NIST UNIVERSITY

Master of Computer Application (MCA)

Syllabus

(2024-2026 Batch)

NIST UNIVERSITY

Institute Park, Pallur Hills, Berhampur,
Odisha, India – 761008

4.2.6 Master in Computer Application (MCA)

Second - Year Course Structure (III Semester)							
S. No.	Course Category	Course Code	Course Title	L	P	T	Credits
1	CC	MCA-600	Design and Analysis of Algorithms	3	0	4	5
2	CC	MCA-601	Data Warehousing and Data Mining	3	0	0	3
3	CC	MCA-602	Python Programming	3	0	4	5
4	CC	MCA-603	Artificial Intelligence	3	0	0	3
5	DSE	MCA-630*	Department Specific Elective - I	3	0	0	3
6	PR	MCA-680	Minor Project with Seminar	0	0	8	4
7	PR	MCA-681	Summer Internship	0	0	2	1
Total Credits							24

*Course Code for PE& OE (XXX-XXX) will be one of the following.

Department Specific Elective - I

MCA-630A: Computer Graphics

MCA-630B: Natural Language Processing

MCA-630C: Wireless Sensor Network

MCA-630D: Soft Computing Techniques

MCA-630E: CBOT (Computer-based optimization techniques)

Second - Year Course Structure (IV Semester)							
S. No.	Course Category	Course Code	Course Title	L	P	T	Credits
1	CC	MCA-604	Machine Learning	3	0	4	5
2	CC	MCA-605	Internet &Web Technology	3	0	4	5
3	DSE	MCA-631*	Department Specific Elective - II	3	0	0	3
4	PR	MCA-682	Major Project	0	0	24	12
Total Credits							25

Department Specific Elective – II

MCA-631A: Internet of Things (IoT)

MCA-631B: Computer Network Security

MCA-631C: Web application Development

MCA-631D: Cloud Computing

MCA-631E: Data Science

MCA-631F: Software Testing and Quality Assurance

MCA-631G: Block-chain Technology

NIST UNIVERSITY

THIRD SEMESTER DETAIL SYLLABUS

MCA-600	Design and Analysis of Algorithms	5 Credits

Course Objective:

1. Translating a plain text problems to convert into an algorithm .Calculate best case, worst case time complexity and space complexities of different algorithm and choosing the best solution from the available options
2. Applying different design paradigm to solve different problems and comparing their best case, worst case scenarios.
3. Designing and applying different data structures over different algorithms for solving different problems.
4. Understand different P-class, NP class problems.

Module-I: (12 Hrs)
Introduction to problems and algorithms , Mathematics for algorithm analysis , Insertion sort Analysing algorithms, Designing of algorithms, Asymptotic notation Standard notations and common functions, Recurrence relations, The substitution method, The recursion-tree method, The master method, Divide and conquer: Min-Max Heap, Heapsort , Quicksort, Merge Sort, Sorting in Linear Time: Lower bounds for sorting: Counting sort

Module-II: (8 Hrs)
Greedy method: Elements of the greedy strategy, Huffman codes Fractional Knapsack problem, Dynamic programming: Assembly-line Scheduling, Matrix-Chain Multiplication, Longest Common Subsequence(LCS), 0/1 Knapsack problem

Module-III: (12 Hrs)
Graph algorithms: Basic Definitions and Application, Representations of graphs, Breadth-first search and Depth-first search, Data Structures for Disjoint Sets, Strongly connected components, Minimum Spanning Trees: The algorithms of Kruskal and Prim. Single-Source Shortest Paths: The Bellman-Ford algorithm, Dijkstra's algorithm, All-Pairs Shortest Paths-Shortest paths algorithm

Module-IV: (8 Hrs)
String Matching: The Naive string-matching algorithm, The Rabin-Karp algorithm, The Knuth-Morris-Pratt algorithm. Network Flow: Flow networks, The Ford-Fulkerson method, Backtracking – n-Queen problem – Hamiltonian Circuit Problem – Subset Sum Problem. NP- Completeness: Classes P and NP, NP-complete problems

Course Outcome:

1. Given a English language problem description define the problem precisely with input/output

requirements, examine its inherent complexity and develop a generic or set of initial solutions and justify their correctness.

2. Given an algorithm descriptions, analyse the time and space complexity of the algorithm in the worst case, average case, and amortized scenario as needed in terms of asymptotic order of complexity.

3. Given a problem definition explore different alternative algorithmic solutions, compare them with respect to time and space complexity and choose the design scheme and /or design parameter and data structure appropriately to obtain the best possible choice(s) that can be converted to an executable programs.

