# DEPARTMENT OF COMPUTER SCIENCE
# VIDYASAGAR UNIVERSITY
## PASCHIM MEDINIPUR– 721102

**M.Sc. SEMESTER – I**

| Course Code | Paper | Teaching Scheme (Per Week) | | | Examination Scheme (Marks) | | | Total (Marks) |
|---|---|---|---|---|---|---|---|---|
| | | Th. | Tu. | Pr. | Int. | Att. | Ext. | |
| CS/MSc/1101 | Discrete Structure<br>M1: Set Theory<br>M2: Graph Theory | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1102 | Advance Computer Architecture & Microprocessor<br>M1: Parallel Architecture<br>M2: Microprocessor (8086 to Pentium) | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1103 | Distributed Operating System | 2 | 1 | 0 | 10 | 0 | 40 | 50 |
| CS/MSc/1104 | Graphics and Image Processing<br>M1: Computer Graphics<br>M2: Image Processing | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1111 | Computer Graphics Lab. & Microprocessor Lab | 0 | 0 | 3+3 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1112 | Operating System Lab | 0 | 0 | 6 | 10 | 0 | 40 | 50 |

## M.Sc. SEMESTER – II

| Course Code | Paper | Teaching Scheme (Per Week) | | | Examination Scheme (Marks) | | | Total (Marks) |
|---|---|---|---|---|---|---|---|---|
| | | Th. | Tu. | Pr. | Int. | Att. | Ext. | |
| CS/MSc/1201 | Database System<br>M1: Advanced Distributed DBMS<br>M2: PHP, My SQL & ASP.NET, ADO.NET | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1202 | Theory of Computation and Compiler<br>M1: Finite Automata<br>M2: Compiler Design | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1203 | Design & Analysis of Algorithm | 2 | 1 | 0 | 10 | 0 | 40 | 50 |
| CS/MSc/1204 | System Analysis, Design and Software Engineering<br>M1: SAD<br>M2: Software Engineering | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1211 | DBMS Lab. & Algorithm Lab | 0 | 0 | 3+3 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/1212 | Compiler Lab & Soft. Eng. Lab | 0 | 0 | 3+3 | 10 | 0 | 20+20 | 25+25 |

## M.Sc. SEMESTER – III

| Course Code | Paper | Teaching Scheme (Per Week) | | | Examination Scheme (Marks) | | | Total (Marks) |
|---|---|---|---|---|---|---|---|---|
| | | Th. | Tu. | Pr. | Int. | Att. | Ext. | |
| CS/MSc/2101 | Network and Internet M1: Computer Network M2: Internet Technology | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/2102 | Advance Programming M1: Object Oriented Programming M2: Advanced JAVA | 2 | 1 | 0 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/2103 | Artificial Intelligence & Neural Network | 2 | 1 | 0 | 10 | 0 | 40 | 50 |
| CS/MSc/2104 | Elective - I | 2 | 1 | 0 | 10 | 0 | 40 | 50 |
| CS/MSc/2111 | Network Lab. & Web-Page Designing Lab | 0 | 0 | 3+3 | 10 | 0 | 20+20 | 25+25 |
| CS/MSc/2112 | AI Lab & JAVA Lab | 0 | 0 | 3+3 | 10 | 0 | 20+20 | 25+25 |

## List of Electives:

1. Pattern Recognition.
2. Bio-informatics.
3. VLSI Design.
4. Embedded System.
5. Mobile Computing.
6. Soft Computing.
7. Fuzzy set theory

## M.Sc. SEMESTER – IV

| Course Code | Paper | Teaching Scheme (Per Week) | | | Examination Scheme (Marks) | | | Total (Marks) |
|---|---|---|---|---|---|---|---|---|
| | | Th. | Tu. | Pr. | Int. | Att. | Ext. | |
| CS/MSc/2211 | External Project & Industrial Training | 0 | 0 | 0 | 0 | 0 | 200 | 200 |
| CS/MSc/2212 | Seminar | 0 | 0 | 0 | 0 | 0 | 50 | 50 |
| CS/MSc/2213 | Grand VIVA | 0 | 0 | 0 | 0 | 0 | 50 | 50 |

## Detailed Syllabus Outline

## M.Sc. SEMESTER – I

### CS/MSc/1101 : DISCRETE STRUCTURE

### M1: SET THEORY

***Set Theory:*** Introduction, Definition & Concepts, Representation of Sets, Finite Sets, Infinite Sets (Definition), Set Operations--- Union, Intersection, Addition, Difference, Symmetric Difference, De Morgan's Law, Subsets, Power Sets, Partition Sets, Mathematical Inductions, Computing Principles, Permutations, Combinations.

### M2: GRAPH THEORY

Graphs, Digraphs, Isomorphism, Walks, Paths, Circuits, Shortest Path Problem, Dijkstra's Algorithm, Trees, Properties of Trees, Cotrees and Fundamental Circuits, Shortest Spanning Trees--- Kruskal's Algorithm, Prim's Algorithm, DFS, BFS, Cut Sets, Cut Vertices, Planar and Dual Graphs, Metric Representation of Graphs, Networks, Flow Augmenting Path, Ford-Fulkerson Algorithm for Maximum Flow.

**Books:**


1. Liu C. L., "Introduction to combinatorial mathematics", McGraw Hill, 1968.
2. Mott J. L., Kandel A. and Baker T. P., "Discrete mathematics for Computer Scientists and Mathematicians", PH, 1986.
3. Rosen—Discrete Mathematics, 2/e,TMH
4. S.K. Mapa—Higher Algebra (Abstract & Modern)

5. Robert J. McElice , Robert B. Ash & Carol Ash, “Introduction to discrete Mathematics”, Tata McGraw Hill
6. Deo N., “Graph Theory with Applications to Engineering and Computer Science”, PHI, 1980
7. Tremblay and Manohar, “Discrete mathematical structures with applications to computer science”, McGraw Hill, 1975
8. Kolamn, Busby and Ross, “Discrete mathematical structures”, 3/ed, PHI, 1996.
9. Fraleigh J. B., “A first course in abstract algebra Narosa”, 1990
10. Smullyan R. M., “First Order Logic Springer Verlag”, 1968

**Reference:**

1. Lipschutz—2000 Solved Problems in Discrete Mathematics, TMH
2. Balakrishnan—Graph Theory (Schaum),MH
3. Hararay—Graph Theory.

