University of Pune
Interdisciplinary School of Scientific Computing

Course work for Ph.D. in Scientific Computing

i) Ph. D. students with Master's Degree in disciplines other than scientific computing need to do one course each from the following groups.

For students other than Scientific Computing.

<u>Subject Courses</u>

Group – I	Group – II	Group – III
SC–101 : (Programming Languages)	SC–103 : (Advanced Database Management Concepts)	Elective Courses
SC–204 : (Numerical Methods-I)	SC–104 : (Mathematics for Scientific Computing)	
SC–202 : (Operating System)	SC–302 : (Scientific Visualization)	
SC–304 : (Numerical Methods-II)		

ii) Ph.D. students with Master's degree in Scientific Computing can take up one course each from following group.

For students with M.Sc. (Scientific Computing)

Subject Courses.

Group – I	Group – II
Courses from Mathematics Statistics/Physics/.Chemistry	Courses from Computer Science / Or Elective course which was not offered during his/her M.Sc.

UNIVERSITY OF PUNE
INTERDISCIPLINARY SCHOOL OF SCIENTIFIC COMPUTING

RESEARCH METHODOLOGY SYLLABUS

INTRODUCTION : Definition of Research, Qualities of Researcher, Components of Research Problem, Various Steps in Scientific Research, Types of Research; Hypotheses Research Purposes - Research Design - Survey Research - Case Study Research, Literature searching

DATA COLLECTION: - Sources of Data: Primary Data, Secondary Data; Procedure Questionnaire - Sampling Merits and Demerits - Experiments - Kinds - Procedure; Control Observation - Merits - Demerits - Kinds - Procedure - Sampling Errors - Type-I Error - Type-II Error.

STATISTICAL ANALYSIS: Introduction to Statistics - Probability Theories - Conditional Probability, Poisson Distribution, Binomial Distribution and Properties of Normal Distributions, Point and Interval Estimates of Means and Proportions; Hypothesis Tests, One Sample Test - Two Sample Tests / Chi-Square Test, Association of Attributes - t-Test - Standard deviation - Co-efficient of variations. Correlation and Regression Analysis. Introduction to statistical packages.

SCIENTIFIC WRITING : Structure and Components of Research Report, Types of Report: papers, thesis., Good Research Report, Pictures and Graphs, citation styles,

References:

Sheldon Ross. Introduction to Probability

Sheldon Ross. Introduction to Probability and Statistics for Engineers and Scientists

Michael Alley. The Craft of Scientific Writing. Springer; 3rd edition (August 29, 1996).

Detailed Syllabus:

SC – 101 Principles of Programming Languages

1. Introduction and Motivation **[8 hrs]**

Idea of analyzing an algorithm through examples, introduction to some notations, comparison of algorithms, notions of space and time efficiency and motivation for algorithm design methods, demonstration of algorithm analysis for some suitable example algorithm, say merge sort.

2. Algorithm Analysis Techniques **[12 hrs]**

(a) Asymptotic Analysis

Detailed coverage of asymptotic notations and analysis. Big Omicron, Big Theta, Big Omega, Small theta, Small omega. Comparison of the Insertion Sort and the Merge Sort Algorithms.

(b) Recurrence Analysis

Introduction to Recurrence equations and their solution techniques (Substitution Method, Recursion Tree Method, and the Master Method), Proof of the Master Method for solving Recurrences. Demonstration of the applicability of Master Theorem to a few algorithms and their analysis using Recurrence Equations. (Example algorithms: Binary Search, Powering a number, Strassen's Matrix Multiplication)

(c) Analysis of more Sorting algorithms: Quick Sort and Counting Sort

3. Algorithm Design Techniques **[12 hrs]**

(a) Types of Algorithms

(b) Dynamic Programming

Introduction and the method for constructing a DP solution, illustrative problems, e.g. assembly line scheduling problem using DP, or solution for the matrix chain multiplication.

(c) Greedy Algorithms

Greedy vs. DP, methodology, illustrative problems, e.g. the knapsack problem using a greedy technique, or activity selection Problem. Construction of Huffman Codes.

(d) Backtracking

Introduction to recursion, solving the 0-1 Knapsack problem using backtracking, pruning in backtracking and how it speeds up the solution for the 0-1 Knapsack problem.

(e) Branch and Bound

Description and comparison with backtracking, the FIFO B&B and the Max Profit B&B using the 0-1 Knapsack problem

4. Graph Theory **[12 hrs]**

(a) Breadth First and Depth First Search Algorithms

(b) Minimum Spanning Trees, Kruskal's Algorithm

(c) Minimum Spanning Trees, Prim's Algorithm

(d) Properties of Shortest paths.

(e) Dijkstra's Algorithm

(f) Bellman Ford Algorithm

5. NP-Completeness **[12 hrs]**

(a) Polynomial time

(b) Polynomial time verification (NP problems)

(c) Concept of NP-Hard with example (Halting problem)

(d) NP-Completeness and Reducibility (without proof)

(e) Some NP-Complete problems

(f) Overview of showing problems to be NP-Complete

Text Reference:

1. Introduction to Algorithms,
T.H.Cormen, C.E.Leiserson, R.L.Rivest,

Prentice Hall India, 2002.

