



IISER
B E R H A M P U R

BS-MS SYLLABUS

Computer & Data Sciences

I Semester

| Course No. | Course Name | LecHr | Lab Hr | Tut Hr | SS Hr | Credit |
|----------------|---------------------------------------|-----------|-----------|-----------|-----------|-----------|
| <u>BIO 101</u> | Introduction to Biological Sciences | 3 | 0 | 1 | 6 | 3 |
| <u>BIO 103</u> | General Biology Laboratory I | 0 | 3 | 0 | 0 | 1 |
| <u>CHM 101</u> | General Chemistry | 3 | 0 | 1 | 6 | 3 |
| <u>CHM 103</u> | General Chemistry Laboratory | 0 | 3 | 0 | 0 | 1 |
| <u>CDS 101</u> | Introduction to Computers | 2 | 1 | 0 | 6 | 3 |
| <u>HSS 103</u> | Basics of Communication Skills | 1 | 0 | 0 | 2 | 1 |
| <u>MTH 101</u> | Introduction to Mathematics | 3 | 0 | 1 | 6 | 3 |
| <u>PHY 101</u> | Mechanics | 3 | 0 | 1 | 6 | 3 |
| <u>PHY 103</u> | Mathematical Methods | 1 | 0 | 1 | 0 | 1 |
| <u>EES 101</u> | Introduction to Earth System Sciences | 3 | 0 | 1 | 6 | 3 |
| Total | | 19 | 07 | 06 | 38 | 22 |

II Semester

| Course No. | Course Name | LecHr | Lab Hr | Tut Hr | SS Hr | Credit |
|----------------|--|-----------|-----------|-----------|-----------|-----------|
| <u>BIO 102</u> | Biochemical and Cellular basis of life | 3 | 0 | 1 | 6 | 3 |
| <u>BIO 104</u> | General Biology Laboratory II | 0 | 3 | 0 | 0 | 1 |
| <u>CHM 102</u> | Basic Inorganic Chemistry | 3 | 0 | 1 | 6 | 3 |
| <u>CHM 104</u> | Inorganic Chemistry Laboratory I | 0 | 3 | 0 | 0 | 1 |
| <u>HSS 104</u> | Oral and Written Communication | 1 | 0 | 0 | 2 | 1 |
| <u>EES 102</u> | Introduction to Environmental Sciences | 3 | 0 | 1 | 6 | 3 |
| <u>MTH 102</u> | Calculus of One Variable | 3 | 0 | 1 | 6 | 3 |
| <u>PHY 102</u> | Electromagnetism | 3 | 0 | 1 | 6 | 3 |
| <u>PHY 104</u> | General Physics Laboratory I | 0 | 3 | 0 | 0 | 1 |
| Total | | 16 | 09 | 05 | 32 | 19 |

III Semester

| Course No. | Course Name | LecHr | Lab Hr | Tut Hr | SS Hr | Credit |
|-------------------------|--|-----------|-----------|-----------|-----------|-----------|
| BIO 201 | Introduction to Genetics and Evolution | 3 | 0 | 1 | 6 | 3 |
| BIO 203 | General Biology Laboratory III | 0 | 3 | 0 | 0 | 1 |
| CHM 211 | Basic Organic Chemistry | 3 | 0 | 1 | 6 | 3 |
| CHM 213 | Organic Chemistry Laboratory I | 0 | 3 | 0 | 0 | 1 |
| EES 201 | Foundation of Earth Sciences: Part 1 (Introduction to Mineralogy, Petrology) | 3 | 0 | 1 | 6 | 3 |
| HSS 209 | Technical Writing | 2 | 0 | 0 | 4 | 2 |
| MTH 201 | Linear Algebra | 3 | 0 | 1 | 6 | 3 |
| PHY 201 | Waves and Introductory Optics | 3 | 0 | 1 | 6 | 3 |
| PHY 203 | General Physics Laboratory II | 0 | 3 | 0 | 0 | 1 |
| Total | | 17 | 09 | 05 | 34 | 20 |