4. Examine and prove whether a problem is of polynomial complexity, hard(np complete) or otherwise and develop optimal and approximate algorithm for them as applicable.

Text Books:

1. Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, MIT Press/McGraw-Hill, 2009.

2. Ellis Horowitz, SartajSahni and SanguthevarRajasekaran, Computer Algorithms/ C++, Second Edition, Universities Press, 2007.

Reference Books:

1. Sanjoy Dasgupta, Christos H. Papadimitriou and Umesh V. Vazirani, Algorithms, McGraw-Hill, 2008.

2. Jon Kleinberg and ÉvaTardos, Algorithm Design, Addison-Wesley/PEARSON EDUCATION-2006.

3. S. Sridhar, ―Design and Analysis of Algorithms‖, Oxford university press, First Edition, 2015.

Course Name	DESIGN AND ANALYSIS OF ALGORITHMS
Course Link	https://nptel.ac.in/courses/106101060/
Course Instructor	Prof. Abhiram G Ranade, Prof. Ajit A Diwan, Prof. Sundar Viswanathan, IIT Bombay

Course Name	DESIGN AND ANALYSIS OF ALGORITHMS
Course Link	https://nptel.ac.in/courses/106106131/
Course Instructor	Prof. Madhavan Mukund, Chennai Mathematical Institute

Course Name	DESIGN AND ANALYSIS OF ALGORITHMS
Course Link	https://online.stanford.edu/courses/cs161-design-and-analysis-algorithms
Course Instructor	Reyna Hulett, CS161, Stanford School of Engineering,

MCA-601	Data Warehousing and Data Mining	3 Credits

Course Objectives:

1. Identify the scope and necessity of Data Mining and Knowledge Discovery.

2. To understand various tools of Data Mining and their techniques to solve the real time problems.

3. To develop ability to design various algorithms based on data mining tools.

4. To develop further interest in research and design of new Data Mining techniques.

Module – I (10 Hrs)

Motivation for Data Mining, Introduction to Data Mining, DBMS vs. Data Mining, Issues and Challenges in Data Mining, Application Areas, Knowledge Discovery steps, Concept of Data Warehousing, 3-Tier Architecture, Multidimensional Data Model, OLAP, ROLAP, and MOLAP Operations,

Module – II (10 Hrs)

Data Preprocessing: Why Preprocess the data, Data Preprocessing – Descriptive data summarization, Data cleaning, Data Integration and Transformation, Data Reduction, Concept Hierarchies, Interestingness Measures, Mining Association Rules, Apriori Algorithm for finding Frequent Item-Sets,, Mining Multilevel Association Rules, Mining Distance-Based Association mining, Correlation Analysis

Module – III (10 Hrs)

Classification and Prediction: Decision Tree based Classification, Bayesian Classification, Classification by Back Propagation, K-Nearest Neighbor Classifier

Cluster Analysis: Categorization of Clustering Methods, Partitioning Methods, K-Means and K-Medoids, Hierarchical Methods, Density-Based Clustering (DBSCAN)

Module – IV (10 Hrs)

Web Mining, Classification of Web Documents, Web Content Mining, Web Structure Mining, Web Usage Mining, Text Mining, Text Clustering, Mining Spatial Databases, Mining Multimedia Databases, Temporal Data Mining, Temporal Association Rules, Sequence Mining.

Course Outcomes:

1. Understand warehousing architectures and tools for systematically organizing large

Database and use their data to make strategic decisions. Understand KDD process for finding interesting pattern from warehouse.

2. Remove redundancy and incomplete data from the dataset using data pre-processing methods.

3. Characterize the kinds of patterns that can be discovered by association rule mining.

4. Discover interesting patterns from large amounts of data to analyze for predictions and

classification

Text Books:

1. Data mining; Concepts and techniques by J. Han and M. Kamber (Morgan Kaufmann)

2. Data Mining by A.K. Pujari (University press)

3. Data Mining by Vikram Pudi and P. Radha Krishna (Oxford University Press)

4. Introduction to Data Mining - Tan, Steinbach & Kumar (Pearson)

5. Data Mining: Practical Machine Learning Tools and Techniques - Ian H. Witten & Eibe Frank (Elsevier India)

6. Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management - Gordon S. Linoff & Michael J. A. Berry (Wiley)

7. Data Mining and Analysis Fundamental Concepts and Algorithms - Zaki & Meira (Cambridge University Press)

8. Jiawei Han, Micheline Kamber, and Jian Pei, “Data Mining Concepts and Techniques”, Third Edition, Elsevier.

9. Data Warehousing, Data Mining & OLAP by Alex & Stephen, McGraw Hill.

10. Vikram Pudi & P. Radha Krishna, Data Mining, Oxford University Press.

11. 10. Reema Thareja, Data Warehousing, Oxford University Press.