2. The Art of Computer Programming, Vols. 1 and 3,
D.E.Knuth,

Addison Wesley, 1998.

3. Design and Analysis of Algorithms,
A.V.Aho, J.E.Hopcroft, J.D.Ullman,

Addison Wesley, 1976.

4. Fundamentals of Computer Algorithms,
E.Horowitz, S.Sahni,

Galgotia Publishers, 1984.

5. Data Structures and Algorithms, Vols.1 and 2,
K.Melhorn,

Springer Verlag, 1984.

6. The Analysis of Algorithms,
P.W.Purdom, Jr. and C.A.Brown,

Holt Rhinehart and Winston, 1985.

SC-103 : ADVANCED DATABASE MANAGEMENT CONCEPTS

1. Review of Database Management Concepts [6 hrs]

Types of database, Normalization (1Nf,2Nf,3Nf, BCNF,5NF), data models, constraints, ER-model, Introduction to Scientific Database

2. Data Storage & Indexing techniques [6 hrs]

Architecture of DBMS, Storage of data on disk & files, File organization & type of file organization, Advanced storage devices – RAID, Type of Indexing

3. SQL query optimization [8 hrs]

Implementation and Evaluation of relational operations, Types of joins and join algorithms, Select of appropriate index, database workload, which index to create, guidelines for index selection, co-clustering, index on multiple attributes, Cost estimation and cost based optimization, Plan evaluation, Tuning conceptual schema

4. Review of Transaction Management & Security [6 hrs]

Introduction to transaction management (ACID property, states of transaction), Concurrency control (locked based concurrency control, optimistic concurrency control, timestamp based concurrency control, deadlock detection & handling), Crash recovery (log based recovery, shadow paging), Security (identification & authentication , authorization matrix, views, encryption techniques, statistical database, polyinstantiation, role of DBA)

5. Parallel and Distributed Database (DDB) [20 hrs]

Introduction to Parallel database, Architecture of Parallel database, Parallel query evaluation & optimization, Introduction to DDB, DDBMS Architecture, Storing data in DDBMS, Catlog Management, Query Processing(non-join queries, joins – fetch as needed, ship to one site, semijoins), Updating database, Transaction Management, Concurrency Control, Recovery

6. Data Mining [8 hrs]

Introduction to data mining, Knowledge Representation Using Rules - Association and Classification Rules, Sequential Patterns, Algorithms for Rule Accessing

ASSIGNMENT:

Case study which should include following things - ER diagram, functional schema,

Normalization, indexing

Text Reference:

1. Database Management System

   Raghuramkrishnan, Gehrke (3rd Edition)

   McGraw Hill

2. Database System Concepts(second edition)

   Silberschatz, Korth, Sudarshan

   McGraw Hill

3. Introduction to Database System

   C.J. Date

4. Fundamentals of Database systems (fourth edition)

   Elmaris, Nawathe

   Pearson Education

5. Distributed Database systems

   Tamer Ozsu

   Pearson Education

SC-104 : MATHEMATICS FOR SCIENTIFIC COMPUTING

1. Function **[2 hrs]**

Definition, examples, graphs of functions

2. Limit of a function **[2 hrs]**

Definition, right hand & left hand limits examples.

3. Continuity **[2 hrs]**

Definition, examples and properties of continuous functions ,Types of discontinuity.

4. Derivatives : **[2 hrs]**

Definition, geometrical interpretation, Derivatives of elementary functions by first principle. Problems on velocity and acceleration product, quotient and chain rule, implicit differentiation, derivative of inverse function.

5. Applications of derivatives **[7 hrs]**

Concavity and points of inflation, Maxima and minima of a function, Related rates, Roll's theorem and Mean value Theorem, L′ Hospitals rule

6. Integration **[2 hrs]**

Introduction, elementary integration formulae, indefinite and definite integrals.

7. Integration Methods **[4 hrs]**

Substitution, Integration by parts, Integration of product and power's of trigonometric functions, partial fractions.

8. Applications of integration **[5 hrs]**

Area under the curve, length of a curve, volumes and surface areas of solids of revolution.

9. Infinite Series **[7 hrs]**

Sequences and series of numbers, Limit of sequence, convergence criteria for series, Tests of convergence for series of positive numbers, Power series – region of convergence, tests of convergence, Term by term integration of power series, Fourier series.

10. Matrices **[8 hrs]**

Determinants – definition and interpretation, Matrices – Rank of matrix, Solutions of linear equations, Eigenvalues and eigenvectors, Similar matrices and diagonalization

11. Vectors **[9 hrs]**

Vector: Introduction, scalar and vector product, Equation of line and plane, using vectors, Projections, Equations of circle, parabola, ellipse and hyperbola cylinders and quadric surfaces, Vector functions: Derivatives, tangent vectors velocity and acceleration, Arc length of space curve, Curvature and normal vectors.

