

TRIBHUVAN UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY

*

*

*

*

*

*

MASTER OF SCIENCE IN COMPUTER SCIENCE AND INFORMATION
TECHNOLOGY

(COURSE OF STUDY)

*

*

*

*

*

EFFECTIVE FROM ACADEMIC YEAR – 2071/2072

Tribhuvan University
Institute of Science and Technology

Course of Study
Master of Science in Computer Science and
Information Technology
(M.Sc. CSIT)
2014

Prepared by
Computer Science Subject Committee

Introduction:

The Master of Science in Computer Science and Information Technology (M.Sc. CSIT) curriculum is designed by closely following the courses practiced in accredited international universities, subject to the condition that the intake students are mostly from Bachelor of Science in Computer Science and Information Technology (B.Sc. CSIT) program of Tribhuvan University (TU). In addition to the standard core and elective Computer Science and Information Technology courses, the program offers several courses that provide knowledge for both research and development in computer science and information technology areas. The foundation and core courses are designed to meet the graduate program requirement, and the service courses are designed to meet the need of fast changing computer technology and application. Students enrolled in the graduate program are required to take courses in design and implementation of computer software systems, foundation in the theoretical model of computer science, and a functional background of computer hardware. All graduate students are required to complete at least 57 credit hours and they may complete maximum of 63 credit hours.

Institute of Science and Technology (IOST) is running this program since 2001. The previous curriculum was designed by focusing the curriculum of B.Sc. Computer Science (3-years) program of TU and after the introduction of B.Sc.CSIT (4-years) program in TU, this program has been redesigned specially focusing the B.Sc.CSIT (4-years) curriculum.

Objective:

The main objective of M.Sc. CSIT is to provide students depth knowledge and skill research and development areas in computer science and information technology including design, theory, programming and application of computers.

Admission Requirement:

The student entering the M.Sc. CSIT Program must have completed B.Sc. CSIT degree offered by TU or its equivalent. Prospective students can apply for admission by submitting a completed form as required by the general rule of the university. The students for admission are selected based on the scores in the entrance test conducted by the admitting college. The program also admits students having Bachelor of Engineering (B.E.) degree Computer, Electronics and Communication, and Electrical.

Course Structure:

Semester I

Course Title	Credit Hour	Full Marks
Advanced Operating Systems (C.Sc. 538)	3	75
Object Oriented Software Engineering (C.Sc. 539)	3	75
Algorithms and Complexity (C.Sc. 540)	3	75
Seminar I (C.Sc. 542)	1	25
Elective I	3	75
Elective II	3	75

List of Electives:

1. Neural Networks (C.Sc. 543)
2. Parallel and Distributed Computing (C.Sc. 544)
3. Algorithmic Mathematics (C.Sc. 545)
4. Advanced Computer Architecture (C.Sc. 546)

Semester II

Course Title	Credit Hour	Full Marks
Compiler Optimization (C.Sc. 558)	3	75
Web Systems and Algorithms (C.Sc. 559)	3	75
Seminar II (C.Sc. 560)	1	25
Elective III	3	75
Elective IV	3	75
Elective V	3	75

List of Electives:

1. Machine Learning (C.Sc. 561)
2. Computational Geometry (C.Sc. 562)
3. Advanced Database Concepts (C.Sc. 563)
4. Data Warehousing and Data Mining (C.Sc. 564)
5. Systems Programming (C.Sc. 565)

Semester III

Courses Title	Credit Hour	Full Marks
Principles of Programming Languages (C.Sc. 618)	3	75
Advanced Cryptography (C.Sc. 619)	3	75
Literature Review Research (C.Sc. 620)	2	50
Elective VI	3	75
Elective VII	3	75
Extra Elective I	3	75

List of Electives / Extra electives:

1. Fuzzy Systems (C.Sc. 621)
2. Embedded Systems (C.Sc. 622)
3. Image Processing and Pattern Recognition (C.Sc. 623)
4. Remote Sensing and GIS (C.Sc. 624)
5. Multimedia Computing (C.Sc. 625)

Semester IV

Course Title	Credit Hours	Marks
Genetic Algorithms (C.Sc. 665)	3	75
Dissertation (C.Sc. 666)	8	200
Extra Elective II	3	75

List of Extra Electives:

1. Information and Coding Theory (C.Sc. 667)
2. Cloud Computing (C.Sc. 668)
3. e-Government (C.Sc. 669)

Course Duration:

The entire course is of four semesters (two academic years). There is a separate semester examination after the end of each semester.

Hours of Instruction:

a) **Working days:** 70 days in a semester

b) **Class hours:**

- 3 credit hours courses with theory and labs is equivalent to 3 lecture hours and 3×4 lab hours = 15 working hrs per week.
- 3 credit hours theory-only course is equivalent 3 lecture hrs and 3×2 assignment hrs = 9 working hrs per week.

Assignment hrs include library work, home work, problem solving session, literature searching on the web etc

c) **Attendance:** 80 percent attendance in each paper is compulsory.

Examination:

Institute of science and technology, Tribhuvan University will conduct the final examination at the end of each semester. 60% weightage will be given to the final examination and 40% for the internal assessment to be conducted by the concerned college.

**FIRST
SEMESTER**

Advanced Computer Architecture

Course Title: Advanced Computer Architecture	Full Marks: 45 + 30
Course No: C.Sc. 546	Pass Marks: 22.5 + 15
Nature of the Course: Theory + Lab	Credit Hrs: 3

Course objectives: This course is designed to provide information of state-of-the-art high performance computer architectures. Topics include performance, ISA, instruction-level parallelism (ILP), Data level parallelism (DLP), thread-level parallelism (TLP), dynamic scheduling, out-of-order execution, register renaming, static scheduling (VLIW/EPIC), cache/memory hierarchy design, speculation techniques, advanced branch predictor design, multiprocessor, coherency issues, multicore processors, popular design case Studies, trends in architecture/microarchitecture development in face of physical design limit.

Course Prerequisite: Computer Organization and Operating System or the equivalent.

Course Contents:

Unit1: Fundamentals of Computer Design (10 Hrs)

Trends in Technology, Cost/Price Measuring and Reporting Performance, Benchmark suits, Principles of Quantitative Computer Design, Amdahl's Law, CPU Performance Equations, Instruction Set Architecture overview, Pipelining and its hazards. Case study: Performance, Price and Power.

Unit 2: Memory Hierarchy Design (10 Hrs)

Classification of cache organization, Cache hierarchy design, Quantifying Cache performance, measuring average memory access time, Cache optimizations techniques: basic to advanced techniques to reduce miss penalty, miss rate and hit time, trace caches, memory technology and optimizations, Main Memory Organization and optimization, virtual memory design. Case study: Memory Hierachies in the ARM Cortex-A8 and Intel Core i7.

Unit 3: Instruction Level Parallelism (12 Hrs)

Instruction Level Parallelism(ILP) overview, types of hazards, Data dependencies, Name dependencies, Control dependencies, instruction scheduling, Branch instruction costs, register renaming, basic compiler technique for exposing ILP: instruction scheduling and loop unrolling, Dynamic hardware prediction, Dynamic instruction scheduling, Hardware based Speculation, Multiple Issue and static scheduling, Superscalar and VLIW or EPIC architectures, Limitations of Instruction Level Parallelism, Multithreading: thread-level parallelism to improve uniprocessor throughput. Case Study: The Intel Core i7 and ARM Cortex-A8.

Unit 4: Data Level Parallelism(5 Hrs)

Data Level Parallelism (DLP) overview, vector processors architecture, SIMD Instruction Set Extensions for Multimedia, Graphics Processing Units, Detecting and Enhancing Loop-Level Parallelism. Case study: Mobile versus Server GPUs and Tesla versus Core i7.

Unit 4: Multiprocessor and Thread Level Parallelism (8 Hrs)

Multicore processors Architectures: SIMD, MIMD, Centralized Shared Memory, Distributed shared memory architecture, Cache Coherence, directory-based coherence, Synchronization, Memory Consistency models, Case Study: Multicore Processors and Their Performance.

Text Book:

1. Hennessy J.L., Patterson D.A., Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5th edition.

References:

1. Hennessy J.L., Patterson D.A., Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 4th edition.

2. David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware / Software Interface.

3. Computer architecture and embedded processor research articles and technical papers.

Advanced Operating Systems

Course Title: Advanced Operating Systems

Full Marks: 45 + 30

Course No: C.Sc. 538

Pass Marks: 22.5 + 15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course Description: Operating systems concepts overview - process management, memory management and storage management, protection and security, distributed system, real-time system, and multimedia system.

Course Objectives:

- Introduce the underlying principles of an operating system, virtual memory and resource management concepts.
- Exposure current state-of-art research in operating system
- Exposure of distributed operating system, real-time operating system and multimedia systems.

Course Contents

Unit 1: Process Management and Synchronization(12 Hrs)

- 1.1. Process management: Process Model, Modeling Multiprogramming, Inter-process communications: lock variables, sleep and wakeup, semaphores and mutexes.
- 1.2. Process scheduling: Round robin, *Weighted Round Robin*, *Round robin with variable quanta (Paper Based Study)*
- 1.3. Deadlocks: Deadlock detection, avoidance and prevention, *recovery from deadlocks (Paper Based study)*.

Unit 2: Memory Management (13 hrs)

- 2.1. Paging: Page tables, speeding up page tables, Page tables for large memories, demand Paging, Page replacement algorithms: Brief overview of LRU and LFU, Details of clock, WSclock, *LRFU*, *LIRS(Paper Based Study)*, Design Issues for paging, Segmentation, Segmentation with paging.
- 2.2. Storage Management: Free-space management, File system layout, Implementing files and directories, log structured file system, journaling file system, virtual file system, Disk scheduling, flash file systems. *high performance flash disks*, *Improving the performance of log structured file systems with adoptive block rearrangement*, *Buffer Cache Management scheme exploiting both temporal and spatial locality(Paper Based Study)*

Unit 3: Protection and Security (8 Hrs)

- 3.1. System Protection: protection principles and domain, access matrix and its implementation, access controls and rights, capability-based system, language-based protection.
- 3.2. System Security: Program threats, system and network threats, Cryptography as a Security Tool, User Authentication, Firewalling, Computer security classification, *Guarded Models for intrusion detection (Paper Based Study)*

Unit 4: Distributed Special Purpose Systems (12 Hrs)

- 4.1. Distributed Operating Systems: Types of Network based Operating Systems, Network structure and topology, Communication structure and protocols, Robustness, Design Issues, Distributed File Systems: Naming and Transparency, Remote File Access, Stateful versus Stateless Service, File Replication.
- 4.2. Real-Time Systems: Real time system characteristics and kernel features, Implementing real-time operating system, real-time CPU scheduling.
- 4.3. Multimedia Systems: Multimedia system overview, multimedia kernels, compression, CPU scheduling, disk scheduling.

Outcomes and Assessment

This course can be learnt in effective way only if we give focus in practical aspects of algorithms and techniques discussed in class. Therefore student should be able to simulate and analyze the algorithms by using any high level language. For this purpose student should simulate at least

- Two page replacement algorithms
- Two disk scheduling algorithms
- Deadlock recovery algorithm
- Two File System techniques

Recommended Book:

1. Silberschatz, Galvin and Gagne. Operating System Concepts, 8th Edition. John-Wiley.
2. Andrew S Tanenbaum, Modern Operating Systems, 3rd Edition 2007, PH1

Algorithmic Mathematics

Course Title: Algorithmic Mathematics

Full Marks: 45 + 30

Course No: C.Sc. 545

Pass Marks: 22.5 + 15

Nature of the Course: Theory

Credit Hrs: 3

Course Objectives: To solve the technical and scientific problems by using the theory of numerical, statistical, and optimal computational procedures along with practical applications.

