Department of Computer Science
West Bengal State University, Barasat

Revised Course Structure M.Sc. (Computer Science) (4 Semesters)				
First semester		Credit	Marks	Total
CMSPCOR01T	Advanced Computer Architecture	4	50	Marks = 275 Credit = 22
CMSPCOR02T	Advanced Database Management Systems	4	50	
CMSPCOR03T	Advanced Data Structures and Algorithms	4	50	
CMSPCOR04T	Computer Networks	4	50	
CMSPCOR05P	Data Structures and Algorithms Laboratory	4	50	
CMSPAEC01P	Programming with SQL	2	25	
Second Semester				
CMSPCOR06T	Object Oriented Programming	4	50	Marks = 275 Credit = 22
CMSPCOR07T	Advanced Operating System	4	50	
CMSPCOR08T	Computer Graphics and Multimedia	4	50	
CMSPCOR09T	Formal Language and Automata Theory	4	50	
CMSPCOR10T	OBJECT ORIENTED Programming Laboratory	4	50	
CMSPSEC01P	Basics of Gaming	2	25	
Third semester				
CMSPCOR11T	Software Engineering	4	50	Marks = 300 Credit = 24
CMSPCOR12T	Compiler Design	4	50	
CMSPCOR13T	Artificial Intelligence	4	50	
CMSPDSE01T	Elective - I	4	50	
CMSPCOR14P	Seminar & Term Paper Leading to Project	4	50	
CMSPGEC01T	Fundamentals of Computers	4	50	
Fourth Semester				
CMSPDSE02T	Elective - II	4	50	Marks = 300 Credit = 24
CMSPDSE03T	Elective - III	4	50	
CMSPCOR15P	Software Engineering Laboratory	4	50	
CMSPCOR16P	Grand Viva-Voce	4	50	
CMSPCOR17P	Project Work	8	100	

- ELECTIVE I
  6. VLSI DESIGN
  7. CRYPTOGRAPHY AND NETWORK SECURITY
  8. CLOUD COMPUTING
  9. OPERATION RESEARCH
  10. EMBEDDED SYSTEM
  6. MORE TO BE ADDDED AS AND WHEN NECESSARY

- ELECTIVE II
  7. GRAPH THEORY
  8. PATTERN RECOGNITION
  9. NATURAL LANGUAGE PROCESSING
  10. SOFT COMPUTING
  11. COMPUTER VISION
  12. MORE TO BE ADDDED AS AND WHEN NECESSARY

- ELECTIVE III
  7. MOBILE COMPUTING
  8. FUZZY LOGIC.
  9. DATA WAREHOUSING AND DATA MINING
  10. QUANTUM COMPUTING
  11. PARALLEL COMPUTING
  12. MORE TO BE ADDDED AS AND WHEN NECESSARY

Department of Computer Science
West Bengal State University, Barasat

Regulations for
Two-Year M.Sc. Course in Computer Science.

1. The Department of Computer Science, West Bengal State University, Barasat shall provide instructions leading to 2-year, 4 Semester M.Sc. Degree in Computer Science.
2. A candidate who has passed 3-year B.Sc. Examination with Honours in Computer Science from West Bengal State University, Barasat (or from any other University or Institution) can apply for admission to the M.Sc. course. Admission for candidates from other Universities/Colleges will be governed according to the university rules.
3. The examinations for the M.Sc. course shall be held in 4 Semesters. At the end of each semester, an examination of the papers covered in that semester would be held. This examination will be refered to as the M.Sc. examination of that semester. In any semester, the study break between the completion of regular classes and the commencement of the Semester Examination will generally be a maximum of 10 calendar days.

4.1 The total credits for the 2 year (4 semester) course in Computer Science will be 100. The distribution of credits for each category is as follows:-

Subject	Credit
Theory and Tutorial	4
Laboratory	3
Project	22
Viva-voce	3
Term Paper	2
Seminar	2

4.2 Examination of a Theoretical Paper is of 2 hour duration and will usually carry 40 marks. 10 marks for each paper will be set aside for continuous assessments to be evaluated by the teacher(s) assigned for those classes.

4.3 For Theoretical papers, paper setters and examiners will be appointed from a Board of Examiners duly constituted.

4.4 Evaluation of performance in a Practical paper will be based on Sessional work in that paper and on end-semester viva-voce. The distribution of marks for each Practical Paper would be as follows;-

i) 50% for experiments performed in the laboratory – the Sessional Work to be evaluated by the Teachers assigned for that course.
ii) 40% for viva-voce on the experiments to be conducted by a Board consisting of the faculty members and / or External examiners.
iii) 10% for Lab report to be evaluated by the viva-voce Board.

Only the total mark is to be shown in the mark-sheet.

4.5 In order to pass a semester examination, a candidate will have to score minimum of 40% marks in each theoretical papers and 50% in each practical paper a candidate must appear in each theoretical and practical papers. Pass marks for project, grand viva-voce and term paper will be 50% as in practical paper.

4.6 Each candidate will have to complete a term paper assignment in 3rd semester. He / She will have to make a presentation and submit a report on the topic of the term paper. This will lead to his/her project.

4.7 Evaluation of the performance in a Term paper will be done by a board of examiners.

4.8 Each student will have to undertake a project work at the beginning of the 3rd Semester. The project work would have to be completed under the supervision of faculty member(s) at the end of the 4th Semester; a student will have to submit, through the respective supervisors, a dissertation on the project work. The project work will be assessed by a Board of Examiners consisting of Faculty members of the Department & External Examiner(s).

4.9 At the end of the 4th Semester, a student will have to appear at a Grand Viva-voce. The grand viva-voce will be conducted by a Board of Examiners consisting of Faculty members and External Examiner(s).

5.1 A candidate shall be eligible to appear at the Semester Examinations provided he/she is present in regular course of studies with proper attendance as per University rules.

5.2 The 2nd to 4th Semester classes will begin immediately after the completion of the previous semester examination.

5.3 All candidates who have completed a semester examination shall join the next semester classes. Candidates failing to qualify in a Semester examination shall automatically revert back to the respective semester in the next academic session immediately after publication of the result. However the candidate failing in a paper in the previous semester has to clear the paper(s) as per 5.4.

5.4 A candidate will get a maximum of three consecutive chances including the first one in his / her regular year in order to pass each of the Semester Examinations.

6. The final result (combining all the Semester results) will be determined by adding marks for all theoretical and practical papers separately. A candidate obtaining 40% will be in theoretical paper (i.e, 480 of 1200) and 50% or more in each of practical examination will be declared as passed with Second Class. A candidate scoring 60% or more marks in the total aggregate of all the Semester examinations will be placed in the First Class.
7. The course structure is as given in Appendix – I

Program Educational Objective(s)

After completing master and few years of masters, the Computer Science students would	
PEO I	Technical Expertise: Implement fundamental domain knowledge of core courses for developing effective computing solutions by incorporating creativity and logical reasoning.
PEO II	Successful Career: Deliver professional services with updated technologies in computer science based career.
PEO III	Soft Skills:Develop leadership skills and incorporate ethics, team work with effective communication & time management in the profession.
PEO IV	Life Long Learning: Conduct research among computing professional as per market needs.

