SYLLABUS & REGULATIONS OF 2-YEAR M.Sc. (COMPUTER SCIENCE) COURSE
(EFFECTIVE FROM ACADEMIC YEAR 2021-2022)
UNIVERSITY OF CALCUTTA

Semester I						
Course Code	**Course Type**	**Course Name**	**L-T-P**	**Sessional Marks**	**End Semester Marks**	**Credit**
CSMC101	Theory	Mathematics for Computing	3-1-0	30	70	4
CSMC102	Theory	Data Structures and Algorithms	3-1-0	30	70	4
CSMC103	Theory	Advanced Computer Architecture	3-1-0	30	70	4
CSMC104	Theory	Object Oriented Analysis and Design	3-1-0	30	70	4
CSMP105	Practical	**Module A:** Data Structure and Algorithms	0-0-3	30	20	2
		Module B: Object Oriented Systems	0-0-3	30	20	2
	Sem-Total		12-4-6	**180**	**320**	**20**
Semester II						
Course Code	**Course Type**	**Course Name**	**L-T-P**	**Sessional Marks**	**End Semester Marks**	**Credit**
CSMC201	Theory	Advanced Database Management System	3-1-0	30	70	4
CSMC202	Theory	Advanced Operating Systems	3-1-0	30	70	4
CSMC203	Theory	Automata & Compiler Design	3-1-0	30	70	4
CSMC204	Theory	Cryptography & Network Security	3-1-0	30	70	4
CSMP205	Practical	**Module A:** Advanced Database Management System	0-0-3	30	20	2
		Module B: Advanced Operating System	0-0-3	30	20	2
	Sem-Total		12-4-6	**180**	**320**	**20**
Semester III						
Course Code	**Course Type**	**Course Name**	**L-T-P**	**Sessional Marks**	**End Semester Marks**	**Credit**
CSME301	Theory	Image Processing & Pattern Recognition	3-1-0	30	70	4
CSME302	Theory	CBCS-I (from other Department)	3-1-0		50	4
CSME303	Theory	CBCS-II (from other Department)	3-1-0		50	4
CSMC304	Theory	Artificial Intelligence	3-1-0	30	70	4
CSMP305	Practical	Artificial Intelligence	0-0-3	30	20	2
CSMP306	Practical	Seminar	0-0-3	30	20	2
	Sem-Total		12-4-6	**120**	**280**	**16**
Semester IV						
Course Code	**Course Type**	**Course Name**	**L-T-P**	**Sessional Marks**	**End Semester Marks**	**Credit**
CSMG401	General	Elective-I	0-0-6	30	70	3
CSMC402	General	Elective-II	0-0-6	30	70	3
CSMG403	General	Project	0-0-16	120	80	6
CSMG404	General	General Viva Voce	0-0-0	-	100	2
	Sem-Total		0-0-22	**180**	**320**	**14**
	Total			**760**	**1240**	**70**

Sl. No.	**Elective I**	**Elective II**
1	**Advanced Software Engineering**	**Distributed & Cloud Computing**
2	**VLSI design**	**Wireless and Mobile Computing**
3	**Computational Geometry**	**GPU and Parallel Computing**
4	**Introduction to Machine Learning**	**Quantum Computing**
5	**Introduction to Data Science**	**Netwotk & Cyber Security**
6	**Natural Language Processing**	**Computational Biology**

SYLLABUS & REGULATIONS OF 2-YEAR M.Sc. (COMPUTER SCIENCE) COURSE
(EFFECTIVE FROM ACADEMIC YEAR 2021-2022)
UNIVERSITY OF CALCUTTA

Courses to be offered from the Department of Computer Science & Engineering for CBCS-I & CBCS-II

1. Introduction to Data Structures & Algorithms
2. Introduction to Programming in C
3. Introduction to Programming – Python
4. Advanced Excel & Security Fundamentals

Paper Code – CSMC101 **Paper Name – Mathematics for Computing**		**Marks: 100**
Module	**Topics**	**Lecture Hours**
Module-1 **Recurrence relations and solution methods**	Definition of recurrence relations, Formulating recurrence relations, solving recurrence relations-Backtracking method, Linear homogeneous recurrence relations with constant coefficients. Solving linear homogeneous recurrence relations with constant coefficients of degree two, Particular solutions of nonlinear homogeneous recurrence relation, Solution of recurrence relation by the method of generating functions Applications- Formulate and solve recurrence relation for Fibonacci numbers, Tower of Hanoi, Intersection of lines in a plane, Sorting Algorithms.	8
Module-2 **Graph theory**	Introduction to Graphs & its Applications, Trees, and Distance, Properties of Trees, Spanning Trees and Enumeration, Matrix-tree computation, Cayley's Formula, Matchings and Covers, Min-Max Theorem, Independent Sets, Covers and Maximum Bipartite Matching, Cuts and Connectivity, k-Connected Graphs, Vertex Coloring, Counting Proper Colorings. Planar Graphs, Characterization of Planar Graphs, Kuratowski's Theorem, Large Graph: representation, visualization and processing,	10
Module-3 **Matrix**	Matrix: Determinant and Trace / Eigenvalues and Eigenvectors / Cholesky Decomposition / Eigen decomposition and Diagonalization / Singular Value Decomposition / Matrix Approximation / Matrix Phylogeny	4
Module-4 **Linear Algebra**	Systems of Linear Equations / Solving Systems of Linear Equations / Vector Spaces. Linear Independence / Basis and Rank / Linear Mappings / Affine Spaces	4
Module-5 **Analytic Geometry**	Norms / Inner Products / Lengths and Distances / Angles and Orthogonality / Orthonormal Basis / Orthogonal Complement / Inner Product of Functions / Orthogonal Projections / Rotations	4
Module-6 **Probability**	Construction of a Probability Space / Discrete and Continuous Probabilities / Sum Rule, Product Rule, and Bayes' Theorem / Summary Statistics and Independence / Gaussian Distribution / Conjugacy and the Exponential Family /Change of Variables / Inverse Transform	10

