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

Affiliated to

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

Evaluation Scheme & Syllabus

For

Bachelor of Technology
Computer Science and Engineering (Artificial Intelligence)
Second Year

(Effective from the Session: 2024-25)

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

Bachelor of Technology

Computer Science and Engineering (Artificial Intelligence)

Evaluation Scheme

SEMESTER-III

S. No.	Subject Codes	Subject	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	Total	PS	TE	PE		
3 WEEKS COMPULSORY INDUCTION PROGRAM														
1	BAS0303	Statistics and Probability	Mandatory	3	1	0	30	20	50		100		150	4
2	BCSE0306	Discrete Structures	Mandatory	3	1	0	30	20	50		100		150	4
3	BCSAI0301	Artificial Intelligence and Machine Learning	Mandatory	3	0	0	30	20	50		100		150	3
4	BCSE0301	Data Structures and Algorithm-I	Mandatory	3	0	0	30	20	50		100		150	3
5	BCSAI0302	Logic Design and Computer Architecture	Mandatory	3	0	0	30	20	50		100		150	3
6	BCSE0352	Object Oriented Techniques using Java	Mandatory	0	0	6				50		100	150	3
7	BCSAI0351	Artificial Intelligence and Machine Learning Lab	Mandatory	0	0	2				25		25	50	1
8	BCSE0351	Data Structures and Algorithm-I Lab	Mandatory	0	0	4				50		50	100	2
9	BCSE0359	Internship Assessment- I	Mandatory	0	0	2				50			50	1
10	BNC0301/ BNC0302	Artificial Intelligence and Cyber Ethics / Environmental Science	Compulsory Audit	2	0	0	30	20	50		50		100	NA
		*Massive Open Online Courses (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL											**1100**	**24**

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

S.No.	Subject Code	Course Name	University / Industry Partner Name	No of Hours	Credits
1	BMC0008	Object Oriented Programming Using Python	Infosys Wingspan (Infosys Springboard)	46 h 13 m	3.5
2	BMC0009	Probability and Statistics using Python	Infosys Wingspan (Infosys Springboard)	16 h	1

PLEASE NOTE: -

- **A 3-4 weeks 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 in 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 and Engineering (Artificial Intelligence)
Evaluation Scheme
SEMESTER-IV

S. No.	Subject Codes	Subject	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	Total	PS	TE	PE		
1	BAS0404	Optimization and Numerical Techniques	Mandatory	3	1	0	30	20	50		100		150	4
2	BASL0401	Technical Communication	Mandatory	2	1	0	30	20	50		50		100	3
3	BCSE0401	Data Structures and Algorithm-II	Mandatory	3	0	0	30	20	50		100		150	3
4	BCSE0404	Theory of Automata and Formal Languages	Mandatory	3	0	0	30	20	50		100		150	3
5	BCSE0403	Operating Systems	Mandatory	2	0	0	30	20	50		50		100	2
6	BCSE0452	Database Management Systems	Mandatory	0	0	6				50		100	150	3
7	BCSE0451	Data Structures and Algorithm-II Lab	Mandatory	0	0	4				50		50	100	2
8	BCSE0453	Operating Systems Lab	Mandatory	0	0	4				50		50	100	2
9	BASL0451	Technical Communication Lab	Mandatory	0	0	2				25		25	50	1
10	BCSE0459	Mini Project	Mandatory	0	0	2				50			50	1
11	BNC0402/ BNC0401	Environmental Science/ Artificial Intelligence and Cyber Ethics	Compulsory Audit	2	0	0	30	20	50		50		100	NA
		*Massive Open Online Courses (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL											**1100**	**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	BMC0010	Comprehensive Training on Unix and Linux OS Fundamentals	Infosys Wingspan (Infosys Springboard)	30 h 13 m	2
2	BMC0011	Building Machine Learning Systems with Tensor Flow	Infosys Wingspan (Infosys Springboard)	27 h 18 m	2

PLEASE NOTE: -

- **A 3-4 weeks Internship shall be conducted during summer break after semester-IV and will be assessed during Semester-V**
- **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 in 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)

A student will be eligible to get Under Graduate 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

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

Subject Name: Statistics and Probability	**L-T-P [3-1-0]**
Subject Code: BAS0303	**Applicable in Department: DS/AI/AIML/CYS**

Pre-requisites of the Subject: Knowledge of Mathematics I and II of B. Tech or equivalent.

Course Objective: The objective of this course is to familiarize the engineers with concept of Statistical techniques, probability distribution, hypothesis testing and ANOVA and numerical aptitude. It aims to show case the students with standard concepts and tools from B. Tech to deal with advanced level of mathematics and applications that would be essential for their disciplines.

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO 1	Understand the concept of moments, skewness, kurtosis, correlation, curve fitting and regression analysis.	K2, K3
CO 2	Understand the concept of Probability and Random variables.	K2, K3
CO 3	Remember the concept of probability to evaluate probability distributions	K1, K3
CO 4	Apply the concept of hypothesis testing and estimation of parameter.	K2
CO 5	Solve the problems of Time & Work, Pipe & Cistern, Time, Speed & Distance, Boat &Stream, Analogy.	K3

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping

1	Descriptive measures	Measures of central tendency – mean, median, mode, measures of dispersion – mean deviation, standard deviation, quartile deviation, variance, Moment, Skewness and kurtosis, least squares principles of curve fitting, Covariance, Correlation and Regression analysis, Correlation coefficient: Karl Pearson coefficient, rank correlation coefficient, uni-variate and multivariate linear regression, application of regression analysis, time series analysis- Trend analysis (Least square method). Applications in Engineering.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment 1.1	CO1
2	Probability and Random variable	Probability Definition, The Law of Addition, Multiplication and Conditional Probability, Bayes' Theorem, Random variables: discrete and continuous, probability mass function, density function, distribution function, Mathematical expectation, mean, variance. Moment generating function, Two dimensional random variables: probability mass function, density function. Applications in Engineering.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-2.1	CO2
3	Probability distribution	Probability Distribution (Continuous and discrete- Normal, Exponential, Binomial, Poisson distribution), Central Limit theorem (Statement). Applications in Engineering.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-3.1	CO3
4	Test of Hypothesis & Statistical Inference	Statistical Inference, Parameter estimation, Maximum Likelihood estimation. Sampling and population, uni-variate and bi-variate sampling, re-sampling, errors in sampling, Sampling distributions, Hypothesis testing- p value, z test, t test (For mean), Confidence intervals, F test; Chi-square test, ANOVA: One-way ANOVA. Applications in Engineering.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-4.1	CO4

5	Aptitude-III	Number System, Permutation & Combination, Probability, Set theory, Function, Non Verbal Reasoning, Data Interpretation, Syllogism.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-5.1	CO5
Total				**40 Hours**		

Textbooks	
Sr No	**Book Details**
1	P. G. Hoel, S. C. Port and C. J. Stone, Introduction to Probability Theory, Universal Book Stall, 2003(Reprint)
2	S. Ross: A First Course in Probability, 6th Ed., Pearson Education India, 2002
3	W. Feller, An Introduction to Probability Theory and its Applications, Vol. 1, 3rd Ed., Wiley, 1968.
Reference Books	
Sr No	**Book Details**
1	B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
2	T.Veerarajan : Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi
3	R.K. Jain and S.R.K. Iyenger: Advance Engineering Mathematics; Narosa Publishing House, New Delhi.
4	J.N. Kapur: Mathematical Statistics; S. Chand & Sons Company Limited, New Delhi.
Links	
Unit 1	**https://archive.nptel.ac.in/courses/111/105/111105042/** **https://archive.nptel.ac.in/courses/110/107/110107114/**

Unit 2	**https://archive.nptel.ac.in/courses/111/102/111102111/**
Unit 3	**https://archive.nptel.ac.in/courses/111/104/111104032/**
Unit 4	**https://archive.nptel.ac.in/courses/103/106/103106120/**
Unit 5	**https://nptel.ac.in/courses/111107058** **https://archive.nptel.ac.in/courses/127/106/127106227/** **https://archive.nptel.ac.in/courses/111/102/111102111/**

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

Subject Name: Discrete Structures	**L-T-P [3-1-0]**
Subject Code: BCSE0306	**Applicable in Department: All Branches**

Pre-requisite of Subject: Some basic knowledge of algebra and logic is usually sufficient to begin studying discrete mathematics for computer science. Familiarity with sets, functions, and basic Boolean algebra is also helpful.

Course Objective: The objective of discrete structure is to enable students to formulate problems precisely, solve the problems, apply formal proofs techniques and hence enhance one's logical thinking and problem-solving skills.

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO1	Apply the basic principles of sets, relations & functions and mathematical induction in computer science & engineering related problems.	K3
CO2	Describe the algebraic structures and its properties to solve complex problems.	K2
CO3	Describe lattices and it's type to simplify digital circuits.	K2
CO4	Infer the validity of statements and construct proofs using predicate logic formulas.	K4
CO5	Design and use non-linear data structure like graphs to solve real world problems.	K4

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping
1 Set Theory	Module 1.1: Set Theory	Set Theory: Definition of sets, countable and uncountable sets, Set operations, Partition of set, Cardinality, Venn Diagrams, proofs of some	Lecture Notes, PPT, Online	8 L	NA	CO1

& Relations		general identities on sets, Applications of set Theory	Videos & R2			
	Module 1.2: Relations	Relation: Definition, types of relation, composition of relations, Equivalence relation, Partial ordering relation, Applications of Relations	Lecture Notes, PPT, Online Videos & R2			
2 Algebraic Structures	Module 2.1: Algebraic Structures	Definition, Properties, types: Semi Groups, Monoid, Groups, Abelian group, Properties of groups, Subgroup, cyclic group, Permutation group, Cosets, Normal subgroup, Homomorphism and isomorphism of Groups, Applications of Algebraic Structure	Lecture Notes, PPT, Online Videos & R2	8 L	NA	CO2
3 Posets, Hasse Diagram and Lattices	Module 3.1: Posets, Hasse Diagram and Lattices:	Introduction, ordered set, Hasse diagrams of partially ordered set, isomorphic ordered set, well ordered set, properties of lattices, types of lattices, Applications of Lattice	Lecture Notes, PPT, Online Videos & R2	8 L	NA	CO3
4 Propositional & Predicate Logic	Module 4.1: Propositional Logic	Propositions and compound Propositions, Basic logical operations, truth tables, tautologies, Contradictions, CNF, DNF Algebra of Proposition, logical implications, logical equivalence, predicates and quantifiers, Rules of Inference, Application of Propositional Logics.	Lecture Notes, PPT, Online Videos & R1	8 L	NA	CO4
	Module 4.2: Predicate Logic	First order predicate, Well-formed formula of Predicate, Quantifiers, Inference Theory of Predicate Logic, Application of Predicate Logics.	Lecture Notes, PPT, Online Videos & R2			
5 Graphs	Module 5.1: Graphs	Definition and terminology, Representation of Graphs, Paths connectivity, Walks, Paths, Cycles, Bipartite, Regular, Planar and connected graphs, Components, Euler graphs, Euler's theorem, Hamiltonian path and circuits, Graph coloring, chromatic number, isomorphism and homomorphism of graphs.	Lecture Notes, PPT, Online Videos & R2	8 L	NA	CO5

		Application of Graphs				
Total				**40 Hours**		

Textbooks	
Sr. No.	**Book Details**
1	Swapanm Kumar Sarkar, “A Textbook of Discrete Mathematics”, S. Chand Publication, 9th Edition, 2021
2	T Veerarajan, “Discrete Mathematics, with Graph Theroy and Combinatorics” TMH Publication, 4th Edition, 2021
Reference Books	
Sr. No.	**Book Details**
1	B. Kolman, R.C. Busby, and S.C. Ross, Discrete Mathematical Structures, Prentice Hall, 6th Edition, 2020.
2	Liptschutz, Seymour, “Discrete Mathematics”, TMH, 4th Edition, 2021.
3	Kenneth H. Rosen, Kamala Krithivasan, “Discrete Mathematics and its Applications”, TMH, 8th Edition, 2021
Links	
Unit 1	https://www.youtube.com/watch?v=hGtOLG3SsjI&list=PLwdnzlV3ogoVxVxCTlI45pDVM1aoYoMHf&index=9 https://www.youtube.com/watch?v=rGcTcGFx9_s&list=PLwdnzlV3ogoVxVxCTlI45pDVM1aoYoMHf&index=10 https://www.youtube.com/watch?v=_BIKq9Xo_5A&list=PL0862D1A947252D20&index=13
Unit 2	https://www.youtube.com/watch?v=dQ4wU0k7JKI&list=PL0862D1A947252D20&index=35 https://www.youtube.com/watch?v=CjmWE-f3vEc&list=PLwdnzlV3ogoVxVxCTlI45pDVM1aoYoMHf&index=41
Unit 3	https://www.youtube.com/watch?v=qPtGlrb_sXg&list=PL0862D1A947252D20&index=40
Unit 4	https://www.youtube.com/watch?v=xlUFkMKSB3Y&list=PL0862D1A947252D20&index=1 https://www.youtube.com/watch?v=DmCltf8ypks&list=PL0862D1A947252D20&index=3
Unit 5	https://www.youtube.com/watch?v=E40r8DWgG40&list=PLEAYkSg4uSQ2fXcfrTGZdPuTmv98bnFY5

