NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA, G.B. NAGAR

(AN AUTONOMOUS INSTITUTE)

Affiliated to

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

Evaluation Scheme & Syllabus

For

Bachelor of Technology

Computer Science (CS)

Second Year

(Effective from the Session: 2025-26)

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA

(AN AUTONOMOUS INSTITUTE)

Bachelor of Technology

Computer Science

EVALUATION SCHEME

SEMESTER-III

Sl. No.	Subject Codes	Subject	Types of Subjects	Periods		Evaluation Schemes					End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	BCSCC0301	Employability Skill Development - I	Mandatory	2	0	0	60	40	100				100	2
2	BAS0303N	Statistics and Probability	Mandatory	3	1	0	30	20	50		100		150	4
3	BCSE0303A	Operating Systems	Mandatory	2	0	0	30	20	50		50		100	2
4	BCSE0301	Data Structures and Algorithms-I	Mandatory	3	0	0	30	20	50		100		150	3
5	BCS0301N	Foundation of Cloud Computing	Mandatory	2	0	0	30	20	50		50		100	2
6	BCSAI0304	High Performance Computing	Mandatory	3	0	0	30	20	50		100		150	3
7	BCSE0353A	Operating Systems Lab	Mandatory	0	0	4				50		50	100	2
8	BCSE0351	Data Structures and Algorithms-I Lab	Mandatory	0	0	4				50		50	100	2
9	BCS0351	Foundation of Cloud Computing Lab	Mandatory	0	0	2				25		25	50	1
10	BCSE0352	Object Oriented Techniques using Java	Mandatory	0	0	6				50		100	150	3
11	BCSE0359X	Social Internship	Mandatory	0	0	2				50			50	1
12	BNC0302/ BNC0301	Environmental Science / Artificial Intelligence and Cyber Ethics	Compulsory Audit	2	0	0	30	20	50				50	NA
		Massive Open Online Courses (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL		**17**	**1**	**18**	**210**	**140**	**350**	**225**	**400**	**225**	**1200**	**25**

*** List of MOOCs Based Recommended Courses for Second year (Semester-III) B. Tech Students**

Sr. No.	Subject Code	Course Name	University / Industry Partner Name	No of Hours	Credits
1	BMC0040	Data Structures and Algorithms using Python - Part 2	Infosys Wingspan (Infosys Springboard)	37h 41m	3
2	BMC0008	Object Oriented Programming Using Python	Infosys Wingspan (Infosys Springboard)	46h 13m	3.5

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 - BNC0301/BNC0302)**
  - 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, GAUTAM BUDDH NAGAR
(AN AUTONOMOUS INSTITUTE)

Bachelor of Technology
Computer Science

Evaluation Scheme
SEMESTER-IV

Sl. No.	Subject Codes	Subject	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	BASCC0401	Employability Skill Development - II	Mandatory	2	0	0	60	40	100				100	2
2	BCSE0402	Database Management Systems	Mandatory	3	0	0	30	20	50		100		150	3
3	BCSE0401	Data Structures and Algorithms-II	Mandatory	3	0	0	30	20	50		100		150	3
4	BCS0402	Big Data Analytics	Mandatory	3	0	0	30	20	50		100		150	3
5		Department Elective - I	Departmental Elective	3	0	0	30	20	50		100		150	3
6	BASL0401N	Technical Communication	Mandatory	2	0	0	30	20	50		50		100	2
7	BCSE0452Z	Database Management Systems Lab	Mandatory	0	0	4				50		50	100	2
8	BCSE0451	Data Structures and Algorithms-II Lab	Mandatory	0	0	2				25		25	50	1
9	BCSE0455	Web Technologies	Mandatory	0	0	6				50		100	150	3
10	BCSE0459	Mini Project	Mandatory	0	0	2				50			50	1
11	BCSCC0452	Problem Solving Approaches	Mandatory	0	0	2				50			50	1
12	BNC0401/ BNC0402	Artificial Intelligence and Cyber Ethics/ Environmental Science	Compulsory Audit	2	0	0	30	20	50				50	NA
		Massive Open Online Courses (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL		**18**	**0**	**16**	**210**	**140**	**350**	**225**	**450**	**175**	**1200**	**24**

*** List of MOOCs Based Recommended Courses for Second year (Semester-IV) B. Tech Students**

S. No.	Subject Code	Course Name	University / Industry Partner Name	No of Hours	Credits
1	BMC0061	Database Management System - Science Graduates	Infosys Wingspan (Infosys Springboard)	55h 23m	4
2	BMC0091	Master Network Automation with Python for Network Engineers	Infosys Wingspan (Infosys Springboard)	23h 24m	1.5

PLEASE NOTE: -

- **Compulsory Audit (CA) Courses (Non-Credit - BNC0401/BNC0402)**
  - 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.

DEPARTMENTAL ELECTIVES

Subject Code	Subject Name	Type	Sem	Branch	Bucket
BCSAI0411	Data Analytics	Department Elective-I	IV	CS	AI Driven Analytics
BCSAI0412	CRM:Fundamentals	Department Elective-I	IV	CS	CRM-RPA
BCSCY0411	Fundamentals of Cyber Security	Department Elective-I	IV	CS	Cyber Security- I
BCSE0411	Python web development with Django	Department Elective-I	IV	CS	Full stack Development

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA, GAUTAM BUDDH NAGAR
(AN AUTONOMOUS INSTITUTE)

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

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

For registration to MOOCs Courses, the students shall follow Coursera 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.

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSCC0301	Course name : Employability Skill Development – I	L	T	P	C
Course Offered in: III Semester		**2**	**0**	**0**	**2**

Pre-requisite: Programming Language C

Course Objectives: This course introduces the fundamentals of computer systems, basic mathematics for computing, and software development principles. It emphasizes algorithm design and C++ programming skills. Through hands-on practice and project-based learning, students develop problem-solving abilities and teamwork while creating real-world applications, mini-games, and simulations, enhancing both technical and collaborative competencies

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply sets, relations, functions to computational problem-solving	K3
CO2	Understand and implement the steps in the software development life cycle using logical reasoning and flowcharts.	K3
CO3	Design and develop small-scale software projects or games using structured programming and project-based approaches.	K6
CO4	Collaborate in teams to plan, develop, and present a complete software project, demonstrating problem-solving and communication skills.	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	2	-	-	-	2	-	-	-	2	-	1
CO2	3	3	3	2	-	-	-	2	-	-	-	-	3	2
CO3	3	3	3	2	-	-	-	2	-	-	-	-	-	3
CO4	3	3	3	3	-	-	-	2	-	-	-	-	-	-

Course Contents / Syllabus

Module 1	**Foundations of Computer Systems and Mathematical Concepts**	**4 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.

Module 2	**Software Development Fundamentals**	**6 hours**

Introduction to Software Development Life Cycle, Step-by-step solution to simple problems, Developing logic/flowchart/pseudocode, simple games, puzzles, Step-wise refinement and Procedural Abstraction

Module 3	**Project-Based Learning**	**10 hours**

Introduction to the basics of C++, Implementation of control structures through practical tasks such as creating a number guessing game using loops and conditions, Functions and scope demonstrated by developing a menu-driven applications using user-defined functions, implement simple logic-based games including puzzles, tic-tac-toe, Hangman etc., the concept of pointers and dynamic memory allocation is introduced by creating a dynamic leader board to store player scores. File handling in C++ to save high scores or game states to external files.		
Module 4	**Project/Game Development**	**10 hours**
Project Planning & Development (Teams, roles, idea pitching, develop C++ game or simulation), Mini Project, Project Demonstration and Review		
		Total Lectures : 30 hours

Reference Books:

S.No	**Book Title**
1	Bajarne Stroustrap ,Programming: Principles and Practice Using C++, 2nd Edition
2	Scott Meyers , Effective Modern C++, Shroff/O'Reilly

NPTEL/ YouTube/ Faculty Video Link:

1	
2	

Mode of Evaluation

CIE							**ESE**	
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**TA3**	**Attendance**		
			10	**10**	**10**	**10**		
60			**40**					

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BAS0303N	Course Name: Statistics and Probability	L	T	P	C
Course Offered in: B.Tech. Second Year Sem-III/IV AI/AIML/AI(TWIN)/AIML(TWIN)/CYS/DS/CS/CSE/CSE-R/IT/M.Tech(Int.)/IT (TWIN)/CSE(TWIN)		3	1	0	4

Pre-requisite: B.Tech. Ist Year Syllabus

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

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

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

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

Course Contents / Syllabus

Module 1	**Statistical Techniques-I**	**6 hours**

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

Module 2	**Statistical Techniques-II**	**10 hours**

Curve Fitting, Method of least squares, fitting of straight lines, Fitting of second-degree parabola, Exponential curves, Correlation and Rank correlation, Linear regression, nonlinear regression and multiple linear regression.

Module 3	**Probability and Random Variable**	**10 hours**

Random Variable: Definition of a Random Variable, Discrete Random Variable, Continuous Random Variable, Probability mass function, Probability Density Function, Distribution functions.

Multiple Random Variables: Joint density and distribution Function, Properties of Joint Distribution function, Marginal density Functions, Conditional Distribution and Density, Statistical Independence, Central Limit Theorem (Proof not expected).

Module 4	**Expectations and Probability Distribution**	**10 hours**

Expectations of single Random Variable, Mean, Variance, Moment Generating Function, Binomial, Poisson, Normal, Exponential distribution

Module 5	**Hypothesis Tests and Control Charts**	**12 hours**

Testing a Hypothesis, Null hypothesis, Alternative hypothesis, Level of significance, Confidence limits, Test of significance of difference of means, Z-test, t-test and Chi-square test, F-test, One way ANOVA.
Statistical Quality Control (SQC), Control Charts, Control Charts for variables (Mean and Range Charts), Control Charts for Variables (p, np and C charts).

Total Lecture Hours	**48 hours**

Textbook:

S.No	**Book Title**	**Author**
1	Textbook of Engineering Mathematics- IV	Bali, N.P.
2	Advanced engineering mathematics	Jain, R.K.
3	Higher engineering mathematics	Grewal, B.S.
4	Statistical methods	Gupta, S.P.
5	Advanced engineering mathematics	ZILL, DENNIS G.

Reference Books:

.

S.No	**Book Title**	**Author**
1	Introduction to Probability Models	Ross, Sheldon M
2	Probability, Random Variables and Stochastic Processes	Papoulis, Athanasios
3	Advanced engineering mathematics	Kreyszig, E.

NPTEL/ Youtube/ Faculty Video Link:

Module 1	https://archive.nptel.ac.in/courses/110/107/110107114/
Module 2	https://archive.nptel.ac.in/courses/111/105/111105042/
Module 3	https://archive.nptel.ac.in/courses/117/105/117105085/ https://archive.nptel.ac.in/courses/111/104/111104032/
Module 4	http://www.digimat.in/nptel/courses/video/111106112/L19.html https://youtu.be/qvUT68tG_bo?si=40-T46aZ8TmQ-wsG
Module 5	https://archive.nptel.ac.in/courses/103/106/103106120/

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1 5**	**TA2 5**	**TA3 5**	**Attendance 5**		
30			**20**				**100**	**150**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSE0303A	**Course Name: Operating Systems**	**L**	**T**	**P**	**C**
Course Offered in: CSE/CSE-R/IT/CS/AI/AIML/ IOT/DS/CYS		**2**	**0**	**0**	**2**

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

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

	Course Outcome: After completion of the course, the student will be able to	Bloom's Knowledge Level (KL)
CO1	Understand operating system architecture and types, and use the Linux CLI for basic Operations.	K2
CO2	Implement the CPU scheduling algorithms including uses of multithreading models.	K4
CO3	Implement concurrency control, process synchronization techniques, and deadlock handling techniques	K4
CO4	Implement memory management strategies and page replacement algorithms to optimize system performance.	K4
CO5	Analyze file systems and configure distributed systems and virtual machines in modern operating systems.	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	2	1	2					1	1	3	1	1
CO2	3	3	3	2	2					1	1	3	2	2
CO3	3	3	3	2	2					1	1	2	1	1
CO4	3	3	3	2	2					1	1	3	2	2
CO5	3	2	3	2	2					1	2	3	2	2

Course Contents / Syllabus

Unit 1	**Fundamentals & Shell scripting**	**04 hours**

Fundamentals of Operating Systems: Overview of Operating Systems, Generations of OS, Operating System Architecture, Interrupt Handling, System Calls and Kernel, Types of Operating Systems: Batch OS, Multiprogramming OS, Multitasking OS, Multiprocessor OS, Real-time OS.

Shell Scripting in Linux: Introduction to Linux Operating System & Architecture, Basic Command Line Interface (CLI) Operations in Linux, Shell Scripting Basics: Variables, Control Structures, Functions

Applications: Automating system administration tasks using shell scripts in Ubuntu/Linux (e.g., backup scheduling).

Unit 2	**Process & Thread Management**	**08 hours**

Process Management: - Process, Transition Diagram, Process Control Block (PCB), Types of Schedulers: Long Term, Mid Term, Short Term Scheduler.