- 1. Solution of Algebraic and Transcendental Equations (3 Hrs)**
Newton Raphson Method, Secant Method, Solution of systems of Nonlinear Equations (Newton Raphson Method)
- 2. Interpolation (5 Hrs)**
Errors in polynomial interpolation, Finite Differences, Differences of a polynomial, Newton's formulae for Interpolation, Bessel's Formula, Everett's Formula, Relation between Bessel's and Everett's Formulae, Lagrange's Interpolation Formula
- 3. Curve Fitting, B-Splines and Approximation (3 Hrs)**
Least-squares Curve Fitting Procedures (Linear, Quadratic and Exponential), B-splines, Approximation of Functions
- 4. Numerical Differentiation and integration (4 Hrs)**
Numerical Differentiation, Trapezoidal Rule, Simpson's 1/3 -Rule, Simpson's 3/8 -Rule, Volume calculation, Newton-Cotes Integration Formulae, General Quadrature Formula, Gaussian Integration
- 5. Matrices and Linear Systems of Equations (3 Hrs)**
Solution of Linear Systems- Direct Methods (Gauss Jordan), Solution of Linear Systems- Iterative Method (Gauss-Seidel), Eigen value Problem (Eigen Value, Eigen Vector)
- 6. Numerical Solution of Ordinary Differential Equations (6 Hrs)**

Solution of Taylor's Series, Euler's Method (Modified Method), Predictor-Corrector Methods, Simultaneous and Higher Order Equations (4th order Runge Kutta Method), Boundary Value Problems (Finite Difference Method)

- 7. Numerical Solution of the Partial Differential Equations (3 Hrs)**
Finite- Difference Approximations to derivatives, Laplace's Equation, Parabolic Equations, Hyperbolic Equations, Iterative Methods for solution of Equations

- 8. Introduction and Descriptive Statistics (2 Hrs)**
An overview of probability and statistics, Pictorial and tabular methods in descriptive statistics, Measures of central tendency, dispersion, and direction, Joint and conditional probabilities, Central limit theorem

- 9. Discrete Random Variables and Probability Distributions (4 Hrs)**
Random variables, Probability distributions for random variables, Expected values of discrete random variables, The binomial probability distribution, Hypothesis testing using the binomial distribution, The Poisson probability distribution

- 10. Hypothesis Testing Procedures (3 Hrs)**
Tests about the mean of a normal population, The t-test, Test procedures for a population variance, Z-tests for differences between two populations means, The two-sample t-test, A confidence interval for the mean of a normal population

- 11. Optimization Techniques (5 Hrs)**
The simplex method, objective function and constraint conditions, changing inequalities to equalities, the conical form, of solution, optimal values of variables, Integer programming, Dynamic programming

- 12. Transformation (4 Hrs)**
Laplace transform, Fourier transform, Discrete Fourier transform, Fast Fourier transform, Z transform and their inverse transform

References:

1. E. Kreyszig, "Advanced Engineering Mathematics", Wiley-Eastern Publication

2. Jay L. Devore, "Probability and Statistics for Engineering and the Sciences", Brooks/Cole Publishing Company, Monterey, California, 1982.
3. Introductory methods of Numerical analysis, S.S. Sastry
4. An Introduction to numerical computations, S. Yakowitz and F. Szidarovszky
5. Numerical Methods, Dr. V.N. Vedamurthy, Dr. N. Ch. S.N . Iyengar
6. Numerical Methods , E. Balagurusamy

Algorithms and Complexity

Course Title: Algorithms and Complexity

Full Marks: 45+30

Course Noe: C.Sc. 540

Pass Marks: 22.5+15

Credit Hours = 3

Nature of the course: Theory +Lab

Course Description

The course can be viewed as a continuation of Design and Analysis of Algorithms, and its goal is to cover basic algorithmic techniques that are not covered in the subject. In particular, we will discuss NP Completeness in detail, approximation algorithms for NP-hard problems, Randomized Algorithms, online algorithms, PRAM algorithms, Mesh Algorithms and Hypercube Algorithms.

Course Objectives

The purpose of this course is to present depth in optimization paradigms, Some advance algorithm design techniques and moderate level understanding in computational complexity theory.

Prerequisites

Linear Algebra, Data Structure and algorithm, Basic Course in Algorithm design, Programming language

Course Contents

Unit 1: Advance Algorithm Analysis and Design Techniques 10 hr

- 1.1. **Advanced Algorithm Analysis Techniques:** Amortized Analysis(Aggregate Analysis, Accounting Method, Potential method), Probabilistic Analysis, Las Vegas and Monte Carlo Algorithms.
- 1.2. **Advance Algorithm Design Techniques:** Greedy Algorithms (Tree Vertex Splitting, Job Sequencing with deadlines), Dynamic Programming(Greedy vs Dynamic, String Editing, Optimal BST), backtracking(Sum of Subsets, Knapsack Problem), Randomized Algorithms(Identifying the repeated elements, Primality Testing, Karger's algorithm)

Unit 2: Computational Complexity Theory 10 hr

- 2.1 **Basic Concepts:** Complexity Theory, Complexity Classes: P, NP, NP-Hard and NP-Complete, Decision Problems and Language Recognition Problems
- 2.2 **Problem Reduction:** Reduction, Polynomial time reduction, Cooks Theorem, Proving NP-Completeness: Formula Satisfiability, 3SAT, CLIQUE, Vertex Cover, Hamiltonian Cycle, TSP, Subset Sum.
- 2.3 **NP-Hard Code Generation Problems:** Code Generation with Common Subexpression, Implementing Parallel Assignment Instructions
- 2.4 **Coping with NP-Completeness:** Performance ratios for approximation algorithms, Approximated Algorithms: Vertex Cover, TSP, Set Covering, Subset Sum

Unit 3: Online and PRAM Algorithms 12 hr

- 3.1 **Online Algorithms:** Introduction, Ski Problem, Load Balancing, Paging and Caching: Last-in First-out (LIFO), Longest Forward Distance (LFD), Least Recently Used (LRU)
- 3.2 **PRAM Algorithms:** Introduction, Computational Model, Fundamental techniques and Algorithms (Prefix Computation, List Ranking), Selection (Maximal Selection with n^2 Processors, Finding Maximum Using n Processors), Merging (A Logarithmic Time Algorithm, Odd-Even Merge), Sorting (Odd-Even Merge Sort, Preparata's Algorithm).

Unit 4: Mesh and Hypercube algorithms 13hr

- 4.1 **Mesh Algorithms:** Computational Model, Packet Routing (Packet Routing on a Linear Array, A Greedy Algorithm for PPR on a Mesh), Fundamental Algorithms (Broadcasting, Prefix Computation, Data Concentration), Selection (Randomized Algorithm for $n=p$, Randomized Selection for $n>p$), Sorting (Sorting on a Linear Array, Sorting on a Mesh)
- 4.2 **Hypercube Algorithms:** Computational Model (Hypercube, Butterfly Network, Embedding of other Networks, PPR Routing (Greedy algorithm, Randomized Algorithm), Fundamental Algorithms (Broadcasting, Prefix Computation, Data Concentration), Selection (Randomized Algorithm for $n=p$, Randomized Selection for $n>p$), Sorting (Odd-Even Merge Sort, Bitonic Sort)

Textbook

- 1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, **Computer Algorithms/C++**, 2nd Edition, Universities Press, 2007

Reference Books

- 1. H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, **Introduction to Algorithms**, 2nd Edition, MIT Press, 2001

2. G. Brassard and P. Bratley, **Fundamentals of Algorithmics**, Prentice-Hall, 1996

Outcomes and Assessment

Student should implement Algorithms described in class and should perform their Empirical Evaluation.

Neural Networks

Course Title: Neural Networks

Full Marks: 45 + 30

Course No: C.Sc. 543

Pass Marks: 22.5 + 15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course Description:

This course deals with resembling the conscious behavior of brain in an artificially connected logical nodes that can learn itself in a supervised and unsupervised environment. It introduces linear algebra for creating and processing input patterns to generate output. It also emphasizes on probabilistic model for classification and prediction within a tolerable and significant limit.

Unit 1: Introduction to artificial neural networks

5 hrs

Biological neural networks; Pattern analysis tasks: Classification, Regression, Clustering; Computational models of neurons; Structures of neural networks; Learning principles

Unit 2: Linear models for regression and classification

7 hrs

Polynomial curve fitting; Bayesian curve fitting; Linear basis function models; Bias-variance decomposition; Bayesian linear regression; Least squares for classification; Logistic regression for classification; Bayesian logistic regression for classification

Unit 3: Feedforward neural networks

7 hrs

Pattern classification using perceptron; Multilayer feedforward neural networks (MLFFNNs); Pattern classification and regression using MLFFNNs; Error backpropagation learning; Fast learning methods: Conjugate gradient method; Autoassociative neural networks; Bayesian neural networks

Unit 4: Radial basis function networks

8 hrs

Regularization theory; RBF networks for function approximation; RBF networks for pattern classification

Unit 5: Kernel methods for pattern analysis

8 hrs

Statistical learning theory; Support vector machines for pattern classification; Support vector regression for function approximation; Relevance vector machines for classification and regression

Unit 6: Self-organizing maps

5 hrs

Pattern clustering; Topological mapping; Kohonen's self-organizing map

Unit 7: Feedback neural networks

5 hrs

Pattern storage and retrieval; Hopfield model; Boltzmann machine; Recurrent neural networks

Text Books:

1. B.Yegnanarayana, Artificial Neural Networks, Prentice Hall of India, 1999
2. Satish Kumar, Neural Networks – A Classroom Approach, Tata McGraw-Hill, 2003
3. S.Haykin, Neural Networks – A Comprehensive Foundation, Prentice Hall, 1998
4. C.M.Bishop, Pattern Recognition and Machine Learning, Springer, 2006

Object-Oriented Software Engineering

Course Title: Object-Oriented Software Engineering
Course No: C.Sc. 539
Nature of the Course: Theory + Case Studies+ Project

Full Marks: 45 + 30
Pass Marks: 22.5 + 15
Credit Hrs: 3

Course Objectives:

- This course aims to give students both a theoretical and a practical foundation in Object - Oriented software engineering.
- In the theoretical part, students will learn about the principles and methods of Object Oriented software engineering, including current and emerging Object Oriented software engineering practices and support tools.
- In the practical part, students will become familiar with the development of Object Oriented software products from an industry perspective.

Unit 1: 4 Hrs

Software life cycle models, Requirement analysis and specification, object oriented software development,

Unit 2: 5 Hrs

Introduction to object-orientation, object- oriented system development-function /data methods, object oriented analysis, construction and testing, object -oriented programming with examples

Unit 3: 10 Hrs

Architecture-model architecture, requirements, analysis, design, implementation and test model, analysis, construction, Real-time-classification of real time systems, database-RDBMS, Object DBMS, components-use of components, component management, testing-on testing, Unit, integration, system and the testing process

Unit 4: 6 Hrs

Managing object-oriented software engineering, Project selection and preparation, product development organization, project organization and management, project staffing, software quality assurance, software metrics

Unit 5: 5 Hrs

Object oriented analysis and design, hierarchical object-Oriented design, object Modeling technique and responsibility-driven design

Unit 6: 15 Hrs

Case studies and project
Warehouse management system, Telecom

References

1. Ivar Jacobson-Object-Oriented software engineering
2. Ian Sommerville-Software Engineering
3. Grady Booch-Object-oriented analysis and design

Parallel and Distributed Computing

Course Title: Parallel and Distributed Computing

Course No: C.Sc. 544

Nature of the Course: Theory + Lab

Full Marks: 45 + 30

Pass Marks: 22.5 + 15

Credit Hrs: 3

Course Description:

This course covers a broad range of topics related to parallel and distributed computing, including foundations, models, and algorithms.