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

Subject Name: Artificial Intelligence and Machine Learning		**L-T-P [3-0-0]**
Subject Code: BCSAI0301		**Applicable in Department: AI/AIML/CYS**
Pre-requisite of Subject: Statistics & Probability, Python		
Course Objective: This course focuses on applying artificial intelligence algorithms to real-world scenarios and designing machine learning algorithms, optimizing models, and reporting their expected accuracy.		
Course Outcomes (CO)		
Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level (KL)**
CO 1	Apply the most suitable search algorithm for a given problem to find the goal state.	K3
CO2	Use feature engineering and data visualization concepts.	K3
CO3	Analyze the strengths and weaknesses of various regression and classification algorithms.	K4
CO4	Apply approaches that incorporate appropriate clustering algorithms to solve a specific data clustering problem.	K3
CO5	Analyze the ensemble learning techniques, probabilistic learning methods, and reinforcement learning algorithms to enhance model performance.	K4
Syllabus		

Unit No	Module Name	Topics Covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping
1	Introduction to Aland problem-solving methods	Introduction to AI and Intelligent agent, Different Approaches of AI, Problem Solving by searching Techniques: Uninformed search- BFS, DFS, Iterative deepening, Bidirectional search, Informed search- Iterative deepening,Bidirectional search, Heuristic search, Greedy BestFirst Search, A* search, Local Search Algorithms- Hill Climbing and Simulated Annealing Adversarial Search, Game Playing- minimax, alpha-beta pruning, constraint satisfaction problems	Lectures, Lab Cum Class (LCC)Mode, Hands-on Exercises	8L+6P	Lab No. 1 to 5	CO1
		Introduction to Machine Learning, Types of Machine Learning,				
2	Machine Learningand Feature Engineering	Feature Engineering: Features and their types, handing missing data, Dealing with categorical features, Working with features: Feature Scaling, Feature selection, Feature Extraction: Principal Component Analysis(PCA) algorithm	Lectures, Lab Cum Class (LCC)Mode, Hands-on Exercises	4L+2P	Lab No. 6	CO2
3	Supervised-Learning	Regression & Classification: Types of regression (Univariate, Multivariate, Polynomial), Mean Square Error, R square error, Logistic Regression. Regularization: Bias and Variance, Overfitting and Under fitting, L1 and L2 Regularization, Regularized Linear Regression, Decision Trees (ID3, C4.5, CART), Confusion matrix, k-folds cross- validation, K Nearest Neighbor, Support vector machine.	Lectures, Lab Cum Class (LCC)Mode, Hands-on Exercises	10L+10P	Lab No. 7 to 19	CO3
4	Unsupervised Machine Learning	Introduction to clustering, Types of clustering:K-means clustering, K-mode, K-medoid, hierarchical clustering, single-linkage, multiple linkage, AGNES and DIANA algorithms, Gaussian mixture models density based clustering, DBSCAN	Lectures, Lab Cum Class (LCC)Mode, Hands-on Exercises	6L+9P	Lab No. 20 to26	CO4

5	Ensemble & Reinforcement Learning	Probabilistic learning: Bayesian Learning, Naïve Bayes Classifier,Bayesian belief networks Ensembles Learning Random Forest, Gradient Boosting, XGBoost. Reinforcement Learning: Introduction to reinforcement learning, models of reinforcement learning: Markov decision process, Q-learning.	Lectures, Lab Cum Class (LCC)Mode, Hands-on Exercises	8L+9P	Lab No. 27 to38	CO5
Total				**(36L+36P) = 72 Hours**		

Textbooks

Sr No	Book Details
1	Artificial Intelligence: A Modern Approach (4th Edition) by Stuart Russell and Peter Norvig 2020.
2	Marco Gori, Machine Learning: A Constraint-Based Approach, Morgan Kaufmann. (2nd Edition) 2023.
3	Ethem Alpaydin, Machine Learning: The New AI, MIT Press-2021
4	Tom M. Mitchell, “Machine Learning”, McGraw-Hill, 2019

Reference Books

Sr No	Book Details
1	Ryszard, S., Michalski, J. G. Carbonell and Tom M. Mitchell, Machine Learning: An Artificial Intelligence Approach, Volume 1, and Elsevier (2014)
2	Stephen Marsland, Taylor & Francis, Machine Learning: An Algorithmic Perspective, 2nd edition (2014)
3	Ethem Alpaydin, 4th edition (2020) “Introduction to Machine Learning (Adaptive Computation and MachineLearning)”, The MIT Press.

Links

Unit 1	https://www.youtube.com/watch?v=XCPZBD9lbVo&list=PLbMVogVj5nJSFZoiF6RDqyz_m6Srjx_MY
Unit 2	https://www.youtube.com/watch?v=T3PsRW6wZSY&list=PLJ5C_6qdAvBGaabKHmVbtryZW9KpICiHC
Unit 3	https://www.youtube.com/watch?v=8PJ24SrQqy8&list=PLJ5C_6qdAvBGaabKHmVbtryZW9KpICiHC&index=6
Unit 4	https://www.youtube.com/watch?v=PNglugooJUQ&list=PLJ5C_6qdAvBGaabKHmVbtryZW9KpICiHC&index=13
Unit 5	https://www.youtube.com/watch?v=YaPSPu7K9S0&list=PLEAYkSg4uSQ0Hkv_1LHlJtC_wqwVu6RQX

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

Subject Name: **Data Structures and Algorithms-I**	**L-T-P [3-0-0]**
Subject Code: BCSE0301	**Applicable in Department: CSE/IT/CS/AI/AIML/IOT/ DS/CYS**
Pre-requisite of Subject: C, Python	
Course Objective: The objective of the course is to learn the basic concepts of algorithm analysis, along with implementation of linear data structures.	

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO 1	Understand the concept of algorithm analysis and its importance for problem solving.	K2
CO2	Implementation of Arrays for searching, sorting and hashing to foster critical thinking.	K3
CO3	Compare and contrast linked list with arrays and implementation of linked list with its applications.	K4
CO4	Understand static and dynamic implementation of stacks, while mastering principle of recursion for effective problem-solving.	K3
CO5	Implementation and analysis of divide & conquer algorithms and greedy approach for efficient problem-solving across diverse contexts.	K3

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping
1 Introduction to Data Structure	Module 1.1: Foundation of Algorithms Analysis and Design	Algorithms, Analyzing Algorithms, Complexity of Algorithms, Amortized Analysis, Growth of Functions, Methods of solving Recurrences, Performance	Lectures, Code Walkthroughs, Hand-on Programming, Problem Solving, Collaborative Learning,	8L+6P	Program to compare the time complexities of various	CO1

and Algorithms		Measurements Time and Space Complexity of an algorithm, Asymptotic notations (Big Oh, Big Theta and Big Omega), Abstract Data Types (ADT).	competitive coding Projects, Assessments. Lectures, Problem Solving, Collaborative Learning, Assessments		algorithms by plotting the graph	
	Module 1.2: Fundamentals of D.S.	Data types: Primitive and non-primitive, Introduction to Data structure, Types of Data Structures- Linear & Non-Linear Data Structures.				
2 Design and Analysis of Algorithms: Arrays , searching and sorting, Hashing	Module 2.1: Arrays	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.	Lectures, Code Walkthroughs, Hand-on Programming, Problem Solving, Collaborative Learning, competitive coding, Projects, Assessments	8L+12P	Implementation of Arrays, Row Major Order, and Column Major Order, Representation of sparse matrix, Linear search, Binary search.	CO2
	Module 2.2: Searching and Sorting	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.				
	Module 2.3: Hashing	Hashing: The symbol table, Hashing Functions, Collision-Resolution Techniques, hashing for direct files.				
3 Design and Analysis of Algorithms: Linked lists Data Structure	Module 3.1: Linked List	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	Lectures, Code Walkthroughs, Hand-on Programming, Problem Solving, Collaborative Learning, competitive coding, Projects, Assessments.	8L+12P	Operations on a Linked List: Insertion, Deletion, Traversal, Reversal, Searching	CO3

4 Design and Analysis of Algorithms based: Stacks Data Structure and Recursion	Module 4.1: Stacks Module 4.2: Recursion Module 4.3: Queue	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	Lectures, Code Walkthroughs, Hand-on Programming, Problem Solving, Collaborative Learning, Projects, Assessments.	8L+12P	Operations on stacks and question. Recursion Application	CO4
5 Design and Analysis of Algorithms: Queues Data Structure	Module 5.1: Divide and Conquer and Greedy Methods	Divide and Conquer concepts with Examples Such as Quick sort, Merge sort, Convex Hull. Greedy Methods with Examples Such as Activity Selection, Task Scheduling, Fractional Knapsack Problem.	Lectures, Code Walkthroughs, Hand-on Programming, Problem Solving, Collaborative Learning, Projects, Assessments.	8L+6P	Divide and conquer methods and greedy methods	CO5
Total No. of Lecture + Practical Labs			**(40L+48P) = 88 Hours**			

Textbooks	
Sr. No.	**Book Details**
1	Michael T. Goodrich, Roberto Tamassia, "Data Structures and Algorithms in Python: An Indian Adaptation", 1st Edition, 2021.
2	Horowitz and Sahani, “Fundamentals of Data Structures”, Computer Science Press, 1st Edition, 1993.
3	Lipschutz, “Data Structures” Schaum’s Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd, 2nd Edition, 2017

Reference Books	
Sr. 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.

Links	
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

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

Subject Name: Logic Design and Computer Architecture	**L-T-P [3-0-0]**
Subject Code: BCSAI0302	**Applicable in Department: AI/AIML/DS/CS/CYS**
Pre-requisite of Subject: 1. Basic knowledge of computer systems. 2. Logic gates and their operations.	
Course Objective: To understand the types of organizations, structures, and functions of computers, design of arithmetic and logic units, and float point arithmetic. To understand the concepts of the memory system, communication with I/O devices, and interfaces.	

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level (KL)**
CO1	Explain the basics of Digital Logic Fundamentals	K1, K2
CO2	Analyze the Functional units of a computers	K2, K3
CO3	Implement the Arithmetic Logic and Control Units	K2, K4
CO4	Understand the basic of Memory Organization	K2, K4, K5
CO5	Explain different ways of communicating with I/O devices	K2, K4, K5

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required	Practical/ Assignment/	CO Mapping

				(L+P)	Lab Nos	
1	Digital Logic Fundamental	Basic of Number System, Boolean Algebra and Logic gates, Introduction of Combinational Logic Circuits: Adders, Substractors, Multiplexers, Demultiplexers, Encoder and Decoder. Basics of Sequential Logic Circuits: Flip-Flops, Register and Counters.	Lecture, Numerical Discussion	8 L	Assignment/Practical/Quizzes	CO1
2	Computer Basics	Functional units of a Digital System and their Interconnections, Buses: Types of Buses, Bus Arbitration and its types. Register and Memory Transfer, Processor Organization: General Registers Organization, Single Accumulator Organization and Stack Organization. Instruction format and Addressing modes.	Lecture, Numerical Discussion	8 L	Assignment/Practical/Quizzes	CO2
3	Arithmetic Logic Unit and Control Unit	ALU: Carry look-ahead Adder. Multiplication: Signed operand multiplication, Booth's Algorithm and Array Multiplier, Division. Floating-point Arithmetic Operation, 1-bit ALU. IEEE Standard for Floating-Point Numbers. CU: Instruction: Instruction types, Instruction cycles and Sub-cycles, Micro-operations and Execution of a complete	Lecture, Numerical Discussion	8 L	Assignment/Practical/Quizzes	CO3

		Instruction. RISC, CISC Architecture. Hardwire and Microprogrammed Control Unit.				
4	Memory Organization	Memory hierarchy concept, RAM: SRAM and DRAM, ROM and SSD. Locality of reference property, Cache Memory: Concept with Design issues, Hit ratio, Address Mapping, Page Replacement Algorithm: FIFO, LRU, LIFO and Optimal page.	Lecture, Numerical Discussion	8 L	Assignment/Practical/Quizzes	CO4
5	Peripheral Devices and Parallel Processing	Peripheral devices, I/O ports and Interfacing, Types of interrupts. Modes of Data Transfer: Programmed I/O, Interrupt Initiated I/O and Direct Memory Access. Serial Communication: Synchronous & Asynchronous communication. Arithmetic and Instruction pipeline, Hazards and Concepts of Parallel Processing.	Lecture, Numerical Discussion	8 L	Assignment/Practical/Quizzes	CO5
Total			**40 Hours**			

Textbooks	
Sr No	**Book Details**
1	M. Mano, “Computer System Architecture”, 3rd Edition, Pearson Publication, 2007.
2	John P. Hayes, Computer Architecture and Organization, Tata McGraw Hill, Third Edition, 1998.
3	William Stallings, Computer Organization and Architecture-Designing for Performance, Pearson Education, Seventhedition,2006.