## CS/MSc/1102: ADVANCE COMPUTER ARCHITECTURE & MICROPROCESSOR

### M1: PARALLEL ARCHITECTURE

Review of Pipelining, Examples of some pipeline in modern processors, pipeline hazards, data hazards, control hazards. Techniques to handle hazards, performance improvement with pipelines and effect of hazards on the performance.
Vector processors- Use and effectiveness, memory to memory vector architectures, vector register architecture, vector length and stride issues, compiler effectiveness in vector processors.
SISD, MISD, MIMD, Single instruction multiple data stream (SIMD) architectures. Array processors, comparison with vector processors, example of array processors such as MMX Technology.
RISC architectures, addressing modes, instructions formats, effect of simplification on the performance, example processors such as MIPS, PA-RISC, SPARC, Power PC, etc.
MIMID Multiprocessors, Centralized shared architectures, distributed shared memory architectures, synchronization and memory consistency models, message passing architectures, comelier issues. Data flow architectures, Interconnection networks.

***Text Books:***

1. Hwang, K. “Advanced Computer architecture with parallel programming”, McGraw Hill, 1993
2. Carter—Computer Architecture ( Schaum Series),TMH
3. Patterson D.A. and Hennessy , J.L. “Computer architecture a quantitative approach”, 2nd ed., Morgan Kaufman, 1996

4. Hwang & Briggs—Computer Architecture & Parallel Processing, TMH
5. Stone, H.S., "Advanced Computerat", Addison Wesley, 1989
6. Siegel, H.J., "Interconnection Network for Large Scale parallel Processing", 2nd Ed., McGraw Hill, 1990

**Reference:**

Quinn—Parallel Processing

**M2: MICROPROCESSOR (8086 to PENTIUM)**

Introduction to 8086 CPU architecture-register organization, addressing modes and their features. Software instruction set and Assembly Language Programming. Pin description and features. Instruction cycle, machine cycle, Timing diagram.
Hardware Interfacing: Interfacing memory, peripheral chips (IO mapped IO & Memory mapped IO).Interrupts and DMA.
Peripherals: 8279, 8255, 8251, 8253, 8237, 8259, A/D and D/A converters and interfacing of the same.
Typical applications of a microprocessor.
Brief overview of some other microprocessors (eg. Pentium).

**References:**

1. Microprocessor architecture, programming and applications with 8085/8085A, Wiley eastern Ltd, 1989 by Ramesh S. Gaonkar.
2. Intel Corp: The 8085 / 8085A. Microprocessor Book – Intel marketing communication, Wiley inter science publications, 1980.
3. An introduction to micro computers Vol. 2 – some real Microprocessor – Galgotia Book Source, New Delhi by Adam Osborne and J. Kane
4. Advanced Microprocessors by Ray and Bhurchandi - TMH
5. Intel Corp. Micro Controller Handbook – Intel Publications, 1994.
6. Microprocessors and Interfacing by Douglas V. Hall, McGraw Hill International Ed. 1992
7. Assembly Language Programming the IBM PC by Alan R. Miller, Subex Inc, 1987
8. The Intel Microprocessors: 8086/8088, 80186, 80286, 80386 & 80486, Bary B. Brey, Prentice Hall, India 1996.

**CS/MSc/1103 : DISTRIBUTED OPERATING SYSTEM**

Principles of distributed systems in general and distributed operating systems in particular. Covered topics include processes and threads, concurrent programming, distributed inter-process communication, distributed process scheduling, shared virtual memory, distributed file systems, security in distributed systems, distributed middleware and applications such as the web and peer-to-peer systems. Some coverage of operating system principles for multiprocessors will also be included. A brief overview of advanced

topics such as multimedia operating systems, real-time operating systems and mobile computing will be provided, time permitting.

***Text Books / References* :**


1. Distributed Systems: Principles and Paradigms
   Andrew Tannenbaum and Maarten van Steen, Prentice Hall, 2001

2. Distributed Systems, 2nd edition
   Sape Mullender, Addison Wesley, 1993


**CS/MSc/1104 : COMPUTER GRAPHICS AND IMAGE PROCESSING**

**M1: COMPUTER GRAPHICS**

**Introduction to Computer Graphics and Graphics Systems:** Overview of Computer Graphics, RGB color model, Plotters, Printers, Digitizers, Light pens, etc.; Active and Passive graphics devices; Computer Graphics Software.
**Scan Conversion:** Points and Lines, Line Drawing Algorihms; DDA Algorithm, Bresenham's Line Algorithm, Circle Generation Algorithm; Ellipse Generating Algorithm; Scan Line Polygon, Fill Algorithm, Boundary Fill Algorithm, Flood Fill Algorithm.
**2D Transformation and Viewing:** Basic Transformations--- Translation, Rotation, Scaling; Matrix Representations and homogeneous coordinates, Transformations between coordinate systems; Reflection, Shear; Transformations of points, lines, parallel lines, intersecting lines. Viewing pipeline, Window to viewport coordinate transformation; Clipping Operations--- Point clipping, Line clipping, Clipping circles, polygons and ellipse.
**3D Transformation and Viewing:** 3D Transformations--- Translation, Rotation, Scaling & other transformations. Rotation about an arbitrary axis in space, Reflection through an arbitrary plane; General, Parallel Projection transformation; Clipping,
**Curves:** Curve representation, Surfaces, Designs, Bezier curves, End conditions for periodic B-spline curves,
**Hidden Surfaces:** Depth comparison, Z-buffer Algorithm, Back face detection,

**M2: IMAGE PROCESSING**

**IMAGE PROCESSING:**
**Introduction:** Background, Digital Image Representation, Fundamental steps in Image Processing, Elements of Digital Image Processing - Image Acquisition, Storage, Processing, Communication, Display.