IV Semester

| Course No. | Course Name | LecHr | Lab Hr | Tut Hr | SS Hr | Credit |
|-------------------------|---|-----------|-----------|-----------|-----------|-----------|
| BIO 202 | Molecular Biology and Developmental Biology | 3 | 0 | 1 | 6 | 3 |
| BIO 204 | General Biology Laboratory IV | 0 | 3 | 0 | 0 | 1 |
| CHM 222 | Classical Thermodynamics | 3 | 0 | 1 | 6 | 3 |
| CHM 224 | Physical Chemistry Laboratory I | 0 | 3 | 0 | 0 | 1 |
| EES 202 | Foundation of Earth Sciences: Part 2 (Introduction to Rock Deformation and Plate Tectonics) | 3 | 0 | 1 | 6 | 3 |
| HSS 207 | Macroeconomics | 1 | 0 | 0 | 2 | 1 |
| MTH 202 | Multivariable Calculus | 3 | 0 | 1 | 6 | 3 |
| PHY 202 | Quantum Physics | 3 | 0 | 1 | 6 | 3 |
| PHY 204 | General PHY Lab III | 0 | 3 | 0 | 0 | 1 |
| CDS 202 | Information for Science and Technology | 2 | 0 | 0 | 4 | 2 |
| Total | | 18 | 09 | 05 | 36 | 21 |

BS-MS SYLLABUS, COMPUTER AND DATA SCIENCE

CDS 101: Introduction to Computers (3)

Course Contents:

Module-1:

Fundamentals of Computing: Introduction to Computers: Computer Systems, Computing Environments, Computer Languages, word processing, spreadsheets, presentation graphics, and databases. Creating and Running Programs, Software Development, Flow charts. Number Systems: Binary, Octal, Decimal, and Hexadecimal.

Module-2:

Problem Solving Techniques Core Elements of a Program, Problem Solving, Machine Interpretation of a Program, Algorithm, Recursion, Debugging, Efficiency and Order of Growth, Memory and Search Methods, Introduction to Simulation and Random Walks, Statistical Thinking, Curve Fitting, Optimization Problems and Algorithms, Using Graphs to Model Problems, Using Graphs to Model Problems, Dynamic Programming, Queuing Network Models.

Module-3:

Control Flow with discussion on structured and unstructured programming: Variable Names, Data Type and Sizes (Little Endian Big Endian), Constants, Declarations, Arithmetic Operators, Relational Operators, Logical Operators, Type Conversion, Increment Decrement Operators, Bitwise Operators, Assignment Operators and Expressions, Precedence and Order of Evaluation, proper variable naming and Hungarian Notation. Statements and Blocks, If-Else-If, Switch, Loops – while, do, for, break and continue, goto labels, structured and un- structured programming

Module-4:

Functions and Program Structure with discussion on standard library: Basics of functions, parameter passing and returning type, C main return as integer, External, Auto, Local, Static, Register Variables, Scope Rules, Block structure, Initialization, Recursion, Pre-processor, Standard Library Functions and return types.

Module-5:

Pointers and Arrays: Pointers and address, Pointers and Function Arguments, Pointers and Arrays, Address Arithmetic, character Pointers and Functions, Pointer Arrays, Pointer to Pointer, Multi-dimensional array and Row/column major formats, Initialisation of Pointer Arrays, Command line arguments, Pointer to functions, complicated declarations and how they are evaluated; Dynamic memory allocation.

Module-6:

Structures: Basic Structures, Structures and Functions, Array of structures, Pointer of structures, Self-referral structures, Table lookup, typedef, unions, Bit-fields

Module-7:Input and Output: Standard I/O, Formatted Output – printf, Formated Input – scanf, Variable length argument list, file access including FILE structure, fopen, stdin, sdtout and stderr, Debugging, Macro, User Defined Header, User Defined Library Function, makefile utility

CDS 201: Introduction to MATLAB for scientists (2)

Course Contents:

Objectives: In this hybrid theory & practical course, the students will learn to formulate a computational problem and use the MATLAB software and some of its toolboxes to create and troubleshoot basic sequential scripts and functions to achieve computational objectives. At the end of the course, the students will be able to use MATLAB as a simulation tool to perform basic array and mathematical operations, import datasets to effectively analyze and visualize data through interpolation, plotting, and curve fitting, apply numerical techniques in computer simulations, generate plots and export them to create scientific publishable figures. The students will also be trained on good programming practices. The final goal of the course is to motivate students to explore more of MATLAB on their own for their individual research projects.

Prerequisite:

There is no formal prerequisite for this course. Basic mathematical courses would be a plus.

Lectures: 15 hours of lecturing + 30 hours of lab

Syllabus:

Introduction: MATLAB environment, editor, scripts, live scripting, and functions; Basics - data types, creating and editing variables, entering commands, use of operators and expressions, generating, editing, running, and saving simple script files.

Good programming practices: planning approach, use of comment lines and indenting, code simplification and debugging.

