "overtalk," and translating concise written chat messages into spoken professional updates.

UNIT-5	WORKPLACE COMMUNICATION IN ACTION	12 hours

- Dynamic Problem-Based Communication: Simulating the "War Room." Explaining sudden technical failures, proposing immediate solutions, and negotiating revised deadlines with stakeholders.
- Listening Under Pressure & Global Adaptability: Interpreting fast, accented, or incomplete instructions (incorporating CEFR-level listening benchmarks). Decoding tone, urgency, and indirect cultural communication styles.
- Collaborative Conflict Resolution: Managing disagreements in team settings using "I-statements" and the "Yes, and…" technique to build consensus without escalating tension.
- The Persuasive Professional: Delivering evidence-based arguments using the PREP (Point, Reason, Example, Point) framework to justify a technical choice or resource allocation.
- Real-Time "Crisis" Simulation (Capstone): A comprehensive role-play where student groups are given a sudden project change or client issue. They must conduct an emergency meeting, delegate tasks verbally, and present a united response to the "client."

Total: 56 Hours

Reference Books:

S. No.	Book Title	Author
1.	A Project-Based Introduction to Programming	Access Point Publishing
2.	Programming: Principles and Practice Using C++	Bjarne Stroustrup
3.	Effective Modern C++	Scott Meyers
4.	Quantitative Aptitude	R S Aggarwal
5.	Verbal & Non-Verbal Reasoning	R S Aggarwal
6.	*Crucial Conversations Tools*	Kerry Patterson et al.
7.	*HBR Guide to Persuasive Presentations*	Nancy Duarte
Self Study Content Link:		Duration(in hours)
MOOCs	nptel.ac.in/courses/109104107	8 weeks
Video Lectures	Employability skills - YouTube	33 m
Web Reference	Employability Skills Self-Management Guide | PDF | Goal | Self Esteem	-

Course Code: CCSE0401	**Course Name: Database Management Systems**	**L**	**T**	**P**	**C**
Course Offered in: CSE/CSE-R/IT/M.Tech Int./CSE (Twin)/IT (Twin)/CSE(Prof)/IT (Prof)/M&C/AI/AI (TWIN)/AIML/AIML (TWIN)/CS/CYS/DS/IOT		**2**	**0**	**0**	**2**

Pre-requisite: Basic understanding of computer fundamentals such as architecture, storage and hardware. In addition, familiarity with data structures, algorithms, and basic programming concepts will be beneficial.

Course Objectives: The objective of the course is to introduce about database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information in relational & non-relational databases.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Apply ER model for conceptual design of the database.	K3
CO2	Execute SQL and apply the normalization to improve the database design.	K3
CO3	Implement complex queries in database with different applications.	K3
CO4	Understand transaction and concurrency control and execute the concept of PL/SQL.	K3
CO5	Analyze and implement effective Relational and Non-Relational database for real-world problems.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3	2	1	1	1	2	1	2	3	1	**2**
CO2	3	2	3	3	2	2	1	2	1	2	2	3	2	**3**
CO3	3	3	2	3	3	2	–	2	1	–	2	3	3	**3**
CO4	3	2	2	2	2	2	1	–	1	1	1	3	3	**3**
CO5	2	2	2	2	3	2	1	–	1	2	2	3	3	**3**

Course Contents / Syllabus

UNIT-1	**INTRODUCTION ABOUT THE DATABASE CONCEPTUAL DESIGNING**	**9 hours**

Basic Concept: Database system concept, architecture, History of Database, Data Independence, Database system Vs File system, Data models & Types of Data Models, schema and instances. **Data Modelling using the Entity Relationship Model:** ER model concepts, Degree of relationship, Notation for ER diagram, mapping constraints, Generalization, Aggregation, Reduction of an ER diagrams to tables. Extended ER Diagram & reduction of EER. Codd Rules. **Introduction on SQL:** Types of SQL commands: - DDL, DML, DCL, TCL. Basic of Relation Algebra & Operations, Query Optimization.

UNIT-2	**BASIC OF SQL & NORMALIZATION**	**9 hours**

Keys & Types of Keys: Super key, Candidate Key, Primary Key, Alternative Key, Foreign Key, unique. Constraints and Types of Constraints. **Use of Functions and Clause:** Aggregate Functions, Scalar Functions, Where, Group by, Having and Order by. **Use of Predicates:** SQL Operators. Like, Between, Aliases, distinct, limit. Functional Dependencies, attribute Closure, Normalization & Types of Normalization, Candidate Key, FD Preservation, Lossy and Lossless decomposition, Canonical Cover of FD's, Denormalization.

UNIT-3	**INTRODUCTION OF COMPLEX QUERIES**	**9 hours**

Use of Operators: Union, Intersect, Minus, Cartesian Product, Joins:Inner Join: Natural Join, Equi Join & Non Equi Join, Outer Join: Left Outer Join, Right Outer Join and Full Outer Join, Division Operator. **Nested Query or Sub Query:** IN, NOT IN, Exists, Not Exists, All and Any. Managing Indexes, Synonyms and Sequences, Managing Views. **Introduction of PL/SQL:**

Implementation of PL/SQL Function, Stored Procedure, Cursor, Trigger, Exception Handling. **Database connectivity:** Database Connectivity with any of the Programming Languages.		
UNIT 4	**TRANSACTION AND CONCURRENCY CONTROL**	**9 hours**
Transaction system: Life cycle of transaction, ACID Properties Schedule & Types of Schedule, Serializability, Recoverability, Deadlock Handling. **Concurrency Control Techniques:** Concurrency Control, Concurrency control Techniques: Locking Techniques, Timestamping, Validation Based Protocol, Transaction & Data Control: -Grant, Revoke, commit & Rollback.		
UNIT 5	**INTRODUCTION OF NOSQL WITH MONGODB**	**9 hours**
Introduction to NoSQL: NoSQL Data Models, Overview of NoSQL Databases with their Types, Uses & Features of NoSQL Document Databases, CAP theorem, BASE Vs ACID, Comparison of relational databases to NoSQL stores, uses and deployment: MongoDB, Cassandra, HBASE, Neo4j, Redis and Riak. **Introduction to MongoDB:** Introduction and Features of MongoDB, MongoDB Operators, MongoDB Collection & Document, MongoDB Shell & their commands, CRUD operations. Introduction of Cloud Database. MongoDB Cloud product : Stitch, Atlas & Cloud Manager.		
Total Lecture Hours		**45 hours**

Textbook:

S.No	Book Title	Author
1.	Database Concepts, McGraw Hill, 7th Edition, 2020	Avi Silberschatz Henry F. Korth S. Sudarshan
2.	Fundamentals of Database Systems, Addision Wesley, 7th edition, 2016	R. Elmasri and S. B. Navathe

Reference Books:

S.No	Book Title	Author
1.	Database Systems: A practical Approach to Design, Implementation and Management. Pearson Education,3rd Edition, 2007.	Thomas Connolly , Carolyn Begg
2.	NoSQL and SQL Data Modeling: Bringing Together Data, Semantics, and Software, Ted Hills, 1st Edition,2016.	Ted Hills

Self-Study Contents Links:		Duration (in Hours)
MOOCs	nptel.ac.in/courses/106105175	16 hours
Video Lectures	DBMS L1 Inauguration & Introduction	30 hours
	DBMS L2 Introduction to Relational Model	
	DBMS L8C Entity Relationship Model	
	DBMS L8D Entity Relationship Model (Problem Solving and Discussion)	
	DBMS L4A Joins, Set Operations and Aggregate Functions	
	DBMS L4B Joins, Set Operations and Aggregate Functions	
	DBMS L9B Relational Database Design	
	DBMS L9C Relational Database Design	
	DBMS L5A Nested Subqueris	
	DBMS L15 Transactions	
	DBMS L16A Concurrency Control	
	DBMS L16B Concurrency Control	
	DBMS L16C Concurrency Control	
	L64: Non-relational database management systems | nosql & cap theorem	
	DBMS L19 Distributed Data Stores and NoSQL Databases	
Web references	https://www.geeksforgeeks.org/dbms/dbms/	2 hours
	https://www.scribd.com/document/666843210/Database-Management-System-DBMS-HandWritten-Notes	

Course Code: CCSML0452	**Course Name: Deep Learning**	**L**	**T**	**P**	**C**
Course Offered in: CSE(AI)/CSE(AIML)		**3**	**0**	**0**	**3**

Pre-requisite: Python, Machine Learning

Course Objectives:

This course aims to provide students with a comprehensive understanding of Artificial Neural Networks (ANNs) and Deep Learning concepts, forming the foundation for modern intelligent systems. It focuses on the development of Convolutional Neural Network (CNN) models for image classification and recognition tasks, while also enabling students to implement object detection techniques and Recurrent Neural Network (RNN) models for sequential data analysis. The course further explores Autoencoders and feature extraction methods for efficient data representation and learning.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Analyze ANN models and evaluate their performance metrics.	K4
CO2	Examine CNN models for multi-class image classification.	K4
CO3	Apply deep learning techniques for object detection and recognition.	K3
CO4	Analyze RNN models for time series forecasting and sequence analysis.	K4
CO5	Apply dimensionality reduction techniques for feature extraction and data representation.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	1	1	1	–	2	1	2	3	**2**
CO2	3	3	2	2	3	–	–	–	1	1	1	3	3	**3**
CO3	3	3	3	2	3	–	–	–	2	1	1	3	3	**3**
CO4	2	3	2	2	2	–	–	1	2	1	1	3	3	**3**
CO5	2	2	3	2	3	–	–	–	1	1	1	3	3	**3**

Course Contents / Syllabus

UNIT-1	**INTRODUCTION**	**10 hours**

Artificial Neural Network: Neuron, Nerve structure and synapse, Artificial Neuron and its model, activation functions, Neural network architecture: Single layer and Multilayer feed forward networks, Various learning techniques; Perception and Convergence rule, Hebb Learning. Perceptron's, Multilayer perceptron, Gradient descent and the Delta rule, Multilayer networks, Derivation of Backpropagation Algorithm, Introduction to Deep Learning. Regularization for Deep Learning -Parameter Norm Penalties, Dataset Augmentation, Noise Robustness, Early Stopping, Dropout, Batch Normalization.