Reference Books	
Sr No	**Book Details**
1	Carl Hamacher, ZvonkoVranesic, SafwatZaky Computer Organization, McGraw-Hill, Fifth Edition, Reprint2012
2	Ray A K, Bhurchandi K M, "Advanced Microprocessors and Peripherals", TM.
3	Kai Hwang " Computer Architecture & Parallel Processing" Mcgraw Hill Education
Links	
Unit 1	tps://www.youtube.com/watch? v=L9X7XXfHYdU&list=PLxCzCOWd7aiHMonh3G6QNKq53C6oNXGrX
Unit 2	tps://www.youtube.com/watch?v=WLgXUPOjKEc
Unit 3	tps://www.youtube.com/watch?v=BPhWlFIU1rc
Unit 4	tps://www.youtube.com/watch? v=6R7JDkpG1Wk&list=PLrjkTql3jnm8HbdMwBYIMAd3UdstWChFH
Unit 5	tps://www.youtube.com/watch?v=nxryfWg5Hm4

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

Subject Name: Object Oriented Techniques using Java	**L-T-P [0-0-6]**
Subject Code: BCSE0352	**Applicable in Department: CSE/IT/AI/AIML/DS/CYS/CS/IOT**

Pre-requisites of the Subject: 1. Student must know at least the basics of computer skills, and should be able to start a command line shell.
2. Knowledge of basic programming concepts.

Course Objective- 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 Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level (KL)**
CO 1	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 Annotations concepts by using Java program.	K3
CO5	Design and develop the GUI based application, Generics and Collections in Java programming language to solve the real-world problem.	K6

Syllabus

Unit No.	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping
1	Module 1.1: Object	Introduction and Pillars of OOP with real life example, jvm architecture and its components	T1, R1, Smart	1 L + 2 P	Setting class path variables,	CO 1

Basics of Java Programming	Oriented Programming		Board/PPT/ Online Programs		Compilation of java file and execute its byte code.	
	Module 1.2: Modelling Concepts	Introduction, Class Diagram and Object Diagram, UML concepts: Association, Composition, aggregation, realization, and Generalization.		1 L + 2 P	Designing object and class diagram with UML concepts.	
	Module 1.3: Control Statements	Decision Making, Looping and Branching, Argument Passing Mechanism: Command Line Argument, Console Input.		1 L + 3 P	Implementation of java programs on control statements.	
	Module 1.4: 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.		2 L + 6 P	Implementation of Java Basics, Class, Object, abstract class interface, garbage collection.	
2 OOPs features, arrays and lambda expressions	Module 2.1: Inheritance	Overview and Types of Inheritance in Java, Access Modifiers, Constructors and super constructor in Inheritance.	T1, R1, Smart Board/PPT/ Online Programs	1 L + 3 P	Implementation of inheritance concept.	CO2
	Module 2.2: Polymorphism	Introduction and Types of Polymorphism, Overloading and Overriding		1 L + 3 P	Implementation of polymorphism concept.	
	Module 2.3: Lambda expression	Introduction and Working with Lambda Variables.		1 L + 1 P	Programs based on Lambda expression.	
	Module 2.4: Arrays	Introduction to Arrays and its Types.		1 L + 3 P	Programs based on array concept.	
3 Packages,	Module 3.1: Packages	Introduction to Packages and its Types, Access Protection in Packages, Import and Execution of Packages.	T1, R1, Smart Board/PPT/ Online	1 L + 2 P	Implementation of java package, Exception handling,	CO3
	Module 3.2: Exception	Exceptions vs. Errors, Handling of Exception. Finally, Throws and Throw keyword, Multiple Catch Block, Nested Try and Finally		2 L + 3 P		

Exception Handling and String Handling	Handling, Assertions and Localizations	Block, Tokenizer. Assertions and Localizations Concepts and its working.	Programs		Assertion, Localization and String handling	
	Module 3.3: String Handling	String Types, Operations, Immutable String, Method of String class, String Buffer and String Builder class.		2 L + 3 P		
4 Concurrency in Java and I/O Stream	Module 4.1: Threads	Overview of Threads, Creating Threads, Thread Life-Cycle, Thread Priorities, Daemon Thread, Runnable Class, Synchronizing Threads etc.	T2, R2, Smart Board/PPT/ Online Programs	2 L + 2 P	Implementation of Multi-threading, Annotation, Character and Byte Stream classes java.io package.	CO4
	Module 4.2: I/O Stream	Common I/O Stream Operations, Interaction with I/O Streams Classes.		1 L + 2 P		
	Module 4.3: Annotations	Introduction, Custom Annotations and Applying Annotations.		1 L + 2 P		
5 GUI Programming, Generics and Collections	Module 5.1: GUI Programming	Swing, AWT, Components and Containers, Layout Managers and User-Defined Layout and Event Handling.	T2, R2, R3 Smart Board/PPT/ Online Programs	2 L + 2 P	Implementation of AWT & Swing components, Layout Manager classes, Generic & Collection, and Wrapper classes	CO5
	Module 5.2: Generics	Introduction to Generic Classes, Initializing a Generic Object, Generic Cell Driver Class, Generic Methods, Use enumerated type.		1 L + 4 P		
	Module 5.3: Collections	Introduction to Collections, Using Method References, Using Wrapper Class, Using Lists, Sets, Maps and Queues, Collection using Generics, Iterators		2 L + 4 P		
Total				**(23T+47P) = 70 Hours**		

List of Practicals		
Sr. No.	**Program Title**	**CO Mapping**
1	Understanding Text Editors to Write Programs, Compile and run first java file and Byte Code and class file	CO1
2	Sketch a class and object diagram by describing the sales order system of a restaurant.	CO1
3	Sketch a class diagram by describing the circle and rectangle class.	CO1

4	Sketch a class diagram for a college platform including, classroom, playground, chair, table, smart board, teaching staff etc.	CO1
5	Sketch a class diagram containing class called Employee, which models an employee with an ID, name and salary. Add method raisesalary(percent) that increases the salary by the given percentage.	CO1
6	Program to display the default value of all Primitive data types	CO1
7	Implement the code using main() method to calculate and print the **Total** and **Average Marks** scored by a student from the input given through the command line arguments and assume that four command line arguments **name, marks1, marks2, marks3** will be passed to the main() method in the below class with name **TotalAndAvgMarks**.	CO1
8	Write code which uses if-then-else statement to check if a given account balance is greater or lesser than the minimum balance. Write a class BalanceCheck with public method checkBalance that takes one parameter balance of type double. Use if-then-else statement and print Balance is low if balance is less than 1000. Otherwise, print Sufficient balance.	CO1
9	A class NumberPalindrome with a public method isNumberPalindrome that takes one parameter number of type int. Write a code to check whether the given number is palindrome or not. For example Cmd Args : 333 333 is a palindrome	CO1
10	Write a class FibonacciSeries with a main method. The method receives one command line argument. Write a program to display fibonacci series i.e. 0 1 1 2 3 5 8 13 21	CO1
11	Write a Java Program to find the Factorial of a given number.	CO1
12	Java Program to create a class, methods and invoke them inside main method.	CO1
13	• Write a Java program to illustrate the abstract class concept. Create an abstract class Shape, which contains an empty method numberofSides(). • Define three classes named Trapezoid, Triangle and Hexagon extends the class Shape, such that each one of the classes contains only the method numberofSides(), that contains the number of sides in the given geometrical figure. • Write a class AbstractExample with the main() method, declare an object to the class Shape, create instances of each class and call numberofSides() methods of each class.	CO1
14	Java program to illustrate the static field in the class.	CO1
15	Java Program to illustrate static class.	CO1
16	Write a java program to access the class members using super keyword	CO1
17	Java program to access the class members using this keyword	CO1
18	Implement an interface named MountainParts that has a constant named TERRAIN that will store the String value “off_road”. The interface will define two methods that accept a String argument name newValue and two that will	CO1

	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.	CO1
20	Java program to demonstrate nested interface inside a class.	CO1
21	Java program to explicit implementation of garbage collection by using finalize() method	CO1
22	JAVA program to implement Single Inheritance	CO2
23	JAVA program to implement multi-level Inheritance	CO2
24	JAVA program to implement constructor and constructor overloading.	CO2
25	JAVA program implement method overloading.	CO2
26	JAVA program to implement method overriding.	CO2
27	Java program to implement lambda expression without parameter.	CO2
28	Java program to implement lambda expression with single parameter.	CO2
29	Java program to implement lambda expression with multi parameter.	CO2
30	Java program to implement lambda expression that iterate list of objects	CO2
31	Java program to define lambda expressions as method parameters	CO2
32	Write a class CountofTwoNumbers with a **public** method compareCountof that takes three parameters one is arr of type int[] and other two are arg1 and arg2 are of type int and returns true if count of arg1 is greater than arg2 in arr. The return type of compareCountof should be boolean. Assummptions: • arr is never null • arg1 and arg2 may be same	CO2
33	JAVA program to show the multiplication of two matrices using arrays.	CO2
34	Java Program to search an element using Linear Search	CO2
35	Java program to search an element using Binary Search	CO2
36	Java Program to sort element using Insertion Sort	CO2
37	Java Program to sort element using Selection Sort – Largest element Method	CO2

38	java program to Sort elements using Bubble Sort	CO2
39	Java program to create user defined package.	CO3
40	Java Program to create a sub- classing of package	CO3
41	Implement the following: • Import package.*; • import package.classname; • Using fully qualified name.	CO3
42	Implement and demonstrate package names collision in java	CO3
43	Java program to handle and Arithmetic Exception Divided by zero	CO3
44	Java Program to implement User Defined Exception in Java	CO3
45	Java program to illustrate finally block	CO3
46	Java program to illustrate Multiple catch blocks	CO3
47	Java program for creation of illustrating throw in exception handling.	CO3
48	Implement the concept of Assertion in Java Programming Language	CO3
49	Implement the concept of Localization in Java Programming Language.	CO3
50	Java program to print the output by appending all the capital letters in the input string.	CO3
51	Java program that prints the duplicate characters from the string with its count.	CO3
52	Java program to check if two strings are anagrams of each other	CO3
53	Java Program to count the total number of characters in a string	CO3
54	Java Program to count the total number of punctuation characters exists in a String	CO3
55	Java Program to count the total number of vowels and consonants in a string	CO3
56	Java Program to show .equals method and == in java	CO3
57	Given a string, return a new string made of n copies of the first 2 chars of the original string where n is the length of the string. The string may be any length. If there are fewer than 2 chars, use whatever is there. If input is "Wipped" then output should be "WiWiWiWiWi".	CO3
58	Given two strings, a and b, create a bigger string made of the first char of a, the first char of b, the second char of a, the second char of b, and so on. Any leftover chars go at the end of the result. If the inputs are "Hello" and "World",	CO3

	then the output is "HWeolrllod".	
59	JAVA program to show the usage of string builder.	CO3
60	JAVA program to show the usage of string buffer.	CO3
61	Creating and Running a Thread	CO4
62	Implementing Runnable Interface	CO4
63	Synchronizing Threads with lock	CO4
64	Synchronizing Threads without lock	CO4
65	JAVA program to implement even and odd threads by using Thread class .	CO4
66	JAVA program to implement even and odd threads by using Runnable interface.	CO4
67	JAVA program to synchronize the threads by using Synchronize statements and Synchronize block.	CO4
68	Demonstrate the concept of type annotations in the JAVA programming language.	CO4
69	Demonstrate the concept of user-defined annotations in the JAVA programming language.	CO4
70	JAVA program to implement that read a character stream from input file and print it into output file.	CO4
71	JAVA program to implement that merge the content of two files (file1.txt, file2.txt) into file3.txt.	CO4
72	Write a Java program that reads the contents of one file and copies them to another file.	CO4
73	Write a Java program that reads a text file and counts the number of words in it.	CO4
74	Write a Java program that reads a text file and counts the frequency of each word in it.	CO4
75	Write a Java program that reads a text file and adds line numbers to each line. The program should create a new file with the line numbers added to the beginning of each line.	CO4
76	Write a Java program that reads two binary files and compares them byte by byte to determine if they are identical. Display a message indicating whether the files are the same or different.	CO4
77	Program to create a frame with three button in AWT and swing	CO5
78	Program to display message with radio buttons in swing	CO5
79	Program to display "All The Best" in 5 different colors on screen. (Using AWT/Swing)	CO5
80	Program to implement event handling in a button “OK”	CO5

81	Java Program to implement BorderLayout	CO5
82	Java Program to implement GridLayout	CO5
83	Java Program to implement BoxLayout	CO5
84	Java Program to implement CardLayout	CO5
85	Java program to implement Generic class	CO5
86	Java program to illustrate Generic methods	CO5
87	Java program to implement wildcard in generics	CO5
88	Java program to implement of methods of HashSet	CO5
89	Java Program to implement methods available in HashMap class	CO5
90	Program to add, retrieve, and remove element from ArrayList	CO5
91	Create a method which can accept a collection of country names and add it to ArrayList with generic defined as String and return the List.	CO5
92	Create a method which can create a HashSet containing values 1-10. The Set should be declared with the generic type Integer. The method should return the Set.	CO5
93	Java program to implement autoboxing	CO5
94	Java program to implement unboxing	CO5
95	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.	CO5
96	Create a method that accepts the names of five countries and loads them to an array list and returns the list.	CO5
97	Create a method which can accept a collection of country names and add it to ArrayList with generic defined as String and return the List.	CO5

Textbooks

Sr. No.	Book Details
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

Sr. No.	Book Details
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
Links	
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

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

Subject Name: Artificial Intelligence and Machine Learning Lab	**L-T-P [0-0-2]**
Subject Code: BCSAI0351	**Applicable in Department: AI/AIML/CYS**
Pre-requisite of Subject: Statistics & Probability, Python	

Lab Experiments

Course Objective: The objective of this course is to implement and evaluate various AI algorithms, apply machine learning algorithms, analyze their performance, and understand the outcomes to develop the ability to address real-world challenges.