12. Functions of two or more variables **[6 hrs]**

Definition, limits and continuity, partial derivatives, Directional derivatives, gradients and tangent planes, Second derivative, maxima, minima, saddle points

13. Differential Equations **[5 hrs]**

First order ODE, variables separable form, Solution of first order linear equation, Second and higher order equations, Solution of constant coefficient second order equation.

Text References:

1. Calculus and Analytical Geometry (9th Edition)

   Thomas and Finney

   Pearson Education

2. Calculus (5th Edition)

   James Stewart

3. Advanced Engineering Mathematics (8th Edition)

   Erwin Kreyszig

   John Willey and Sons

4. Linear Algebra (2nd edition)

   Hoffman and Kunz

   Prentice Hall International

5. Linear Algebra

   Peter D. Lax

6. Differentials Equations with applications and Historical notes.

   Simmons G.F.

SC-202 : OPERATING SYSTEMS

1. Introduction to UNIX. **[10 hrs]**

Evolution of UNIX: Past, Present and future, Philosophy of UNIX: System's Internal Structure, the process interface, OS features, OS systems Concepts: File Systems, Processes.

Deviation from Unix: The GNU project.

2. Implementation of buffer cache. **[4 hrs]**

Structure and Philosophy of the cache implementation. Algorithms used by the buffer cache. Also, touch upon inode cache

3. File system. **[10 hrs]**

The file system switchers table, VFS architecture, File systems implementation on disk, File system handling kernel algorithms, Issues for file system handling, System calls for file system manipulations.

4. Process. **[7 hrs]**

State transitions, Process structure and layout, Multiprocessing details-Context, context switches, memory management concepts, System calls.

5. Process Scheduler. **[7 hrs]**

Class specific implementation, Priorities, system calls.

6. Memory Management Techniques. **[3 hrs]**

Swapping, Demand paging, Hybrid, Virtual Memory.

7. Time and Clock. **[3 hrs]**

8. I/O Subsystems. **[5 hrs]**

Concepts, data structures, device drivers, streams.

9. Interprocess Communication, and thread communication **[5 hrs]**

Threads- Thread creation. Inter thread control, Thread synchronization, Inter process communication.

ASSIGNMENTS:

1. Programming using Unix system calls

2. File system & directories, File operations.

3. Programming using Unix system calls

4. Process system, using commands like strace, top, lsof, ps.

Text Reference:

1. Design of the Unix System,
   M. J. Bach
   Prentice Hall

2. Operating System Concepts
   J. L. Peterson, A. Silberschatz, Galvin

   (Addision Wesley)

3. Modern Operating System
   Andrew Tanenbaum

   Pearson Education.

4. Advanced Concepts in Operating System

   Mukesh Singhal, Niranjan Shivrathri

5. Operating System Internals and Design Principles

   William Stallings

SC-204 : NUMERICAL METHODS FOR SCIENTIFIC COMPUTING-I

1. Number System and Errors **[4 hrs]**

Representation on integers and floating point numbers, Errors in computation, loss of significance.

2. Solutions of Equations in one variable **[6 hrs]**

Bisection Method, Newton Raphson Method, Fixed Point iteration, Error Analysis, Accelerating Convergence, Polynomial Evaluation – Horner's rule, Zeros of polynomials and Muller's Method

3. Systems of Linear Equations **[8 hrs]**

Gaussian Elimination, Triangular decomposition, Pivoting strategies, Error analysis and Operations count, Ill-conditioning and condition number of system, Evaluation of determinants

4. Eigenvalue Computations **[11 hrs]**

Diagonalization of system of ODE, Power Method, Gerschgorin theorem, Jacobi's Method, Given's and Householder's methods for Tridiagonalization, Method of sturm sequences for tridiagonal matrix, Lanczos Method, QR Factorization

5. Curve fitting and Approximation **[12 hrs]**

Lagrange's interpolation, Polynomial wiggle problem, Spline interpolation, Least Square Method – line and other curves, Orthogonal Polynomials, Tchebyshev interpolation, Fourier approximation and Fast Fourier, Transforms (FFT) algorithm.

6. Numerical Differentiation and Integration **[9 hrs]**

Numerical Differentiation – Richardson Extrapolation method, Numerical Integration – Newton Cotes quadrature for equidistant points, Gaussian Quadrature, Integration using Tchebyshev Polynomials

ASSIGNMENT:

1. Solutions of nonlinear equations

2. Fixed point iterations

3. Linear equation solvers

4. Eigenvalue computations

5. Curve fitting, interpolation

6. Numerical Integration

Text References:

1. Numerical Methods for Mathematics, Science and Engineering
   John H. Mathews.

2. Numerical Analysis (7th Edition)

   Richard and J. Douglas Faires

3. Numerical Analysis

   C. E. Froberg

4. Numerical Analysis – A practical Approach

   Maron M.J.