1. Discrete Mathematics and its Applications, by Kenneth H. Rosen, McGraw-Hill Kenneth H. Rosen, McGraw-Hill
2. Discrete Mathematics, Lecture Notes, Yale University, Spring 1999,L. Lov'asz and K. Vesztergombi Mathematics for Computer Science, Lehman, Leighton & Meyer
3. Online resources complementing the book by Rosen
4. Graph Theory with Applications to Engineering & Computer Science : NARSINGH DEO
5. R. Ash & C. Doleans-Dade : Probability and Measure Theory
6. A. K. Basu : Measure Theory and Probability
7. G. Hadley : Linear Algebra

Paper Code – CSMC102 **Paper Name – Data Structures and Algorithms**		**Marks: 100**
Module	**Topics**	**Lecture Hours**
Module-1: **Advanced Data Structures**	**Review of Basic Data Structures:** Stacks, Queues, Arrays, Linked lists, BST, Balanced Tree Schemes, AVL trees, 2-3 trees, Red-Black tree, Hashing. **External Memory Data Structures:** B-tree, B*-tree. **Sorting:** Comparison-based and tree structure-based sorting algorithms. **Dictionary and Priority Queue:** Heap, Binomial heaps, Fibonacci heaps. **Set Manipulation:** Disjoint Set Data Structures-Union find algorithm.	16
Module-2: **Algorithm Design Paradigms**	**Amortized Analysis of Algorithms:** Comparison with asymptotic notations, Credit balance, Amortized cost. **Greedy Algorithm:** Activity scheduling, Huffman encoding, Greedy algorithms on matroids. **Divide and Conquer:** Selection algorithm, Integer multiplication. **Dynamic Programming:** Longest common subsequence, Optimal binary search tree	6
Module-3: **NP-Completeness**	**Complexity Classes:** P and NP, Cook's theorem, Reducibility, NP-completeness, NP-hardness. **Some NP-Complete Problems:** 3-SAT, Clique computation, Travelling salesman, Vertex cover, Hamiltonian cycle, 3-Coloring.	6
Module-4: **Coping with NP-Completeness**	**Approximation Algorithm:** Approximation algorithms vs. Approximation schemes, Vertex cover, Travelling salesman, Knapsack, Job-scheduling problems.	6
Module-5: **Network Flow and Matching**	**Maximum Flow Problem:** Flow properties, Max flow-Min cut theorem, Ford-Fulkerson algorithm, Edmond-Karp algorithm. **Matching Problem:** Bipartite matching for weighted and unweighted graphs.	6

Textbooks:

1. Introduction to Algorithms (Third Edition), T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, 2009.
2. Fundamentals of Computer Algorithms (Second Edition), E. Horowitz, S. Sahni, S. Rajasekaran, 2008.
3. Design and Analysis of Computer Algorithms, A. V. Aho, J. E. Hopcroft, and J. D. Ullman, Pearson, 1974.
4. Lecture notes of D. A. Mount, http://www.cs.umd.edu/class/fall2020/cmsc420-0201/Lects/cmsc420-fall2020-lects.pdf

References:

5. Algorithm Design, J. Kleinberg and E. Tardos, Pearson New International Edition, 2005.
6. Algorithms, S. Dasgupta, C. Papadimitriou, and U. Vazirani, Mc-Graw Hill Education (India) Edition, 2006.

Paper Code – CSMC103 **Paper Name – Advanced in Computer Architecture**		**Marks: 100**
Module	**Topics**	**Lecture Hours**
Module-1: **Introduction**	Computer Architecture & Organization. Basic Parallel Processing Architecture, Taxonomy- SISD. MISD, SIMD, MIMD structures, Serial, Parallel & Concurrent Computation, CISC vs RISC, Structure of Instruction of instruction sets and Desirable Attributes.	6
Module-2: **Pipelining**	Basic Concepts of pipelining, Instruction Pipelining. Hazards, Reservation Tables, Collision, Latency, Dynamic pipeline, Vector processing & Vector processors.	6
Module-3: **Memory Systems**	Cache Memory & Virtual Memory: Structure, Analysis & Design.	4
Module-4: **I/O Systems**	Design Issues, Performances Measures.	2
Module-5: **Multiprocessor Architecture**	Loosely Coupled & Tightly Coupled Systems, Concurrency & Synchronization, Scalability, Models of Consistency, Application of SIMD Structure.	3
Module-6: **Interconnection Network**	Definition. Types of Interconnected Networks; Baselines, Shuffle-Exchange, Omega, Cuba, Comparison & Application.	5
Module-7: **Systolic Architecture**	Systolic processor, Mapping Algorithm to array structures, Mapping design & Optimization, Systolization Procedure	5
Module-8: **Data Flow Architecture**	Data Flow Architecture, Different forms of DFA, Data Flow Graphs, Petri nets	3
Module-9: **Programming Environment**	Different Models, Languages, Compilers, dependency Analysis. Message Passing, Program mapping to Multiprocessors, Synchronization	4
Module-10: **Case Study**	Basic Features of Current Architectural Trends. DSP Processor, Multicore Technology	2

Text book:

1. John L. Hennessey and David A. Patterson, "Computer Architecture A Quantitative Approach", Morgan Kaufmann/ Elsevier, Fifth Edition, 2012.
2. Kai Hwang and Faye Briggs, "Computer Architecture and Parallel Processing", Mc Graw-Hill International Edition, 2000.