UNIT-2	**CONVOLUTION NEURAL NETWORK**	**07 hours**

Introduction to Computer Vision and CNN, Train a simple convolutional neural net, Explore the design space for convolutional nets, Pooling layer motivation in CNN, Padding and Edge Detection, Strided Convolution, Design a convolutional layered application, Understanding and visualizing a CNN, Transfer learning and fine-tuning CNN, Image classification, Text classification, Image classification and hyper-parameter tuning.

UNIT-3	**DETECTION & RECOGNITION**	**10 hours**

Object Detection Fundamentals: Bounding Box and Localization, IoU; Evaluation metrics: Precision, Recall, mAP; Real-Time Detection: YOLO, SSD; Advanced Object Detection Models: YOLOv8, RetinaNet; Object Tracking: Tracking by Detection, Algorithm: SORT, Deep SORT

UNIT-4	**RECURRENT NEURAL NETWORKS**	**10 hours**

Sequential Data, Sequence models, Recurrent Neural Network Model, Notation, Back-propagation through time (BTT), Different types of RNNs, Language model and sequence generation, Sampling novel sequences, Vanishing gradients with RNNs, Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM), Bidirectional RNN, Deep RNNs

UNIT-5	AUTO ENCODERS IN DEEP LEARNING	08 hours
Definition and Motivation, Architecture of Autoencoder, Loss Functions in Autoencoders, Type of Autoencoder, Variational Autoencoder, Stacked auto-encoders, Sequence Autoencoder Application of Autoencoder.		
	Total Lecture Hours	**45 hours**

Textbook:	
Book Details	**Authors Name**
Deep Learning: An MIT Press Book	Ian Goodfellow and Yoshua Bengio Aaron Courville.
Neural Networks and Deep Learning, Determination Press,2015.	Michael Nielson
Neural networks: A classroom Approach, Tata McGraw-Hill Education, 2004.	Satish Kumar

Reference Books:	
Book Details:	**Authors Name**
Deep Learning with Python, Manning publications, 2018	Francois Chollet
Advanced Deep Learning with Keras, PACKT Publications, 2018	Rowel Atienza

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01274814254931148859_shared/overview	34h 51m
Video Lectures	YOLOv8 Full Tutorial Playlist | Object Detection, Segmentation & Classification Explained - YouTube:	10h 30m
Web References	https://www.datacamp.com/blog/yolo-object-detection-explained	2h

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Course Code: CCSE0403	Course Name: Theory of Computation	L	T	P	C
Course Offered in: CSE/IT/CSE R/CSE(Twin)/IT(Twin)/CSE(AI)/CSE(AI Twin)/CSE(AIML)/CSE(AIML TWIN)		3	0	0	3

Pre-requisite: Mathematical Foundations , Fundamentals of Computer System

Course Objectives: The primary objective of this course is to provide a foundational understanding of Automata Theory and its role in the language, processing systems, also explores their application in fields like Natural Language Processing (NLP), speech recognition,

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Identify the fundamental concepts of automata theory, formal languages and compiler components.	K2
CO2	Understand the role of finite automata, regular expressions, and grammar rules in language processing.	K2
CO3	Demonstrate context-free grammars, pushdown automata, and syntax-directed translation to construct intermediate code for language processors.	K3
CO4	Analyze various parsing strategies, code translation methods, and intermediate representations in compiler phases.	K4
CO5	Analyze the functioning of Turing machine models and optimization techniques in code generation for performance improvement.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	–	–	–	–	–	–	2	3	2	**2**
CO2	3	3	3	2	1	–	–	–	–	–	2	3	2	**3**
CO3	3	3	3	3	1	–	–	–	–	–	2	3	2	**3**
CO4	3	3	3	3	1	–	–	–	–	–	2	3	2	**3**
CO5	3	3	3	3	1	–	–	–	–	–	2	3	2	**3**

Course Contents / Syllabus

UNIT-1	**INTRODUCTION TO FINITE AUTOMATA AND COMPILER**	**10 hours**

Introduction: Role of Automata and Formal languages, Alphabet, String, Grammar, Language, Chomsky Hierarchy of languages. **Introduction to Compiler:** Translators, Language Processing System, Phases and passes of compilation **Use Case 1 :** Role of Finite Automata in NLP and Speech Recognition.

UNIT-2	**REGULAR EXPRESSION AND TOKENIZATION**	**9 hours**

Regular Expression: Regular Expression, Regular Sets, Properties of Regular Expression, Finite Automata and Regular Expression, Arden's Theorem, Regular Grammars-Right Linear and Left Linear grammars. **Lexical Analyzer**: Role of lexical Analyzer, Specifications and Recognition of tokens, Lex,

UNIT-3	**CONTEXT FREE GRAMMAR AND PUSH DOWN AUTOMATA**	**09 hours**

Context Free Grammar (CFG): Definition and Language, Derivations, Parse Trees and Ambiguity, Simplification of CFG, **Push Down Automata:** Definition of the Pushdown Automata, Languages of PDA

UNIT-4	**PARSER AND INTERMEDIATE REPRESENTATION**	**10 hours**

Parser: Role of parser, Top down Parsing-LL (1) parser, Bottom up parsing- shift reduce parser and LR (0), SLR parser, Introduction to Syntax directed Translation, **Intermediate-Code Generation:** Three-Address Code- Quadruples, Triples, Indirect triples **Use Case-2**:Role of CFG and parsing in Voice Assistant		
UNIT-5	**TURING MACHINE AND OPTIMIZATION**	**10 hours**
Turing Machine: Basic Concept of Turing Machine, Variants of Turing Machine, Universal Turing Machine **Code optimization and generation:** Basic Block, Flow graph, DAG, Optimization Techniques		
Total Lecture Hours		**48 hours**

Textbook:

S.No	Book Title	Author
1	Theory of Computer Science-Automata, Languages and Computation	K.L.P. Mishra, and N. Chandrasekharan ,
2	Compilers: Principles, Techniques and Tools	Alfred V. Aho, Jeffrey D Ullman

Reference Books:

S.No	Book Title	Author
1	Introduction to languages and the theory of computation	J Martin
2	Compiler Design in C	Allen I. Holub

Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://archive.nptel.ac.in/courses/106/106/106106049/	18 hours
	https://archive.nptel.ac.in/courses/106/108/106108113/	
	https://archive.nptel.ac.in/courses/106/108/106108113/	
	https://archive.nptel.ac.in/courses/106/106/106106049/	
	https://archive.nptel.ac.in/courses/106/106/106106049/	
	https://archive.nptel.ac.in/courses/106/108/106108113/	
Video Lectures	https://www.youtube.com/watch?v=539Bk9fFOyo	42 hours
	https://www.youtube.com/watch?v=6b40kKe2SFg	
	https://www.youtube.com/watch?v=1qOMlqE6LhU	
	https://www.youtube.com/watch?v=1qOMlqE6LhU	
	https://www.youtube.com/watch?v=BR6fHjKFqa0	
Web References	Last Minute Notes - Theory of Computation - GeeksforGeeks	-

Course Code: CCSCY0411	Course Name: Fundamental Of CYBER SECURITY	L	T	P	C
Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AIML)/CSE(IOT)/CSE(AI)/CSE(DS)/CSE-R/M.Tech Int.		3	0	0	3

Pre-requisite: Basic knowledge of Computer Systems, Familiarity with Internet Usage and Web Browsing, Understanding of basic Networking Terms (e.g., IP, LAN, Router), Awareness of Digital Tools (Email, Cloud, Antivirus), Fundamental Logical Thinking.

Course Objectives:

Basic knowledge of Computer Systems, Familiarity with Internet Usage and Web Browsing, Understanding of basic Networking Terms (e.g., IP, LAN, Router), Awareness of Digital Tools (Email, Cloud, Antivirus), Fundamental Logical Thinking.

Course Outcome		Bloom's Knowledge Level (KL)
CO1	Understand the basic principles and terminology of cyber security.	K1
CO2	Recognize common cyber threats and attack vectors.	K2
CO3	Demonstrate knowledge of basic cyber defense tools and techniques.	K2
CO4	Adopt safe online behavior and promote cyber hygiene.	K2
CO5	Interpret and apply basic cyber laws and ethical principles.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	-	1	-	1	-	2	-	3	1
CO2	3	3	2	2	2	-	2	-	2	-	2	2	3	2
CO3	3	3	3	2	3	-	3	2	2	1	3	3	3	3
CO4	2	2	2	1	2	1	3	2	2	-	3	-	3	3
CO5	2	2	2	2	1	1	3	2	2	1	3	-	2	3

Course Contents / Syllabus

UNIT 1	INTRODUCTION TO CYBER SECURITY	9 hours

Definition, Evolution, and Need of Cyber Security, Difference between Information Security and Cyber Security, Cyber Forensics, The CIA Triad (Confidentiality, Integrity and Availability), **Basic Terminologies:** Threats, Vulnerabilities, Exploits, Risks, Cyber Security Objectives: Prevention, Detection, Response and Recovery, **Cyber Security Domains:** Network Security, Information Security,

Application Security, Cloud Security and IoT Security, Security Goals, Roles of Security Policies, Procedures, and Awareness.

UNIT 2	**CYBER THREATS AND ATTACKS**	**9 hours**

Introduction to cyber-attacks, **Malware Types:** Virus, Worm, Trojan Horse, Ransomware, Spyware, Adware, **Social Engineering Attacks:** Phishing, Baiting, Pretexting, Tailgating, Web-Based Attacks: SQL Injection, Cross-Site Scripting (XSS), Clickjacking, **Network Attacks:** Denial-of- Service (DoS), DDoS, Spoofing, Sniffing, Insider threats and APTs (Advanced Persistent Threats), Emerging Threats: IoT Vulnerabilities, Mobile Threats. Application security (design, development and testing), operations security,

monitoring, identifying threats, and remediating them.

