

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA, GAUTAM BUDDH 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 (Internet of Things)
Second Year

(Effective from the Session: 2026-27)

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA

(An Autonomous Institute)

Computer Science and Engineering (Internet of Things)

Evaluation Scheme

SEMESTER -III

Sl. No.	Subject Codes	Subject Name	Type of Subject	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	CCSCC0301	Employability Skill Development –I	Mandatory	2	0	0	60	40	100				100	2
2	CCSIOT0302	Microcontrollers for Embedded Systems	Mandatory	3	1	0	30	20	50		100		150	4
3	CCSIOT0301	Introduction to IoT 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	CCSIOT0303	Sensors and It's Applications	Mandatory	2	0	0	30	20	50		50		100	2
6	CCSE0303B	Operating Systems and RTOS	Mandatory	3	0	0	30	20	50		100		150	3
7	CCSIOT0351	Introduction to IoT 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	CCSIOT0352	Microcontrollers for Embedded Systems 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	CCSE359	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
		MOOCs (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL		17	1	18	210	140	350	225	400	225	1200	25

*** List of MOOCs (Infosys Springboard) Based Recommended Courses for Second year B. Tech Students**

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1.	CMC0018	Linux for Beginners	Infosys Wingspan (Infosys Springboard)	6h 2m	0.5
2.	CMC0039	Internet of Things 101	Infosys Wingspan (Infosys Springboard)	8h	0.5

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 (IoT)
Evaluation Scheme
SEMESTER IV

Sl. No.	Subject Codes	Subject Name	Type of Subject	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
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	CCSIOT0401	Mobile Application Development	Mandatory	3	0	0	30	20	50		100		150	3
4	CCSIOT0402	Privacy and Security in IoT	Mandatory	3	0	0	30	20	50		100		150	3
5		Departmental 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	CCSIOT0401	Mobile Application Development Lab	Mandatory	0	0	2				25		25	50	1
9	CCSE0455	Web Technologies	Mandatory	0	0	6				50		100	150	3
10	CCSE0459	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	30	20	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 (Infosys Springboard) Based Recommended Courses for Second year B. Tech Students**

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1.	CMC0064	IoT Automation with ESP8266 with Projects	Infosys Wingspan (Infosys Springboard)	7h 21 m	0.5
2.	CMC0065	Data Science for Beginners	NASSCOM	30 h	2
3.	CMC0063	Mobile Apps Development - Advanced Applications	Infosys Wingspan (Infosys Springboard)	14h 23m	1

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 Department Elective (if any): -

S. No.	Subject Code	Subject Name	Bucket Name	Semester
1	CCSAI0411	Data Analytics	AI Driven Analytics	IV
2	CCSCY0411	Introduction to Cyber Security	Cyber Security- I	IV
3	CCSE0411	Python web development with Django	Full Stack Development	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 to 18 =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:											2	0	0	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.														
SECTION B: Enhance numerical ability and logical reasoning skills through speed mathematics, number systems, and analytical problem-solving techniques for competitive aptitude development.														
SECTION C: 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)														
CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	-	2	3	1	-	3	2	2	2	1	1	2
CO2	3	3	3	3	3	2	1	3	2	2	2	2	2	2

CO3	3	3	2	1	-	-	-	1	2	2	2	3	2	2
CO4	-	-	-	-	1	2	-	3	2	2	-	-	-	-
CO5	-	2	1	-	1	2	-	3	2	2	1	-	-	-

Course Contents / Syllabus

Section-A

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.

SECTION-B

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

SECTION-C

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 signaling 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:

BOOK TITLE	AUTHOR
A Project-Based Introduction to Programming	Access Point Publishing
Programming: Principles and Practice Using C++	Bjarne Stroustrup
Effective Modern C++	Scott Meyers
Quantitative Aptitude	R S Aggarwal

Verbal & Non-Verbal Reasoning		R S Aggarwal
<i>Crucial Conversations Tools</i>		Kerry Patterson et al.
<i>HBR Guide to Persuasive Presentations</i>		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: CCSIoT0302	Course Name: Microcontrollers for Embedded Systems	L	T	P	C
Course Offered in: CSE(IoT)		3	1	0	4

Pre-requisite: Basic Electronics, Digital Logic Design and C programming

Course Objectives:

- | |
|--|
| 1. To introduce the fundamental concepts of small and medium scale embedded system design using microcontrollers. |
| 2. To develop understanding of the architecture and internal organization of 8051 microcontrollers. |
| 3. To provide knowledge of assembly language programming for 8051 microcontrollers. |
| 4. To introduce C language programming techniques for embedded system development. |
| 5. To explain the architecture and features of the ATmega32 microcontroller. |
| 6. To develop programming skills for ATmega32 using assembly language and Embedded C. |
| 7. To introduce the architecture and peripherals of the ATmega328P microcontroller. |
| 8. To enable students to design and implement embedded applications using 8051, ATmega32, and ATmega328P microcontrollers. |

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

[illegible]

CO1	Demonstrate the working of the 8051 architecture and its memory structure using diagrams and write assembly language programs.	K3
CO2	Understand the AVR architecture, memory organization and pin configuration; the use of GPIO, timer ports; and serial communication.	K2
CO3	Implement embedded C programs for AVR microcontroller using timers, interrupts, and serial ports.	K3
CO4	Describe the fundamentals of ATmega 328P and embedded systems.	K2
CO5	Analyse ATmega 328P and embedded systems to write assembly-level programming.	K4

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

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

Course Contents / Syllabus

Unit 1	Introduction To Microcontrollers (K3)	12 hours
---------------	--	-----------------

Microprocessors v/s Micro-controllers, Types of Micro-controllers, Processor Architecture – Harvard v/s Von Neumann; CISC v/s RISC.

Combinational Logic Concepts: MUX, DEMUX, Encoder, Decoder

Overview of the 8051 family: 8051 architecture, address, data and control bus, working registers, SFRs, Clock and RESET circuits

Timing & Clock Concepts: Clock signal, frequency, delay, Stack and Stack Pointer, Program Counter, I/O pins, Interrupt Fundamentals, Polling vs Interrupt, ISR Timers, Peripherals., Memory Structures, Data and Program Memory, Timing diagrams and Execution Cycles

Instruction set of 8051, Instruction syntax, Instruction Execution Fundamentals: Fetch–Decode–Execute, Opcode vs Operand assembler directives, Classification of Instructions, Addressing Modes, Assembly language Programs, Memory Interfacing.

Unit 2	Introduction To AVR Microcontroller (K2)	12 hours
---------------	---	-----------------

Overview of AVR family, AVR Microcontroller architecture, status register, RAM, ROM & EEPROM space, On-Chip peripherals, ATmega 32 pin configuration & function of each pin, Fuse bits of AVR.

GPIO ports, Timer ports and Serial communication, Serial vs Parallel, baud rate, Signal Fundamentals: Analog vs Digital, PWM

Concept: Duty cycle, frequency, Analog Concepts: ADC basics. Analog Concepts: ADC Working Principle, ADC Resolution & Step Size, ADC Registers.

Unit 3	AVR Programming In C (K3)	08 hours
AVR Data types, AVR I/O port programming, Logic Programming, Data conversion, Timer programming, Input capture and Wave Generator, Interrupt programming, Serial Port Programming using USART, PWM programming.		
Unit 4	Introduction To ATMEGA 328P Microcontroller (K2)	07 hours
The architecture of ATmega 328P Microcontroller, AVR CPU Core, ATmega 328P register summary, ADC in AVR / ATmega328P, GPIO: Output pin-muxing, Internal Pull up Resistor, PORT Register Descriptions, Assembly Instructions.		
Unit 5	ATMEGA 328P Microcontroller Ports and Assembly Instructions (K4)	06 hours
Timer Ports: Introduction, PWM, Input capture, Timer Register Descriptions Analog input ports: Analog Comparator, Register Descriptions. Interrupt Processing: ISR, Pertinent Register Descriptions. Serial communication: Introduction, UART, USART, Pertinent Register Descriptions.		
Total Lecture Hours		45 hours
Textbook:		
1.The 8051 Microcontroller A Systems Approach 2.The AVR Microcontroller and Embedded System		
Reference Books:		
1. The 8051 Microcontroller And Embedded Systems 2. AVR Microcontroller and Embedded Systems: Using Assembly and C		
Self-Study Content Link		Duration (In Hrs)
MOOCs	Arduino UNO Bootcamp 2023 for Starters - TOC Infosys Springboard	6hrs
Video Lectures	https://youtube.com/playlist?list=PLgwJf8NK-2e4LaLfRo0v9NJL4EIJ677dp&si=hjnv8LbhcQ20VXg4	10hrs
	https://youtube.com/playlist?list=PL80IMgeg12Yt41NB6JY9B2u9-l54Nliev&si=Bhruzi-P5KSk787	20hrs
Web References	Atmel C51 Hardware Manual ATmega32 Microchip Technology ATmega328P	



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



Course Code: CCSIoT0301	Course Name: Introduction to IoT Systems	L	T	P	C
Course Offered in: CSE(IoT)		2	0	0	2

Pre-requisite: Basic Electronics and C programming

Course Objectives:

This course emphasizes the study about introduction of IoT technology, Components, architecture, network communications and applications protocols of IoT. Course also aims at understanding various hardware for IoT, programming concepts using Arduino and Raspberry Pi and study about applications of IoT.