Paper Code – CSMC104 **Paper Name – Object Oriented Analysis and Design**		**Marks: 100**
Module	**Topics**	**Lecture Hours**
Module-1: **Object Oriented Programming Concepts**	Abstraction and Encapsulation, Generalization and Specialization, Functional Decomposition and Object Oriented Decomposition, Coupling and Cohesion, Modularity and Hierarchy, Relationships among Classes, Relationships among Objects, Identification of Classes, Objects and Relationship, Dynamic Dispatch and Dynamic Binding, Generic Programming	8
Module-2: **Object Oriented Analysis and Design with UML**	Overview of UML, SDLC Phases and UML Diagrams, **Static UML(Structural) Diagrams:** Class Diagrams, Object Diagram, Component Diagram, Package Diagram, Composite Structure Diagram **Dynamic UML(Behavioral) Diagrams:** Activity Diagram, Use Case Diagram, State Machine Diagram, Sequence Diagram, Communication Diagram, Interaction Overview Diagram	12
Module-3: **Design Patterns for reusable Object Oriented Designs**	Design Pattern basics, Benefits of using Design Pattern, **Creational Pattern:** Reflections, Singleton, Object Pool, Factory, Abstract Factory, Builder, Prototype **Structural Pattern:** Adapter, Bridge, Composite, Decorator, Flyweight, Facade, Proxy **Behavioral Pattern:** Interpreter, Template Method, Chain of Responsibility, Command, Iterator, Mediator, Memento, Observer, State, Visitor, Strategy	20

Textbook:

1.Object-Oriented Software Development Using Java. Xiaoping Jia. Addison Wesley, ISBN 0-201-73733-7.

References:

2. Head First Object-Oriented Analysis and Design. Brett D. McLaughlin, Gary Pollice, and Dave West. O'Reilly.
3. Head First Design Patterns. Eric Freeman and Elizabeth Freeman. O'Reilly.

Paper Code – CSMP105 **Paper Name – MODULE 1: Data Structure and Algorithms** **MODULE 2: Object oriented Systems**		**Marks: 100**
Module	**Topics**	**Lecture Hours**
Module 1: **Data Structure and Algorithms**	**Data Structure Practical:** Representations of Graphs; Application of stacks, queues, and arrays; DFS/BFS and their applications; Representation of polynomials and their operations; Representation of sparse matrices; Trees and their applications; Representation of balanced tree schemes; B-tree and B*-tree; Binomial heaps and Fibonacci heaps; Disjoint set data structures; Amortized Analysis of Algorithms **Algorithms Practical:** Sorting algorithms; Divide and Conquer algorithm: Finding the *k*-th smallest number, Large Integer Multiplication; Greedy Algorithms: Huffman encoding; Dynamic Programming: Optimal binary search tree; Maximum Flow problem: Ford-Fulkerson algorithm; Matching problem: Bipartite Matching algorithm.	6 hours per week
Module 2: **Object Oriented Systems**	**Basic object oriented features:** Constructors and Destructors, Method Overloading and Overriding, Inheritance, Modifiers and Access Specifiers, Visibility Control, Exception Handling. **UML Tools and Design Patterns:** Identify use cases and develop the Use Case model, Identify the conceptual classes, develop Domain Models and derive Class Diagrams; Represent interactions between objects using UML Sequence and Collaboration Diagrams; State Chart and Activity Diagrams; Identify and apply appropriate design patterns for the above steps.	6 hours per week

Paper Code – CSMC201 **Paper Name – Advanced Database Management System**	**Marks: 100**
TOPICS	**HOURS**
Relational Database Design: Problem Solving on Normalization & Functional Dependency, Multi-valued Dependencies; Indexing: Cost model of basic file Organizations like Heap file, Sorted file, Hashed file, Need of Indexing and Hashing, Tree Structured Indexing: ISAM, B+ Tree; Hash based Indexing: Static, Extendable, Linear Hashing schemes, Index selection guideline with small use cases.	**8**
Query Processing and Optimization: Query Evaluation: External Sorting, Evaluation of relational Operators (Select & Join) including Join algorithms, Query Optimization: Heuristic based & Cost based optimization, Structure of Query Optimizer with small use case. Database tuning through reframing schema, query, view.	**6**
Concurrency Control and Recovery: Transaction & Schedule, ACID property, Serializabilty, Anomalies with Interleaved execution, Conflict & View serializability, Concurrency Control techniques: Locking and Timestamp based protocols, Multi-version and Validation based schemes, Multiple Granularity locking, Deadlock handling, Crash Recovery: ARIES, Recovery Data structure Log, Write Ahead Logging, Check-pointing, Recovery from a system crash.	**10**
Alternative Data Models: Weakness of RDBMS, Contribution of Object & Object Relational Data Model, Semi Structured and Unstructured Data Handling in Database application: XML Document-DTD-XML Schema, XPath- XQuery	**6**
Distributed Database: Architecture, Fragmentation and Allocation Transparency, Basic Concept of Distributed Database Design (through small use cases), Sharding and Replication, CAP Theorem.	**5**
No SQL Databases: Features of various types of No-SQL databases, Brief Concept on Key-value database, Document Store, Column Family Stores and Graph databases.	**5**