UNIT 3	**CYBER DEFENSE MECHANISM**	**9 hours**

Authentication Mechanisms: Passwords, OTPs, Biometrics, Access Control Models: DAC, MAC, RBAC, Firewalls: Types, Configurations, Limitations, Intrusion Detection and Prevention Systems (IDS/IPS), **Cryptography:** Basic Idea of Encryption and Decryption, **Endpoint Protection:** Antivirus, Anti-Malware, Backup Types: Full, Incremental, Differential, Incident Response Basics.

UNIT 4	**NETWORK AND SYSTEM SECURITY BASICS**	**9 hours**

Networking basics (home network and large-scale business networks), Networking concepts (CIDR, subnets), and protocols (DNS, DHCP, IP). Basic Network Security Concepts: Security of protocols, IP, MAC, Ports, Protocols (HTTP, HTTPS, FTP), Working of World Wide Web (www), the Internet and the HTTP protocol, security of browser to web server interaction**, Network Security Devices:** Routers, Switches, Firewalls, Proxies, **Secure System Configuration:** OS Hardening, User Privileges, Patch Management and Software

Updates, Secure Coding Principles and Common Software Flaws, Safe Browsing Habits, Secure Downloads, Email Security.

UNIT 5	**CYBER ETHICS LAW AND DATA HYGIENE**	**9 hours**

Cyber Ethics: Responsible use of Technology, Netiquette, Importance of Cyber Laws and Data Protection Regulations (GDPR), Overview of Indian IT Act 2000 and Amendments (Sections 43, 66, 67, etc.), Data Privacy and Data Protection Principles, Intellectual Property Rights (IPR) in Cyberspace, Digital Footprint, Identity Protection, Password Hygiene, Cyber Bullying, Trolling, Online Frauds,

Fake News Awareness.

Total: 45 Hours

TEXTBOOK

BOOK TITLE	**AUTHOR**
Cybersecurity: Principles and Practice *(5th Edition)*	*William Stallings*

Computer Security Fundamentals *(4th Edition)*	*Chuck Easttom*

REFRENCE BOOKS:

BOOK TITLE	AUTHOR
Fundamentals of Cyber Security (1st Edition)	Kutub Thakur, Al-Sakib Khan Pathan
Cyber Security (1st Edition)	Nina Godbole, Sunit Belapure

SELF-STUDY CONTENT LINKS:		DURATION (IN HRS)
MOOCS	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013638 2610228674561182_shared/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013 0752807862845445_shared/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_338 8902307073574000_shared/overview https://infyspringboard.onwingspan.com/web/en/page/lex_auth_ 014234794236551168189	**1** **h** **11h** **5m** **7h** **3m** **1h**
VIDEO LECTURES	https://www.youtube.com/watch?v=z5nc9MDbvkw https://nptel.ac.in/courses/106106129 https://www.youtube.com/watch?v=BdluJhRaAMA https://www.youtube.com/watch?v=uqvn2vRBOqk&list=PLm3g 0kXO2llm8Qd6EpkQc2gUiCz-bf3f2 https://nptel.ac.in/courses/106105183	**38 Min 19 Sec** **8 Min 15 Sec** **12 Min 34 Sec** **4 Min** **12 Min**
WEB REFERENCES	https://www.geeksforgeeks.org/cybersecurity/what-is-cyber-security/ https://www.geeksforgeeks.org/types-of-cyber-attacks/ https://www.ironnet.com/topics/what-is-cyber-defense https://www.geeksforgeeks.org/computer-networks/types-of-security- mechanism/	**24 Hrs**

Course Code: CCSE0411	Course Name: PYTHON WEB DEVELOPMENT WITH DJANGO	L	T	P	C
Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AIML)/CSE(IOT)/CSE(AI)/CSE(DS)/CSE-R/M.Tech int		3	0	0	3

Pre-requisite: Students should have good knowledge of Python Programming and Python coding experience.

Course Objectives:

This course focuses on how to design and build statistics as well as dynamic webpages and interactive web-based applications. These courses mainly focus on how Python operates within web development using the increasingly popular Django framework.

Course Outcome		Bloom's Knowledge Level (KL)
CO1	Apply Python programming concepts and client-side technologies to develop Django-based web applications.	K3
CO2	Demonstrate the use of the Django web framework to design and implement dynamic web pages and interactive web-based applications.	K3
CO3	Analyze and implement account management and authentication mechanisms in Django applications.	K4
CO4	Analyze the role of database connectivity using SQLite in web application development and its impact on modern e-commerce and social platforms.	K4
CO5	Design, develop, and deploy a functional Django web application on a cloud platform by integrating web technologies, databases, and authentication services.	K6

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	3	1	3	-	1	3	-	3	3	3	3	3
CO2	2	2	3	1	3	-	1	-	-	2	3	3	3	3
CO3	2	2	2	2	2	-	-	2	-	2	2	2	2	2
CO4	2	1	2	1	2	-	-	1	1	2	2	3	3	3

CO5	2	1	3	2	3	-	-	3	2	3	3	2	2	2

Course Contents / Syllabus

UNIT 1	**PYTHON LIBRARIES FOR WEB DEVELOPMENT**	**9 hours**

Collections-Container datatypes, Tkinter-GUI applications, Requests-HTTP requests, BeautifulSoup4-web scraping, Scrapy, Zappa, Dash, CherryPy, Turbo Gears, Flask, Web2Py, Bottle, Falcon, Cubic Web, Quixote, Pyramid.

UNIT 2	**INTRODUCTION TO DJANGO FRAMEWORK**	**9 hours**

Understanding Django environment, Features of Django and Django architecture, MVC and MTV, URL's and Views, Mapping the views to URLs, Django Template, Template inheritance Django Models, Creating model for site, Converting the model into a table, Fields in

Models, Integrating Bootstrap into Django, creating tables, Creating grids, Creating carousels.

UNIT 3	**INTEGRATING ACCOUNTS & AUTHENTICATION ON DJANGO**	**9 hours**

Introduction to Django Authentication System, Security Problem & Solution with Django Creating Registration Form using Django, Adding Email Field in Forms, configuring email settings, Sending emails with Django, Adding Grid Layout On Registration Page, Adding Page Restrictions, Login Functionality Test and Logout.

UNIT 4	**CONNECTING SQLITE WITH DJANGO**	**9 hours**

Database Migrations, Fetch Data From Database, Displaying Data On Templates, Adding Condition On Data, Sending data from url to view, Sending data from view to template, Saving objects into database, Sorting objects, Filtering objects, Deleting objects, Difference

between session and cookie, Creating sessions and cookies in Django.

UNIT 5	**DEPLOYING DJANGO WEB APPLICATION ON CLOUD**	**9 hours**

Creating a functional website in Django, Four Important Pillars to Deploy, registering on Heroku and GitHub, Push project from Local System to GitHub, Working with Django Heroku, Working with Static Root, Handling WSGI with gunicorn, Setting up Database

& adding users.

Total: 45 Hours

TEXTBOOK

BOOK TITLE	**AUTHOR**
"Python: The Complete Reference Paperback", 4th Edition 2018, *McGraw Hill Education Publication.*	*Martin C. Brown*

"Python Programming: Using Problem Solving Approach", 3rd Edition 2017, Oxford University Press Publication.	*Reema Thareja*
" Beginning Django Web Application Development and Deployment with Python", 2nd Edition 2017, Apress Publication.	*Daniel Rubio, Apress*
"Python Django Web Development: The Ultimate Django web framework guide for Beginners", 2nd Edition 2019, Kindl e Edition.	*William Jordon,*

REFRENCE BOOKS:

BOOK TITLE	**AUTHOR**
"Building Django 2.0 Web Applications: Create enterprise-grade, scalable Python web applications easily with Django 2.0", 2nd Edition 2018, and Packt Publishing.	*Tom Aratyn,*
"Build a website with Django", 1st Edition 2019, GNW Independent Publishing Edition.	*Nigel George*
" Django in 8 Hours: For Beginners, Learn Coding Fast! 2nd Edition 2020, independently published Edition.	*Ray Yao*
"Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium, and JavaScript", 2nd Edition 2019, Kindle Edition.	*Harry Percival*

SELF-STUDY CONTENT LINKS:		**DURATION (IN HRS)**
MOOCS	https://online.codingblocks.com/courses/python-django	**4 WEEKS**
VIDEO LECTURES	https://youtu.be/QXeEoD0pB3E?list=PLsyeobzWxl7poL9JTVyndKe62ieoN-MZ3 https://youtu.be/9MmC_uGjBsM?list=PL3pGy4HtqwD02GVgM96-V0sq4_DSinqvf	**1 HRS 03 MIN** **27 MIN 29 SEC**
WEB REFERENCES	https://www.geeksforgeeks.org/python/python-web-development-django/	

Course Code: CCSIOT0411	**Course Name: Sensors and Instrumentations**	**L**	**T**	**P**	**C**
Course Offered in: CSE(IoT)		**3**	**0**	**0**	**3**

Pre-requisite: Basics of Electronics

Course Objectives:

- To provide understanding of the fundamentals, classification, and operating principles of sensors and transducers used for measurement systems.
- To familiarize students with techniques for measurement of displacement, force, pressure, temperature, position, flow, and level using different sensors.
- To introduce the concepts of virtual instrumentation, graphical programming, and software-based instrumentation for industrial automation.
- To provide knowledge of data acquisition methods, analog and digital interfacing, ADC/DAC techniques, and networked communication.
- To acquaint students with smart sensors, their characteristics, selection criteria, and applications in automation and control systems.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Explain the principles, classification, characteristics, and applications of sensors and transducers used for displacement, force, pressure, and smart devices.	K2
CO2	Analyze suitable sensors for measuring physical parameters, such as temperature, position, flow, level, and vibration, in engineering applications.	K4
CO3	Examine virtual instrumentation applications using graphical programming concepts, loops, arrays, and clusters, as well as software-based industrial automation tools.	K4
CO4	Illustrate data acquisition methods, including Analog/Digital I/O, ADC, DAC, counters, timers, and networked communication techniques.	K3
CO5	Evaluate intelligent and smart sensors for industrial, robotics, and automobile control applications and select suitable sensors for practical systems.	K5

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	2	-	-	-	-	-	-	1	3	1
CO2	3	3	2	2	2	-	-	-	-	-	-	1	3	2
CO3	2	2	3	2	3	-	-	-	1	1	-	2	2	3
CO4	3	2	2	2	3	-	-	-	-	-	-	1	2	3
CO5	3	3	3	2	2	1	1	-	1	-	-	2	3	3

Course Contents / Syllabus

Unit 1	**Sensors & Transducers**	**07 hours**

Sensors & Transducers: Definition, Classification of sensors, Measurement of displacement using Potentiometer, LVDT & Optical Encoder, Measurement of force using strain gauge, Measurement of pressure using LVDT based diaphragm & piezoelectric sensor, Introduction to sensors most widely used in computer and mobile phones: Temperature, Pressure, Humidity, Toxic Gas, pH sensor, sound sensor, magnetic field sensor, Motion (Infra-red) sensor, barometer, and Gyroscope.