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level (KL)**
CO 1	Apply a program that solves the state space search problem using searching algorithm.	K3
CO2	Analyze the performance of linear regression, classification and clustering algorithms on various datasets.	K4
CO3	Implement ensemble learning techniques, probabilistic learning methods, and reinforcement learning algorithms to enhance model performance.	K3

List of Practicals

Sr. No	**Program Title**	**CO Mapping**

1	Implement Breadth First Search and Depth First Search algorithm.	CO1
2	Implement Best first search Algorithm on given heuristic value in a Graph and find out goal.	CO1
3	Implement A* search Algorithm on given heuristic value in a Graph and find out goal.	CO1
4	Solve Tic-toc-toe game problem using Min-Max algorithm for any given state.	CO1
5	Develop a program that solves the knapsack problem, where items of different weights and values need to be packed into a knapsack with a maximum weight capacity, maximizing the total value.	CO1
6	Implement Principal Component Analysis (PCA) algorithm.	CO2
7	Fit a linear regression model to predict housing prices based on the size of the house.	CO2
8	Implementing a class having functions for Mean Absolute Error, Root Mean Square Error, Log loss, R-square and Adjusted R Square.	CO2
9	Implement Gradient Descent algorithm and analyze the effect of learning rate and derivatives.	CO2
10	Perform multiple linear regression to predict a student's test score based on hours studied, number of assignments completed, and previous test scores.	CO2
11	Apply polynomial regression to predict stock prices based on historical data.	CO2
12	Implement K-Nearest Neighbor regression from scratch to predict housing prices based on the size of the house. Analyze the effect of value of K on error functions.	CO2
13	Understand Under fitting and Over fitting in already implemented regression algorithms, Hyper-parameter tuning.	CO2
14	Implementation of regularized linear regression: Lasso and Ridge regression.	CO2
15	Logistic regression for multiclass classification.	CO2
16	Implement K-Nearest Neighbor regression from scratch for classification.	CO2
17	Use the ID3 algorithm to build a decision tree to predict whether a customer will purchase a product based on their browsing behavior on an e-commerce website.	CO2
18	Use a support vector machine (SVM) to classify images into different categories using the CIFAR-10 dataset.	CO2
19	Comparative study of KNN, Decision Tree, SVM and Bayesian Learning on a common dataset in form of classification report.	CO2
20	Introduction to Clustering: Load a dataset and visualize it using scatter plots. Apply K-means clustering algorithm to the dataset and visualize the clusters.	CO2

21	K-means Clustering: Generate a synthetic dataset using make blobs from *sklearn* datasets. Implement K-means clustering algorithm to cluster the dataset. Visualize the resulting clustersusing scatter plots.	CO2
22	Hierarchical Clustering (AGNES - Agglomerative Nesting): Generate a synthetic dataset using make_blobs from *sklearn.datasets*. Apply the AGNES hierarchical clustering algorithm to the dataset. Visualize the resulting dendrogram.	CO2
23	Hierarchical Clustering: DIANA (Divisive Analysis): Load a dataset and pre-process it if necessary. Implement DIANA hierarchical clustering algorithm. Visualize the resulting *dendrogram.*	CO2
24	Density-based Clustering: Generate a synthetic dataset using *make_moons* or *make_circles* from *sklearn.datasets*. Apply DBSCAN algorithm to the dataset.Visualize the resulting clusters.	CO2
25	Clustering Evaluation: Load a dataset and apply a clustering algorithm of your choice. Evaluate the quality of the clustering using metrics like silhouette coefficient or adjusted Rand index.	CO2
26	Clustering on Image Data: Load an image dataset (e.g., MNIST digits) and pre-process the images. Apply a clustering algorithm (e.g., K-means) to cluster the images based on their features. Visualizethe clusters and analyze the results.	CO2
27	Implement Bayesian classifier from scratch.	CO3
28	Bayesian Learning: Implement Bayesian learning using *SKlearn* library on a public dataset. Evaluate the performance of the classifier on the testing set.	CO3
29	"Bagging and Boosting: Implement a bagging ensemble using *sklearn.ensemble.BaggingClassifier*.Implement a boosting ensemble using *sklearn.ensemble.AdaBoostClassifier*. Compare the performance of the two ensemble methods on the testing data."	CO3
30	Random Forest: Implement a random forest classifier using *sklearn.ensemble.RandomForestClassifier*. Tune the hyperparameters of the random forest using cross-validation.	CO3
31	Gradient Boosting Machines: Implement a gradient boosting classifier using *sklearn.ensemble.GradientBoostingClassifier*.	CO3
32	XGBoost: Implement an XGBoost classifier using xgboost library. Tune the hyperparameters of the XGBoost classifier using cross-validation.	CO3
33	Ensembles: Stacking: Implement a stacking ensemble using *mlxtend* library. Combine multiple base classifiers and a meta-classifier to make predictions.	CO3
34	Ensembles: Voting Classifier: Implement a voting classifier using *sklearn.ensemble.VotingClassifier*. Combine multiple base classifiers using majority voting or weighted voting.	CO3
35	Introduction to Reinforcement Learning: Implement a simple reinforcement learning agent thatlearns to navigate a grid world environment. Use concepts like state, action, reward, and policy to	CO3

	train the agent using a basic reinforcement learning algorithm.	
36	Markov Decision Process (MDP): Define a Markov Decision Process environment with states,actions, rewards, and transition probabilities. Implement the value iteration algorithm to solve the MDP and find the optimal policy.	CO3
37	Q-Learning: Implement the Q-learning algorithm to train an agent to play a simple game or solve a problem. Update the Q-values based on the agent's interactions with the environment.	CO3
38	Reinforcement Learning with OpenAI Gym: Install OpenAI Gym and select an environment to work with (e.g., CartPole, Mountain Car). Implement a Q-learning or policy gradient algorithm to train anagent to perform well in the chosen environment.	CO3

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

Subject Name: **Data Structures and Algorithms-I Lab**	**L-T-P [0-0-4]**
Subject Code: BCSE0351	**Applicable in Department: CSE/IT/CS/AI/AIML/IOT/ DS/CYS**
Pre-requisite of Subject: C, Python	

Lab Experiments

Course Objective: Learn to implement linear data structures.

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO 1	Implementing Single and Multi-dimensional array with their applications like searching and Sorting techniques.	K3
CO2	Implement Link list, Stack and Queues with their applications	K3
CO3	Implementation and analysis of various operation like searching sorting and hashing.	K4

List of Practicals

Sr. No	**Program Title**	**CO Mapping**
1	Construct a program to compare the time complexities of selection, bubble and insertion sort by plotting the graph	CO1
2	Construct a program to compare the time complexities of various algorithms by varying size "n".	CO1
3	Construct a Code to find the maximum element in an array.	CO2

4	Construct a Code to calculate the sum of all elements in an array.	CO2
5	Construct a Code to reverse the elements of an array.	CO2
6	Construct a Code to check if an array is sorted in ascending order.	CO2
7	Construct a Code to count the occurrence of a specific element in an array.	CO2
8	Construct a Code creation and traversal of 2D Array in row major and column major order.	CO2
9	Construct a code to print the transpose of a given matrix using function	CO2
10	Program to find if a given matrix is Sparse or Not and print Sparse Matrix	CO2
11	Construct a code to represent a sparse matrix in triplet form.	CO2
12	Construct a code to Implement Linear Search	CO2
13	Construct a code to implement Binary Search	CO2
14	Construct a program to Implement Selection Sort	CO2
15	Construct a program to Implement Bubble Sort	CO2
16	Construct a program to Implement Insertion Sort	CO2
17	Construct a program to Implement Shell Sort	CO2
18	Construct a program to Implement Counting Sort	CO2
19	Create a single linked list and perform basic operations (insertion, deletion, traversal).	CO3
20	Create a double linked list and perform basic operations (insertion, deletion, traversal).	CO3
21	Create a circular linked list and perform basic operations (insertion, deletion, traversal).	CO3
22	Create a circular double linked list and perform basic operations (insertion, deletion, traversal).	CO3
23	Reverse a single linked list.	CO3
24	Check if a linked list is palindrome.	CO3
25	Reverse a double linked list.	CO3
26	Find the middle element of a single linked list.	CO3
27	Find the middle element of a double linked list.	CO3

28	Merge two sorted single linked lists.	CO3
29	Detect and remove a loop in a circular linked list.	CO3
30	Construct a code to add two polynomials using linked list	CO3
31	Construct a program to Implement stack using array	CO3
32	Construct a program to Implement stack using a linked list	CO4
33	Construct a code to Infix to postfix conversion using a stack	CO4
34	Construct a code for Balanced parentheses checker using a stack	CO4
35	Implement Reverse a string using a stack.	CO4
36	Implement Binary Search using Recursion.	CO4
37	Construct a python program to print Fibonacci Series using Recursion.	CO4
38	Construct a code to implement Tower of Hanoi.	CO5
39	Construct a program to Implement queue using array.	CO5
40	Construct a code for Implementing a circular queue.	CO5
41	Construct a program to Implement queue using stack	CO5
42	Construct a program to Implement priority queue	CO5
43	Construct a program to Implement double ended queue	CO5
44	Construct a program to Implement Merge Sort with recursion	CO5
45	Construct a program to Implement Quick Sort with recursion	CO5
46	Construct a program to Implement Merge Sort using iteration	CO5
47	Construct a program to Implement Quick Sort using iteration	CO5
48	Construct a program to Implement fractional knapsack	CO5
49	Construct a program to Implement Activity selection problem	CO5
50	Construct a program to Implement Job scheduling problem	CO5
*Competitive coding list will be shared with the students.		

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

Subject Name: Artificial Intelligence and Cyber Ethics	**L-T-P [2-0-0]**
Subject Code: BNC0301	**Applicable in Department: All Branches**
Prerequisite of Subject: Basic understanding of computer systems and ethics.	
Course Objective: 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 (CO)

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

Syllabus

Unit No	Module	Topics Covered	Pedagogy	Lecture Required (T=L+P)	Aligned Practical/Assignment/Lab	CO Mapping
1	An overview to AI Ethics	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.	Lecture and Case studies	5 L	Assignment	CO1
2	Fairness and Favoritism in Machine Learning	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.	Lecture and Case studies	6 L	Assignment	CO2
3	AI Ethics and Cybersecurity Principles	Importance of privacy and security in AI, AI specific security tools and software, privacy-preserving machine learning (PPML) and privacy-preserving data mining (PPDM) Ethical considerations in phases of AI development life cycle,	Lecture and Case studies	8 L	Assignment	CO3

		Risk management: Risk assessment and incident response Regulatory compliance: GDPR, HIPAA Case studies: Implementation of AI ethics guidelines and best practices in engineering projects, Ethical decision-making processes and tools for engineers working with AI technologies				
4	Cybercrimes, IPR and Legal Measures	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	Lecture and Case studies	5 L	Assignment	CO4

5	AI Contribution to Social Evolution	Positive and negative political impacts of AI, Role of AI in social media and communication platforms, AI-generated content and deepfakes, Applications of AI in addressing global challenges, Key technical stakeholders in AI deployment: developers, researchers, policymakers, Technical Impacts on Employment and Workforce: Automation technologies: robotic process automation (RPA), autonomous systems	Lecture and Case studies	6 L	Assignment	CO5
Total				**30 Hours**		

Text Books

Sr No	Book Details
1	Introduction to Information Security and Cyber Laws, Simplified Chinese Edition by Surya Prakash Tripathi, Ritendra Goel, 1 January ,2014.
2	AI ETHICS: Paving the Path for Responsible Machine Learning, Shivanand Kumar, 2014.

Reference Books

Sr No	Book Details
1	AI ETHICS (The MIT Press Essential Knowledge series), by Mark Coeckelbergh, 2018
2	Computers, Internet and New Technology Laws by Karnika Seth – by Karnika

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

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

Subject Name: Optimization and Numerical Techniques	**L-T-P [3-1-0]**
Subject Code: BAS0404	**Applicable in Department: DS/AI/AIML/CYS**
Pre-requisites of the Subject: Knowledge of Mathematics I and II of B. Tech or equivalent.	
Course Objective: The objective of this course is to familiarize the engineers with concept of Linear Programming Problem (LPP), Integer Programming Problems, Constraint programming, various numerical techniques for mathematical task such as roots, integration, differential equations and numerical aptitude. It aims to show case the students with standard concepts and tools from B. Tech to deal with advanced level of mathematics and applications that would be essential for their disciplines.	