Program Outcome(s)

Students will be able to

PO1: Apply knowledge of mathematics, science and algorithm in solving complex Computer engineering problems.

PO2: Generate solutions by conducting experiments and applying techniques to analyze and interpret data

PO3: Design component, or processes to meet the needs within realistic constraints.

PO4: Identify, formulate, and solve Software Engineering, Networking and Data Mining problems.

PO5: Comprehend professional and ethical responsibility in computing profession.

PO6: Express effective communication skills.

PO7: Participate in global, economic, environmental, and societal context.

PO8:Recognize the need for, and an ability to engage in life-long learning.

PO9: Knowledge of contemporary issues and emerging developments in computing profession.

PO10: Utilize the techniques, skills and modern computer Engineering tools, Software and techniques necessary for Engineering practice.

1PO11: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings.

PO12: Design research problems and conduct research in computing environment.

Mapping of PEO & PO

Programme Educational Objective(s)	Program Outcome(s)	
PEO I	Technical Expertise: Implement fundamental domain knowledge of core courses for developing effective computing solutions by incorporating creativity and logical reasoning.	1,2,4,7,8,10
PEO II	Successful Career: Deliver professional services with updated technologies in computer science based career.	3,5,6,11
PEO III	Soft Skills: Develop leadership skills and incorporate ethics, team work with effective communication & time management in the profession.	3,9
PEO IV	Life Long Learning: Conduct research among computing professionals per market needs.	12

Department of Computer Science
West Bengal State University, Barasat

M.Sc. (Computer Science)
(4 Semesters)

First Semester

CS101: ADVANCED COMPUTER ARCHITECTURE Full Marks:- 50

Objectives

The course should enable the student

1. To learn the basics of stored program concepts.
2. To learn the principles of pipelining.
3. To learn mechanism of data storage
4. To distinguish between the concepts of serial, parallel, pipeline architecture.

Outcomes

The Student should be able to

1. Learn pipelining concepts with a prior knowledge of stored program methods
2. Learn about memory hierarchy and mapping techniques.
3. Study of parallel architecture and interconnection network .

Computer Architecture & Organization. Control unit design, Basic Parallel Processing Architecture, Taxonomy- SISD. MISD, SIMD, MIMD structures, Serial, Parallel & Concurrent Computation, CISC Vs RISC, Structure of Instruction of instruction sets and Desirable Attributes. Basic Concepts of pipelining, Instruction Pipelining. Hazards, Reservation Tables, Collision, Latency, Dynamic pipeline, Vector processing & Vector processors. Cache Memory & Virtual Memory: Structure, Analysis & Design. I/O Systems: Design Issues, Performances Measures. Loosely Coupled & Tightly Coupled Systems, Concurrency & Synchronization, Scalability, Models of Consistency, Application of SIMD Structure. Definition. Types of Interconnected Networks; Baselines, Shuffle- Exchange, Omega, Cuba, Comparison & Application. Mapping Algorithm to array structures, Systolic processors. Mapping design & Optimization, Wave Front Array processor. Data Flow Graphs, Petri nets, Static & Dynamic DFA. Different Models, Languages, Compilers, dependency Analysis. Message Passing, Program mapping to Multiprocessors, Synchronization.
Case Study: Basic Features of Current Architectural Trends. DSP Processor, Dual core Technology

Text Book:

1. Kai Hwang, Advanced Computer Architecture: Parallelism, Scalability, Programmability, McGraw-Hill,
2. Kai Hwang, Fayé Alayé Briggs, Computer architecture and parallel processing, McGraw-Hill,

References

1. D.A. Patterson, J.L. Hennessy, D. Goldberg, Computer Architecture : A Quantitative Approach 2nd Edn, Addison- Wesley.
2. Harold Stone, High-performance Computer Architecture (3rd edition), Addison Wesley.
3. Naresh Jotwani, ADVANCED COMPUTER ARCHITECTURE- 2/E :Parallelism-Scalability- Programmabilit, Kai Hwang, Tata Mcgraw - Hill Education
4. P. V. S. Rao, Perspectives in Computer Architecture, PHI.
5. Hayes, Computer Architecture & Organization, 2nd & 3rd Edn

CS 102: ADVANCED DATABASE MANAGEMENT SYSTEMS Full Marks:- 50

Objectives

The course should enable the student

1. To make a study of SQL and relational database design.
2. To understand the internal storage structures using different file and indexing techniques which will help in physical DB design.
3. To know the fundamental concepts of transaction processing- concurrency control techniques and recovery procedure.
4. To have an introductory knowledge about the emerging trends in the area of distributed DB- OODB- Data mining and Data Warehousing.
5. To learn the basics of query evaluation and optimization techniques.

Outcomes

The Student should be able to

1. Explore the basic concepts of database systems.
2. Write SQL queries for a given scenario.
3. Describe relational database theory, and be able to write relational algebra expressions for queries.
4. Design logical data models
5. Evaluate and optimize queries
6. Implement transaction processing and concurrency control
7. Develop Object oriented dB, Distributed dB using XML, data warehousing

Query Processing, Query Optimization Algorithms. Transaction concepts, Recovery and Concurrency Control, Locking and Timestamp based protocols, Multiversion and Optimistic Concurrency Control schemes, Threats and countermeasures. Object-oriented and Object Relational Databases, Distributed Databases, Data Warehouse and Data Mining, Database Security, Emerging Technologies.

Text Book:

1. Elmasri,Navathe,Fundamentals of Database System,3/e,Pearson Education.

2. Ozsu,Principals of Distributed Database System,Pearson Education.

References

1. R. Chakrabarti, S. Dasgupta, ADVANCED DATABASE MANAGEMENT SYSTEM, Wiley
2. Carolyn Begg, Thomas Connolly, Database Systems, 4th Edition,Addison-Wesley
3. Dr. S. Sumathi, S. Esakkirajan , Fundamentals of Relational Database Management Systems, ISBN
4. Raghu Ramakrishnan , Database Management System 2nd Edition
5. Korth, Database Management System

CS 103: ADVANCED DATA STRUCTURES AND ALGORITHMS Full Marks: 50

Objectives

The course should enable the student

1. To introduce algorithms analysis and design techniques
2. To understand algorithms of various data structures used for searching, sorting,
3. indexing operation and dynamic programing

Outcomes

The Student should be able to

1. Ability in using the appropriate algorithm for searching, sorting, indexing operations
2. Designing of new algorithms
3. Analyzing complexity issues of algorithms

Basic concepts about Algorithms, Data Structures, Recursion, Iteration, Big-O Notation, Brief Foundations and Applications of Trees – Definitions, Representations, Binary Tree and Its Usefulness, Binary Search Tree, Tree Traversal, Threaded Binary Trees, Binary Tree Representation of any Tree other than Binary Tree, Decision Trees, Balanced Tree Schemes – AVL Trees, 2-3 Trees. Basic concepts about Searching, B-Trees, Hashing. complexity issues of different Sorting Algorithms. Binomial Heaps, Fibonacci Heaps, Amortized Analysis of Algorithms, Divide and Conquer algorithms: Multiplications of Large integers , Strassen's Matrix Multiplication algorithm. Dynamic Programming :shortest path, chained matrix multiplication, optimal binary search trees, Travelling salesman problem. Greedy Algorithm Knapsack problem. Computational Complexity and Intractability. Introduction to NP.