Unit 2	**Measurement of physical parameters**	**08 hours**

Measurement of temperature using Thermistors, Thermocouple & RTD, Concept of thermal imaging, Measurement of position using Hall effect sensors, Proximity sensors: Inductive & Capacitive, Use of proximity sensor as accelerometer and vibration sensor, Working Principles of Flow Sensors: Ultrasonic
& Laser, Level Sensors: Ultrasonic & Capacitive.

Unit 3	**Virtual Instrumentation**	**10 hours**

Virtual Instrumentation: Graphical programming techniques, Data types, Advantage of Virtual Instrumentation techniques, Concept of WHILE & FOR loops, Arrays, Clusters & graphs, Structures: Case, Sequence & Formula nodes, Need of software based instruments for industrial automation.

Unit 4	**Data Acquisition Methods**	**12 hours**

Data Acquisition Methods: Basic block diagram, Analog and Digital IO, Counters, Timers, Types of ADC: successive approximation and sigma-delta, Types of DAC: Weighted Resistor and R-2RLadder type, Use of Data Sockets for Networked Communication.

Unit 5	**Advanced Sensors**	**08 hours**

Intelligent Sensors: General Structure of smart sensors & its components, Characteristic of smart sensors: Self calibration, Self-testing & self-communicating, Selection of Sensors for Practical Applications, Application of smart sensors: Automatic robot control & automobile engine control.

	Total Lecture Hours	**45 hours**

Textbook:

1. DVS Murthy, Transducers and Instrumentation, PHI 2nd Edition 2013
2. D Patrana bis, Sensors and Transducers, PHI 2nd Edition 2013.

Reference Books:

1. Arun K. Ghosh, Introduction to measurements and Instrumentation, PHI, 4th Edition 2012.
2. Hermann K.P. Neubert, "Instrument Transducers" 2nd Edition 2012, Oxford University Press.

Self-Study Content Links:		Duration (in Hrs)
MOOCs	**Sensors and Actuators** https://onlinecourses.nptel.ac.in/e-learning/preview/noc24_ee45	30 hours
Video Lectures	https://www.youtube.com/watch?v=9S6-fXxZruY&list=PL5vhyNmOQ5OSOSy63jf7tStFM2p78aUS4	08 hours
Web References	https://www.ni.com/en.html	08 hours

Course Code: CCSAI0412	Course Name: CRM- FUNDAMENTALS	L	T	P	C
Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AIML)/CSE(IOT)/CSE(AI)/CSE(DS)/CSE-R/M.Tech Int.		3	0	0	3

Pre-requisite: None

Course Objectives:

To provide an understanding of CRM fundamentals, customer relationship strategies, e-CRM technologies, and the organizational, technological, and human aspects of CRM implementation.

Course Outcome		Bloom's Knowledge Level (KL)
CO1	Understand the fundamental concepts of Customer Relationship Management (CRM).	K1, K2
CO2	Explain the strategies and framework of Customer Relationship Management.	K2
CO3	Describe the fundamentals of Cloud-Based Customer Relationship Management.	K1
CO4	Apply CRM concepts to business use cases and customer-centric solutions.	K3
CO5	Demonstrate the basics of Customer Relationship Management implementation.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	–	–	2	–	–	–	–	–	1	1	2	1
CO2	2	3	1	–	2	1	–	–	–	–	2	2	2	2
CO3	2	2	2	1	3	–	–	–	–	–	2	3	2	3
CO4	2	3	3	2	2	2	–	1	2	1	2	2	3	2
CO5	2	3	3	2	3	2	–	2	2	2	3	3	2	3

Course Contents / Syllabus

UNIT 1	INTRODUCTION	9 hours

CRM- definition, history, goals. Sources of CRM value. Components of CRM: people, process, technology. Evolution of CRM: marketing and its principles, customer relations to CRM. Dynamics of Customer Supplier Relationships, Nature and context of CRM, Strategy and Organization of CRM: strategy, The relationship-oriented organization: Mission, Culture, Structure, People, Communication & Information Systems.

UNIT 2	CRM STRATEGY AND FRAMEWORK	9 hours

Developing a CRM strategy. Customer oriented (C in CRM), Relationship driven, 360-degree view of customer. CRM system features-functions, application, benefits and solutions. Importance of loyalty- active, passive, split, shifting and switchers, customer profiling, customer segmentation model, Customer Experience, relationship marketing and journey, Case study.		
UNIT 3	**SOLUTION DESIGN AND ARCHITECTURE**	**9 hours**
CRM system solution-specifications. Data Analysis, Solution Requirements. Types of CRM- On-Premises, cloud based. Pros and Cons of each. Integration CRM with other enterprise applications. The Technology of CRM: Data warehouses and customer relationships, creating data mart model, components of operational data warehouse.		
UNIT 4	**CRM FOR BUSINESS**	**9 hours**
CRM in Sales, Service, Marketing, E-commerce. Social Customer Relationship Management. Analytical CRM: Predictive Analytics Vs Operational Analytics. Channel Partner Relationship management, Collaborative CRM (using data pooling), Business Benefits of Cloud Based System, SLAs, Practical Challenges.		
UNIT 5	**CRM IMPLEMENTATION**	**9 hours**
Building CRM roadmaps: current processes, customers, strategic goals, technology issues, pilot and proof of concept projects. Preliminary Roadmap and its template, developing roadmap midstream. Design stage, custom development, integration, reporting, data migration, and implementation, testing, launching and application management. Introduction to following CRM tools: ZOHO, Pega, Microsoft Dynamics 365, Sales force.		
		Total: 45 Hours

TEXTBOOK

BOOK TITLE	AUTHOR
CRM Fundamentals	*Scott Kostojohn, Mathew Johnson, Brian Paulen*
Customer Relationship Management: How to Develop and Execute a CRM Strategy	*Michael Pearce*

REFRENCE BOOKS:

BOOK TITLE	AUTHOR
The CRM Handbook: A Business Guide to Customer Relationship Management	Jill Dyché
Customer Relationship Management Systems Handbook	Duane E. Sharp

SELF-STUDY CONTENT LINKS:		DURATION (IN HRS)
VIDEO LECTURES	https://onlinecourses.nptel.ac.in/e-learning/preview/noc20_mg57	**7 min**
WEB REFERENCES	https://www.geeksforgeeks.org/software-engineering/customer-relationship-management-crm/	

LAB Course Code: CCSML0452	LAB Course Name: Deep Learning Lab	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)		0	0	2	1

Pre-requisite: Python Programming, Machine Learning

Course Objectives:

The course aims to develop practical skills in designing, training, and evaluating deep learning models using ANN, CNN, RNN, Autoencoders, and object detection techniques for solving real-world intelligent system and pattern recognition problems.

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

CO1	Apply ANN models, learning rules, optimization, and regularization techniques for intelligent systems.	K3
CO2	Develop and evaluate CNN models for image classification and object detection tasks.	K4
CO3	Implement RNNs and Autoencoders for sequence learning, prediction, and feature representation.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	1	2	2	–	–	–	–	1	1	3	1	**2**
CO2	2	3	1	3	1	–	–	–	1	2	1	3	3	**3**
CO3	2	3	2	3	3	–	–	–	2	2	2	2	1	**3**

List Of Practical's (Indicative & Not Limited To)		CO
S.No.	**Experiment (Case Study Based Question)**	**CO Mapping**
1	An electronics company wants to design a system to predict whether a circuit will turn ON or OFF. Implement an artificial neuron model using Python to simulate this binary decision-making system.	CO1
2	A biomedical system needs nonlinear transformations for classification. Implement and compare activation functions: Sigmoid, ReLU, and Tanh.	CO1
3	A digital automation system must perform logical operations. Design a Perceptron model to simulate AND and OR gates.	CO1
4	A training system must ensure learning convergence. Implement the Perceptron learning rule and verify convergence.	CO1
5	A neural system adapts based on signal correlation. Implement Hebbian Learning rule.	CO1
6	An AI model requires optimization to minimize error. Implement Gradient Descent and Delta Rule.	CO1
7	A company wants to classify customer data. Implement a Multilayer Perceptron (MLP).	CO1
8	A prediction system needs error correction. Implement Backpropagation algorithm.	CO1
9	A model suffers from overfitting. Apply L1 and L2 Regularization techniques.	CO1

10	A model must generalize better. Implement Dropout and Early Stopping techniques.	CO1
11	A surveillance system processes images. Perform image preprocessing using OpenCV.	CO2
12	A digit recognition system is required. Train a CNN model using MNIST dataset.	CO2
13	An engineer wants to understand convolution. Implement convolution operation manually.	CO2
14	An image system must preserve edges. Implement padding and stride operations.	CO2
15	A CNN must reduce dimensions efficiently. Implement Max and Average pooling.	CO2
16	A startup wants to classify product images. Design a CNN for a custom dataset.	CO2
17	A research team wants to visualize CNN learning. Visualize feature maps.	CO2
18	A company wants faster deployment. Implement transfer learning using VGG/ResNet.	CO2
19	A model underperforms. Perform hyperparameter tuning in CNN.	CO2
20	A smart city system needs real-time detection. Implement object detection using YOLO.	CO2
21	A model must be evaluated. Calculate IoU, Precision, Recall.	CO2
22	A traffic system needs detection. Implement SSD object detection.	CO2
23	A surveillance system tracks objects. Implement SORT tracking algorithm.	CO2
24	A weather system uses time-series data. Implement basic RNN.	CO3
25	A sequence model needs temporal learning. Implement BPTT.	CO3
26	A chatbot needs sequence prediction. Implement LSTM.	CO3
27	A stock system needs efficient modeling. Implement GRU.	CO3
28	A compression system needs feature reduction. Implement Autoencoder.	CO3
29	A noisy system needs reconstruction. Implement Denoising Autoencoder.	CO3
30	A probabilistic system is needed. Implement Variational Autoencoder.	CO3
31	A deep feature system is required. Implement Stacked Autoencoder.	CO3
32	A sequence compression system is needed. Implement Sequence Autoencoder.	CO3