5. A First Course in Numerical Analysis

   Ralston and Rabinowitz.

SC-302 : SCIENTIFIC VISUALIZATION

1. Introduction to Computer Graphics. **[5 hrs]**

Examples of Graphics applications. Key journals in Graphics input and output graphics devices, world coordinate systems viewports and world to viewport mapping.

2. Raster graphics technique **[5 hrs]**

Line drawing algorithms scan converting circles and ellipses, polygon filling with solid colors and filling patterns, half toning and dithering techniques.

3. Vectors and their use in graphics **[2 hrs]**

Operation with vectors, adding, scaling, subtracting, linear spaces, dot product, cross Product, Scalar triple product, application of dot product, cross product and scalar triple product, application of vectors to polygons.

4. Transformation of pictures **[3 hrs]**

2D Affine transformations, use of Homogenous coordinates, 3D affine transformations.

5. 3D viewing **[6 hrs]**

Synthetic camera approach, describing objects in view coordinates, perspective and parallel Projections, 3D clipping.

6. 3D graphics, Write frame models **[5 hrs]**

Marching cube algorithms for contour generation and surface generation from a given data over 2D/3D grid.

7. Hidden line and surface removal, backface culling **[7 hrs]**

8. Light and shading models, rendering polygonal masks Flat, Gouraud, phone shading.

[8 hrs]

9. Ray Tracing **[11 hrs]**

Overview, intersecting ray with plane, square, cylinder, cone, cube and sphere, drawing shaded pictures of scenes, reflections and transparency.

ASSIGNMENTS:

1. 2D transformations
2. 3D transformations
3. Projections, Clipping, Shading
4. Contour, Marching cube
5. Creating a graphics package using above assignments.

Text References:

1. Computer Graphics: A programming approach,
   S. Harrington

   (McGraw Hill, 1986)

2. Computer Graphics – Principles and Practice
   Foley, Van Dam

3. Computer Graphics,
   F. S. Hill Jr.

   (Macmillan, 1990)

4. Procedural Elements for Computer Graphics.
   D.F. Rogers

   (McGraw Hill, 1995)

5. Principles of Interactive Computer Graphics
   William Newman, Robert Sproull

SC-304 : NUMERICAL METHODS FOR SCIENTIFIC COMPUTING-II.

1. Numerical Solutions of ODE **[18 hrs]**

Single step and Multistep methods Predictor-Corrector Methods. Boundary value problems stiff Equations.

2. Numerical Solutions of PDE **[12 hrs]**

Elliptic Equation, Parabolic Equation, Hyperbolic Equation

3. Numerical Optimization **[15 hrs]**

Linear Optimization: Simplex method, Transportation Problem.

Nonlinear Optimization: Golden ratio method, Nelder Mead method, Conjugate gradient method.

ASSIGNMENTS:

1. ODE solvers – Euler, Runge Kutta, Multistep methods
2. PDE solvers – Finite difference methods
3. Simple method and transportation
4. Conjugate gradient method, Golden search method

Text References:

1. Numerical Methods for Mathematics Science & Engineering. (Second Edition)

John H. Mathews

Prentice Hall of India.

2. Numerical Analysis (Seventh Edition)

Burden and Faires

Thomson Asia PTE. LTD

3. Numerical Analysis- A practical Approach

Maron
Mc Millan,1982

4. Numerical Analysis

Forberg

McGraw Hill,1979

5. A First Course in the Numerical Analysis of Differentials Equations

Arieh Iserles

The Press Syndicate of the University of Cambridge.

6. Numerical Solution of Partial Differentials Equations (Third Edition)

G.D.Smith

Oxford University Press

7. Numerical Methods for Scientists & Engineers

H.M.Antia

THM,1991

EL-1: APPLICATIONS OF COMPUTERS TO CHEMISTRY.

1. Computational Chemistry. [1 hrs]

Why learn Computational Chemistry? Applications areas: fundamental understanding. Predictions, design, structure of biomolecules, polymer design, catalyst and drug design.

2. Fundamentals of Chemistry. [2 hrs]

Concepts in chemistry like valency, hybridization, electronegativity, covalent bond, ionic bond, hydrogen bond, co-ordinate bond. Geometries of molecules like linear, angular, tetrahedral, etc.

3. Molecular Representations and Search [4 hrs]

Connectivity matrix, Adjacency matrix, SMILES notation, substructure search

4. Molecular Graphics and fitting. [4 hrs]

3-dimensional structures, steric pictures, CPK models, molecular dimensions and van der Waals volume.

5. Force Field (FF) Methods [4 hrs]

Molecular mechanics expressions for bond stretch, bond angle, torsion, improper torsion, Van der Waals, electrostatics and cross terms. Types of force fields, computational aspects in FF, parameterization in FF. Evaluation of number of energy terms for a given molecule

6. Classical energy minimization techniques [3 hrs]

Energy minimization by simplex, steepest descent, conjugate gradient and Newton-Raphson methods.