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO 1	Understand the concepts to formulate and to solve a Linear Programming Problem.	K1, K3
CO 2	Understand the concepts of Integer Programming Problem.	K1, K3
CO3	Understand the concepts of Non-Linear Programming Problem.	K1, K3
CO 4	Apply the concept of numerical techniques to evaluate the zeroes of the Equation, concept of interpolation and numerical methods for various mathematical operations and tasks, such as integration, the solution of linear system of equations and the solution of differential equation.	K3
CO 5	Solve the problems of Time & Work, Pipe & Cistern, Time, Speed & Distance, Boat &Stream, Analogy.	K3

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required	Practical/ Assignment/	CO Mapping

				(L+P)	Lab Nos	
1	Linear Programming	Introduction, Mathematical formulation of LP Models, Graphical Method, Description of simplex method, Big-M method, Two phase method, Alternative optimum solutions, unbounded solutions, Degeneracy, Duality in LPP.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment 1.1	CO1
2	Integer Programming	Introduction, Importance of Integer Programming Problems, Gomory's Cutting Plane method, Branch-and-Bound Method, Cargo Loading for Knapsack problem, Applications of Integer Programming.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-2.1	CO2
3	Non-linear programming	Basic facts of maxima, minima & convex optimization, Convex sets and convex functions, Continuity and differentiable properties of convex functions, Constrained Optimization- Local and Global Solution Introduction, Elements of Constraint Programming, Lagrange multiplier method, Kuhn Tucker Condition.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-3.1	CO3
4	Numerical Techniques	Error analysis, Zeroes of transcendental and polynomial equations using Bisection method, Regula-Falsi method and Newton-Raphson method, Interpolation: Lagrange's and Newton's divided difference formula. Solution of system of linear equations, Gauss Elimination method, Gauss- Seidel method. Numerical integration, Trapezoidal rule, Simpson's one third and three-eight rules.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-4.1	CO4
5	Aptitude-IV	Time & Work, Pipe & Cistern, Time, Speed & Distance, Boat & Stream, Sitting Arrangement, Analogy.	Class room Teaching, Smart Board, PPT, M-tutor.	8 L	Assignment-5.1	CO5
Total			**40 Hours**			

Textbooks	
Sr No	**Book Details**
1	Sharma J K - Operations Research (Pearson, 3rd Edition.
2	Rao S.S,"Optimization – Theory and applications", Wiley Easter Ltd., 1979.
3	Introduction to Linear Optimization by Dimitris Bertsimas & John N. Tsitsiklis, Athena Scientific 1997.
4	TahaHamdy - Operations Research - An Introduction (Prentice-Hall, 9th edition).
5	B. S. Grewal, Higher Engineering Mathematics, Khanna Publisher, 2005.
Reference Books	
Sr No	**Book Details**
1	An introduction to Optimization by Edwin P K Chong, Stainslaw Zak.
2	Hillier F S and Lieberman G J, Operations Research, Holden Day Inc., San Francisco.
3	David G.Luerbeggan, "Introduction to Linear and Non Linear Programming", Addison Wesley Publishing Co. 1973.
Links	
Unit 1	**https://nptel.ac.in/courses/112106134**
Unit 2	**https://www.youtube.com/watch?v=Lt7OZP_F3jY** **https://www.youtube.com/watch?v=BbrZsG7zesE**
Unit 3	**https://archive.nptel.ac.in/courses/111/107/111107104/**
Unit 4	**https://archive.nptel.ac.in/courses/111/107/111107105/**
Unit 5	**https://www.youtube.com/watch?v=KZ_M5RWaP6A** **https://www.youtube.com/watch?v=WP4jsNRgfa4** **https://www.youtube.com/watch?v=jPaQDKbahU8** **https://www.youtube.com/watch?v=FwiWJLicakg**

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

Subject Name: Technical Communication	**L-T-P [2-1-0]**
Subject Code: BASL0401	**Applicable in Department: All Branches**
Pre-requisite of Subject: B2 (CEFR level) in the Core Skills test; B1/B2 in the Speaking and Writing tests	
Course Objective: To develop communication and critical thinking skills necessary for succeeding in the diverse and ever-changing workplace of the twenty first century and help the students communicate effectively, creatively, accurately, and appropriately.	

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO 1	Comprehend the principles and functions of technical communication.	K2
CO2	Write for a specific audience and purpose to fulfil the provided brief.	K5
CO3	Identify and produce different kinds of technical documents.	K2, K3
CO4	Apply effective speaking skills to efficiently carry out official discourses.	K3
CO5	Demonstrate understanding of communication through digital media.	K5

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment / Lab Nos	CO Mapping
1	Introduction to Technical Communication	• Definition, Process, Types, Levels, Flow and Barriers to Technical Communication with		6 L		CO1

		emphasis on cultural differences and gender sensitivity. Gender-neutral language. • Need for and Importance of Technical Communication - Significance of audience in technical communication • Tone- Formality and Informality	Interactive & Flipped classroom method		Assignment 1	
2	Technical Writing 1	• Technical writing and technical vocabulary • Business letters/emails a) Types and format, Content Organization b) Cultural Variety, Tone, and Intention c) Bad news message, good news message d) Advertisements, Editorial press releases • Notices, agenda, and minutes of meeting • Job application, CV, and Resume'	Interactive & Flipped classroom method	10 L	Assignment 2	CO2
3	Technical Writing 2	• Technical reports – types & formats • Structure of a report (short & long) • Ethical Writing – Copy Editing, Referencing and Plagiarism • Technical Proposal - structure and types • Technical/ Scientific paper writing	PPT, Activities	7 L	Assignment 3	CO3
4	Public Speaking	• Components of effective speak • Seminar and conference presentation • Conducting/ participating in meetings • Appearing for a job interview	Interactive sessions, activities, mock interviews	8 L	Assignment 4	CO4
5	Virtual/Remote Communication	• Understanding remote work – using different online platforms • Virtual etiquette- email ids, usernames • Developing online written correspondence- blogs, WhatsApp, LinkedIn. What not to write on social media. • Participating in online Conferences/seminars/meetings • Mobile Etiquette	Interactive sessions, activities	8 L	Assignment 5	CO5

Total	39 Hours

Textbooks	
Sr No	**Book Details**
1	Technical Communication – Principles and Practices by Meenakshi Raman & Sangeeta Sharma, 4th Edition, Oxford University Press, 2023, New Delhi.
Reference Books	
Sr No	**Book Details**
1	Technical Communication: A Practical Guide by William S. Pfeiffer and Kaye A. Adkins, Pearson, 2020, UK.
2	The Essentials of Technical Communication by Elizabeth Tebeaux and Sam Dragga, Oxford University Press, 2021, UK.
3	Technical Communication Today by Richard Johnson-Sheehan, Pearson, 2020, UK
4	Strategic Communication in Technical Professions" by Susan K. Miller-Cochran and Jason Tham, Routledge, 2020, UK.
5	Technical Writing for Engineers & Scientists by Michelle V. Z. Holmes, McGraw Hill, 2020, US.
6	Speaking: Second Language Acquisition, from Theory to Practice by William Littlewood, Cambridge University Press, 2022, UK.
7	The Writing Revolution: A Guide to Advancing Thinking Through Writing in All Subjects and Grades by Judith C. Hochman and Natalie Wexler, Jossey-Bass, 2022, USA.

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

Subject Name: Data Structure and Algorithms-II	**L-T-P [3-0-0]**
Subject Code: BCSE0401	**Applicable in Department: CSE/IT/CS/AI/AIML/IOT/DS/CYS**
Pre-requisite of Subject: C, Python	
Course Objective: The objective of the course is to learn the basic concepts of algorithm analysis, along with the implementation of non-linear data structures.	

Course Outcomes (CO)

	Course outcome: After completion of this course students will be able to:	**Bloom's Knowledge Level(KL)**
CO1	Apply tree structures effectively demonstrating proficiency in tree operations and algorithms.	K3
CO2	Analyse the graph data structure and implement various operations for problem solving.	K4
CO3	Implementation and analysis of dynamic programming for efficient problem-solving across diverse contexts.	K4
CO4	Apply efficient backtracking and branch &bound techniques across diverse problem-solving scenarios.	K3
CO5	Understand advanced data structures, their implementation and application for efficient data manipulation and retrieval.	K2

Syllabus

Unit No	Module	Topics Covered	Pedagogy	Lecture Required L=T+P	Practical/Assignment/Lab	CO Mapping
1	Module 1.1: Trees	Trees: Terminology used with Trees, Binary Tree, Memory representation of Tree, Traversal Algorithms:	Lectures, Code Walkthroughs, hands-on		.	

Design and Analysis of Algorithms: Trees		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.	programming, Problem Solving, Collaborative Learning, competitive coding Projects, and Assessments.	8L+10P		CO1
	Module 1.2: Application of Trees	Priority Queue, Heap Sort, Huffman codes.				
2 Design and Analysis of Algorithms: Graphs	Module 2.1: 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,	Lectures, Code Walkthroughs, hands-on programming, Problem Solving, Collaborative Learning, competitive coding, Projects, and Assessments.	8L+10P	Depth First Search and Breadth First Search. Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prim's and Kruskal's algorithm. Directed-Acyclic Graph, Transitive Closure, and Shortest Path algorithms: Dijkstra	CO2
	Module 2.2: 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.				
3	Module 3.1: Dynamic Programming	Dynamic Programming concepts 0/1 Knapsack, Longest Common Sub	Lectures, Code Walkthrough			

Dynamic Programming		Sequence, Matrix Chain Multiplication, Resource Allocation Problem.	s, hands-on programming, Problem Solving, Collaborative Learning, competitive coding, Projects, and Assessments.	8L+8P		CO3
4 Backtracking, Branch and Bound	Module 4.1: Backtracking	Backtracking, Branch, and Bound with Examples Such as Travelling Salesman Problem, Graph Colouring, n-Queen Problem, Hamiltonian Cycles, and Sum of Subsets.	Lectures, Code Walkthroughs, hands-on programming, Problem Solving, Collaborative Learning, Projects, Assessments.	8L+10P		CO4
5 Advanced-Data Structures	Module 5.1: Advanced-Data Structures	Red-Black Trees, B – Trees, B+ Trees, Binomial Heaps, Fibonacci Heaps, Tries.	Lectures, Code Walkthroughs, hands-on programming, Problem Solving, Collaborative Learning, Projects, Assessments.	8L+10P		CO5
Total No. of Lecture + Practical Labs			**(40L+48P) = 88 Hours**			
Textbooks						
Sr. No.	**Book Details**					
1	Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, “Data Structures and Algorithms in Python (An Indian Adaptation)”, Wiley Publication, 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	
Sr. 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.
Links	
Unit 1	https://www.youtube.com/watch?v=tORLeHHtazM&pp=ygUMdHJlZXMgIG5wdGVs
Unit 2	https://www.youtube.com/watch?v=9zpSs845wf8&pp=ygUcZ3JhcGggIGRhdGEgc3RydWN0dXJlICBucHRlbA%3D%3D
Unit 3	https://www.youtube.com/watch?v=5dRGRueKU3M&pp=ygUUZHluYW1pYyBwcm9ncmFtbWluZyA%3D
Unit 4	https://www.youtube.com/watch?v=DKCbsiDBN6c&list=PL-Y5_GYVx275I87vW3LUzEJ-g7TDgn0Ts https://www.youtube.com/watch?v=3RBNPc0_Q6g&pp=ygUuYmFja3RyYWNraW5nIGFuZCBicmFuY2ggYW5kIGJvdW5kIHByb2dyYW1taW5nIA%3D%3D
Unit 5	https://www.youtube.com/watch?v=8h80p_rYv1Y&list=PLv9sD0fPjvSHqIOLTIvHJWjkdH0IdzmXT

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

Subject Name: Theory of Automata & Formal Languages	**L-T-P [3-0-0]**
Subject Code: BCSE0404	**Applicable in Department: CSE/IT/CS/AI/AIML/IOT/DS/CYS**

Pre-requisite of Subject: 1. Mathematical Foundations
2. Fundamental of Computer System

Course Objective: The Theory of Automata and formal Languages is a comprehensive study of both foundational principles and practical application in Computer Science. It delves into formal methods of computation, exploring theoretical frameworks like formal languages and the classification of machines based on language recognition capabilities.