Course Outcome: After completion of the course, the student will be able to	Bloom's Knowledge Level (KL)
CO1 Describe vision, definition, conceptual framework, architecture of IoT, and M2M Communication.	K1
CO2 Use sensors, actuators, and microcontrollers in IoT implementation.	K3
CO3 Execute and verify programs with the help of Arduino, Node MCU, and Raspberry Pi.	K4
CO4 Integrate the communication protocols with applications like Smart metering system, Smart streetlights, home automation and smart city applications.	K4

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

Course Contents / Syllabus

Unit 1	Introduction of IoT and Design Principles	6 hours
Introduction to IoT concepts and principles, Overview of Conceptual Framework and Architectural Framework of IoT, M2M Communication, IoT/M2M systems layers and design standardization, Data enrichment and consolidation, Data aggregation & Dissemination, Introduction to Integrated Developed Environments, Tools and Programming.		
Unit 2	Hardware Components	7 hours
Sensors, different types of Sensors, Transducer, Actuators, Radio Frequency Identification (RFID) Technology. Overview of IOT supported Hardware Computational platforms such as Arduino, Node MCU, Raspberry pi and ARM Microcontrollers.		
Unit 3	Programming Arduino and Raspberry Pi	7 hours
Arduino platform boards anatomy, Arduino IDE coding, using emulator, using libraries, arithmetic addition in Arduino IDE, programming the Arduino for IoT. Programming with Node MCU, Introduction to Raspberry Pi Board. Interfacing and programming the various sensors, IO's etc. with different platforms		
Unit 4	IoT communications and Its Applications	10 hours
Application Protocols: Layered Architecture of IoT Protocols, Communication Technologies, Low range protocols: BLE, ZigBee. Long range protocols: LoRa, Sensor deployment & Node discovery, Smart metering, e-health, Smart city automation, Automotive applications, home automation, Designing of smart streetlights in smart city, Ideation of Mini Project.		

Total Lecture Hours	30 hours
----------------------------	-----------------

Textbook:

S.No	Book Title	Author
1.	"The Internet of Things" Pearson. 1st Edition March 2015	Michael Miller,
2.	"INTERNET OF THINGS", McGraw-Hill, 2nd Edition, May 2022	Raj Kamal
3.	"Internet of Things", Khanna Publications. 1st Edition Jan 2018	Jeeva Jose

Reference Books:

S.No	Book Title	Author
1.	“Programming arduino next steps”, 2 nd Edition, Mc Graw-Hill Education 2018	Simon Monk
2.	“Internet of Things (A Hands-on-Approach)”, 1st Edition, VPT, 2015.	Vijay Madisetti and Arshdeep Bahga
3.	“Rethinking the Internet of Things: A Scalable Approach to Connecting Everything”, 1st Edition, A press Publications, 2013.	Francis daCosta
4	“From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”, 1st Edition, Academic Press, 2014.	Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatios Karnouskos, David Boyle,
5	“The Internet of Things key applications and protocols”. 1st Edition Dec 2011.	Olivier Hersent, David Boswarthick, Omar Elloumi
Self-Study Contents Links:		Duration (in hours)
MOOCs	Introduction To Internet Of Things, IIT Kharagpur https://www.nptel.ac.in/courses/106105166	30 hours approx
Video Lectures	Design of Internet of Things https://www.youtube.com/watch?v=urUBLmXFKl0&list=PLgMDNELGJ1CaBrefq-0eYatfOnoncW0y-	11 hours
Web References	https://www.cisco.com/site/us/en/learn/topics/industrial-iot/what-is-iot.html Arduino Official Documentation https://docs.arduino.cc/	8 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												Bloom's Knowledge Level (KL)			
CO1	Apply tree structures to solve specific problems using various tree operations and algorithms.											K3			
CO2	Analyse the graph data structure and evaluate the efficiency of its operations for problem solving.											K4			
CO3	Evaluate dynamic programming solutions for efficient problem-solving across diverse contexts.											K4			
CO4	Apply efficient backtracking and branch –and –bound techniques across diverse problem-solving scenarios.											K3			
CO5	Understand principles of advanced data structures, including their implementation and applications.											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	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	2	
CO4	3	3	3	2	2	2	1	1	2	1	-	2	1	2	
CO5	3	3	3	2	2	2	1	1	2	2	-	2	1	2	
Course Contents / Syllabus															
UNIT 1		DESIGN AND ANALYSIS OF ALGORITHMS: TREE										9 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: GRAPH										9 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	9 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	9 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	9 HOURS
Red-Black Trees, B – Trees, B+ Trees, Binomial Heaps, Fibonacci Heaps		
Total Lecture Hours		45 HOURS
Textbook:		
Book Details		Authors Name
<i>Data Structures and Algorithms in Python: An Indian Adaptation, 1st Edition, 2021</i>		<i>Michael T. Goodrich, Roberto Tamassia</i>
<i>Data Structures Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd, 2nd Edition, 2017.</i>		<i>Lipschutz</i>
<i>Introduction to Algorithms, Printice Hall of India, 4th Edition, 2022</i>		<i>Thomas H. Coreman, Charles E. Leiserson and Ronald L. Rivest,</i>
Reference Books:		
Book Details		Authors Name
<i>Data Structure Using C, Oxford University Press, 2nd Edition, 2014.</i>		<i>Reema Thareja</i>
<i>Data Structure Using C, Pearson Education India, 2nd Edition, 2011.</i>		<i>AK Sharma</i>
<i>C and Data structure, Wiley Dreamtech Publication, 1st Edition, 2004.</i>		<i>P. S. Deshpandey</i>
Self-Study Content Links:		Duration (in Hrs)

MOOC'S	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01384203240484864010470_shared/overview	18 hrs
	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0130944399140945922584_shared/overview	3 hrs
VIDEO LECTURES	https://www.youtube.com/watch?v=zWg7U0OEAOE&list=PLBF376_3AF2E1C572F https://www.youtube.com/watch?v=LQx9E2--p5c&pp=ygUMYXJyYXlIG5wdGVs https://www.youtube.com/watch?v=K7VIKIUdo20&pp=ygUPbGluay_BsaXN0IG5wdGVs https://www.youtube.com/watch?v=g1USSZVWDsY&list=PLB3CD0BBB95C1BF09&index=2&pp=iAQB https://www.youtube.com/watch?v=THMyk2_p530&pp=ygUccXVldWUgZGF0YSBzdHJlY3RlcmUgICBucHRlbA%3D%3D https://www.youtube.com/watch?v=_VV9v41FIq0&pp=ygUZZGI2aWRIIGFuZCBjb25xdWVyICBucHRlbA%3D%3D https://www.youtube.com/watch?v=ARvQcqJ_-NY&list=PLfFeAJ-vQopt_S5XlayyvDFL_mi2pGJE3	40 hrs
WEB REFERENCES	https://www.msajce-edu.in/academics/it/LectureNote/CD3291-LN.pdf	

	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. Use cases of Smart Sensors: iroute, Marine Traffic Tracker	
--	--	--

		Total Lecture Hours	30 hours
Textbook:			
S.No	Book Title	Author	
1	“Sensors Handbook”, McGraw-Hill, 2 nd Edition (2009)	Sabrie Soloman	
2	“Sensor & Instrumentation” , Dhanpat Rai & Co., 1 st Edition (2014)	A.K. Sawhney	
Reference Books:			
S.No	Book Title	Author	
1	“Introduction To Sensors and Transducers”, Notion Press, 1 st Edition, (2022)	Mr. M.sivasubramanian	
2	“Sensor Systems Fundamentals and Applications”, CRC Press 1 st Edition (2016)	Clarence W De Silva	

Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://onlinecourses.nptel.ac.in/e-learning/preview/noc24_ee45	30 hours
Video Lectures	https://www.youtube.com/playlist?list=PLgMDNELGJ1CbufZjqWa8uoSlQWKqVwPN7	20 hours
Web References	https://ieee-sensors.org/	5 hours

Course Code: CCSE0303B	Course Name: Operating Systems and RTOS	L	T	P	C
Course Offered in: CSE-IoT		3	0	0	3

Pre-requisite: Computer Fundamentals, Programming Basics, Data Structures Basics.

Course Objectives:

1. To understand the basic concepts, architecture, functions, and services of Operating Systems and Real-Time Operating Systems (RTOS).
2. To learn process management, task management, CPU scheduling, and RTOS scheduling algorithms for efficient system performance.
3. To study synchronization techniques, inter-process communication, and inter-task communication mechanisms in OS and RTOS environments.
4. To analyze memory management, virtual memory, and real-time memory allocation techniques used in operating systems.
5. To understand file management concepts, disk scheduling methods, and popular RTOS platforms such as FreeRTOS, VxWorks, and QNX.

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

Course Outcome:		Bloom s Level
CO1	Explain operating system fundamentals, system architecture, real-time systems, and RTOS concepts.	K2
CO2	Analyze process management, task management, CPU scheduling, and RTOS scheduling algorithms.	K4
CO3	Apply synchronization, communication, semaphore, mutex, and inter-task communication techniques in OS and RTOS.	K3
CO4	Analyze memory management, virtual memory, paging, segmentation, and RTOS memory allocation techniques.	K4
CO5	Explain file management, disk scheduling, and compare popular RTOS platforms and real-time applications.	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	PSO3
CO1	3	2	-	-	2	-	-	1	-	1	1	2	1	1
CO2	3	3	2	2	2	-	-	1	-	1	1	2	2	2
CO3	3	3	2	2	3	-	-	1	-	1	1	2	2	2
CO4	3	2	1	1	2	-	-	1	-	1	1	2	2	2
CO5	3	3	2	2	3	-	-	1	1	1	2	2	2	3

Course Contents / Syllabus

Unit 1	Operating System Fundamentals and Introduction to RTOS	09 hours
Operating System Functions and Architecture, Types of Operating Systems: Batch, Multiprogramming, Multitasking, Time Sharing and Real-Time Systems, System Calls and Kernel, Process Concept, Introduction to Real-Time Systems, Characteristics of Real-Time Systems, Hard, Soft and Firm Real-Time Systems, Difference between GPOS and RTOS, Features, Objectives and Applications of RTOS.		
Unit 2	Process Management and Scheduling	08 hours
Process States, Task States, Process Control Block (PCB), Task Control Block (TCB), Threads, CPU Scheduling Algorithms: FCFS, SJF, Priority Scheduling and Round Robin, Scheduler and Dispatcher, Context Switching, Introduction to RTOS Scheduling, Preemptive and Non-Preemptive Scheduling, Priority-Based Scheduling, Rate Monotonic Scheduling (RMS), Earliest Deadline First (EDF) Scheduling and Timer Management.		

