Academic Programme offered by

Department of Mathematics, IIIT Lucknow

M. Sc. Data Science

Eligibility for Admission:
Bachelor's degree (B.A. /B.Sc. /B.Stat. /B.Math./ B.Tech/B.E. or equivalent) with a background in Mathematics, Statistics or Computer Science. Those in the final year of their undergraduate program are also eligible to apply.

How to Apply:
Selection for the programme will be through Joint Admission Test for Masters (JAM).

The course is planned to start by July 2023.

M. Sc. Data Science

Course Structure 2023

Semester I	Semester II	Semester III	Semester IV
Computational Thinking through Programming (4 Credits)	Big Data Analysis (4 Credits)	Design and Analysis of Algorithms (4 Credits)	Reinforcement Learning (4 Credits)
AI & ML with Python (4 Credits)	Deep Learning (4 Credits)	Bayesian Data Analysis (4 Credits)	Elective II (4 Credits)
Mathematical Methods for Data Science (4 Credits)	Data Structures (4 Credits)	Mathematical Finance (4 Credits)	Thesis III (8 Credits)
Probability and Statistics (4 Credits)	Discrete Mathematics (4 Credits)	Elective I (4 Credits)	
Professional Communication (3 Credits)	Competitive Coding (2 Credits)	Thesis II (4 Credits)	
Competitive Coding (2 Credits)	Thesis I (2 Credits)		
Sports (1 Credit)			
(22 Credits)	(20 Credits)	(20 Credits)	(20 Credits)

List of Electives

Algorithmic Graph Theory

Natural Lanuage Processing

Data Mining and Warehousing

Data Engineering

Topological Data Analysis

Convex Optimization

Algorithmic Trading

Financial Data Analysis

Note: All electives are of 4 Credits

Fee Structure for M Sc Data Science 2023 Batch

	Semester 1	**Semester 2**	**Semester 3**	**Semester 4**
Total Fees (Without Hostel)	79000	56700	60500	56700
Total Fees (Single Occupancy*)	99000	76700	80500	66700
Total Fees (Two Seated Occupancy*)	89000	66700	70500	56700
Total Fees (Three Seated Occupancy*)	85000	62700	66500	62700
Total Fees (More than Three Occupancy*)	82000	59700	63500	59700
Mess Fees				
Total	17500	15000	15000	15000

*Hostel allotment depends upon availability.

Seat Matrix

Seat matrix							
Open	Open-PwD	EWS	EWS-PwD	SC	ST	OBC-NCL	Total
11	1	3	0	5	2	8	30

NEP 2020 Implementation for M.Sc. 2023 Batch Onwards

- Multiple Entry Multiple Exit
  According to 11.5 of NEP 2020 (pp 37), we have implemented multiple entry and exit point for our M.Sc. 2023 batch onwards in the following manner:

  - Any student can leave the course after FIRST Year of M.Sc.; such students will be granted PGD in Data Science.

  - Cooling Period: The student who wants to leave his/her study after FIRST year can leave, however to complete the M.Sc. Degree, (s)he has to come back and join the course directly in SECOND year within the BLOCK period of two years. (S)He will get at most four years to complete M.Sc. degree.

- Multidisciplinary Education : According to 11.7 of NEP 2020 (pp 37), A course in Professional Communication and Sports have been included in the course structure.

- Multi-Mode and Digital Education: According to The Point 24 of NEP 2020 (pp 58),

  - The subjects which are running in online mode, will be evaluated through proctored online examination.

Course Syllabus

FIRST SEMESTER

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Computer Science

Semester: I

Course Code: CTP1301C

Course Name: Computational Thinking through Programming

Credits	L	T	P	Section (Group)
4	3	0	1	M.Sc.

Course Module Details

Objective(s)	The objective of this course is to grow the computational thinking and problem solving ability of students. Moreover, the aim of this subject is to create various programming concepts such as inputs/outputs, variables, control statements, functions, arrays, pointers, structures, etc. For coding or writing the programs, syntaxes of C language will be taught.
Pre-Requisites	No prior programming experience is assumed. However, logical and rational maturity at the level of a first year engineering or science undergraduate is assumed.
Description	• **Introduction to Digital Computer and Programming(2 hours):** Basic components of computer, binary representation, bits and bytes, program, software. • **Introduction to Computational Thinking (2 hours):** Procedural computational approach to real life problems, idea of algorithms, creating flowcharts and pseudo-code. • **Introduction to Computational Problem Solving through C (2 hours):**Programming language concepts and its applicability on problem solving, introducing C programming language, inputs and outputs, compiling and running C program • **C Fundamentals (4 hours):** C character set, identifiers and keywords, data type, consonants, declarations, operators (arithmetic, relational, logical, assignment, unary, bitwise, etc.). • **Control Statements (4 hours):** Branching: if-else, Looping: while, do-while, for, nested control, switch, break, continue, goto. • **Functions (4 hours):** Defining a function, accessing a function, function prototypes, argument passing, recursion. • **Variables (3 hours):** variable and their scopes, automatic, external/global, static variables.

Description	• **Arrays (4 hours):** Defining an array, processing arrays, passing arrays to functions, multi-dimensional arrays. • **Structure and Unions (3 hours):** Defining and processing a structure, user defined data types, structures and pointers, passing structure to functions, self-referential structures, Unions. • **Data Files (3 hours):** File handling, multi-file programming. • **String (2 hours):** Defining and processing string. Various operations on string. • **Program analysis (1 hour):** Debugging programs with gdb, memory analysis using valgrind. • **Capstone Project (2 hours):** A capstone project using majority of the above modules.
Laboratory Experiments:	Implementation of all the above modules covered in theory through C programming.
Learning Outcomes Expected:	After completing the course, the student will be able to:. • Computationally think and analyze a real-life problem. • Write pseudo codes and corresponding program in C for a undertaken project. • Comprehend the logic and procedural flow of a program. • Acquire knowledge various syntaxes and concepts of C programming. • Undertake some advanced courses, e.g., Algorithms, Advanced Programming Languages, etc.