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO1	Understand the concepts of Finite State Machines for modeling and their power to recognize the languages.	K2
CO2	Understand and identify the equivalence between the Regular Expression and Finite Automata.	K2
CO3	Define Grammar for Context Free Languages and use Pumping Lemma to disprove a Formal Language being Context-Free.	K3
CO4	Implement Pushdown Automata (PDA) for Context Free Languages and Transform the PDA to Context Free Grammar and vice-versa.	K3
CO5	Implement Turing Machine for Recursive and Recursive Enumerable Languages.	K4

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping
1 Introduction to Finite Automata:	Module 1.1: Introduction to Finite Automata	Role of Automata and Formal languages, Alphabet, String, Grammar, Language, Chomsky Hierarchy of languages. Introduction to Finite State Machine, Deterministic Finite Automaton (DFA) and Non-Deterministic Finite Automaton (NFA), Equivalence of NFA and DFA, NFA with ∈-Transition, Equivalence of NFA's with and without ∈-Transition, Minimization of Finite Automata, Limitations and Applications of Finite Automata, Concepts of Moore and Mealy Machine's, Equivalence of Moore and Mealy Machine.	Lectures, PPTs, Notes and Smart Interactive Panel	12 L	Practice Questions Based on Finite Automata, Equivalence of Finite Automata	CO1
2 Regular Language and Finite Automata	Module 2.1: Regular Language and Finite Automata	Regular Expressions, Regular Sets, Properties of Regular Expression, Identity Rules, Finite Automata and Regular Expression, Arden's theorem, Regular Grammars-Right Linear and Left Linear grammars, Conversion of FA into Regular grammar and Regular grammar into FA, Regular and Non-Regular Languages- Closure properties and Decision properties of Regular Languages, Pumping Lemma, Application of Pumping Lemma.	Lectures, PPTs, Notes and Smart Interactive Panel	9 L	Practice Questions Based on Regular Expression and Applications of Pumping Lemma	CO2
3 Context Free Language and Grammar	Module 3.1: Context Free Language and Grammar	Context Free Grammar (CFG)-Definition, Derivations, Derivation Trees and Ambiguity, Simplification of CFG, Normal Forms- Chomsky Normal Form (CNF), Greibach Normal Form (GNF), Pumping Lemma for CFL, Closure properties of CFL, Decision Properties of CFL.	Lectures, PPTs, Notes and Smart Interactive Panel	8 L	Practice Questions Based on the Context Free Grammar and Context Free Language	CO3
4 Pushdown Automata	Module 4.1: Pushdown Automata	Introduction to Pushdown Automata, Representation, Deterministic and Non-Deterministic Pushdown Automata, The	Lectures, PPTs, Notes and Smart	8 L	Practice Questions Based on Designing of	CO4

		Language of PDA: Acceptance by Final State and Acceptance by Empty Stack, Design of Pushdown Automata, Equivalence of Context Free Grammars and Pushdown Automata, Applications of Push Down Automata, Two Stack Pushdown Automata**.**	Interactive Panel		PDA ,CFG to PDA and Vice Versa	
5 Turing Machine	Module 5.1: Turing Machine	Basic Concept of Turing Machine, Model, Representation of Turing Machines, Techniques for Turing Machine Construction, Variants of Turing Machine, Universal Turing machine, Linear Bounded Automata, Church's Thesis, Recursive and Recursively Enumerable language, Closure Properties of Recursive and Recursively Enumerable Languages, Introduction to Undecidability, Halting Problem, Post's Correspondence Problem (PCP), Modified Post Correspondence Problem(MPCP), Applications of Turing Machine.	Lectures, PPTs, Notes and Smart Interactive Panel	8 L	Practice Questions Based on Construction of Turing Machine and Decidability	CO5
Total				**45 Hours**		

Textbooks	
Sr. No.	**Book Details**
1.	K.L.P. Mishra, and N. Chandrasekharan,"Theory of Computer Science-Automata, Languages and Computation",PHI , 3rd Edition, 2006.
2.	Adesh K. Pandey and Manisha Sharma,"Automata Theory and Formal Languages", S K Kataria and Sons , 1st Edition, 2019.
Reference Books	
Sr. No.	**Book Details**
1.	A. M. Padma Reddy, "Finite Automata and Formal Languages- A simple Approach", Cengage Learning India Private Limited, 2019.
2.	A.A. Puntambekar," Formal Languages and Automata Theory", Vikas Publishing House,2nd Edition, 2008

3.	J Martin, “Introduction to languages and the theory of computation”, Tata McGraw Hill ,3rd Edition, 2002.
Links	
Unit 1	https://onlinecourses.nptel.ac.in/noc24_cs71/preview
Unit 2	https://www.youtube.com/watch?v=VOaAuHAwHT4&list=PL_obO5Qb5QTEihQ35PgzjZSh7PveVt-iF
Unit 3	https://www.youtube.com/watch?v=9kuynHcM3UA&list=PLmXKhU9FNesSdCsn6YQqu9DmXRMsYdZ2T
Unit 4	https://www.youtube.com/watch?v=eqCkkC9A0Q4&list=PLEbnTDJUr_IdM___FmDFBJBz0zCsOFxfK
Unit 5	https://www.youtube.com/watch?v=XslI8h7cGDs&list=PLxCzCOWd7aiFM9Lj5G9G_76adtyb4ef7i

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

Subject Name: Operating Systems	**L-T-P [2-0-0]**
Subject Code: BCSE0403	**Applicable in Department: CSE/IT/CS/AI/AIML/DS/CYS/IOT**
Pre-requisite of Subject: Basic knowledge of computer fundamentals, C programming, Data structure and Computer organization.	
Course Objective: The objective of this course is to provide an understanding of the basic and modern concepts of operating system and deliver the skills needed to develop and customize Linux shell programming	

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO1	Understand various operating systems architecture with utilizing the command line interface (CLI) within a Linux environment.	K2
CO2	Understand and implement the various CPU scheduling algorithms.	K4
CO3	Analyse deadlock, concurrency, and synchronization into the system architecture.	K4
CO4	Identify and implement the memory management techniques and algorithms.	K3
CO5	Analyse file management system and implement distributed and virtual machine configurations on modern operating systems.	K4

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping
1 Fundamentals & Shell scripting	Module 1.1: Fundamentals of Operating Systems	Overview of Operating Systems, Operating system architecture, Types of Operating System: Batch OS, Multiprogramming OS, Multitasking OS, Multiprocessor OS, Real time OS, System call and kernel	Lectures, PPTs, Notes and Smart Interactive Panel	4L+8P	Experiment/ Program 1.1 to 1.4	CO1

	Module 1.2: Shell Scripting in Linux	Introduction to Linux Operating System, Basic Command Line Interface (CLI) Operations in Linux, Shell Scripting Basics: Variables, Control Structures, Functions				
2 Process Management	Module 2.1: Process Management	Process Performance Criteria, 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), Processes and Threads, Linux Process Management: ps, top, kill, nice	Lectures, PPTs, Notes and Smart Interactive Panel	8L+12P	Experiment/ Program 1.1 to 1.4	CO2
3 Concurrency and Deadlock Management	Module 3.1: Concurrency and Deadlock Management	**Concurrency**: Race Condition, Critical Section, Inter Process Communication, Classical problem: Producer consumer, Dinning Philosopher, Reader writer, Sleeping barber **Process Synchronization**: Lock variable, Peterson's Solution, Strict alternation, Lamport Bakery Solution, Test and set lock, and semaphore- counting, binary and monitor, **Deadlock:** Deadlock characterization, Prevention, Deadlock Avoidance: Bankers Algorithms, Deadlock detection, Recovery from Deadlock	Lectures, PPTs, Notes and Smart Interactive Panel	8L+8P	Experiment/ Program 1.1 to 1.4	CO3
4 Memory Manage	Module 4.1: Memory Management	Memory Management function, Loading and linking Address Binding, Memory management techniques, Contiguous technique- Fixed Partitions, variable	Lectures, PPTs, Notes and Smart Interactive Panel	8L+10P	Experiment/ Program 1.1 to 1.4	CO4

ment		partitions, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Non-contiguous, Paging, Segmentation, Segmented paging, Virtual Memory Concepts, Demand Paging, Performance of Demand Paging, Page Replacement Algorithms: FIFO, LRU, Optimal and LFU, Belady's Anomaly, Thrashing				
5 File Management & Modern Operating System	Module 5.1: File Management	File Management: Access Mechanism, File Allocation Method, Free Space Management: -Bit Vector, Linked List, DISK: Disk Architecture, HDD vs SDD, Disk Scheduling	Lectures, PPTs, Notes and Smart Interactive Panel	4L+10P	Experiment/ Program 1.1 to 1.4	CO5
	Module 5.2: Modern Operating System	Modern Operating System: -Overview of modern operating system, Shared Memory concepts, Distributed system, Parallel system & its architecture, Virtual machines – hypervisor, Introduction to GPUs **Case Study:** Large File Storage in a Distributed Manner				
Total				**(32T + 48P) = 80 Hours**		

Textbooks

Sr No	Book Details
1.	Abraham Silberschatz, Peter Baer Galvin and Greg Gagne" Operating System Concepts Essentials" , Willey Publication,8th Edition,2017.
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

Sr. No.	Book Details

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.

Links	
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&index=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) Operating System #32 Dealing with Deadlocks | Deadlock Avoidance & Prevention (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&index=4 https://www.youtube.com/watch?v=_BtDcroOTSA

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

Subject Name: Database Management Systems	**L-T-P [0-0-6]**
Subject Code: BCSE0452	**Applicable in Department: CSE/IT/CS/AI/AIML/ IOT/DS/CYS**
Pre-requisite of Subject: - 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 Objective: - 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 Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO 1	Understand and 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 and justify the complex queries in database with different applications.	K5
CO4	Understand and execute the concept of PL/SQL, transaction and concurrency control.	K3
CO5	Evaluate and implement Relational and Non-Relational database on different tools for real-world applications.	K5

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required (L+P)	Practical/ Assignment/ Lab Nos	CO Mapping

1 Introduction of Database & Conceptual Designing	Module 1.1: Introduction about the DBMS	Basic Concept: - Introduction of SDLC, Data, Information, Database, DBMS, History of Database, Database system Vs File system, Data models & Types of Data Models Relational Database term: - Relation, Tuple, Attribute and Domain, Codd Rules	Chalk & Duster/ Lectures, PPTs, Notes and Smart Interactive Panel	8L+8P	Experiment/ Program 1.1 to 1.8	CO1
	Module1.2: Design & Implement the ER Diagram	Data Modelling using the Entity Relationship Model: ER model concepts, Degree of relationship, Notation for ER diagram, mapping constraints reduction of an ER diagrams to tables. Extended Entity Relationship Diagram & reduction of EER				
	Module 1.3: Introduction on SQL, Implement the DDL, DML, DCL & TCL	Introduction on SQL & Types of SQL commands: -DDL, DML, DCL, TCL				
	Module 1.4: Introduction on Relational Algebra	Basic of Relation Algebra & Operations, Query Optimization				
2 Basic of SQL &	Module2.1: Implementation on the Keys	Keys & Types of Keys: - Super key, Candidate Key, Primary Key, Alternative Key Composite Primary key, Foreign Key, unique and Composite Unique key	Chalk & Duster/ Lectures, PPTs, Notes and Smart Interactive	7L+10P	Experiment/ Program 2.1 to 2.11	CO2

Normalization	Module 2.2: Implementation of Data Constraint	Data Constraint: -Null, Not Null, Default and check Constraint	Panel			
	Module 2.3: Implementation of Aggregate function & clause	Use of Aggregate Function Uses of String Functions in SQL Uses of mathematical functions in SQL Uses of Advanced Functions in SQL Use of Clause: Where, Group by, Having and Order by				
	Module 2.4: Understand & implement the normalization and different types of functions in SQL.	Functional Dependencies, Normalization & Types of Normalization, Candidate Key, Minimal Cover of FD's				
3 Introduction of Complex Queries	Module3.1: Operator & Predicates	Operator & Predicates: - Like, Between, Aliases, distinct, limit, Implementation of Logical operator: - And, Or, Not	Chalk & Duster/ Lectures, PPTs, Notes and Smart Interactive Panel	7L+10P	Experiment/ Program 3.1 to 3.9	CO3
	Module3.2: Set Theory Operator	Set Theory Operator: - Union, Intersect, Minus.				

	Module3.3: Binary Operator	Binary Operator: - 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				
	Module 3.4: Nested Query	Nested Query or Sub Query: -IN, NOT IN, Exists, Not Exists, All and Any				
	Module 3.5: Understand& Implementation the database connectivity	Database connectivity with Java/Python and other Programming Languages				
4 Introduction of PL/SQL and Transaction & Concurrency control concept	Module 4.1: Implementation index, Views and Array	Managing Indexes, Synonyms and Sequences, Managing Views, Managing Data in Different Time Zones, Array Function & Operators,	Chalk & Duster/ Lectures, PPTs, Notes and Smart Interactive Panel	6L+8P	Experiment/ Program 4.1 to 4.10	CO4
	Module 4.2: Implementation of PL/SQL	Introduction of PL/SQL Implementation of PL/SQL Function, Procedure, Trigger, Cursor				
	Module 4.3: Implementation of Transition management & concurrency	Transaction system: - Life cycle of transaction, ACID Properties Schedule & Types of Schedule, Recoverability Concurrency Control Techniques: Concurrency Control, Locking Techniques for concurrency control, 2-phase Locking protocol Transaction & Data Control: -Grant, Revoke, commit & Rollback				

	control					
5 Introduction of NoSQL With MongoDB	Module 5.1: Understand NoSQL Concept and implement the CURD operations	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 Risk	Chalk & Duster/ Lectures, PPTs, Notes and Smart Interactive Panel	8L+12P	Experiment/ Program 5.1 to 5.10	CO5
	Module 5.2: Implement the MongoDB Cursor, relation and Aggregation in MongoDB.	Introduction and Features of MongoDB, MongoDB Operators, MongoDB Collection & Document, CRUD operations, MongoDB Shell & their commands,				
	Module 5.3: Understand the concept of cloud database.	Introduction of Cloud Database. MongoDB Cloud product : Stitch, Atlas & Cloud Manager.				
Total				**(36L+48P) = 84 Hours**		

Textbooks	
Sr. No.	**Book Details**
1.	Abraham Silberschatz, Henry F. Korth, and S. Sudarshan," Database Concepts", McGraw Hill ,7th Edition, 2020.
2.	Elmasri, Navathe, "Fundamentals of Database Systems", Addision Wesley ,7th Edition, 2016.
3	Ivan Bayross, "SQL, PL/SQL – The Programming Language of Oracle", BPB Publication 5th Edition ,2023.
4.	Dan Sullivan, "NoSQL for Mere Mortals", Addison-Wesley Professional ,1st edition. 2015.