Books:

1. Avi Silberschatz, Hank Korth, and S. Sudarshan, "Database System Concepts", 6 th Ed. McGraw Hill, 2010.
2. Ramez Elmasri, B.Navathe, "Fundamentals of Database Systems", 7th edition, Addison Wesley,2014
3. Sadalage, P., Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pearson Addision Wesley, 2012
4. Database System Concepts, Abraham Silberschatz, Henry Korth, and S. Sudarshan, McGraw-Hill.

Paper Code – CSMC202 **Paper Name – Advanced Operating System**	**Marks: 100**
MODULES	**HOURS**
Introduction: Distributed system, distribution transparency, loosely couple versus tightly coupled system architecture, distributed shared memory.	**3**
Message Passing: Inter-process communication, group communication, broadcasting algorithms, case study with MPI.	**3**
Clock synchronization: Event ordering, event precedence, Logical Clock model, Vector Clock.	**3**
State Recording: Naïve State Recording algorithm, Chandy-Lamport's state recording algorithm.	**3**
Mutual exclusion in distributed environment: Ricart-Agrawala Algorithm, Token based ME algorithm for Ring topology, Raymond's Algorithm	**6**
Deadlock detection for Distributed Systems: Mitchell-Merritt Algorithm, Ho-Ramamurthy Algorithms, Termination Detection algorithm – weight-throwing algorithm, diffusion computation algorithm.	**4**
Process Management: Process migration, Pre-emptive and non-preemptive process migration, Resource migration, Resource-Process binding, Resource-Machine binding, Challenges and limitations for creating global references for resources	**6**
Remote Procedure Call: RPC Model, stub generation, server management, parameter passing, call semantics, communication protocols, Client-Server binding, exception handling, security, optimization, Case study on SUN RPC.	**8**
Distributed File Systems: System wide deployment of DFS, Client-Server model for DFS, Indexed Block model for DFS, Case study on Hadoop File System, Naming in distributed systems, directory services, DNS.	**4**

Books:

1. Mukesh Singhal, Niranjan Shivaratri: Advanced Concepts in Operating Systems, Tata McGraw-Hill Education, 2001.
2. S. Tanenbaum: Distributed Operating Systems, Prentice Hall of India, New Delhi, 1996.
3. G. F. Colouris, J. Dollimore and T. Kindberg: Distributed Systems: Concepts and Design, 2nd ed., Addison-Wesley, Reading, Mass., 1994.
4. S. J. Mullender (Ed.): Distributed Systems: An Advanced Course, 2nd ed., Addison-Wesley, Reading, Mass., 1993.
5. P. K. Sinha: Distributed Operating Systems, IEEE Press, Los Alamos, California, 1997.

Paper Code – CSMC203 **Paper Name – Automata Theory and Compiler Design**	**Marks: 100**
MODULES	**HOURS**
I. Automata Theory	
Introduction: Definition of a Finite Automaton, Non-deterministic Finite state Automaton (NDFA), Deterministic Finite state Automaton (DFA), NDFA to DFA conversion, Minimized Equivalent Machine, State minimization algorithm–row elimination method, Implication Table Method, Basics of regular expression.	**3**
Formal languages and grammar: Introduction to Formal Grammar and Language, Chomsky's Classification of Grammar – Type 0, Type-1 or Context Sensitive, Type-2 or Context Free and Type-3 or Regular Grammar, CNF, GNF. Illustration of each of these classes with example, Derivation tree, Parse Tree, Syntax Tree, Ambiguous and Unambiguous Grammar. Regular expression to Finite Automata conversion, FA to Regular Grammar and Regular Grammar to FA conversion	**3**
Push-down automata (PDA): Definition, PDA and CFL: design and conversion, acceptance of strings	**3**
Turing Machine: Introduction, Turing Machine Model, computable languages and function	**4**
II. Compiler Design	
Introduction to Compiling Introduction, Analysis-synthesis model, Phases of the compiler.	**2**
Lexical analysis Role of lexical analyser, Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of tokens, Design of a lexical analyser generator (Lex).	**2**
Syntax analysis The role of a parser, Context free grammars, Writing a grammar, Top down Parsing, Non-recursive Predictive parsing (LL), Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR), Parser generators (YACC). Error recovery strategies for different parsing techniques, syntax directed translation, Syntax directed definitions, construction of syntax trees, bottom-up evaluation of S-attributed defn., L-attributed definitions, bottom-up evaluation of inherited attributes	**9**
Run time environment: Parameter passing, symbol table, dynamic storage allocation techniques	**2**
Intermediate code generation: Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples)	**4**
Code generation and optimization: Issues in the design of code generator, a simple code generator, Register allocation and assignment, Introduction to code optimization, Basic blocks & flow graphs, Transformation of basic blocks, DAG representation of basic blocks, the principle sources of optimization, Loops in flow graph, Peephole optimization.	**8**
Books: 1. J. E. Hopcroft, R. Motwani, J. D. Ullman, "Introduction to Automata Theory, Language and Computation, Addision-Wesley. 2. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, Compilers: Principles, Techniques and Tools, Addison-Wesley. 3. Andrew W. Appel, Modern Compiler Implementation in C/Java, Cambridge University Press 4. Keith D. Cooper and L Torczon, Engineering a Compiler, Elsevier 5. Allen I. Holob, Compiler Design in C, Prentice-Hall 6. Stevens S. Muchnik, Advanced Compiler Design and Implementation, Elsevier	