CPU Scheduling: Pre-emptive and Non-Pre-emptive Algorithm (FCFS, SJF, SRTF, Non-Pre-emptive Priority, Pre-emptive Priority, Round Robin, Multilevel Queue Scheduling and Multilevel Feedback Queue Scheduling) **Thread:** Processes Vs Threads, Thread states, Benefits of threads, Types of threads, Multithread Model, Concept of Hyper-Threading **Applications:** Analyse and implement CPU Scheduling in Real-Time Embedded Systems and RTOS		
Unit 3	**Concurrency and Deadlock Management**	**08 hours**
Concurrency: Introduction of Concurrency, Types of Process, Race Condition, Critical Section, Inter Process Communication, Producer consumer problem. **Process Synchronization**: Lock variable, Peterson's Solution, Strict alternation, Lamport Bakery Solution, Test and set lock, Semaphore- counting, binary and monitor, **Classical Problem of Synchronization:** - Bound Buffer, Dinning Philosopher, Reader writer, Sleeping barber. **Deadlock:** Deadlock, Deadlock characterization, Deadlock Prevention, Deadlock Avoidance: Bankers Algorithms, Deadlock Detection, Recovery from Deadlock. **Applications:** Deadlock avoidance in database transaction management systems like Oracle or MySQL.		
Unit 4	**Memory Management**	**08 hours**
Memory Management: Memory Management function, Loading and linking Address Binding, Memory management techniques, Contiguous technique- Fixed Partitions, variable partitions, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Non-contiguous, Paging, Segmentation, Segmented paging, **Virtual Memory:** Virtual Memory Concepts, Demand Paging, Performance of Demand Paging, Page Replacement Algorithms: FIFO, LRU, Optimal and LFU, Belady's Anomaly, Thrashing **Applications:** Virtual memory management in modern OS like Windows 10 and how paging impacts performance.		
Unit 5	**File Management & Modern Operating System**	**04 hours**
File Management: File Management: Access Mechanism, File Allocation Method, Free Space Management: -Bit Vector, Linked List, **DISK:** Disk Architecture, HDD vs SSD, Disk Scheduling Algorithms **Modern Operating System:** Overview of modern operating system, Modern OS features: Multitasking, virtualization, security, scalability, Shared Memory concepts, Distributed system, Parallel system & its architecture, Virtual machines: hypervisor, Introduction to GPU **Applications:** Large File Storage in a Distributed Manner.		
	Total Lecture Hours	**32 hours**
Textbook:		
1	Abraham Silberschatz, Peter Baer Galvin and Greg Gagne" Operating System Concepts Essentials" , Willey Publication,10th Edition,2018.	
2	Marks G. Sobell "A practical guide to Linux: Commands, Editors and Shell Programming", CreateSpace Independent Publishing Platform, 4th Edition,2017.	
3	Jason Cannon "LINUX for beginners", 1stEdtion,2014	
Reference Books:		
1	William Stallings "Operating Systems: Internals and Design Principles", Pearson Education, 9th Edition, 2019.	
2	Charles Patrick Crowley, "Operating System: A Design-oriented Approach", McGraw Hill Education, 2017.	
3	Ganesh Naik "Learning Linux Shell Scripting", Packt Publishing, 2nd Edition 2018.	
NPTEL/ Youtube/ Faculty Video Link:		

Unit 1	CS162 Lecture 1: What is an Operating System? (youtube.com) Operating System #01 Introduction to OS, its Roles & Types (youtube.com) Operating System #14 What is an Interrupt? Types of Interrupts - YouTube https://www.youtube.com/watch?v=akU1Ji8Vzdk&list=PLbMVogVj5nJRa3VKt_eyZdJ_DitCz1cvQ https://www.youtube.com/watch?v=rRGCGZ6OHw8&list=PLbMVogVj5nJRa3VKt_eyZdJ_DitCz1cvQ&in dex=2
Unit 2	Operating System #03 Programs & Processes, System Calls, OS Structure (youtube.com) Operating System #18 CPU Scheduling: FCFS, SJF, SRTF, Round Robin - YouTube Operating System #19 Priority Scheduling Algorithms, Multilevel Queues - YouTube Operating System #20 Multi Processor Scheduling (youtube.com) Operating System #33 Threads: Thread Model, Thread vs Process, pthread library (youtube.com) Operating System #34 Threads: User level & Kernel level thread, Threading issues (youtube.com) https://www.youtube.com/watch?v=3eG27YUbzyM&list=PLbMVogVj5nJRa3VKt_eyZdJ_DitCz1cvQ&index=3
Unit 3	CS162: Lecture 6: Synchronization 1: Concurrency and Mutual Exclusion (youtube.com) CS162: Lecture 6.5: Concurrency and Mutual Exclusion (Supplemental) (youtube.com) Operating System #04 CPU Sharing, Race Conditions, Synchronization, CPU Scheduling (youtube.com) Operating System #26 Bakery Algorithm - YouTube Operating System #27 Hardware Locks: Spinlock & its Usage (youtube.com) Operating System #31 Deadlocks: Deadlock Detection & Recovery (youtube.com)
Unit 4	Operating System #05 Memory Management: Process, Fragmentation, Deallocation, (youtube.com) Operating System #06 Virtual Memory & Demand Paging in Operating Systems (youtube.com) Operating System #07 MMU Mapping | How Virtual Memory Works? – YouTube
Unit 5	https://www.youtube.com/watch?v=qbQCQ0U6H0o https://www.youtube.com/watch?v=SnKgEuUfV4k https://www.youtube.com/watch?v=cVFyK1f5lDw https://www.youtube.com/watch?v=Z0Vkrn9faoM&list=PLbMVogVj5nJRa3VKt_eyZdJ_DitCz1cvQ&inde x=4 https://www.youtube.com/watch?v=_BtDcroOTSA CUDA Programming Course – High-Performance Computing with GPUs

Mode of Evaluation

CIE						ESE	Total
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**Attendance** **10**		
30			**20**			**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

Course Code: BCSE0301	Course Name: DATA STRUCTURES AND ALGORITHMS-1	L	T	P	C
Course Offered in: CSE/CS/CSR-R/M.TECH(INT) /IT/CSE(AI)/CSE(AIML)/CSE(DS)/CSE(CS)		3	0	0	3

Pre-requisite: The concept of Programming Language.

Course Objective:

The objective of the course is to learn the basic concepts of algorithm analysis, along with the implementation of linear data structure.

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

S. No	Course Outcome	Bloom's Level
CO1	Understand the concept of algorithm analysis and its importance computational for problem solving.	K2
CO2	Implement arrays for searching, sorting, and hashing to foster critical thinking.	K3
CO3	Analyse the performance and structural differences of linked lists with arrays and the implementation of linked list with their applications	K4
CO4	Apply the concept of Stacks and Queues to implement Linear Data Structures and solve real-world computational problems.	K3
CO5	Implement and analyse divide & conquer algorithm and greedy approaches for efficient problem-solving across diverse context.	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	2	3	3	1	1	1	1	1	2	1	-	1
CO2	3	3	2	2	3	1	1	1	1	1	2	3	1	2
CO3	3	3	2	2	3	1	1	1	1	1	2	3	1	2
CO4	3	3	3	2	3	1	1	1	1	1	2	3	1	2
CO5	3	3	3	3	3	1	1	1	1	1	3	3	1	2

Course Contents / Syllabus

Unit 1	Introduction to Data Structure and Algorithms	10 hours

Algorithms: Analysing Algorithms, Complexity of Algorithms, Amortized Analysis, Growth of Functions, Methods of solving Recurrences, Performance Measurements, Time and Space Complexity of an algorithm, Asymptotic notations (Big Oh, Big Theta and Big Omega), Abstract Data Types (ADT).
Data types: Primitive and non-primitive, Introduction to Data structure, Types of Data Structures- Linear & Non-Linear Data Structures.

Unit 2	Design and Analysis of Algorithms: Arrays, searching and sorting, Hashing	9 hours

Arrays: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Derivation of Index Formulae for 1-D, 2-D, 3-D and n-D Array Application of Arrays: Sparse Matrices and their Representations.

Searching algorithm with analysis: Linear search, Binary search. Sorting algorithm with analysis: Bubble sort, Insertion sort, Selection sort, Shell Sort, sorting in Linear Time- Counting Sort.

Hashing: The symbol table, Hashing Functions, Collision-Resolution Techniques, Hashing for direct files.

Unit 3	**Design and Analysis of Algorithms: Linked lists Data Structure**	**10 hours**

Comparison of Array, List and Linked list Types of linked list: Singly Linked List, Doubly Linked List, Circular Linked List Polynomial Representation and Addition of Polynomials.

Unit 4	**Design and Analysis of Algorithms: Stacks Data Structure, Recursion and Queue Data Structure**	**10 hours**

Primitive Stack operations: Push & Pop, Array and Linked List Implementation of Stack, Application of stack: Infix, Prefix, Postfix Expressions and their mutual conversion, Evaluation of postfix expression.

Principles of recursion, Tail recursion, Removal of recursion, Problem solving using iteration and recursion with examples such as binary search, Fibonacci series, and Tower of Hanoi, Trade-offs between iteration and recursion.

Merge sort and Quick sort algorithms with analysis.

Array and linked List implementation of queues, Operations on Queue: Create, Insert, Delete, Full and Empty, Circular queues, Dequeue and Priority Queue algorithms with analysis

Unit 5	**Design and Analysis of Algorithms: Divide and Conquer Algorithm and Greedy Algorithms**	**9 hours**

Divide and Conquer concepts with Examples Such as Quick sort, Merge sort.

Greedy Methods with Examples Such as Activity Selection, Task Scheduling, Fractional Knapsack Problem, Huffman Encoding.

	Total Lecture Hours	**48 hours**

Textbook:

1. Michael T. Goodrich, Roberto Tamassia, "Data Structures and Algorithms in Python: An Indian Adaptation", 1st Edition, 2021.
2. Lipschutz, “Data Structures” Schaum’s Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd, 2nd Edition, 2017.
3. Horowitz and Sahani, “Fundamentals of Data Structures”, Computer Science Press, 1st Edition, 1993.

Reference Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, 4th ed. Cambridge, MA, USA: MIT Press, 2022.
2. N. Karumanchi, Data Structures and Algorithms Made Easy: Data Structure and Algorithmic Puzzles, 5th ed. Noida, India: CareerMonk Publications, 2016.
3. A. Y. Bhargava, Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People, 2nd ed. Shelter Island, NY, USA: Manning Publications, 2024.
4. R. Sedgewick and K. Wayne, Algorithms, 4th ed. Boston, MA, USA: Addison-Wesley, 2011.
5. S. S. Skiena, The Algorithm Design Manual, 2nd ed. London, U.K.: Springer, 2011.

NPTEL/ YouTube/ Faculty Video Link:

Unit 1	https://youtu.be/u5AXxR4GnRY
Unit 2	https://www.youtube.com/watch?v=LQx9E2--p5c&pp=ygUMYXJyYXlzIG5wdGVs
Unit 3	https://www.youtube.com/watch?v=K7VIKlUdo20&pp=ygUPbGluayBsaXN0IG5wdGVs
Unit 4	https://www.youtube.com/watch?v=g1USSZVWDsY&list=PLB3CD0BBB95C1BF09&index=2&pp=iAQB

	https://www.youtube.com/watch?v=THMyk2_p530&pp=ygUccXVldWUgZGF0YSBzdHJ1Y3R1cmUgICBucHRlbA%3D%3D
Unit 5	https://www.youtube.com/watch?v=_VV9v41FIq0&pp=ygUZZGl2aWRlIGFuZCBjb25xdWVyICBucHRlbA%3D%3D https://www.youtube.com/watch?v=ARvQcqJ_-NY&list=PLfFeAJ-vQopt_S5XlayyvDFL_mi2pGJE3

Mode of Evaluation:

CIE						ESE	Total
ST1	**ST2**	**ST3**	**TA1 (5)**	**TA2 (5)**	**Attendance (10)**		
30			**20**			**100**	**150**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCS0301N	**Course Name: Foundation of Cloud Computing**	**L**	**T**	**P**	**C**
Course Offered in: Computer Science		**2**	**0**	**0**	**2**

Pre-requisite: Knowledge of basic computing units

Course Objectives: To introduce students to the core concepts, models, and technologies of cloud computing, enabling foundational understanding of cloud-based services and infrastructure.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Interpret foundational concept of cloud computing and its evolution.	K2
CO2	Compare cloud services and their deployment models.	K4
CO3	Understand architectural principles and standards in cloud design.	K2
CO4	Relate security measures in cloud computing to enable interoperability.	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	**2**	**1**	**1**	**1**	**1**	**2**	**1**	**1**	**-**	**1**	**2**	**1**	1	1
CO2	**2**	**2**	**1**	**2**	**1**	**2**	**1**	**1**	**-**	**2**	**2**	**1**	1	1
CO3	**1**	**2**	**2**	**2**	**1**	**2**	**2**	**1**	**-**	**2**	**2**	**2**	2	1
CO4	**2**	**2**	**3**	**2**	**2**	**3**	**2**	**2**	**1**	**3**	**2**	**2**	2	1

Course Contents / Syllabus

Module 1	**Introduction to Cloud**	**7 hours**

Introduction to Cloud Computing, Definition of Cloud, Evolution of Cloud Computing, Underlying Principles of Parallel and Distributed Computing, Cloud, Characteristics, Multi-tenancy & Elasticity in Cloud, On-demand Provisioning, Cloud economics, Merits of Cloud computing, Obstacles for cloud technology, Cloud vulnerabilities, Cloud challenges.