Text Book:

1. T. H. Cormen et al -Introduction to Algorithms, PHI
2. E.Horowitz, S. Sahani- Fundamentals of Computer Algorithms –Galgotia.

References

1. S. Sahani, Data Structures, Algorithms And Applications In C++ 2nd Edition, ORIENT BLACKSWAN PVT LTD
2. Robert Sedgewick and Philippe Flajolet, An Introduction to the Analysis of Algorithms, Addison-Wesley
3. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Data Structures and Algorithms, Addison-Wesley
4. Udi Manber, Introduction to Algorithms: A Creative Approach, Addison-Wesley
5. Thomas H. Cormen, Algorithms Unlocked, MIT

CS 104: COMPUTER NETWORKS Full Marks: 50

OBJECTIVES

The course should enable the student to

1. Understand the concepts of data communications.
2. Study the functions of the data link layer and to introduce IEEE standards employed in computer networking.
3. Learn the functions of the network layer and to get familiarized with the different protocols involved.
4. Learn the functions of the transport layer and to get familiarized with the different protocols involved.
5. Understand multiplexing , Domain name space and protocols.
6. Cryptography.

OUTCOMES

The Student should be able to

1. Describe various components and categories of data communications, types of connections, topologies, protocols and standards, various transmission media and modems.
2. Detect and correct the errors using various algorithmic techniques, be aware of the various Ethernet standards and bridges.
3. Explain various switching techniques used and implement the various routing and router protocols.
4. Illustrate multiplexing and demultiplexing, UDP, TCP protocols and Congestion Control mechanisms.
5. Illustrate Network Applications.
6. Security over cryptography.

Review on Computer Networks Basis Physical layer, Data Link control: Line discipline, Flow and error control protocols, Physical addressing, HDLC MAC Protocols: Dynamic channel allocation, Random access and Controlled access techniques, IEEE Standards. LAN Interconnection technologies and High Speed LANs, Virtual LANs. Virtual Circuit approach in WANs. IP address – subnetting, NAT, IP datagrams address mapping, error reporting and multicasting in network layer Static and Adaptive routing, Distance vector and Link-State routing, Broadcast routing, Unicast routing protocols: interior and exterior routing protocol. RIP, OSPF and BGP, Multicast routing protocols – Source-Based tree and Group-Shared tree approach. Reliable and Unreliable transport service, Flow and error control mechanism in transport layer. Congestion control and Quality of Service DNS, Electronic mail, FTP. Internet –introduction, addressing schemes, IPv4 and IPv6; World Wide Web. Protocols; HTTP, Telnet. FTP and other Net utilities;

Web mail. Netiquette. Searching: portals, search engines, concepts of crawlers, web mining. Information Theory: Measure of Information, Entropy, Discreate and Continious channel, Shannon's encoding algorithms.

Text Book:

1. B.Forouzan – Data Communication and Networking
2. A Tanenbaum – Computer Networks.

Reference

1. William Stallings, Computer Networking with Internet Protocols and Technology, TMH
2. James F. Kurose, Keith W Ross., Computer Networking: A Top-Down Approach Featuring the Internet,TMH
3. Poorna,Computer Networks, SCITECH
4. Olivier Bonaventure, Computer Networking : Principles,Protocols and Practice, The Saylor Foundation

Practical

CS 105(P): DATA STRUCTURES AND ALGORITHMS LABORATORY Full Marks: 50

Objectives

The course should enable the students to:

1. Realize the concept of Divide and Conquer algorithm design technique
2. Understand the concept of Dynamic Programming algorithm design technique
3. Realize the concept of Greedy algorithm design technique
4. Understand the concept of Backtracking algorithm design technique

Outcomes

At the end of the course the student should be able to:

1. Implement Divide and Conquer algorithm design technique for various applications
2. Implement dynamic programming algorithm design technique for various applications
3. Implement Greedy algorithm design technique for various applications
4. Implement backtracking algorithm design technique for various applications

Assignments on developing programs and functions related to the theoretical paper coverage on ANALYSIS OF ALGORITHMS.

AECC [CS 106(P)] : Programming with SQL Full Marks: 25

OBJECTIVES

The course should enable the students to:

1. Learn to write a query
2. Learn to design a simple dB using data modeling techniques
3. To provide the knowledge of various dB tools

OUTCOMES

At the end of the course the student should be able to:

1. Populate and query a database using SQL DML/DDL commands
2. Write programs using PL/SQL including stored procedures , cursors, packages etc.
3. Construct real time database application using current techniques

The SQL programing course is very important to prepare various software. This has immense importance now a days. This course will help students to development various software which in turn will increase their potential for employability and entrepreneurship. This is one of the state of art field of study in todays society.

Introduction tom SQL, Database Schema Design, Database Creation, SQL Programming and Report Generation using a RDBMS. Students are to be exposed to front-end development tools, ODBC; Internet based access to databases and database administration. Assignments on developing programs and functions related to the theoretical paper coverage on DATABASE LABORATORY.

Second Semester

CS 201: Object Oriented Programming Full Marks:- 50

OBJECTIVES

The course should enable the students to

1. Learn Object-oriented programming paradigm.
2. Understand advanced object-oriented features through JAVA programming language.
3. Learn exception handling.
4. Learn Generic classes and templates.
5. Multithreading and synchronization
6. Applet programming.
7. Event driven programming.

OUTCOMES

The student should be able to

1. Explain concepts in object oriented programming.

2. Write simple programs in JAVA.
3. Demonstrate the concept of functions, operator overloading, inheritance through JAVA programs.
4. Demonstrate the concepts of exception handling, generic functions, and templates.
5. Multi-threading and synchronization are included.
6. Applet programming.
7. Event driven programming.

Abstraction, Encaputulation, Modularity, Links and Association, Generalization, Inheritance, Aggregation, Polymorphism, using Instantiation, Metadata & Metaclass, Typing, Concurrency, Persistence; Events & States, Concurrency, Advanced Dynamic Model, Relation of Object and Dynamic Model. DFD, Constraints, Relation of Functional to Object and Dynamic Model. Analysis using Object, Dynamic and Functional Model. System Design: Subsystems, Concurrency, Allocating Subsystems to Processors & Tasks, Software Control Implementation, System Architecture Object Design: Combining three Models, Designing Algorithms, Design Optimization, Control Implementation, Design of Association, Packaging. Design Modeling using UML OO Languages Features, Survey of OO Languages, Multi method vs. Object Based vs. Class based languages, Java and C++, OO Data Model, Complex Object, Persistence, Transaction, Concurrency Control, OODB Architecture, Query Language for OO Relational Databases, Gemstone / O_2 / Orion CORBA

Text Book:

1. Ali Bahrami, - "Object –Oriented System Development" - Mc Graw Hill.
2. Rambaugh, James Michael, Blaha - "Object Oriented Modelling and Design" - Prentice Hall India

Reference:

1. Patrick Naughton, Herbert Schildt – "The complete reference-Java2" - TMH
2. Priestley – " Practical Object Oriented Design using UML" - TMH
3. Jana, C++ & Object Oriented Programming, PHI
4. Alhir, learning UML, SPD/O'Reily
5. E. balaguruswamy,object oriented programming in java
6. Buyya Buyya R., Object Oriented Programming with JAVA Essentials & Applications, TMH

CS 202: - ADVANCED OPERATING SYSTEM Full Marks :50

Objectives

The course should enable the student

1. Learn operating system design and concept
2. Understand process, process life cycle
3. Difference between different types of operating system

OUTCOMES

The student should be able to

1) The students will understand the design approaches of advanced operating systems
2) Analyze the design issues of distributed operating systems.
3) Evaluate design issues of multi processor operating systems.
4) Identify the requirements of operating systems.
5) Formulate the solutions to schedule the real time applications.