**Digital Image Formation :**
A Simple Image Model, Geometric Model- Basic Transformation (Translation, Scaling, Rotation), Perspective Projection, Sampling & Quantization - Uniform & Non uniform.

**Mathematical Preliminaries :**
Neighbour of pixels, Connectivity, Relations, Equivalence & Transitive Closure; Distance Measures, Arithmetic/Logic Operations, Fourier Transformation, Properties of The Two Dimensional Fourier Transform, Discrete Fourier Transform, Discrete Cosine & Sine Transform.

**Image Enhancement :**
Spatial Domain Method, Frequency Domain Method, Contrast Enhancement -Linear & Nonlinear Stretching, Histogram Processing; Smoothing - Image Averaging, Mean Filter, Low-pass Filtering; Image Sharpening. High-pass Filtering, High-boost Filtering, Derivative Filtering, Homomorphic Filtering; Enhancement in the frequency domain - Low pass filtering, High pass filtering.

**Image Segmentation :**
Point Detection, Line Detection, Edge detection, Combined detection, Edge Linking & Boundary Detection - Local Processing, Global Processing via The Hough Transform; Thresholding - Foundation, Simple Global Thresholding, Optimal Thresholding; Region Oriented Segmentation - Basic Formulation, Region Growing by Pixel Aggregation, Region Splitting & Merging.

**Text Books:**


1. Hearn, Baker – " Computer Graphics ( C version 2nd Ed.)" – Pearson education
2. Z. Xiang, R. Plastock – " Schaum's outlines Computer Graphics (2nd Ed.)" – TMH
3. D. F. Rogers, J. A. Adams – " Mathematical Elements for Computer Graphics (2nd Ed.)" – TMH
4. Mukherjee, Fundamentals of Computer graphics & Multimedia, PHI
5. Digital Image Processing, Gonzalves,Pearson
6. Digital Image Processing, Jahne, Springer India
7. 3.Digital Image Processing & Analysis,Chanda & Majumder,PHI
8. 4.Fundamentals of Digital Image Processing, Jain, PHI
9. 5.Image Processing, Analysis & Machine Vision, Sonka, VIKAS
10. Getting Started with GIS- Clarke Keith. C; PE.
11. Concepts & Techniques of GIS - Lo C.P, Albert, Yeung K.W- PHI.


***Reference Books:***


1. Foley, Vandam, Feiner, Hughes – "Computer Graphics principles (2nd Ed.) – Pearson Education.
2. W. M. Newman, R. F. Sproull – "Principles of Interactive computer Graphics" – TMH.
3. Elsom Cook – "Principles of Interactive Multimedia" – McGraw Hill

## CS/MSc/1111 M1: MICROPROCESSOR LAB

**Sl. No.** Name of the Experiments

**1.** Familiarization with 8086 register level architecture and trainer kit components, including the memory map. Familiarization with the process of storing and viewing the contents of memory as well as registers.

**2.**
a) Study of prewritten programs on trainer kit using the basic instruction set ( data transfer, Load/Store, Arithmetic, Logical)
b) Assignments based on above.

**3.**
a) Familiarization with 8086 Assembler
c) Study of prewritten programs using basic instruction set (data transfer, Load/Store, Arithmetic, Logical) on the simulator.
b) Assignments based on above

**4.** **Programming using kit/Assembler for**

i) table look up
ii) Copying a block of memory
iii) Shifting a block of memory
iv) Packing and unpacking of BCD numbers
v) Addition of BCD numbers
vi) Binary to ASCII conversion
vii) String Matching
viii) Multiplication using Booth's Algorithm

**5.** Program using subroutine calls and IN/OUT instructions using 8255 PPI on the trainer kit eg, subroutine for delay, reading switch state & glowing LEDs accordingly, finding out the frequency of a pulse train etc

**6.** Interfacing any 8-bit Latch (eg, 74LS373) with trainer kit as a peripheral mapped output port with absolute address decoding

**7.** **Interfacing with I/O modules:**

a) ADC
b) Speed control of mini DC motor using DAC
c) Keyboard
d) Multi-digit Display with multiplexing
e) Stepper motor

**8.** Writing programs for 'Wait Loop (busy waiting)' and ISR for vectored interrupts (eg, counting number of pulses within specified time period)

## M2: GRAPHICS LAB

- Point plotting, line & regular figure algorithms
- Raster scan line & circle drawing algorithms

- Clipping & Windowing algorithms for points, lines & polygons
- 2-D / 3-D transformations
- Simple fractals representation
- Filling algorithms
- Web document creation using Dreamweaver.
- Creating Animation using Flash.
- DirectX Supported Graphics

- 2D Graphics using DirectX

- 3C Graphics using DirectX

**CS/MSc/1112 : OPERATING SYSTEM LAB**

1. **Shell programming:** creating a script, making a script executable, shell syntax (variables, conditions, control structures, functions, commands).
2. **Process:** starting new process, replacing a process image, duplicating a process image, waiting for a process, zombie process.
3. **Signal:** signal handling, sending signals, signal interface, signal sets.
4. **Semaphore:** programming with semaphores (use functions semctl, semget, semop, set_semvalue, del_semvalue, semaphore_p, semaphore_v).
5. **POSIX Threads :** programming with pthread functions(viz. pthread_create, pthread_join, pthread_exit, pthread_attr_init, pthread_cancel)
6. **Inter-process communication :** pipes(use functions pipe, popen, pclose), named pipes(FIFOs, accessing FIFO)

## M.Sc. SEMESTER – II

**CS/MSc/1201 : DATABASE SYSTEM**

**M1: ADVANCED DISTRIBUTED DBMS**

Distributed DBMS features and needs. Reference architecture. Levels of distribution transparency, replication. Distributed database design - fragmentation, allocation criteria. Storage mechanisms. Translation of global queries. / Global query optimisation. Query execution and access plan. Concurrency control - 2 phases locks. Distributed deadlocks. Time based and quorum based protocols. Comparison. Reliability- non-blocking commitment protocols.