Paper Code – CSMC204 **Paper Name – Cryptography and Network Security**	**Marks: 100**
MODULES	**HOURS**
Introduction: Classification of Possible attacks, Traditional Encryption Techniques: Affine, Play Fair, Hill cipher and Vernam cipher with subsequent strength analysis. Symmetric key & Asymmetric key cryptography, Block & Stream Cipher, Stream Cipher generation technique LFSR.	**4**
Symmetric Key Cryptography: Modular Arithmetic, Extended Euclidean Algorithm, Group, Ring and Finite Field, Polynomial Arithmetic, Shannon's Theorem, Feistel structure, DES and AES algorithm with strength analysis, Diffie Hellman Key Exchange Problem & Man-in-the Middle attack, 2 DES and 3 DES. Algorithmic Modes.	**9**
Asymmetric Key Cryptography: Fermat's and Euler's Theorem, Primality Testing, Discrete Logarithm, The Chinese Remainder Problem, RSA, Elgamal, Eliptic Curve algorithms with necessary mathematical analysis,	**9**
Message Integrity: Hash function, Hash function criteria, Evaluating the security of Cryptographic hash functions, MAC, Brief idea on MD5, SHA-1, H-MAC.	**3**
Authentication techniques: Password Based and Challenge Response based authentications, Role of KDC in Key-exchange and Authentication, Needham Schroder algorithm, Kerberos	**3**
Security layers in Network Protocol Stack: IP Sec, AH & ESP, Transport & Tunnel Modes, Security Association, IKE protocol, Secure Socket Layer, Security protocols used in Application layer like PGP, SHTTP etc.	**6**
Digital Signature: Concepts and the techniques through RSA, Basics of Steganography.	**4**
Network Defense tools: Firewalls, Intrusion Detection, Filtering, Security in Mobile Platforms: Threats in mobile applications, analyzer for mobile apps to discover security vulnerabilities.	**2**

Books:

1. Wiliam Stallings, Cryptography and Network Security, Sixth Edition, Pearson.
2. B.A. Forouzan, Cryptography and Network Security, Special Indian Edition, TMH publishing Company Limited.
3. Atul Kahate, Cryptography and Network Security, Tata McGraw Hill Publication.
4. Bruce Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd Edition, Willy Publication.

Paper code- CSME301 **Paper Name- Image Processing and Pattern Recognition**		**Marks: 100**
Module	**Topics**	**Hours**
Module-1: **Image Fundamentals**	Analog and digital images, image sensing and acquisition: Image formation, Sampling and quantization, Color space: Color (RGB, CMYK, HSI) vs gray level images, Matrix representation and intensity modification of digital images, Pixel adjacency and distance measure, Arithmetic, logical and set operations, Image file formats, Fundamental steps in DIP, Applications and state of the art in DIP.	6
Module-2: **Transformation and Filtering**	Point processing: Identity, image negatives, log transform, power law, contrast stretching, histogram equalization and specification. Spatial filtering: Linear filters: max, min, mean, median; order statistics filters. Frequency based transforms: Low and high pass filter, DFT Image restoration concept: Noise models, Image denoising and deblurring	10
Module-3: **Image segmentation**	Segmentation techniques, Threshold based segmentation, Importance of derivative and gradients in edge detection, Masks: Roberts, Prewitt, Sobel; Canny edge detection, Region growing and Split-Merge algorithms, Clustering based techniques, basics of Hough transform.	9
Module-4: **Image Compression**	Compression basics: Lossless, lossy, compression ratio, image compression models, evaluation criteria of a compression scheme, compression techniques: Huffman encoding, Run length, Arithmetic encoding.	5
Module-5: **Pattern recognition**	Introduction and applications. Feature extraction and reductions: Histogram of Gradient (HoG), Principal Component Analysis (PCA). Learning: Supervised and unsupervised; Clustering and Classification techniques: K-Nearest Neighbor Classifier, Support Vector Machine, K-means algorithm, Density-based Clustering.	10

Textbooks:

1. Digital Image Processing by Rafael C. Gonzalez, Richard E. Woods; Pearson; 4th edition (2017)
2. Image Processing: Principles and Applications by by Tinku Acharya, Ajoy K.Ray; Wiley-Interscience; 1st ed. (2005)
3. Digital Image Processing by William K. Pratt; John Wiley & Sons; 4th Edition (2007)
4. Digital image processing with MATLAB and LabView, Vipula Singh, Elsevier, 2013.
5. Pattern Classification by Richard O. Duda, David G. Stork, Peter E.Hart, Wiley; Second edition (2007)
6. Pattern Recognition by Sergios Theodoridis and Konstantinos Koutroumbas, Academic Press, 2008.
7. Pattern Recognition and Machine Learning by Christopher M. Bishop and Nasser M. Nasrabadi., New York: Springer, 2006.
8. Pattern recognition principles, Tou and Gonzalez, Addison Wesley, 1974.