Review of operating system. Introduction to Parallel and Distributed Systems. State recovery and clock models for distributed systems. Classification of control algorithms for dist and parallel systems process and mode synchronization, classical OS, Process Migration, termination detection, Remote Procedure Call. Case study on various operating systems.

Text Book:

1. Tanenbum,A.S.,Distributed O/S ,Pearson Education.
2. Singhal,Shivaratri,Advanced Concepts In O/S,Tmh.

References:

1. P. K.Sinha,Distributed O/S,Phi
2. Balakrishna Prasad, Operating Systems & Systems Programming - 2nd Edn., Scitech
3. Avi Silberschatz ,Peter Baer Galvin ,Greg Gagne , Operating System Concepts *Eight Edition*
4. Allen B. Downey, Think Os A Brief Introduction To Operating Systems, Green Tea Press
5. G. Coulouris, J. Dollimore, T. Kindberg, G. Blair, Distributed Systems Concepts And Design 5th Edition

CS 203: - COMPUTER GRAPHICS AND MULTIMEDIA Full Marks:- 50

Objectives

The course should enable the student

1. Learn the rules and algorithms in generating graphical outputs.
2. Learn 3-dimensional objects using suitable transformations.
3. Understand the architecture for design of multimedia system.
4. Realize the issues related to multimedia file handling.
5. Understand hypermedia standards in
6. developing multimedia applications.

Outcomes

The Student should be able to

1. Develop algorithms to draw fundamental drawings
2. Develop real-time rendering graphics
3. Create 2D and 3D images
4. Have an understanding on the basics of creating multimedia applications
5. Design and Develop multimedia applications

Review of Computer Graphics, definitions of CG, types of CG, storage tubes displays, CRT technologies - Raster Scan Display, Computer graphics software. Points & lines, Line drawing algorithms; DDA algorithm, Bresenham's line algorithm, Circle generation algorithm; Ellipse generating algorithm; scan line polygon, fill algorithm, boundary fill algorithm, flood fill algorithm. Basic transformations: translation, rotation, scaling; Matrix representations & homogeneous coordinates, transformations between coordinate systems; reflection shear; Transformation of points, lines, parallel lines, intersecting lines. Viewing pipeline, Window to Viewport co-ordinate transformation, clipping operations, point clipping, line clipping, clipping circles, polygons & ellipse. 3D transformations: translation, rotation, scaling & other transformations. Rotation about an arbitrary axis in space; reflection through an arbitrary plane; general parallel projection transformation; clipping, Viewport clipping, 3D viewing, perspectives & Depth Cueing. Curve representation, surfaces, designs, Bezier curves, B-spline curves, end conditions for periodic B-spline curves, rational B-spline curves. Depth comparison, Z-buffer algorithm, Back face detection, BSP tree method, the Printer's algorithm, scan-line algorithm; Hidden line elimination, wire frame methods, fractal – geometry. Introduction, Modeling Light Intensities and Sources, Diffuse Reflection, Lambert's Cosine Law, Specular Reflection, Halftoning, Color Models - RGB Color, CMY Color. Multimedia: definition, characteristics- interactive and non-interactive; local (standalone CD, DVD) and networked (videoconferencing, web video broadcasting, multimedia Email); large data volume, real-time property, continuous display, delay requirement. Various media types: captured and. synthesized; discrete (space dimension) and continuous (space and time dimension); text- plain and rich; graphics (revisable) and images (not revisable); video (captured or synthesized); animation; sound -speech and non-speech, natural and structured. File formats- text (doc, RTF, PDF): audio (WAV, MIDI). Data compression and coding: entropy coding, lossy and lossless; text (run length; Huffman, arithmetic, vector, LZ, LZW); audio (Dolby), image and video standards- JPEG and. MPEG techniques.

Text Book:

1. D. P. Mukherjee, Fundamentals Of Computer Graphics And Multimedia, Phi
2. M.C. Trivedi, N.N. Jani, Kamaljit I. Lakhtaria & Gopal M. Dave, Computer Graphics & Animation, Jaico

Reference

1. John Dimarco, Computer Graphics And Multimedia: Applications, Problems And Solutions
2. A.P.Godse, D.A.Godse Computer Graphics And Multimedia
3. N. I. Badler, C. B. Phillips, B. L. Webber, Simulating Humans: Computer Graphics Animation And Control, Oxford
4. Vaka Murali Mohan, Computer Graphics, Scitech

CS 204: FORMAL LANGUAGE AND AUTOMATA THEORY Full Marks:- 50

Objectives

The course should enable the student

1. Introduce Formal Languages, Automata Theory, PDA, Turing Machine.
2. Learn Abstract models of Computation and Computability, Computational complexities and NP – Completeness.
3. Gain knowledge in computational theory.

Outcomes

The Student should be able to

1. Apply the theoretical concepts and techniques in designing finite automata
2. Convert regular expressions to FA and minimize Automata.
3. Write context free Grammar and design PDA for the Grammar.
4. Design turing machine and identify recursively enumerable languages.
5. Define undecidability and identify class P and NP problems.

Introduction and Review of Finite State Machines: Deterministic, Nondeterministic M/cs, Minimization of FSM, Inverse FSM. Regular Expression, properties applications : Definition, Regular Expression, Two way FA, Linear Bound Automata, Applications Regular Set: Definition, Properties, Pumping Lemma, Decision Algorithm, Minimization Grammar, Different types, Derivation Tree, Different Normal Forms, Ambiguous Grammar and its implications, Chomsky hierarchy, Context Sensitive Languages, Different Classes of Languages, Deterministic Context Free Language and its Properties Pushdown Automata: Definition, PDA and CFL, Alternative Forms of PDA Turing Machine: Introduction, Turing Machine Model, Church's Hypothesis Decidability and reccursively enumerable languages. Computability concept of Turing Machine.

Text Book:

1. Aho, Ulman, Hopcroft, Ajtometa, Pearson Education.
2. Zvi Kohavi, Switching and finite automata theory, McGraw-Hill

References:

1. Hopcroft, Introduction to Automata Theory, languages and Computation, 2/e,Pearson Education
2. K.P. Mishra, N. Chandrasekaran, Theory of Computer Science Automata, Languages And Computation Third Edition, Prentice Hall

Practical

CS 206(P): - Object Oriented Programming Laboratory Full Marks :50

Objectives

The course should enable the students to

1. Simple java programming
2. Multi-threading and synchronization programming
3. Package and interface programming
4. Inheritance
5. Polymorphism
6. Applet programming
7. Event driven programming

Outcomes

The students should be able to

1. Use platform independent programming for different application.
2. Use multithread and synchronization they can run real time programming.
3. Using applet they can design different APPs.
4. They can handle web page for different networking interface.