Course Name	Data and Web Mining
Course Link	https://nptel.ac.in/courses/106/105/106105174/

Course Instructor	PROF. PABITRA MITRA Department of Computer Science and Engineering, IIT Kharagpur

Course Name	Data and Web Mining
Course Link	https://nptel.ac.in/noc/courses/noc20/SEM1/noc20-cs12/
Course Instructor	

MCA-602	Python Programming	5 Credits

Course Objective:

1. To understand programming skills in core Python.
2. To learn Abstract data type in Python i.e. array, string, list, set, dictionary.
3. To understand Object Oriented Programming concepts using Python
4. To learn Exception Handling, and Database Connectivity in Python.

Module-I (10 Hrs)

Python: Features of Python, Installing Python for windows and setting up paths, writing and Executing of a Python programs, Python Virtual machine, Frozen binaries, Comparison between C, Java and Python, Comments, Docstrings, How Python sees variables, Data types in Python, built in types, sequences in Python, sets, literals in Python, user defined data types, identifiers & reserved words, Naming convention in Python. Various Operators in Python, Input & Output, Control statements, if statements, while loop, for loop, infinite loop, nested loop, else suit, break, continue, pass ,assert, return statements, command line arguments.

Module-II (10 Hrs)

Arrays and Strings in Python, Creating strings, Length of a string, Indexing in strings, Slicing strings, Concatenation and Comparing strings, Finding Sub-Strings, Replacing a String.

Iterator objects in Python: list, tuple, string, or dictionary and their usages.

Functions in Python, define a function, calling a function, return from function, pass by object Reference, Positional arguments, Default arguments, Recursive functions.

Module-III (10 Hrs)

Introduction to OOP, features of OOP, creating classes, the self-variable, constructor, types of variables, namespaces, types of methods.

Inheritance: Define inheritance, types of inheritance, constructors in inheritance, overriding super class constructors & methods, the super() method, MRO

Polymorphism: Duck typing philosophy of Python, operator overloading, method overriding, interfaces in Python.

Module-IV (8 Hrs)

Exceptions: Errors in a Python program, Exceptions, Exception handling, Types of Exceptions, Exception block, the assert statement, user defined exceptions

Python Database Connectivity: DBMS, types of databases used with Python, installation of MySQL database, Working with MySQL connector, database, retrieving rows ,deleting rows, updating rows in a table.

Course Outcomes:

1: Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.

2: Express proficiency in the handling of strings and functions. Understand and determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.

3: Identify the commonly used operations involving file systems and regular expressions.

4: Understand and implement inheritance, exceptions and database handling

Text Books

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011.

2. Core Python Programming, Dr. R. Nageswar Rao , Dreamtech Press

3. Python Programming for Absolute Beginners, Michael Dawson, CENGAGE Learning

Reference Books

1. Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computer scientist : learning with Python , Freely available online.2012

Course Name	Python Programming
Course Material Link	Python Tutorial/Documentation www.python.or 2015
Course Material Link	http://docs.python.org/3/tutorial/index.html
Course Material Link	http://interactivepython.org/courselib/static/pythonds

MCA-603	Artificial Intelligence	3 Credits

Course Objective:

1. To learn the concepts of Agent based search techniques.
2. To understand adversal searching methods and knowledge representation.
3. To learn the types of uncertainty, planning, and Bayesian networks.
4. To understand the concepts of Expert Systems and it design procedure.

Module-I: (12 Hrs)

Intelligence and AI, Agents, Model of different types of agents: reactive, deliberative, goal-driven, utility-driven, and learning agents, Environment, Properties of Environment, State Space, Knowledge, Rationality, and Turing Test. Search Techniques definition and importance, uninformed search–DFS, BFS, iterative deepening, iterative broadening, depth limited search, Issues In Design Of heuristics, Best First search, A* and AO* search, Hill climbing, Simulated Annealing, ConstraintSatisfactionProblem,8-puzzle problem, Crypt-arithmetic Problem.

Module-II: (10 Hrs)

Adversarial Searching Techniques: Game Playing, minimax search, alpha-beta pruning. Knowledge Representation in AI, Logic-propositional, predicate, FirstOrder Logic. Normal forms. Modus Ponens & Modus Tollens, Theorem Proving, Principle of Resolutions, Non-Monotonic Reasoning. Semantic Net Frame.