Course Contents:

Unit 1: Foundations

1.1. Parallel and Distributed Computing: the Scene, the Props, the Players (2 hrs)

A Perspective, Parallel Processing Paradigms, Modeling and Characterizing Parallel Algorithms, Cost vs. Performance Evaluation, Software and General Purpose PDC

1.2. Semantics of Concurrent Programming (2 hrs)

Models of Concurrent Programming, Semantic Definitions, Axiomatic Semantic Definitions, Denotational Semantic Definitions, Operational Semantic Definitions

1.3. Formal Methods: A Petri Nets Bases Approach (2 hrs)

Process Algebras, PETRI Nets, High-Level New Models

1.4. Complexity Issues in Parallel and Distributed Computing (3 hrs)

Introduction, Turing Machines as the Basis and Consequences, Complexity Measures for Parallelism, Parallel Complexity Models and Resulting Classes, VLSI Computational Complexity, Complexity Measures for Distributed Systems, Neural Networks and Complexity Issues

1.5. Distributed Computing Theory (3 hrs)

The Computational Model, A Simple Example, Leader Election, Sparse Network Covers and Their Applications, Ordering of Events, Resource Allocation, Tolerating Processor Failures in Synchronous Systems, Tolerating Processor Failures in Asynchronous Systems, Other Types of Failures, Wait-Free Implementations of Shared Objects

Unit 2: Models

2.1. PRAM Models (5hrs)

Introduction; Techniques for the Design of Parallel Algorithms; The PRAM Model; Optimality and Efficiency of Parallel Algorithms; Basic PRAM Algorithms; The NC-Class; P-Completeness: Hardly Parallelizable Problems; Randomized Algorithms and Parallelism; List Ranking Revisited: Optimal $O(\log n)$ Deterministic List Ranking; Taxonomy of Parallel Algorithms; Deficiencies of the PRAM Model

2.2. Broadcasting with Selective Reduction: A Powerful Model of Parallel Computation (3 hrs)

Introduction; A Generalized BSR Model; One Criterion BSR Algorithms; Two Criteria BSR Algorithms; Three Criteria BSR Algorithms; Multiple Criteria BSR Algorithms

2.3. Dataflow Models (3 hrs)

Kinds of Dataflow; Data-Driven Dataflow Computing Models; Demand-driven Dataflow Computing Models; Unifying Data-Driven and Demand-Driven

2.4. Partitioning and Scheduling (4 hrs)

Program Partitioning; Task Scheduling; Scheduling System Model; Communication Models; Optimal Scheduling Algorithms; Scheduling Heuristic Algorithms; Scheduling Nondeterministic Task Graphs; Scheduling Tools; Task Allocation; Heterogeneous Environments

2.5. Checkpointing in Parallel and Distributed Systems (3 hrs)

Introduction; Checkpointing Using Task Duplication; Techniques for Consistent Checkpointing

2.6. Architecture for Open Distributed Software Systems (2 hrs)

Introduction to Open Distributed Systems Architecture; Computational Model; Engineering Model; ODP Application

Unit 3: Algorithms

3.1. Fundamentals of Parallel Algorithms (3 hrs)

Introduction; Models of Parallel Computation; Balanced Trees; Divide and Conquer; Partitioning; Combining

3.2. Parallel Graph Algorithms (5 hrs)

Graph-Theoretic Concepts and Notation; Tree Algorithms; Algorithms for General Graphs; Algorithms for Particular Classes of Graphs

3.3. Data Parallel Algorithms (5 hrs)

Chapter Overview; Machine Model; Impact of Data Distribution; CU/PE Overlap; Parallel Reduction Operations; Matrix and Vector Operations; Mapping Algorithms onto Partitionable Machines; Achieving Scalability Using Set of Algorithms

Recommended Books:

1. Parallel and Distributed Computing Handbook, Albert Y. Zomaya, Editor, McGraw-Hill
2. Introduction to Parallel Computing; Ananth Grama, Anshul Gupta, George Kaprypis, Vipin Kumar; Pearson Education
3. An Introduction to Parallel Programming, Peter Pacheco
4. Quinn, Michael J., Parallel Programming in C with MPI and OpenMP, McGraw-Hill, 2004.
5. Quinn, M., Parallel Computing: Theory and Practice, McGraw Hill, 1994.

Seminar I

Course Title: Seminar I
Course No: C.Sc. 542
Nature of the Course: Seminar

Full Marks: 25
Pass Marks: 12.5
Credit Hrs: 1

Course Description: The seminar-I is of full marks 25 offered in the curriculum of the M. Sc. first year first semester. A student pursuing the seminar prepares a seminar report and presents the seminar in the department. Once accepted by the department, the students have to submit the final copy of the report.

Introduction:

Each student is required to write a comprehensive report about the seminar. The report should consist of 5 to 10 pages describing the topic selected. Students can choose the seminar topics of their relevant subject area. The students are suggested to select the research oriented topics rather than just informative ones. The report should be in the format as described below;

Arrangement of Contents:

The sequence in which the seminar report material should be arranged and bound should be as follows:

1. Cover Page & Title Page
2. Abstract
3. Chapters:
 - a. Introduction
 - b. Previous Works, Discussions and Findings
 - c. Conclusion
4. References

Format of References

1. References

A list of all publications (articles, texts, monographs, etc.) must be supplied as the last section of the paper. Each article or paper used must be listed alphabetically by last name of the author and the list must be numbered sequentially. The following are examples of the format for various types of entries in the list.

Journal: Stalling, W., RSA and its computational aspects, *Infoworld*, 12, 28 (Jul. 2012), 42-49.

Book: Bishop, M. and Boneh, D., *Elements of Computer Security*, Pearson Education., 2009.

Proceedings: Shamir, A., Controlling attacks on public key cryptography. *Proceedings of OOPSLA 86* (Sept. 1986., Portland), 405-416.

Articles Which Have Established Citation Pages at ACM (Electronically published articles): Smith, J., An algorithm for the traveling salesman problem, *Journal of the ACM* 54 (June 2011), 234-245. (<http://www.acm.org/jacm/2011/SmithtspAlgo/>)

2. Citations

Whenever material from a publication is used in the paper it must be followed by a citation which is simply the number of the reference in the list of references enclosed in square brackets (for example, a reference to the third article listed in the list of references would contain the citation [3].) Multiple citation numbers can be incorporated within one citation when required (for example, references to the fourth, eighth, and eleventh entries in the reference list would appear as [4, 8, 11]).

Text of the seminar paper format:

1. The paper can be prepared using a word processor or LATEX. The students are highly recommended to use LATEX.
2. Margins - All margins must be one inch.
3. The text must be spaced by 1.5.
4. The text must be typed in 12 point font. The text must be typed in Times New Roman font.

**SECOND
SEMESTER**

Advanced Database Concepts

Course Title: Advanced Database Concepts

Course No: C.Sc. 563

Nature of the Course: Theory + Lab

Full Marks: 45 + 30

Pass Marks: 22.5 + 15

Credit Hrs: 3

Course Objectives: To study the further advanced database techniques beyond the fundamental database techniques which were covered in the graduate level course, and thus, to acquaint the students with some relatively advanced issues. At the end of the course students should be able to gain an awareness of the basic issues in file organizations, indexes, query processing and optimization, database security and authorization, objected oriented, and object relational databases, and distributed, and client server architectures.

Unit 1: File Organizations and Indexes

8 Hrs

Introduction; Secondary Storage Devices; Buffering Blocks; Placing File Records on Disk; Operations on Files; Files of Unordered Records; Files of Ordered Records; Hashing Techniques; Other Primary File Organizations; Parallelizing Disk Access Using RAID Technology; Storage Area Networks; Indexing Structures for Files

Unit 2: Algorithms for Query Processing and Optimization

8 Hrs

Translating SQL Queries into Relational Algebra; Algorithms for External Sorting; Algorithms for SELECT and JOIN Operations; Algorithms for PROJECT and SET Operations Implementing Aggregate Operations and Outer Joins; Combining Operations Using Pipeline; Using Heuristics in Query Optimization; Using Selectivity and Cost Estimates in Query Optimization; Semantic Query Optimization

Unit 3: Database Security and Authorization

8 Hrs

Introduction to Database Security Issues; Discretionary Access Control Based on Granting and Revoking Privileges; Mandatory Access Control and Role-Based Access Control for Multilevel Security; Introduction to Statistical Database Security; Introduction to Flow Control; Encryption and Public Key Infrastructure

Unit 4: Object-Oriented Databases

8 Hrs

Overview of Object-Oriented concepts, Object identity, Object structure, and type constructors, Encapsulation of operations, Methods, and Persistence, Type hierarchies and Inheritance, Type extents and queries, Complex objects; Database schema design for OODBMS; OQL, Persistent programming languages; OODBMS architecture and storage issues; Transactions and Concurrency control, Example of ODBMS

Unit 5: Object Relational and Extended Relational Databases

8 Hrs

Database design for an ORDBMS - Nested relations and collections; Storage and access methods, Query processing and Optimization; an overview of SQL3, Implementation issues for extended type; Systems comparison of RDBMS, OODBMS, ORDBMS

Unit 6: Parallel and Distributed Databases and Client-Server Architecture

8 Hrs

Architectures for parallel databases, Parallel query evaluation; Parallelizing individual operations, Sorting, Joins; Distributed database concepts, Data fragmentation, Replication, and allocation techniques for distributed database design; Query processing in distributed databases; Concurrency control and Recovery in distributed databases. An overview of Client-Server architecture

References:

1. Elmasri and Navathe, Fundamentals of Database Systems [4e], Pearson Education
2. Raghu Ramakrishnan, Johannes Gehrke, Database Management Systems [3e], McGraw-Hill
3. Korth, Silberchatz, Sudarshan , Database System Concepts, McGraw-Hill.
4. Peter Rob and Coronel, Database Systems, Design, Implementation and Management, Thomson Learning.
5. C.J.Date, Longman, Introduction to Database Systems, Pearson Education

Compiler Optimization

Course Title: Compiler Optimization

Full Marks: 45 + 30

Course No: C.Sc. 558

Pass Marks: 22.5+15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course Description:

Theoretical and practical aspects of building optimizing compilers that effectively exploit modern architectures. The course will begin with the fundamentals of compiler optimization, and will build upon these fundamentals to address issues in state-of-the-art commercial and research machines. Topics include: dependence analysis, control flow analysis, redundancy elimination, loop optimizations, interprocedural analysis. Students will implement significant optimizations within the framework of a modern research compiler.

Course Objectives:

This course has following objectives:

- To provide students with the background knowledge necessary to understand the process of constructing an optimizing compilers.
- To review common analysis and code transformation techniques used in optimizing compilers.
- To help prepare students to do research in compilers.

Unit1: Review of Compiler Structure and Modern Architecture

5 Hrs

Review of compiler structure, Importance of code optimization, Structure of optimizing compilers, Compiler challenges for modern architectures: Instruction pipeline, compiling multiple issue processors, Processor parallelism.

Unit2: Dependence Analysis and Testing

8 Hrs

Fundamentals of dependence, Dependence in loops, Types of dependences, Direction vectors and distance vectors, Fundamental theorem of parallelization, Testing for dependence: separable, SIV, MIV and GCD testing, Construction of direction and distance vectors.

Unit3: Preliminary Transformations

6 Hrs

Loop normalization, Data flow analysis: Dead-code elimination, Constant propagation, SSA form, Induction variable substitution: Forward expression, Induction-variables, substitution process.

Unit4: Loop Optimization

8 Hrs

Loop interchange and vectorization, Loop skewing, Scalar and array expansion, Array renaming, Node splitting, Index-set splitting, Forward substitution, Alignment, Code replication, Loop fusion, Nested loops, Parallel code generation and its problems.

Unit5: Control Dependence

8 Hrs

If conversion: Branch classification, Forward, Exit and Backward branches, Iterative dependences, Control dependence: Control dependence in loops, Application of control dependence to parallelization, Program dependence graph.