Unit 3	Synchronization and Inter-Task Communication	10 hours
Concurrency, Critical Section Problem, Classical Synchronization Problems, Semaphores and Monitors, Inter-Task Communication and Synchronization in RTOS, Semaphore and Mutex, Message Queues and Pipes, Priority Inversion and Priority Inheritance, Interrupt Handling and Task Communication Mechanisms.		
Unit 4	Memory Management in OS and RTOS	10 hours
Memory Allocation Strategies: First Fit, Best Fit and Worst Fit, Paging, Segmentation, Virtual Memory, Demand Paging, Page Replacement Algorithms: FIFO and LRU, Memory Management in RTOS, Real-Time Kernel Concepts and Memory Allocation Techniques.		
Unit 5	File Management and RTOS Platforms	8 hours
File Access Methods, File Allocation Methods, Bit Vector Method, Disk Scheduling and Disk Management, Introduction to FreeRTOS, VxWorks and QNX, Comparison of Popular RTOS Platforms, RTOS Applications in Embedded Systems and Case Studies of Real-Time Applications.		
Total Lecture Hours		45 hours
Textbook: 1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, <i>Operating System Concepts</i>, Wiley Publications. 2. Raj Kamal, <i>Embedded Systems: Architecture, Programming and Design</i>, Tata McGraw Hill.		
Reference Books 1. William Stallings, <i>Operating Systems: Internals and Design Principles</i>, Pearson. 2. Jane W. S. Liu, <i>Real-Time Systems</i>, Pearson Education. 3. Andrew S. Tanenbaum, <i>Modern Operating Systems</i>, Pearson. 4. Richard Barry, <i>Mastering FreeRTOS</i>, Real Time Engineers Ltd.		
Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://nptel.ac.in/courses/106105214 https://nptel.ac.in/courses/106105172 Complete UNIX & Linux OS Fundamentals Training TechA Bash Shell Scripting Certification	30 hrs
Video Lectures	https://nptel.ac.in/courses/106105214 https://nptel.ac.in/courses/106105172 https://youtube.com/playlist?list=PLEBQazB0HUyQ4hAPU1cJED6t3DU0h34bz&si=3lbsDgiKO_QIGVD_F https://youtube.com/playlist?list=PLxCzCOWd7aiGz9donHRrE9I3Mwn6XdP8p&si=Ynnpz_TrpVQyv3fY	15 hrs
Web References	https://github.com/Rayyan9477/Operating-Systems-and-Shell-Scripting https://github.com/gowriganeshns/OS-Linux-commands-Shell-script https://www.freertos.org/	10 hrs



LAB Course Code: CCSIOT0351				LAB Course Name: Introduction to IoT systems Lab							L	T	P	C	
Course Offered in: CSE(IoT)											0	0	4	2	
Pre-requisite: Basic Electronics and C programming															
Course Objectives: To familiarize the students to the basics of Internet of things, sensors, development board, actuators, hardware and protocols.															
Course Outcome: After completion of the course, the student will be able to													Bloom's Knowledge Level (KL)		
CO1	Describe the functionality of computing, sensing and actuating components of Internet of things.													K1	
CO2	Develop IoT applications using Arduino IDE.													K6	
CO3	Design, develop and deploy real time mini projects of IoT Applications.													K6	
CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)															
CO-PO Mappings	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3	
CO1	3	2	—	—	2	—	—	—	—	—	—	2	—	—	
CO2	2	2	3	1	3	—	—	—	1	1	—	3	1	—	
CO3	2	3	3	2	3	1	1	1	2	1	1	3	2	1	
List Of Practical's (Indicative & Not Limited To)															
Lab No.	Unit	Title of Program												CO Mapping	
1	1	Installation of Arduino IDE and introduction to tools, assembly and libraries.												CO 1	
2	1	Getting Programming board Info and configure boot loader settings using Arduino IDE.												CO 1	
3	1	Study and design IoT reference architecture for IoT based applications like Smart home												CO 1	
4	2	Study Hardware Architecture and Pin Out of Arduino UNO Board. Compare Arduino Uno, Arduino Nano and Arduino Mega. Identification of their use case according to given scenario.												CO 1	
5	2	Study Hardware Architecture and Pin Out of Node MCU and ESP8266. Identification of their use case according to given snapshot.												CO 1	
6	2	a. Study Pin out Architecture of Sensors and actuators b. DHT 11 Sensor c. MQ 135 Sensor d. MQ 7 Sensor e. MQ 3 Sensor f. Ultrasonic Sensor HC-04 g. Rain Sensor h. Soil moisture Sensor i. PIR Sensor j. LDR Sensor k. Line Sensor l. Colour Sensor m. Servo Motor n. Relay												CO 1	

7	3	a. Working with structures using Arduino IDE b. Working with Variables using Arduino IDE c. Working with Flow control using Arduino IDE d. Working with Digital i/o using Arduino IDE e. Working with Analog i/o using Arduino IDE f. Working with Time function using Arduino IDE g. Working with Math function using Arduino IDE h. Working with Random function using Arduino IDE i. Working with Serial communication using Arduino IDE j. Working with loops and control statements using Arduino IDE k. Working with PinMode function using Arduino IDE l. Working with analog Read, analog Write, digital Read, digital Write using Arduino IDE m. Blinking LED Program using Arduino Uno	CO 2
8	3	"Write a program using Arduino uno to generate a random number in between 0 to 25. Use 4 LEDs (Red, Green, Blue, Yellow) and design LED patterns as (i) if random number is less than 5 then only Red LED should glow. (ii) if random number is in between 5-10 then only Blue LED should glow. (iii) if random number is in between 11-20 then only Yellow LED should glow. (iv) if random number is greater than 20 then only Green LED should glow." "Write a program using Arduino uno for addition of digits of a user defined number. Example: number is 257 then output should be 14." Write a program to take LED color as input from user and glow that LED using Arduino Uno.	CO 2
9	3	Interfacing of DHT 11 Sensor with Arduino Uno. Implement LED mechanism for notifying rise in temprature. Interfacing of MQ 135/MQ7 Sensor with Arduino Uno. Implement alarm mechanism for notifying rise in amount of hazardous gases in air. Interfacing of MQ 3 Sensor with Arduino Uno. Implement alarm mechanism for checking amount of alcohol in the air.	CO 2
10	3	a. Interfacing of Ultrasonic Sensor HC-04 with Arduino Uno. b. Interfacing of Rain Sensor with Arduino Uno. Implement buzzer mechanism as the sensor identify rain. c. Interfacing of Soil moisture Sensor with Arduino Uno. d. Interfacing of PIR Sensor with Arduino Uno. e. Interfacing of LDR Sensor with Arduino Uno. f. Interfacing of LCD with Arduino Uno g. Interfacing of I2C LCD with Arduino Uno	CO 2
11	4	a. Interfacing Bluetooth Module with Arduino Uno b. Connecting Node MCU with Wi-fi Hotspots using Arduino IDE c. Interfacing of DHT 11 Sensor with Node MCU d. Interfacing of MQ 135 Sensor with Node MCU	CO 2

		e. Interfacing of MQ 7 Sensor with Node MCU f. Interfacing of MQ 3 Sensor with Node MCU	
12	4	a. Interfacing of Ultrasonic Sensor HC-04 with Node MCU b. Interfacing of Rain Sensor with Node MCU c. Interfacing of Soil moisture Sensor with NodeMCU d. Interfacing of PIR Sensor with Node MCU e. Interfacing of LDR Sensor with Node MCU	CO 2
13	5	a. Sending Data to Thingspeak Cloud Server using Node MCU b. Detection of LPG Gas using MQ6 and Node MCU. Notify Thing speak server that "LPG gas Leakage has been detected".	CO 2
14	5	Controlling LED with Node MCU using Blynk cloud App.	CO 2
15	5	Development of Mini Project Sample Projects: Introduction to IoT Projects.xlsx	CO 3
Total Hours: 60 hrs.			



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



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) /IT//CSE(AI)/CSE(AIML)/CSE(DS)/CSE(CS)												0	0	4	2
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															
SNO	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	2	
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: 60 hrs.		