Module 2	**Cloud Services and Deployment Models**	**8 hours**

Types of Deployment models (Public, Private, Hybrid and community cloud), Service Models in cloud (SaaS, PaaS, IaaS- Storage-as-a-Service), AWS cloud Services (EC2, S3, Elastic File Storage, Elastic Block Storage, Relational Database Services, Virtual Private Cloud, Virtual Machines, CloudWatch), Managed and Unmanaged services.

Module 3	**Cloud Architecture**	**7 hours**

Layered Cloud Architecture Design, NIST Cloud Computing Reference Architecture, Service Level Agreement, Role of governance, IBM-CCRA, Architectural Design Challenge, Open Architecture challenges, Service Oriented Architecture, Web Services, Publish-Subscribe Model, SOAP and REST architecture.

Module 4	**Cloud Security and Resource Management**	**8 hours**

Cloud Security Challenges, Security Governance, IAM, Security Standards, MFA, Authentication and Authorization, CIA, Introduction to Firewall, Security Group, User & Access control.

Inter Cloud Resource Management, Resource Provisioning and Resource Provisioning Methods, Global Exchange of Cloud Resources, Interoperability, Portability, Migration in cloud, Disaster Recovery, Federation in the Cloud, Four Levels of Federation, Federated Services and Applications, Future of Federation

	Total Lecture Hours	**30 hours**

Textbook:

S.No	**Book Title**	**Author**
1.	Fundamentals of Cloud Computing, Nitya Publication, 2020	Dr. Arun Singh Chouhan, Bipin Pandey, Vishwas Srivastava
2.	Cloud Computing Basics: A Non-Technical Introduction, Apress, 2022	Anders Lisdorf

Reference Books:

S.No	Book Title	Author
1.	Cloud Computing revised and updated edition, 2023	Nayan B. Ruparelia

NPTEL/ Youtube/ Faculty Video Link:

1.	https://nptel.ac.in/courses/106/104/106104182/ https://www.youtube.com/watch?v=M988_fsOSWo&t=4s https://www.youtube.com/watch?v=JYq1AQkMdhE https://www.youtube.com/watch?v=iSG_72VNBVs&t=55s
2.	https://nptel.ac.in/courses/106/105/106105167/ https://youtu.be/FZR0rG3HKIk?si=i9Ol3TdIeWtC-UUJ
3.	https://aws.amazon.com/ https://www.youtube.com/watch?v=36zducUX16w https://www.youtube.com/watch?v=3WIJ4axzFlU
4.	4 https://www.youtube.com/watch?v=m8iz4CFVWK0 https://www.youtube.com/watch?v=IKxigcbhsGk https://www.youtube.com/watch?v=NbkPRn1mqlU
5.	https://youtube.com/playlist?list=PL1TLTEHdRxDbFyipEb0KENRuBTI9yUu26&si=Si2LGUG6fu6v0Jr3

Mode of Evaluation

CIE							ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	TA3 5	Attendance 5		
30			20				50	100

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSAI0304	Course Name: High Performance Computing	L	T	P	C
Course Offered in: CSE(DS), CSE(AI) , CSE(AIML) , CS		3	0	0	3

Pre-requisite: C Language

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

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Identify the need, applications of HPC and emerging paradigms.	K2
CO2	Analyze architectures and parallelism for HPC performance.	K4
CO3	Apply message passing for parallel algorithm design.	K3
CO4	Analyze fault tolerance in cluster computing.	K4
CO5	Implement shared memory programs using OpenMP.	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	2	1	1	1	-	2	1	2	3	3
CO2	3	3	2	2	3	-	-	-	1	1	1	3	3	3
CO3	3	3	3	2	3	-	-	-	2	1	1	3	3	3
CO4	2	3	2	2	2	-	-	1	2	1	1	3	3	3
CO5	2	2	3	2	3	-	-	-	1	1	1	3	3	3

Course Contents / Syllabus

Module 1	**Introduction to HPC**	**10 hours**

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

Module 2	**Computing Architectures**	**10 hours**

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

Module 3	**Distributed Memory**	**10 hours**

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

Module 4	**Cluster Computing**	**9 hours**

BLAS, LAPAC, Cluster Computing, Clustering Models, Clustering Architectures, Clustering Architectures key factors, types of clusters, Mission critical Vs Business Critical Applications, Fault Detection and Masking Algorithms, Check pointing, Heartbeats, Watchdog Timers, Fault recovery through Failover and Failback Concepts.

Module 5	**OpenMP**	**9 hours**

OpenMP, Key concepts, Fork-join model, Goals of OpenMP, Supported platforms and compilers, OpenMP API components: directives, runtime routines, environment variables, General code structure, Core syntax and structured blocks, Compiling OpenMP programs, Parallel region construct, Creating and managing threads, Specifying the number of threads, Thread management and thread IDs, Applications of OpenMP.

Total Lecture Hours	**48 hours**

Textbook:.

S.No	Book Title	Author
1	The Sourcebook of Parallel Computing	Jack Dongarra, Geoffrey Fox, and Ken Kennedy
2	Petascale Computing: Algorithms and Applications	David A. Bader (Ed.)
3	Parallel and High Performance Computing	Robert Robey and Yuliana Zamora
4	High Performance Computing: Modern Systems and Practices	Thomas Sterling, Maciej Brodowicz, and Matthew Anderson

Reference Books:

S.No	Book Title	Author
1	High Performance Computing: Modern Systems and Practices	Thomas Sterling, Matthew Anderson, and Maciej Brodowicz
2	Using OpenMP: Portable Shared Memory Parallel Programming	Barbara Chapman, Gabriele Jost, and Ruud van der Pas
3	Introduction to Parallel Computing	Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar
4	The OpenMP Common Core: A Hands-on Exploration	Tim Mattson et al.
5	A Hands-on Introduction to OpenMP	Tim Mattson and Larry Meadows
6	OpenMP Tasking Explained	Ruud van der Pas
7	An Introduction to Parallel Programming	Peter Pacheco

NPTEL/ Youtube/ Faculty Video Link:

Unit 1	https://youtu.be/tGIobcyKViI?si=e6QKYtz8z1KHG5GI
Unit 2	https://www.youtube.com/watch?v=FTXikoQr46U
Unit 3	https://www.youtube.com/watch?v=gE8KgD9D69k
Unit 4	https://youtu.be/7rooWbLe1iI?si=hgkzh6bYDcxLxRQd
Unit 5	https://youtu.be/gcadMpUXcXs?si=DCWMph8Hgn6bt6-g

Mode of Evaluation

CIE							ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	TA3 5	Attendance 5		
30			20				100	150

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSE0353A	LAB Course Name: Operating Systems Lab	L	T	P	C
Course Offered in: CSE/CSE-R/IT/CS/AI/AIML/ IOT/DS/CYS		0	0	4	2

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

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

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

		Bloom's Knowledge Level (KL)
CO1	Execute basic Linux commands and shell scripts to automate file management and system administration tasks.	K3
CO2	Implement and compare various CPU scheduling algorithms, process synchronization solutions using semaphores and deadlock handling algorithms.	K4
CO3	Simulate memory allocation techniques and page replacement algorithms, disk management strategies and explore modern OS features including virtualization and distributed computing.	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	2	1	1	1	3	-	-	2	2	-	3	2	-	1
CO2	3	3	3	2	2	-	-	2	2	-	2	1	-	1
CO3	3	2	3	2	3	-	-	2	2	-	3	1	-	-

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

1	**Implementation of Linux Commands** i. Introduction of Unix/Linux Operating system and their architecture ii. Display system information using uname, hostname, and date etc. iii. File operations using cat, touch, cp, mv, rm, and chmod ,umask etc. iv. Create, view, and navigate directories using mkdir, rmdir, cd, pwd, ls etc. v. Disk Commands df,du,mount,unmount,mkfs,fsck etc. vi. Use redirection and piping in commands vii. File compression and archiving using tar, gzip, zip, unzip etc. viii. Process commands ps,kill, killall,nice, pgrep, top,htop etc. ix. Network commands ifconfig, ping, netstat, host,ip route etc. x. Administrator Commands Adduser,Passwd, deluser, usermod, groupadd etc xi. Implement different types of system calls in Unix/Linux.

2	**Shell Scripting Programming** i. Write a shell script to ask your name, program name and enrollment number and print it on the screen. ii. Write a shell script to find the sum, the average and the product of the four integers entered. iii. write shell script to find average of numbers given at command line iv. Write a shell program to exchange the values of two variables v. Write a shell program to Print Numbers 1 to 10 using while & do while loop. vi. Write a shell program to Print Numbers 1 to 10 using for loop. vii. Write a shell script to display the digits which are in odd position in a given 5-digit number**.** viii. Write a shell program to search for a given number from the list of numbers provided using binary search method. ix. Write a shell program to concatenate two strings and find the length of the resultant string x. Write a shell script to find the smallest of three numbers xi. Write a shell program to count number of words, characters, white spaces and special symbols in a given text
	Process & Thread Management
3	Introduction to C Programming (Statement, Conditional Statement, Loop, Array & Function)
4	Implement FCFS CPU Scheduling algorithm.
5	Implement the SJF CPU Scheduling algorithm (For both Pre-emptive and Non-pre-emptive).
6	Implement PRIORITY CPU Scheduling Algorithm (For both Pre-emptive and Non-pre-emptive).
7	Implement Round-Robin CPU Scheduling Algorithm.
8	Implement Multi-Level Queue CPU Scheduling algorithm.
9	Implement Multilevel Feedback Queue CPU Scheduling Algorithm.
	Concurrency and Deadlock Management
10	Execute the RACE Condition of Process Synchronization.
11	Implement the Producer–consumer problem using semaphores.
12	Design a code and implement the Dinning Philosopher problem.
13	Implement Banker's algorithm of Deadlock Avoidance.
14	Execute an algorithm for Deadlock Detection.
	Memory Management
15	Implement the Memory Fixed-size partition scheme.
16	Implement the Memory Variable-size partition scheme.
17	Simulate the First-Fit contiguous memory allocation technique.
18	Simulate the Best-Fit contiguous memory allocation technique.
19	Simulate the Worst-Fit contiguous memory allocation technique.
20	Implement the Non-contiguous Memory Allocation by using Paging.
	Page Replacement
21	Write a Program to simulate the FIFO page replacement algorithm.
22	Write a Program to simulate the LRU page replacement Algorithm.
23	Write a Program to simulate the Optimal page replacement Algorithm.
	Disk Scheduling
24	Write a program to simulate FCFS Disk Scheduling Algorithm.
25	Write a Program to simulate the SSTF Disk Scheduling Algorithm.
26	Write a program to simulate SCAN Disk Scheduling Algorithm.
27	Write a Program to simulate the C SCAN Disk Scheduling Algorithm.

28	Write a Program to simulate the LOOK Disk Scheduling Algorithm.
29	Simulate all file allocation strategies a) Sequential b) Indexed c) Linked.
	Modern Operating System
30	Introduction of CUDA Programming.
31	Write a program in CUDA print message “Welcome CUDA programming”
32	Implement matrix multiplication using shared memory in CUDA.
33	Connects to VMware vCenter and lists all virtual machines along with their power state.
34	Create a new virtual machine in Azure with specified configurations.
35	Deploy a simple HTTP-triggered distributed Azure Function.
	Total Hours: 48 hrs.

Mode of Evaluation

CIE			**PE**	**Total**
PS1 (10)	**PS2 (20)**	**PS3 (20)**		
50			**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science & Information Technology

LAB Course Code: BCSE0351	**LAB Course Name: DATA STRUCTURE AND ALGORITHMS-I LAB**	**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: The concept of Programming Language

Course Objective:

The objective of the course is to compare the time complexities of various algorithm and implementation of linear data structure.

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

		Bloom's Knowledge Level (KL)
CO1	Implement array and matrix operations along with searching and sorting algorithms to solve computational problems.	K3
CO2	Implement Link list, Stack and Queues with their applications.	K3
CO3	Implement divide and conquer and greedy algorithms to solve problems like sorting, scheduling and optimization.	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	3	3	1	1	1	1	1	2	3	1	1
CO2	3	3	3	2	3	1	1	1	1	1	2	3	1	1
CO3	3	3	3	3	3	1	1	1	1	1	3	3	1	1

List of Practical (Indicative & Not Limited To)

1. Construct a program to compare the time complexities of selection, bubble and insertion sort by plotting the graph.
2. Construct a program to compare the time complexities of various algorithms by varying size "n".
3. Construct a program to find the maximum element in an array.
4. Construct a program to calculate the sum of all elements in an array.
5. Construct a program to reverse the elements of an array.
6. Construct a program to check if an array is sorted in ascending order.
7. Construct a program to count the occurrence of a specific element in an array.
8. Construct a program for creation and traversal of 2D Array in row major and column major order.
9. Construct a program to print the transpose of a given matrix using function.
10. Construct a program to find if a given matrix is Sparse or Not and print Sparse Matrix.