Matrix laboratory: Creating, manipulating, and accessing data in vectors and arrays (matrices), key inbuilt MATLAB functions & input/output statements. Introduction to user-defined functions.

Graphics with MATLAB: File types, file management (importing and exporting data), plotting (2D, 3D, subplot), plot properties, axis properties, exporting figures.

Programming with MATLAB: Conditional statements (logical operators, if, else elseif, switch), Introduction to loops (for and while), Nested loops.

Mathematical computing: Algebraic expressions and linear equations, symbolic calculus, differential equations, integration, and differentiation. Numerical techniques and Fourier transforms.

References:

- MATLAB Help Center
- MATLAB Academy
- MATLAB Cody

CDS 202 : Information for Science and Technology (2)

Learning Objectives:

Aim of this course is to sensitise students about the role of data & information in the science and technology, as well as in day-to-day life. Further, it aims at creating awareness about what technologies, tools, standards and processes are available for students as they pursue their career in different specialised domains of S&T.

Course Contents:

- Data definition, history & origin of data science, data paradigm in science & technology, data intensive understanding, data intensive analysis, data intensive discovery, data intensive forecasting
- Data science principles and theory, role of data & metadata in S&T, data life cycle in S&T processes, data & metadata standards across disciplines, data and metadata organisational tools, data as economy, information banks for S&T studies, data interoperability standards, tools, practices and processes.
- Overview of the Data discovery challenges and infrastructures, data discovery services - Bibliographic & abstracting services, data repositories - planning to implementation, specialised global and national data repositories.
- Introductions to the S&T Data publishing - data publishing framework in S&T, metadata to data papers, data, fostering data papers in specialised areas of S&T
- Introductions to the Data management, analysis & modelling technologies & tools: databasing, BIG data, analytical tools, neural networks, artificial intelligence, visualisation, virtual reality, etc.
- Introductions to the informatics landscape in different areas of S&T, especially, life, earth, environment & sustainability science (Biodiversity Informatics, Ecosystem Informatics, Bioinformatics, Environmental Informatics, GeoInformatics, Chemoinformatics, Climate Change analysis & modelling, health informatics, physicoinformatics, etc.).
- Collaborations for data & information in S&T, CODATA, TDWG, Open research data policy and practices, Global efforts to rescue of data at threat, OPEN data access, FAIR Research Data Management, Research Data Alliance, Persistent Identifiers, etc.

Suggested Readings :

1. Data Science: Concepts & Practice by Vijay Kotu&Bala Deshpande (2018), Morgan & Kaufmann Publishers.
2. Build a Career in Data Science by Emily Robinson& Jacqueline Nolis (2020), Manning Publications Co.
3. <https://www.oreilly.com/library/view/getting-data-right/9781491935361/ch04.html>

CDS 301: Programming and Data Structures Lab I
The page will be updated soon!!

CDS 302: Programming and Data Structures Lab II

The page will be updated soon!!

CDS 303: Theory of Computation

Course Contents:

Module-1:

Introduction, definition, concept of sequential circuits, state table & state assignments, concept of synchronous, asynchronous and linear sequential machines. Basic definition, mathematical representation, Moore versus Mealy m/c, capability & limitations of FSM, state equivalence & minimization, machine equivalence, incompletely specified machines, merger graph & compatibility graph, merger table, Finite memory, definite, information loss less & inverse machines: testing table & testing graph.

Module-2:

FSA, NFSA, NFSA with ϵ moves, Regular expressions, Equivalence of regular expression and FSA, Equivalence of type 3 grammars and FSA, Pumping lemma, Closure and decidability results, Myhill-Nerode theorem, Minimization.

Module-3:

Grammars, Languages generated, Chomsky Hierarchy, CFG, Ambiguity, Reduced grammars, Normal forms. Pumping lemma & its application, closure properties. Context Free Grammars: Introduction, definition, derivation trees, simplification, CNF & GNF.

Module-4:

Pushdown Automata, Acceptance by final state and empty stack, Equivalence to CFG, Deterministic PDA.

Module-5:

Turing Machines - Construction, Techniques of TM construction, TM as acceptor and i/o device, Problems. Generalized and restricted versions.

Module-6:

Halting problems - Universal TM-recursive and recursively enumerable sets - Decidability - Rice's Theorem, PCP.