Unit6: Interprocedural Analysis and Optimization

10 Hrs

Side effect analysis, Constant propagation and alias analysis, Flow-insensitive and flow-sensitive problems, Side effects to arrays, Inline substitution, Linkage tailoring and procedure cloning, Management of interprocedural analysis and optimization.

Recommended Books:

Textbook:

Allen and Kennedy, Optimizing Compilers for Modern Architectures, Morgan-Kaufmann

Reference:

Advanced Compiler Design Implementation, Steven S. Muchnick, Morgan-Kaufmann

Computational Geometry

Course Title: Computational Geometry

Full Marks: 45+30

Course No: C.Sc. 562

Pass Marks: 22.5+15

Credit Hours: 3

Nature of the course: Theory +Lab

Course Description: Computational Geometry is the study of the representation and storage of geometric data and relationships, and the design & implementation and analysis of computational algorithms that operate on geometric data to answer questions of practical interest. In this course fundamental algorithms in computational geometry will be covered. Covered topics include;

Elementary geometric objects, line segment intersection, triangulation of polygons, polygon partitioning, convex hull computation in 2D and 3D, proximity problems: Voronoi diagrams , motion path planning, mesh generation, range search

Course Objectives: Introduce the development of algorithmic tools for solving problems having geometric flavor. Apply such tools for solving problems related to computer graphics, geographic information systems (GIS), robotics, and others---in which geometric algorithms play a fundamental role.

Course Content:

Unit1: Arrangements of geometric objects and spatial decomposition

1.1 Representation of points , Lines, Line segments, Ray, Polygon, Visibility inside polygon, Triangulation dual, Mouth and Ear of a polygon, Meisters two ear theorem, Coloring of polygon, Polygon triangulation, Art gallery theorem, Notion of turn tests, Computing area of a polygon, Segment Intersection Computation and detection of segment intersection, Plane-sweep approach for Segment Intersection

6hrs

1.2 Diagonal tests, Polygon triangulation: Linear time triangulation, Monotone partitioning, Trapezodalization, Plan-sweep approach for monotone partitioning, intersection of polygons, Convex Partitioning Problem (CPP), Essential diagonals, Hertel's Mehlhorns Algorithm for CPP, Optimal convex partitioning

7hrs

Unit2: Convex Hull Problems

2.1 Convex hull computation in 2D: convex hull of point sets, naïve algorithm, gift-wrap method, Graham scan algorithm, divide and conquer, quick hull approach, incremental algorithm **8hrs**

2.2 Convex hull computation in 3D: Notion of planar graphs, Doubly Connected Edge List, Euler Theorem, box enclosing problem, gift-wrap algorithm, divide and conquer approach, incremental algorithm. **4hrs**

Unit3: Proximity Problems and Motion Planning

3.1 Voronoi polygon and Voronoi diagram, Delaunay Triangulation, Relationship between Delaunay triangulation and Voronoi diagrams, Construction of Voronoi diagram, Largest Empty Circle Problem, Nearest Neighbor problem, Computing closest pair of points, using Voronoi diagram to compute closest pairs. **7hrs**

3.2 Range Search, 1D-Range Searching querying, Kd-trees, Range Trees, Higher Dimensional Range Trees, **4 hrs**

3.3 Robot Motion Planning, Path planning and motion planning in 2D, visibility graphs, computing shortest paths by using visibility graphs, object growing, Minkowski Sum, Use of Minkowski Sum for growing obstacles, Motion planning of point robots, disc robots and polygonal robots. **5hrs**

Unit 4 : Mesh

4.1 Mesh and its types, Algorithms for generation two dimensional mesh, application of Delaunay triangulation and quad-tree methods for generating meshes, mesh conversion, mesh refinements, **4hrs**

Recommended Books:

1. J.O' Rourke, *Computational Geometry in C*, Cambridge University Press.
2. Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars, *Computational Geometry: Algorithms and Applications*, Springer Berlin Heidelberg
3. Handbook of Computational Geometry, JR Sack, J. Urrutia, Elsevier.

Data Warehousing and Data Mining

Course Title: Data Warehousing and Data Mining

Full Marks: 45 + 30

Course No: C.Sc. 564

Pass Marks: 22.5 + 15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course Objectives: To provide an overview of the techniques and development on data warehousing and data mining. It focuses on providing information regarding establishment of data warehouse and Online Analytical Processing (OLAP). It introduces broad research areas for further development.

Unit- I [5 Hrs.]

The Evolution Of Data Warehousing (The Historical Context), The Data Warehouse - A Brief History, Today's Development Environment. Principles Of Data Warehousing (Architecture And Design Techniques) : Types of Data and Their Uses, Conceptual Data Architecture, Design Techniques, introduction to the Logical Architecture.

Unit- II [5 Hrs.]

Creating the Data Asset: Business Data Warehouse Design, Populating the Data warehouse, Unlocking the Data Asset for End Users (The Use of Business Information) : Designing, Business Information Warehouses, Populating Business information Warehouses, User Access to Information, Information Data in Context.

Unit- III [4 Hrs.]

Implementing The Warehouse (Managing the Project and Environment): Obstacles to Implementation, Planning your Implementation, Justifying the Warehouse, Organization Implications of Data Warehousing, The data Warehouse in your Organization, Data Warehouse Management, Looking to the Future.

Unit- IV [6 Hrs.]

Differences between Operational Database Systems and Data Warehouses, a multidimensional Data Model, Data warehouse and OLAP technology, multidimensional data models and different OLAP operations, OLAP Server: ROLAP, MOLAP and HOLAP. Data warehouse implementation, efficient computation of data cubes, processing of OLAP queries, indexing OLAP data.

Unit- V [3 Hrs.]

Data Mining Primitives, Languages, and System Architectures, graphical user interfaces. Concept Description: Characterization and Comparison, Data generalization and summarization-based characterization, Analytical characterization, analysis of attribute relevance, mining class comparisons, and mining descriptive statistical measures in large databases.

Unit- VI [6 Hrs.]

Mining Association Rules in Large Databases, Mining single-dimensional Boolean association rules from transactional databases, mining multilevel association rules from transaction databases, Mining multidimensional association rules from relational databases and data warehouses, From association mining to correlation analysis, Constraint-based association mining.

Unit- VII [6 Hrs.]

Classification and prediction, issues, classification by decision induction, Bayesian classification, classification by back propagation, classification based on concepts from association rule mining other classification methods.

Unit- VIII [6 Hrs.]

Cluster Analysis Introduction : Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Outlier Analysis.

Unit- IX [4 Hrs.]

Mining Complex Types of Data: Multi-Dimensional Analysis and Descriptive Mining of Complex Data Objects, Mining Spatial databases, Mining Multimedia databases, Mining Time-Series and Sequence data, Mining Text databases, mining the World Wide Web.

Text Books:

1. Data Mining Concepts and Techniques, Morgan Kaufmann J. Han, M Kamber Second Edition ISBN: 978-1-55860-901-3
2. Data Warehousing in the Real World – Sam Anahory and Dennis Murray, Pearson Edition Asia.

References:

1. Data Mining, Alex Berson, Stephen Smith, Korth Theorling, TMH.
2. Data Mining, Adriaans, Addison-Wesley Longman.
3. Data Mining and Warehousing, Chanchal Singh, Wiley.
4. Data Mining, John E, Herbert P.
5. Data Mining Techniques – Arun K Pujari, University Press.

Machine Learning

Course Title: Machine Learning
Course No: C.Sc.561
Nature of the Course: Theory + Lab

Full Marks: 45 + 30
Pass Marks: 22.5+15
Credit Hrs: 3

Course Description

This course provides a broad introduction to machine learning and statistical pattern recognition. Topics include: supervised learning (generative/discriminative learning, parametric/non-parametric learning, neural networks, support vector machines); unsupervised learning (clustering, dimensionality reduction, kernel methods); learning theory (bias/variance tradeoffs; VC theory; large margins); reinforcement learning and adaptive control.

Course Objective

Purpose of this course is to present different machine learning techniques and also analyze their pros and cons. In addition to this, this course also concepts on learning theory and their applications

Prerequisites

Computer Programming, Probability Theory, Linear Algebra

Unit 1: Introduction

5 hrs

The Motivation & Applications of Machine Learning, The Definition of Machine Learning, The Overview of Supervised Learning, The Overview of Learning Theory, The Overview of Unsupervised Learning, The Overview of Reinforcement Learning

Unit 2: Supervised learning

12 hrs

Application of Supervised Learning, Linear Regression, Gradient Descent, Batch Gradient Descent, Stochastic Gradient Descent (Incremental Descent), Matrix Derivative Notation for Deriving Normal Equations, Derivation of Normal Equations, The Concept of Underfitting and Overfitting, The Concept of Parametric Algorithms and Non-parametric Algorithms, Locally Weighted Regression, The Probabilistic Interpretation of Linear Regression, The motivation of Logistic Regression, Supervised learning setup, Least Mean squares, Logistic Regression, Perceptron Learning Algorithm, Discriminative Algorithms, Generative Algorithms, Gaussian Discriminant Analysis (GDA), GDA and Logistic Regression, Naive Bayes, Laplace Smoothing,

Intuitions about Support Vector Machine (SVM), Notation for SVM, Functional and Geometric Margins, Optimal Margin Classifier, Lagrange Duality, Karush-Kuhn-Tucker (KKT) Conditions, SVM Dual, The Concept of Kernels, Mercer's Theorem, Non-linear Decision Boundaries and Soft Margin SVM, Coordinate Ascent Algorithm, The Sequential Minimization Optimization (SMO) Algorithm, Applications of SVM

Unit 3: Learning theory

9 hrs

Bias/variance Tradeoff, Empirical Risk Minimization (ERM), The Union Bound, Hoeffding Inequality, Uniform Convergence - The Case of Finite H, Sample Complexity Bound, Error Bound, Uniform Convergence Theorem & Corollary, Uniform Convergence - The Case of Infinite H, The Concept of 'Shatter' and VC Dimension, SVM Example, Model Selection, Cross Validation, Feature Selection, Bayesian Statistics and Regularization, Online Learning, Advice for Applying Machine Learning Algorithms, Debugging/fixing Learning Algorithms, Diagnostics for Bias & Variance, Optimization Algorithm Diagnostics, Diagnostic Example Error Analysis, Getting Started on a Learning Problem

Unit 4: Unsupervised learning

9 hrs

The Concept of Unsupervised Learning, K-means Clustering Algorithm, K-means Algorithm, Mixtures of Gaussians and the EM Algorithm, Jensen's Inequality, The EM Algorithm, Mixture of Gaussian, Mixture of Naive Bayes - Text clustering (EM Application), Factor Analysis, Restrictions on a Covariance Matrix, The Factor Analysis Model, EM for Factor Analysis, The Factor Analysis Model, EM for Factor Analysis, Principal Component Analysis (PCA), PCA as a Dimensionality Reduction Algorithm, Applications of PCA, Face Recognition by Using PCA, Latent Semantic Indexing (LSI), Singular Value Decomposition (SVD) Implementation, Independent Component Analysis (ICA), The Application of ICA, Cumulative Distribution Function (CDF), ICA Algorithm, The Applications of ICA

Unit 5: Reinforcement learning and control

10 hrs

Applications of Reinforcement Learning, Markov Decision Process (MDP), Defining Value & Policy Functions, Value Function, Optimal Value Function, Value Iteration, Policy Iteration, Generalization to Continuous States, Discretization & Curse of Dimensionality,

Models/Simulators, Fitted Value Iteration, Finding Optimal Policy, State-action Rewards, Finite Horizon MDPs, The Concept of Dynamical Systems, Examples of Dynamical Models, Linear Quadratic Regulation (LQR), Linearizing a Non-Linear Model, Computing Rewards, Riccati Equation, Advice for Applying Machine Learning, Debugging Reinforcement Learning (RL) Algorithm, Linear Quadratic Regularization (LQR), Differential Dynamic Programming (DDP), Kalman Filter & Linear Quadratic Gaussian (LQG), Predict/update Steps of Kalman Filter, Linear Quadratic Gaussian (LQG), Partially Observable MDPs (POMDPs), Policy Search, Reinforce Algorithm, Pegasus Algorithm, Pegasus Policy Search, Applications of Reinforcement Learning

Text Books

1. Christopher Bishop, Pattern Recognition and Machine Learning. Springer, 2006
2. Richard Duda, Peter Hart and David Stork, Pattern Classification, 2nd ed. John Wiley & Sons, 2001.