11. Construct a program to represent a sparse matrix in triplet form.
12. Construct a program to implement Linear Search.
13. Construct a program to implement Binary Search.
14. Construct a program to implement Selection Sort.
15. Construct a program to implement Bubble Sort.
16. Construct a program to implement Insertion Sort.
17. Construct a program to implement Shell Sort.
18. Construct a program to implement Counting Sort.
19. Construct a program to create a single linked list and perform basic operations (insertion, deletion, traversal).
20. Construct a program to create a double linked list and perform basic operations (insertion, deletion, traversal).
21. Construct a program to create a circular linked list and perform basic operations (insertion, deletion, traversal).
22. Construct a program to create a circular double linked list and perform basic operations (insertion, deletion, traversal).
23. Construct a program to reverse a single linked list.
24. Construct a program to check if a linked list is palindrome.
25. Construct a program to reverse a double linked list.
26. Construct a program to find the middle element of a single linked list.
27. Construct a program to find the middle element of a double linked list.
28. Construct a program to merge two sorted single linked lists.
29. Construct a program to detect and remove a loop in a circular linked list.
30. Construct a program to add two polynomials using linked list.
31. Construct a program to implement stack using array.
32. Construct a program to implement stack using a linked list.
33. Construct a program to infix to postfix conversion using a stack.
34. Construct a program for balanced parentheses checker using a stack.
35. Construct a program to reverse a string using a stack.
36. Construct a program to implement Binary search using recursion.
37. Construct a program to print Fibonacci series using recursion.
38. Construct a program to implement Tower of Hanoi.
39. Construct a program to implement queue using array.
40. Construct a program for implementing a circular queue.
41. Construct a program to implement queue using stack.
42. Construct a program to implement priority queue.
43. Construct a program to implement double ended queue.
44. Construct a program to implement Merge Sort with recursion.

45. Construct a program to implement Quick Sort with recursion.	
46. Construct a program to implement Merge Sort using iteration.	
47. Construct a program to implement Quick Sort using iteration.	
48. Construct a program to implement fractional knapsack.	
49. Construct a program to implement Activity selection problem.	
50. Construct a program to implement Job scheduling problem.	
Total Hours	**48 Hours**

Mode of Evaluation		
CIE	**PE**	**Total**
PS		
50	**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code:BCS0351	LAB Course Name: Foundation of Cloud Computing Lab	L	T	P	C
Course Offered in: Computer Science		**0**	**0**	**2**	**1**

Pre-requisite: Basic knowledge of computer science

Course Objectives: Learn to provide students with practical experience in using cloud services and tools. This involves setting up virtual machines, storage solutions, databases, networking configurations.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Demonstrate and navigate to AWS services using the AWS Management Console.	K2
CO2	Configure and monitor virtual resources such as EC2, S3, EBS, and EFS in AWS.	K4
CO3	Design scalable cloud solutions using VPC, IAM, and RDS while applying monitoring and security best practices.	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	2	2	1	2	1	2	1	1	-	2	2	1	1	1
CO2	1	2	2	2	1	2	2	1	-	2	2	2	2	1
CO3	2	2	3	2	2	3	2	2	1	3	2	2	2	1

List Of Practicals (Indicative & Not Limited To)

1. Introduction to AWS services and the Management Console interface.
2. Create and configure Linux/Windows EC2 instances with key pairs and security groups.
3. Start, stop, reboot, terminate, and connect to EC2 instances via SSH or RDP.
4. Use Amazon CloudWatch to monitor CPU, disk, and network metrics of EC2 instances.
5. Create S3 buckets, upload/download files, set access policies, and enable versioning.
6. Create, attach, detach, and snapshot EBS volumes.
7. Create an EFS and mount it across multiple EC2 instances.
8. Set up a MySQL/PostgreSQL database using Amazon RDS and connect it to an application.
9. Configure subnets, route tables, internet gateway, and NAT for a private cloud setup.
10. Create IAM users/groups, assign policies, and demonstrate fine-grained access control.
11. Configure launch templates, auto scaling groups, and scaling policies for EC2.
12. Track user activity and API usage using CloudTrail and create custom CloudWatch dashboards.

Total Hours: 24 hrs.

Mode of Evaluation

CIE			PE	Total
PS1	**PS2**	**PS3**	**(If mentioned in curriculum)**	
5	**10**	**10**		
25			**25**	**50**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSE0352	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:

1. Basic Programming Knowledge
2. 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

S.No	Course Outcome	Bloom's Level
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	Implement Concurrency control, I/O Streams and Java Socket Programming Concepts.	K3
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	2	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	2
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	3

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.

E**xception Handling, Assertions and Localizations:** Introduction and Types, Exceptions vs. Errors, Handling of Exception. Finally, Throws and Throw keyword, Multiple Catch Block, Nested Try and Finally Block, Customized Exceptions, Tokenizer. Assertions and Localizations Concepts and its working.

String Handling: Introduction and Types, Operations, Immutable String, Method of String class, String Buffer and String Builder class.

Unit 4	**Concurrency in Java and I/O Stream**	**16 hours**

Threads: Introduction and Types, Creating Threads, Thread Life-Cycle, Thread Priorities, Daemon Thread,Runnable Class, Synchronizing Threads etc.

I/O Stream: Introduction and Types, Common I/O Stream Operations, Interaction with I/O Streams Classes. character and byte oriented stream classes with example

Java Socket Programming: Introduction and types(TCP, UDP), java socket program with server-side and client-side by using connection.

Unit 5	**GUI Programming, Generics and Collections**	**16 hours**

GUI Programming: Introduction and Types, Swing, AWT, Components and Containers, Layout Managersand 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:

1. Herbert Schildt," Java: A Beginner's Guide", McGraw-Hill Education 2nd edition
2. E Balagurusamy, "Programming with Java A Primer", TMH, 4th edition.

Reference Books:

1. Cay S. Horstmann, "Core Java Volume I – Fundamentals", Prentice Hall
2. Joshua Bloch," Effective Java", Addison Wesley
3. Herbert Schildt," Java - The Complete Reference", McGraw Hill Education 12th edition

NPTEL/ Youtube/ Faculty Video Link:

Unit 1	https://www.youtube.com/watch?v=r59xYe3Vyks&list=PLS1QulWo1RIbfTjQvTdj8Y6yyq4R7g-Al
Unit 2	https://www.youtube.com/watch?v=ZHLdVRXIuC8&list=PLS1QulWo1RIbfTjQvTdj8Y6yyq4R7g-Al&index=18
Unit 3	https://www.youtube.com/watch?v=hBh_CC5y8-s
Unit 4	https://www.youtube.com/watch?v=qQVqfvs3p48
Unit 5	https://www.youtube.com/watch?v=2qWPpgALJyw

Mode of Evaluation

CIE							ESE	
ST1	**ST2**	**ST3**	**TA1 (5)**	**TA2 (5)**	**TA3 (5)**	**Attendance (5)**		
30			**20**				**100**	

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

1.Understanding Text Editors to Write Programs Compile and run first java file Byte Code and class file

2. Sketch a class and object diagram describing the sales order system of restaurant

3. Sketch a class diagram describing the circle and rectangle class

4. Sketch a class diagram for a college platform including, classroom, playground, chair, table, smart board, teaching staff etc.

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.

6.Program to display default value of all Primitive data types

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.

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.

9. A class NumberPalindrome with a public method isNumberPalindrome that takes one parameter number of 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
11. Write a Java Program to find the Factorial of a given number.
12. Java Program to create a class, methods and invoke them inside main method.
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.
14. Java program to illustrate the static field in the class.
15. Java Program to illustrate static class.
16. Write a java program to access the class members using super keyword
17. Java program to access the class members using this keyword
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.
19.Java program to demonstrate nested interface inside a interface.
20.Java program to demonstrate nested interface inside a class.
21.Java program to explicit implementation of garbage collection by using finalize() method
22.Java program to implement Single Inheritance
23.Java program to implement multi- level Inheritance
24.Java program to implement constructor and constructor overloading.
25.Java program implement method overloading.
26.Java program to implement method overriding.
27.Java program to implement lambda expression without parameter.
28.Java program to implement lambda expression with single parameter.
29.Java program to implement lambda expression with multi parameter.
30. Java program to implement lambda expression that iterate list of objects
31.Java program to define lambda expressions as method parameters
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. Assummptions: • arr is never null • arg1 and arg2 may be same

33.Java program to show the multiplication of two matrices using arrays.
34.Java Program to search an element using Linear Search
35.Java program to search an element using Binary Search
36.Java Program to sort element using Insertion Sort
37.Java Program to sort element using Selection Sort– Largest element Method
38.Java program to Sort elements using Bubble Sort
39.Java program to create user defined package.
40.Java Program to create a sub- classing of package
41.Implement the following: 1. Import package.*; 2. import package.classname; 3. Using fully qualified name.
42.Implement and demonstrate package names collision in java
43.Java program to handle and Arithmetic Exception Divided by zero
44.Java Program to implement User Defined Exception in Java
45.Java program to illustrate finally block
46.Java program to illustrate Multiple catch blocks
47.Java program for creation of illustrating throw in exception handling.
48.Implement the concept of Assertion in Java Programming Language
49.Implement the concept of Localization in Java Programming Language.
50.Java program to print the output by appending all the capital letters in the input string.
51.Java program that prints the duplicate characters from the string with its count.
52.Java program to check if two strings are anagrams of each other
53.Java Program to count the total number of characters in a string
54.Java Program to count the total number of punctuation characters exists in a String
55.Java Program to count the total number of vowels and consonants in a string
56.Java Program to show .equals method and == in java
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".

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 "HWeolrllod".
59.Java program to show the usage of string builder.
60.Java program to show the usage of string buffer.
61.Creating and Running a Thread
62.Implementing Runnable Interface
63.Synchronizing Threads with lock
64.Synchronizing Threads without lock
65.Java program to implement even and odd threads by using Thread class .
66.Java program to implement even and odd threads by using Runnable interface.
67.Java program to synchronize the threads by using Synchronize statements and Synchronize block.
68.Write a program where the client sends a message to the server, and the server prints it by using TCP
69.Implement a server that can handle multiple clients simultaneously using UDP
70.Write a client-server application where the client uploads a file and the server saves it by using TCP/UDP.
71. Java program to implement that read a character stream from input file and print it into output file.
72.Java program to implement that merge the content of two files (file1.txt, file2.txt) into file3.txt.
73.Write a Java program that reads the contents of one file and copies them toanother file.
74.Write a Java program that reads a text file and counts the number of words in it.
75.Write a Java program that reads a text file and counts the frequency of each word in it.
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.
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.
78.Program to create a frame with three button in AWT and swing
79.Program to display message with radio buttons in swing
80.Program to display "All The Best" in 5 different colors on screen. (Using AWT/Swing)
81.Program to implement handling in a button “OK”
82.Java Program to implement BorderLayout

83.Java Program to implement GridLayout
84. Java Program to implement BoxLayout
85. Java Program to implement CardLayout
86. Java program to implement Generic class
87. Java program to illustrate Generic methods
88.Java program to implement wildcard in generics
89.Java program to implement of methods of HashSet
90.Java Program to implement methods available in HashMap class
91.Program to add, retrieve, and remove element from ArrayList
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.
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.
94. Java program to implement autoboxing
95.Java program to implement unboxing
96.Develop a java class with a method storeEvenNumbers(int N) using ArrayList to store even numbers from 2 to N, where N is a integer which is passed as a parameter to the method storeEvenNumbers(). The method should return the ArrayList (A1) created.
97.Create a method that accepts the names of five countries and loads them to an array list and returns the list.
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.

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

BGREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BNC0302	**Course Name:** Environmental Science	**L**	**T**	**P**	**C**
Course Offered in: All the branches		**2**	**0**	**0**	**-**

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

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 environment. Ecosystem: Basic concepts, components of ecosystem, food chains and food webs. Ecological pyramids, biodiversity.	K1,K2
CO2	Understand the different types of natural recourses like food, forest, Minerals and energy and their conservation.	K1,K2
CO3	Understand the different types of pollution, pollutants, their sources, effects and their control methods.	K1,K2
CO4	Understand the basic concepts of sustainable development, Environmental Impact Assessment (EIA) and different acts related to environment	K1,K2

CO-PO Mapping

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

Course Contents / Syllabus

Module 1	**Basic Principle of Ecology and Biodiversity**	**4 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.

Module 2	**Natural Resources and Ecological succession**	**4 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

Module 3	**Pollution and Waste Management**	**4 hours**

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

Module 4	**Environmental Assessment and Legislation**	**4 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 UNSDsGs, System Thinking and Sustainability.

Total Lecture Hours	**20 hours**

Textbook:

S.No	**Book Title**
1	Brady, N.C. 1990. The nature and properties of Soils, Tenth Edition. Mac Millan Publishing Co., New York
2	Sodhi G.S. 2005, Fundamentals of Environmental Chemistry: Narosa Publishing House, New Delhi.
3	Dash, M.C. (1994), Fundamentals of Ecology, Tata Mc Graw Hill, New Delhi.