Module-III: (10 Hrs)

Planning and its Importance. Classical Partial order planning, Conditional Planning. Uncertainty, type of uncertainty, Probabilistic Reasoning-joint distribution reasoning, Bayesian networks, learning, Explanation based learning, induction learning-Decision Tree, statistical learning-Bayesian Learning, expectation maximization, hidden Markov model, closed world problems.

Module-IV: (6 Hrs)

Expert Systems–Design Techniques, components, Problem And Knowledge Domain, Knowledge Engineering Approach, error in design of expert system, life cycle of expert system, MYCIN and Dendral–an expert system.

Course Outcome:

1. Ability to comprehend AI&ES to analyze and mapreal world activities to digital world
2. Ability to identify problems that are amenably solved by AI methods

3. Ability to design and carryout an empirical evaluation of different AI algorithms

Textbooks:

1. Artificial Intelligence–Knight&Rich, McGrawHill, 3rdEdition.
2. Principles of Artificial Intelligence –N.J.Nilson, 2ndEdition, Narosa Publishing.

Reference Books:

1. ArtificialIntelligenceAModernApproach–Russel&Norvig,2ndEdition,Pearson.
2. IntroductiontoArtificialIntelligenceandExpertSys–D.W.Patterson,PrenticeHall.
3. Expert System: Principle and programming-Joseph Giarratano, GaryRiley
4. NPTEL course-https://nptel.ac.in/courses/106106126/

Digital Learning Resources

Course Name	Artificial Intelligence
Course Link	https://nptel.ac.in/courses/106/105/106105079/
Course Instructor	Prof.P.Dasgupta, IIT Kharagpur

MCA-630A	Department Specific Elective – I (Computer Graphics)	3 Credits

Course Objective:

1. To understand the basic principles of the geometric objects in the 2D and 3D coordinates.
2. To learn various transformation techniques, and Window and View Coordinate systems.
3. To study the rendering and shading technique used to produce realistic illustrations, and understand the curves tracing algorithms.
4. To understand the different projection method, back-face detection, and shading techniques.

Module-I: (10 Hrs.)

Overview of Graphics System: Video Display Units, Raster-Scan and Random Scan Systems, Graphics Input and Output Devices. Output Primitives: Line drawing Algorithms: DDA and Bresenham's Line Algorithm, Circle drawing Algorithms: Midpoint Circle Algorithm and Bresenham's Circle drawing Algorithm.

Module-II: (10 Hrs.)

Two-Dimensional Geometric Transformation: Basic Transformation (Translation, Rotation, Scaling) Matrix Representation, Composite transformations, Reflection, Shear, Transformation between coordinate systems. Two-Dimensional Viewing: Window-to- View Port Coordinate Transformation. 3D Geometric and Modelling Transformations: Translation, Rotation, Scaling, Reflections, shear, Composite Transformation.

Module-III: (8 Hrs.)

Clipping: Line Clipping (Cohen-Sutherland Algorithm) and Polygon Clipping (Sutherland-Hodgeman Algorithm), Aliasing and Antialiasing, Half Toning, Thresholding, Dithering. Polygon Filling: Seed Fill Algorithm, Scan line Algorithm. Two-Dimensional Object Representations: Spline Representation, Bezier Curves, B-Spline Curves.

Module-IV: (10 Hrs.)

Projections: Parallel Projection, Perspective Projection. Visible Surface Detection Methods: Back-Face Detection, Depth Buffer, Z- Buffer, Scan- Line Algorithm, Painters Algorithm. Illumination Models: Basic Models, Displaying Light Intensities. Surface Rendering Methods: Polygon Rendering Methods: Gouraud Shading, Phong Shading.

Course Outcome:

1. Student will understand the basic principles for representation of the geometric objects in the 2D and 3D coordinates.

2. Student will learn the implement the algorithms and techniques necessary to produce geometric objects in 2D and 3D space illustrations.
3. Student will understand Shading algorithms to determine how a surface should be shaded to produce realistic illustrations. Curves and surfaces methods for rendering and shading curved objects.
4. The students will be able to design and implement a substantial computer graphics system to represent some complex illustrations with the help of graphics system.