Programming with OOP : Pointers, Enumeration, References, Function Overloading, Classes and Objects, Constructors and Destructors, Self reference- This, Operator Overloading, Derived classes and Inheritance, Virtual Function, Virtual Base Class, Strings, Template, Exception Handling, Files & Streams, Standard Library, Header Files. Java:-Data types, Operators, Statements, Methods, Class declaration, Java Programming, Objects, Inheritance, Argument Passing, Arrays and Strings, I/O to Text Files.

SEC [CS 205(P)]: - Basics of Gaming Full Marks :25

The course should enable the student

1. Understanding basics of Game programing.
2. Understand algorithms to generate graphical outputs for games.
3. Programmability of 3-dimensional objects using suitable transformations.
4. Understand the architecture for design of multimedia system.
5. Realize the issues related to multimedia file handling.

Outcomes

The students should be able to

1 Make use of algorithms to draw 2D and 3D objects
2 Show transformations and projections for 2D and 3D objects
3 Manipulate a graphical object using clipping algorithms and viewing technique
4 Use an image editing tool for image manipulation and enhancement
5 Utilize the authoring tool to develop a 3D scene and to perform 2D animation

The programing for construction of game plan to prepare various software for playing games has an immense importance now a days. This course will help students to development various game which in turn will increase their potential for employability and entrepreneurship. This is a state of art field of study in todays society.

Assignments on developing programs and functions related to the theoretical paper coverage on COMPUTER GRAPHICS.

List Of Experiments

1. Program to implement Line using "DDA" algorithm.
2. Program to implement line using Bresenham's line drawing algorithm.
3. Program to implement circle using Midpoint algorithm.
4. Program to implement Circle using Bresenham’s Circle Drawing algorithm.
5. Write a c++ program that implement Boundary Fill algorithm?
6. Write a c++ program that implement Shearing algorithm?
7. Program to implement Translation of the Square.
8. Program to implement Rotation of square.
9. Program to implement Reflection.
10. Game design.

Third Semester

CS 301: - SOFTWARE ENGINEERING Full Marks:- 50

Objectives

The course should enable the student to

1. Understand the software life cycle models
2. Learn Requirement analysis and design concepts
3. Learn various software construction technologies.
4. Acquire knowledge on testing.
5. Understand the importance of SCM and release management.

Outcomes

The student should be able to
1. Apply the concepts of life cycle models to choose the appropriate model.
2. Analyse the requirements and design the software.
3. Construct or implement the software based on the industry standards
4. Design and develop test cases
5. Work with version control and work on configuration and release management plans

A generic view. Software architecture, Review of Software Development stages- analysis , design, implementation, testing .Program verification. Module relationship- Coupling, Cohesion. Effort Estimation models . Project Scheduling and project management Risk management .Software Maintenance. Software Quality Models. Software Reliability –Basics, Time-dependent and Time-independent models. Software metric. Software Configuration management . Object- oriented software Engineering. Unified Modelling Languages – features and case study. software reuse, emerging trends.

Text Book:
1. Roger S Pressman- Software Engineering.
2. Rajib Mall, Fundamentals of Software Engineering,PHI

Reference
1. Ian Somerville – Software Engineering.
2. P Jalote – An Integrated Approach to Software Engineering.
3. Pratap K.J. Mohapatra, Software Engineering, New Age International Publishers
4. Datta, Software Engineering ,Oxford
5. Behforooz,Hudson, Software Engineering Fundamentals, Oxford
6. Jawadekar, Software Engineering: Prime, TMH

CS 302: Compiler Design Full Marks:- 50

OBJECTIVES
The course should enable the students to
1. Understand the basic principles of the compiler, Compiler construction tools and lexical analysis.
2. Learn the context of Context Free Grammar, Parsing and various parsing techniques.
3. Learn the process of intermediate code
Generation.
4. Learn the process of Code Generation and various Code optimization techniques.
5. Understand need of Program verification and Lambda calculus in verification.

OUTCOMES
The student should be able to
1. Differentiate the various phases of a compiler.
2. Apply parsing techniques and able to write Context Free Grammars for various languages.
3. Design the structure of intermediate code for various types of statements and expressions.
4. Design code generator and apply code optimization techniques.
5. Can design own compiler of any work specific application.

Compiler design, various phases; lexical analyzer, token, lexeme, and patterns. Regular definitions, Transition Diagrams, Syntax Analysis, ambiguity, associatively, precedence, Top down Parsing, recursive-descent parsing, predictive parsing, Bottom up Parsing, Operator precedence grammar, LR parsers Syntax directed definitions: inherited and synthesized attributes. Type checking, Symbol Tables. Runtime systems, Activation tree, Activation record, Basic Blocks, Dataflow analysis, Code optimization and code generation.

Text Book:
1. Aho, Compilers: Principals, Techniques and Tools, Pearson Education.
2. Muneeswaran, Compiler Design,Oxford

References:
1. Sudha Sadasivam, Compiler Design - 2nd Edn., SCITECH
2. Niklaus Wirth, Theory and Techniques of Compiler Construction, Addison-Wesley

CS 303: ARTIFICIAL INTELLIGENCE Full Marks:- 50

Objectives
The course should enable the student
1. To provide the foundations for AI problem solving techniques and knowledge representation formalisms

Outcomes
The Student should be able to
1. Ability to identify and formulate appropriate AI methods for solving a problem
2. Ability to implement AI algorithms
3. Ability to compare different AI algorithms in terms of design issues, computational complexity, and assumptions

Importance of Al, Scope of Al, Goals of AI, AI and Related fields, State-Space Graphs, Implicit and Explicit Graphs, Production Systems, Formulating the State-Space; Uniformed search: Depth-first Search, Breadth-first Search; Uniform Cost algorithm; Use of Heuristics, A* Algorithm, Admissibility of A*; Analysis and comparison of Search algorithms;
Two-agent games, AND/OR Graphs, Minimax Procedure, α-β pruning procedure, Learning evaluation functions; Introduction to ES, Knowledge-Based systems, Knowledge Representation: Rule_Based approach: Forward and Backward Chaining, Semantic-Nets Based approach, Frame Based approach; Introduction to Constrained Satisfaction Problems(CSP), Applications, Algorithms to CSPs, Symbolic constraints & Propagation; Introduction to programming in logic. Declarative and Procedural Meaning, Data Objects, Lists, Operators, Controlled Backtracking.
Soft computing basics. Fuzzy Systems: Fuzzy sets, Fuzzy logic. Fuzzy relations, Approximate Reasoning, Fuzzy logic control systems. Artificial Neural Networks: Feedforward Networks and Supervised Learning Perception learning rules, Adaline, Back propagation. Unsupervised Learning Networks. Genetic Algorithm (GA): Evolutionary Computing. Basics of Genetic Algorithms Reproduction, Crossover Mutation, Schemata, Finess function.