S.No	**Book Title**
1	Rao M.N. and H.V.N. Rao, 1989 : Air Pollution, Tata McGraw Hill Publishing Co. Ltd., New Delhi
2	A Text Book of environmental Science By Shashi Chawla

Unit 1:	https://www.youtube.com/watch?v=T21OO0sBBfc, https://www.youtube.com/watch?v=qt8AMjKKPDo
Unit 2:	https://www.youtube.com/watch?v=mOwyPENHhbc, https://www.youtube.com/watch?v=yqev1G2iy2 https://www.youtube.com/watch?v=_74S3z3IO_I, https://www.youtube.com/watch?v=jXVw6M6m2
Unit 3:	https://www.youtube.com/watch?v=7qkaz8ChelI, https://www.youtube.com/watch?v=NuQE5fKmfME https://www.youtube.com/watch?v=9CpAjOVLHII, ttps://www.youtube.com/watch?v=yEci6iDkXYw

Unit 4	https://www.youtube.com/watch?v=ad9KhgGw5iA, https://www.youtube.com/watch?v=nW5g83NSH9 M, https://www.youtube.com/watch?v=xqSZL4Ka8xo

Mode of Evaluation

CIE							ESE	Total
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**TA3** **5**	**Attendance** **5**		
30			**20**				**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BASCC00401	**Course Name: Employability Skill Development – II**	**L**	**T**	**P**	**C**
Course Offered in:		**2**	**0**	**0**	**2**

Pre-requisite: Basic understanding of elementary mathematics

Course Objectives:

The objective of this course is to develop students' quantitative aptitude and logical reasoning skills through number theory, analytical puzzles, and business mathematics, enabling them to solve real-world and competitive exam problems with speed, accuracy, and logical thinking.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply fundamental number theory concepts such as divisibility, HCF & LCM, remainder theorem, and cyclicity to solve quantitative problems efficiently.	K2, K3
CO2	Solve problems involving logical reasoning and analytical thinking, including direction sense, blood relations, series patterns, and time-based puzzles like clocks and calendars.	K3
CO3	Solve real-life business math problems involving percentages, profit and loss, discounts, interest average calculations and using appropriate mathematical methods	K2, K3
CO4	Solve real-life business math problems involving averages, mixtures, and ratios using appropriate mathematical methods	K2, 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	**1**	**1**	**1**	**1**	-	-	-	-	-	-	-	3	2	2
CO2	**1**	**1**	**1**	**1**	-	-	-	-	-	-	-	2	2	3
CO3	**1**	**1**	**1**	**1**	-	-	-	-	-	-	-	3	3	2
CO4	**1**	**1**	**1**	**1**	-	-	-	-	-	-	-	2	2	2

Course Contents / Syllabus

Module 1	**Speed Math and Number System**	**8 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

Module 2	**Analytical and Logical Reasoning**	**8 hours**

Direction and Sense, Blood Relation, Number Series and Letter Series, Coding Decoding,

Module 3	**Business Math I**	**8 hours**

Percentage, Profit and Loss, Discount, Simple Interest and Compound Interest, Average

Module 4	Business Math II	8 hours
Ratio & Proportion, Partnership, Mixture & Allegation, Clock , Calendar		

Reference Books:

S.No	Book Title
1	M. Tyra (BSC publication co. Pvt. Ltd), Quicker math
2	RS Aggarwal , Quantitative Aptitude
3	RS Aggarwal, Verbal & Non-Verbal Reasoning
4	Sarvesh K Verma, Quantitative Aptitude - Quantum CAT

Mode of Evaluation

CIE							ESE	Total
ST1	ST2	ST3	TA1	TA2	TA3	Attendance		
			5	5	5	5		
30			20				50	100

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSE0402	**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		**3**	**0**	**0**	**3**

Pre-requisite: It is recommended to have fundamental computer knowledge that includes concepts of computer architecture, storage and hardware. Knowledge of data structures and algorithms and programming will be an added benefit.

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

Course Outcome- After completion of this course students will be able to		Bloom's Knowledge Level (KL)
CO 1	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.	K5
CO4	Execute the concept of PL/SQL, transaction and concurrency control.	K3
CO5	Implement Relational and Non-Relational databases using different tools and evaluate their effectiveness in real-world applications..	K5

CO-PO Mapping

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

Course Contents / Syllabus

Module 1	**Introduction about the Database Conceptual Designing**	**8 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.

Module 2	Basic of SQL & Normalization	8 hours
Keys & Types of Keys: Super key, Candidate Key, Primary Key, Alternative Key, Foreign Key, unique. Constraints and Types of Constraints. **Use of Functions, Clause and Predicates:** Aggregate Function, Scalar Functions, Where, Group by, Having and Order by, SQL Operators. Like, Between, Aliases, distinct, limit. **Normalization:** Functional Dependencies, attribute Closure, Normalization & Types of Normalization, Candidate Key, Canonical Cover of FD's.		
Module 3	**Introduction of Complex Queries**	**8 hours**
Use of Operators: Union, Intersect, Minus, Cartesian Product, join:-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, Procedure, Trigger, Cursor. **Database connectivity:** Database Connectivity with Java/Python Programming Languages.		
Module 4	**Transaction and Concurrency Control**	**8 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.		
Module 5	**Introduction of NoSQL With MongoDB**	**8 hours**
Introduction of NoSQL With MongoDB : Introduction of 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 and Riak. Introduction and Features of MongoDB, MongoDB Operators, MongoDB Collection & Document, MongoDB Shell & their commands, CRUD operations. **Cloud Database**Introduction of Cloud Database. MongoDB Cloud product : Stitch, Atlas & Cloud Manager.		
	Total Lecture Hours	**40**

Textbook:

S.No	Book Title
1	Abraham Silberschatz, Henry Korth and S. Sudarshan, "Database Concepts", McGraw Hill, 7th Edition, 2020
2	Elmasri, Navathe, "Fundamentals of Database Systems", Addison Wesley, 7th edition, 2016

Reference Book

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

NPTEL/ Youtube/ Faculty Video Link:

Unit 1:	DBMS L1 Inauguration & Introduction (youtube.com) DBMS L2 Introduction to Relational Model (youtube.com)

	DBMS L3 Introduction to SQL (youtube.com) DBMS L8C Entity Relationship Model (youtube.com) DBMS L8D Entity Relationship Model (Problem Solving and Discussion) (youtube.com)
Unit 2:	DBMS L4A Joins, Set Operations and Aggregate Functions (youtube.com) DBMS L9A Relational Database Design - YouTube DBMS L9B Relational Database Design (youtube.com) DBMS L9C Relational Database Design (youtube.com) DBMS L9D Discussion on Normalization (youtube.com) Relational Data Model and Notion of Keys - YouTube Introduction to Relational Algebra (youtube.com) Operators in Relational Model – YouTube
Unit 3:	DBMS L4B Joins, Set Operations and Aggregate Functions (youtube.com) DBMS L5A Nested Subqueris (youtube.com) DBMS L6A Intermediate SQL (youtube.com) DBMS L7 Advanced SQL (youtube.com) DBMS L12A Indexing and Hashing (youtube.com)
Unit 4	DBMS L15 Transactions – YouTube DBMS L16A Concurrency Control - YouTube DBMS L16B Concurrency Control (youtube.com) DBMS L16C Concurrency Control (youtube.com)
Unit 5	DBMS L10A Application Design and Development - YouTube DBMS L10B Application Design and Development (youtube.com) DBMS L19 Distributed Data Stores and NoSQL Databases (youtube.com) DBMS L18B Map Reduce and Hadoop - YouTube NoSQL Databases #1 (Data Models, CAP Theorem, BASE Property) - YouTube https://youtu.be/ekuQjQUnj20?si=_aL4T12EkHBZsvEK

Mode of Evaluation

CIE							ESE	Total
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**TA3**	**Attendance**		
			5	**5**	**5**	**5**		

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSE0401	Course Name: DATA STRUCTURES AND ALGORITHMS-II	L	T	P	C
Course Offered in: CSE/CS/CSR-R/M.TECH(INT) /IT//CSE(AI)/CSE(AIML)/CSE(DS)/CSE(CS)		3	0	0	3

Pre-requisite: C, Python

Course Objectives:

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

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

S.No	Course Outcome	Bloom's Level
CO 1	Apply tree structures effectively demonstrating proficiency in tree operations and algorithms.	K3
CO 2	Analyse the graph data structure and implement various operations for problem solving.	K4
CO 3	Implementation and analysis of dynamic programming for efficient problem-solving across diverse contexts.	K4
CO 4	Apply efficient backtracking and branch &bound techniques across diverse problem-solving scenarios.	K3
CO 5	Understand advanced data structures, their implementation and application for efficient data manipulation and retrieval.	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	-	3	1	2
CO2	3	3	2	1	**1**	2	1	1	2	1	-	3	2	2
CO3	3	3	2	2	2	2	1	1	2	2	-	3	2	2
CO4	3	3	3	2	2	2	1	1	2	1	-	3	2	2
CO5	3	3	3	2	2	2	1	1	2	2	-	3	2	2

Course Contents / Syllabus

Unit 1	Design and Analysis of Algorithms: Tree	8 hours

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

Application of Trees: Priority Queue, Heap Sort, Huffman codes.

Unit 2	Design and Analysis of Algorithms: Graphs	8 hours

Graphs: Terminology used with Graph, Data Structure for Graph Representations: Adjacency matrices, Adjacency

List.

Graph Traversal: Depth First Search and Breadth First Search. Connected Component, Spanning Trees.

Algorithms on Graphs: Minimum Cost Spanning

Trees: Prim' s and Kruskal's algorithm. Directed- Acyclic Graph, Transitive Closure and Shortest Path algorithms: Dijkstra Algorithm, Bellman Ford Algorithm, Floyd Warshall's Algorithm.

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

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

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

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

Unit 5	**Advanced- Data Structures**	**8 hours**

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

	Total Lecture Hours	**40 hours**

Textbook:

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

Reference Books:

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

NPTEL/ YouTube/ Faculty Video Link:

Unit 1	https://youtu.be/u5AXxR4GnRY
Unit 2	https://www.youtube.com/watch?v=LQx9E2--p5c&pp=ygUMYXJyYXlzIG5wdGVs
Unit 3	https://www.youtube.com/watch?v=K7VIKlUdo20&pp=ygUPbGluayBsaXN0IG5wdGVs
Unit 4	https://www.youtube.com/watch?v=g1USSZVWDsY&list=PLB3CD0BBB95C1BF09&index=2&pp=iAQB https://www.youtube.com/watch?v=THMyk2_p530&pp=ygUccXVldWUgZGF0YSBzdHJ1Y3R1cmUgICBucHRlbA%3D%3D
Unit 5	https://www.youtube.com/watch?v=_VV9v41FIq0&pp=ygUZZGl2aWRlIGFuZCBjb25xdWVyICBucHRlbA%3D%3D https://www.youtube.com/watch?v=ARvQcqJ_-NY&list=PLfFeAJ-vQopt_S5XlayyvDFL_mi2pGJE3

Mode of Evaluation

CIE						ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	Attendance 5		
35			15			100	150

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

Course Code: BCS0402	**Course Name: Big Data Analytics**	**L**	**T**	**P**	**C**
Course Offered in: Computer Science		**3**	**0**	**0**	**3**

Pre-requisite: Java and event handling concept

Course Objectives: To understand the basic concepts of Big Data in cloud and analyse sample dataset using big data ecosystem.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Identify Big Data and relevance of Big Data Analytics.	K2
CO2	Analyze Map Reduce and demonstrate its use in features extraction.	K4
CO3	Explain the YARN and HDFC in Data management	K2
CO4	Articulate the concept of Cloud Computing and evolution of cloud computing with characteristics .	K3
CO5	Analyze the components of open stack & Google Cloud platform	K4

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

CO-PO Mapping	**PO1**	**PO2**	**PO 3**	**PO4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO9**	**PO10**	**PO1 1**	**PSO 1**	**PSO 2**	**PSO 3**
CO1	**2**	**2**				**1**			**2**		**2**	**1**	1	2
CO2	**2**	**3**	**2**		**2**				**2**			**2**	1	2
CO3	**2**	**3**			**2**				**2**			**2**	2	2
CO4	**2**	**3**				**2**			**2**		**2**	**2**	2	2
CO5	**2**	**3**	**2**		**2**				**2**		**3**	**2**	2	2

Course Contents / Syllabus

Module 1	**INTRODUCTION TO BIG DATA AND CLOUD**	**9 hours**

Introduction to Big Data: Types of digital data, history of Big Data innovation, introduction to Big Data platform, drivers for Big Data, Big Data architecture and characteristics, 5 Vs of Big Data, Big Data technology components, Big Data importance and applications, Big Data features, Big Data Analytics, modern data analytic tools.
Introduction to Cloud Computing: Definition of Cloud, Evolution of Cloud Computing, Underlying Principles of Parallel and Distributed Computing, Cloud Characteristics.

Module 2	**HADOOP AND MAP-REDUCE**	**10 hours**

Hadoop: History of Hadoop, Apache Hadoop, the Hadoop Distributed File System, components of Hadoop, data format, analyzing data with Hadoop, scaling out, Hadoop streaming, Hadoop pipes, Hadoop Echo System.
Map Reduce: Map-Reduce framework and basics, how Map Reduce works, anatomy of a Map-Reduce job run, failures, job scheduling, shuffle and sort, task execution, Map Reduce types, input formats, output formats, Map Reduce features, Real-world Map Reduce. **Hadoop Eco System and YARN:** Hadoop ecosystem components, Hadoop 2.0 New Features, MRv2, YARN

Module 3	**HADOOP ARCHITECTURE & FRAMEWORK**	**10 hours**

HDFS (Hadoop Distributed File System): Design of HDFS, HDFS concepts, benefits and challenges, file sizes, block sizes and block abstraction in HDFS, how does HDFS store, read, and write files, Flume and Scoop, Hadoop archives, Hadoop I/O: compression, serialization, Avro and file-based data structures. Hadoop Eco- System Frameworks: PIG , HIVE , HBASE , ZOOKEEPER.
Importing and Handling Relational Data in Hadoop using Sqoop , Scala , spark.