7. Conformational Analysis **[4 hrs]**

What is a conformation? Systematic, Monte Carlo and genetic algorithm based conformational analysis. Polling method of conformational analysis, simulated annealing method

8. Semi-empirical QM calculations. **[6 hrs]**

Cluster model calculations for the electronic structure of extended systems, Prospect and pitfalls in the usage.

9. Molecular Docking **[4 hrs]**

Concepts in docking. Parameterization in docking, Rigid docking, flexible docking, virtual screening, Scoring functions

10. Molecular Descriptors **[4 hrs]**

Molecular connectivity indices, topological indices, electro-topological indices, information theory indices, etc.

11. Quantitative Structure Activity Relationship **[6 hrs]**

Generation of training and test set methods, variable selection methods like stepwise forward, backward, etc. Regression methods like multiple regression, principal component regression.

12. Futuristic modeling techniques. **[2 hrs]**

Expert systems, Neural networks, Artificial Intelligence and virtual reality.

Text Reference:

1. An Introduction to Chemoinformatics,

A.R. Leach; V.J. Gillet,

Kluwer Academic Publishers, The Netherlands, 2003.

2. Introduction to Computational Chemistry

Frank Jensen

Chichester, Wiley, 2006.

3. Molecular Modeling: Principals and Applications

A.R. Leach

Pearson Education Limited, Essex, 2001

4. Essentials of Computational Chemistry: theories and models

Christopher J. Cramer

John Wiley, 2004

5. Optimization in Computational Chemistry and Molecular Biology: local and global

Approaches

M. Panos, Christodoulos A. Pardalos

Floudas - Science – 2000

6. Chemoinformatics: A Textbook

J. (Johann) Gasteiger

Thomas Engel – 2003

EL-2: PARALLEL PROCESSING AND GRID COMPUTING

1. Introduction **[3 hrs]**

Need for high-speed computing, need of parallel computers, Features of parallel computers, hardware requirements.

2. Solving Problem in parallel **[2 hrs]**

Temporal parallelism, data parallelism, comparison of temporal and data parallelism with specialized processors intertask dependency.

3. Structure of parallel computers **[5 hrs]**

Pipelined parallelism computers, array processors, a generalized structure of parallel computer, shared memory multiprocessors, message passing multicomputers multilink multidimensional computing system.

4. Programming parallel computers **[14 hrs]**

Programming message-passing multicomputers, programming shared memory parallel computer, programming vector computers.

5. Case Studies **[4 hrs]**

Matrix Multiplication, Graph theory, N-Body Simulation, Computer Vision and Image Processing applications

6. Grid Computing **[12 hrs]**

Introduction, grid architecture, resource sharing and allocation, job scheduling, grid security, Globus, Open grid services architecture (OGSA)

ASSIGNMENTS:

1. Matrix multiplication

2. Computing value of Pi
3. Solution to linear equation using Jacobi method

4. Finding patterns in a text file

5. Producer-consumer problem

6. All-pair shortest paths problem

7. N-body simulation

Text Reference:

1. Introduction to Parallel Computing, 2nd Edition
   Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar

   Pearson Education Ltd., 2004

2. Parallel Computers-Architecture and Programming
   V. Rajaraman and C. Siva Ram Murthy

   Prentice-Hall (India), 2000

3. Designing and Building Parallel Programs
   Ian Foster

   Addison-Wesley Inc., 1995

4. Elements of Parallel Computing
   V. Rajaraman

   Prentice Hall, 1990

5. Introduction to Grid Computing
   Bart Jacob, Michael Brown, Kentaro Fukui, and Nihar Trivedi

   IBM Redbook, 2005

EL-3: STATISTICAL COMPUTING

1. Introduction **[3 hrs]**

Computation of averages and measures of central tendency, skewness and kurtosis. Preparation of frequency tables, computation of Pearson and rank correlation coefficients.

2. Statistical Distributions **[6 hrs]**

Evaluation of standard probability mass functions, cumulative density functions, and quantile functions.

3. Monte Carlo **[4 hrs]**

Random number generation including Markov Chain Monte Carlo (MCMC). Statistical simulation and Monte Carlo studies in statistics.

4. Numerical optimization and root finding methods, including the Newton-Raphson method and the EM algorithm. **[5 hrs]**

5. Resampling techniques, including the permutation test and the bootstrap. **[5 hrs]**

6. Classification using discriminant functions, rough sets, artificial neural networks, and decision trees. **[12 hrs]**

7. Clustering techniques **[10 hrs]**

8. Multiple regression Analysis **[6 hrs]**

Text References:

1. Simulation Modeling and analysis
   Averill M. Law and W. David Kelton

2. Applied Multivariate Statistical Analysis
   Richard A. Johnson and Dean W. Wichern

3. Data Mining: Concepts and Techniques
   Jiawei Han and Micheline Kamber

4. Data Mining : Multimedia, Soft Computing, Bioinformatics
   Sushmita Mitra and Tinku Acharya

5. Computational Statistics
   Geof H. Given and Jennifer A. Hoeting

EL-4: Applications of Computer to Physics

1. Modeling and Simulations in Physics: **[2 hrs]**

Introduction to modeling and simulations. Various types of modeling like deterministic, Stochastic, etc. Advantages of Simulations and complementary nature of simulation to theory and experimental probes of nature. (Refer Ingels first two chapters and first chapter of Allen and Tildesley)