Paper code- CSMC304 **Paper Name- Artificial Intelligence**		**Marks: 100**
Module	**Topics**	**Lectures**
I. Introduction to AI	Turing Test and Rational Agent approaches to AI; Distributed AI; Applications	2
II. Introduction to state Space search	Agents & environment, nature of environment, structure of agents, goal-based agents, utility-basedagents, learning agents. Problems, Problem Space & search: Defining the problem as state space search, Water Jug Problem; production system, problem characteristics, issues in the design of search programs. Solving problems by searching: Problem solving agents, searching for solutions; uniform search strategies: breadth first search, depth first search, depth limited search, bidirectional search, comparing uniform search strategies	6
III. Heuristic search	Greedy best-first search, A* search, AO* algorithm; memory bounded heuristic search: local search algorithms & optimization problems: Hill climbing search, simulated annealing search, local beam search, constraint satisfaction problems, local search for constraint satisfaction problems. Adversarial search: Games, optimal decisions & strategies in games, the minimax search procedure, alpha-beta pruning, additional refinements, iterative deepening.	6
IV. Knowledge representation and Reasoning	Knowledge representation issues, representation & mapping, approaches to knowledge representation, issues in knowledge representation Predicate logic: Representing simple fact in logic, Modus ponen and tollen; Common Sense; representing instant & ISA relationship, computable functions & predicates, resolution, natural deduction; Representing knowledge using rules: Procedural verses declarative knowledge, logic programming, forward verses backward reasoning, matching, control knowledge.	8
V. Soft Computing Approaches	Overview, Representing knowledge in an uncertain domain, the semantics of Bayesian networks, Dempster- Shafer theory, Fuzzy vs Crisp; Fuzzy sets & fuzzy logic. Rough set; Genetic Algorithm: Multi-objective optimization, Pareto optimal front	6
VI. Neural Network Learning	Biologically Inspired model, Various activation functions; Perceptron; Backpropagation: Gradient Decent; MAXNET; ADALINE, MADALINE, SOM, ART	8
VII. Expert system	Definition; Features of an expert system; Organization; Characteristics; Prospector; Knowledge Representation in expert systems; Expert system tools Representing and using domain knowledge; expert system shells, knowledge acquisition	4

Books:

1. Dan.W. Patterson, Introduction to AI and Expert Systems – PHI, 2007
2. Stuart Russel and Peter Norvig, 'Artificial Intelligence - A Modern Approach', Second Edition, Pearson Education, 2003 / PHI.
3. George F. Luger, 'Artificial Intelligence – Structures and Strategies for Complex Problem Solving', Fourth Edition, Pearson Education, 2002.
4. Elain Rich and Kevin Knight, 'Artificial Intelligence', Second Edition Tata McGraw Hill, 1995.
5. Simon Haykin, "Neural Networks and Learning Machines", Prentice Hall, 2009
6. George J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic: Theory and Applications", Prentice-Hall (1995).

CSMP305: Artificial Intelligence (Practical)

Programming language - Small / tiny models will be developed from the scratch to study the method in Python language without build-in library for the technique.
Results will be compared with build-in library.

1. Assignments on Heuristic searches
2. Assignments on Constraints Satisfaction like CNF etc.
3. Assignments on branch and bound problems
4. Assignments on Gradient Descendent Search
5. Assignments on Genetic Algorithm to simple LLP Problems.
6. Assignments on Classification Algorithms – KNN, ANN
7. Assignments on Clustering Algorithms - Kmeans, Kmedoides, Density Based Clustering

SYLLABUS & REGULATIONS OF 2-YEAR M.Sc. (COMPUTER SCIENCE) COURSE
(EFFECTIVE FROM ACADEMIC YEAR 2021-2022)
UNIVERSITY OF CALCUTTA

Examination Regulations for Two-Year Post GraduateM.Sc. Courses (Under CBCS) with Computer Science (with effect from the academic year 2021-2022)

1. The Two-Year Post Graduate M.Sc. Course with Computer Science (Under CBCS) shall be for a minimum duration of four (4) consecutive semesters of six (6) months each, i.e., two (2) years. The odd Semester will start ordinarily in July and the even Semester in January of every year. A candidate pursuing a regular course of study for a two-year Post Graduate M.Sc. Course with Computer Science (Under CBCS) **shall have to clear all the semesters in all respects within four years from the year of admission to the particular course and combination, failingwhich enrolment of the candidate shall stand canceled.**
2. End semester examinations are to be held ordinarily at the end of the concerned Semester, i.e., Semester I and Semester III in December-January and Semester II and Semester IV in June-July.
3. A candidate shall be eligible for appearing at any of the Semesters of PG Examination, provided he/she prosecutes a regular course of studies in the concerned subject offered by the University keeping the percentage of attendance as specified bellow-
    a) A student attending at least 75% of the total number of classes* held shall be allowedto appear at the concerned Semester Examination subject to fulfillment of other conditions laid down in the regulations.
    b) Students attending at least 60% but less than 75% of the total number of classes* held shall, however, be eligible to appear at the concerned Semester Examination. They have to take condonation orders from the Vice-Chancellor (Principal for affiliated colleges) and on payment of requisite condonation fees for short attendance fees as may be prescribed by the University from time to time.
    c) Students attending less than 60% of the total number of classes* held shall not be allowed to appear at the concerned Semester Examinations. They have to take re-admission to the same Semester in the next year to attend the classes and appear at the said Semester Examination subject to fulfillment of clause (a) or (b) above.