Text Book:
1. E. Rich and K. Knight: Artificial Intelligence, TMH
2. Dan W. Patterson: Introduction to Artificial Intelligence and Expert Systems

Reference:
1. S. Russel and P. Norvig, "Artificial Intelligence, A modern Approach"
2. Cloksin & Mellish , Programming In Prolog ,Narosa Publishing House
3. Nillson, Principles Of Artificial Intelligence, Harcourt Asia & Morgan
4. Janakiraman, Sarukesi & Gopal Krishnan , Foundation Of Artificial Intelligence & Expert System, Macmillan

CS 304: Elective 1

Practical

CS 306 (P) : SEMINAR Total Marks: 25

Presentation on assigned topics.

OBJECTIVES
The course should enable the student to
1. Learn the concepts of Seminar
2. Learn techniques and approaches that is specific to the Seminar.
3. Able to analyze and understanding Soft Skill & presentation
4. Prepare the seminar in any recent topic.

OUTCOMES
The student should be able to
1. Have gained the knowledge in Seminar.
2. Have interdisciplinary knowledge.
3. Expose in Research and Development.
4. Be confident in Seminar Soft Skill & presentation.

CS 307 (P) : TERM PAPER LEADING TO PROJECT Total Marks: 25

Initial survey and presentation on the topic selected for project.

OBJECTIVES
The course should enable the student to
1. Learn the concepts of Project
2. Learn techniques and approaches that is specific to the Project.
3. Able to analyze and understanding the project

OUTCOMES
The student should be able to
1. Have gained the knowledge in Seminar and learn the basis of Project.
2. Have interdisciplinary knowledge.
3. Expose in Research and Development.
4. Be confident in Project.

CMSPGEC01T: **FUNDAMENTALS OF COMPUTERS** Total Marks: 50

Objective: The objective of the paper is to facilitate the student with applied working knowledge of computers. This is a generic course on computing and does not assume any pre-requisite.

UNIT I

Five Component Model of a Computer, System and Application software (introduction) storage devices , primary (RAM, ROM, PROM, EPROM, cache) Memory and secondary (magnetic tape, hard disk, Compact disks) memory , peripheral devices , printers. **(8 Hours)**

UNIT II

Operating Systems: DOS Internal, External commands, Windows (2000 and NT) , Overview of architecture of Windows , tools and system utilities including registry , partioning of hard disk , Overview of Linux architecture, File system , file and permissions , concept of user and group , installation of rpm and deb based packages.

(8 Hours)

UNIT III

Basics of programming through flow chart , Networking Basics - Uses of a network and Common types of networks , Network topologies and protocols , Network media and hardware , Overview of Database Management System.

(8 Hours)

UNIT IV

Office Writer : Editing and Reviewing, Drawing, Tables, Graphs, Templates
Office Calc : Worksheet Management , Formulas, Functions, Charts
Office Impress: designing powerful power-point presentation
Internet Security: Security, Privacy Ethical Issues & Cyber Law **(8 Hours)**

Text:

[1] Peter Norton, Introduction to computers, Sixth Edition Tata McGraw Hill (2007).
[2] Andrews Jean, A+Guide to Managing & Maintaining Your PC, Cengage Publication 6/e

References:

[R\1] Anita Goel, Computer Fundamentals, Pearson Education.
[2] Joiner Associates Staff, Flowcharts: Plain & Simple: Learning & Application Guide , Oriel Inc
[3] http://www.openoffice.org/why/
[4] http://www.libreoffice.org/get-help/documentation/

Fourth Semester

CS 401 Elective-II Total Marks: 50

CS402 Elective-III Total Marks: 50

Practical

CS 305(P): SOFTWARE ENGINEERING LABORATORY Full Marks:- 50

Objectives
The course should enable the student to
1. Develop prototype for software life cycle models
2. Understand requirement analysis and design concepts
3. Understand testing.
4. Understand the importance of SCM and release management.

Outcomes
The Student should be able to
1.Explain the software engineering process and project management
2 Demonstrate software requirements and analysis using UML
3 Outline the software design process and user interface
4 Compare and contrast various software testing
5 Discuss about the software integration and project management
6.Use testing tools to do software testing.
7.Use version control tools and create build files

Design and development of Softwares- Application and System Softwares. e.g. Railway Reservation System , Examination System, Student Registration System, Problems on compilation, Entity relationship. Designing of test data for testing procedural and object-oriented programs. Design and development of software for measurement of quality attributes of software. Implementation of use-case diagrams and related notations Assignments on developing programs and functions related to the theoretical paper coverage on SOFTWARE ENGINEERING.

CS 403 (P): Grand Viva – Vice Total Marks: 50

Objectives
The course should enable the students to
1. Able to evaluate overall technical knowledge and industry readiness.

2. Able to go under a virtual environment of technical interview.
3. Able to analyze various application of Computer Science & Engineering in real life problem solving.
4. Able to analyze student's learning and understanding during the course of their post graduate programmed.
5. Prepare the students to face interview both at the academic and the industrial sector

Outcomes
The students should be able to
1. Face any interviews.
2. Interdisciplinary knowledge .
3. Research and development.
4. Exposure to real world.

CS 404 (P): Project Total Marks: 150

Objectives
The course should enable the student
1. To gain hands on experience on innovative technology project
2. To prepare the students to solve/work on the real world/practical/theoretical problems involving issues in computer science and engineering

Outcomes
The Student should be able to
1. Ability to design and model a system
2. Ability to plan and execute well defined objective
3. Ability to work in team at component level and system level
4. Ability to troubleshoot
5. Ability to reuse- or integrate with- existing components
6. Ability to derive performance metrics and assess quantitatively the performance of system
7. Ability to report and present the findings in standard formats

Elective Papers:

Elective 1:
1.1 VLSI DESIGN
Introduction to VLSI System Design: MOS Devices, Circuits and Fabrication, Design Principles and Characteristics of MOS Devices in Logic Circuits, Logic Implementation with nMOS, pMOS, CMOS and PLAs, Pass and Transistor Logic, Size and Complexity of Integrated Circuits, Feature Size, Impact of Shrinking, Clocking, Scaling, PLA Minimization and Folding, Inverters and Logic Gates, Design Rules and Layouts, Stick Diagram, Transistor Sizing. Logic Design: Static nMOS and CMOS Circuits, Steering Logic, Dynamic CMOS Circuits, Static vs. Dynamic CMOS Designs, Domino and NORA Logic Circuits, Charge Sharing, Clock Generation and Distribution, Transmission Gates. VLSI Design Process: System Specification, Functional Design, Logic Design, Circuit Design, Physical Design, Verification, Fabrication and Packaging. Design Styles: Custom Design, Standard-Cell Design, Gate-Array Design, FPGA and MCMs. Physical Design Issues: Partitioning, Floor-Planning and Placement, Routing, Compaction, Complexity Issues, Algorithms and Data Structures for Layout Designs.
Text Book:
1. Pucknell D.A and Eshraghian K "Basic VLSI Design"
2. Michael John Sebastian Smith "Application-Specific Integrated Circuits "

Reference:
1. Keshab K. Parhi - VLSI Digital Signal Processing Systems: Design and Implementation
2. Wolf - Modern VLSI Design - System - on - Chip Design, 3ed
3. Jayaram Bhasker - A VHDL Primer (3rd Edition)
4. N.SHERWANI, KLUWER - "ALGORITHM FOR VLSI DESIGN & AUTOMATION".