Reference Books	
Sr. No.	**Book Details**
1	Thomas Cannolly and Carolyn Begg, "Database Systems: A Practical Approach to Design, Implementation and Management", Pearson Education,3rd Edition, 2007.
2	Raghu Ramakrishan and Johannes Gehrke "Database Management Systems", McGraw-Hill, 3rdEdition, 2014.
3	NoSQL and SQL Data Modeling: Bringing Together Data, Semantics, and Software, Ted Hills, 1st Edition,2016.
4	Brad Dayley "NoSQL with MongoDB in 24 Hours", Sams Publisher, 1st Edition, 2014.

Links	
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) DBMS L14A Query Optimization (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) DBMS L17A Recovery System - YouTube
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

List of Practicals		
Lab No.	**Program Logic Building**	**CO Mapping**
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.	CO1
3	Design an ER diagram for a travel agency that includes entities such as travellers, bookings, destinations, and itineraries. also implement the relationship and cardinalities between the entities with their relevant attribute.	CO1
4	Converting Company & Travel Agency ER Model to Relational Model (Represent entities and relationships in tabular form, represent attributes as columns, identifying keys).	CO1
5	Each students create at least one ER & EER diagram from real world problem and convert in tabular from with all needed constraint.	CO1
6	Implement DDL and DML commands	CO1
7	Implement DCL & TCL commands	CO1
8	i. Create Database, Rename Database, Delete Database in relational database tool. ii. Create table employee with attributes Emp_no<datatype><size> E_name<datatype><size>	CO1

	JOB <datatype><size> Address <datatype><size> Salary<datatype><Size> iii. Insert data into the table iv. Implementation of select command v. Implementation of update command vi. Implementation of alter command vii. Implementation of delete command viii. Implementation of rename command. ix. Implementation of rollback command x. Implementation of commit Command xi. Implementation of Truncate Command xii. Implementation of Drop Command	
9	Implementation of I/O Constraint: Primary Key, composite primary key, Foreign Key with on delete set null and on delete set null constraint	CO2
10	Implementation of constraint: Unique Key and Composite unique key and uses Unique key as foreign key.	CO2
11	Implementation of Business Constraint: Null, Not Null, Default, Check.	CO2
12	Implement and apply the different form of normalization approach on company /Travel Agency Database .	CO3
13	Reduction & Implementation in SQL for ER Diagram of Company Database: - **i.** Create table for EMPLOYEE, DEPARTMET, PROJECT, DEPENDENTS and WORK_ON with all needed keys and other constraints. **ii.** Populated all table with atleast Ten records in each table as per as applied constraints.	CO2
14	Practicing Queries using Like, Between, Aliases, distinct Operator & Predicate.	CO2
15	Implementation of Aggregate Functions.	CO2
16	Implementation of Scalar, Mathematical and Advanced functions.	CO2
17	Implementation of Queries using Where, Group by, Having and Order by Clause.	CO2
18	Implementation and uses of clause and operators on Company/ Travel Agency or other database. i. Find the name of employee whose name start with A. ii. Find the name of employee where ‘hi’ in any position. iii. Find the name of employee whose ‘r’ have in second position. iv. Find the details of employee whose salary is less than 70000. v. Find the name of employee whose name start with V and end with l. vi. Find the average salary of each department vii. Find the max salary of each department	CO2

	viii. Find the sum of salary of department that have more than three employees in ascending order. ix. Find the empid of Employee who work in more than 3 project. x. Find the empid who have more than one dependent. xi. K. Implement the concept of rollback and commit on Employee Table	
19	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 Salary. v. List the employees who are either 'CLERK' or 'ANALYST' . vi. List the employees who joined on 1-MAY-81, 3-DEC-81, 17-DEC-81,19-JAN-80 . vii. List the employees who are working for the Deptno 10 or 20. viii. List the E-names those are starting with 'S' . ix. Display the name as well as the first five characters of name(s) starting with 'H' x. List all the emps except 'PRESIDENT' & 'MGR" in asc order of Salaries. xi. Display total salary spent for each job category. xii. Display lowest paid employee details under each manager. xiii. Display number of employees working in each department and their department name. xiv. Display the details of employees sorting the salary in increasing order. xv. Show the record of employee earning salary greater than 16000 in each department. xvi. Add constraints to check, while entering the empno value (i.e) empno > 100. xvii. Define the field DEPTNO as unique. xviii. Create a primary key constraint for the column (EMPNO).	
20	Implementation of Queries using set theory operators UNION, INTERSECT, MINUS.	CO3
21	Implementation of Queries using Inner Join:- Natural Join , Equi Join & Non Equi Join	CO3
22	Implementation of Queries using Outer Join :- Left Outer Join, Right Outer Join and Full Outer Join	CO3
23	Implementation of Queries nested Queries or Sub Queries: - IN, NOT IN, Exists, Not Exists, All and Any.	CO3
24	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.	CO3

	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 any project. (f') 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 xii. salary among all employees. xiii. Retrieve the names of all employees whose supervisor's supervisor has '888665555' for Ssn. xiv. For each department that has more than 5 employees retrieve the dno and no. of its employees who are making more than 6,00,000 xv. Find the sum of salaries of all employees of 'ACCOUNTS' department as well as the MAX(SAL), MIN(SAL),AVG(SAL) in this department xvi. Show the resulting salary for employee working on IOT project is given a 10% raise	
25	**Requirement:** A college consists of number of employees working in different departments. In this context, create two tables' employee and department. Employee consists of columns empno, empname, basic, hra, da, deductions, gross, net, date-of-birth. The calculation of hra, da are as per the rules of the college. Initially only empno, empname, basic have valid values. Other values are to be computed and updated later. Department containsdeptno, deptname, and description columns. Deptno is the primary key in department table and referential integrity constraint exists between employee and department tables. Perform the following operations on the database: i. Create tables department and employee with required constraints. ii. Initially only the few columns (essential) are to be added. Add the remaining columns separately by using appropriate SQL command 3. Basic column should not be null. iii. The default value for date-of-birth is 1 Jan, 1990. iv. When the employees called daily-wagers are to be added the constraint that salary should be greater than or equal to 5000 should be dropped. v. Display the information of the employees and departments with description of the fields. vi. Display the average salary of all the departments.	CO3

	vii. Display the average salary department wise. 9. Display the maximum salary of each department and also all departments put together. viii. Commit the changes whenever required and rollback if necessary. ix. Find the employees whose salary is between 5000 and 10000 but not exactly 7500. x. Find the employees whose name contains 'en'. xi. 12.Create alias for columns and use them in queries. xii. 13. List the employees according to ascending order of salary. xiii. 14. List the employees according to ascending order of salary in each department. xiv. Find the employees who are born on Feb 29. xv. Find the departments where the salary of at-least one employee is more than 20000. xvi. Find the departments where the salary of all the employees is less than 20000. xvii. Add the column dept_location in department table.	
	Understand & implement the Database Connectivity with Java/Python etc. programming language	CO3
26	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. D To retrieve the SSN of all employee who work as a supervisor and also manage the department. v. We want to retrieve a list of names of each female employee's dependents vi. 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. vii. Retrieve the names of employees who have no dependents. viii. List the names of all employees with two or more dependents. ix. List the names of managers who have at least one dependent. x. Retrieve the names of all employees who do not have supervisors. xi. Retrieve the name of each employee who has a dependent with the same first name and is the same sex as the employee.	CO3
27	Create Desktop/Web application using the database connectivity.	CO3
28	Implementation of Array Function	CO4
29	Implementation of Array Operators	CO4
30	Implementation of Indexing, Views and sequence	CO4

31	i. Write a PL/SQL Program t3o Add Two Numbers ii. Write PL/SQL Program for Fibonacci Series iii. Write PL/SQL Program to Find Greatest of Three Numbers	CO4
32	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.	CO4
33	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.	CO4
34	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:	CO4
35	Implementation of commit and rollback statement with amount transfer example.	CO4
36	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	CO4
37	i. Write a PL/SQL block to calculate the incentive of an employee whose ID is 110. ii. Grant and revoke DCL command used on Employee table -GRANT SELECT ON Employee TO emp_name; -Granting multiple privileges on Employee table -Granting all privileges on Employee table; -Granting privilege to a role in Employee table -Granting the WITH GRANT OPTION on Employee table. -Revoke all the permission applied on Employee table. iii. Create the CUSTOMERS table having the following attributes: - (ID, NAME, AGE, ADDRESS, SALARY)	CO5

	- Insert ten records in customer table. -In Customer table delete those records which have age = 25 and then COMMIT the changes in the database. -In Customer table delete those records which have age = 30 and then Rollback the changes in the database. - Create three savepoint for customer table in that the three deletions have taken place. - Apply the savepoint 2 with rollback on customer table and display the table record. - Apply the SET Transaction command.	
38	Study of Open Source NOSQL Database and installation of MongoDB	CO5
39	Create, drop, rename the database in MongoDB	CO5
40	Implementation the MongoDB Operators.	CO5
41	Implementation the CRUD Operation in MongoDB	CO5
42	Implementation of the MongoDB Shell commands	CO5
43	Implementation of MongoDB Cursor and their methods	CO5
44	Implementation of relation in MongoDB	CO5
45	Implementation of Aggregate in MongoDB	CO5
46	Deployment the data on different tools like HBASE, Riak and Cassandra	CO5
47	Implementation of all CRUD operation, Cursor and aggregate etc. on real world problem. Connect to MongoDB (by using mongo shell) i. Create database with name (ems) - use ems; ii. Create collection with following fields: {"name", age", gender","exp, subjects, "type"" qualification"}, iii. Insert the Ten documents into "faculty" collection (Use insertMany()) Write the following queries: i. Get the details of all the faculty. ii. Get the count of all faculty members. iii. Get all the faculty members whose qualification is “Ph.D”. iv. Get all the faculty members whose experience is between 8 to 12 years. v. Get all the faculty members who teach “MATHS” or “NETWORKING”. vi. Get all the faculty members who teach “MATHS” and whose age is more than 30 years and qualification must be “Ph.D”. vii. Get all the faculty members who are working part-time or who teach “JAVA”. viii. Add the following new faculty members: {"name":"Ankita ", "age":34,"gender":"F","exp":25, subjects: ["MATHS","DE"],"type":"Full Time", "qualification":"Ph.D"} ix. Update the data of all faculty members by incrementing their age and exp by one year. x. Update the faculty “Sivani” with the following data: update qualification to “Ph.D” and type to “Full Time”.	CO5

	xi. Update all faculty members who are teaching “DBMS” such that they should now also teach “Java Programming”. xii. Delete all faculty members whose age is more than 55 years. xiii. Get only the name and qualification of all faculty members. xiv. Get the name, qualification and exp of all faculty members and display the same in ascending order of exp. xv. Sort the faculty details by their age (descending order) and get the details of the first five faculty members only.	
48	Implementation of case Study on different domain 1. E-commerce Platform 2. Inventory Management 3. Railway System 4. Hospital Data Management 5. Voice-based Transport Enquiry System 6. SMS-based Remote Server Monitor system 7. Banking System 8. AI based	CO1, CO2, CO3, CO4, CO5

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

Subject Name: Data Structure and Algorithms -II Lab	**L-T-P [0-0-4]**
Subject Code: BCSE0451	**Applicable in Department: CSE/IT/CS/AI/AIML/IOT/DS/CYS**
Pre-requisite of Subject: C, Python	

Lab Experiments

Course Objective: Learn to implement non-linear data structures.

Course Outcomes (CO)

Course Outcome: After completion of this course students 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

List of Practical's

Sr. No.	**Program Title**	**CO Mapping**
1	Write a program to implement an in-order traversal of a binary tree and print the nodes.	CO1

2	Write a program to implement a pre-order traversal of a binary tree and print the nodes.	CO1
3	Write a program to implement a post-order traversal of a binary tree and print the nodes.	CO1
4	Write a program to count number of nodes in a binary tree	CO1
5	Write a program to find the height of the tree	CO1
6	Write a program to check if the Binary tree is balanced or not.	CO1
7	Write a Program to search a number in Binary Search Tree (BST)	CO1
8	Write a program to insert a node in a Binary Search Tree (BST).	CO1
9	Write a program to delete a node from a Binary Search Tree (BST).	CO1
10	Write a program to implement a max-heap and perform heap sort on an array of integers.	CO1
11	Write a Program to implement human coding algorithm	CO1
12	Write a program to implement priority queue using max heap.	CO1
13	Write a program to create a graph using an adjacency matrix.	CO2
14	Write a program to create a graph using an adjacency list.	CO2
15	Write a program to perform Depth-First Search (DFS) on a graph.	CO2
16	Write a program to perform Breadth-First Search (BFS) on a graph.	CO2
17	Write a program to check if there is a path between two nodes in a graph using DFS.	CO2
18	Write a program to find all the vertices reachable from a given vertex in a graph using BFS.	CO2
19	Write a program to detect a cycle in an undirected graph using DFS.	CO2
20	Write a program to detect a cycle in a directed graph using DFS.	CO2
21	Write a program to find the degree of each vertex in an undirected graph.	CO2
22	Write a program to count the number of connected components in an undirected graph.	CO2
23	Write a program to implement Dijkstra Algorithm.	CO2
24	Write a program to implement Prims Algorithm.	CO2
25	Write a program to implement Kruskal Algorithm.	CO2

26	Write a program to implement Floyd Warshall's all pair shortest path algorithm.	CO3
27	Write a program to implement Bellman ford Algorithm.	CO3
28	Write a program to implement Longest common subsequence (LCS).	CO3
29	Write a program to implement sum of subset problem using backtracking.	CO3
30	Write a program to implement insertion and search operations in a Tree.	CO3

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

Subject Name: Operating Systems Lab	**L-T-P [0-0-4]**
Subject Code: BCSE0453	**Applicable in Department: CSE/IT/CS/AI/AIML/DS/CYS/IOT**
Pre-requisite of Subject: Basic knowledge of computer fundamentals, C programming, Data structure and Computer organization.	