Partitioned networks. Checkpoints and cold starts. Management of distributed transactions- 2 phase unit protocols. Architectural aspects. Node and link failure recoveries.

Distributed data dictionary management. Distributed database administration. Heterogeneous databases-federated database, reference architecture, loosely and tightly coupled.
Alternative architecture. Development tasks, Operation- global task management. Client server databases-SQL server, open database connectivity. Constructing an application.
Introduction to advanced databases-homogeneous and heterogeneous databases- temporal databases-spatial databases-data mining and warehousing- deductive databases-semistructured and web databases-mobile databases. OODBMS – introduction to object orientation- persistence of objects- object relational databases.

**Books:**

1. Database System Concepts, Silberschatz Korth, Sudarshan, MH
2. Distributed Database, Tannenbaum, Pearson
3. Principles of Distributed Database Systems, M. Tamerozsu Patrick Valduriez, Pearson
3. Database Management Systems,Ramakrishnan, MH

**M2: PHP, My SQL & ASP.NET, ADO.NET**

Covers the PHP scripting language and the MySQL database to create dynamic Web pages. Topics covered include: PHP scripting fundamentals; creating, accessing, and manipulating data with the MySQL database within a PHP program; creating HTML forms; and writing secure PHP programs.
Introduction to the Web and .NET Platform, C# as a Language for the Web, Introduction to ASP.NET, Web Forms Architecture, Server Side Controls, Databases and Data access on the Web, Accessing Data with ADO.NET

**CS/MSc/1202 : THEORY OF COMPUTATION AND COMPILER**

**M1: THEORY OF COMPUTATION**

Finite State Machines: Definition, concept of sequential circuits, state table & state assignments
Finite State Models : Basic definition, mathematical representation, Moore versus Mealy m/c,
Finite Automation : Definition, recognition of a language by an automata - idea of grammar, DFA, NFA, equivalence of DFA and NFA, NFA with e-moves, regular sets & Regular expressions : equivalence with finite automata, NFA from regular expressions, regular expressions from DFA, equivalence of Moore & Mealy machines, applications of finite automata.
Minimization of finite automata
Context Free Grammars: Introduction, definition, derivation trees, simplification, CNF & GNF.

Pushdown Automata : Definition, moves, Instantaneous Descriptions, language recognized by PDA, deterministic PDA, acceptance by final state & empty stack, equivalence of PDA and CFL.

***Text books* :**


1. Hopcroft JE. and Ullman JD., "Introduction to Automata Theory, Languages & Computation", Narosa.
2. K.L.P Mishra & N. Chandrasekharan – "Theory of Computer Science", PHI
3. Ash & Ash – "Discrete Mathematics",TMH
4. Martin—Introduction
5. Lewis H. R. and Papadimitrou C. H., "Elements of the theory of Computation", P.H.I.
6. Kain, "Theory of Automata & Formal Language", McGraw Hill.


***References* :**


1. Kohavi ZVI, "Switching & Finite Automata", 2nd Edn., Tata McGraw Hill.
2. Linz Peter, "An Introduction to Formal Languages and Automata", Narosa
3. "Introduction to Formal Languages", Tata McGraw Hill, 1983.


## M2: COMPILER DESIGN

**Introduction to Compiling :**Compilers, Analysis of the source program, The phases of the compiler, Cousins of the compiler.
**Lexical Analysis :**The role of the lexical analyzer**,** Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of a tokens, Finite automata, From a regular expression to an NFA, From a regular expression to NFA, From a regular expression to DFA, Design of a lexical analyzer generator (Lex).
**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)
**Syntax directed translation:**Syntax director definitions, Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes.
**Type checking :**Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions
**Code optimization :**Introduction, Basic blocks & flow graphs, Transformation of basic blocks, Dag representation of basic blocks, The principle sources of optimization, Loops in flow graph, Peephole optimization.
**Code generations :**Issues in the design of code generator, a simple code generator, Register allocation & assignment.

**Text books:**


1. Aho, Sethi, Ullman - "Compiler Principles, Techniques and Tools" - Pearson Education.

2. Holub - “Compiler Design in C” - PHI.

**CS/MSc/1203** : **DESIGN AND ANALYSIS OF ALGORITHM**

***Asymptotic notation:*** Time and Space complexity, Big-O, Omega, Theta etc., Finding time complexity of well-known algorithms like--- Heap sort, Quick sort, Search algorithm etc.
***Algorithm Design Techniques:*** Recursion (definition), Use, Limitations, Examples: Tower of Honoi problem, Tail recursion.
***Divide and Conquer:*** Basic concepts, Examples: Merge sort, Quick sort, Binary search.
***Dynamic Programming:*** Basic method, Use, Examples: Matrix-chain multiplication, All pair shortest paths, Single source shortest path, Traveling salesman problem.
***Branch and Bound:*** Basic method, Use, Examples: 15-puzzle problem.
***Backtracking:*** Basic method, Use, Examples: Eight queens problem, graph Coloring problem, Hamiltonian problem.
***Greedy Method:*** Basic method, Use, Examples: Knapsack problem, Job Sequencing with deadlines, Minimum Spanning tree (Prim’s and Kruskal’s algorithms).
***Lower Bound Theory:*** Bounds on sorting and searching techniques using partial and total orders.
***Matrix Manipulation algorithms:*** Different types of algorithms and solution to Simultaneous equations, DFT and FFT algorithm; Integer multiplication schemes.
***Notion of NP-completeness:*** P class, NP-hard class, Circuit Satisfiability problem, Clique Decision Problem.
***Approximation Algorithms:*** Necessity of approximation scheme, Performance guarantee, Polynomial time approximation schemes: 0/1 Knapsack problem.