Module 4	**HADOOP IN CLOUD**	**10 hours**

Cloud Technologies And Advancements Hadoop: MapReduce, Cloud overview & characteristics, cloud service model (iaas, paas, saas) , cloud deployment model (public, private, hybrid), Google cloud platform (gcp) infrastructure

overview create gcp account & console overview, Virtual Box , Google App Engine, Programming Environment for Google App Engine Open Stack Federation in the Cloud, our Levels of Federation, ederated Services and Applications, Future of Federation.

Module 5	**NETWORK AND DATA STORAGE SERVICES**	**9 hours**

Virtual networks: virtual private cloud (vpc) & types, subnets , ip addresses (public/private), nic ,routes & route table , firewalls , network topology options .
Google cloud storage overview & Structure: cloud datastore, cloud bigtable : nosql big data service bigquery basics, how to use machine learning with Bigquery.

Total Lecture Hours	**48 hours**

Textbook:

S. No	**Book Title**	**Author**
1.	"Programming Hive", O'Reilley, 2012. 7. Lars George, "HBase: The Definitive Guide", O'Reilley, 2011.	E. Capriolo, D. Wampler, and J. Rutherglen
2.	"Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012. 5. Eric Sammer, "Hadoop Operations", O'Reilley, 2012.	Tom White

Reference Books:

S. No	**Book Title**	**Author**
1.	**"Programming Pig", O'Reilley,**	Alan Gates
2.	**Big Data: A Revolution that will transform how we live,work and think.**	Viktor Mayer-Schonberger, ennethCukier

NPTEL/ Youtube/ Faculty Video Link:

1.	(4) noc19-cs33 Lecture 1-Introduction to Big Data - YouTube
2.	(4) Lecture 26: Map-reduce and Hadoop - YouTube(3) Lecture 2 | Image Classification - YouTube
3.	(4) Hadoop Ecosystem | Big Data Analytics Tools | Hadoop Tutorial | Edureka - YouTube (4) What is HDFS | Hadoop Distributed File System (HDFS) Introduction | Hadoop Training | Edureka - YouTube
4.	(4) Hive Tutorial for Beginners | Hive Architecture | Hadoop Hive Tutorial | Hadoop Training | Edureka - YouTube (4) HBase Tutorial for Beginners | Introduction to Apache HBase | Hadoop Training | Edureka - YouTube https://www.youtube.com/watch?v=Qhc6RMaDkgY
5.	(4) Sqoop Tutorial - How To Import Data From RDBMS To HDFS | Sqoop Hadoop Tutorial | Simplilearn - YouTube (4) Java in Spark | Spark-Submit Job with Spark UI Example | Tech Primers - YouTube (4) Java in Spark | Spark-Submit Job with Spark UI Example | Tech Primers - YouTube

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1 5**	**TA2 5**	**TA3 5**	**Attendance 5**		
30			**20**				**100**	**150**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

Course Code: BCSAI0411	**Course Name: Data Analytics**	**L**	**T**	**P**	**C**
Course Offered in: DATA ANALYTICS		**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: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the fundamental concepts of data analytics in key areas relevant to data science.	K1
CO2	Explain and exemplify common data types and their representations.	K2
CO3	Apply data pre-processing techniques to real-world datasets.	K3
CO4	Analyze datasets through exploratory data analysis techniques.	K4
CO5	Evaluate and illustrate appropriate data visualization methods for varied datasets and scenarios.	K5

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

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

Course Contents / Syllabus

Module 1	**Introduction To Data Science**	**15 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.

Module 2	**Data Handling**	**14 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.

Module 3	**Data Pre-processing**	**15 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

Module 4	**Exploratory Data Analysis**	**14 hours**

AWS global infrastructure, How to select a region, What is edge location and regional edge cache, AWS global, regional and zonal services, Local zones, wavelength zones and outposts, Benefits of using AWS cloud, AWS shared responsibility model, AWS acceptable use policy, Virtualization and hypervisors, Regions and availability zones, EC2 - Old Console vs New Console, Launch EC2 instance, Create instances with Elastic block storage (EBS), Elastic File Systems (EFS) and Simple Storage Service (S3), Amazon Dynamo DB introduction, Dynamo DB : Create table and add items, Dynamo DB: Scan and query operations, Different types of NoSQL databases, SQL vs NoSQL.

Module 5	**Data Visualization**	**14 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, AVR, 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 Lecture Hours	**72 hours**

Textbook:

S.No	Book Title	Author
1	Making sense of Data: A practical Guide to Exploratory Data Analysis and Data Mining, John Wiley Publishers, 2007	Glenn J. Myatt
2	Data Analysis and Data Mining, 2nd Edition, John Wiley & Sons Publication, 2014.	Glenn J. Myatt

Reference Books:

S.No	Book Title	Author
1	Open Data for Sustainable Community: Glocalized Sustainable Development Goals	Neha Sharma, Santanu Ghosh, Monodeep Saha
2	The Data Science Handbook, John Wiley & Sons, Inc., 2017 (1st edition)	Field Cady
3	Data Mining Concepts and Techniques (3rd Edition), Morgan Kaufmann (Elsevier), 2012 (often noted June 2011)	Jiawei Han, Micheline Kamber, Jian Pei

NPTEL/ Youtube/ Faculty Video Link:

1	https://www.youtube.com/playlist?list=PL15FRvx6P0OWTlNBS_93NHG2hIn9cynVT
2	https://www.youtube.com/playlist?list=PLLy_2iUCG87DxxkLX4Pc3wCvsF1yAvz0T
3	https://www.youtube.com/watch?v=lhO3fBiMDag
4	https://www.youtube.com/watch?v=q4pyaVZjqk0
5	https://www.youtube.com/playlist?list=PLWPirh4EWFpGXTBu8ldLZGJCUeTMBpJFK

Mode of Evaluation

CIE							ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	TA3 5	Attendance 5		
50							100	150

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

Course Code: BCSAI0412	**Course Name: CRM FUNDAMENTALS**	**L**	**T**	**P**	**C**
Course Offered in: CSE/CSE(R)/ CSE(TWIN)/ IT/ IT(TWIN)/ M.Tech(Int.)		**3**	**0**	**0**	**3**

Pre-requisite: None

Course Objective: This course is designed to help in understanding the fundamentals of CRM. It will help in providing better services for Sales, Marketing and Customer Relations in an Enterprise. To make the students understand the organizational need, benefits and process of creating long-term value for individual customers. To disseminate knowledge regarding the concept of e-CRM and e-CRM technologies. To enable the students understand the technological and human issues relating to implementation of Customer Relationship Management in the organizations.

Course Outcome- After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO 1	Understand the basic concepts of Customer relationship management.	K1, K2
CO 2	To understand strategy and framework of Customer relationship management.	K2
CO 3	Learn basics of Cloud Based Customer relationship management.	K1
CO 4	Understand Customer relationship management in context with business use cases.	K2, K3
CO 5	Understand implementation basics of CRM.	K2, K3

CO-PO Mapping

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

Course Contents / Syllabus

Module 1	Introduction	**8 hours**

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

Module 2	CRM Strategy and Framework	**8 hours**

Developing a CRM strategy. Customer oriented (C in CRM), Relationship driven, 360 degree view of customer.

CRM system features- functions, application, benefits and solutions. Importance of loyalty- active, passive, split, shifting and switchers, customer profiling, customer segmentation model, Customer Experience, relationship marketing and journey, Case study.

Module 3	Solution Design and Architecture	**8 hours**

CRM system solution- specifications. Data Analysis, Solution Requirements. Types of CRM- On-Premise, cloud based. Pros and Cons of each. Integration CRM with other enterprise applications.
The Technology of CRM: Data warehouses and customer relationships, creating data mart model, components of operational data warehouse.

Module 4	CRM for Business	**8 hours**

CRM in Sales, Service, Marketing, E-commerce. Social Customer Relationship Management. Analytical CRM: Predictive Analytics Vs Operational Analytics. Channel Partner Relationship management, Collaborative CRM (using data pooling), Business Benefits of Cloud Based System, SLAs, Practical Challenges.

Module 5	CRM implementation	**8 hours**

Building CRM roadmaps: current processes, customers, strategic goals, technology issues, pilot and proof of concept projects. Preliminary Roadmap and its template, developing roadmap midstream. Design stage, custom development, integration, reporting, data migration, and implementation, testing, launching and application management. Introduction to following CRM tools: ZOHO, Pega, Microsoft Dynamics 365, Sales force.

	Total Lecture Hours	**40 hours**

Textbook:

S.No	**Book Title**
1	The CRM Handbook-A Business Guide to Customer Relationship Management by Jill Dyché; Addison-Wesley (for case studies)
2	Customer Relationship Management Systems handbook by Duane E Sharp. AUERBACH PUBLICATIONS by CRC Press Company

Reference Books

S.No	**Book Title**
1	The CRM Handbook-A Business Guide to Customer Relationship Management by Jill Dyché; Addison-Wesley (for case studies)
2	Customer Relationship Management Systems handbook by Duane E Sharp. AUERBACH PUBLICATIONS by CRC Press Company

NPTEL/ YouTube/ Faculty Video Link:

1.	https://onlinecourses.nptel.ac.in/noc20_mg57/preview
2.	https://archive.nptel.ac.in/courses/110/105/110105145/

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**TA3**	**Attendance**		
			5	**5**	**5**	**5**		
30			**20**				**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSCY0411	Course Name: Fundamentals of Cyber Security	L	T	P	C
Course Offered in: Departmental Elective		3	0	0	3

Pre-requisite: Basic knowledge of Computer Systems, Familiarity with Internet Usage and Web Browsing.

Course Objectives:

To introduce the fundamental concepts and scope of cyber security, attacks, and vulnerabilities and explore basic security mechanisms and protective technologies to prepare the students for future learning in advanced security domains.

Course Outcome: After completion of the course, the student will be able to	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.	K3
CO4: Adopt safe online behavior and promote cyber hygiene.	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	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	1	-	1	-	1	-	2	-	2	1
CO2	3	3	2	2	2	2	-	2	-	2	-	2	2	3	2
CO3	3	3	3	2	3	1	-	3	2	2	1	3	3	3	3
CO4	2	2	2	1	2	2	1	3	2	2	-	3	-	3	3

Course Contents / Syllabus

Module 1	Introduction to Cyber Security	8 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.

Module 2	Cyber Threats and Attacks	8 hours

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.

Module 3	Cyber Defense Mechanisms	8 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.

Module 4	Network & System Security Basics	6 hours

Basic Network Security Concepts: IP, MAC, Ports, Protocols (HTTP, HTTPS, FTP), **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.

Total Lecture Hours	**30 hours**

Textbook:

1. William Stallings – Cybersecurity: Principles and Practice, Pearson.
2. Chuck Easttom – Computer Security Fundamentals, Pearson.

Reference Books:

1. Fundamentals of Cyber Security, CRC Press
2. Cyber Security, Wiley India

NPTEL/YouTube/Faculty Video Link:

Unit 1	https://www.youtube.com/watch?v=z5nc9MDbvkw
Unit 2	https://nptel.ac.in/courses/106106129
Unit 3	https://www.youtube.com/watch?v=BdluJhRaAMA
Unit 4	https://nptel.ac.in/courses/106105183

Mode of Evaluation

CIE						ESE	Total
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**Attendance**		
			5	**5**	**10**		
30			**20**			**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSE0411	Course Name: PYTHON WEB DEVELOPMENT WITH DJANGO	L	T	P	C
Course Offered in: CSE/CSE(R)/ CSE(TWIN)/ IT/ IT(TWIN)/ M.Tech(Int.)		3	0	0	3

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

Course Objective: 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- After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO 1	Apply the knowledge of python programing that are vital in understanding Django application and analyze the concepts, principles and methods in current client-side technology to implement Django application over the web.	K3,K6
CO 2	Demonstrate web application framework i.e. Django to design and implement typical dynamic web pages and interactive web based applications.	K3, K6
CO 3	Implementing and analyzing the concept of Integrating Accounts & Authentication on Django.	K3, K4
CO 4	Understand the impact of web designing by database connectivity with SQLite in the current market place where everyone uses to prefer electronic medium for shoping, commerce, and even social life also.	K2, K3
CO 5	Analyzing and creating a functional website in Django and deploy Django Web Application on Cloud.	K3, K6

CO-PO Mapping

CO/PO	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	2	1
CO2	2	2	3	1	3	-	1	-	-	2	3	3	3	2
CO3	2	2	2	2	2	-	-	2	-	2	2	2	2	2
CO4	2	1	2	1	2	-	-	1	1	2	2	2	3	2
CO5	2	1	3	2	3	-	-	3	2	3	3	3	3	3

Course Contents / Syllabus

Module 1	Python libraries for web development	**8 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.

Module 2	Introduction to Django Framework	**8 hours**

Understanding Django environment, Features of Django and Django architecture, MVC and MTV, Urls 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.

Module 3	Integrating Accounts & Authentication on Django	**8 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.		
Module 4	Connecting SQLite with Django	**8 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.		
Module 5	Deploying Django Web Application on Cloud	**8 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 Lecture Hours	**40 hours**

Textbook:

S.No	Book Title
1	Martin C. Brown, "Python: The Complete Reference Paperback", 4th Edition 2018, McGraw Hill Education Publication.
2	Reema Thareja, "Python Programming: Using Problem Solving Approach", 3rd Edition 2017, Oxford University Press Publication.
3	Daniel Rubio, Apress," Beginning Django Web Application Development and Deployment with Python", 2nd Edition 2017, Apress Publication.
4	William Jordon, "Python Django Web Development: The Ultimate Django web framework guide for Beginners", 2nd Edition 2019, Kindle Edition.