Course Code: CASL0401	Course Name: Technical Communication	L	T	P	C
Course Offered in: B. Tech. All branches (except CSBS)		2	0	0	2

Pre-requisite: Intermediate level (CEFR) and above

Course Objectives:

1. **Demonstrate effective verbal and non-verbal communication skills** in diverse professional settings, including meetings, presentations, and interpersonal interactions.
2. **Develop and apply clear, concise, and audience-appropriate written communication**, such as emails, letters, memos, resume', using correct grammar, tone, and format.
3. **Adapt communication style based on cultural, organizational, and situational contexts** to foster inclusive and respectful professional relationships.
4. **Employ digital communication tools and platforms** (e.g., video conferencing, business messaging apps) responsibly and effectively in remote or hybrid work environments.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Comprehend the principles and functions of technical communication.	K2
CO2	Write for specific audience and purpose to fulfil the provided brief	K3
CO3	Recognize and produce different kinds of technical documents.	K3
CO4	Apply effective speaking skills to efficiently carry out official discourses.	K3
CO5	Demonstrate their understanding of communication through digital media.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1	1	1	2	1	2	3	1	1	–	–	**1**
CO2	1	1	1	1	1	1	1	2	3	1	1	1	–	**2**
CO3	1	1	1	1	1	1	1	2	3	1	1	2	1	**2**
CO4	1	1	1	1	1	1	1	2	3	1	1	1	1	**2**
CO5	1	1	1	1	1	1	1	2	3	1	1	1	1	**3**

Course Contents / Syllabus

UNIT-1	INTRODUCTION TO TECHNICAL COMMUNICATION	4 Hours

Technical Communication: Definition, Process, Types, Levels, and Flow; **Barriers to Technical Communication:** emphasis on gender neutral language and cultural sensitivity;

Significance of audience in technical communication

UNIT-2	TECHNICAL WRITING 1	5 Hours

Technical writing skill: characteristics, examples; **Business letters/emails**: Content organization, Tone and intent; **Agenda & Minutes of Meetings**

UNIT-3	TECHNICAL WRITING 2	5 Hours

Job application, **Resume'**; **Report, proposal**; **Technical paper**: Abstract; **Ethical Writing:** Copy Editing, Referencing and Plagiarism

UNIT-4	PUBLIC SPEAKING	6 Hours

Components of effective speaking: Simplicity, order, balance in arranging ideas. Importance of *KOPPACT;* **Appearing for a job interview:** FAQs; **Telephonic & Online**

Interviews

UNIT-5	VIRTUAL/REMOTE COMMUNICATION	4 Hours

Remote work: online platforms; **Video conferencing**; **Virtual etiquette**: email ids, usernames; **Writing Blogs & creating Vlogs**

Textbook:		
SNO	**Books Title**	**Author**
1	Technical Communication – Principles and Practices, 4th Edition , Oxford Univ. Press, 2022, New Delhi.	Meenakshi Raman & Sangeeta Sharma
Reference Books:		
S.NO	**Books Title**	**Author**
1	Technical Communication, 15th Edition , Pearson, 2021	John M. Lannon & Laura J. Gurak
2	Spoken English- A Manual of Speech and Phonetics (5th Edition) by , 2024, New Delhi.	R K Bansal & J B Harrison, Orient Blackswan
3	Business Correspondence and Report Writing (6 Edition), Tata McGraw Hill & Co. Ltd., 2020, New Delhi.	Prof. R C Sharma, Krishna Mohan, and Virendra Singh Nirban
4	Intercultural Communication in Virtual Exchange , Cambridge Univ. Press, 2024.	Francesca Helm
NPTEL/ You tube/ Faculty Video Link:		
Unit 1	https://onlinecourses.nptel.ac.in/noc24_ge37/preview	
Unit 2	https://archive.nptel.ac.in/courses/109/106/109106094/	
Unit 3	https://www.youtube.com/watch?v=kOJlwMJxEG0&t=8s	
Unit 4	https://www.youtube.com/watch?v=Sg7Q_dC_fWU&list=PLPuC5CMHiqmuzq_KQ4aw0V9Q7xJY6aezb	
Unit 5	https://www.youtube.com/watch?v=ymLFJDpjgCk&list=PLPuC5CMHiqmuzq_KQ4aw0V9Q7xJY6aezb&index=6	

LAB Course Code: CCSE0451	LAB Course Name: Database Management Systems Lab	L	T	P	C
Course Offered in: CSE/CSE-R/IT/M.Tech Int./CSE (Twin)/IT (Twin)/CSE(Prof)/IT (Prof)/M&C/AI/AI (TWIN)/AIML/AIML (TWIN)/CS/CYS/DS/IOT		0	0	4	2

Pre-requisite: Basic knowledge of computer fundamentals, programming, data structures, relational database concepts.

Course Objectives: To familiarize the students to the basics of database design and implementation.

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

CO1	Design and implement ER/EER models to solve real-world problems and transform them into relational schemas using appropriate database tools.	K3
CO2	Apply SQL and PL/SQL to create complex data queries, and procedural operations including triggers and functions, along with database connectivity.	K3
CO3	Analyze and ensure database integrity using constraints, and implement unstructured databases using MongoDB with appropriate query operations.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO–PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3	2	1	–	1	2	1	2	3	1	**2**
CO2	3	3	3	3	2	2	–	2	1	2	2	3	3	**3**
CO3	2	2	2	2	3	2	–	–	1	2	2	3	3	**3**

List Of Practical's (Indicative & Not Limited To)

S.No	Experiment	CO Mapping
1	Understand and implement the different ER diagram notation with their relationship and cardinalities.	CO1
2	Choose a case study on different domains: I. E-commerce Platform II. Inventory Management III. Railway System IV. Hospital Data Management V. Company System VI. Banking System	CO1
3	Create an ER diagram for the chosen case study. Identify suitable entities, relationships, cardinalities and attributes.	CO1
4	Implement DDL, DML, DCL & TCL commands.	CO2
5	Implementation of I/O Constraints: Primary Key, Composite Primary Key, Foreign Key, Unique Key.	CO2
6	Implementation of Business Constraints: Null, Not Null, Default, Check.	CO2
7	Practice queries using LIKE, BETWEEN, Aliases, DISTINCT, Predicate and Aggregate functions.	CO2
8	Implementation of queries using WHERE, GROUP BY, HAVING and ORDER BY.	CO2
9	Create EMPLOYEE table and write SQL queries (i–xiv): Manager salary list, salary comparisons, designations, experience, clerk/analyst, join dates, names starting with S, total salary/job, lowest paid under manager, employee count/department, salary sorting, salary >16000, CHECK(empno>100), UNIQUE(DEPTNO), PRIMARY KEY(EMPNO).	CO2
10	Implementation of queries using set theory operators: UNION, INTERSECT, MINUS.	CO2
11	Implementation of Inner Join and Outer Join with types.	CO2
12	Implementation of Nested/Sub Queries: IN, NOT IN, EXISTS, NOT EXISTS, ALL, ANY.	CO2
13	Apply set theory operators, joins and nested queries on the chosen case study.	CO2
14	Understand and implement Database Connectivity with any programming language.	CO2

15	Implementation of Indexing, Views and Sequence.	CO2
16	PL/SQL Programs: (I) Add Two Numbers (II) Fibonacci Series (III) Greatest of Three Numbers.	CO2
17	PL/SQL block to calculate area of a circle (radius 3–7) and store Radius and Area in Areas table.	CO2
18	PL/SQL block to accept account number, check minimum balance and deduct Rs.100 if applicable.	CO2
19	Create row-level trigger on CUSTOMERS table for INSERT/UPDATE/DELETE to display salary difference.	CO2
20	Implementation of COMMIT and ROLLBACK using amount transfer example.	CO2
21	Study of Open Source NoSQL Database and installation of MongoDB.	CO3
22	Implementation of MongoDB Shell commands.	CO3
23	Implementation of CRUD Operations in MongoDB.	CO3
24	Implementation of Aggregate Functions in MongoDB.	CO3

LAB Course Code: CCSML0452	Course Name: Deep Learning Lab	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)		0	0	2	1

Pre-requisite: Python Programming, Machine Learning

Course Objectives:

The course aims to develop practical skills in designing, training, and evaluating deep learning models using ANN, CNN, RNN, Autoencoders, and object detection techniques for solving real-world intelligent system and pattern recognition problems.