    ** Such attendance will be calculated from the date of commencement of classes or date of admission, whichever is later.*
4. Theoretical examinations would be held after completing the curriculum at the end of each Semester (clause 2). Duration of theoretical Examinations:

Marks	Duration
Upto 25	1 hour
26 to 50	2 hours
51 to 80	3 hours
81 to 100	4 hours

5. A candidate must appear (in examination) in each paper / course / module/part / group of the respective syllabus. To be declared to have passed an examination, a candidate must obtain at least 40% marks (including Internal Assessment) in each paper/course. In the case of a paper/course containing both theoretical and practical portions, a candidate is required to secure at least 40% marks (including Internal Assessment) separately in the theoretical and practical portions.
6. A candidate who is **eligible** to appear at any of the End Semester Examinations but does not enroll or appear at the examinations or fails to secure pass marks in any paper(s) at theconcerned examinations, as stated in Clauses 5 above, will be allowed to attend the classes in the next higher Semester, as applicable.
7. Internal Assessment has to be done in the Semester in which a candidate becomes eligible to appear in the concerned end semester examination. The candidates remaining absent in the written exam for Internal Assessment will be awarded zero (0) marks in the written part of the internal examination.
   Marks obtained in Internal Assessment (i.e., marks for attendance, if any, and marks of written examination or any other component of internal assessment, taken together) shall be retained for the entire duration of his / her enrolment.

8. Suppose a candidate secures pass marks in Practical Paper(s)/Module(s) /Project Work but fails to secure pass marks in other papers. In that case, the marks of Practical Paper(s)/Module(s) /Project Work along with Internal Assessment of the other papers shall be retained for the entire duration of his/her enrolment.
9. A candidate may continue his/her course of study for the next higher Semester and appear at any higher End Semester examination without appearing at the lower End SemesterExaminations subject to Clause 3(c).
10. A candidate who fails to secure pass marks (as stated in Clause 5 above) in one or more semester papers may appear in those paper(s) when the concerned End Semester Examinationswill be held next.
11. Suppose a candidate fails to secure pass marks in one or more papers in 3rd and/or 4th End-semester examinations. In that case, he/she may appear in the supplementary examinations to be held after three months of publication results of 4^{th}-semester examinations.
12. Non-appearance (absence) in any examination for any reason shall be counted as a chance. Failure to fill up the examination form shall also be considered as a missing chance.
13. A failed candidate intending to re-appear in a subsequent semester examination has to submit an examination form/application to the Controller of Examinations through the Head of the Department (Principal for affiliated colleges) as per the notice to be issued by the Department ofController of Examinations. **The candidate has to clear the entire course within four years from the year of first admission in that course**.
14. The schedule for the End-Semester Examination shall be prepared and announced by the Controller of Examinations in consultation with the Post Graduate Board of Studies concerned.
15. Names of the paper-setters (at least one internal paper setter from the concerned academic Department of this University and one external*/substantive teacher of the affiliated colleges having teaching experience of the paper concerned in PG Course), head-examiner (preferably regular faculty of the concerned academic Department of the University), examiners, scrutineers, re-examiners, third examiners (if required) of each subject/course/paper and board of moderators (regular faculty members of the concerned academic Department of this University and at least one external*) shall be recommended by the Post Graduate Board of Studies and approved by the Vice-Chancellor.

**Any expert having teaching experience in Master Degree and are not engaged in teachingeither in concerned academic Department of this University or any affiliated college of this University (if viable otherwise).*

16. Post-Graduate Board of Studies will recommend the following for approval of the higher authorities.
    a. The medium of communication of the entire course will be in the English language only.
    b. The BoS of the corresponding examination will decide the exact question pattern of theoretical papers.
    c. The BoS of the corresponding examination will decide the modalities of internal assessment for theoretical papers.
    d. The BoS of the corresponding examination will decide the modalities of practical papers.
    e. The BoS of the corresponding examination will decide the modalities of submission and evaluation of the dissertation or project.
    f. The Bos of the corresponding examination will decide the modalities of holding the General viva-voce examination.
17. An examination shall always be held under the current syllabus (if viable otherwise).
18. Question papers shall be set in English Version. The candidates shall have to write their answers in English Language only.
19. The provisional result of each Semester will be published showing the details of courses studied (code, title, marks secured, credits, grade point, and letter grade) along with SGPA of that Semester and Cumulative Grade Point Average (CGPA) of the last Semester.
20. Award of Degree
    a) The final result shall be determined based on CGPA on a 10 point scale.
    b) Grade Card shall be made as per the grading system. Course-wise marks (Internal and End Semester

Examination added together) will be converted into a percentage of marks. Percentage of marks will be converted into Letter Grade and Grade Point. Credit and Grade points will be converted into Credit points. Finally, Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA) will be computed.

c) The Grade Card of a Semester shall be issued only after completing that Semester.

d) The date of publication of the final result for a regular student, who shall clear all the semesters in the normal course, shall be the date of publication of the result of the 4th Semester. The final date of publication of results for students clearing previous Semester (s) after their clearing 4th Semester examination shall **be the date of publication of the last result clearing all papers.**

e) Grading and marking system will be followed as-

% of Marks (M)*	Letter Grade	Grade Point (GP)
$M \geq 90$	O (Outstanding)	10
$80 \leq M < 90$	A+ (Excellent)	09
$70 \leq M < 80$	A (Very Good)	08
$60 \leq M < 70$	B+ (Good)	07
$50 \leq M < 60$	B (Above Average)	06
$40 \leq M < 50$	C+ (Average)	05
$M < 40$	F (Fail)	00
Absent	AB	00

21. A candidate failing to obtain the pass marks in a semester examination due to shortage of one mark (in aggregate / in any course) shall be given the benefit of one additional mark in the paper in which he/she secured the lowest marks and the same shall be shown in the Tabulation Rolls. However, only the total marks shall be shown in the Mark-sheet after such addition.