Reference Books

S.No	
1	Tom Aratyn, "Building Django 2.0 Web Applications: Create enterprise-grade, scalable Python web applications easily with Django 2.0", 2nd Edition 2018, and Packt Publishing.
2	Nigel George, "Build a website with Django", 1st Edition 2019, GNW Independent Publishing Edition.
3	Ray Yao," Django in 8 Hours: For Beginners, Learn Coding Fast! 2nd Edition 2020, independently published Edition.
4	Harry Percival, "Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium, and JavaScript", 2nd Edition 2019, Kindle Edition.

	NPTEL/ YouTube/ Faculty Video Link:
1.	https://youtu.be/eoPsX7MKfe8?list=PLIdgECt554OVFKXRpo_kuI0XpUQKk0ycO https://youtu.be/tA42nHmmEKw?list=PLh2mXjKcTPSACrQxPM2_1Ojus5HX88ht7 https://youtu.be/8ndsDXohLMQ?list=PLDsnL5pk7-N_9oy2RN4A65Z-PEnvtc7rf https://youtu.be/QXeEoD0pB3E?list=PLsyeobzWxl7poL9JTVyndKe62ieoN-MZ3 https://youtu.be/9MmC_uGjBsM?list=PL3pGy4HtqwD02GVgM96-V0sq4_DSinqvf
2.	https://youtu.be/F5mRW0jo-U4 https://youtu.be/yD0_1DPmfKM?list=PLQVvvaa0QuDe9nqlirjacLkBYdgc2inh3 https://youtu.be/rHux0gMZ3Eg https://youtu.be/jBzwzrDvZ18 https://youtu.be/RiMRJMbLZmg
3.	https://youtu.be/8DF1zJA7cfc https://youtu.be/CTrVDi3tt8o

	https://youtu.be/FzGTpnI5tpo https://youtu.be/z4lfVsb_7MA https://youtu.be/WuyKxdLcw3w
4.	https://youtu.be/UxTwFMZ4r5k https://youtu.be/2Oe55iXjZQI https://youtu.be/zV8GOI5Zd6E https://youtu.be/uf2tdzh7Bq4 https://youtu.be/RzkVbz7Ie44
5.	https://youtu.be/kBwhtEIXGII https://youtu.be/Q_YOYNiSVDY https://youtu.be/_3AKAdHUY1M https://youtu.be/6DI_7Zja8Zc https://youtu.be/UkokhawLKDU

Mode of Evaluation

CIE							ESE	Total
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**TA3**	**Attendance**		
			5	**5**	**5**	**5**		
30			**20**				**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

Course Code: BASL0401N	**Course Name: Technical Communication**	**L**	**T**	**P**	
Course Offered in: B. Tech. All branches (except CSBS)		**2**	**0**	**0**	**2**

Pre-requisite: Intermediate level (CEFR) and above

Course Objectives:

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

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

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

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

Course Contents / Syllabus		
Module 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**		
Module 2	**Technical Writing 1**	**5 Hours**
Technical writing skill: characteristics, examples; **Business letters/emails**: Content organization, Tone and intent; **Agenda & Minutes of Meetings**		
Module 3	**Technical Writing 2**	**5 Hours**
Job application, **Resume'**; **Report, proposal**; **Technical paper**: Abstract; **Ethical Writing:** Copy Editing, Referencing and Plagiarism		
Module 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**		
Module 5	**Virtual/Remote Communication**	**4 Hours**
Remote work: online platforms; **Video conferencing**; **Virtual etiquette**: email ids, usernames; **Writing Blogs & creating Vlogs**		
Total Lecture Hours		**24 Hours**

Textbook:	
1	Technical Communication – Principles and Practices, 4th Edition by Meenakshi Raman & Sangeeta Sharma, Oxford Univ. Press, 2022, New Delhi.
Reference Books:	
1	Technical Communication, 15th Edition by John M. Lannon & Laura J. Gurak, Pearson, 2021
2	Spoken English- A Manual of Speech and Phonetics (5th Edition) by R K Bansal & J B Harrison, Orient Blackswan, 2024, New Delhi.
3	Business Correspondence and Report Writing by Prof. R C Sharma, Krishna Mohan, and Virendra Singh Nirban (6 Edition), Tata McGraw Hill & Co. Ltd., 2020, New Delhi.
4	Intercultural Communication in Virtual Exchange by Francesca Helm, Cambridge Univ. Press, 2024.
NPTEL/ You tube/ Faculty Video Link:	
Unit 1	https://onlinecourses.nptel.ac.in/noc24_ge37/preview
Unit 2	https://archive.nptel.ac.in/courses/109/106/109106094/

Unit 3	https://www.youtube.com/watch?v=kOJlwMJxEG0&t=8s
Unit 4	https://www.youtube.com/watch?v=Sg7Q_dC_fWU&list=PLPuC5CMHiqmuzq_KQ4aw0V9Q7xJY6aezb
Unit 5	https://www.youtube.com/watch?v=ymLFJDpjgCk&list=PLPuC5CMHiqmuzq_KQ4aw0V9Q7xJY6aezb&index=6

Mode of Evaluation

CIE						ESE	Total
ST1	**ST2**	**ST3**	**TA1 5**	**TA 2 5**	**Attendance 10**		
30			**20**			**50**	**100**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

LAB Course Code : BCSE0452Z	LAB Course Name: Database Management Systems Lab	L	T	P	C
Course Offered in: CSE/CSE-R/IT/M.Tech Int./CSE(Twin)/IT(Twin)/CSE(Prof)/IT(Prof)/M&C/AI/AI(TWIN)/ AIML/AIML(TWIN)/ CS/CYS/DS/IOT		0	0	4	2

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

Course Objectives: To familiarize the students to the basics of Database Design and Implementation.

Course Outcomes (CO)

After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Design ER/EER models to solve real-world problems and Implement them into relational schemas using appropriate database tools.	K6
CO2	Apply SQL and PL/SQL to create complex data queries, and procedural operations comprising triggers and functions, along with database connectivity.	K6
CO3	Analyze database integrity using constraints, and implement unstructured databases using MongoDB with appropriate query operations.	K4

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

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

Sr. No	Program Title
1	Understand and implement the different ER diagram notation with their relationship and Cardinalities.
2	Creating ER Diagram for company Database. Company database have entities like employee, departments, projects and dependents also implement the relationship and cardinalities between the entities with their relevant attribute.
3	Implement DDL, DML, DCL & TCL commands
4	Implementation of I/O Constraint: Primary Key, composite primary key, Foreign Key with on delete set null and on delete set null constraint, Unique Key
5	Implementation of Business Constraint: Null, Not Null, Default, Check.
6	Practicing Queries using Like, Between, Aliases, distinct Operator & Predicate. And Implement Aggregate Functions
7	Implementation of Queries using Where, Group by, Having and Order by Clause.
8	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.

	i. List the E_no, E name, Salary of all employees working for MANAGER. ii. Display all the details of the employee whose salary is more than the Sal of any IT PROFF. iii. List the employees in the ascending order of Designations of those joined after 1981. iv. List the employees along with their Experience and Daily v. List the employee who are either 'CLERK' or 'ANALYST'. vi. List the employees who joined on 1-MAY-81, 3-DEC-81, 17-DEC-81. vii. List the e_name those are starting with 'S'. viii. Display total salary spent for each job category. ix. Display lowest paid employee details under each manager. x. Display number of employees working in each department and their department name. xi. Display the details of employees sorting the salary in increasing order. xii. Show the record of employee earning salary greater than 16000 in each department. xiii. Add constraints to check, while entering the empno value (i.e) empno> 100. xiv. Define the field DEPTNO as unique. Create a primary key constraint for the column (EMPNO).
9	Implementation of Queries using set theory operators UNION, INTERSECT, MINUS.
10	Implementation of Queries using Inner Join:- Natural Join , Equi Join & Non Equi Join, Outer Join
11	Implementation of Queries nested Queries or Sub Queries: - IN, NOT IN, Exists, Not Exists, All and Any.
12	1. Apply the set theory operators, join's and nested queries on company database (Case Study-1) Write the SQL Queries for the following statement. I. Retrieve the names of employees in department 5 who work more than 10 hours per week on the 'ProductX'project. II. List the names of employees who have a dependent with the same first name as themselves. III. Find the names of employees that are directly supervised by 'Franklin Wong'. IV. For each project, list the project name and the total hours per week (by all employees) spent on that project. V. Retrieve the names of all employees who work on every project controlled by department 5. VI. Retrieve the names of all employees who do not work on every project VII. For each department, retrieve the department name, and the average salary of employees working in that department. VIII. Retrieve the average salary of all female employees. IX. Find the names and addresses of all employees who work on at least one project located in Houston but whose department has no location in Houston. X. List the last names of department managers who have no dependents. XI. Retrieve the names of all employees who work in the department that has the employee with the highest salary among all employees.
13	Understand & implement the Database Connectivity with Java/Python etc. programming language
14	Implementation and apply all the set theory operators, join and nested queries concept on Case study 1. I. Make a list of all project members for projects that involve an employee whose name is SCOTT either as a worker or as a manager of the department that controls the project. II. To retrieve the Social Security numbers of all employees who either work in department 5 or directly supervise an employee who works in department 5. III. To retrieve the SSN of all employee who work as a supervisor not a manager. IV. We want a list of all employee names as well as the name of the departments they manage if they happen to manage a department; if they do not manage one, we can indicate it with a NULL value. V. Retrieve the names of employees who have no dependents. VI. List the names of all employees with two or more dependents.

	VII. List the names of managers who have at least one dependent. VIII. Retrieve the names of all employees who do not have supervisors. IX. Retrieve the name of each employee who has a dependent with the same Last name as the employee.
15	Implementation of Indexing, Views and sequence
16	I. Write a PL/SQL Program to Add Two Numbers II. Write PL/SQL Program for Fibonacci Series III. Write PL/SQL Program to Find Greatest of Three Numbers
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.
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.
19	Create a row level trigger for the customers table that would 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:
20	Implementation of commit and rollback statement with amount transfer example.
21	Implementation array, indexing, transaction concept on Case study 1. I. Implementation of Array Functions & Operators II. Implementation of Sequence • Creating Sequences • Modifying a Sequence Definition • Removing Sequences III. Implementation of Views • Creating Simple and Complex Views • Modifying Views • Removing Views IV. Implementation of Indexes • Manual and Automatic Indexes • Creating Indexes Removing Indexes
22	Study of Open Source NOSQL Database and installation of MongoDB
23	Implementation of the MongoDB Shell commands
24	Implementation of the CRUD Operation in MongoDB
25	Implementation of Aggregate in MongoDB
26	Implementation of case Study on different domain I. E-commerce Platform II. Inventory Management III. Railway System IV. Hospital Data Management V. Voice-based Transport Enquiry System VI. SMS-based Remote Server Monitor system Banking System

Total Hours: 30 hrs.

Mode of Evaluation

CIE	PE	Total
PS	(If mentioned in curriculum)	
50	50	100

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

LAB Course Code: BCSE0451	LAB Course Name: DATA STRUCTURES AND ALGORITHMS –II LAB	L	T	P	C
		0	0	2	1

Course Offered in: CSE/CS/CSR-R/M.TECH(INT) /IT//CSE(AI)/CSE(AIML)/CSE(DS)/CSE(CS)

Pre-requisite: C, Python

Course Objectives:

1. Learn to implement non-linear data structures.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Implementation of tree data structures for basic operations like insertion, deletion, searching and traversal	K3
CO2	Implementation of algorithms based on graph data structures for solving real world problems.	K3
CO3	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	-	3	2	1
CO2	3	3	2	1	1	2	1	1	2	1	-	3	3	2
CO3	3	3	2	2	2	2	1	1	2	2	-	3	3	2

S.No.	Program Title
1	Write a program to implement an in-order traversal of a binary tree and print the nodes.
2	Write a program to implement a pre-order traversal of a binary tree and print the nodes.
3	Write a program to implement a post-order traversal of a binary tree and print the nodes.
4	Write a program to count number of nodes in a binary tree
5	Write a program to find the height of the tree
6	Write a program to check if the Binary tree is balanced or not.
7	Write a Program to search a number in Binary Search Tree (BST)

8	Write a program to insert a node in a Binary Search Tree (BST).
9	Write a program to delete a node from a Binary Search Tree (BST).
10	Write a program to implement a max-heap and perform heap sort on an array of integers.
11	Write a Program to implement human coding algorithm
12	Write a program to implement priority queue using max heap.
13	Write a program to create a graph using an adjacency matrix.
14	Write a program to create a graph using an adjacency list.
15	Write a program to perform Depth-First Search (DFS) on a graph.
16	Write a program to perform Breadth-First Search (BFS) on a graph.
17	Write a program to check if there is a path between two nodes in a graph using DFS.
18	Write a program to find all the vertices reachable from a given vertex in a graph using BFS.
19	Write a program to detect a cycle in an undirected graph using DFS.
20	Write a program to detect a cycle in a directed graph using DFS.
21	Write a program to find the degree of each vertex in an undirected graph.
22	Write a program to count the number of connected components in an undirected graph.
23	Write a program to implement Dijkstra Algorithm.
24	Write a program to implement Prims Algorithm.
25	Write a program to implement Kruskal Algorithm.
26	Write a program to implement Floyd Warshall's all pair shortest path algorithm.
27	Write a program to implement Bellman ford Algorithm.
28	Write a program to implement Longest common subsequence (LCS).
29	Write a program to implement sum of subset problem using backtracking.
30	Write a program to implement insertion and search operations in a Tree.