***Text Books:***

1. A.Aho, J.Hopcroft and J.Ullman “The Design and Analysis of algorithms”
2. D.E.Knuth “The Art of Computer Programming”, Vol. I & Vol.2
3. Horowitz Ellis, Sahani Sartaz, R. Sanguthevar " Fundamentals of Computer Algorithms".
4. Goodman: Introduction to Design and Analysis Of Algorithms TMH

***Reference:***

1. K.Mehlhorn , “Data Structures and algorithms- Vol. I & Vol. 2 “
2. S.Baase “Computer algorithms”
3. E.Horowitz and Shani “Fundamentals of Computer algorithms”
4. E.M.Reingold, J.Nievergelt and N.Deo- “Combinational algorithms- Theory and Practice”, Prentice Hall , 1997
5. A.Borodin and I.Munro, “The computational complexity of Algebraic and Numeric problems”

## CS/MSc/1204 : SYSTEM ANALYSIS, DESIGN AND SOFTWARE ENGINEERING

### M1: SYSTEM ANALYSIS AND DESIGN

***Overview of System Analysis & Design:*** System Development Life Cycle, Requirements Determination, Logical Design, Physical Design, Program Design, Feasibility analysis, SRS, Prototyping, Risk analysis.
***Information Requirement Analysis:*** Process modeling with physical and logical data flow diagrams, Data modeling with Entity-Relationship Diagrams, Feasibility studies, Cost benefit analysis.
***System Design:*** Process descriptions, Input/Output controls, Object modeling, Database design, User interface design, Documentation.
Introduction to Project management, Project Scheduling, Measurement of quality and productivity, ISO and capability maturity models, Strategic planning, System audit.
***Quality assurance:*** Reviews, Walkthroughs, Inspection.

**Books:**

1. Analysis & Design of Information Systems,Senn,MH.
2. Information Systems:Analysis & Design, Ram Bansal 'Vigyacharya',New Age International
3. Analysis, Design of Information System,Rajaraman, PHI
4. System Analysis & Design,Parthasarathi,EPH
5. System Analysis, Design & MIS, EXCEL BOOKS
6. Analysis, Design & implementation of Information Systems, Sharma, VIKAS
7. System Analysis & Design Hand Book, V.K. Jain, Wiley Dreamtech

### M2: SOFTWARE ENGINEERING

System Development Life Cycle, Waterfall Model , Spiral Model, Feasibility Analysis, Technical Feasibility, Cost- Benefit Analysis, COCOMO model.
System Requirement Specification – DFD, Data Dictionary, ER diagram, Process Organization & Interactions.
System Design – Problem Partitioning, Top-Down And Bottop-Up design ;Decision tree, decision table and structured English; Functional vs. Object- Oriented approach.
Coding & Documentation - Structured Programming, OO Programming, Information Hiding, Reuse, System Documentation.
Testing – Levels of Testing, Integration Testing, Test case Specification, Reliability Assessment . , Validation & Verification Metrics, Monitoring & Control.
Software Project Management – Project Scheduling , Staffing, Software Configuration Management, Quality Assurance, Project Monitoring.
CASE TOOLS : Concepts, use and application.

**Books:**

**Text:**

1. R. G. Pressman – Software Engineering, TMH
2.Behforooz, Software Engineering Fundamentals,OUP
3. Ghezzi, Software Engineering, PHI
4.Pankaj Jalote – An Integrated Approach to Software Engineering, NAROSA.
5.Object Oriented & Classical Software Engineering(Fifth Edition), SCHACH,TMH
6.Vans Vlet, Software Engineering, SPD
7.Uma, Essentials of Software Engineering, Jaico
8.Sommerville, Ian – Software Engineering, Pearson Education
9.Benmenachen, Software Quality, Vikas

**Reference:**

1. IEEE Standards on Software Engineering.
2. Kane, Software Defect Prevention, SPD

**CS/MSc/1211 :**

**DBMS LAB**

**Structured Query Language**

1. **Creating Database**

- Creating a Database
- Creating a Table
- Specifying Relational Data Types
- Specifying Constraints
- Creating Indexes

2. **Table and Record Handling**

- INSERT statement
- Using SELECT and INSERT together
- DELETE, UPDATE, TRUNCATE statements
- DROP, ALTER statements

3. **Retrieving Data from a Database**

- The SELECT statement
- Using the WHERE clause
- Using Logical Operators in the WHERE clause
- Using IN, BETWEEN, LIKE , ORDER BY, GROUP BY and HAVING

**Clause**

- Using Aggregate Functions
- Combining Tables Using JOINS
- Subqueries

4. **Database Management**

- Creating Views
- Creating Column Aliases
- Creating Database Users
- Using GRANT and REVOKE

Cursors in Oracle PL / SQL
SQL SERVER
Implementing Data Integrity
Implementing Indexing
Implementing Stored Procedures
Implementing Triggers
Implementing User-defined Functions
Managing Transactions and Locks
Writing Oracle PL / SQL Stored Procedures

## ALGORITHM LAB

Laboratory: The laboratory component will emphasize two areas:
Implementation of algorithms covered in class: This will involve running the algorithms under varying input sets and measuring running times, use of different data structures for the same algorithm (wherever applicable) to see its effect on time and space, comparison of different algorithms for the same problem etc.
Design of Algorithms: This will involve design and implementation of algorithms for problems not covered in class but related to a topic covered in class.
The exact set of algorithms to design and implement is to be decided by the instructor. In addition, there will be at least one significantly large design project involving some real world application. An efficient design of the project should require the use of multiple data structures and a combination of different algorithms/techniques.

**CS/MSc/1212:**

## COMPILER LAB

**List of Experiments**

1. Write a Program in LEX/YACC to check whether a given string is a valid ID (Identifier), Keyword, RELOP (Relational Operator) or others.

2. Write a program in LEX/YACC to check whether a given expression (relational or assignment or bitwise operator) is valid or not and it gives the type of expression as output.

3. Write a program in LEX /YACC to eliminate white space and collect numbers as a token.

4. Construct a syntax directed translation scheme that translates integers into roman numerals. Implement translator from integers to roman numerals based on above syntax directed translation using LEX/YACC.

5. Write a program using FLEX/YACC, which recognize regular expression.

6. Write a program in LEX/YACC to create and maintain a symbol table, which stores the name of the variable, type of the variable, structure name etc.