1.2 CRYPTOGRAPHY AND NETWORK SECURITY
Foundations of Cryptography and Security. Principles of Security. Types of Attacks. Cryptographic Techniques: Design Principal of Block Ciphers, Block Cipher Algorithms. Steganography. Computer Based Symmetric Key Cryptographic Algorithms: Data Encryption Standard (DES). International Data Encryption Algorithm (IDEA). Computer Based Asymmetric Key Cryptographic Algorithms: RSA Algorithm. Hashes and Message Digests, Digital Signatures, Certificates and standards, Authentication, Electronic Mail Security, IP and Web Security Protocols, System Security: Computer Virus, Firewall and Intrusion Detection.
Text Book:
1. Atul Kahate "Cryptography and Netwrok Security"
2. A Kahate and Godbole "Web Technologies"

Reference:
1. William Stallings, "Network Security Essentials"
2. Gollmann, Dieter, "Computer Security"
3. Micki Krause, Harold F. Tipton, "Handbook of Information Security Management"
4. Pearlman and Kaufman "Private Communication in a Public World"
5. Behrouz A. Forouzan, Debdeep Mukhopadhyay., Cryptography and Network Security, TMH
6. William Stallings, Cryptography and Network Security, TMH

7. Kahate, Cryptography and Network Security 3rd edition, McGrawHill
8. RAJARAM, NETWORK SECURITY AND CRYPTOGRAPHY, SCITECH

1.3 Cloud Computing

Introduction – Component of CC – Comparing CC with Virtualization, Grids, Utility Computing, clientserver model, P-to-P Computing – Impact of CC on Business – Key Drivers for Cloud Computing - Cloud computing Service delivery model
Cloud Types – Private, Public and Hybrid, when to avoid public cloud, Cloud API
Introduction & benefit of Virtualization – Implementation Levels of Virtualization- VMM Deisgn Requirements and Providers – Virtualization at OS level – Middleware support for Virtualization – Virtualization structure/tools and mechanisms: Hypervisor and Xen Architecture, Binary Translation with full Virtualization, Para Virtualization with Compiler Support –
Virtualization fo CPU, Memory and I/O Devices, Hardware support for Virtualization in intel x86 processor – CPU Virtualization – Memory Virtualization and I/O Virtualization – Virtualization in Multicore processors.
XaaS, IaaS, PaaS- Leveraging PaaS for Productivity- Languages for PaaS- DBaaS(Database as a services) – SaaS (Software as a service) – Comparison of various cloud computing providers/ Softwares
Key Business Drivers for CC- Cloud computing and out sourcing – Types of Scalability – Security issues in Cloud Computing- time to Market Benefits- Distribution over Internet – Three levels of Business value from Cloud computing.
Eucalyptus and Open Stack Architecture Features – Components – Various mode of operations – Installation and configuration process of both open source – Cloud Administration and Management Task – Creating User Interface (Web Interface) of Private cloud.
Factors for Successful Cloud Deployment – Network Requirements – Potential Problem areas in a cloud Network and their Mitigation – Cloud Network Topologies – Automation and Self-service feature in a cloud –cloud performance.
Security for Virtualization Platform – Host security for SaaS, PaaS and IaaS – Data Security – Data Security Concerns – Data Confidentiality and Encryption – Data Availability – Data Integrity – Cloud Storage Gateways – Cloud Firewall.
Cloud Application requirements- Architecture for traditional Vs Cloud Applications- Multi-ties Application ArchitectureSOA for Cloud applications – Resource oriented SOA – Method –oriented SOA and Event Driven SOA – Parallelization within Cloud Applications – Leveraging Inmemory Operations for Cloud Application.
Programming Support for Google Apps engine: GFS, Big Tables, Googles NO SQL System, Chubby, Google Distibuted Lock Service, Programming Support for Amazon EC2: Amazon S3, EBS and Simple DB etc.
Adoption of Public cloud by SMBs- Public Cloud Adoption phase for SMBs- Vendor liability and Management Adoption process of Public clouds by Enterprises – Managed Private clouds Migrating Application to the cloud – Impact of Shared Resources and Multi-Tenancy on cloud Applications – Phases during Migration an Application to An IaaS Cloud
Risk Assessment and Management – Rosk of Vendor Lock-in – Risk of Loss of control over IT services- Risk of Poor Provisioning – Risk of Multi-tenant environment – Risk failure of cloud provider – SLA risk –security, malware and Internet Attacks – Risk with Application Licensing.
AAA model – SSO for Clouds – Authentication management and Authorization management in clouds – Accounting for Resource utilization.
What can security as service offer- Benefits for Security as a service – Issues with Security as a Service- Identity Management as a Service

Text Book

1. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley,2011

Reference

1. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
2. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010
3. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010

1.4 OPERATION RESEARCH

Resource Allocation - graphical solutions of two-product, multiple-resource production environments, simplex method of solution, linear programming (using the Solver plug-in for MS Excel) Network Analysis and Design - Euler and Hamilton circuits, minimum traversal (i.e. shortest paths) algorithms, Dijkstra's method, spanning trees, Kruskal's method, and maximum flow networks. Planning and Task Scheduling - list-processing algorithm, critical paths, critical path method (CPM), PERT, Hargrove and Nemhauser's method, EOQ model Forecasting Techniques - moving average, exponential smoothing, regression Deterministic Inventory models - classic EOQ, EOQ with bulk purchasing, EOQ with storage limitations Transportation models - least cost method, NW corner method, stepping stone method, Vogel's approximation

TEXT BOOK:

6. Operations Research - Ronald Rardin, PHI
7. Operations Research: Applications and Algorithms,Wayne L. Winston, third edition, 1994

1.5 EMBEDDED SYSTEM

Embedded Computing: Complex System and Microprocessors- Embedding Computers, Embedded System Design Process Requirements, Specifications, Design and Integration, formalisms for System Design, Design Example Instruction Sets: Preliminaries, ARM and SHARC Processor - Processor and Memory Organization, Data Operations. Flow of Control, Parallelism within Instruction. CPU: Programming I/O, Supervisor Mode. Exception and Traps, Co-processors, Memory System Mechanisms, Performance Pipelining. Super Scalar Execution, Caching, CPU Power Consumption, Design Example. The Computing System: CPU bus-protocol, DMA, Configurations, Examples, Memory Devices- Organization, RAM. ROM. I/O Devices- Timers, Counters,

A/D and4)/A Converters, Keyboards, LEDs, Display, Touch Screen, Interfacing Memory and Device, Microprocessor-based Design-Architecture. Hardware Design, Development and Debugging, Manufacturing Testing, Design Example Program Design and Analysis: Design Patterns, Models, Assembling and Linking, Compilation Techniques, Interpreters and JIT Compilers, Analysis and Optimization- Execution Time, Energy and Power, Program Size, Validation and Vesting, Safety-critical System, Design Example

Text Book:

1. Dreamtech Software Team, "Programming for Embedded Systems: Cracking the Codes"
2. John Catsoulis, " Designing Embedded Hardware"

Reference:

1. Daniel Wesley lewis, "Fundamentals of Embedded Software: Where C & Assembly Meet"
2. Das, Embedded System , Pearson

2.1 Graph Theory

What is graph – Application of graphs finite and infinite graphs – Incidence and Degree – Isolated vertex, pendent vertex and Null graph. Paths and circuits – Isomorphism, sub graphs, walks, paths and circuits, Connected graphs, disconnect graphs.