Lab Experiment

Course Objective: 1. This course gives an ability to Hands-on and practical experience with usage of the Linux OS and basics of Shell Programming.
2. The OS Lab aims to provide an experience to implement and analyze algorithms related to process management, CPU scheduling, memory management, file systems, and concurrency control and simulate modern operating systems.

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO1	Execute the Linux file system using basic shell commands.	K3
CO2	Implement CPU Scheduling Algorithms, Process Synchronization and deadlock handling techniques.	K3
CO3	Simulate memory allocation concepts, as well as distributed and virtual machine configurations, on modern operating systems.	K3

List of Practical's

Sr. No.	Program Title	CO Mapping
	Variables and Control Structures:	**CO1**
1	Write a shell script to determine the Area and Perimeter of a Rectangle.	CO1

2	Write a shell script to count the words, characters, and lines in the file.	CO1
3	Write a shell script that calculates the sum and average of an array of numbers	CO1
4	Write a shell script to calculate the Fibonacci sequence.	CO1
5	Write a shell script that finds prime numbers inside a user-specified range.	CO1
6	Write a shell script to determine whether a given string is palindrome.	CO1
	File Manipulation:	
7	Write shell script that allows users to create, delete, and list files in a directory.	CO1
8	Write a shell script that Count Lines in Each File in a Directory.	CO1
9	Write a shell script that find and Replace Text in Files.	CO1
10	Write a shell script that find Files Modified in the Last N Days.	CO1
	Directory Navigation:	
11	Write a shell script to list contents of a directory.	CO1
12	Write a shell script to change directory (cd) based on user input.	CO1
13	Write a shell script to navigate to the directory that contains a specific file.	CO1
	Process Management:	
14	Write a shell Script to display running processes and their details.	CO1
15	Write a shell Script to kill processes based on name or ID.	CO1
16	Write a shell Script to automatically Restart a Process if it Crashes	CO1
	User/Group Management:	
17	Write a shell Script to create, modify, and delete user accounts.	CO1
18	Write a shell Script to add or remove users from groups.	CO1
	Toolkit of Shell Scripts Demonstrating Shell Scripting of Functions:	
19	Write a shell script to file Backup Script with Custom Retention Policy	CO1
20	Write a shell script for database Backup and Restore Script.	CO1

21	Write a shell script for Network Configuration Script with Error Handling	
	Intercepting System Calls Using Dynamic Tracing Tools:	CO1
22	Write a shell Script to intercept system calls using strace and log process ID, system call name, arguments, and return values.	CO1
23	Write a shell Script to intercept library calls using ltrace and capture similar information.	CO1
24	Write a shell script to monitor process forks using “ps”	CO1
	Collecting and Analyzing Network Statistics:	
25	Write a shell script to collect packet counts using tools like tcpdump or tshark.	CO1
26	Write a shell script to measure bandwidth usage using iftop or nload.	CO1
27	Write a shell script to analyze latency using ping or traceroute.	CO1
28	Write a shell script to check connection status using netstat or ss.	CO1
29	Write a shell script to visualize network data using gnuplot or matplotlib for graphs and charts.	CO1
	Miscellaneous Commands:	
30	Print Current Date and Time: Write a shell script to Display the current date and time using date command.	CO1
31	Generate Random Password: Write a shell script to Use openssl rand to generate a random password.	CO1
	View System Information:	CO1
32	Write a shell script to show system information like kernel version, CPU info, etc., using uname, lscpu, etc.	CO1
33	Display System Uptime: Write a shell script to show system uptime using uptime command.	CO1
34	View Disk Usage: Write a shell script to Display disk space usage of files and directories using du and df commands.	CO1
35	Check System Load: Write a shell script to monitor system load averages using w or top commands.	CO1
36	Display Calendar: Write a shell script to show the calendar for a specific month using cal.	CO1
37	Search Text in Files: Write a shell script to Use grep to search for specific text within files.	CO1
38	Count Lines in a File: Write a shell script to Use wc -l to count the number of lines in a file.	CO1
39	Check System Users: Write a shell script to Display currently logged-in users using who or w commands.	CO1

40	Implement FCFS CPU Scheduling algorithm.	CO2
41	Implement the given CPU Scheduling algorithm a) SJF b) Priority Based	CO2
42	Implement Multi-level Queue CPU Scheduling algorithm.	CO2
43	Implement PRIORITY CPU Scheduling Algorithm (For both Pre-emptive and non-pre-emptive).	CO2
44	Implement Round-Robin CPU Scheduling Algorithm	CO2
45	Implement Multilevel Queue CPU Scheduling Algorithm.	CO2
46	Execute the RACE Condition of Process Synchronization.	CO3
47	Implement the Producer–consumer problem using semaphores.	CO3
48	Design a code and implement the Dinning Philosopher problem	CO3
49	Execute an algorithm for deadlock detection.	CO3
50	Implement Banker's algorithm of Deadlock Avoidance	CO3
51	Implement Contiguous memory fixed size partition scheme.	CO4
52	Implement Contiguous memory variable size partition scheme.	CO4
53	Simulate the First-Fit contiguous memory allocation technique.	CO4
54	Simulate the Best-Fit contiguous memory allocation technique.	CO4
55	Simulate the Worst-Fit contiguous memory allocation technique.	CO4
56	Implement the Non-contiguous	CO4
57	Memory Allocation by using Paging.	CO4
58	Write a Program to simulate the FIFO page replacement algorithm.	CO5
59	Write a Program to simulate the LRU page replacement Algorithm.	CO1
60	Write a Program to simulate the Optimal page replacement Algorithm.	CO5
61	Write a program to simulate FCFS Disk Scheduling Algorithm	CO5
62	Program to simulate the SSTF Disk Scheduling Algorithm	CO5
63	Connects to VMware vCenter and lists all virtual machines along with their power state.	CO5

64	Creates a new virtual machine with specified configurations in Azure.	CO5
65	Demonstrate how to set up and deploy a simple distributed function using Azure Functions. The function should be able to handle HTTP requests and run in a distributed manner across Azure's infrastructure.	CO5
66	Write a shell script for the mount command, which is used to attach file systems to the file system hierarchy at a mount point.	CO5
67	Write a shell script for the umount command, which is used to detach a mounted file system.	CO5
68	Write a shell script for Automate backups using cron with the tar command.	CO5
	Variables and Control Structures:	
69	Write a shell script to determine the Area and Perimeter of a Rectangle.	CO5
70	Write a shell script to count the words, characters, and lines in the file.	CO5

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

Subject Name: Technical Communication Lab	**L-T-P [0-0-2]**
Subject Code: BASL0451	**Applicable in Department: CSE/CSE (R)/IT/DS/IoT/AI/AIML/CS/BT/ECE/CYS/ME**

Prerequisite of Subject: B2 (CEFR level) in the Core Skills test; B1/B2 in the Speaking and Writing tests

Lab Experiments

Course Objective: To develop communication and critical thinking skills necessary for succeeding in the diverse and ever-changing workplace of the twenty first century and help the students communicate effectively, creatively, accurately, and appropriately.

Course Outcomes (CO)

Course outcome: After completion of this course students will be able to:		**Bloom's Knowledge Level(KL)**
CO1	Comprehend the principles and functions of technical communication.	K2
CO2	Write for a specific audience and purpose to fulfil the provided brief.	K5
CO3	Identify and produce different kinds of technical documents.	K2, K3
CO4	Apply effective speaking skills to efficiently carry out official discourses.	K3
CO5	Demonstrate understanding of communication through digital media.	K5

List of Practical's

Lab No.	Topic	Program Logic Building	CO Mapping

1	Case Study Analysis	The students will be able to develop their critical thinking and analytical skills.	CO1
2	Email Role Reversal: Writing and responding to emails in peer groups	The students will practice writing and responding to professional emails.	CO2
3	Infographics – Data Analysis and Interpretation Task	The students will develop their ability to decipher important information from charts, graphs, tables, and diagrams.	CO3
4	Document Redesign Challenge: Redesigning existing technical documents to improve readability	The students will develop their ability to write and edit professional documents.	CO3
5	Abstract Formulation and Referencing	The students will be able to write research papers with proper source citations.	CO3
6	Case Study presentations	The students will improve their analytical skills and by presenting improve their speaking skills.	CO4
7	Presentation on Project Report	The students will develop professional speaking skills.	CO4
8	Ted talk simulation – summarising a Ted Talk	The students improve their ability to condense speeches.	CO4
9 & 10	Mock Interviews	The students will practice and enhance their interview skills.	CO4
11 & 12	Webinar Presentations/Online Interviews	The students will improve their ability to make presentations in professional scenarios and perform well in online interviews.	CO5

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

Subject Name: Environmental Science	**L-T-P [2-0-0]**
Subject Code: BNC0402	**Applicable in Department: All Branches**
Pre-requisite of Subject: Environmental science is an interdisciplinary field that requires a solid foundation in various subjects to fully understand the complex interactions within the environment. Building a strong foundation in subjects like physics, chemistry, biology, maths, geography, economics will equip students with the knowledge and skills necessary to tackle complex environmental challenges and contribute to sustainable solutions.	
Course Objective: To help the students in realizing the inter-relationship between man and environment and help the students in acquiring basic knowledge about environment.	

Course Outcomes (CO)

Course outcome: After completion of this course students 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	K1,K1
CO2	Understand the different types of natural recourses like food, forest, Minerals and energy and their conservation	K1,K2
CO3	Understand the importance of biodiversity, Threats of biodiversity and different methods of biodiversity conservation.	K1,K2
CO4	Understand the different types of pollution, pollutants, their sources, effects and their control methods.	K1,K2
CO5	Understand the basic concepts of sustainable development, Environmental Impact Assessment (EIA) and different acts related to environment	K1,K2

Syllabus

Unit No	Module Name	Topic covered	Pedagogy	Lecture Required	Practical/ Assignment/	CO Mapping

				(L+P)	Lab Nos	
1	Basic Principle of Ecology	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. Basic concepts of sustainable development, SDGs, Ecosystem services, UN Decade for Eco restoration	Smart board, PPTS, Reference Books,	4 L	NA	CO1
2	Natural Resources and Associated Problems	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, 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.	Smart board, PPTS, Reference Books,	4 L	NA	CO2
3	Biodiversity Succession and Non-Renewable Energy Resources	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	Smart board, PPTS, Reference Books	4 L	NA	CO3

		biodiversity conservation in-situ and ex-situ conservation strategies Mega diversity zones and Hot spots, concepts, distribution and importance. Succession: Concepts of succession, Types of Succession. Trends in succession. Climax and stability.				
4	Pollution and Solid Waste Management	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, Climate change, global warming, acid rain, ozone layer depletion.	Smart board, PPTS, Reference Books	4 L	NA	CO4
5	Role of Community and Environmental Protection Acts	Role of community, women and NGOs in environmental protection, Bio indicators and their role, Natural hazards, Chemical accidents and disasters risk management, Environmental Impact Assessment (EIA), Salient features of following Acts: a. Environmental Protection Act, 1986, Wildlife (Protection) Act, 1972.b. Water (Prevention and control of pollution) Act, 1974.c. Air (Prevention and control of pollution) Act, 1981. Forest (Conservation) Act, 1980.d. Wetlands (Conservation and Management) Rules, 2017; e. Chemical safety and Disaster Management law.F. District Environmental Action Plan. Climate action plans.	Smart board, PPTS, Reference Books	4 L	NA	CO5
Total			**20 Hours**			

Textbooks

Sr No	Book Details
1.	Brady, N.C. 1990. The nature and properties of Soils, Tenth Edition. Mac Millan Publishing Co., New York.

2	Botkin, D.B and Kodler E.A., 2000, Environmental Studies : The earth as a living planet. John Wiley and Sons Inc. Environmental studies and Environmental engineering –By Dr. H.H
3	Environmental Studies By Dr B.S.Chauhan

Reference Books	
Sr No	**Book Details**
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
3	Environmental studies- R, Rajagopalan -Oxford Pubtiotion20051

Links	
Unit 1	Ecosystems and Biomes | Classroom Learning Video - YouTube
Unit 2	Environmental Science EVS Unit 3 Natural Resources Land Resources AEC semester 1/2 DU SOL NCWEB P -1 (youtube.com)
Unit 3	'Biodiversity & its Conservation' In Just 24 Minutes🔥🔥| Ultimate Revision Series | Neet 2022 (youtube.com)
Unit 4	Air Pollution | What Causes Air Pollution? | The Dr Binocs Show | Kids Learning Videos Peekaboo Kidz (youtube.com)
Unit 5	Environmental Pollution - Environment and Ecology for UPSC IAS Part 2 (youtube.com)