2. Monte Carlo methods. **[12 hrs]**

Random number generator, elementary ideas on random walk, Markov Chain, Metropolis algorithm. Approach to Equilibrium, particle in a box, Entropy, correlation and chaos. Microcanonical ensemble, simulation of classical ideal gas, Boltzmann distribution. Application to statistical mechanics of Ising model (one and two dimensions). Ising Hamiltonian, calculation of themodynamic quantities like specific heat and susceptibility, Tc, the optical exponent.

3. Percolation, Fractals, Kinetic Growth Models and Cellular Automata: **[12 hrs]**

Percolation threshold, site percolation on square and triangular lattices, continuum percolation, cluster size distribution, cluster labeling, Critical exponents and finite-size scaling. Fractal dimension, regular fractals & self-similarity, fractal growth models and cellular automata.

4. Classical molecular dynamic. **[14 hrs]**

Interacting particles and inter-atomic potentials. Hard sphere model. Elementary ideas of ensembles. Lagrangean equation of motion for N particles, Verlet algorithms and Measurements.

ASSIGNMENTS:

A full implementation of at least five codes from the following problems are expected:

1. Hard-sphere model (as in Allen & Tildesley)

2. MD for Argon (any other system of at least 2 to12 atoms) using (as in Allen & Tildesley)

(a) Leap-frog method

(b) Verlet Algorithm

3. Velocity Verlet

4. Generation of Random numbers, and their tests. (as in Gould & Tobochnik)

5. 1D and 2D Ising model Metropolis-Monte Carlo (as in Gould & Tobochnik)

6. Random walk for Polymers (as in Gould & Tobochnik)

7. Microcanonical ensemble Monte Carlo – Ideal gas (as in Gould & Tobochnik)

8. As many exercises possible from Gould and Tobochnik; chapters 11 to 15.

Text Reference:

1. What every Engineer should know about computer Modeling and Simulations Don M. Ingles, Marcel Decker, 1985 (First two chapters).

2. Computer Simulation Methods- Parts I & II
   Gould and Tobochnik, Addison Wesley (Chapters 6, 11, 12, 14 and 15).
3. Computer Simulation of Liquids
   M.P. Allen and D. Tildesley, Clarendon Press Oxford 1987.

EL-5. BIOLOGICAL SEQUENCE ANALYSIS

1. Analysis of DNA and Protein sequence-distribution, frequency statistics, pattern and motif searches-randomization etc.-sequence segmentation. **[10 hrs]**

2. Sequence alignment – scoring matrices- PAM and BLOSUM-Local and global alignment concepts-dynamic programming methodology- Needleman and Wunsch algorithm, Smith –Waterman algorithm – statistics of alignment score-Multiple sequence alignment –Progressive Alignment. Database searches for homologous sequences – Fasta and Blast versions. **[15 hrs]**

3. Fragment assembly, Genome sequence assembly – Gene finding methods:- content and signal methods-background of transform techniques – Fourier transformation and gene prediction – analysis and predictions of regulatory regions. **[10 hrs]**

4. Neural network concepts and secondary structure prediction Probabilistic models: Markov chain-random walk-Hidden Markov models. Gene identification and other applications. **[10 hrs]**

5. Evolutionary analysis: distances-clustering methods – rooted and unrooted tree representation – Bootstrapping strategies **[5 hrs]**

Text References:

1. Bioinformatics: a practical guide to the analysis of genes and proteins.

   A. Baxevanis and F.B.F. Ouellette(Eds.) John Wiley, New York (1998).

2. Introduction to computational biology: maps, sequences, and genomes.

   M.S. Waterman, Chapman and Hall, London (1995).

3. Proteome research: new frontiers in functional genomic

   M.R. Wilkins, K.L. Williams, R.D. Appel and D.H. Hochstrasser (Eds.), Springer, Berlin (1997).

EL-6. MODELLING OF BIOLOGICAL SYSTEM.

1. Concepts and principles of modeling. Limitations of models. **[20 hrs]**

Identifying the components of a process / system – variables, controlled variables, rate constants, relationships between variables. Writing a set of equations, describing a process or a system. Types of solutions – integration, equilibrium solutions, numerical solutions. Models involving space: spatial simulations.

2. Models of behavior. **[15 hrs]**

Foraging theory, Decision making – dynamic models, Game theory.