Course Outcome: After completion of the course, the student will be able to		Level
CO1	Apply ANN models, learning rules, optimization, and regularization techniques for intelligent systems.	K3
CO2	Examine CNN models for image classification and object detection tasks.	K4
CO3	Implement RNNs and Autoencoders for sequence learning, prediction, and feature representation.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	1	2	2	–	–	–	–	1	1	3	1	**2**
CO2	2	3	1	3	1	–	–	–	1	2	1	3	3	**3**
CO3	2	3	2	3	3	–	–	–	2	2	2	2	1	**3**

List Of Practical's (Indicative & Not Limited To)

S.No.	Experiment (Case Study Based Question)	CO Mapping
1	An electronics company wants to design a system to predict whether a circuit will turn ON or OFF. Implement an artificial neuron model using Python to simulate this binary decision-making system.	CO1
2	A biomedical system needs nonlinear transformations for classification. Implement and compare activation functions: Sigmoid, ReLU, and Tanh.	CO1
3	A digital automation system must perform logical operations. Design a Perceptron model to simulate AND and OR gates.	CO1
4	A training system must ensure learning convergence. Implement the Perceptron learning rule and verify convergence.	CO1
5	A neural system adapts based on signal correlation. Implement Hebbian Learning rule.	CO1
6	An AI model requires optimization to minimize error. Implement Gradient Descent and Delta Rule.	CO1
7	A company wants to classify customer data. Implement a Multilayer Perceptron (MLP).	CO1
8	A prediction system needs error correction. Implement Backpropagation algorithm.	CO1
9	A model suffers from overfitting. Apply L1 and L2 Regularization techniques.	CO1
10	A model must generalize better. Implement Dropout and Early Stopping techniques.	CO1
11	A surveillance system processes images. Perform image preprocessing using OpenCV.	CO2
12	A digit recognition system is required. Train a CNN model using MNIST dataset.	CO2

13	An engineer wants to understand convolution. Implement convolution operation manually.	CO2
14	An image system must preserve edges. Implement padding and stride operations.	CO2
15	A CNN must reduce dimensions efficiently. Implement Max and Average pooling.	CO2
16	A startup wants to classify product images. Design a CNN for a custom dataset.	CO2
17	A research team wants to visualize CNN learning. Visualize feature maps.	CO2
18	A company wants faster deployment. Implement transfer learning using VGG/ResNet.	CO2
19	A model underperforms. Perform hyperparameter tuning in CNN.	CO2
20	A smart city system needs real-time detection. Implement object detection using YOLO.	CO2
21	A model must be evaluated. Calculate IoU, Precision, Recall.	CO2
22	A traffic system needs detection. Implement SSD object detection.	CO2
23	A surveillance system tracks objects. Implement SORT tracking algorithm.	CO2
24	A weather system uses time-series data. Implement basic RNN.	CO3
25	A sequence model needs temporal learning. Implement BPTT.	CO3
26	A chatbot needs sequence prediction. Implement LSTM.	CO3

Course Code: CCSE0455	Course Name: Web Technologies	L	T	P	C
Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AIML)/CSE(IOT)/CSE(AI)/CSE(DS)		0	0	6	3

Pre-requisite:

1. Basic Programming Knowledge
2. Knowledge of any programming language (e.g., C, C++, Python/Java)
3. Familiarity with basic concepts of Internet.

Course Objectives:

This course covers different aspect of web technology such as HTML, CSS, Java Script and provide fundamental concepts of Internet, Web Technology and Web Programming. Students will be able to build a proper responsive website.

Course Outcome		Bloom' Level
CO1	Identify the basic facts and explaining the basic ideas of Web technology and internet.	K1, K2
CO2	Applying and creating various HTML5 semantic elements and application with working on HTML forms for user input.	K3, K6
CO3	Understanding and applying the concepts of Creating Style Sheet CSS3 and bootstrap.	K2, K3
CO4	Analysing and implementing concept of JavaScript and its applications.	K4, K6
CO5	Creating and evaluating dynamic web pages using the concept of PHP.	K5, K6

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO–PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	–	2	–	–	–	2	2	–	3	1	**2**
CO2	3	2	3	2	3	–	–	–	2	1	2	1	2	**3**
CO3	3	2	3	–	3	–	–	–	2	2	2	2	3	**3**
CO4	3	3	3	2	3	–	1	–	2	2	2	3	2	**3**
CO5	3	3	3	2	3	–	–	–	2	2	2	2	1	**3**

Course Contents / Syllabus

Unit 1	Basics of Web Technology & Testing	10 hours

Introduction: Introduction to Web Technology, History of Web and Internet, Connecting to Internet, Introduction to Internet services and tools, Client- Server Computing, Protocols Governing Web, Basic principles involved in developing a web site, Planning process, Types of Websites, Web Standards and W3C recommendations. Web Hosting: Web Hosting Basics, Types of Hosting Packages, Registering domains, Defining Name Servers, Using Control Panel, Creating Emails in Cpanel, Using FTP Client, Maintaining a Website.

Unit 2	Introduction to HTML & XML	14 hours

HTML: What is HTML, DOM- Introduction to Document Object Model, Basic structure of an HTML document, Mark up Tags, Heading-Paragraphs, Line Breaks, Understand the structure of HTML tables. Lists, Working with Hyperlinks, Image Handling, Understanding Frames and their needs, HTML forms for User inputs. New form Elements- date, number, range, email, search and data list, Understanding audio, video and article tags. XML: Introduction, Tree, Syntax, Elements, Attributes, Namespaces, Display, HTTP request, Parser, DOM, XPath, XSLT, XQuerry, XLink, Validator, DTD, Schema, Server.

Unit 3	Concepts of CSS3 & Bootstrap	16 hours

Concept of CSS 3: Creating Style Sheet, CSS Properties , CSS Styling(Background, Text Format, Controlling Fonts) , Working with block elements and objects , Working with Lists and Tables , CSS Id and Class, Box Model(Introduction, Border properties, Padding Properties, Margin properties) CSS Advanced(Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute sector) , CSS Color, Creating page Layout and Site.
Bootstrap: Introduction, Bootstrap grid system, Bootstrap Components.

Unit 4	**JavaScript and ES6**	**16 hours**

JavaScript Essentials: Introduction to Java Script , Javascript Types , Var, Let and Const Keywords, Operators in JS , Conditions Statements , Java Script Loops, JS Popup Boxes , JS Events , JS Arrays, Working with Arrays, JS Objects ,JS Functions , Using Java Script in Real time , Validation of Forms, Arrow functions and default arguments, Template Strings, Strings methods, Callback functions, Object de-structuring, Spread and Rest Operator, Typescript fundamentals, Typescript OOPs- Classes, Interfaces, Constructor etc. Decorator and Spread Operator Difference == & ===, Asynchronous Programming in ES6, Promise Constructor, Promise with Chain, Promise Race.

Unit 5	**JavaScript and ES6**	**16 hours**

Introduction to PHP, Basic Syntax, Variables & Constants, Data Type, Operator & Expressions, Control flow and Decision making statements, Functions, Strings, Arrays. Working with files and directories: Understanding file& directory, Opening and closing, a file, Coping, renaming and deleting a file, working with directories, Creating and deleting folder, File Uploading & Downloading.
Session & Cookies: Introduction to Session Control, Session Functionality What is a Cookie, Setting Cookies with PHP. Using Cookies with Sessions, Deleting Cookies, Registering Session variables, Destroying the variables and Session.

Total Lecture Hours	**72 hours**

Textbook:

S.No.	**Book Details**	**Author**
1	"Web Technology and Design", 1nd Edition 2003, New Age International.	C Xavier
2	"Internet and Web Technologies", 2nd Edition 2017,Mc Graw Hill Education.	Raj Kamal,
3	"Beginning PHP Laravel",2nd Edition 2020, kindle Publication.	Oluwafemi Alofe

Reference Books:

S.No.	**Book Details**	**Author**
1	"Collaborative Web Development" 5th Edition 1999, Addison Wesley Publication.	Burdman, Jessica
2	"Fundamentals of Web Development",3rd Edition 2016	Randy Connolly
3	HTML, DHTML, Java Script, Perl & CGI", 4th Edition 2010 BPB Publication	Ivan Bayross,

NPTEL/ Youtube/ Faculty Video Link:

Unit-1	https://youtu.b e/96xF9phMsWA https://youtu.be/Zopo5C79m2k https://youtu.be/ZliIs7jHi1s https://youtu.be/htbY9-yggB0
Unit-2	https://youtu.be/vHmUVQKXlVo https://youtu.be/qz0aGYrrlhU https://youtu.be/BsDoLVMnmZs https://youtu.be/a8W952NBZUE

Unit-3	https://youtu.be/1Rs2ND1ryYc https://youtu.be/vpAJ0s5S2t0 https://youtu.be/GBOK1-nvdU4 https://youtu.be/Eu7G0jV0ImY
Unit-4	https://youtu.be/-qfEOE4vtxE https://youtu.be/PkZNo7MFNFg https://youtu.be/W6NZfCO5SIk https://youtu.be/DqaTKBU9TZk
Unit-5	https://youtu.be/_GMEqhUyyFM https://youtu.be/ImtZ5yENzgE https://youtu.be/xIApzP4mWyA https://youtu.be/qKR5V9rdht0

	https://youtu.be/qKR5V9rdht0	
WEB REFERENCES	**Web Development Technologies - GeeksforGeeks**	

LIST OF PRACTICAL

S.N.	List Of Practical's (Indicative & Not Limited To)	CO
1	A.Overview and Installation of various code editors.	CO1
2	B. Overview and Installation of various servers	CO1
3	Implementing HTML program that represent in the document as a start tag, which gives the name and attributes	CO2
4	Implementing HTML program that represents a document	CO2
5	Implementing HTML program to display your simple CV	CO2
6	Creating html document that represents document object model	CO2
7	To Create a table to show your class time table.	CO2
8	Apply various colors to suitably distinguish keywords , also apply font styling like italics, underline and two other fonts to words you find appropriate , also use header tags.	CO2
9	Create a webpage with HTML describing your department use paragraph and list tags	CO2
10	Implementing HTML program that for Heading	CO2
11	Implementing program that implement paragraph and line-break	CO2
12	Use tables to provide layout to your HTML page describing your college infrastructure.	CO2
13	Use and tags to provide a layout to the above page instead of a table layout	CO2
14	Create links on the words e.g. —Wi-Fi and —LAN to link them to Wikipedia pages	CO2