LAB Course Code: CCSIoT0352					LAB Course Name: Microcontrollers For Embedded Systems Lab							L	T	P	C
Course Offered in: CSE-IoT												0	0	4	2
Pre-requisite: Basic Electronics, Digital Logic Design and C programming															
Course Objectives:															
Course Objective: It gives hands-on training of interfacing external sensors and actuators with microcontroller. to develop assembly and C language programming skills for real time applications of 8051, ATmega32, and ATmega 328P Microcontroller															
Course Outcome: After completion of the course, the student will be able to															
CO1.					Implement programs to perform operations and interfacing using 8051 and ATmega32 microcontroller.										(K3)
CO2					Design circuit and programs to perform operations and interfacing using ATmega328P microcontroller.										(K6)
CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)															
COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3	
CO1	3	3	3	2	3	–	–	1	2	1	2	3	2	1	
CO2	3	3	3	3	3	1	–	1	2	2	2	3	3	2	
List Of Practical's (Indicative & Not Limited To)															
1. To gain an understanding of the 8051 microcontrollers, including its introduction and architectural features.															
2. Write and simulate different assembly programs to study 8051 data transfer instructions using Keil compiler and 8051 kit.															
3. Write and simulate different assembly programs to study 8051 Arithmetic, Logical and Boolean instructions.															
4. Write and simulate different assembly programs to study 8051 branching instructions.															
5. To understand the basic concepts and architectural structure of the ATmega32 microcontroller															
6. To install and configure Microchip Studio IDE/ MPLAB X IDE for AVR development.															
7. Write and simulate different C language programs to study about LED and switch interfacing with ATMEGA32 using Microchip studio compiler/ MPLAB X IDE.															
8. Write and simulate different C language programs to study about Keypad and LCD interfacing with ATMEGA32 using Microchip studio compiler/ MPLAB X IDE.															
9. Design a circuit to read analog signal using an ATMEGA32 microcontroller and display it on LCD using Microchip studio compiler /MPLAB X IDE.															
10. Design a circuit to interface Serial Communication with ATMEGA32 microcontroller using Microchip studio compiler.															
11. Design a circuit to interface a Stepper motor using ATMEGA32 microcontroller using Microchip studio compiler /MPLAB X IDE															
12. Design a circuit to interface a DC motor using ATMEGA32 microcontroller using Microchip studio compiler /MPLAB X IDE															
13. Design a circuit to interface a Relay using ATMEGA32 microcontroller using Microchip studio compiler /MPLAB X IDE															
14. To understand the required hardware, tools and setup for programming the ATmega328P microcontroller assembly language programming.															
15. To integrate hardware and software components to build a complete project workflow for ATmega328P Programming using Microchip Studio and MPLAB® PICKit™ 4/5 In-Circuit Debugger															
16. Write and execute an assembly program for blinking an LED using ATmega 328P and MPLAB® PICKit™ 4/5 In-Circuit Debugger.															
17. Write and execute an assembly program to use Buttons as input using ATmega 328P and MPLAB® PICKit™ 4/5 In-Circuit Debugger.															
18. Write and execute an assembly program to configure and use external interrupts for responsive event handling															

using ATmega 328P and MPLAB® PICKit™ 4/5 In-Circuit Debugger.
19. Write and execute an assembly program to transmit and receive data via UART using Atmega 328P and MPLAB® PICKit™ 4/5 In-Circuit Debugger.
20. Write and execute an assembly program to generate precise delays using Timer/Counter modules using Atmega 328P and MPLAB® PICKit™ 4/5 In-Circuit Debugger.
21. Write and execute an assembly program to control servo motor angles using PWM generated by Timer1 using Atmega 328P and MPLAB® PICKit™ 4/5 In-Circuit Debugger.
22. Write and execute an assembly program to read analog temperature data and process it using Atmega 328P and MPLAB® PICKit™ 4/5 In-Circuit Debugger.

Course Code: CCSE0355				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	PO1	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	2	
CO3	3	3	3	3	3	2	2	-	2	1	2	3	3	1	
CO4	3	3	3	3	3	2	2	1	2	1	2	3	2	2	
CO5	3	3	3	3	3	2	2	1	2	1	2	3	3	2	
Course Contents / Syllabus															
Unit 1		Basics of Java Programming												16 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. Exception 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_0125409722749255681063_shared/overview	46h 13m
Video Lectures	https://www.youtube.com/watch?v=iw1Xf_33YM0&list=PLQeARBV9gAFujcBWJhBT2XXsuMIIfETBy	30 h
Web References	1. https://www.geeksforgeeks.org/dsa/introduction-of-object-oriented-programming/ 2. https://en.wikipedia.org/wiki/Object-oriented_programming	20 m 58 m

S.N.	List Of Practical's (Indicative & Not Limited To)	CO
1	Understanding Text Editors to Write Programs Compile and run first java file Byte Code and class file	CO1
2	Sketch a class and object diagram describing the sales order system of restaurant	CO1
3	Sketch a class diagram describing the circle and rectangle class	CO1
4	Sketch a class diagram for a college platform including, classroom, playground, chair, table, smart board, teaching staff etc.	CO1
5	Sketch a class diagram containing class called Employee, which models an employee with an ID, name and salary. Add method raiseSalary(percent) that increases the salary by the given percentage.	CO1
6	Program to display default value of all Primitive data types	CO1
7	Implement the code using main() method to calculate and print the Total and Average marks scored by a student from the input given through the command line arguments. Assume that four command line arguments name, marks1, marks2, marks3 will be passed to the main() method in the below class with name TotalAndAvgMarks.	CO1
8	Write code which uses if-then-else statement to check if a given account balance is greater or lesser than the minimum balance. Write a class BalanceCheck with public method checkBalance that takes one parameter balance of type double. Use if-then-else statement and print Balance is low if balance is less than 1000. Otherwise, print Sufficient balance.	CO1
9	A class NumberPalindrome with a public method isNumberPalindrome that takes one parameter number of	CO1

	type int. Write a code to check whether the given number is palindrome or not. For example Cmd Args : 333 333 is a palindrome	
10	Write a class FibonacciSeries with a main method. The method receives one command line argument. Write a program to display fibonacci series i.e. 0 1 1 2 3 5 8 13 21	CO1
11	Write a Java Program to find the Factorial of a given number.	CO1
12	Java Program to create a class, methods and invoke them inside main method.	CO1
13	Write a Java program to illustrate the abstract class concept. Create an abstract class Shape, which contains an empty method numberOfSides(). Define three classes named Trapezoid, Triangle and Hexagon extends the class Shape, such that each one of the classes contains only the method numberOfSides(), that contains the number of sides in the given geometrical figure. Write a class AbstractExample with the main() method, declare an object to the class Shape, create instances of each class and call numberOfSides() methods of each class.	CO1
14	Java program to illustrate the static field in the class.	CO1
15	Java Program to illustrate static class.	CO1
16	Write a java program to access the class members using super keyword	CO1
17	Java program to access the class members using this keyword	CO1
18	Implement an interface named MountainParts that has a constant named TERRAIN that will store the String value "off_road". The interface will define two methods that accept a String argument name newValue and two that will return the current value of an instance field. The methods are to be named: getSuspension, setSuspension, getType, setType.	CO1
19	Java program to demonstrate nested interface inside a interface.	CO1
20	Java program to demonstrate nested interface inside a class.	CO1
21	Java program to explicit implementation of garbage collection by using finalize() method	CO1
22	Java program to implement Single Inheritance	CO1
23	Java program to implement multi- level Inheritance	CO1
24	Java program to implement constructor and constructor overloading.	CO1
25	Java program implement method overloading.	CO1
26	Java program to implement method overriding.	CO1
27	.Java program to implement lambda expression without parameter.	CO1
28	Java program to implement lambda expression with single parameter.	CO1
29	Java program to implement lambda expression with multi parameter.	CO1
30	Java program to implement lambda expression that iterate list of objects	CO2
31	.Java program to define lambda expressions as method parameters	CO2
32	Write a class CountOfTwoNumbers with a public method compareCountOf that takes three parameters one is arr of type int[] and other two are arg1 and arg2 are of type int and returns true if count of arg1 is greater than arg2 in arr. The return type of compareCountOf should be boolean. Assumptions: • arr is never null arg1 and arg2 may be same	CO2
33	Java program to show the multiplication of two matrices using arrays.	CO2
34	Java Program to search an element using Linear Search	CO2
35	.Java program to search an element using Binary Search	CO2
36	Java Program to sort element using Insertion Sort	CO2
37	Java Program to sort element using Selection Sort– Largest element Method	CO2
38	Java program to Sort elements using Bubble Sort	CO2
39	Java program to create user defined package.	CO3
40	Java Program to create a sub- classing of package	CO3
41	Implement the following: 1. Import package.*;	CO3

	2. import package.classname; Using fully qualified name.	
42	Implement and demonstrate package names collision in java	CO3
43	Java program to handle and Arithmetic Exception Divided by zero	CO3
44	.Java Program to implement User Defined Exception in Java	CO3
45	Java program to illustrate finally block	CO3
46	Java program to illustrate Multiple catch blocks	CO3
47	Java program for creation of illustrating throw in exception handling.	CO3
48	Implement the concept of Assertion in Java Programming Language	CO3
49	Implement the concept of Localization in Java Programming Language.	CO3
50	Java program to print the output by appending all the capital letters in the input string.	CO3
51	Java program that prints the duplicate characters from the string with its count.	CO3
52	Java program to check if two strings are anagrams of each other	CO3
53	Java Program to count the total number of characters in a string	CO3
54	.Java Program to count the total number of punctuation characters exists in a String	CO3
55	Java Program to count the total number of vowels and consonants in a string	CO3
56	Java Program to show .equals method and == in java	CO3
57	Given a string, return a new string made of n copies of the first 2 chars of the original string where n is the length of the string. The string may be any length. If there are fewer than 2 chars, use whatever is there. If input is "Wipped" then output should be "WiWiWiWiWi".	CO3
58	Given two strings, a and b, create a bigger string made of the first char of a, the first char of b, the second char of a, the second char of b, and so on. Any leftover chars go at the end of the result. If the inputs are "Hello" and "World", then the output is "HWeolrlld".	CO3
59	Java program to show the usage of string builder.	CO3
60	Java program to show the usage of string buffer.	CO3
61	Creating and Running a Thread	CO3
62	Implementing Runnable Interface	CO3
63	Synchronizing Threads with lock	CO3
64	Synchronizing Threads without lock	CO3
65	Java program to implement even and odd threads by using Thread class .	CO4
66	Java program to implement even and odd threads by using Runnable interface.	CO4
67	Java program to synchronize the threads by using Synchronize statements and Synchronize block.	CO4
68	.Write a program where the client sends a message to the server, and the server prints it by using TCP	CO4
69	Implement a server that can handle multiple clients simultaneously using UDP	CO4
70	Write a client-server application where the client uploads a file and the server saves it by using TCP/UDP.	CO4
71	Java program to implement that read a character stream from input file and print it into output file.	CO4
72	.Java program to implement that merge the content of two files (file1.txt, file2.txt) into file3.txt.	CO4
73	Write a Java program that reads the contents of one file and copies them to another file.	CO4
74	Write a Java program that reads a text file and counts the number of words in it.	CO4
75	Write a Java program that reads a text file and counts the frequency of each word in it.	CO4
76	Write a Java program that reads a text file and adds line numbers to each line. The program should create a new file with the line numbers added to the beginning of each line.	CO4
77	Write a Java program that reads two binary files and compares them byte by byte to determine if they are identical. Display a message indicating whether the files are the same or different.	CO4
78	Program to create a frame with three button in AWT and swing	CO4
79	Program to display message with radio buttons in swing	CO4
80	Program to display "All The Best" in 5 different colors on screen. (Using AWT/Swing)	CO5
81	Program to implement handling in a button "OK"	CO5
82	Java Program to implement BorderLayout	CO5
83	.Java Program to implement GridLayout	CO5
84	Java Program to implement BoxLayout	CO5
85	Java Program to implement CardLayout	CO5
86	Java program to implement Generic class	CO5