Contact Details: Dr. Chandranath Adak, Department of Computer Science, IIITL, chandra@iiitl.ac.in

Courseware and Reference Books

- **Text Books**
  1. Byron Gottfried, *Schaum's Outline: of Programming with C*, 4th Edition, McGraw-Hill, 2018.
  2. E. Balaguruswamy, *Programming in ANSI C*, 8th Edition, Tata McGraw-Hill, 2019.
- **References**
  1. Brian W. Kernighan and Dennis M. Ritchie, *The C Programming Language*, Second Edition, Prentice Hall of India, 1988.
  2. Herbert Schildt, *C: The Complete Reference*, 4th Edition, McGraw Hill Education, 2017.

Department of Mathematics

Semester: I

Course Code: PSC3300C

Course Name: Probability and Statistics

Credits	L	T	P	Section (Group)
4	3	1	0	M.Sc.

Course Module Details

Objective(s)	To provide a balanced introduction to probability theory and mathematical statistics along with their applications.
Pre-Requisites	Basics of Linear Algebra
Description	• **Probability(3 hours):** Combinatorial probability, Independence of events, Conditional probabilities • **Random variables (5 hours):** Random variables, densities, Expectation, Variance and moments, Standard univariate distributions, Independence of random variables, Moment Generating Functions • **Law of Large Numbers (4 hours):** Tchebychev's inequality and weak law of large numbers, Central Limit Theorem. • **Probability Distributions (6 hours):** Marginal Distribution, Conditional Distribution, Conditional expectation, Regression, Correlation, Bivariate normal distribution, Multivariate normal distribution • **Introduction to Statistics with examples of its use (6 hours):** Draw random samples, Descriptive statistics, Graphical statistics: Histogram, scatter diagram, Pie diagram, estimates sample moments, sample mean, sample standard deviation • **Sampling distributions based on normal populations (6 hours):** t, chi-square and F distributions • **Sufficient statistics (5 hours):** Point and Interval Estimation, Consistency, Minimum Variance Unbiased Estimator (statement only), method of moments estimators, maximum likelihood estimator, consistency and asymptotic normality of MLE's (statement only) • **Testing of Hypothesis (4 hours):** One sample and two sample tests based on t, chi-square and F distributions. - Error probabilities, statistical power of test, p-values, log-likelihood ratio test

Learning Outcomes Expected:	After completing the course, the student will be able to:. • Understand the basic principles of probability, different probability distributions of discrete, continuous, joint random variables and their characteristics • estimation of population parameters from data • learn the basic components of hypothesis testing and perform various hypothesis tests.

Contact Details: Dr. Mary Samuel, Department of Mathematics, IIITL, marysamuel@iiitl.ac.in

Courseware and Reference Books

- **Text Books**

1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying E. Ye, *Probability and Statistics for Engineers and Scientists* ,by Pearson, Ninth Edition, 2013
2. Sheldon Ross, *A First Course in Probability* , Pearson, Ninth Edition, 2018
3. Prabhanjan N. Tattar, Suresh Ramaiah, B. G. Manjunath, *A Course in Statistics with R*, Wiley, 2018

- **References**

1. William W. Hines, Douglas C. Montgomery , David M. Goldsman, Connie M. Borror, *Probability and Statistics in Engineering* ,fourth edition, John Wiley & Sons, 2003.
2. Sheldon M. Ross, *Introduction to Probability and Statistics for Engineers and Scientists* , fifth edition, Academic Press, 2014.
3. Irwin Miller Marylees Miller *John E. Freund's Mathematical Statistics with Applications*, Eighth Edition, Pearson Education Limited, 2014.
4. Charles M.Grinstead, J. Laurie Snell, *Introduction to Probability*,second revised edition, American Mathematical Society 1997.
5. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, *Probability and Statistics for Engineers and Scientists*, Ninth edition, Pearson 2017.
6. V.K. Rohatgi & A.K. Md. E. Saleh, *An Introduction to Probability and Statistics*, second edition, Wiley, 2001.
7. Alexander Mood, Franklin Graybill, Duane Boes, *Introduction to the Theory of Statistics*,third edition, McGraw-Hill,1974.

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Management and Humanities

Semester: I

Course Code: PCO2300C

Course Name: Professional Communication

Credits	L	T	P	Section (Group)
3	3	0	0	M.Sc.

Course Module Details

Objective(s)	The objective of the course is to build a toolkit of communication skills that will enable students to become an effective communicator. It aims at advancing the soft-skills in students to increase their employability prospects. It prepares them for dealing with stressful situations in their professional career.
Pre-Requisites	Basic Proficiency in English Language
Description	• **Basics of Communication (2 hours):** Overview of Communication, Types of Communication, 7 Cs of Communication, Barriers to Communication, Need for Professional Communication, Role of Professional Communication in Industries, Job Opportunities in Professional Communication • **Conversation (6 hours):** Creating a Communication Strategy, Introducing Yourself, Networking, Conversation and Dialogues: Starting a Conversation, Ending a Conversation, Telephonic Conversation, How to Handle Difficult Conversations, What to Say and What Not to Say in Crisis Situation • **Non-Verbal communication (8 hours):** Body Language: Facial Expressions, Posture, Eye Contact, Kinesics, Proxemics, Chronemics, Haptics, Cross-Cultural Communication, Voice Features: Tone, Voice Modulation, Fluency, Rate of Speech, Pitch • **Effective Speaking (22 hours):** How to Cope with Public Speaking Anxiety, Presentation Skills: Planning, Composition, Review, Oral Presentation, Online Presentation, Interview, Group Communication-Introducing Others, Giving Feedback, Delivering Bad News, Group Discussions. • **Communicating with People in Stress (4 hours):** Awareness about Psychological Impact of Stress, What People Experience in a High-Stress Environment, Positive Communication: Practicing Empathy, A Good Listener, Reassurance, Follow Up.