References

1. Tom Mitchell, Machine Learning. McGraw-Hill, 1997.
2. Richard Sutton and Andrew Barto, Reinforcement Learning: An introduction. MIT Press, 1998

Systems Programming

Course Title: Systems Programming

Full Marks: 45+30

Course Code: C.Sc. 565

Pass Marks: 22.5+15

Credit Hours = 3

Nature of the course: Theory +Lab

Course Description

This course will introduce the design and implementation of machine dependent, as well as machine independent aspects of assembler, loader, linker, microprocessor and some aspects of compiler. A project involving implementation of an assembler, a linker, a loader, and a compiler will form an integral part of the course.

Course Objectives

The purpose of this course is to present the basic structure and design of a micro-assembler, a linker, a loader, and a compiler. Since software components are best learned by implementation, each student will complete a project independently which will involve the design and implementation of these three software components.

Prerequisites

Logic Design, Data Structure and algorithm, Programming language, Familiarity with assembly language programming

Course Contents

Unit 1: SIC and SIC/XE Machine Structure 4 hrs

- 1.1 Introduction
- 1.2 System software and machine architecture
- 1.3 Simplified Instructional computers SIC, SIC/XE architecture.
- 1.4 RISC and CISC machine architecture.

Unit 2: Assembler Design 20 hrs

- 2.1 Basic assembler functions: Simple SIC Assembler, Data structures for Assembler

- 2.2 Machine dependent assembler features: Instruction formats and addressing modes, program relocation
- 2.3 Machine independent assembler features: Literals, Symbol defining statements, program blocks, control sections, program linking
- 2.4 Assemblers Design Options: One pass assembler, multi-pass assembler

Unit 3: Loader and Linker Design 8 hrs

- 3.1 Basic loader functions: Design of absolute loader, Simple Bootstrap loader
- 3.2 Machine dependent features: Relocation, Program Linking
- 3.3 Machine independent Loader Features: Automatic library Search, loader options
- 3.4 Design Options: Linking Loader, Linkage Editor, Dynamic Linking

Unit 4: Macro processor Design 8 hrs

- 4.1 Basic Macro processor functions: Macro Definition and Macro Expansion, Data structures for Macro processor
- 4.2 Machine Independent features: Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion
- 4.3 Macro processor Design Options: Recursive Macro Expansion, General Purpose Macro Processors

Unit 5: Compiler Design 5 hrs

- 5.1 Basic Compiler Functions: Grammar, Lexical Analysis, Syntactic Analysis (operator precedence parsing, Recursive Descent Parsing), intermediate code generation

Textbook

- 1. System Software: An Introduction to Systems Programming by Leyland L.Beck: Addison Wesley Publishing Company.

Reference Books:

- 1. Daniel H. Marcellus, Systems programming for Small Computers, prentice-Hall, New 44 Jersey
- 2. Roy S. Ellzey, Computer System Software, The Programmer/Machine Interface, Inc.,
- 3. Robert M. Graham , Principles of Systems Programming, John Wiley & Sons
- 4. Thomas G. Windeknecht, 6502Systems programming, Little Brown & Company

Outcomes and Assessment

Student should design and implement Assembler and Linking Loader for simple hypothetical computer architecture.

Web Systems and Algorithms

Course Title: Web Systems and Algorithms

Course No: C.Sc. 559

Nature of the Course: Theory + Lab

Full Marks: 45 + 30

Pass Marks: 22.5+15

Credit Hrs: 3

Course Description:

This course covers the Internet systems research including the intelligent web, search engine architecture and algorithms, information retrieval, crawling, text analysis, personalization and context, collaborative environments, and the semantic web.

Course Contents:

Unit 1: Introduction (4 hrs)

Examples of intelligent web applications, Basic elements of intelligent applications; What applications can benefit from intelligence?; How can I build intelligence in my own application?; Machine learning, data mining, and all that; Eight fallacies of intelligent applications

Unit 2: Searching (8 hrs)

Searching with Lucene; Why search beyond indexing?; Improving search results based on link analysis; Improving search results based on user clicks; Ranking Word, PDF, and other documents without links; Large-scale implementation issues; Is what you got what you want? Precision and recall

Unit 3: Creating Suggestions and Recommendations (7 hrs)

An online music store: the basic concepts; How do recommendation engines work?; Recommending friends, articles, and news stories; Recommending movies on a site; Large-scale implementation and evaluation issues

Unit 4: Clustering: Grouping Things Together (7 hrs)

The need for clustering; An overview of clustering algorithms; Link-based algorithms; The k-means algorithm; Robust Clustering Using Links (ROCK); DBSCAN; Clustering issues in very large datasets

Unit 5: Classification: Placing Things Where They Belong (7 hrs)

The need for classification; An overview of classifiers; Automatic categorization of emails and spam filtering; Fraud detection with neural networks; Are your results credible?; Classification with very large datasets

Unit 6: Combining Classifiers (6 hrs)

Credit worthiness: a case study for combining classifiers; Credit evaluation with a single classifier; Comparing multiple classifiers on the same data; Bagging: bootstrap aggregating; Boosting: an iterative improvement approach

Unit 7: Semantic Web (6hrs)

Building models, Calculating with knowledge, Exchanging information, Semantic web technologies, Introduction to Resource Description Language RDF and Web Ontology Language OWL

Recommended Books:

1. Algorithms of the Intelligent Web, Haralambos Marmanis and Dmitry Babenko, Manning Publications, 2009
2. Foundations of Semantic Web Technologies, Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph, CRC Press/Chapman and Hall (2009)

Seminar II

Course Title: Seminar II
Course No: C.Sc. 560
Nature of the Course: Seminar

Full Marks: 25
Pass Marks: 12.5
Credit Hrs: 1

Course Description: The seminar-II is of full marks 25 offered in the curriculum of the M. Sc. first year second semester. A student pursuing the seminar prepares a seminar report and presents the seminar in the department. Once accepted by the department, the students have to submit the final copy of the report.

Introduction:

Each student is required to write a comprehensive report about the seminar. The report should consist of 5 to 10 pages describing the topic selected. Students can choose the seminar topics of their relevant subject area. The students are suggested to select the research oriented topics rather than just informative ones. The report should be in the format as described below;

Arrangement of Contents:

The sequence in which the seminar report material should be arranged and bound should be as follows:

5. Cover Page & Title Page
6. Abstract
7. Chapters:
 - a. Introduction
 - b. Previous Works, Discussions and Findings
 - c. Conclusion
8. References

Format of References

1. References

A list of all publications (articles, texts, monographs, etc.) must be supplied as the last section of the paper. Each article or paper used must be listed alphabetically by last name of the author and the list must be numbered sequentially. The following are examples of the format for various types of entries in the list.

Journal: Stalling, W., RSA and its computational aspects, *Infoworld*, 12, 28 (Jul. 2012), 42-49.

Book: Bishop, M. and Boneh, D., *Elements of Computer Security*, Pearson Education., 2009.

Proceedings: Shamir, A., Controlling attacks on public key cryptography. *Proceedings of OOPSLA 86* (Sept. 1986., Portland), 405-416.

Articles Which Have Established Citation Pages at ACM (Electronically published articles): Smith, J., An algorithm for the traveling salesman problem, *Journal of the ACM* 54 (June 2011), 234-245. (<http://www.acm.org/jacm/2011/SmithtspAlgo/>)

2. Citations

Whenever material from a publication is used in the paper it must be followed by a citation which is simply the number of the reference in the list of references enclosed in square brackets (for example, a reference to the third article listed in the list of references would contain the citation [3].) Multiple citation numbers can be incorporated within one citation when required (for example, references to the fourth, eighth, and eleventh entries in the reference list would appear as [4, 8, 11]).

Text of the seminar paper format:

5. The paper can be prepared using a word processor or LATEX. The students are highly recommended to use LATEX.
6. Margins - All margins must be one inch.
7. The text must be spaced by 1.5.
8. The text must be typed in 12 point font. The text must be typed in Times New Roman font.

**THIRD
SEMESTER**

Advanced Cryptography

Course Title: Advanced Cryptography

Full Marks: 45 + 30

Course No: C.Sc. 619

Pass Marks: 22.5 + 15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course Description: The objective of the course is to understand the principles of cryptographic systems. The course consists of detailed knowledge about cryptographic functions, hash functions and message authentication codes, key management issues, and secret sharing schemes.

Course Content:

Unit 1: Shannon Theory and Classical Cryptography

5 hrs

Crypto Systems, Shannon Theory, Evaluation of Cryptosystems: computational security, provable security, unconditional security, Entropy, Classical Cryptosystems: shift cipher, affine cipher, Vigenere cipher, hill cipher, permutation cipher, Cryptanalysis of shift, affine, Vigenere, hill and permutation ciphers

Unit 2: Block and Stream Ciphers

10 hrs

Block Cipher Principles, Review of DES, Strength of DES: Linear-Cryptanalysis, Differential-Cryptanalysis, Review of AES and its strength, International Data Encryption Algorithm (IDEA), Blowfish, Stream Ciphers: Synchronous Stream Ciphers: LFSR, NLFSR, Self-Synchronous Stream Ciphers, RC4, RC5

Unit 3: Public Key Cryptography and Discrete Logarithms

12 hrs

Principles of Public-Key Cryptosystems, RSA, Security of RSA: Brute Force Attack, Mathematical Attack, Timing Attack, Chosen Cipher Text Attack, Rabin Cryptosystem, The Discrete Logarithm System, The Discrete Logarithm Problem, ElGamal Cryptographic system, ElGamal Digital Signature Scheme, Some

Basic Facts of Elliptic Curves, The Geometry of Elliptic Curves, Line Through Two Distinct Points, A Tangent Line, Addition of Points on Elliptic Curves, Cryptosystems Defined over Elliptic Curves, The Discrete Logarithm Problem over Elliptic Curves, Elliptic Curve Cryptosystems, Digital Signatures using ECC

Unit 4: Cryptographic Hash Function and Message Authentication

9 hrs

Hash functions and Data Integrity, Message Authentication, Digital Signatures, Security of Hash functions: Random Oracle Model, Iterated Hash Functions: The Merkle-Damgard Construction, SHA family, SHA-3, Message Authentication Codes: Nested MACS, CBC-MAC and Authenticated Encryption, MAC based on Hash Functions: HMAC, MAC based on Block Ciphers: DAA

Unit 5: Key Management

5 hrs

Session and Interchange Keys, Key Distribution, Diffie-Hellman Key Distribution, Blom Key Distribution Scheme, Session Key Exchange: Simple Key Exchange Protocol, Needham-Schroeder Protocol, Denning-Sacco Protocol, Otway-Rees Protocol, Kerberos System, Key Generation, Cryptographic Key Infrastructures, Storing and Revoking of Keys

Unit 6: Secret Sharing Scheme

4 hrs

The Fiat-Shamir Protocol, Schnorr's Identification Protocol, Threshold Schemes, Threshold Schemes with Liars, Access Structures and General Secret Sharing Schemes, Visual Secret Sharing Schemes

Recommended Books:

4. Douglas R. Stinson, Cryptography: Foundations and Theory and Practice, Chapman and Hall/ CRC
5. William Stallings, Cryptography and Network Security: Principles and Practice, 5th Edition, Pearson
6. Matt Bishop, Computer Security: Art and Science, AddisonWesley

Embedded Systems

Course Title: Embedded Systems

Course No: C.Sc. 622

Nature of the Course: Theory + Lab

Course Synopsis: Advanced concept of Embedded System

Full Marks: 45 + 30

Pass Marks: 22.5 + 15

Credit Hrs: 3

Goal: This course covers a broad range of topics related to embedded system hardware, software and real time operating Systems

UNIT 1

10 Hrs

Introduction to Embedded System: Components of Embedded System – Classification - Characteristic of embedded system- Microprocessors & Micro controllers- Introduction to embedded processors - Embedded software architectures: Simple control loop - Interrupt controlled system - Cooperative multitasking - Preemptive multitasking or multi-threading - Micro kernels - Monolithic kernels - Exotic custom operating systems.