87	Java program to illustrate Generic methods	CO5
88	Java program to implement wildcard in generics	CO5
89	Java program to implement of methods of HashSet	CO5
90	Java Program to implement methods available in HashMap class	CO5
91	Program to add, retrieve, and remove element from ArrayList	CO5
92	Create a method which can accept a collection of country names and add it to ArrayList with generic defined as String and return the List.	CO5
93	Create a method which can create a HashSet containing values 1-10. The Set should be declared with the generic type Integer. The method should return the Set.	CO5
94	Java program to implement autoboxing	CO5
95	Java program to implement unboxing	CO5
96	Develop a java class with a method <i>storeEvenNumbers(int N)</i> using ArrayList to store even numbers from 2 to N, where N is a integer which is passed as a parameter to the method <i>storeEvenNumbers()</i> . The method should return the ArrayList (A1) created.	CO5
97	Create a method that accepts the names of five countries and loads them to an array list and returns the list.	CO5
98	Create a method which can accept a collection of country names and add it to ArrayList with generic defined as String and return the List.	CO5

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 resources 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 SO _x , NO _x , CO _x , 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	7 min 42 sec
	https://www.youtube.com/watch?v=qt8AMjKKPDo	15 min
	https://www.youtube.com/watch?v=brF0RWJyx9w	3 min 25 sec
	https://www.youtube.com/watch?v=GK_vRtHJZu4	4 min 25 sec
WEB REFERENCES	https://www.youtube.com/watch?v=yqev1G2iy20 ,	28 min
	https://www.youtube.com/watch?v=_74S3z3IO_I ,	31 min 35 sec
	https://www.youtube.com/watch?v=jXVw6M6m2g0	11 min

Syllabus Semester IV



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



Course Code: CASCC0401				Course Name: EMPLOYABILITY SKILL DEVELOPMENT - II							L	T	P	C
Course Offered in: IV Semester											2	0	0	2
Note: Section C - offered in B.Tech. III/IV SEM														
Pre-requisite: Students should have successfully completed communication skills courses in the first year.														
Course Objectives:														
Section A: Develop proficiency in C++ programming, data management, and project development by applying concepts, teamwork, and problem-solving to build practical applications.														
Section B: Enhance quantitative aptitude skills through business mathematics concepts, enabling accurate problem-solving in financial calculations, ratios, and real-world numerical applications.														
Section C: 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.									K6			
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		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)														
CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	-	2	3	1	-	3	2	2	2	-	-	-

CO2	3	3	3	3	3	2	1	3	2	2	2	-	-	-
CO3	3	3	2	1	-	-	-	1	2	2	2	-	-	-
CO4	-	-	-	-	1	2	-	3	2	2	-	-	-	-
CO5	-	2	1	-	1	2	-	3	2	2	1	-	-	-

Course Contents / Syllabus

Section-A

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

SECTION-B

UNIT 3	BUSINESS MATH 1	16 hours
---------------	------------------------	-----------------

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

SECTION-C

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		
<i>*A list of problem statements will be provided by the mentor; you may choose anyone.</i> <i>Note: Students may choose an alternative problem statement with prior approval from the mentor.</i>		
REFERENCE BOOKS:		
BOOK TITLE	AUTHOR	
A Project-Based Introduction to Programming	Access Point Publishing	
Programming: Principles and Practice Using C++	Bjarne Stroustrup	
Effective Modern C++	Scott Meyers	
Quantitative Aptitude	R S Aggarwal	
Verbal & Non-Verbal Reasoning	R S Aggarwal	
<i>Crucial Conversations Tools</i>	Kerry Patterson et al.	

<i>HBR Guide to Persuasive Presentations</i>		Nancy Duarte
SELF-STUDY CONTENT LINKS:		DURATION (IN HRS)
MOOCs	nptel.ac.in/courses/109104107	8 Weeks
Video Lectures	Employability Skills Training Session - YouTube	1 hour
Web References	https://www.scribd.com/document/690580893/EMPLOYABILITY-SKILLS-NOTES-full-1	-



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												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 Mapping		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	2
CO3		3	3	2	3	3	2	-	2	1	-	2	3	3	2
CO4		3	2	2	2	2	2	1	-	1	1	1	3	3	2
CO5		2	2	2	2	3	2	1	-	1	2	2	3	3	2
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: 45 Hours		
TEXTBOOK		
BOOK TITLE		AUTHOR
Database Concepts, McGraw Hill, 7th Edition, 2020		Avi Silberschatz Henry F. Korth S. Sudarshan
Fundamentals of Database Systems, Addison Wesley, 7th edition, 2016		R. Elmasri and S. B. Navathe
REFERENCE BOOKS:		
BOOK TITLE		AUTHOR
Database Systems: A practical Approach to Design, Implementation and Management. Pearson Education,3rd Edition, 2007.		Thomas Connolly , Carolyn Begg
NoSQL and SQL Data Modeling: Bringing Together Data, Semantics, and Software, Ted Hills, 1st Edition,2016.		Ted Hills
SELF-STUDY CONTENT LINKS:		DURATION (IN HRS)
MOOCS	nptel.ac.in/courses/106105175	16 HOURS
VIDEO LECTURES	DBMS L1 Inauguration & Introduction 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	30 HOURS

	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/ https://www.scribd.com/document/666843210/Database-Management-System-DBMS-HandWritten-Notes	2 HOURS



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



Course Code: CCSIOT0401	Course Name: Mobile Application Development	L	T	P	C
--------------------------------	--	----------	----------	----------	----------

Course Offered in: CSE(IoT)	3	0	0	3
------------------------------------	----------	----------	----------	----------

Pre-requisite: Overview of programming language: JAVA and XML.

Course Objectives:

1. To introduce the fundamentals of mobile application development and programming technologies.
2. To provide knowledge of Android and Flutter frameworks for building mobile applications.
3. To understand mobile architecture, UI design principles, and development practices.
4. To explore engineering challenges, tools, and methodologies used in mobile app development.
5. To enable students to design, develop, test, and deploy efficient mobile applications.

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

Course Outcome	Blooms Level
CO1 Describe the fundamental concepts of mobile applications, Android ecosystem, UI components, layouts, activities, and intents.	K2
CO2 Apply Android architecture concepts, lifecycle components, testing methods, and deployment processes to build functional mobile applications.	K3
CO3 Analyze mobile application features such as notifications, SMS services, GPS, Wi-Fi, sensors, and REST API-based communication.	K4
CO4 Apply Flutter and Dart programming concepts to develop cross-platform mobile applications using widgets, layouts, and UI design components.	K3
CO5 Develop advanced mobile applications using Flutter by integrating Firebase Firestore, SQLite, gesture controls, and animation techniques.	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	–	–	1	–	–	–	1	–	1	2	1	–
CO2	3	3	2	1	3	–	–	1	1	1	1	3	2	1
CO3	2	3	2	2	3	1	–	1	1	–	1	3	1	1
CO4	2	2	3	1	3	–	–	1	1	1	2	3	2	1
CO5	2	3	3	2	3	1	1	2	1	1	2	3	2	2

Course Contents / Syllabus

Unit 1	Mobile Fundamentals	08 hours
Mobile applications, History of mobile application, types of mobile applications, Introduction to Android, Android ecosystem, Android SDK and Installing and running applications on Android Studio, Creating AVDs, Android API levels (versions & version names), Android Development Tools. Fundamental UI design, layout and view types, Text Views, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers, Resources, Activities, Intents.		
Unit 2	Architecture & Lifecycle	08 hours
Mobile Hardware Architecture, SoC architecture; Mobile Software Architecture, Basic Building blocks , Activities, Services, Broadcast Receivers & Content providers, Event Handling. Testing, Types of testing, Testing Tools in Android Studio, Debugging and Profiling Tools, Publishing the Application.		
Unit 3	Features & Integration	10 hours
Notifications – Creating and Displaying notifications, Toasts, Handling SMS, Sending SMS. Interaction with server-side applications – Using Google Maps, GPS and Wi-Fi, Interfacing sensor data with mobile application. Concepts of Notifications and Toasts, Fundamentals of SMS communication, client-server communication and REST APIs, Overview of GPS, Wi-Fi, and location-based services, Principles of Google Maps integration, Introduction to mobile sensors, Concepts of runtime permissions and security.		
Unit 4	Flutter Development	10 hours