Learning Outcomes Expected:	After completing the course, the student will be able to:. • effectively use soft skills in professional settings • employ communication strategies in situations of crisis • plan and make effective oral presentations with/ without visual aid • communicate effectively in high-stress environment

Contact Details: Dr. Neelu, Department of Management & Humanities, neelu@iiitl.ac.in

Courseware and Reference Books

- **Text Books**
  1. Raman, M., & Sharma, S. , *Technical communication: Principles and practice*, Oxford University Press, 2015.
- **References**
  1. Anderson, P. V, *Communicative English for engineers and professionals*, Pearson Education India, 2010.
  2. Mishra, S., & Muralikrishna, C. , *Communication Skills for Engineers*, Pearson Education India, 2011
  3. Nitin, B. , *Communicative English for engineers and professionals*, Pearson Education India, 2010
  4. Farrell, A., & Geist-Martin, P. , *Communicating social health: Perceptions of wellness at work*, Management Communication Quarterly, 18(4), 543-592, 2005

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Mathematics

Semester: I

Course Code:

Course Name: Mathematical Methods for Data Science

Credits	L	T	P	Section (Group)
4	3	1	0	M.Sc.

Course Module Details

Objective(s)	To emphasize abstract vector spaces and linear maps.
Pre-Requisites	No prerequisites other than the usual demand for suitable mathematical maturity.
Description	• **Fundamentals (4 hours):** Norms, The Singular Value Decomposition, More on the SVD • **QR Factorization and Least Squares (5 hours):** Projectors, QR Factorization, Gram-Schmidt Orthogonalization, Householder Triangularization, Least Squares Problems • **Conditioning and Stability (5 hours):** Conditioning and Condition Numbers, Floating Point Arithmetic, Stability, More on Stability • **Systems of Equations (2 hours):** Cholesky Factorization • **Eigenvalues (6 hours):** Eigenvalue Problems, Overview of Eigenvalue Algorithms, Reduction to Hessenberg or Tridiagonal Form, Rayleigh Quotient, Inverse Iteration, Computing the SVD • **Iterative Methods(5 hours):** Overview of Iterative Methods,The Arnoldi Iteration, How Arnoldi Locates Eigenvalues • **Multi-variable analysis (6 hours):** Differentiation, convexity, gradient and Hessian of a multivariate function, Taylor's expansion, necessary and sufficient conditions for the existence of an extremal point, Newton's method, Lagrange multipliers, gradient and conjugate gradient methods. • **Sequences and series of functions(6 hours):** Pointwise and Uniform Convergence, Interchange of Limits, The Exponential and Logarithmic Functions, The Trigonometric Functions, Absolute Convergence, Tests for Absolute Convergence, Tests for Nonabsolute Convergence, Series of Functions

Learning Outcomes Expected:	• Attain a deep understanding of the definitions, theorems, and proofs. • Able to develop the ability to understand and manipulate the objects of linear algebra. • Describe the basic differences between the rational and the real numbers. • Understand and perform simple proofs • Answer question concerning uniform convergence of concrete numerical sequences and series

Contact Details:

- Dr. Mary Samuel, Department of Mathematics, IIITL, marysamuel@iiitl.ac.in

Courseware and Reference Books

- **Text Books**

  1. Gilbert Strang, *Introduction to Linear Algebra*, fifth edition Wellesley-Cambridge Press, 2016.
  2. Gilbert Strang, *Linear Algebra and Learning from Data*, Wellesley-Cambridge Press, 2019.
  3. Lloyd N. Trefethen and David Bau, *III: Numerical linear algebra*, SIAM ,1997
  4. Donald R. Sherbert, Robert G. Bartle *Introduction to Real Analysis*, Fourth edition, 2010

- **Reference Books**

  1. Sheldon Axler, *Linear Algebra Done Right*, Third Edition, John Willey and Sons, 2011.
  2. Gilbert Strang, *Linear Algebra And Its Applications*, Fourth Edition, 2014
  3. Boris Demidovich, *Problems in Mathematical Analysis*, Mir Publication, 1970

SECOND SEMESTER

Semester: II

Course Code: DLE6301C

Course Name: Deep Learning

Credits	L	T	P	Section (Group)
4	3	0	1	M.Sc.

Course Module Details

Objective(s)	The objective of this course is to grow the knowledge on recent trends of advanced machine learning techniques..
Pre-Requisites	Machine Learning, Some basic knowledge of Linear Algebra and Calculus.
Description	• **Deep Neural Network (11 hours):** Introduction to Deep Learning: A brief overview of supervised, unsupervised, reinforcement learning, Difference between classification, regression, Traditional classifiers, Multilayer Perceptron: Feed-Forward Neural Network with Backpropagation, Different activation functions their advantages and disadvantages: Sigmoid (vanishing gradient problem), ReLU (exploding gradient problem), Leaky ReLU, tanh, etc., Various loss and cost functions: MSE, log-loss, cross-entropy, hinge loss, etc., Bias vs Variance trade-off, Regularization: L2 regularization, early stopping, data augmentation, Ensembling, Dropout, etc., Optimization: Gradient Descent (GD), Batch GD, Stochastic GD, Minibatch GD, GD with momentum, Adagrad, RMSprop, Adam, etc. • **Convolutional Neural Network (7 hours):** Introduction to Convolution Neural Network (CNN), Different operations of CNN (convolution, pooling), Different concepts of CNN (Kernel, Filter, Padding, Stride), Different CNN architecture (LeNet, AlexNet, VGG Net, GoogLeNet, SqueezeNet, Xception net, Residual block and ResNet, Dense Net, etc.), Transfer Learning, Similarity learning, Siamese Net, Triplet Net

	• **Advanced Topics on Deep Learning (7 hours):** Autoencoder: Denoising autoencoder, Sparse autoencoder, Variational autoencoders, etc., Generative Adversarial Network (GAN) and some of its variants, e.g., DCGAN, CycleGAN • **Applications of Deep Learning (8 hours):** Application of Deep Learning (DL) in Computer Vision: Object Segmentation: U-Net, V-Net, Object Detection: RCNN, YOLO, etc., Application of DL in Natural Language Processing (NLP): e.g., Sentiment Analysis from reviews
Learning Outcomes Expected:	After completing the course, the student will be able to • Tackle real-life computation problems that can be addressed through deep learning, • Understand various concepts of deep learning, • Solve various computer vision and natural language processing problems, • Demonstrate an understanding of some ethical issues related to artificial intelligence.