22. The performance of a candidate in *n*-th Semester examination, who earns all the Credit of that Semester, will be assessed by the "**Semester Grade Point Average**" (**SGPA**) , "S_n" to be computed as :

$$\text{SGPA}[S_n] = \frac{\sum_{k=1}^{D_n} C_{nk}\ GP_{nk}}{\sum_{k=1}^{D_n} C_{nk}}$$

where, GP_{nk} denotes the grade point obtained in *k*-th Course of *n*-th Semester;
C_{nk} denotes the credit of *k*-th Course of *n*-th Semester; D_n denotes the number of Courses in *n*-th Semester.

Example of Computation of SGPA:

Course	Credit	Letter Grade	Grade Point	Credit Point
Course 1	3	A	8	3×8=24
Course 2	4	B+	7	4×7=28
Course 3	3	B	6	3×6=18
Course 4	4	B	6	4×6=24
Total	**14**			**94**

SGPA = 94/14 = 6.71.

23. On completion of the Post Graduate course, the overall performance of a candidate will be assessed by the 'Cumulative Grade Point Average' (CGPA) to be computed as:

$$CGPA[C] = \frac{\sum_{n=1}^{4} Q_n \ S_n}{\sum_{n=1}^{4} Q_n}$$

$$where, Q_n = \sum_{k=1}^{D_n} C_{nk} = total \ \ credit \ \ in \ \ nth \ \ Semester$$

Example of Computation of CGPA:

	Semester 1	Semester 2	Semester 3	Semester 4
SGPA	6.71	6.31	6.60	6.71
Credit	14	16	15	14
SGPA × Credit	94	101	99	94

CGPA: 388 (i.e. 94+101+99+94)/59 (i.e. 14+16+15+14) = 6.58.

24. Final Grade and Class will be determined as follows-

CGPA(C)	Letter Grade	Class
$9.00 \leq C \leq 10.00$	O (Outstanding)	I
$8.00 \leq C < 9.00$	A^+ (Excellent)	I
$7.00 \leq C < 8.00$	A (Very Good)	I
$6.50 \leq C < 7.00$	A (Good)	I
$5.00 \leq C < 6.50$	B (Above Averaage)	II
$4.00 \leq C < 5.00$	B (Below Average)	II
$C < 4.00$	F	Fail

Both SGPA and CGPA will be rounded off to the second place of the decimal and will be shown on the mark sheet.

25. Candidates appearing in a semester examination shall have to take admission to the next higher Semester (wherever applicable) immediately after completion of examination.

26. RULES FOR RE-EXAMINATION:

Candidates seeking review may apply to the University in a prescribed form along with requisite fees within seven (7) working days from the date of issue of Grade Card subject to the following conditions:

a) Application for review shall be restricted to theoretical modules/courses/papers only.

b) A candidate would be eligible to re-examine his / her script if and only if he/she appeared in the saidSemester of examination as a whole but not appeared as a supplementary candidate.

c) Maximum two (02) theoretical modules/courses in any semester examination may be re-examined on request by the examinee subject to the condition that they secure a minimum of 40% marks in the rest of theoretical modules/courses in a semester.

d) In a re-examination of any theoretical paper(s) of a semester in the course, the marks awarded by the re-examiner will be taken as the marks obtained by the examinee in the re-examined paper.

e) Suppose on re-examination in a module/course the marks get enhanced by more than 15% or get reduced by more than 5% than that awarded by the original examiner (the percentage be calculated based on the total marks in that module/course). In that case, the script of the module/course is referred to a third examiner, and the average of two marks (excluding the lowest one) as awarded by the three examiners shall be taken as the marks obtained by the examinee in that module/course, provided that such a final award does not result in the failure of the examinee or in lowering of the Letter Grade of SGPA/CGPA / Class or status obtained by the examinee before re-examination in which cases the original awardbe retained.

27. **Cancellation of Examination:**

Candidate may apply for cancellation of enrolment of any end semester examination within ten (10) days from the completion of the theoretical paper examinations. The said cancellation of the examination will also be counted as a chance.

28. After Re-examination, FSI, and another process of Final Semester Examination, a Rank Certificate in order of Merit shall be issued to-

a) First ten candidates (in each subject who has completed/cleared all the semester examinations at the first available chance) with first-class (CGPA $\geq$ 6.50) marks in aggregate (second class will not be considered).

b) The first three candidates (in each subject who has completed/cleared all the semester examination at the first available chance) are in second class in aggregate, where no First class candidate is available.

29. Each candidate shall receive his/her degree in the form of a Diploma/Certificate stating the year of passing, letter grade, CGPA, and class on successful completion of the course in the specific format-

(Roll No……) Diploma No……

(Registration No.....)

University of Calcutta

LOGO

This is to certify that ……..(Name)……obtained the degree of Master of Arts/Science/Commerce in …… (Subject)…… in this University in the semester system examination (under CBCS) in theyear……., and that he/she obtained / was placed in ……(Letter) ….Grade with CGPA .....and …… class.

Senate House, Vice-Chancellor

(Date of publication of result)

Logo embossed