History and Features of Flutter, Installing Flutter and Setting Up IDE, Flutter Architecture Overview, Creating Your First Flutter App: Hello World, Introduction to Dart, Data Types and Variables, Control Flow, Functions in Dart, Object-Oriented Programming (OOPs). Widget, Scaffold Class, Carousel Slider, Staggered Grid View, Tabs & TabBarView, Horizontal ListView, Slidable Widget.		
Unit 5	Animation & Database	09 hours
Introduction to Data Persistence, Importance of local vs. cloud data storage, SQLite in Flutter, Firebase setup and configuration in Flutter, Connecting Flutter App to Firestore, Creating Firestore Collections and Documents, Performing CRUD Operations, Real-time Data Sync. Gesture Detection in Flutter, Tap, Swipe, Drag, Long Press, Animations in Flutter, Implicit & Explicit Animations		
Total Lecture Hours		45 hours
Textbook: 1. Neil Smyth, Android Studio Flamingo Essentials – Java Edition, Payload Media, 2023 2. Rap Payne, Beginning App Development with Flutter, Apress, 2019 3. Deepti Chopra & Roopal Khurana, Flutter and Dart: Up and Running, BPB Publications, 2023		
Reference Books: 1. Bill Phillips & Brian Hardy, Android Programming: Big Nerd Ranch Guide, Big Nerd Ranch, 2019 2. Eran Boudjnah, Clean Architecture for Android, BPB Publications, 2022 3. Eric Windmill, Flutter in Action, Manning, 2020 4. Scott Gowell & Jeff McWherter, Professional Mobile Application Development, Wrox, 2012		
Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://onlinecourses.nptel.ac.in/e-learning/preview/noc20_cs52	30hrs
Video Lectures	https://www.youtube.com/watch?v=fis26HvvDII	11hrs
Web References	https://docs.flutter.dev/install https://developer.android.com/studio/install	3-4hrs

Course Code: CCSIoT0402				Course Name: Privacy and Security in IoT								L	T	P	C
Course Offered in: CSE(IoT)												3	0	0	3
Pre-requisite: Python, IoT fundamentals															
Course Objectives:															
Course Objectives: The objective of the course is to understand the importance of security and privacy in IoT systems and apply fundamental security techniques to protect IoT devices and data.															
Course Outcome: After completion of the course, the student will be able to															
CO1:	Understand fundamental concepts of IoT security, including CIA triad, security architecture, and common threats in IoT environments.														K1
CO2:	Apply techniques to identify device-level vulnerabilities, firmware risks, and lightweight cryptography in IoT systems.														K3
CO3:	Apply secure communication techniques and protocols to mitigate network attacks														K3
CO4:	Implement authentication, authorization, and trust mechanisms in IoT Devices.														K5
CO5:	Analyze secure IoT solutions using real-world case studies and vulnerability assessment.														K4
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	–	–	2	3	1	2	
CO2	3	3	2	1	2	–	–	–	–	–	2	2	1	1	
CO3	3	2	2	1	3	–	–	–	–	–	2	2	2	2	
CO4	3	2	3	2	2	–	–	1	1	–	2	2	1	1	
CO5	2	3	3	2	2	1	–	1	1	1	2	3	1	1	
Course Contents / Syllabus															
Unit 1	Introduction to IoT Security														08 hours
Introduction to IoT Security, Importance of security in IoT, Basic security concepts (Confidentiality, Integrity, Availability) , Common threats in IoT devices, IoT Security Architecture, Examples of IoT security issues															
Unit 2	IoT Devices and Security														10 hours
Introduction to firmware and updates, Risks in connected devices, Attack types (Device-based attacks, Sensor attacks, , Firmware attacks), Device-level security basics, Lightweight Cryptography (AES, ECC, SHA)															
Unit 3	Network and Communication Security														09 hours
Basics of IoT communication (WiFi, Bluetooth), Secure data transmission (encryption basics), Common network attacks (e.g., sniffing, spoofing), Introduction to secure communication protocols, BLE Security															
Unit 4	Authentication, Authorization & Trust Management														10 hours
Authentication Techniques (Password-based, Token-based, Certificate-based, Biometric-based, Authorization), Role-Based Access Control (RBAC) , Attribute-Based Access Control (ABAC), Device Identity Management, Secure Device Provisioning Trust Management in IoT															
Unit 5	Applications and Case Studies														08 hours
Vulnerability Scanning of IoT Device, Privacy-preserving Sensor Data Transmission, Mirai Botnet Attack (2016), Itron Smart Infrastructure Cyberattack (2026), Hikvision Camera Vulnerability Exploitation (2025)															
Total Lecture Hours														45 hours	

Textbook:	
1. Arshdeep Bahga, Vijay Madiseti, Internet of Things A Hands on Approach, Universities Press, 2015 Rap Payne, <i>Beginning App Development with Flutter</i> , Apress, 2019 2. Rajkumar Buyya, Amir Vahid Dastjerdi, Internet of Things Principles and Paradigms, Morgan Kaufmann, 2016 3. Security and Privacy in Internet of Things Iots: Models, Algorithms, and Implementations, 2016	
Reference Books:	
1. Peter Waher, Learning Internet of Things, Packt Publishing, 2015 2. Eran Boudjnah, <i>Clean Architecture for Android</i> , BPB Publications, 2022 3. Deepak Gupta, Fundamentals of IoT and Wearable Technology Design, CRC Press, 2021 4. Fei Hu, Security and Privacy in Internet of Things IoTs Models Algorithms and Implementations, CRC Press, 2016	
Self-Study Content Links:	Number of hours
MOOCs https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01330133520384819210095_shared/overview	20
Video Lectures 1. https://nptel.ac.in/courses/106105166 2. https://www.youtube.com/watch?v=dIQoXrAq8IM	15
Web References 1. https://www.ibm.com/think/topics 2. https://www.hivemq.com/blog/mqtt-essentials-part-1-introducing-mqtt/	10

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:															
Demonstrate effective verbal and non-verbal communication skills in diverse professional settings, including meetings, presentations, and interpersonal interactions. Develop and apply clear, concise, and audience-appropriate written communication, such as emails, letters, memos, resume’, using correct grammar, tone, and format. Adapt communication style based on cultural, organizational, and situational contexts to foster inclusive and respectful professional relationships. 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 Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3	PSO4
CO1	1	1	1	1	1	2	1	2	3	1	1	-	-	-	-
CO2	1	1	1	1	1	1	1	2	3	1	1	1	-	1	1
CO3	1	1	1	1	1	1	1	2	3	1	1	2	1	2	2
CO4	1	1	1	1	1	1	1	2	3	1	1	1	1	2	2
CO5	1	1	1	1	1	1	1	2	3	1	1	1	1	1	2
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, 4 th Edition , Oxford Univ. Press, 2022, New Delhi.	Meenakshi Raman & Sangeeta Sharma
Reference Books:		
S.NO	Books Title	Author
1	Technical Communication, 15 th Edition , Pearson, 2021	John M. Lannon & Laura J. Gurak
2	Spoken English- A Manual of Speech and Phonetics (5 th 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
Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://onlinecourses.nptel.ac.in/noc24_ge37/preview https://archive.nptel.ac.in/courses/109/106/109106094/	4 weeks
Video Lectures	https://www.youtube.com/watch?v=kOJlwMJxEG0&t=8s https://www.youtube.com/watch?v=Sg7Q_dC_fWU&list=PLPuC5CMHiqmuzq_KQ4aw0V9Q7xJY6aezb https://www.youtube.com/watch?v=ymLFJDpjcCk&list=PLPuC5CMHiqmuzq_KQ4aw0V9Q7xJY6aezb&index=6	2.5 hours
Web References	Technical Communication And Soft Skills For Engineers	



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



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)/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, and relational database concepts.															
Course Objectives:To familiarize the students with the basics of database design and implementation.															
Course Outcome: After completion of the course, the student will be able to												Bloom’s Knowledge Level (KL)			
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)															
COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO 11	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	3	3	3	2	1	-	1	2	1	2	3	1	2	1
CO2	3	3	3	3	2	2	-	2	1	2	2	3	3	2	1
CO3	2	2	2	2	3	2	-	-	1	2	2	3	3	2	2
List Of Practical’s (Indicative & Not Limited To)														CO	
S.No	Experiment													CO Mapping	
1	Understand and implement 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 above chosen case study (Program-2). Identify the suitable entities and find the relationship and cardinalities between the entities with their relevant attributes.													CO1	
4	Implement DDL, DML, DCL, and 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	Practicing Queries using Like, Between, Aliases, distinct Operator & Predicate and Implement Aggregate functions.													CO2	
8	Implementation of Queries using Where, Group by, Having and Order by Clause.													CO2	

9	Create a table EMPLOYEE with following schema:-(Emp_no, E_name, E_address, E_ph_no, Dept_no, Dept_name, Job_id, Designation, Salary) Write SQL statements for the following query. - List the E_no, E name, Salary of all employees working for MANAGER. - Display all the details of the employee whose salary is more than the Sal of any IT PROFF. - List the employees in the ascending order of Designations of those joined after 1981. - List the employees along with their Experience and daily. - List the employee who are either 'CLERK' or 'ANALYST'. - List the employees who joined on 1-MAY-81, 3-DEC-81, 17-DEC-81. - List the e_name those are starting with 'S'. - Display the total salary spent for each job category. - Display lowest paid employee details under each manager. - Display number of employees working in each department and their department name. - Display the details of employees sorting the salary in increasing order. - Show the record of employee earning salary greater than 16000 in each department. - Add constraints to check, while entering the empno value (i.e) empno > 100. - Define the field DEPTNO as unique. Create a primary key constraint for the colum(EMPNO).	CO2
10	Implementation of Queries using set theory operators UNION, INTERSECT, MINUS.	CO2
11	Implementation of Queries using Inner Join and Outer Join along with their types.	CO2
12	Implementation of Queries, nested Queries or Sub Queries: IN, NOT IN, Exists, Not Exists, All and Any.	CO2
13	Apply the set theory operators, joins and nested queries on the chosen above case-study (Program-2)	CO2
14	Understand and implement Database Connectivity with any programming language.	CO2
15	Implementation of Indexing, Views, and sequence.	CO2
16	- Write a PL/SQL Program to Add Two Numbers - Write PL/SQL Program for Fibonacci Series - Write PL/SQL Program to Find Greatest of Three Numbers	CO2
17	Write a PL/SQL code block to calculate the area of a circle for a value of radius varying from 3 to 7. Store the radius and the corresponding values of calculated area in an empty table named Areas, consisting of two columns Radius and Area.	CO2
18	Write a PL/SQL code block that will accept an account number from the user, check if the users' balance is less than the minimum balance, only then deduct Rs.100/- from the balance.	CO2

19	Create a row level trigger for the customers table that will fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old values and new values.	CO2	
20	Implementation of commit and rollback statement with amount transfer example.	CO2	
21	Study of Open-Source NOSQL Database and installation of MongoDB.	CO3	
22	Implementation of the MongoDB Shell commands.	CO3	
23	Implementation of the CRUD Operations in MongoDB.	CO3	
24	Implementation of Aggregate Functions in MongoDB.	CO3	
Total Hours: 60 hrs.			