UNIT 2

10 Hrs

Embedded Hardware Architecture – 32 Bit Microcontrollers: 32 Bit microcontrollers and family of processors, Register, Memory and Data transfer, Arithmetic and Logic instructions, Assembly Language, I/O operations interrupt structure, Networks for Embedded systems

UNIT 3

10 Hrs

Software Development: Embedded Programming, Source Code Engineering Tools for Embedded, Program Modeling Concepts in Single and Multiprocessor Systems - Software Development Process - Software Engineering Practices in the Embedded Software Development – Hardware / Software Co-design in an Embedded System.

UNIT 4

15 Hrs

Real Time Operating Systems in embedded system: Tasking Models, Task States, Services and Transitions - Real-Time Scheduling Algorithms: Round-Robin, FIFO, Priority-Based Preemptive Scheduling - Rate-Monotonic Scheduling - Priority Inversion and Priority Ceiling - Deadlocks - Process Synchronization – IPC - Shared Memory, Memory Locking, Memory Allocation - Signals – Semaphore Flag as Resource key – Message Queues – Mailboxes – Pipes – Virtual Sockets, RTOS System Level Functions – Task Service Functions– Time Delay Functions – Memory Allocation Related Functions – Semaphore Related Functions– Mailbox Related Functions – Queue Related Functions – Case Studies of Programming with RTOS.

REFERENCES

1. Noergaard, “Embedded System Architecture”, Elsevier India Private Limited, latest edition
2. Steve Heath, “Embedded Systems Design”, Newnes, latest edition
3. Sriram Iyer and Pankaj Gupta, “Embedded Real Time Systems Programming”, Tata McGraw-Hill, latest edition

Fuzzy Systems

Course Title: Fuzzy Systems

Full Marks: 45 + 30

Course No: C.Sc. 621

Pass Marks: 22.5 + 15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course Description: This course deals with introduction, fuzzy mapping, membership functions, fuzzy knowledge based system, fuzzy controller, nonlinear systems and adaptive fuzzy controller, and hybrid systems.

Course Objectives:

- Introduce the concept of fuzzy logic.
- Design using fuzzy system.

1. Introduction to fuzzy set theory:

7Hrs

- 1.1 Probabilistic reasoning,
- 1.2 Fuzzy sets,
- 1.3 mathematics of fuzzy set theory,
- 1.4 operations on fuzzy sets,
- 1.5 comparison of fuzzy and crisp set theory.

2. Fuzzy mapping:

7Hrs

- 2.1 one to one mapping,
- 2.2 max-min principle,
- 2.3 extension principle,
- 2.4 implication rules – mamdani implications.

3. Membership functions:

8Hrs

- 3.1 Universe of discourse,
- 3.2 mapping inside fuzzy domain,
- 3.3 fuzzy membership mapping methods,
- 3.4 application to real world problems.

4. Fuzzy knowledge based systems:

7Hrs

- 4.1 Fuzzification,

- 4.2 Fuzzy knowledge base, rule base,
- 4.3 Data base for fuzzy,
- 4.4 Inference rules,
- 4.5 defuzzification methods of defuzzification.

5. Fuzzy controller:

6Hrs

- 5.1 Control strategies,
- 5.2 general PID controller,
- 5.3 Implementation of fuzzy systems in control ,
- 5.4 Direct fuzzy controller,
- 5.5 PI and PID controller,
- 5.6 Indirect fuzzy controller – fuzzy in handling the inner loops of control systems.

6. Nonlinear systems and adaptive fuzzy controller:

6Hrs

- 6.1 Nonlinear systems,
- 6.2 modification in fuzzy systems for nonlinear control,
- 6.3 Adaptive control,
- 6.4 Adaptive control using fuzzy,
- 6.5 fuzzy sliding mode controls.

7. Hybrid systems:

4Hrs

- 7.1 Neuro- fuzzy and fuzzy genetic systems,
- 7.2 Applications to scientific problems.

Laboratory:

The laboratory exercises should contain all the features mentioned above.

Reference books:

1. Neural Networks and Fuzzy Logics, Bart Kosko, Pearson Education, NewDelhi, 2011
2. Fuzzy logic to engineering applications –Timothy J. Ross.
3. Fuzzy control –Driancov.
4. Fuzzy modeling and control –Yager.

Image Processing and Pattern Recognition

Course Title: Image Processing and Pattern Recognition

Full Marks: 45 + 30

Course No: C.Sc. 623

Pass Marks: 22.5+15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course objectives: To be familiar with processing of the digital images, recognition of the pattern and their applications.

Unit 1: Introduction (6 Hrs)

Introduction: Digital Image, Fundamental steps in Image Processing, Elements of Digital Image Processing systems, Element of visual perception.

Digital Image Fundamentals: A Simple Image Model, Sampling and Quantization, Some Basic Relationships between Pixels, Image Geometry transforms in 2D.

Unit 2: Image Enhancement and Filtering (14 Hrs)

Image Enhancement in the Spatial Domain: Introduction to Spatial and Frequency Methods, Basic Gray Level Transformations, Histogram Equalization, Histogram Matching, Histogram Processing, Local Enhancement, Image Subtraction, Image Averaging, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters.

Fourier Transforms: Introduction to Fourier Transformation, Discrete Fourier Transformation, Fast Fourier Transformation, Fourier Properties, 2D FT, Inverse Fourier Transform.

Filtering in the Frequency Domain: Filtering in the Frequency Domain, Correspondence between Filtering in Spatial and Frequency Domain, Smoothing Frequency Domain Filters, Sharpening Frequency Domain Filters.

Unit 3: Image Restoration and Compression (10 Hrs)

Image Restoration: Models for Image degradation and restoration process, Noise Models, Estimation of Noise Parameters, Restoration Filters, Bandrejected Filters, Bandpass Filters, Inverse Filtering, Wiener Filtering.

Image Compression: Image compression models, standards and coding Techniques.

Unit 4: Image Segmentation and Representation (8 Hrs)

Image Segmentation: Point Detection, Line Detection, Edge Detection, Gradient Operator, Edge Linking and Boundary Detection, Hough Transform, Thresholding, Region-oriented Segmentation.

Representation: Chain Codes, Polygonal Approximations, Signatures, Boundary Segments, Skeleton of a Region, Boundary Descriptors, Shape Numbers, Fourier Descriptors, Regional Descriptors, Simple Descriptors, Topological Descriptors.

Unit 5: Pattern Recognition (7 Hrs)

Introduction to Pattern Recognition, Patterns and Pattern Classes, strategies and models, Pattern Classifiers, Neural Network and Neural learning tools for pattern recognition, Structural Methods.

Laboratory Work: Developing programs of above features using C/C++/MATLAB.

Text Book:

Rafel C. Gonzalez and Richard E. Woods, "Digital Image Processing", 3rd Edition, Pearson education

Reference Books:

W. K. Pratt, "Digital Image Processing", 3rd Edition, John Wiley and Sons, New York.

Literature Review Research

Course Title: Literature Review Research

Course No: C.Sc. 620

Nature of the Course: Seminar

Full Marks: 50

Pass Marks: 25

Credit Hrs: 2

Course Description: The objectives of the literature review research are to provide students with an opportunity to develop in-depth expertise and to provide an opportunity for some individual interaction, planning and executing a substantial research work in the field of computer science.

Introduction:

The format of term papers with regard to content is based upon the style used in the ACM publication Communications of the ACM and guidelines for presentation and typing are specified by the department. Students are strongly encouraged to develop a familiarity with the editorial style and format of Communications prior to manuscript preparation.

The literature review research paper consists of four parts:

1. Title Page - a standardized page for specifying the title and author of the paper. A signature page is required for the file paper.
2. Abstract - a brief, concise summary of the paper as described in the suggested outline. It must be printed on a page by itself.
3. The text / body of the paper.
4. References

Suggested Outline of the Body:

The paper is to be written based on the instructions of the course instructor (term paper) or the research supervisor (file paper). A suggestion outline is given below.

Abstract:

Give a brief summary of the intent of the paper and a description of the major concepts, approaches, algorithms, and/or significance of the paper. The abstract is limited to one paragraph of at most ten sentences.

Introduction, Motivation and Purpose:

This section introduces the topic and the purpose of the paper. It must address:

- Why is the topic important?
- What is the past and future for this topic? (Why did you choose this topic?)
- What is the purpose of the research in this topic area? There must be a specific, well-reasoned goal for the paper which must be clearly and concisely stated in this section.

- A brief introduction of the type of paper (survey, comparative study, analysis, algorithms developed, etc.) should be described.

Overview of Previous Work

Perform a literature search to collect everything written on the topic. Concentrate on the most current research, but do not ignore foundational or seminal papers in the area. Summarize the significant material that pertains primarily to your topic. Keep in mind that the references to research should be kept narrowly focused and directed to the primary goal of your paper.

Presentation and Discussion

This section is the most significant part of the paper. It presents the details of the approach, methodology, algorithm, analysis, comparative study, result, etc.

Conclusion

Summarize and evaluate the paper in this section. Specifically, draw substantive conclusions from the paper. Evaluate the results and comment on the positive points revealed in the study and identify and comment on any negative results. Identify possibilities for future studies or research which you may have discovered.

References

Papers must come from refereed journals or computer magazines or proceedings and must follow the format specified.

Format of references, citations, tables, figures:

1. References

A list of all publications (articles, texts, monographs, etc.) must be supplied as the last section of the paper. Each article or paper used must be listed alphabetically by last name of the author and the list must be numbered sequentially. The following are examples of the format for various types of entries in the list.

Journal: Stalling, W., RSA and its computational aspects, *Infoworld*, 12, 28 (Jul. 2012), 42-49.

Book: Bishop, M. and Boneh, D., *Elements of Computer Security*, Pearson Education., 2009.

Proceedings: Shamir, A., Controlling attacks on public key cryptography. *Proceedings of OOPSLA 86* (Sept. 1986., Portland), 405-416.

Articles Which Have Established Citation Pages at ACM (Electronically published articles): Smith, J., An algorithm for the traveling salesman problem, *Journal of the ACM* 54 (June 2011), 234-245. (<http://www.acm.org/jacm/2011/SmithtspAlgo/>)

2. Citations

Whenever material from a publication is used in the paper it must be followed by a citation which is simply the number of the reference in the list of references enclosed in square brackets (for example, a reference to the third article listed in the list of references would contain the citation [3].) Multiple citation numbers can be incorporated within one citation when required (for example, references to the fourth, eighth, and eleventh entries in the reference list would appear as [4, 8, 11]).

3. Tables

Tables of data must appear *after* the first reference to them. They may appear on a separate page, but the page must follow the page on which the first reference is made. A reference may be incorporated into the text as part of a sentence, for example, “Table 1 contains...”. Tables must be numbered consecutively and have a title which appears centered above the table as shown in the example. Any table not original with the author must have a citation that references the source document. Do not refer to the physical location of the table, e.g., “The table below ...”