Contact Details: Dr. Chandranath Adak, Department of Computer Science, IIITL, chandra@iiitl.ac.in

Courseware and Reference Books

- **References**

1. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016. Online: `https://www.deeplearningbook.org`.
2. A. Zhang, Z. C. Lipton, M. Li, A. J. Smola, *Dive into Deep Learning*, arXiv:2106.11342, 2021. Online: `https://d2l.ai`.

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Computer Science

Semester: II

Course Code: DST2301C

Course Name: Data Structures

Credits	L	T	P	Section (Group)
4	3	0	1	M.Sc.

Course Module Details

<table>
<tr><td>Objective(s)</td><td>

To make students develop knowledge of basic data structures for storage and retrieval of ordered or unordered data.
Choose appropriate searching and sorting techniques and apply graph algorithms for various practical problems.
Formulate new/improved solutions for programming problems using learned data structure.

</td></tr>
<tr><td>Pre-Requisites</td><td>Fundamentals of Computer Programming.</td></tr>
<tr><td>Description</td><td>

Module 1 (14 hours):

Introduction to abstract data types, variables, storage types.
Introduction to Array, Array representation, Contiguous storage.
Linear list (Abstract data type, sequential and linked representations).
Linked list (Single Linked list, Doubly Linked list, Circular Linked list).
Stack (Parenthesis matching, towers of Hanoi)
Queue (Queue, Priority queue).

Module 2 (14 hours) :

Introduction to sorting and searching methods.
Sorting (Bubble sort, Insertion sort, Selection sort, Radix sort, Merge sort, Quick sort, Heap sort).
Searching (Linear search, Binary search, search efficiency, insertion and deletion operations, importance of balancing, AVL trees, Infix, Prefix, Postfix)

</td></tr>
</table>

	• **Module 3 (14 hours):** – Introduction to non-linear data structures. Tree (Binary trees and their properties, terminology, sequential and linked implementations, tree traversal methods and algorithms, heaps as priority queues, heap implementation, insertion and deletion operations) – Graph (Definition, terminology, directed and undirected graphs, properties, connectivity in graphs, applications, implementation adjacency matrix and linked adjacency chains, graph traversal – breadth first and depth first, spanning trees). – Hashing (Search efficiency in lists and skip lists, hashing as a search structure, hash table, collision avoidance, linear open addressing, chains).
Learning Outcomes Expected:	After completing the course, the student will be able to: • Understand the strength and weakness of different data structures. • Use the appropriate data structure in context of solution of given problem. • Develop programming skills which require to solve given problem.

Contact Details: Dr. Rahul Kumar Verma, Department of Computer Science, IIITL, rahul@iiitl.ac.in

Courseware and Reference Books

- **Text Books**
  1. "Fundamentals of Data Structures in C" by Horowitz, Sahni and Anderson-Freed, 2nd Edition (2008).
  2. "Data Structures Through C in Depth" by S. K. Srivastava and Deepali Srivastava (2011).
  3. "Data Structures, Algorithms, and Applications in C++" by S. Sahani, 2nd Edition (2004).
  4. "Data Structures and Algorithms in Java" by Robert Lafore, 2nd Edition (2003).
- **References**
  1. "Data Structures and Algorithm Analysis in JAVA" by Mark Allen Weiss, 3rd Edition, (2011).
  2. "Data Structures and Algorithms" by V. Aho, J. E. Hopcroft, and J. D. Ullman, 1st edition, (1983).

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Mathematics

Semester: II

Course Code: MCS4300C

Course Name: Discrete Mathematics

Credits	L	T	P	Section (Group)
4	4	0	0	M.Sc.

Course Module Details

<table>
<tr><td>Objective(s)</td><td>Discrete Mathematics is the study of discrete/distinct structures/objects in nature.This course provides the mathematical basis for the understanding of computers and modern computation. This course serves more than one purpose. After successful completion of this course, students should learn a particular set of mathematical facts and how to apply them; more importantly, this course should teach students how to think logically and mathematically. This course stresses mathematical reasoning and the different ways problems are solved. It is the backbone of computer science and has a lot of applications in cryptography and engineering.</td></tr>
<tr><td>Pre-Requisites</td><td>Nil</td></tr>
<tr><td>Description</td><td>
Logic, Proofs, and Counting (4 hours): Propositional Logic, Direct Proof, Proof by Contradiction, Proof by Contrapositive, Constructive Proofs, Counterexamples, and Vacuous Proofs, Counting
Basic Structures (8 hours): Set Theory – Cartesian Product & Binary Relation Partition, Function, Countable & Uncountable Sets
Introduction to Abstract Algebra (13 hours): Group Theory, Rings and Fields, Vector Spaces, Finite Fields
Introduction to Number Theory (9 hours): Divisibility and Modular Arithmetic, Integer Representations and Algorithms, Primes and Greatest Common Divisors, Solving Congruences
Introduction to Graph Theory (8 hours): Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamiltonian Graphs, Shortest-Path Problems, Planar Graphs, Graph Coloring
</td></tr>
<tr><td>Learning Outcomes Expected:</td><td>After completing the course, the student will be able to:
use logical notation to define and reason mathematically about the fundamental data types and structures (such as numbers, sets) applied in computer algorithms and systems
identify and apply properties of combinatorial structures and properties
</td></tr>
</table>

Learning Outcomes Expected:	- visualize the different abstract structures (like Group, Rings and Fields) - apply the concept of abstract algebra and number theory for the development of various cryptographic primitives - understand the various types of graph Algorithms and graph theory properties along with model real world problems using graph theory

Contact Details: Dr. Dhananjoy Dey, Department of Mathematics, IIITL, ddey@iiitl.ac.in

Courseware and Reference Books

- **Text Books**

  1. Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, Eighth Edition, McGraw-Hill Education, 2019.
  2. John B. Fraleigh, *A First Course in Abstract Algebra*, Seventh Edition, Pearson Education India, 2013.