3. Modeling in Epidemiology and Public Health SIR models; Stochastic models and Spatial models. **[10 hrs]**

Text Reference:

1. Models in Biology.

   B. Brown and P. Rothary , John Wiley and Sons, New York.

2. Evolutionary Genetics

   J. M. Smith,. Oxford University Press, Oxford (1989).

EL - 7 ARTIFICIAL INTELLIGENCE

1. Conventional AI - Reasoning and Belief Systems

a) Logical Inference **[3 hrs]**

Reasoning Patterns in Propositional Logic, Propositional inference, Predicate calculus, Predicate and arguments, ISA hierarchy, Frame notation, resolution, Natural deduction etc.

b) Reasoning under Uncertainty **[3 hrs]**

Belief and uncertainty handling mechanisms, certainty, possibility and probability, Dempster Schaeffer theory , fuzzy inference, structure knowledge representation, semantic net, Frames, Script, Conceptual dependency etc.

c) Goal Driven Intelligence(Planning, Search and Perception) **[6 hrs]**

i) Planning: Formulation of Planning Problem, decomposition, representation of states, goals and actions, action schema, partial order planning, planning graphs Block world, strips, Implementation using goal stack, Non linear Planning with goal stacks, Hierarchical planning, List Commitment strategy.

ii) Game Playing and Search: Heuristic search techniques. Best first search, mean and end analysis, A* and AO* Algorithm, Minimize search procedure, Alpha beta cutoffs, waiting for Quiescence, Secondary search, ***Perception -*** Action, Robot Architecture, Vision, Texture and images, representing and recognizing scenes, waltz algorithm, Constraint determination, Trihedral and non trihedral figures labeling

d)Expert systems **[3 hrs]**

Utilization and functionality, architecture of expert system, knowledge and rule bases, rule chaining strategies, conflict resolution, RETE algorithm, uncertainty handling in expert systems

2. Intelligent Agents and Computational Intelligence

a) Agent Oriented Programming and Intelligent agents **[3 hrs]**

Agent oriented programming as a paradigm, Agent orientation vs. object orientation, autonomous and intelligent agents, "Agency" and Intelligence, logical agents, multi agent systems, planning, search and cooperation using agents.

b) Evolutionary Algorithms **[3 hrs]**

Evolutionary paradigms, genetic algorithms and genetic programming, Ant colonies and optimization, evolutionary search strategies.

c) Agents , internet and Softbots **[3 hrs]**

Interface agents and reactive systems , Softbots and info agents, the three layer model , process automation and agents,

d) AI paradigms from biological, physical and social sciences **[3 hrs]**

Swarms and collective intelligence, programming with swarms, fault tolerant systems, spin glasses and neural networks, self organizing systems, cellular automata and amorphous computing.

3. Statistical Learning Theory

a) Learning Theory **[3 hrs]**

Formulation of learning as a statistical problem, estimation of probability measure, empirical and structural risk minimization, Linear methods, supervised and unsupervised learning, regularization and kernel methods, model selection, inference and averaging, boosting and additive methods

b) Applications and algorithms **[3 hrs]**

Perceptrons and Neural Networks, Support vector machines, Classification and regression trees, nearest neighbors and EM clustering, Kohonen maps.

c) Text Mining and Natural language processing **[2 hrs]**

Sentence, syntax and semantic analysis, document classification, sentiment perception, theme and association mining

4. Hybrid Systems **[6 hrs]**

Integration of data driven and concept driven methodologies, Neural Networks and Expert Systems hybrid, neural networks and game tree search hybrid, evolutionary systems and supervised learning hybrid, neuro-fuzzy systems, genetic programming for rule induction

Text References:

1. AI: a modern approach

   Russell and Norvig:

2. AI

   Winston

3. Mathematical Methods in Artificial Intelligence

   Bender

4. Reasoning about Intelligent Agents

   Woolbridge

5. Artificial Intelligence.

   Elaine Rich and Kerin Knight:

6. Artificial Neural Network

   Kishen Mehrotra, Sanjay Rawika, K Mohan;.

EL-8 : QUALITY ASSURANCE AND SOFTWARE TESTING

1. Introduction to Software Quality Management Principles **[2 hrs]**

2. Software Quality Assurance & Quality Control **[3 hrs]**

What and how Software Quality, Quality Goals for Software, Process Quality Goal, Product Quality Goal, Quality policy and Quality Objectives: Linkage and control

3. Quality Models **[3 hrs]**

ISO 9001 – 2000 model, CMMI models, IS27000

4. Software Verification, Validation & Testing (VV & T) **[3 hrs]**

Understanding verification, validation & testing, Quality improvement through activities (like Reviews, Inspection, Walkthroughs, Testing), Process improvement through VV & T

5. Software Testing Principles and Concepts **[5 hrs]**

Software Testing Vocabulary, Testing and Quality, Who should Test, Independence in Testing, When Should Testing start?, Static versus Dynamic Testing, Testing and Debugging, The Cost of Quality, General Testing Process - Test Planning, Test Case Design, Test Case Execution, Test Analysis and Defect Reporting, Test Closure