7. Write a C code analyzer in LEX/YACC: comments, code, white space, count braces, keywords etc. Try to identify function definition and declaration, which are names followed by '('outside of any braces.

8. Write programs in LEX/YACC, which create a simple desk calculator program that performs addition, subtraction, and multiplication and division operation. This calculator program also allows you to assign values to variables (each designated by a single lower case letter) and then use the variables in calculation.

9. Write a program in LEX/YACC to check whether a sentence of English language is grammatically correct or not.

10. Write a program in LEX/YACC which takes a English sentence as input and gives the output as the parts of speech.

11. Write a program in LEX/YACC which takes a C program as inputs and delete the comment, white space and Count the no of lines.

12. Write a program in LEX/YACC which counts the no of lines, total no of characters, total no of vowels and total no of punctuation marks in a paragraph.

**SOFTWARE ENGENEERING LAB**

Developing a software following all the phases of SDLC and full documentation of the software

## M.Sc. SEMESTER – III

## CS/MSc/2101 : COMPUTER NETWORK AND INTERNET

## M1: COMPUTER COMMUNICATION NETWORK

**Overview of data communication and Networking:** Introduction; Data communications: components, direction of data flow(simplex, half duplex, full duplex); Networks: distributed processing, network criteria, physical structure (type of connection, topology), categories of network (LAN, MAN,WAN); OSI reference model, TCP/IP reference model, their comparative study.
**Physical level:** Overview of data (analog & digital), transmission (analog & digital)& transmission media ( guided & non-guided); TDM, FDM, Circuit switching: time division & space division switch,
**Data link layer:** Types of errors, framing(character and bit stuffing), error detection & correction methods; Flow control; Protocols: Stop & wait ARQ, Go-Back- N ARQ, Selective repeat ARQ, HDLC;
**Medium access sub layer:** Point to point protocol, LCP, NCP, FDDI, token bus, token ring; Reservation, polling, concentration; Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, Traditional Ethernet, fast Ethernet;
**Network layer:**Internetworking & devices: Repeaters, Hubs, Bridges, Switches, Router, Gateway; Addressing : Internet address, classful address, subnetting; Routing : techniques, static vs. dynamic routing , routing table for classful address; Routing algorithms: shortest path algorithm, flooding, distance vector routing, link state routing; Protocols: ARP, RARP, IP, ICMP, IPV6; Unicast and multicast routing protocols.
**Transport layer:** Process to process delivery; UDP; TCP; Congestion control algorithm: Leaky bucket algorithm, Token bucket algorithm,
**Application layer:** DNS; SMTP, SNMP, FTP, HTTP & WWW; Security: Cryptography, user authentication, security protocols in internet, Firewalls.
**Modern topics:** ISDN services & ATM ; DSL technology, Cable modem, Sonet.
Wireless LAN: IEEE 802.11; Introduction to blue-tooth, VLAN's, Cellular telephony & Satellite network.

***Text Books:***


1. B. A. Forouzan – "Data Communications and Networking (3rd Ed.) " – TMH
2. A. S. Tanenbaum – "Computer Networks (4th Ed.)" – Pearson Education/PHI
3. W. Stallings – "Data and Computer Communications (5th Ed.)" – PHI/ Pearson Education
4. Zheng & Akhtar, Network for Computer Scientists & Engineers, OUP
5. Black, Data & Computer Communication, PHI
6. Miller, data Communication & Network, Vikas
7. Miller, Digital & Data Communication, Jaico
8. Shay, Understanding Data Communication & Network, Vikas

***Reference Books:***

1. Kurose and Rose – " Computer Networking -A top down approach featuring the internet" – Pearson Education
2. Leon, Garica, Widjaja – "Communication Networks" – TMH
3. Walrand – "Communication Networks" – TMH.
4. Comer – "Internetworking with TCP/IP, vol. 1, 2, 3($4^{th}$ Ed.)" – Pearson Education/PHI

**M2: INTERNET TECHNOLOGY**

1. To provide the student with basic knowledge of networking technology, including knowledge and understanding of basic network structure and the characteristics of star, bus, mesh, and ring topologies, and their advantages and disadvantages.
2. To provide the student with basic knowledge of the characteristics of segments and backbones.
3. To provide the student with basic knowledge of the major network operating systems, including Microsoft Windows NT, Novell NetWare, and Unix.
4. To provide the student with basic knowledge of the clients that best serve specific network operating systems and their resources, and the directory services of the major network operating systems.

5. ASP.NET, Internet Information Server, XML Web Services, AJAX and ATLAS, Difference between Web Form and Web Service.

**CS/MSc/2102: ADVANCE PROGRAMMING**

**M1: OBJECT-ORIENTED PROGRAMMING**

Evolution of object oriented programming: (a) Brief history (b) Application domain (c) Existing-programming languages.
Concept of object & classes, class as type, properties of a class : Structural & behavioral. Class construction from existing classes. Defining behavior by messages & methods. Concepts of polymorphism, operator overloading, encapsulation, public & private properties, types of inheritance, singles multiple &
restricted.
Concept of generic objects & Classes.
Introduction to .NET Platform, .NET namespaces, assemblies, object oriented programming features, Introduction to C# Programming Language, Classes and objects, Memory Management and Pointers, Overloading, Inheritance, Access Specifiers, Types of Inheritance—Private, Protected and Public Inheritance—Overloading—Direct and

Indirect base Classes –Abstract base Classes, Dynamic Binding, Structured Exception Handling, Reflection; Threading.

**M2: ADVANCED JAVA**

Advanced Web-based programming with an emphasis on the Java language and platform. basic constructs and syntax of the language. core advanced features which include such topics as Networking and Sockets, Remote Method Invocation (RMI), Java Beans, Multi-Threading and Lightweight Components (Swing). Use the java.io package to read and write files, use object serialization, use JDBC to access data from relational databases, and perform basic network communication by using sockets.