- **References**

  1. Owen D. Byer, Deirdre L. Smeltzer, & Kenneth L. Wantz, *Journey into Discrete Mathematics*, AMS/MAA Textbooks, Volume 41, 2018.
  2. David M. Burton, *Elementary Number Theory*, Seventh Edition, The McGraw-Hill Companies, 2011.
  3. F. Harary, *Graph Theory*, Narosa Publishing House, 2001.
  4. I. N. Herstein, *Topics in Algebra*, Second Edition, John Wiley & Sons, 1975.
  5. Harry Lewis, & Rachel Zax, *Essential Discrete Mathematics for Computer Science*, Princeton University Press, 2019.
  6. Gerard O'Regan, *Guide to Discrete Mathematics: An Accessible Introduction to the History, Theory, Logic and Applications*, Springer 2016.

THIRD SEMESTER

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Information Technology

Semester: III

Course Code: DAA3301C

Course Name: Design and Analysis of Algorithms

Credits	L	T	P	Section (Group)
4	3	0	1	M.Sc.

Course Module Details

Objective(s)	The designing of algorithm is an important component of computer science and information technology. The objective of this course is to make students aware of various techniques used to evaluate the efficiency of a particular algorithm. Students eventually should learn to design efficient algorithm for a particular program. Analytical skills will be tested and improved.
Pre-Requisites	Data structures.
Description	• **Introduction to Algorithm analysis (8 hours) :** Algorithm Design paradigms, motivation, concept of algorithmic efficiency, run time analysis of algorithms, Asymptotic Notations, Insertion sort example, order of growth. Solving Recurrences- substitution method, recursion tree method, master method, Iteration method. Types of solutions, Introduction to Randomized algorithms • **Divide and conquer approach (6 hours) :** structure of algorithms, building recurrence relations, runtime analysis. Examples- Mergesort, quicksort, Binary search, Strassen's Matrix multiplication • **Greedy Algorithms (8 hours) :**design technique, greedy choice property, optimal substructure, approximate algorithms. Examples- Coin selection problem, Activity selection problem, Knapsack problem, Travelling salesman problem, minimum cost spanning tree, Single source shortest paths • **Dynamic Programming (6 hours):** design technique. Examples- Shortest path in graph, chain matrix multiplication, Traveling salesman Problem, longest Common sequence problem, knapsack problem • **Graphs and Trees (3 hours) :** Overview, Representation, Types, Problem formulation and conversion, Traversal methods and their analysis • **Back tracking (3 hours) :** Overview, DFS, 8-queen problem and Knapsack problem • **Branch and bound (3 hours) :** Overview, BFS, LC and FIFO. Examples- 0/1 Knapsack problem, Traveling Salesman Problem

	• **Computational Complexity (3 hours) :** Complexity measures- P, NP, NP-H, NP-C complexity classes. Examples- SAT, TSP etc.
Lab Exercises:	The lab programs will be solve by using C/C++ Programming Language. For all the practicals, students will have to present the complexity analysis in best, worst and average cases • Implementation of Linear search method • Implementation of Recursive binary search method • Implementation of Recursive Quicksort method • Implementation of Coin selection problem using Greedy approach • Implementation of Fractional Knapsack problem using Greedy approach • Implementation of 0/1 Knapsack problem using Dynamic approach • Implementation of LCS using Dynamic approach • Implementation of n-Queen's problem using Backtracking approach • Develop a Hamiltonian Path in an undirected graph is a path that visits each vertex exactly once. A Hamiltonian cycle (or Hamiltonian circuit) is a Hamiltonian Path such that there is an edge (in graph) from the last vertex to the first vertex of the Hamiltonian Path. Develop a program to implement the solution of Travelling Salesman Problem by considering the Hamiltonian cycle approach. • A road network can be considered as a graph with positive weights. The nodes represent road junctions and each edge of the graph is associated with a road segment between two junctions. The weight of an edge may correspond to the length of the associated road segment, the time needed to traverse the segment or the cost of traversing the segment. Using directed edges it is also possible to model one-way streets. Such graphs are special in the sense that some edges are more important than others for long distance travel (e.g. highways). This property has been formalized using the notion of highway dimension. There are a great number of algorithms that exploit this property and are t herefore able to compute the shortest path a lot quicker than would be possible on general graphs. Develop a program to find the shortest path from each node to solve the road network problem.
Learning Outcomes Expected:	After completing the course, the student will be able to: • Apply knowledge of mathematics, science, engineering and computing appropriate to the discipline. • Analyze a problem, and identify and define the computing requirements appropriate to its solution. • Design, implement, and evaluate a computer-based system, process, component, or programmer to meet desired needs • Use current techniques, skills, and tools necessary for computing practice

Contact Details:

1. Dr. Deepshikha Agarwal, Department of Information Technology, IIITL, deepshikha@iiitl.ac.in
2. Dr. Vishal Krishna Singh, Department of Computer Science, IIITL, vks@iiitl.ac.in

Courseware and Reference Books

- **Text Books**

  1. T. H. Cormen, Leiserson, Rivest and Stein, *Introduction of Computer algorithm* , PHI Publication
  2. E. Horowitz, S. Sahni, and S. Rajsekaran, *Funadmentals of Computer Algorithms*, Universities Press

- **References**

  1. Sara Basse, A. V. Gelder, *Computer Algorithms*, Addison Willey Publication
  2. J.E Hopcroft, J.D Ullman, *Design and analysis of algorithms*, TMH Publication
  3. D. E. Knuth, *The art of Computer Program*, PHI Publication

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Mathematics

Semester: III

Course Code:

Course Name: Bayesian Data Analysis

Credits	L	T	P	Section (Group)
4	3	1	0	M.Sc.