Text Books:

- J. E. Hopcroft, J. D. Ullman and R. Motwani: Introduction to Automata Theory, Languages and Computation, Addison- Wesley, California, 2001.
- M. Sipser: Introduction to the Theory of Computation, PWS Pub. Co., New York, 1999.
- Peter Linz, An Introduction to Formal Languages and Automata.
- M. R. Garey and D. S. Johnson: Computers and Intractability: A Guide to The Theory of NP-Completeness, Freeman, New York, 1979.
- M. D. Davis, R. Sigal and E. J. Weyuker: Complexity, Computability and Languages, Academic Press, New York, 1994.
- N. J. Cutland: Computability: An Introduction to Recursive Function Theory, Cambridge University Press, London, 1980.

CDS 304:Discrete Structures and Computation

Course Contents:

Module-1:

Review of Sets, Operations, Principles of Inclusion and Exclusion. Functions, relations, Equivalence relations. Countable and uncountable sets. Review of Pigeonhole principle.

Module-2:

Introduction to Propositional Logic, Equivalence and Implications. Truth tables, De Morgan's Law, Quantifiers, Inference and Proofs. Introduction to First Order Logic, Syntax and Semantics, Soundness and Completeness.

Module-3:

Mathematical Induction, Recursions, First order linear recurrence, Geometric series, Recursion trees and growth rates of solutions to recurrences, Master Theorem. Generating Functions.

Module-4:

Introduction to counting, sum and product principles, counting subsets. Binomial coefficients and Pascal's triangles. Polya's theory of counting (optional).

Module-5:

Arithmetic Algorithms: Computing GCD, primality testing, RSA.

Module-6:

Graph Theory: Graphs, representations, connectivity, cycles, trees, Spanning tree of a graph, Algorithms to find minimum spanning trees. Eulerian Cycle and Hamiltonian paths, independence number and clique number, chromatic number, Dominating Sets, and Covering Sets. Planar Graphs. Directed Graphs and tournaments.

Module-7:

Probabilistic tools, Tail Bounds and Applications.

Module-8:

Linear Algebraic tools in Combinatorics.

CDS 305: Introduction to Data Science I

The page will be updated soon!!

CDS 306: Introduction to Data Science II

The page will be updated soon!!

CDS401-DESIGN AND ANALYSIS OF ALGORITHM

Introduction:

Example: Air Travel / Xerox shop / Document similarity; Introduction and motivation; Input size, worst case, average case; Quantifying efficiency: $O()$, $\Omega()$, $\Theta()$; Examples: Analysis of iterative and recursive algorithms

Searching & Sorting:

Arrays and lists; Searching in an array; Bubble Sort; Selection Sort; Insertion sort; Merge sort; Merge sort – analysis; Quicksort; Quicksort – analysis; Sorting - Concluding remarks

Graphs:

Introduction to graphs; Representing graphs; Breadth-first search (BFS); Depth-first search (DFS); Applications of BFS and DFS; Directed acyclic graphs: topological sort; Directed acyclic graphs: longest paths

Weighted Graph:

Single source shortest paths: Dijkstra's algorithm; Dijkstra's algorithm: analysis; Negative edge weights: Bellman-Ford algorithm; All pairs shortest paths; Minimum Cost Spanning Trees; Prim's Algorithm; Kruskal's algorithm

Divide & Conquer:

Union-Find using arrays; Union-Find using pointers; Priority queues; Heaps; Heaps: Updating values, sorting; Counting inversions; Closest pair of points

Greedy Algorithm:

Binary Search Trees; Balanced search trees; Interval scheduling; Scheduling with deadlines: minimizing lateness; Huffman codes

Dynamic Programming:

Introduction to dynamic programming; Memoization; Grid Paths; Common subwords and subsequences; Edit distance; Matrix multiplication

Network Flow & Intractability:

Linear Programming; LP modelling: Production Planning; LP modelling: Bandwidth allocation; Network Flows; Reductions; Checking Algorithms; P and NP

Suggested Resources:

1. Textbook:
 - "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.
 - "Algorithm Design Manual" by Steven S. Skiena.
2. Lecture Notes and Slides:
 - MIT OpenCourseWare: Introduction to Algorithms (Course 6.006)
 - University: Design and Analysis of Algorithms (CS161)

3. Online Courses and Video Lectures:
 - Coursera: "Algorithms, Part I" and "Algorithms, Part II" by Robert Sedgewick and Kevin Wayne.
4. Additional Readings and Resources:
 - "Algorithms, Part I" and "Algorithms, Part II" by Robert Sedgewick and Kevin Wayne (companion books to the Coursera courses).
 - "Algorithms Illuminated" series by Tim Roughgarden.