**CS/MSc/2103 : ARTIFICIAL INTELLIGENCE & NEURAL NETWORK**

**ARTIFICIAL INTELLIGENCE:** Scope of AI: Games, theorem proving, natural language processing, vision and speech processing, robotics, expert systems, AI techniques-search knowledge, abstraction.

Problem solving: State space search : Production systems. Search space control : Depth first, breadth first seach, heuristic search – Hill climbing, best first search, branch and bound. Minimax search : Alpha-Beta cut offs.
Knowledge Representation :Predicate Logic , Skolemizing queries, Unification. Modus pones. Resolution, dependency directed backtracking.
Rule Based Systems : Forward reasoning : Conflict resolution. Backward reasoning : Use of no backtrack .
Structured Knowledge Representations : Semantic Net : slots, exceptions and defaults Frames.
Handling uncertainty, Probabilistic reasoning. Use of certainty factors, Fuzzy logic.
Learning: Concept of learning, learning automation, genetic algorithm, learning by induction, neural nets-back propagation.

**NEURAL NETWORK:** Motivation; limitations of symbolic approach to machine intelligence Perceptrons; training; convergence theorem; linear separability Multilayer perceptron; Backpropagation; Boltzmann machine and simulated annealing Hopfield net; energy; stability; capacity; Application to optimization problems Unsupervised learning; Adaptive Resonance Theory; Kohonen network Capabilities of multilayer perceptrons; complexity issues; importance of Kolmogorov complexity; Probably Approximately Correct learning and neural networks Hybrid systems; importance of integrating symbolic and connectionist systems; connectionist expert systems

**CS/MSc/2104 : ELECTIVE - I**

## List of Electives:

1. Image Processing & Pattern Recognition.
2. Bio-informatics.
3. VLSI Design.
4. Embedded System.
5. Mobile Computing.
6. Soft Computing.
7. Fuzzy set theory

### 1. PATTERN RECOGNITION

Introduction to pattern recognition and applications to OCR, speech recognition, fingerprints, signatures etc. Commercial importance of applications. Introduction to Statistical, Neural and Structural Approaches. Statistical Pattern Recognition: Patterns and classification, discriminant functions, Bayes decision rule, nearest neighbour rule, probability of error. Linear discriminant functions: Perceptrons and training, LMSE approaches. Unsupervised learning and clustering. Feature extraction. Neural Approach: Introduction to artificial neural networks, feed forward networks, delta rule and back propagation, Hopfield networks and unsupervised learning, Adaptive resonance architectures, related techniques. Pattern associators and content addressable memories, hardware realizations. Syntactic pattern recognition: Formal languages and grammars Pattern grammars and higher dimensional grammars, Parsing, automata realizations, stochastic grammars, Grammatical Inference, computational learning theory, Valiant"s framework.

### 2. BIO-INFORMATICS:

Sequence similarity, homology, and alignment. Pairwise alignment: scoring model, dynamic programming algorithms, heuristic alignment, and pairwise alignment using Hidden Markov Models. Multiple alignment: scoring model, local alignment gapped and ungapped global alignment. Motif finding: motif models, finding occurrence of known sites, discovering new sites.
Gene Finding: predicting reading frames, maximal dependence decomposition. Analysis of DNA microarray data using hierarchical clustering, model-based clustering, expectation-maximization clustering, Bayesian model selection.

**3. VLSI DESIGN:**

Introduction to VLSI Technology: CMOS logic; behavioral, structural and physical representations; MOS transistor theory; processing technology; circuit characterization; performance estimation; circuit and logic design.

Systems Design and Methods: design strategies; CMOS chip design options, including reprogrammable gate arrays, sea-of-gates and standard cells; design methods; design capture tools, including hardware definition languages such as VHDL and packages such as MCE BX.

Design Verification and Testing: simulation at various levels, including circuit and logic level; design rule verification; timing verification; fault models; design strategies for testing; chip level test techniques; system level test techniques.

CMOS Subsystem Design: datapath operations; memory elements; control, including finite state machines and control logic implementations via PLAs and ROM.

Students will be required to undertake, as private study, significant practical work asociated with the issues of the class.

**4. EMBEDDED SYSTEM:**

Introduction and Hardware Environment Overview of embedded system, categories of embedded system, processor technology, design technology, applications : consumer electronics, control & industrial automation, network information appliances, wireless communications Hardware architecture : processor, memory, latches, buffers, ports, timers, counters, watchdog timers, UART, pulse width modulators, LCD controllers, keypad controllers, stepper motor controllers, analog-to-digital converters, real time clocks.

Communication Principles : Parallel, serial, wireless and layering, Protocols : 12C, CAN, FireWire, USB, PCI bus, ARM bus, IrDa, Bluetooth, IEEE 802.11, operating system, kernel architecture, embadded operating system, context switch, task synchronization, real time and mobile operating system, programming languages, development tools for host & target machines, embedded system development system, interrupt basics, interrupt handling.

VC++ Programming Introduction to MFC & windows, MFC fundamentals, processing messages, message boxes, menus, dialog boxes, common controls (Radio buttons, check boxes, scroll bars, buttons, cursor, icons, managing texts), properties sheet.

Project Studies Simple LED blinking program, device driver programming, serial communication programming for PC-to-PC communication, development of navigation system, protocol converter.

Books :

1. Programming for embedded system by Dr. Prasas, Vikas Gupta, Das & Verma, Pub, WILEY Dreamtech india Pvt.
2. Embadded System Design. by Frank Vashid & Tony Givergis, Pub, WILEY.
3. MFC Programming. by Herbert Schildt, Pub. TataMcGraw Hill.