LAB Course Code: CCSIoT0401	LAB Course Name: Mobile Application Development Lab	L	T	P	C
-----------------------------	---	---	---	---	---

Course Offered in: CSE(IoT)	0	0	2	1
-----------------------------	---	---	---	---

Pre-requisite: Basic programming (Java/Dart), OOP, XML, DB concepts.

Course Objectives:

1. Learn mobile app development (Android & Flutter)
2. Understand architecture and UI design
3. Develop and deploy mobile applications

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

CO1	Develop Android applications using UI components and layouts.	K5
CO2	Create Android applications integrating SMS, GPS, notifications, and basic functionality	K6
CO3	Design Flutter applications with Firebase integration, widgets, and animations.	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	3	2	2	1	3	–	–	1	1	–	1	3	1	–
CO2	3	3	3	2	3	1	1	1	1	1	2	3	2	1
CO3	3	3	3	2	3	1	1	2	1	1	2	3	2	2

List Of Practical's

	CO
1. - Install, configure, and verify the proper functioning of Android Studio and the required SDK tools for Android application development. - Develop a basic application to display “Hello World” on the screen. - Develop an application to implement TextView, EditText, Button, ImageButton, and ToggleButton components.	CO1
2. - Develop an application to demonstrate the use of Linear Layout and Absolute Layout. - Develop an application to demonstrate the functionality and arrangement of UI components using Frame, Table, and Relative layouts. - Design and implement a program demonstrating the use of CheckBox, RadioButton, and RadioGroup in Android.	CO1
3. - Design and implement a program demonstrating the use of a Progress Bar component in Android. - Design and implement a program demonstrating the functionality of Date Picker and Time Picker controls in Android. - Design and implement a program demonstrating a customized Toast notification in Android.	CO2
4. - Create an application that enables SMS communication by sending and receiving messages. - Create a native application that performs arithmetic calculations such as addition, subtraction, multiplication, and division. - Create an application that enables registration of student information through a user interface.	CO2
5. - Design and implement a Flutter application demonstrating the use of core widgets including Scaffold, Carousel Slider, Staggered Grid View, Tabs with TabBarView, and Horizontal ListView. - Design and implement a user registration system using Flutter integrated with Firebase Firestore. - Develop a Flutter application to demonstrate implicit animation using AnimatedContainer with tap gesture	CO3

interaction. d. Display live temperature data from IoT sensor in flutter application.	
Total Hours: 30 hrs.	



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



Course Code: CCSE0455	Course Name: Web Technology	L	T	P	C	
------------------------------	------------------------------------	----------	----------	----------	----------	--

Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AI ML)/CSE(IOT)/CSE(AI)/CSE(DS)	0	0	6	3
---	----------	----------	----------	----------

Pre-requisite: Basic Programming Knowledge, Knowledge of any programming language (e.g., C, C++, Python/Java), 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: After completion of the course, the student will be able to

S.No	Course Outcome	Bloom' s Level
CO 1	Identify the basic facts and explaining the basic ideas of Web technology and internet.	K2
CO 2	Applying and creating various HTML5 semantic elements and application with working on HTML forms for user input.	K6
CO 3	Understanding and applying the concepts of Creating Style Sheet CSS3 and bootstrap.	K3
CO 4	Analysing and implementing concept of JavaScript and its applications.	K6
CO 5	Creating and evaluating dynamic web pages using the concept of PHP.	K6

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

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO1 1	PSO1	PSO2	PS O3
CO1	3	3	2	-	2	-	-	-	2	2	-	3	1	1
CO2	3	2	3	2	3	-	-	-	2	1	2	1	2	3
CO3	3	2	3	-	3	-	-	-	2	2	2	2	3	2
CO4	3	3	3	2	3	-	1	-	2	2	2	3	2	3
CO5	3	3	3	2	3	-	-	-	2	2	2	2	1	2

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 to ols, 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, XQuery, 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 selector) , 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, Copying, 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	Book Details	Author
1	“Web Technology and Design”, 1st 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	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

Self-Study Content Links:	Duration (in Hrs)
----------------------------------	--------------------------

MOOCs	https://onlinecourses.swayam2.ac.in/e-learning/preview/nou24_cs09	12 Weeks
--------------	---	-----------------

Video Lectures	https://youtu.be/96xF9phMsWA https://youtu.be/Zopo5C79m2k https://youtu.be/ZliIs7jHi1s https://youtu.be/htbY9-yggB0 https://youtu.be/vHmUVQKXIVo https://youtu.be/qz0aGYrrlhU https://youtu.be/a8W952NBZUE https://youtu.be/BsDoLVMnmZs https://youtu.be/1Rs2ND1ryYc https://youtu.be/vpAJ0s5S2t0 https://youtu.be/GBOK1-nvdU4 https://youtu.be/Eu7G0jV0ImY https://youtu.be/-qfEOE4vtxE https://youtu.be/PkZNo7MFNFg https://youtu.be/W6NZfCO5SIk https://youtu.be/DqaTKBU9TZk https://youtu.be/_GMEqhUyyFM https://youtu.be/ImtZ5yENzgE https://youtu.be/xIApzP4mWyA https://youtu.be/qKR5V9rdht0	17m 8.50m
Web References	Web Development Technologies - GeeksforGeeks	-



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



Course Code: CNC0401						Course Name: Artificial Intelligence and Cyber Ethics						L	T	P	C
Course Offered in: All B.tech 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			
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	-	1	-	-		1	2	-	-	-	2	1	1	1	
CO2	2	3	2	-	2	1	2	-	2	1	2	2	1	1	
CO3	2	3	2	1	2	3	3	-	2	2	2	2	2	1	
CO4	2	2	-	-	1	3	3	-	2	2	2	2	1	1	
Course Contents / Syllabus															
Unit 1	An Overview to AI Ethics												6 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 Cyberlaws, Legal frameworks.															
Unit 2	Fairness and Favoritism in Machine Learning												8 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.															
Unit 3	AI Ethics and Cybersecurity Principles												8 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.															
Unit 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).															
Total Lecture Hours													30 hours		
Textbook:															
S.NO	Book Title										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 Title	Author
1.	Artificial Intelligence and Ethics , BPB Publications, 2023.	S. B. Kishor, Debajit Biswas
2.	Cyber Security and Cyber Laws ,Cengage India, 2022.	Alfred Basta, Nadine Basta, Sattwik Panda,
Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_INTEL_02	64 hours
Video Lectures	https://www.youtube.com/watch?v=hVJqHggF59A https://www.youtube.com/watch?v=VqFqWIqOB1g https://www.youtube.com/watch?v=O5RX_T4Tg24 https://www.youtube.com/watch?v=RJZ0pxcZsSQ	5.2m 6.52m .45m 5.02m
Web References	CCS345 Ethics and AI Lecture Notes PDF Artificial Intelligence Intelligence (AI) & Semantics	-



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



Course Code: CCSAI0411				Course Name: DATA ANALYTICS								L	T	P	C
Course Offered in: CSE/CS/IT/CSE(AI)/CSE(AI ML)/CSE(IOT)/CSE(AI)/CSE(DS)/CSE-R/M.Tech int												3	0	0	3
Pre-requisite: Basic Knowledge of Statistics and Probability															
Course Objectives:															
The objective of this course is to understand the fundamental concepts of Data analytics and learn about various types of data formats and their manipulations. It helps students to learn exploratory data analysis and visualization techniques in addition to R/Python/Tableau programming language.															
Course Outcome												Bloom’s Knowledge Level (KL)			
CO1		Understand the fundamental concepts of data analytics in the areas that plays major role within the realm of data science.										K1			
CO2		Explain and exemplify the most common forms of data and its representations.										K2			
CO3		Understand and apply data pre-processing techniques.										K3			
CO4		Analyze data using exploratory data analysis.										K4			
CO5		Illustrate various visualization methods for different types of data sets and application scenarios.										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	-	-		-	2	-	3	1	2	
CO2	2	3	-	1	2	-	-	1	-	1	-	2	1	1	
CO3	3	3	2	2	3	-	-		-	2	-	3	2	2	
CO4	2	3	2	3	3	-	-	1	-	2	1	3	2	2	
CO5	2	2	2	2	3	-	-	2	-	1	1	3	2	2	