15	Insert an image and create a link such that clicking on image takes user to other page	**CO2**
16	Change the background color of the page; At the bottom create a link to take user to the top of the page.	**CO2**
17	Creating HTML program to implement three articles with independent, self-contained content.	**CO2**
18	Creating a XML document that defines the self-descriptive tags	**CO2**
19	Designing XML document that store various book data such as: book category, title, author, year and price	**CO2**
20	To Describe the various types of XML key components	**CO2**
21	Design XML DTD to define the structure and legal element and attribute of XML document	**CO2**
22	To implement internal and external DTD	**CO2**
23	Use frames such that page is divided into 3 frames 20% on left to show contents of pages, 60% in center to show body of page, remaining on right to show remarks.	**CO2**
24	Design a HTML registration form that takes user name, user password and mobile number with submit button control	**CO2**
25	Design a HTML5 document that implement of date, number, range, email, search and data list.	**CO2**
26	Implementation in HTML5 that include native audio and video support without the need for Flash.	**CO2**
27	Create a simple form to submit user input like his name, age, address and favourite subject, movie and singer.	**CO2**
28	Add few form elements such as radio buttons, check boxes and password field. Add a submit button at last.	**CO2**
29	Design a XML Schema that describes the structure of an XML document.	**CO2**
30	Design a XML document that describe the well-formed XML document	**CO2**
31	Design a XML document of CD Catalog through each <CD> element, and displays the values of the <ARTIST> and the <TITLE> elements in an HTML table	**CO2**
32	3. Create a XSL document for and taken xml document by you.	**CO2**
33	Create a XSLT document for and taken xml document by you with all steps	**CO2**
34	Design a web page by applying CSS grouping and dimensions property.	**CO3**
35	Demonstrating the CSS Box model with consists of: borders, padding, margins, and the actual content.	**CO3**
36	Add CSS property assign a style or behavior to an HTML element such as: color, border, margin and font-style.	**CO3**

37	Add To Style Text Elements with Font, Size, and Color in CSS	**CO3**
38	Applying a block element in CSS acquires up the full width available for that content.	**CO3**
39	Design a web page by applying CSS Display and Positioning property.	**CO3**
40	Design a web page by applying CSS Display and Positioning property .	**CO3**
41	Design a web page by applying CSS pseudo classes.	**CO3**
42	Creating a Java Script code to implement all data types.	**CO4**
43	Design a basic structure of Bootstrap Grid system.	**CO4**
44	Design All Bootstrap Components with example.	**CO4**
45	Implementing a program in Java script to implement augmented function.	**CO4**
46	Implementing a program to implement calculator application as real time.	**CO4**
47	Design a HTML form validation using Java Script.	**CO4**
48	Write a program to implement Arrow function with default argument in ES6	**CO4**
49	Implementing a program in ES6 to implement Template string concepts	**CO4**
50	Implementing a program in ES6 to implement all string methods.	**CO4**
51	Creating a Java Script program to implement Dialog, Confirm and Message Popup Boxes.	**CO4**
52	Implementing a Java Script program to implement onClick and onSubmit event	**CO4**
53	Creating a java script code to implement ‘let’ keyword	**CO4**
54	Creating a java script code to implement ‘const’ keyword	**CO4**
55	Implementing a program to implement call back functions in ES6.	**CO4**
56	Implementing a program for de-structuring of an array in ES6	**CO4**
57	Javascript code to implement object and class concepts in Typescript.	**CO4**
58	Write a Typescript program that implement interface and constructor	**CO4**
59	Write a code in typescript that implement decorator and spread operator	**CO4**
60	Create a constant by using define() function with its proper syntax	**CO4**

61	Creating PHP script that return any data types whatever you use.	**CO4**
62	Implementing a code in Java Script to implement Spread and rest operator	**CO4**
63	Javascript code that should compile by Typescript compiler as'tsc'	**CO4**
64	Write a code in typescript that implement Asynchronous Programming concepts.	**CO4**
65	Write a program in Typescript that implement promise constructor	**CO4**
66	Implementing promise and chain concepts in Typescript	**CO4**
67	Write a code in typescript that implement Promise.race() static method.	**CO4**
68	Crating a program that implement control flow and decision making statement.	**CO4**
69	Creating PHP to implements parameterized function	**CO4**
70	Creating program in PHP to store multiple string and concatenate these string and print it.	**CO5**
71	Write a PHP script to create and delete directory structure	**CO5**
72	Program to upload and download a file in PHP	**CO5**
73	Implements single dimension array in PHP	**CO5**
74	Write a PHP code to open and close a file in a proper manner	**CO5**
75	Write a PHP script to copying, renaming and deleting a file.	**CO5**
76	PHP program to create and destroy a session.	**CO5**
77	PHP program to set and delete a cookie.	**CO5**
78	PHP program to manually register the session variable	**CO5**
79	PHP program to manually destroy the session variable	**CO5**
80	PHP program to store the session data on one page and would be available on second page.	**CO5**

LAB Course Code: CCSCC0452	LAB Course Name: Problem Solving Approaches	L	T	P	C
Course Offered in: CSE/IT /CSE-TWIN/IT-TWIN		0	0	2	1

Pre-requisite: Programming Language C/C++ or Java or Python

Course Objectives:

Problem-solving in computer programming involves a structured approach to identifying, analyzing, and resolving coding challenges. The process typically includes thoroughly
understanding the problem, decomposing it into smaller, manageable parts, designing an appropriate algorithm, implementing the solution through code, and performing testing and debugging to ensure correctness and efficiency

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Develop logic-based solutions using control statements, recursion and bit manipulation to solve basic and intermediate computational problems.	K6
CO2	Implement and manipulate arrays and strings using fundamental and advanced searching sorting techniques.	K3
CO3	Analyze and debug code for logical errors and improve the efficiency of the solution using appropriate data structures and algorithmic patterns.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO–PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2	2	1	2	2	–	–	2	3	3	**2**
CO2	3	3	2	2	2	–	2	–	–	–	2	2	2	**2**
CO3	3	3	2	2	3	1	2	2	–	–	3	3	3	**3**

List Of Practical's (Indicative & Not Limited To)

Problem Statements need to be discussed in lab session: Control Statements

1. Secure Password Generator

A company wants to create a secure password generator for their employees. The password must be based on specific numeric properties to enhance its complexity and security. Write a program to validate and generate a secure password according to the following rules:

1. Prime Number Validation:

- The user must input a 3-digit number. The program should first check if the number is a prime number.
- If it is not a prime number, the user should be prompted to enter another number until a valid prime number is provided.

2. Sum of Digits Check:

- Once a valid prime number is entered, calculate the sum of its digits. If the sum of the digits is not divisible by 3, ask the user to enter another prime number until a valid one is found.

3. Armstrong Number Check:

- Check entered prime number is Armstrong or not? If Armstrong are found, prompt the user to enter another prime number and repeat the process.

Password Generation:

Concatenate the 1 if entered prime number is Armstrong otherwise 2 with the sum of the digits of the valid prime number to form the secure password.

Example Scenario:

Sample Input

Enter a 3-digit prime number: 153 Sum of digits of 153 = 9

The sum is divisible by 3. 153 is Armstrong number Sample Output
Secure Password: 19

2. Write a function to input electricity unit charges and calculate total electricity bill according to the given condition: For first 50 units Rs. 0.50/unit
For next 100 units Rs. 0.75/unit For next 100 units Rs. 1.20/unit For unit above 250 Rs. 1.50/unit
An additional surcharge of 20% is added to the bill

3. Write a method to generate a secure code which the sum of all possible palindrome numbers between given two numbers. For Example:

Input: 10, 80

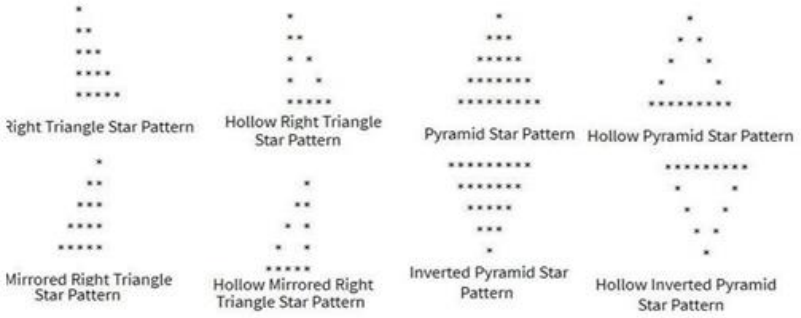
Output: 308

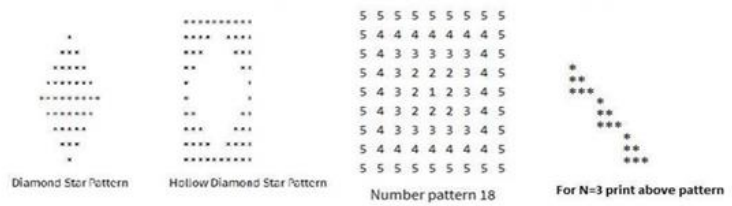
Explanation: All palindrome numbers between 10 & 80 are:

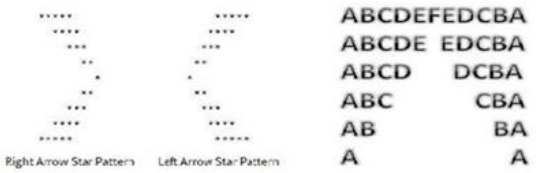
11,22,33,44,55,66,77 Password=

11+22+33+44+55+66+77 = 308

4. Draw **the following Patterns for N=5**







Problem Statements need to be discussed in lab session: Recursive Approach (Basic)

1. Write a program that takes an integer n as input and prints the multiplication table of n from n * 1 to n * 10. The output should clearly show each multiplication step.

2. Write a program to calculate the sum of all integers from 1 to a given number N. The program should take N as input and output the total sum using iteration or recursion.

3. Find the GCD of Two Numbers Using Recursion:
Write a recursive function to calculate the Greatest Common Divisor (GCD) of two numbers using Euclid's algorithm. The function should take two integers as input and return their GCD.

4. Find the LCM of Two Numbers Using Recursion:
Write a program to compute the Least Common Multiple (LCM) of two numbers using recursion. You may use the relationship LCM(a, b) = |a* b| / GCD(a, b) and a recursive function for GCD.

Problem Statements need to be discussed in lab session: Bit Manipulation

1. Write a program to count the number of set bits (1s) in the binary representation of a given integer. The program should efficiently use bitwise operations to perform the task without converting the number to a string.