Course Module Details

Objective(s)	To learn advanced hierarchical models that are used in realistic data analysis
Pre-Requisites	Basic calculus
Description	• **Basics (7 hours):** Introduction, Probability, Bayes' theorem and marginalization, Bayes, Laplace and orthodox statistics • **Parameter estimation I (8 hours):** Different priors, Reliabilities: best estimates, error-bars and confidence intervals, The coin example, Asymmetric posterior,Multimodal posterior, Gaussian noise and averages, The central limit theorem • **Parameter estimation II (8 hours):** Marginal distributions, Binning the data, Reliabilities: best estimates, correlations and error-bars, Generalization of the quadratic approximation, Brute force and ignorance, The joys of linearity, Iterative linearization, Approximations: maximum likelihood and least-squares, Error-propagation: changing variables, A useful short cut, Taking the square root of a number • **Model selection(8 hours):** Introduction, Comparison with parameter estimation, Hypothesis testing, An algorithm, Simulated data, Real data, Other examples: means, variance, dating, The analysis of means and variance, Luminescence dating • **Assigning probabilities(8 hours):** Ignorance: indifference and transformation groups, The binomial distribution, Location and scale parameters, Testable information: the principle of maximum entropy, The monkey argument, The Lebesgue measure, Averages and exponentials, Averages and exponentials, MaxEnt and the binomial distribution, Counting and Poisson statisticsCounting and Poisson statistics, Approximations: interconnections and simplifications
Learning Outcomes Expected:	After completing the course, the student will be able to: • Learn the basic theory behind Bayesian statistical inference and its applications to common problems in environmental and biological sciences

Learning Outcomes Expected:	• Understand the Bayes theorem and the related concepts, including prior, posterior and predictive distribution and likelihood function. • Familiar with graphical model representation and basics in model assessment and criticism. • Able to apply Bayes theorem to write down simple hierarchical Bayesian models for common data analysis problems such as basic parametric models and (generalized) linear regression. • Familiar with the basic concept of Markov chain Monte Carlo (MCMC) and are able to apply MCMC methods to solve hierarchical Bayesian models using computer softwares.

Contact Details:

- Dr. Mary Samuel, Department of Mathematics, IIITL, marysamuel@iiitl.ac.in

Courseware and Reference Books

- **Text Books**

1. D. S. Sivia, *Data Analysis : A Bayesian Tutorial*,Second Edition, Oxford Science Publications, 2006
2. Andrew Gelman, John B. Carlin, *Bayesian Data Analysis*,Third edition, 2021

- **References**

1. John Kruschke, *Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan*, Academic Press, 2014
2. Carl Edward Rasmussen and Christopher K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006
3. Sourish Das, Sasanka Roy, Rajiv Sambasivan,*Fast Gaussian Process Regression for Big Data, Big Data Research*, Volume 14, December 2018, Pages 12-26:
4. Sourish Das, Aritra Halder, Dipak K Dey, *Regularizing Portfolio Risk Analysis: A Bayesian Approach, Methodology and Computing in Applied Probability*, September 2017, Volume 19, Issue 3, pp 865-889
5. Gelman. A, Carlin J. B, Stern H. S., Dunson D. B, Vehtari A. and Rubin, D. B., *Bayesian Data Analysis*, Chapman & Hall/CRC. Second or third edition, 2013

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Mathematics

Semester: III

Course Code:

Course Name: Mathematical Finance

Credits	L	T	P	Section (Group)
4	3	1	0	M.Sc.

Course Module Details

Objective(s)	To study the foundations of financial theory, including asset pricing and financial institutions
Pre-Requisites	Multivariable calculus, probability, and linear algebra.
Description	• **Introduction to Mathematical Finance (19 hours):** – Securities markets, the time value of money, Markowitz portfolio theory, capital market theory, and portfolio risk measures , binomial security pricing, Itô's formula and geometric Brownian motion, forwards, futures, and options, and call option pricing with applications • **Introduction to Financial Derivatives (20 hours) :** – Modeling underliers in discrete time, stochastic calculus and modeling underliers in continuous time, general aspects of forwards, futures, swaps, and options, including trading strategies, the Black-Scholes-Merton (BSM) model, BSM p.d.e. approach to pricing European-style options, risk-neutral approach to pricing European-style options, applications to warrants, delta hedging, managing portfolio risk, and extension of the BSM model to the Merton jump-diffusion model
Learning Outcomes Expected:	After completing the course, the student will be able to: • Understand theory and applications of the mathematical models forming key pillars of modern finance • Keep a good balance between mathematical derivation and description

Contact Details:

- Dr. Mary Samuel, Department of Mathematics, IIITL, marysamuel@iiitl.ac.in

Courseware and Reference Books

- **Text Books**

  1. Arlie O. Petters, Xiaoying Dong, *An Introductionto Mathematical Finance with Applications*, Springer Undergraduate Texts in Mathematics and Technology
  2. Sheldon M. Ross, *An Elementary Introduction to Mathematical Finance*, Third Edition, 2011

- **References**

  1. M. Schweizer and E. W. Farkas, *Lecture Notes Mathematical Foundations for Finance*,ETH Zurich, 2020

FOURTH SEMESTER

ELECTIVES

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Mathematics

Semester: III/IV

Course Code: AGT6310E

Course Name: Algorithmic Graph Theory

Credits	L	T	P	Section (Group)
4	3	1	0	M.Sc.