Ref. Books :

1. An Embedded software primer by David E. Simon, Pub. Low Price Edition.
2. Programming Embedded Systems by Michael Barr, Pub. O'REILLY

## 5. MOBILE COMPUTING

**Introduction to wireless networking.**
Advantages and disadvantages of wireless networking
**Characteristics of radio propagation.**
Fading, Multipath propagation
**Introduction to digital transmission.**
Definition of bit-rate and signalling rate. Introduction to synchronous transmission. The need for pulse shaping, synchronisation and line-coding. Calculation of bit-error probabilities when the channel is affected by the addition of Gaussian noise.
**Narrowband digital modulation.**
The need for modulation. Binary and multi-level (M-ary) amplitude-shift keying (ASK), frequency-shift keying (FSK) and phase-shift keying (PSK).
**Wideband modulation techniques to cope with intersymbol interference**
Direct sequence spread spectrum Adaptive Equalization Orthogonal frequency division multiplex
**Medium Access Control (MAC).**
MAC protocols for digital cellular systems such as GSM. MAC protocols for wireless LANs such as IEEE802.11 and HIPERLAN I and II. The near far effect. Hidden and exposed terminals. Collision Avoidance (RTS-CTS) protocols.

**Protocols supporting mobility.**

Mobile network layer protocols such as mobile-IP, Dynamic Host Configuration Protocol (DHCP). Mobile transport layer protocols such as mobile-TCP, indirect-TCP. Wireless Application Protocol (WAP).

## 6. SOFT COMPUTING

**Introduction to artificial neural network**

Neural Networks: History, overview of biological Neuro-system, Mathematical Models of Neurons, ANN architecture, Learning rules, Learning Paradigms-Supervised, Unsupervised and reinforcement Learning, ANN training Algorithms-perceptions, Training rules, Delta, Back Propagation Algorithm, Multilayer Perceptron Model, Applications of Artificial Neural Networks.

Competitive learning networks, Kohonen self organizing networks, Hebbian learning; Hopfield Networks, Associative Memories, The boltzman machine; Applications.

**Fuzzy Logic**

Fuzzy Logic: Introduction to Fuzzy Logic, Classical and Fuzzy Sets: Overview of Classical Sets, Membership Function, Fuzzy rule generation. Operations on Fuzzy Sets: Compliment, Intersections, Unions, Combinations of Operations, Aggregation Operations. Fuzzy Arithmetic: Fuzzy Numbers, Linguistic Variables, Arithmetic Operations on Intervals & Numbers, Lattice of Fuzzy Numbers, Fuzzy Equations.Fuzzy Logic: Classical Logic.

Genetic algorithms(Gas),Evolution strategies(Ess),Evolutionary programming(EP),Genetic Programming(GP),Selecting,crossover,mutation,schema analysis,analysis of selection algorithms;convergence;Markov & other stochastic models**.**

Other Soft computing approaches

Simulated Annealing, Tabu Search, Ant colony based optimisation, etc.

**Text:**


1. “Neuro-Fuzzy and Soft computing”, Jang, Sun, Mizutani, Pearson
2. “Neural networks: a comprehensive foundation”, Haykin, Pearson
3. “Genetic Algorithms”, Goldberg, Pearson
4. “Fuzzy Sets & Fuzzy Logic”, G.J. Klir & B. Yuan, PHI.


**Reference:**


1. “An Introduction to Neural Networks”, Anderson J.A., PHI, 1999.
2. “Introduction to the Theory of Neural Computation”, Hertz J. Krogh, R.G. Palmer, Addison-Wesley, California, 1991.
3. “An Introduction to Genetic Algorithm”, Melanie Mitchell, PHI, 1998.
4. “Neural Networks-A Comprehensive Foundations”, Prentice-Hall International, New Jersey, 1999.

5. “Neural Networks: Algorithms, Applications and Programming Techniques”, Freeman J.A. & D.M. Skapura, Addison Wesley, Reading, Mass, (1992).

## 7. FUZZY SET THEORY

**Introduction** : Objective, Overview.

**Fuzzy Set Theory**: Fuzzy sets- Fuzzy sets and classic fuzzy operators , MF formulation and parameterization , Extended fuzzy union, intersection, and complement ; Fuzzy rules and fuzzy reasoning - Extension principle , Fuzzy relations , Fuzzy if-then rules ,Fuzzy reasoning ; Fuzzy inference systems-Mamdani's fuzzy models , Sugeno's fuzzy models ,Tsukamoto's fuzzy models ,Other variants ;Others-Fuzzy arithmetic, Fuzzy clustering .

**Regression and optimization**:-Least-squares estimator-Matrix techniques ,Least-squares estimator and its geometric interpretation ,Recursive Least-squares estimator ,Recursive Least-squares estimator with forgetting factors ,Maximum likelihood estimator ;Gradient-based optimization -Steepest descent ,Newton's method ,Step size determination ,Gauss-Newton method ,Levenberg-Margquardt method ; Gradient-free optimization -Genetic algorithms ,Simulated annealing ,Downhill search ,Random search .

**Adaptive fuzzy inference systems** :-Adaptive networks -Architectures ,Learning rules ;Adaptive neuro-fuzzy inference systems (ANFIS) –Architectures,Hybrid learning rules .

**Applications:-**Data modeling ,Pattern recognition ,Adaptive fuzzy control

## CS/MSc/2111: NETWORK LAB & WEB-PAGE DESIGNING LAB

**NETWORK LAB:**

- History of networking
- Binary arithmetic
- Network devices
- Network models
- Topologies
- Protocols
- Addressing
- Routing
- LAN & WAN

**WEB-PAGE DESIGNING LAB:**

Visual Web Developer tools and features, Simple ASP.NET Web Page, Adding Data on a Web Page, Simple Web Service, Adding a Web Service to a Web Site, User Interface Design, Adding Master Pages

**CS/MSc/2106**

**M1: ARTIFICIAL INTELLIGENCE LAB**

Programming in PROLOG.

**M2: JAVA LAB**

1. Assignments on class, constructor, overloading, inheritance, overriding
2. Assignments on wrapper class, vectors, arrays
3. Assignments on developing interfaces- multiple inheritance, extending interfaces
4. Assignments on creating and accessing packages
5. Assignments on multithreaded programming, handling errors and exceptions, applet programming and graphics programming
6. Use of CASE tools

**Note: Use Java and C++ as programming language, Use UML where applicable.**

## M.Sc. SEMESTER – IV

External Project. Student will do their project in any Industry / academic Institute.