Table 1. Caption of the table

Test	Test	Test

4. Figures

Figures and illustrations must appear *after* the first reference to them. They may appear on a separate page, but the page must follow the page on which the first reference is made. Several figures may appear on the same separate page, but they must follow the first reference made to them in the text. A reference to a figure may be incorporated into the text as part of a sentence, for example, “Figure 1 shows...” Figures must be numbered consecutively and have a title which appears centered below the figure as shown in the example. Any figure or illustration not original with the author must have a citation that references the source document. Do not refer to the physical location of the figure, e.g., “The figure below ...”

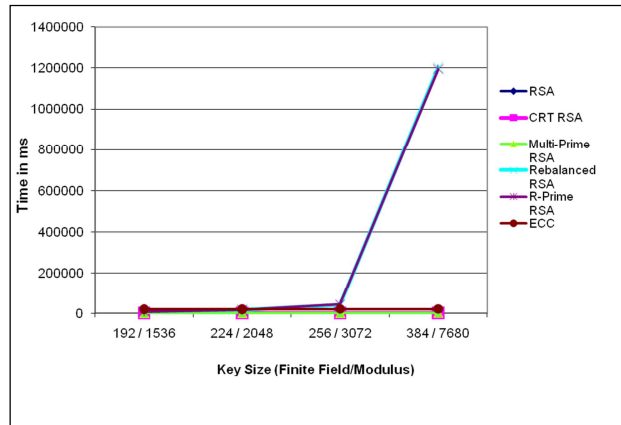


Figure1. Graph showing the variation of encryption time with increasing key

Text of the paper- presentation format:

9. The paper can be prepared using a word processor or LATEX. The students are highly recommended to use LATEX.
10. Margins - All margins must be one inch.
11. The text must be spaced by 1.5.
12. The text must be typed in 12 point font. The text must be typed in Times New Roman font.

Multimedia Computing

Course Title: Multimedia Computing

Full Marks: 45 + 30

Course No: C.Sc. 625

Pass Marks: 22.5+15

Nature of the Course: Theory + Lab

Credit Hrs: 3

Course Objectives:

The main objective of this course cover three main objectives on multimedia technology these are devices, systems and applications

Unit 1: Introduction

4 Hrs

Global structure of Multimedia, Medium, Multimedia system and properties

Unit 2: Sound / Audio System

5 Hrs

Concepts of sound system, Music and speech, Speech Generation, Speech Analysis, Speech Transmission

Unit 3: Images and Graphics

5 Hrs

Digital Image Representation, Image and graphics Format, Image Synthesis, analysis and Transmission

Unit 4: Video and Animation

5 Hrs

Video signal representation, Computer Video Format, Computer- Based animation, Animation Language, Methods of controlling Animation ,Display of Animation, Transmission of Animation

Unit 5: Data Compression

6 Hrs

Coding Requirements, Source, Entropy and Hybrid Coding, Lossy Sequential DCT- based Mode, Expanded Lossy DCT-based Mode, Hierarchical mode, Video and Audio Encoding, Audio and still Image Encoding

Unit 6: Communication Systems in Multimedia

6 Hrs

Application and Transport subsystem, Quality of service and resource management, Trends in collaborative Computing, Trends in Transport Systems

Unit 7: User Interfaces

4 Hrs

Basic Design Issues, video and Audio at the User Interface, User- friendliness as the Primary Goal

Unit 8: Abstractions for programming

5 Hrs

Abstractions Levels, Libraries, System Software, Toolkits, Higher Programming Languages, Object –oriented approaches

Unit 9: Multimedia Computing Applications

5 Hrs

Media composition, integration, communication, entertainment

References:

1. Multimedia: Computing, Communications and Applications, Ralf Steinmetz and Klara Nahrstedt, Pearson Education Asia
2. Multimedia Communications, Applications, Networks, Protocols and Standards, Fred Halsall, Pearson Education Asia
3. Multimedia Systems, John F. Koegel Buford, Pearson Education Asia

Principles of Programming Languages

Course Title: Principles of Programming Language s **Full Marks:** 45 + 30

Course No: C.Sc. 618 **Pass Marks:** 22.5 + 15

Nature of the Course: Theory + Lab **Credit Hrs:** 3

Course Description: Design, evaluation, implementation issues of programming language. Data types, data abstraction, sequence control, procedural abstraction, parameter passing techniques, scope, storage management, object oriented approach, concurrency, co routines and overview of a few high level programming languages.

Course Objectives:

- Introduce the design and implementation issue of high level programming languages.
- Principles for the evaluation of procedural, object- oriented and concurrent programming languages.

Unit 1:

- 1.1. Introductory Concepts: History of the development of programming language, Reason for studying concept of Programming Language, Programming Language Domains, Attributes of Programming Language, Standardization, Programming Environments. 4hrs
- 1.2. Language Design Issue: Different architectures, Virtual Machine and Binding times 3hrs
- 1.3. Translation Issues: Translation; Syntax and Semantics: Syntactic criteria, Syntactic elements Formal methods of describing syntax – BNF, RG, Parse tree, Ambiguity, Parsing Algorithm Semantic modeling – attribute grammar, denotational semantics. 7hrs

Unit 2:

- 2.1. Data Types: Introduction, Specification and Implementation of primitive and other data 4hrs

types, Declaration, initialization and assignment, Type checking and conversion.

- 2.2. Abstract Data Type, Information hiding, Encapsulation, Type Equivalence, Storage Management (static, stack-based, heap-based). 4hrs
- 2.3. Inheritance: derived class, abstract class, object, message, and polymorphism. 3hrs

Unit 3:

- 3.1. Sequence Control with basic statement, Sequence Control with arithmetic statement, Sequence Control with non arithmetic statement, Sequence Control between statements. 4hrs
- 3.2. Subprogram control: Introduction, Subprogram sequence control, Parameter transmission, Scope (static/dynamic) and lifetime, Block Structure, Local referencing environment. 4hrs

Unit 4:

- 4.1. Distributed Processing: Exceptions, Propagation and Exception handlers, Co routines, Parallel Programming 4hrs
- 4.2. Paradigms and Languages (Overview): Procedural (FORTRAN & C),Block Structure (PASCAL),Object Oriented (C++, SMALLTALK), Functional(LISP), and Logic (PROLOG). 8hrs

Text Book:

- 1. T.W. Pratt and M.V. Zelkowitz , Programming Languages: Design and Implementation. Prentice Hall.

Reference Book:

- 1. R.W. Sebesta , Concepts of Programming Languages, Addison Wesley, ISBN : 0-201385961

Remote Sensing and GIS

Course Title: Remote Sensing and GIS

Course No: C.Sc. 624

Nature of the Course: Theory + Lab

Full Marks: 45 + 30

Pass Marks: 22.5+15

Credit Hrs: 3

Course Description:

This course covers the concepts and principles of remote sensing, global navigation satellite System (GNSS) and GIS

Course Contents:

Unit 1: Concept of Remote Sensing

4 hrs

Introduction; Distance and Definition of Remote Sensing; Remote Sensing: Art and/or Science; Data; Remote Sensing Process; Source of Energy; Interaction with Atmosphere; Interaction with Target; Recording of Energy by Sensor; Transmission, Reception, and Processing; Interpretation and Analysis; Applications, Advantages, and Limitations of Remote Sensing; Ideal Remote Sensing System

Unit 2: Types of Remote Sensing and Sensor Characteristics

3 hrs

Introduction; Types of Remote Sensing; Characteristics of Image; Orbital Characteristics of Satellite; Remote Sensing Satellites; Concept of Swath and Nadir; Sensor Resolutions; Image Referencing System

Unit 3: Photographic Imaging

3 hrs

Introduction; Camera Systems; Types of Camera; Filter; Film; Geometry of Aerial Photography; Ideal Time and Atmosphere for Aerial Remote Sensing

Unit 4: Digital Imaging

4 hrs

Introduction; Digital Image; Sensor; Imaging by Scanning Technique; Hyper-spectral Imaging; Imaging by Non-scanning Technique; Thermal Remote Sensing; Other Sensors

Unit 5: Microwave Remote Sensing

4 hrs

Introduction; Passive and Active Microwave Remote Sensing; Radar Imaging; Airborne Versus Space-borne Radars; Radar Systems

Unit 6: Ground Truth Data and Global Positioning System

5 hrs

Introduction; Requirements of Ground-Truth Data; Instruments for Ground Truthing; Parameters of Ground Truthing; Factors of Spectral Measurement; Global Navigation Satellite System

Unit 7: Photogrammetry

6 hrs

Introduction; Development and Classification; Photogrammetric Process; Acquisition of Imagery and its Support Data; Orientation and Triangulation; Stereo Model Compilation; Stereoscopic 3D Viewing; Stereoscopic Measurement; DTM/DEM Generation; Counter Map Generation; Orthorectification; 3D Feature Extraction; 3D Scene Modeling; Photogrammetry and LiDAR; Radargrammetry and Radar Interferometry; Limitations

Unit 8: Visual Image Interpretation**5 hrs**

Introduction; Information Extraction by Human and Computer; Remote Sensing Data Products; Border or Marginal Information; Image Interpretation; Elements of Visual Image Interpretation; Interpretation Keys; Generation of Thematic Maps; Thermal Image Interpretation; Radar Image Interpretation

Unit 9: Digital Image Processing**3 hrs**

Introduction; Categorization of Image Processing; Image Processing Systems; Digital Image; Media for Digital Data Recording, Storage, and Distribution; Data Formats of Digital Image; Header Information; Display of Digital Image; Pre-processing; Image Enhancement; Image Transformation; Image Classification

Unit 10: Data Integration, Analysis, and Presentation**4hrs**

Introduction; Multi-approach of Remote Sensing; Integration with Ground Truth and Other Ancillary Data; Integration of Transformed Data; Integration with GIS; Process of Remote Sensing Data Analysis; The Level of Detail; Limitations of Remote Sensing Data Analysis; Presentation

Unit 11: Concept of GIS**4 hrs**

Introduction; Definition; Key Components; GIS-An Integration of Spatial and Attribute Information; GIS-Three Views of Information System; GIS and Related Terms; GIS-A Knowledge Hub; GIS-A Set of Interrelated Subsystems; GIS-An Information Infrastructure; Origin of GIS; Functions of GIS

Recommended Books:

6. Remote Sensing and GIS, Basudeb Bhatta, Second Edition, Oxford University Press
7. Remote Sensing and GIS Integration, Theories, Methods, and Application, Qihao Weng, McGrawHill

**FOURTH
SEMESTER**

Course Title: Cloud Computing

Full Marks: 45+30

Course No: C.Sc. 668

Pass Marks: 22.5+15

Credit Hours: 3

Nature of the course: Theory + Lab

Course Description: This course gives the concepts of cloud computing and its infrastructures. In this course we explore aspects of cloud computing like; introduction to cloud computing, cloud architecture, cloud service models, cloud applications and paradigms, and security issues in cloud computing.

Course Objectives: The objective is to provide the students with the knowledge, understanding, and skills required for designing, building, and evolving systems and infrastructures to exploit cloud computing paradigm.

Course Content:

Unit 1: Introduction

(6 Hrs)

Defining the Cloud, The Emergence of Cloud Computing, Grid Computing or Cloud Computing, Types of Cloud, Cloud Computing Paradigms and Services, Components of Cloud Computing, Ethical Issues in Cloud Computing, Cloud Vulnerabilities, Characteristics of Cloud Computing.