Course Module Details

Objective(s)	To present graph theory as an useful analytical tool for computer scientists.
Pre-Requisites	Some exposure to a high level, procedural and preferably recursive programming language, to be familiar with elementary set notation and to be at ease with theorem proving.
Description	• **Introducing graphs and algorithmic complexity(5 hours):** Introducing graphs,Introducing algorithmic complexity,Introducing data structures and depth-first searching. • **Spanning-trees, branchings and connectivity(6 hours):** Spanning-trees and branchings, Circuits, cut-sets and connectivity. • **Planar graphs(6 hours):** Basic properties of planar graphs, Genus, crossing-number and thickness, Characterisations of planarity,A planarity testing algorithm. • **Networks and flows(6 hours):** Networks and flows, Maximising the flow in a network,Menger's theorems and connectivity, A minimum-cost flow algorithm. • **Matchings(5 hours):** Definitions, Maximum-cardinality matchings, Maximum-weight matchings. • **Eulerian and Hamiltonian tours(5 hours):** Eulerian paths and circuits, Hamiltonian tours. • **Colouring graphs(6 hours):** Dominating sets, independence and cliques, Colouring graphs, Face-colourings of embedded graphs. • **Introduction to NP-completeness(1 hour)**
Learning Outcomes Expected:	After completing the course, the student will be able to:. • To understand the basic language of graph theory and of algorithmic complexity. • To get an idea of how spanning-trees play an important role in connection with the circuit space and with the separability of a graph.

	• To determine what graphs can be arranged on a plane surface. • To be able to solve a variety of problems is to model them in terms of some flow along the edges of a digraph. • To understand matchings and to search for certain matchings can be an important subtask for some larger problems. • To characterise the graphs that contain either Eulerian or Hamiltonian tours. • To partition or colour the vertices, edges or faces of a graph in a way dependent upon their various adjacencies. • To differentiate between those algorithms whose execution times are bounded by a polynomial in the problem size and those which are not.

Contact Details: Dr. Mary Samuel, Department of Mathematics, IIITL, marysamuel@iiitl.ac.in

Courseware and Reference Books

- **Text Books**

  1. Alan Gibbons, *Algorithmic Graph Theory*, Cambridge University Press, 1985.
  2. Gary Chartrand, Ortrud R. Ollermann, *Applied and Algorithmic Graph Theory*, McGraw- Hill, London ISBN 0-07-557101-3, hardbound,395 pp. 1993.

- **References**

  1. Cormen, Leiserson and Rivest, *Introduction to Algorithms*, McGraw-Hill, 1986.
  2. James McHugh, *Algorithmic Graph Theory*, Prentice-Hall, 1989.
  3. M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs,* , Volume 57 in the series Annals of Discrete Mathematics. North Holland, second edition, 2004.

Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Mathematics

Semester: III/IV

Course Code: CO640E

Course Name: Convex Optimization

Credits	L	T	P	Section (Group)
4	3	1	0	M.Sc.

Course Module Details

Objective(s)	Convex optimization generalizes least-squares, linear and quadratic programming, and semidefinite programming,and forms the basis of many methods for non-convex optimization. This course focuses on recognizing and solving convex optimization problems that arise in applications, and introduces a few algorithms for convex optimization.
Pre-Requisites	Background knowledge in linear algebra and least-squares problems.
Description	• **Introduction (6 hours):** Mathematical Optimization, Least-Squares and Linear Programming, Convex Optimization, Non-Linear Optimization • **Convex Sets and Convex Functions(8 hours):** Affine and Convex sets, Generalized Inequalities, Operations that preserves Convexity, Seperating and Supporting Hyperplanes, Properties of Convex functions • **Convex Optimization Problems(10 hours):** Optimization Problems, Convex Optimization, Linear and Quadratic Optimization Problems, Exercises. • **Duality (8 hours):** The Lagrange dual function and dual problem, Geometric interpretation, Optimality Conditions, Exercises. • **Applications and Algorithms (10 hours):** Norm approximation, Least-Norm Problems, Descent Methods, Gradient Descent Method, Steepest Descent Method, Exercises.
Learning Outcomes Expected:	After completing the course, the student will be able to: ‘ • Define appropriate optimization problem for a given practical problem and solving standard convex optimization problems arising in various scientific and engineering applications. • Programming projects based on mathematical modeling followed by an application of the Convex Optimation is expected.

Contact Details: Mr. Jomin K J, Department of Mathematics, IIITL, jomin@iiitl.ac.in

Courseware and Reference Books

- **Text Books**
  - Convex Optimization, Stephen Boyd and Lieven Vandenberghe, Cambridge University Press, 2004.
- **References**

1. Convex Analysis and Optimization, Bertsekas D.P., Nedic A., Ozdaglar A.E., Athena Scientific, 2003.
2. Lectures on Convex Optimization, Yurri Nesterov, Springer, 2018

Department of Computer Science

Semester: III/IV

Course Code: NLP7301E

Course Name: Natural Language Processing

Credits	L	T	P	Section (Group)
4	3	1	0	M.Sc.

Course Module Details

Objective(s)

The general approach in the course will be covering (i) a language phenomenon, (ii) the corresponding language processing task, and (iii) techniques based on deep learning, classical machine learning and knowledge base(s). On one hand we will understand the language processing task in detail using linguistics, cognitive science, utility *etc.*, on the other hand we will delve deep into techniques for solving the problem. In addition to the graded labs, non-graded labs, and course project (graded), this course entails a *mid-semester* and an *end-semester* examination.

Pre-Requisites: Machine Learning, Linear Algebra, Calculus, and basics of Python programming.

Description

- **Week 01** (3 hours) - Introduction to Natural Language Processing (NLP)
  Course Introduction & Motivation, Multilingualism, Morphology in Languages, and
  Part-of-Speech (Pos) Tagging [Introduction].
- **Week 02** (3 hours) - PoS Tagging Layer of NLP
  Mathematics of PoS tagging, Sequences in NLP, and
  NLP Lab 1 (Non-graded) - Simple Matrix Operations, NumPy, scikit-learn
- **Week 03** (3 hours) - Hidden Markov Models (HMM) in NLP
  PoS Tagging (HMM), Viterbi Decoding for Tagging and Sequences, and
  NLP Lab 2 (Non-graded) - Most Frequent POS Tagging assignment.
- **Week 04** (3 hours) - Handling Sequential Tasks
  Shallow parsing, Named Entity Recognition (NER), Introduction to Conditional Random Field (CRF), and Challenges due to Morphological Richness.
- **Week 05** (3 hours) - Feature Engineering
  CRF (contd.), Maximum Entropy Markov Model (MEMM),
  Feature Extraction and Engineering, and
  NLP Lab 3 (Non-graded) - NER Task for multiple languages.
- **Week 06** (3 hours) - Knowledge Bases and Ambiguity
  Ambiguity and NLP, Knowledge Bases (WordNet, FrameNet, VerbNet *etc.*),
  Word Sense Disambiguation (WSD), and
  NLP Lab 4 (Graded) - Sense Disambiguation Task