Total Hours: 30 hrs.

Mode of Evaluation

CIE	PE	Total
PS	**(If mentioned in curriculum)**	
25	**25**	**50**

NIET Greater Noida Autonomous Institute	**NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY** **GREATER NOIDA-201306** **(An Autonomous Institute)** **School of Computer Science in Emerging Technologies**

Course Code: BCSE0455	**Course Name: Web Technologies**	**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:

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

Course Objectives:

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

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

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

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

Course Contents / Syllabus

Unit 1	**Basics of Web Technology & Testing**	**10 hours**

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

Unit 2	**Introduction to HTML & XML**	**14 hours**

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

Unit 3	**Concepts of CSS3 & Bootstrap**	**16 hours**

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

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

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

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

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

Total Lecture Hours	**72 hours**

Textbook:

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

Reference Books:

1	Burdman, Jessica, "Collaborative Web Development" 5th Edition 1999, Addison Wesley Publication.
2	Randy Connolly, "Fundamentals of Web Development",3rd Edition 2016
3	Ivan Bayross," HTML, DHTML, Java Script, Perl & CGI", 4th Edition 2010 BPB Publication

NPTEL/ Youtube/ Faculty Video Link:

Unit 1	https://youtu.be/96xF9phMsWA https://youtu.be/Zopo5C79m2k https://youtu.be/ZliIs7jHi1s https://youtu.be/htbY9-yggB0
Unit 2	https://youtu.be/vHmUVQKXlVo https://youtu.be/qz0aGYrrlhU https://youtu.be/BsDoLVMnmZs https://youtu.be/a8W952NBZUE
Unit 3	https://youtu.be/1Rs2ND1ryYc https://youtu.be/vpAJ0s5S2t0 https://youtu.be/GBOK1-nvdU4 https://youtu.be/Eu7G0jV0ImY
Unit 4	https://youtu.be/-qfEOE4vtxE https://youtu.be/PkZNo7MFNFg https://youtu.be/W6NZfCO5SIk https://youtu.be/DqaTKBU9TZk
Unit 5	https://youtu.be/_GMEqhUyyFM https://youtu.be/ImtZ5yENzgE https://youtu.be/xIApzP4mWyA https://youtu.be/qKR5V9rdht0

Mode of Evaluation

CIE	ESE	

ST1	ST2	ST3	TA1 5	TA2 5	TA3 5	Attendance 5		
30			20				100	

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

Sr. No.	Program Title
1	A.Overview and Installation of various code editors.
2	B. Overview and Installation of various servers
3	Implementing HTML program that represent in the document as a start tag, which gives the name and attributes
4	Implementing HTML program that represents a document
5	Implementing HTML program to display your simple CV
6	Creating html document that represents document object model
7	To Create a table to show your class time table.
8	Apply various colors to suitably distinguish keywords , also apply font styling like italics, underline and two other fonts to words you find appropriate , also use header tags.
9	Create a webpage with HTML describing your department use paragraph and list tags
10	Implementing HTML program that for Heading
11	Implementing program that implement paragraph and line-break
12	Use tables to provide layout to your HTML page describing your college infrastructure.
13	Use and tags to provide a layout to the above page instead of a table layout
14	Create links on the words e.g. ―Wi-Fi and ―LAN to link them to Wikipedia pages
15	Insert an image and create a link such that clicking on image takes user to other page
16	Change the background color of the page; At the bottom create a link to take user to the top of the page.
17	Creating HTML program to implement three articles with independent, self-contained content.
18	Creating a XML document that defines the self-descriptive tags
19	Designing XML document that store various book data such as: book category, title, author, year and price

20	To Describe the various types of XML key components
21	Design XML DTD to define the structure and legal element and attribute of XML document
22	To implement internal and external DTD
23	Use frames such that page is divided into 3 frames 20% on left to show contents of pages, 60% in center to show body of page, remaining on right to show remarks.
24	Design a HTML registration form that takes user name, user password and mobile number with submit button control
25	Design a HTML5 document that implement of date, number, range, email, search and data list.
26	Implementation in HTML5 that include native audio and video support without the need for Flash.
27	Create a simple form to submit user input like his name, age, address and favourite subject, movie and singer.
28	Add few form elements such as radio buttons, check boxes and password field. Add a submit button at last.
29	Add CSS property assign a style or behavior to an HTML element such as: color, border, margin and font-style.
30	Add To Style Text Elements with Font, Size, and Color in CSS
31	Applying a block element in CSS acquires up the full width available for that content.
32	Demonstrating the CSS Box model with consists of: borders, padding, margins, and the actual content.
33	Design a web page by applying CSS grouping and dimensions property.
34	Design a XML Schema that describes the structure of an XML document.
35	Design a XML document that describe the well-formed XML document
36	Design a XML document of CD Catalog through each <CD> element, and displays the values of the <ARTIST> and the <TITLE> elements in an HTML table
37	Create a XSL document for and taken xml document by you.
38	Create a XSLT document for and taken xml document by you with all steps
39	Design a web page by applying CSS Display and Positioning property.
40	Design a web page by applying CSS Display and Positioning property .

41	Design a web page by applying CSS pseudo classes.
42	Creating a Java Script code to implement all data types.
43	Design a basic structure of Bootstrap Grid system.
44	Design All Bootstrap Components with example.
45	Implementing a program in Java script to implement augmented function.
46	Implementing a program to implement calculator application as real time.
47	Design a HTML form validation using Java Script.
48	Write a program to implement Arrow function with default argument in ES6
49	Implementing a program in ES6 to implement Template string concepts
50	Implementing a program in ES6 to implement all string methods.
51	Creating a Java Script program to implement Dialog, Confirm and Message Popup Boxes.
52	Implementing a Java Script program to implement onClick and onSubmit event
53	Creating a java script code to implement ‘let’ keyword
54	Creating a java script code to implement ‘const’ keyword
55	Implementing a program to implement call back functions in ES6.
56	Implementing a program for de-structuring of an array in ES6
57	Javascript code to implement object and class concepts in Typescript.
58	Write a Typescript program that implement interface and constructor
59	Write a code in typescript that implement decorator and spread operator
60	Create a constant by using define() function with its proper syntax
61	Creating PHP script that return any data types whatever you use.
62	Implementing a code in Java Script to implement Spread and rest operator
63	Javascript code that should compile by Typescript compiler as’tsc’
64	Write a code in typescript that implement Asynchronous Programming concepts.
65	Write a program in Typescript that implement promise constructor

66	Implementing promise and chain concepts in Typescript
67	Write a code in typescript that implement Promise.race() static method.
68	Crating a program that implement control flow and decision making statement.
69	Creating PHP to implements parameterized function
70	Creating program in PHP to store multiple string and concatenate these string and print it.
71	Write a PHP script to create and delete directory structure
72	Program to upload and download a file in PHP
73	Implements single dimension array in PHP
74	Write a PHP code to open and close a file in a proper manner
75	Write a PHP script to copying, renaming and deleting a file.
76	PHP program to create and destroy a session.
77	PHP program to set and delete a cookie.
78	PHP program to manually register the session variable
79	PHP program to manually destroy the session variable
80	PHP program to store the session data on one page and would be available on second page.

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technology

LAB Course Code: BCSCC0452	**LAB Course Name: Problem Solving Approaches**	**L**	**T**	**P**	**C**
Course Offered in:		**0**	**0**	**2**	**1**

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

Course Objectives:

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

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

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

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

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

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

1. **Secure Password Generator**

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

   1. Prime Number Validation:

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

   2. Sum of Digits Check:

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

3. Armstrong Number Check:

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

Password Generation:

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

Example Scenario:

Sample Input

Enter a 3-digit prime number: 153

Sum of digits of 153 = 9

The sum is divisible by 3.

153 is Armstrong number

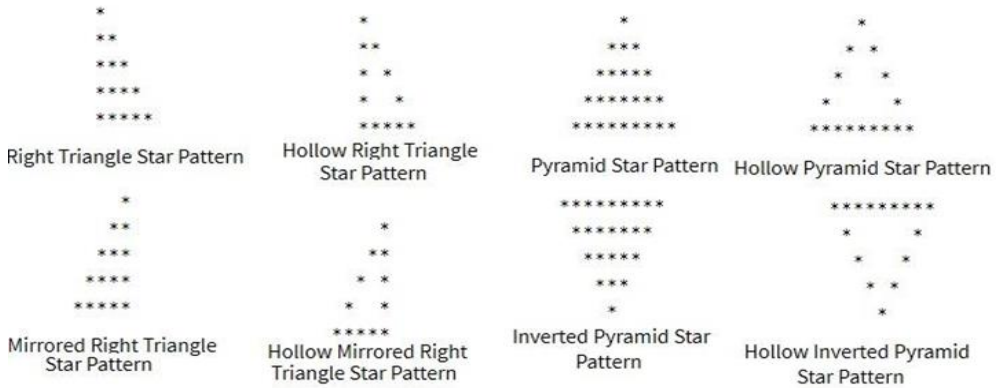
Sample Output

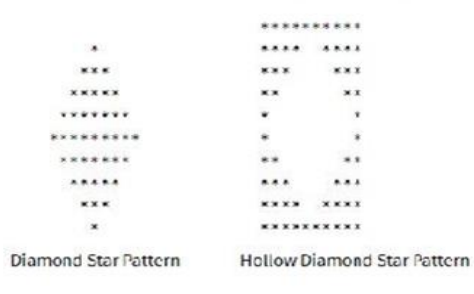
Secure Password: 19

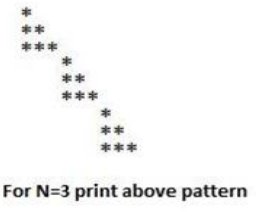
2. Write a function to input electricity unit charges and calculate total electricity bill according to the given condition:
For first 50 units Rs. 0.50/unit
For next 100 units Rs. 0.75/unit
For next 100 units Rs. 1.20/unit
For unit above 250 Rs. 1.50/unit
An additional surcharge of 20% is added to the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

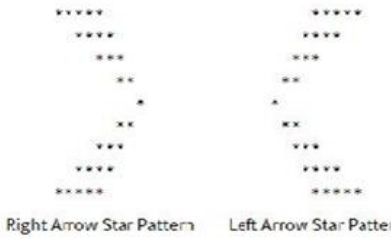
3. Write a method to generate a secure code which the sum of all possible palindrome numbers between given two numbers.
For Example:
Input: 10, 80
Output: 308
Explanation: All palindrome numbers between 10 & 80 are: 11,22,33,44,55,66,77
Password= 11+22+33+44+55+66+77 = 308

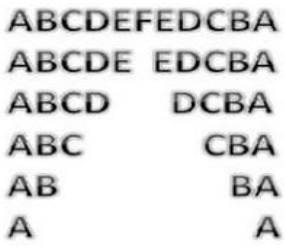
4. Draw **the following Patterns for N=5**











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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Scenario:

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

For example:

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

Output: Password: 971

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

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

- Original Maximum Subarray Sum: 6 (achieved from [4, -1, 2, 1])
- Maximum Sum After Modification: 10(achieved from [8, -1, 2, 1], where the value 4 is doubled to 8).

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] = target and i < j < k.
Examples:
Input: arr[] = [-3, -1, -1, 0, 1, 2], target = -2
Output: 4

5. 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 (Dutch National Flag algorithm).
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)

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

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

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

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

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

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

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

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.***

Total Hours: 30 hrs.

Mode of Evaluation

CIE			**PE**	**Total**
PS1 **10**	**PS2** **20**	**PS3** **20**	**(If mentioned in curriculum)**	
50				**50**

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

Course Code: BNC0401	Course Name: Artificial Intellgence and Cyber Ethics	L	T	P	C
Course Offered in: B. Tech.		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

Module 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 Cyber laws, Legal frameworks.

Module 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.

Module 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.

Module 4	**Cybercrimes, IPR and Legal Measures**	**8 hours**

Types of Cybercrimes and their Impact, Legal measures for Cybercrime Prevention and Prosecution, IPR: Copyrights, Trademarks, Patents, and Trade Secrets, Ethical Implications of Intellectual Property, Cyber Security and Privacy Issues, Cyber Crime Investigations and Digital Evidence Handling, Overview of Indian Cyber Laws (IT Act 2000 and Amendments), Comparative Overview: Indian vs Global Cyber Laws, Case Study: The ATM Heist – Cosmos Bank Cyber Attack (India, 2018).

	Total Lecture Hours	**30 hours**

Textbook:

1. Artificial Intelligence: A Guide for Thinking Humans by Melanie Mitchell, Penguin Books, 2019.
2. Cyber Ethics: Morality and Law in Cyberspace by Richard Spinello, Jones & Bartlett Learning, 7th Edition (2023).

Reference Books:

1. 2.	Artificial Intelligence and Ethics by S. B. Kishor, Debajit Biswas, BPB Publications, 2023. Cyber Security and Cyber Laws by Alfred Basta, Nadine Basta, Sattwik Panda, Cengage India, 2022.	

NPTEL/ YouTube/ Faculty Video Link:

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

Mode of Evaluation

CIE							ESE	Total
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**TA3** **5**	**Attendance** **5**		
30			**20**				**50**	**100**