6. Life Cycle Testing **[5 hrs]**

Various Software Development Models, Levels of Testing (Unit Testing, Integration Testing, System Testing, User Acceptance Testing), OO-oriented Testing, Model Based Testing, The "V" Model of Testing, Early Testing, Verification and Validation, Retesting and Regression Testing

7. Testing Techniques **[6 hrs]**

Static Testing Techniques (Reviews, Informal review, Walkthrough, Technical review, Inspection), Functional /Specification based Testing Techniques, Structural Testing Techniques, Experienced based techniques, Choosing test techniques, Test Oracle, Building Test Cases, Process for Building Test Cases, Test Case Execution, Recording Test Results, Problem Deviation, Problem Effect, Problem Cause, Use of Test Results

8. Test Reporting Process **[6 hrs]**

Prerequisites to Test Reporting, Define and Collect Test Status Data, Define Test Metrics used in Reporting, Define Effective Test Metrics, Test Tools used to Build Test Reports, Pareto charts, Cause and Effect Diagrams, Check Sheets, Histograms, Run Charts, Scatter Plot Diagrams, Regression Analysis, Multivariate Analysis, Control Charts, Test Tools used to Enhance Test

eporting, Benchmarking, Quality Function Deployment, Reporting Test Results, Current Status Test Reports, Final Test Reports, Guidelines for Report Writing

9. Test Management **[5 hrs]**

Testing in an Organization, Test management documentation, Test plan documentation, Test estimation and scheduling of Test Planning, Analyzing Testing metrics, Test progress monitoring and control, Testing and risk, Risk management, Software Configuration Management, Change Management

10. Test tools **[3 hrs]**

Types of Testing Tools and their use, Tool selection and implementation

Text References:

1. Software Testing Techniques: Finding the Defects that Matter (Programming Series)
   Michael Shannon, Geoffrey Miller, Richard, Jr. Prewitt,

2. Software Testing Fundamentals: Methods and Metrics
   M. Hutcheson

3. "Software Testing: A Craftsman's Approach, Second Edition,"
   Paul C Jorgensen,

   CRC Press, June 26, 2002.

4. "The Art of Software Testing," 2nd ed.,
   Glenford J. Myers,

   John Wiley & Sons, Inc.

5. "Lessons Learned in Software Testing: a Context-Driven Approach,"
   Cem Kaner, James Bach, and Bret Pettichord

   John Wiley & Sons, Inc.

EL-9 Soft Computing

1. Neural Networks **[15 hrs]**

a) Characterization

b) The brain, neural networks and computers

c) Neural networks and artificial intelligence

Background, Applications, Neural network software, Learning paradigms - Supervised learning, Unsupervised learning, Reinforcement learning, Learning algorithms

d) Neural networks and neuroscience

Types of models, Current research

e) History of the neural network analogy

2. Fuzzy Systems **[15 hrs]**

a) Antilock brakes

b) Fuzzy sets

Fuzzy control in detail, Building a fuzzy controller

c) History & applications

d) Logical interpretation of Fuzzy control

3. Evolutionary Computing **[15 hrs]**

a) Concept of Population genetics, probability, evolution principle

b) Genetic Algorithms or Evolutionary Strategies

i) Genetic Algorithms - General mechanism and terminologies, Selection, Crossover, Mutation

ii) Evolutionary Strategies - Two-membered Evolutionary strategy, Multi-member Evolutionary strategy, Recombination

c) Swarm Intelligence (one of the following three)

Ant colony optimization, Particle swarm optimization, Stochastic diffusion search

Applications

ASSIGNMENT:

To implement at least two of these major computational methods.

Text Reference:

1. Neural Networks, A Classroom Approach
   Satish Kumar

   Tata McGraw-Hill Publishing Company Limited

2. Artificial Neural Networks
   Kishan Mehrotra, Chilkuri K. Mohan, Sanjay Ranka
   Penram International Publishing (India)

3. Neural Networks, A Comprehensive Foundation
   Simon Haykin

   Pearson Education

4. Genetic Algorithms, in Search, Optimization & Machine Learning
   David E. Goldberg

   Pearson Education

5. Artificial Intelligence and Intelligent Systems
   N P Padhy

   OXFORD University Press

EL 10- DESIGN CONCEPTS AND MODELING

1. Introduction to design process. [6 hrs]

Building models suitable for the stages of a software development project. Introduction to UML.

2. Inception phase. [8 hrs]

Structured analysis, scenario structures.

3. Elaboration phase. [10 hrs]

Object modeling. Interfaces and abstraction. Information hiding.

4. Construction phase. [10 hrs]

Coupling and object interaction. Responsibilities, defensive programming and exceptions. Functional decomposition, module and code layout. Variable roles, object state, verification and assertions. Design patterns.

5. Transition phase. [10 hrs]

Inspections, walkthroughs, testing, debugging. Iterative development, prototyping and refactoring. Optimization.

Text Reference:

1. Code complete: a practical handbook of software construction.
   McConnell, S. (1993)

   Microsoft Press.

2. UML distilled. Addison-Wesley (2nd ed.).

   Fowler, M. (2000).