- **Week 07** (3 hours) - Applications of Neural Networks (NN) in NLP
  Cognate Detection and its applications,
  NER using NNs, Text Classification using NNs
  Transformer Architecture, and Introduction to Distributional Semantics.
- **Week 08** (3 hours) - Distributional Semantics
  word2vec, doc2vec, sent2vec, sub-words in NLP, and FastText
  NLP Lab 5 (Non-graded) - word2vec, GloVe and FastText (pre-trained models),
  Embeddings Space Visualization.
- **Week 09** (3 hours) - Language Models (LMs)
  Introduction to State-of-the-Art LMs, BERTology, BERT-based fine-tuning for various NLP tasks, and
  NLP Lab 6 (Graded Lab) - NER Task with LMs.
- **Week 10** (3 hours) - Machine Translation (MT)
  Introduction to Machine Translation (MT),
  Statistical MT (SMT), Neural MT (NMT),
  NLP Lab 7 (NG) - SMT, Moses, Alignment Task.
- **Week 11** (3 hours) - Sentiment Analysis (SA)
  Introduction to Sentiment Analysis (SA), Aspect Based SA, Sarcasm Detection,
  Thwarting, and Introduction to Course Project.
- **Week 12** (3 hours) - Information Extraction (IE)
  Question Answering, Summarization, Essay Grading, and
  NLP Lab 8 (Graded) - Aspect-based SA.
- **Week 13** (3 hours) - Cognitive NLP
  Cognitive Behaviour, Introduction to Eye-tracking (ET) / EEG,
  Ethics and Bias in NLP, Features from ET, NLP Tasks with ET.
- **Week 14** (1 hour) - Course Project
  One hour for discussion on project progress,
  Other two hours for evaluation of the project.

Learning Outcomes Expected:

At the end of this course, all the attending students are expected to be able to:

- Demonstrate an understanding for various NLP sub-problems,
- Design solutions for real-world NLP challenges,
- Solve NLP problems/challenges with the help of Machine Learning- and Deep Learning-based approaches,
- Demonstrate an understanding of ethical issues, and bias in Artificial Intelligence (AI)-based problems.

Contact Details:

`https://dipteshkanojia.github.io` Dr Diptesh Kanojia, Department of CS, IIITL, diptesh@iiitl.ac.in,
`https://murthyrudra.github.io/` Dr Rudra Murthy V, Department of CS, IIITL, rudramurthy@iiitl.ac.in

Courseware and Reference Books

Note: This is not an exhaustive list of reading proposed by the course instructors. We shall add to this list as the course progresses.

- Allen, James, *Natural Language Understanding*, Second Edition, Benjamin/Cumming, 1995. Charniack, Eugene, Statistical Language Learning, MIT Press, 1993
- Jurafsky, Dan and Martin, James, *Speech and Language Processing*, Speech and Language Processing (3rd ed. draft), Draft chapters in progress, October 16, 2019.

- Manning, Christopher and Heinrich, Schutze, *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.
- Jacob Eisenstein, *Introduction to Natural Language Processing*, MIT Press, 2019.
- Deep Learning, Ian Goodfellow, Yoshua Bengio, and Aaron Courville, MIT Press, 2016.
- Radford, Andrew et. al., Linguistics, an Introduction, Cambridge University Press, 1999.
- Pushpak Bhattacharyya, Machine Translation, CRC Press, 2017.
- **Journals**: Computational Linguistics, Natural Language Engineering, Machine Learning, Machine Translation, Artificial Intelligence
- **Conferences**: Annual Meeting of the Association of Computational Linguistics (ACL), Computational Linguistics (COLING), European ACL (EACL), Empirical Methods in NLP (EMNLP), Annual Meeting of the Special Interest Group in Information Retrieval (SIGIR), Human Language Technology (HLT).
- Sag, Ivan A., Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. "Multiword expressions: A pain in the neck for NLP." In International conference on intelligent text processing and computational linguistics, pp. 1-15. Springer, Berlin, Heidelberg, 2002.
- Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. Comput. Linguist., 19(2):313–330, June.
- Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. CoRR, abs/1603.01354. (comment: POS and NER)
- Anders Søgaard and Yoav Goldberg. 2016. Deep multi-task learning with low level tasks supervised at lower layers. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 231–235, Berlin, Germany, August. Association for Computational Linguistics.
- Chapter 2 of Machine Translation by Professor Pushpak Bhattacharyya.
- Smoothing from Manning and Schutz, "Foundations of Statistical Natural Language Processing", Page 199 (general) and Page 354 (POS specific).
- Sha, Fei, and Fernando Pereira. "Shallow parsing with conditional random fields." In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, pp. 213-220. 2003.
- Ratnaparkhi, Adwait. "A maximum entropy model for part-of-speech tagging." In Conference on empirical methods in natural language processing. 1996.
- Harshada Gune, Mugdha Bapat, Mitesh Khapra and Pushpak Bhattacharyya, Verbs are where all the Action Lies: Experiences of Shallow Parsing of a Morphologically Rich Language, Computational Linguistics Conference (COLING 2010), Beijing, China, August 2010.
- Erik F. Tjong Kim Sang and Sabine Buchholz, Introduction to the CoNLL-2000 Shared Task: Chunking. In: Proceedings of CoNLL-2000, Lisbon, Portugal, 2000.
- John Lafferty, Andrew McCallum, and Fernando C.N. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data", ICML 2001.
- Toutanova, Kristina; Manning, Christopher D. (2000). "Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger". Proc. J. SIGDAT Conf. on Empirical Methods in NLP and Very Large Corpora (EMNLP/VLC-2000). pp. 63–70.