2. Write a program that takes a number and a bit position as input and checks whether the bit at that position is set (1) or clear (0). Use bitwise operators to perform the check

3. Given a number and a position, write a program to toggle (invert) the bit at the given position using bitwise operations. The result should reflect the updated value of the number after flipping the bit.

4. Write a program to compute the XOR of all numbers from 1 to n using a mathematical pattern (not a loop). Use bitwise XOR properties to achieve an efficient solution.

5. Given an array of size n-1 containing unique elements from 1 to n, find the missing number using bit manipulation (preferably XOR approach) without sorting or using extra space.

6. Given an array where all elements repeat twice except two elements that appear only once, write a program to find the two non-repeating elements using bitwise operations in linear time and constant space.

7. Write a program to check if a given number is a power of two using bit manipulation. A number is a power of two if it has exactly one set bit in its binary representation.

8. Given two integers A and B, write a program to count how many bits need to be flipped to convert A to B. Use XOR to find differing bits and count the number of set bits.

9. Write an efficient program to count the total number of set bits in binary representations of all numbers from 1 to n. Optimize the approach using bitwise logic and recursion.

10. Write a program to calculate the square of a number using only bitwise operations and addition. Do not use multiplication, division, or any power functions.

11. Write a function to add two integers using bitwise operations only. Avoid using the + or - operators. Implement logic using XOR and AND operations for binary addition.

12. Write a program to generate the power set (all subsets) of a given set using bitwise representation. Each subset can be represented by a binary number where each bit indicates inclusion of the corresponding element.

Problem Statements need to be discussed in lab session: Arrays (Try to use sliding window, prefix sum, cadence, recursion, bit manipulation, two pointer approaches)

1. Sarah is assisting the "MathMinds Club" in creating passwords for their online platform. They have a list of numbers, some stable and some unstable. Define a function that can help Sarah calculate the password according to the given scenario.
Scenario:

- There are N numbers provided.
- A number is stable if each digit appears the same number of times.
- A number is unstable if the frequency of its digits is not the same.
- The password is computed as the sum of all stable numbers minus the sum of all unstable numbers.
- Consider only those numbers in the list that have more than equal to three digits.

For example:

Input: N=5 List: 12, 1313, 122, 678, 898

Output: Password: 971

2. Given an array of integers, including possible negative values, you are allowed to modify at most one element by doubling its value. The goal is to find the maximum possible sum of any subarray after making this modification.

Input:
arr = [-2, 1, -3, 4, -1, 2, 1, -5, 4]
Expected Output:

- Original Maximum Subarray Sum: 6 (achieved from [4, -1, 2, 1])
- Maximum Sum After Modification: 10(achieved from [8, -1, 2, 1], where the value 4 is doubled to 8). Given k, find the k-th smallest and k-th largest element in the array. Input: arr = [7, 10, 4, 3, 20, 15], k = 3 Output: Kth Smallest: 7, Kth Largest: 10

7. Sort a binary Input: [0, 2, 1, 2, 0]
Output: [0, 0, 1, 2, 2]
array with values 0, 1, and 2 using constant space and one pass (Dutch National Flag algorithm).

8. Find **longest consecutive subsequence:**
Return the length of the longest
consecutive elements sequence.
Input: [1, 9, 3, 10, 4, 20, 2]
Output: 4 (Sequence: 1, 2, 3, 4)

9. Given a number of bits and a number K. In one flip you can toggle exactly K consecutive bits. With only this flip operation available, convert the string into all 1. Input String: 0000110000 and K=3
Following are four flip
operations by using which all
bits converted into 1's. Flip1-
1110110000 Flip2- 1110110111
Flip3-1111000111 Flip4- 1111111111
If it is not possible to convert all bits into one's then print "IMPOSSIBLE".

10. Given a list of non-negative integers, arrange them in such a way that they form the largest possible number. Since the result can be very large, return it as a string in **O(N log N)** time complexity.

Example-1	**Example-2**
Input: N = 5 Arr[] = {3, 30, 34, 5, 9} **Output**: 9534330	**Input**: N = 4 Arr[] = {54, 546, 548, 60}

11. Given an array arr[] of size n containing distinct integers within the range [1, n+2], find the two missing numbers from the first n+2 natural numbers.

Constraints:

- The solution must run in **O(N) time** and use **O(1) extra space**.
- The array does not contain duplicate values.

Examples:
Input: arr[] = [1, 2, 4, 6, 3, 8], n = 6
Output: 5, 7

12. Given a string str of lowercase alphabets and a number k, the task is to print the minimum value of the string after removal of k characters. The value of a string is defined as the sum of squares of the count of each distinct character present in the string. Return the minimum possible required value. **Examples:**
Input: str = "abccc", k = 1
Output: 6
Input: str = "aabcbcbcabcc", k = 3
Output: 27
Expected Time Complexity: O(n+klog(p))
Note: Here n is the length of string and p is number of distinct alphabets and k number of alphabets to be removed.

13. Given a non-negative integer **S** represented as a string, remove **K** digits from the number so that the new number is the smallest possible.
Note : The given *num* does not contain any leading zero.
Expected Time Complexity: O(|S|).

	Example 1: **Input:** S = "149811", K = 3 **Output:** 111	**Example 2:** **Input**: S = "1002991", K = 3 **Output:** 21

14. You are given a two-dimensional grid board[][] of size n * m consisting of English letters and a string target. Your task is to determine whether the target word can be formed by sequentially connecting letters from the grid. You may move to adjacent cells **horizontally or vertically** (not diagonally), and **a cell may not be reused** once it is part of the current path.

Examples:

Course Code: CNC0401	Course Name: Artificial Intelligence and Cyber Ethics	L	T	P	C
Course Offered in: All Branches		2	0	0	-

Pre-requisite: Basic understanding of AI, Cybercrime, Computer System and Ethics

Course Objectives: The course aims to foster critical thinking about ethical issues, promote responsible use of technology, and ensure students can identify, analyze, and address ethical dilemmas in Artificial Intelligence and cyber domains.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Learn key principles of AI ethics, summarizing ethical considerations and applications in AI development and deployment.	K2
CO2	Apply policies and framework for Fairness in AI and Machine Learning.	K3
CO3	Apply privacy and security concepts, risk management and regulatory compliance in the field of AI and Cyber Security.	K3
CO4	Understand the nature of cybercrimes, the principles of intellectual property rights (IPR), and the legal measures necessary to address and prevent these issues.	K2
CO5	Describe the impact of AI in Society, employment and workforce.	K2

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO–PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	1	1	2	2	3	1	2	1	2	1	3	**2**
CO2	2	3	3	2	2	2	3	1	2	1	2	1	3	**3**
CO3	3	3	2	3	2	2	3	1	1	1	2	3	3	**3**
CO4	2	2	1	1	1	3	3	1	2	1	2	2	2	**2**
CO5	1	1	1	1	1	3	3	2	3	2	3	1	2	**2**

Course Contents / Syllabus

An Overview to AI Ethics	**5 hours**

Definition of AI Ethical principles in AI, Sources of AI data, Legal implications of AI Security Breaches, Privacy and AI Regulations, Key Principles of Responsible AI, Transparency and Accountability, Dual-Use Dilemma, Human-Centric Design, Introduction to Cyber Laws and Ethics, Historical Development of Cyber laws, Legal frameworks.

Module 2	**Fairness and Favoritism in Machine Learning**	**6 hours**

Introduction to Fairness and Bias in AI, Types of Fairness and Bias, Impact of Bias and Fairness in AI, Techniques for Measuring Fairness and Bias, Techniques for Mitigating Bias, Current Policies and Frameworks for Fairness in AI, Bias in Data Collection, Fairness in Data Processing, Generative AI, Types of Bias in Generative AI.

Module 3	**AI Ethics and Cybersecurity Principles**	**5 hours**

Importance of Privacy and Security in AI, AI specific Security Tools and Software, Privacy-Preserving Machine Learning (PPML) and Privacy-Preserving Data Mining (PPDM), Risk Management: Risk Assessment and Incident Response, Regulatory Compliance: GDPR, HIPAA, Case Studies: Implementation of AI Ethics guidelines and best practices in engineering projects.		
Module 4	**Cybercrimes, IPR and Legal Measures**	**8 hours**
Types of Cybercrimes and their Impact, Legal measures for Cybercrime Prevention and Prosecution, IPR: Copyrights, Trademarks, Patents, and Trade Secrets, Ethical Implications of Intellectual Property, Cyber Security and Privacy Issues, Cyber Crime Investigations and Digital Evidence Handling, Overview of Indian Cyber Laws (IT Act 2000 and Amendments), Comparative Overview: Indian vs Global Cyber Laws, Case Study: The ATM Heist – Cosmos Bank Cyber Attack (India, 2018).		
Module 5	**AI Contribution to Social Evolution**	**6 hours**
Positive and Negative Political impacts of AI, Role of AI in Social Media and Communication Platforms, AI-Generated Content and Deepfakes, Key Technical Stakeholders in AI Deployment: Developers, Researchers, Policymakers, Technical Impacts on Employment and Workforce Automation Technologies: Robotic Process Automation (RPA), Autonomous Systems.		
	Total Lecture Hours	**30 hours**

Textbook:

S.NO	Book	Author
1.	Artificial Intelligence: A Guide for Thinking Humans , Penguin Books, 2019.	Melanie Mitchell
2.	Cyber Ethics: Morality and Law in Cyberspace , 7th Edition (2023).	Richard Spinello, Jones & Bartlett Learning,

Reference Books:

S.NO	Book	Author
1.	Artificial Intelligence and Ethics by , BPB Publications, 2023	S. B. Kishor, Debajit Biswas
2.	Cyber Security and Cyber Laws Cengage, India, 2022.	Alfred Basta, Nadine Basta, Sattwik Panda,

NPTEL/ YouTube/ Faculty Video Link:

1.	https://www.youtube.com/watch?v=VqFqWIqOB1g
2.	https://www.youtube.com/watch?v=hVJqHgqF59A
3.	https://www.youtube.com/watch?v=O5RX_T4Tg24
4.	https://www.youtube.com/watch?v=RJZ0pxcZsSQ
5.	https://www.youtube.com/watch?v=I9FOswjTSGg