Suggested Books:

1. Computer Graphics, C version; D. Hearn and M. P. Baker; Pearson Education, 2nd Edition, 2002

2. Computer Graphics Principle and Practice, J.D. Foley, A. Dam, S.K. Feiner, Addison Wesley, 4th Edition, 2014.

3. Procedural Elements of Computer Graphics, David Rogers, TMH. 1998

Course Name	Programming and Data Structures
Course Link	https://nptel.ac.in/courses/106105085/4
Course Instructor	Dr. P. P. Chakrabarti Department of Computer Science and Engineering Indian Institute of Technology Kharagpur

MOOC courses:

https://www.coursera.org/learn/interactive-computer-graphics by Takeo Igarashi (Professor) Department of Computer Science, Graduate School of Information Science and Technology, University of Tokyo.

MCA-630B	Department Specific Elective – I (Soft Computing)	3 Credits

Course Objective:

1. Learn about soft computing techniques and their applications, gain knowledge of different neural network architectures.

2. Gain knowledge of Back-Propagation network, Radial basis function, and Competitive learning.

3. Gain knowledge of Fuzzy logic, Fuzzy inferencing, Defuzzification methods.

4. Analyze the genetic algorithms and their applications.

Module -I:

Introduction: Introduction to soft computing, application areas of soft computing, classification of soft computing techniques, structure & functioning of biological brain & Neuron, and concept of learning/training. Model of an Artificial Neuron, transfer/activation functions, Neural Network architectures, Perceptron learning model, linear separability, Characteristics and Applications of Neural Network.

Module-II:

Multilayer Neural Networks: Feed Forward network - significance, training, loss function, Back-Propagation Network, Learning Factors of Back-Propagation Network, Radial Basis Function Network. Feedback network -Hopfield Nets: architecture, energy functions, training algorithms, competitive learning, self-organizing maps. Introduction to CNN and RNN network.

Module -III:

Fuzzy Systems: fuzzy set theory, fuzzy sets and operations, membership functions, concept of fuzzy relations and their composition, concept of fuzzy Measures. Fuzzy logic: fuzzy rules, inferencing, Features of the Membership Functions, Fuzzyfication, De-fuzzification: Lambda-Cuts for Fuzzy sets, Lambda-Cuts for Fuzzy Relations, De-fuzzification Methods, applications of fuzzy system.

Module -IV:

Fundamentals of Genetic Algorithms: Genetic Algorithms: History, Basic Concepts, Creation of Offspring's, Working Principle, Encoding, Fitness Function, Reproduction. Genetic Modelling: Inheritance Operators, Cross Over, Inversion, And Deletion, Mutation Operator, Bit-Wise Operators, Bit-Wise Operators used in GA, Generational Cycle, Convergence of Genetic Algorithms.

Course outcomes:

1. Learn about soft computing techniques and their applications

2. Analyze various neural network architectures

3. Understand perceptron and counter propagation networks.

4. Define the fuzzy systems and Analyze the genetic algorithms and their applications

Books Recommended

1. S.N. Sivanandam & S.N. Deepa, Principles of Soft Computing, Wiley Publications

2. S, Rajasekaran & G.A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic & Genetic Algorithms, Synthesis & applications, PHI Publication

3. Bose, Neural Network fundamental with Graph , Algo.& Appl, TMH Kosko: Neural Network & Fuzzy System, PHI Publication

4. Klir & Yuan ,Fuzzy sets & Fuzzy Logic: Theory & Appli.,PHI Pub. Hagen, Neural Network Design, Cengage Learning

MCA-600L	Design and Analysis of Algorithms Lab	2 Credits

Lab Assignments:

1. Insertion Sort/ Selection Sort
2. Divide and Conquer: Fibonacci search/Binary search
3. Divide and Conquer: Merge Sort/Quicksort/Heap Sort
4. Divide and Conquer: Convex hull/Finding closet pair
5. Dynamic Programming: MCM/LCS
6. Dynamic Programming: Rod Cutting problem /Assembly line Scheduling
7. Greedy method: Activity Selection/Huffman Coding
8. Graph Search: BFS/DFS
9. Graph Greedy MST: Kruskal/Prim's
10. Graph Greedy Shortest Path: Bellman ford/Dijkstra
11. Rabin Karp string matching algorithm/Subset Sum problem using Branch and Bound

Prerequisite: Each student should have a good knowledge on basic data structures like Stack, Queue, List, Heap, Matrix

MCA-602L	Python Programming Lab	2 Credits

Lab Assignments:

1. Basic Syntax and Operators
2. Conditional Statements
3. Loops and Patterns
4. Lists and Tuples
5. Strings and String Functions
6. Dictionaries and Sets
7. Functions and Recursion
8. Object-Oriented Programming
9. File Handling
10. Exception Handling
11. Python Database Connectivity