Unit 2: Role of Networks in Cloud Computing

(6 Hrs)

Parallel Computing, Distributed Systems, Network Architecture for Cloud: Data Center Network, Data Center Interconnect Network, Public Internet, Foundations of Cloud Computing Infrastructures: Virtualization, Automation, & Standard, Network Architecture for Hybrid Deployment

Unit 3: Role of Grid Computing in Cloud Computing

(5 Hrs)

Grid Computing, Interaction of Models of Grid and Cloud Computing, Distributed Computing in Grid and Cloud, Layered Models and Usages Patterns in Grid and Cloud, Interoperability in grids and clouds

Unit 4: Cloud Service Models and Cloud Infrastructure

(10 Hrs)

Jericho Cloud Cube Model, Communication-as-a-Service, Infrastructure-as-a-Services, Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), Cloud Computing at Amazon, Amazon Web Services, Cloud Computing from the Google Perspective, Window Azure and Online Services, Open Source Software Platforms for Private Clouds

Unit 5: Cloud Computing Applications and Paradigms

(6 Hrs)

Existing Cloud Applications: Processing Pipelines, Batch Processing Systems, Web Applications, Architectural Style for Cloud Applications, Workflow in Cloud Applications, Coordination Models, Hadoop System, Social Computing and Cloud Computing

Unit 6: Building Cloud Networks

(6 Hrs)

Evolution from Managed Service Providers (MSP) to Cloud Computing, Single Purpose Architectures to Multi-purpose Architectures, Data Center Virtualization, Cloud Data Center, Service Oriented Architectures (SOA), Combining and SOA, Characterizing SOA, Open Source Software in Data Centers

Unit 7: Security in Cloud Computing

(6 Hrs)

Cloud Security Challenges, Dimensions of Cloud Security: Security & Privacy, Compliance, and Legal or Contractual Issues, Risk Management, Security Monitoring, Incident Response Planning, Security Architecture Design, Vulnerability Assessment, Data and Application Security, Virtual Machine Security, Handling Disasters in Cloud

Recommended Books:

1. Dan C. Marinescu, *Cloud Computing: Theory and Practice* (For Unit 1 4 5)
2. John W. Rittinghouse and James F. Ransome, *Cloud Computing: Implementation Management and Security*, (Recommended for Unit 1, 4, 6, 7)
3. George Reese, *Cloud Application architecture*, O'Reilly Media Inc. (Recommended for Unit 7)
4. Judith Hurwitz, Robin Bloor, Marcia Kaufman, Fern Halper, *Cloud Computing for Dummies*, Wiley Publishing Inc. (Recommended for Unit 6)
5. Borko Furht, Armando Escalante, *Handbook of cloud computing*, Springer, 2010 (Recommended for Unit 1, 2, 3)
6. David S. Linthicum, *Cloud Computing and SOA Convergence in your Enterprise*, a step by step guide, Addison Wesley (Recommended for Unit 1, 4, 6)

e-Government

Course Title: e-Government

Course No: C.Sc. 669

15

Nature of the Course: Theory + Case Studies

Full Marks: 45 + 30

Pass Marks: 22.5 +

Credit Hrs: 3

Objective of the course:

The objective of this course is to study about implementing and managing of e-Government. It defines e-Government in broad sense: all use of ICT in the public sector. It covers a broad range of managerial issues: from high-level strategy to detail tactics; from the technicalities of data flows of e-Government, e-government infrastructure development, application and use of Data warehousing and Data Mining in e-Government and also case studies of different countries.

Course Contents:

Unit 1: Introduction

3 Hrs

Conceptual Overview of e-Government, and E-Governance, e-Government Models, Need for e-government, Maturity Phases of e- Government

Unit2: Managing e-Government

3 Hrs

Approaches to management of e-Government systems, e-government strategy, Managing public data, public private partnership for e-Government

Unit 3: e-Government Readiness

6 Hrs

e-Readiness, e-Readiness Framework, steps use in e-Government Readiness, Issue in e-Government Readiness

Unit 4: Infrastructure development

6 Hrs

Nationwide Communication Infrastructure, Computing Infrastructure, Data Centers, e-government Architecture, Interoperability Framework, e-Government Gateway

Unit 5: Security for e-Government

6 Hrs

Security for e-Government , Challenges of e-Government security, approach to security for e-Government model of security management-Government security architecture, security standards

Unit 6: Implementing e-Government

6 Hrs

e-Government system life cycle and project assessment, analysis of current system, design of the new e-Government system, e-Government risk assessment and mitigation-Government system construction, implementation and beyond

Unit 7: Data warehousing and data mining in e-Government

5 Hrs

Data Warehousing Architecture, Building Data Warehouse, Need for Data Warehousing, Data Warehousing Tools, Data Mining and Business Intelligence, National Data warehouses, applications of Data Warehousing and data mining in e-Government

Unit 8: case studies of e-government of different developed and developing countries 10 hrs

References:

1. Richard Heeks, Implementing and managing e-Government
2. C.S.R Prabhu, e-Governance: concepts and case studies, prentice hall of India Pvt. Ltd.
3. J. Satyanarayana, e-Government , prentice hall of India Pvt. Ltd
4. e-Government master plan, Government of Nepal.

Genetic Algorithms

Course Title: Genetic Algorithms
Course No: C.Sc. 665
Nature of the Course: Theory + Lab

Full Marks: 45 + 30
Pass Marks: 22.5+15
Credit Hrs: 3

Course Objectives

General introduction to the genetic algorithms and its literature, mathematical foundation: population, mutation, crossover. Data structure for genetic algorithms and current applications of genetic algorithms.

Course Contents:

Unit 1

1.1 Introduction to Genetic Algorithm, Historical development, difference between traditional algorithms and genetic algorithms, mathematical foundation of genetic algorithm, building block hypothesis

8hrs

1.2 Primary data structures for genetic algorithm, reproduction, crossover, and mutation, mapping objective functions to fitness form, fitness scaling, coding, a multiparameter, mapped, fixed point coding, discretization, constrains into genetic algorithm search.

8hrs

1.3 The rise of genetic algorithms, genetic algorithm applications of historical interest, De Jong and function optimization, Improvement in basic techniques, current application of genetic algorithms.

7hrs

Unit 2

2.1 Advanced operators and techniques in genetic search: dominance, diploidy, and abeyance; Inversion and other reordering operators; other micro-operators, Niche and speciation, multi-objective optimization, knowledge-based techniques

12hrs

Unit 3

3.1 Genetic based machine learning, classifier system, rule and message system

4hrs

3.2 Application of genetic based machine learning, the rise of GBML, development of CS-1, Smith's poker player

6hrs

Text Book:

David E. Goldberg, "Genetic algorithms in Search, Optimization, and Machine Learning".

Reference Book:

Malanie Mitchell, "An Introduction to Genetic Algorithms"

Information and Coding Theory

Course Title: Information and Coding Theory

Full Marks: 45 + 30

Course No: C.Sc. 667

Pass Marks: 22.5 + 15

Nature of the Course: Theory

Credit Hrs: 3

Course Description: This course deals with the basic concepts of information theory and coding. Students will get idea about entropy, data compression, channel capacity and error control coding

1. Entropy

12Hrs

- 1.1 entropy,
- 1.2 relative entropy,
- 1.3 mutual information,
- 1.4 chain rules,
- 1.5 data processing inequality,
- 1.6 the asymptotic equi-partition property,
- 1.7 entropy rates for stochastic processes.

2. Data Compression

11hrs

- 2.1 source coding theorem,
- 2.2 Kraft inequality,
- 2.3 Shannon-Fano codes,
- 2.4 Huffman codes,
- 2.5 universal source codes.

3. Channel Capacity

11hrs

- 3.1 discrete channels,
- 3.2 random coding bound and converse,
- 3.3 Gaussian channels,
- 3.4 parallel Gaussian channels and "water-pouring",
- 3.5 bandlimited channels.

4. Error Control Coding

11hrs

- 4.1 linear block codes and their properties,
- 4.2 hard-decision decoding,
- 4.3 convolutional codes,
- 4.4 Viterbi decoding algorithm,
- 4.5 iterative decoding.

Reference:

1. T.M. Cover and J.A. Thomas, *Elements of Information Theory*, John Wiley India Limited, New Delhi, 2010.

Dissertation

Course Title: Dissertation
Course No: C.Sc. 666
Nature of the Course: Dissertation

Full Marks: 200
Pass marks: 100
Credit Hrs: 8

Course Description:

Dissertation is a research paper of full mark 200 offered in the curriculum of the M.Sc. second year. A student, opting for the dissertation must prepare a proposal under the supervision of a supervisor and defend in the department. Once accepted, a student can start the work under the supervision of the supervisor. This work provides students with skill and knowledge in conducting research on fundamental and application aspects of computer science and information technology.

Introduction:

All the students are required to successfully complete a research project as a part of their course. This is a major component of your degree, being worth 8 credits. The topic of the dissertation work must be relevant to M.Sc. CSIT degree. This is why it is important to agree in advance the topic of your dissertation with your supervisor and what it will entail. It is important to know that the degree is a science degree, and therefore all dissertations must be subject to scientific research.

Structure of the Report:

The best dissertations and reports usually all follow much the same structure, as described here. The exact layout of dissertations tends to vary depending upon the nature of the material and the style of the author. It is recommended that you discuss this in detail with your supervisor. However the following might be considered to be a typical layout:

1. **Title page:** With a signed declaration that the dissertation is your own work.
2. **Abstract:** Giving a short (500 words max) overview of the work.
3. **Acknowledgements:** Thanking anyone who has helped you in any way.
4. **Table of contents:** Giving page numbers for all major section headings.
5. **Introduction:** Set the scenes, explain why you are doing this work and what is the problem being solved. Most importantly you should clearly explain what the aims and objectives of your work are.
6. **Related work:** Explain the current state of the art in your area. What works have other people done (published or commercial) that is relevant to yours.
7. **Methodology:** Explain what tools and technologies have you used. If you have collected data then explain how it is collected and analyzed.
8. **Description of the work:** Explain what exactly you have done. If this is a software project, describe your software in detail. If it is a data-based project, present and explain your data in detail.
9. **Discussion:** Explain what your work means. In a software project you should evaluate the functionality of your software. In a research project you should interpret your experimental results. In all cases you should evaluate what you have achieved against the aims and objectives you outlined in the introduction. The discussion should always end with a

Conclusions section - in which you should briefly explain what conclusions you have come to as a result of doing this work.

10. **References:** provide a list of papers, books and other publications that are explicitly referred to in the text. This list must be prepared alphabetically by last name of the author and the list must be numbered sequentially. For example, "Bishop, M. and Boneh, D., *Elements of Computer Security*, Pearson Education., 2009".
11. **Appendices:** Supplementary material should be included in appendices - these are optional, but they might contain:
 - a. Code listings – A listing of the code you have written for the project. You should NOT include code listings for code you have not written!! If your project involves modifying code previously written by others, then you may include this other code as long as you indicate clearly in the code listing what parts have been written by you.
 - b. User manuals
 - c. Technical documentation
 - d. Raw data – If your dissertation involved data collection then this should usually be included in appendices. This should provide supporting evidence for claims made in the main part of the dissertation (eg copies of a user evaluation questionnaire and some sample responses).
 - e. Examples of test data
 - f. Electronic material on a floppy disk or CD taped inside the back cover. This might contain executable software, source code, graphics, slides used for your presentation, etc. Where the appendices are long (e.g. code listings) do not print them out, rather provide them on a CD.

Citations:

Whenever materials from the list of references are used in the report it must be followed by a citation which is simply the number of the reference in the list enclosed in square brackets. For example, a reference to the third article listed in the list of references would contain the citation [3]. Multiple citation numbers can be incorporated within one citation when required. For example, references to the fourth, eighth, and eleventh entries in the reference list would appear as [4, 8, 11].

Report Format:

13. The paper can be prepared using a word processor or LATEX. The students are highly recommended to use LATEX.
14. Margins - All margins must be one inch.
15. The text must be spaced by 1.5.
16. The text must be typed in 12 point font. The text must be typed in Times New Roman font.