Course Contents / Syllabus		
UNIT 1	INTRODUCTION TO DATA SCIENCE	8 hours
Introduction to Data Science, Big Data, the 5 V's, Evolution of Data Science, Datafication, Skillsets needed, Data Science Lifecycle, types of Data Analysis, Data Science Tools and technologies, Need for Data Science, Analysis Vs Analytics Vs Reporting, Big Data Ecosystem, Future of Data Science, Applications of Data Science in various fields, Use cases of Data science-Facebook, Netflix, Amazon, Uber, AirBnB.		
UNIT 2	DATA HANDLING	9 hours
Types of Data: structured, semi-structured, unstructured data, Numeric, Categorical, Graphical, High Dimensional Data, Transactional Data, Spatial Data, Social Network Data, standard datasets, Data Classification, Sources of Data, Data manipulation in various formats, for example, CSV file, pdf file, XML file, HTML file, text file, JSON, image files etc. import and export data in R/Python		
UNIT 3	DATA PRE-PROCESSING	9 hours
Form of Data Pre-processing, data Attribute and its types, understanding and extracting useful variables, KDD process, Data Cleaning: Missing Values, Noisy Data, Discretization and Concept hierarchy generation (Binning, Clustering, Histogram), Inconsistent Data, Data Integration and Transformation. Data Reduction: Data Cube Aggregation, Data Compression, Numerosity Reduction.		
UNIT 4	EXPLORATORY DATA ANALYSIS	9 hours
Handling Missing data, Removing Redundant variables, variable Selection, identifying outliers, Removing Outliers, Time series Analysis, Data transformation and dimensionality reduction techniques such as Principal Component Analysis (PCA), Factor Analysis (FA) and Linear Discriminant Analysis (LDA), Univariate and Multivariate Exploratory Data Analysis. Data Munging, Data Wrangling- APIs and other tools for scrapping data from the web/ internet using R/Python.		
UNIT 5	DATA VISUALIZATION	10 hours
Introductions and overview, Debug and troubleshoot installation and configuration of the Tableau. Creating Your First visualization: Getting started with Tableau Software, Using Data file formats, connecting your Data to Tableau, creating basic charts (line, bar charts, Tree maps), Using the Show me panel. Tableau Calculations: Overview of SUM, AVG, and Aggregate features Creating custom calculations and fields, Applying new data calculations to your visualization. Manipulating Data in Tableau: Cleaning-up the data with the Data Interpreter, structuring your data, Sorting, and filtering Tableau data, Pivoting Tableau data. Advanced Visualization Tools: Using Filters, Using the Detail panel Using the Size panels, customizing filters, Using and Customizing tooltips, formatting your data with colours, Creating Dashboards & Stories, Distributing & Publishing Your Visualization		
Total: 45 Hours		

TEXTBOOK		
BOOK TITLE		AUTHOR
<i>Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining.</i> 2nd Edition, John Wiley & Sons, 2014.		Myatt, G. J., & Johnson, W. P.
<i>Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data.</i> John Wiley & Sons, 2015.		Dietrich, D., Heller, B., & Yang, B
REFERENCE BOOKS:		
BOOK TITLE		AUTHOR
<i>An Introduction to Statistical Learning with Applications in Python.</i> Springer, 2023.		James, G., Witten, D., Hastie, T., & Tibshirani, R.
<i>R for Data Science.</i> 2nd Edition, O'Reilly, 2023.		Wickham, H., & Grolemund, G.
SELF-STUDY CONTENT LINKS:		DURATION (IN HRS)
MOOCS	https://www.netacad.com/courses/dataanalyticsessentials?courseLang=en-US	30Hrs
VIDEO LECTURES	https://www.youtube.com/watch?v=r-uOLxNrNk8	12Hrs
WEB REFERENCES	https://jakevdp.github.io/PythonDataScienceHandbook/	3Hrs



NIET
Greater Noida
Autonomous Institute

Noida Institute of Engineering & Technology, Greater Noida

नोएडा अभियांत्रिकी एवं प्रौद्योगिकी संस्थान, ग्रेटर नोएडा

(An Autonomous Institute)

Approved by AICTE, Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow Uttar Pradesh, (Formerly Uttar Pradesh Technical University)



Course Code: CCSCY0411				Course Name: INTRODUCTION TO 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

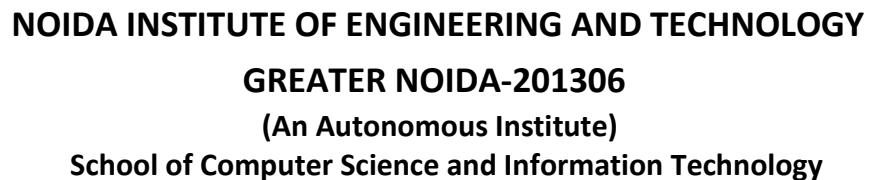
Cybersecurity: Principles and Practice (5th Edition)		William Stallings
Computer Security Fundamentals (4th Edition)		Chuck Easttom
REFERENCE BOOKS:		
BOOK TITLE		AUTHOR
Fundamentals of Cyber Security (1st Edition)		Kutub Thakur, Al-Sakib Khan Pathan
Cyber Security (1st Edition)		Nina Godbole, Sunit Belapure
SELF-STUDY CONTENT LINKS:		DURATION (IN HRS)
MOOCS	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0136382610228674561182_shared/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0130752807862845445_shared/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_3388902307073574000_shared/overview https://infyspringboard.onwingspan.com/web/en/page/lex_auth_014234794236551168189	1h 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=PLm3g0kXO2llm8Qd6EpkQc2gUiCz-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(AI ML)/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, Kindle Edition.		William Jordon,
REFERENCE 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=PLsyebzWxl7poL9JTVyndKe62ieoN-MZ3	1 HRS 03 MIN
	https://youtu.be/9MmC_uGjBsM?list=PL3pGy4HtqwD02GVgM96-V0sq4_DSinqvf	27 MIN 29 SEC
WEB REFERENCES	https://www.geeksforgeeks.org/python/python-web-development-django/	



CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)																
	CO-PO Mapping	PO1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO9	PO10	PO11	PSO 1	PSO 2	PSO3	PSO4
	CO1	3	3	3	2	2	1	2	2	-	-	2	3	3	2	2
	CO2	3	3	2	2	2	-	2	-	-	-	2	2	2	2	2
	CO3	3	3	2	2	3	1	2	2	-	-	3	3	3	2	2
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	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

<pre> * ** *** **** ***** </pre>	<pre> * ** * * * * ***** </pre>	<pre> * *** ***** ***** ***** </pre>	<pre> * * * * * * * ***** </pre>
Right Triangle Star Pattern	Hollow Right Triangle Star Pattern	Pyramid Star Pattern	Hollow Pyramid Star Pattern
<pre> * ** *** **** ***** </pre>	<pre> * ** * * * * ***** </pre>	<pre> ***** ***** ***** *** * </pre>	<pre> ***** * * * * * * * </pre>
Mirrored Right Triangle Star Pattern	Hollow Mirrored Right Triangle Star Pattern	Inverted Pyramid Star Pattern	Hollow Inverted Pyramid Star Pattern

<pre> * *** xxxxx yyyyyyy xxxxxxxxx xxxxxxx *** * </pre>	<pre> ***** **** *** ** * * ** *** **** ***** </pre>	<pre> 5 5 5 5 5 5 5 5 5 5 4 4 4 4 4 4 4 5 5 4 3 3 3 3 3 4 5 5 4 3 2 2 2 3 4 5 5 4 3 2 1 2 3 4 5 5 4 3 2 2 2 3 4 5 5 4 3 3 3 3 3 4 5 5 4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 </pre>	<pre> * ** *** ** * ** *** ** * </pre>
Diamond Star Pattern	Hollow Diamond Star Pattern	Number pattern 18	For N=3 print above pattern
<pre> xxxxx xxxx xxx xx x xx xxx xxxx xxxxx </pre>	<pre> xxxxx xxxx xxx xx x xx xxx xxxx xxxxx </pre>	<pre> ABCDEFEDCBA ABCDE EDCBA ABCD DCBA ABC CBA AB BA A A </pre>	
Right Arrow Star Pattern	Left Arrow Star Pattern		

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

- 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.
- 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

- 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).

3. For a given string, generate a pattern based on the following rules:

Input: A string of characters (e.g., "HAT").

Output: Generate patterns by replacing characters with the numeric value 1 and process the patterns as described below:

1. Replace one character at a time with 1:

- For each character in the string, replace it with 1, keeping the other characters unchanged.
- Example for "HAT": 1AT, H1T, HA1

2. Replace two characters at a time with 1:

- Replace every combination of two characters with 1, keeping the remaining character unchanged.
- If 1s are consecutive, replace them with their sum (e.g., 11T becomes 2T).
- Example for "HAT":
11T → 2T, H11 → H2, 1A1

3. Replace all characters with 1:

- Replace all characters in the string with 1.
- If there are consecutive 1s, sum them up (e.g., 111 becomes 3).
- Example for "HAT": 111 → 3

Final Output

For the string "HAT", the output should be: 1AT, H1T, HA1, 2T, H2, 1A1, 3.

4. Given a sorted array arr [] and a target value, the task is to count triplets (i, j, k) of valid indices, such that $arr[i] + arr[j] + arr[k] = \text{target}$ and $i < j < k$.

Examples:

Input: arr[] = [-3, -1, -1, 0, 1, 2], target = -2

Output: 4

You are given an array prices[] where prices[i] represents the price of a given stock on day i. You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock. Write a program to return the maximum profit you can achieve from this transaction. If no profit is possible, return 0.

6. Find the "Kth" max and min element of an array:

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 array with values 0, 1, and 2 using constant space and one pass
Input: [0, 2, 1, 2, 0]
Output: [0, 0, 1, 2, 2]

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)

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".

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} Output: 6054854654

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

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 = "aabcbbcbcabcc"`, `k = 3`

Output: 27

Expected Time Complexity: $O(n+k\log(p))$

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

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:

Input:

board [][] = [['C', 'A', 'T'], ['R', 'A', 'K'], ['T', 'O', 'N']],

target = "CART" **Output:** true

Explanation:

You can trace the word "CART" through the path: C → A → R → T (moving horizontally and vertically, without repeating cells).

15. Given an encoded string **s**, the task is to decode it. The encoding rule is:

- **k[encodedString]**, where the **encodedString** inside the square brackets is being repeated exactly **k** times. Note that **k** is guaranteed to be a positive integer, and encodedString contains only

Note: The test cases are generated so that the length of the output string will never exceed 10^5 .

Examples:

Input: s = "1[b]"

Output: "b"

Input: s = "3[b2[ca]]"

Output: "bcacabcacabcaca"

**Competitive coding list will be shared with the students.*