Euler graphs, Hamiltonian paths and circuits, Dirac's theorem for Hamiltonicity, Travelling salesman problem. Directed graphs types of digraphs, Digraphs and binary relation.

Trees – properties, pendent vertex, Distance and centres - Rooted and binary tree, counting trees, spanning trees.

Vertex Connectivity, Edge Connectivity, Cut set and Cut Vertices, Fundamental circuits, Planar graphs, Different representation of planar graphs, Euler's theorem, Geometric dual, Combinatorial dual.

Matrix representation of graphs- Adjacency matrix, Incidence Matrix, Circuit matrix, Fundamental Circuit matrix and Rank, Cut set matrix, Path matrix

Graphs theoretic algorithms - Algorithm for computer representation of a graph, algorithm for connectedness and components, spanning tree, shortest path.

Text Books

1. Douglas B. West, Introduction to Graph Theory, Prentice Hall India Ltd., 2001
2. Narasingh Deo, Graph theory, PHI, 1979.

References

1. R. Diestel, Graph Theory, free online edition, 2016: diestel-graph-theory.com/basic.html
2. Robin J. Wilson, Introduction to Graph Theory, Longman Group Ltd., 2010
3. N. Alon and J. Spenser, "Probabilistic Methods", John Wiley and Sons, 2nd edition, 2000.

2.2. PATTERN RECOGNITION

Pattern Recognition: Introduction, decision boundaries, discriminant functions (linear and non-linear), Bayesian classification, training and test sets, parametric and non-parametric learning, minimum distance classifiers, k-NN rule, unsupervised learning, basic hierarchical and non-hierarchical clustering algorithms, dimensionality reduction, similarity measures, feature selection criteria and algorithms, principal components analysis, some applications.

Text Book:

1. M. K. Pkhira, Digital Image Processing And Pattern Recognition PHI.
2. R. O. Duda, P. E. Hart and D. G. Stork: Pattern Classification and Scene Analysis, 2nd ed., Wiley, New York.

Reference:

1. J. T. Tou and R. C. Gonzalez: Pattern Recognition Principles, Addison-Wesley, London, 1974.
2. Frank Y. Shih, Image Processing and Pattern Recognition: Fundamentals and Techniques, Wiley-IEEE Pres
3. Bishop, Neural Networks for pattern recognition, Oxford

2.3 NATURAL LANGUAGE PROCESSING

Parsing & Grammar - Lexical Functional Grammar, Tree Adjoining Grammar, Government & Binding, Paninian Grammar. Comparison of Paninian Grammar with others.

- Semantic Interpretation - Logical Semantics, Script, Conceptual Dependency.
- Discourse Interpretation - Paragraph, Story, Dialogue understanding. Anaphora Resolution.
- Natural Language Generation.
- Machine Translation with special reference to Indian Languages.
- NLP systems - Natural Language Interfaces to Databases.

Text Book:

1. Grasz, Jones & Webber (Ed.): Readings in Natural Language Processing, Morgan Kaufmann, 1986.
2. Gazdar & Mellish: Natural Language Processing in PROLOG, Addision Wesley, 1989.

References

1. Leonard Bolc. (Ed.): Natural Language Parsing Systems, Springer Verlag, 1987.
2. McDonald & Bolc. (Ed.): Natural Language Generation Systems, Springer Verlag, 1987.

3. W. J. Hutchins: Machine Translation - Past, Present & Future, Ellis Horwood, 1986.
4. Bharati, Chaitanya and Sangal: Natural Language Processing- a Paninian perspective, PHI, 1985.

2.4 SOFT COMPUTING

Fuzzy Systems: Fuzzy sets, Fuzzy logic. Fuzzy relations, Approximate Reasoning, Fuzzy logic control systems Applications of Fuzzy Theory: Fuzzy Pattern Recognition. Fuzzy Database Human Maching Interactions
Artificial Neural Networks: Feedforward Networks and Supervised Learning Perception learning rules, Adaline, Back propagation. Associative Memories, Hopfield networks, Unsupervised Learning Networks, Self-organizing feature map. Adaptive Resonance Theory, Radial Basis function. Recurrent Neural, Networks Reinforcement Learning Applications of Neural Networks Sensor processing. Communication. System Identification and Control Genetic Algorithm (GA):Evolutionary Computing. Basics of Genetic Algorithms Reproduction, Crossover Mutation, Schemata, Finess function. Optimization problems with Constraints, Stochastic models Applications orGA:GA in Machine Learning, Navigational Planning for Robots, GA in Optimization Problems, Intelligent Search Integrated Systems. Fuzzy Neural Systems for Pattern Recognition, Neural Fuzzy Controllers, Neural Network-driven Fuzzy Reasoning

Text Book:

1. Jang, Sun, Mizutani, "Neuro-Fuzzy and Soft computing", Pearson
2. Haykin, "Neural networks: a comprehensive foundation", Pearson

Reference

1. Goldberg, "Genetic Algorithms", Pearson
2. G.J. Klir & B. Yuan, "Fuzzy Sets & Fuzzy Logic", PHI.

2.5 COMPUTER VISION

Fundamentals of Image Processing Binary Image Analysis Pattern Recognition Concept Filtering and Enhancing Images Color, Shading and Texture Content-Based Image Retrieval Motion from 2D Image Sequence Image Segmentation Matching in 2D Perceiving 3D from 2D Images 3D Sensing and Object Pose Computation Models and Matching in 3D Virtual Reality Integration of a Machine Vision System Case Studies

Text Book:

1. Davis, E. R. 1997. Machine Vision. 2nd Ed. San Diego, California: Academic Press.
2. Jain, R. J., R. Kasturi and B. G. Schunck. 1995. Machine Vision. New York: McGraw-Hill, Inc.

Reference:

1. Haralick, R. M. and L. G. Shapiro. 1992. Computer and Robot Vision. Vol. 1 & 2. Reading, Massachusetts: Addison-Wesley Publishing Company, Inc.
2. Faugeras, O. 1999. Three-Dimensional Computer Vision: A Geometric Viewpoint. Cambridge, Massachusetts: The MIT Press.

3.1 MOBILE COMPUTING

Mobile Computing Vs wireless Networking – Mobile Computing Applications – Characteristics of Mobile computing – Structure of Mobile Computing Application. MAC Protocols – Wireless MAC Issues – Fixed Assignment Schemes – Random Assignment Schemes – Reservation Based Schemes.
Overview of Mobile IP – Features of Mobile IP – Key Mechanism in Mobile IP – route Optimization. Overview of TCP/IP – Architecture of TCP/IP- Adaptation of TCP Window – Improvement in TCP Performance.
Global System for Mobile Communication (GSM) – General Packet Radio Service (GPRS) – Universal Mobile Telecommunication System (UMTS).
Ad-Hoc Basic Concepts – Characteristics – Applications – Design Issues – Routing – Essential of Traditional Routing Protocols – Popular Routing Protocols – Vehicular Ad Hoc networks (VANET) – MANET Vs VANET – Security.
Device Operating Systems – Special Constrains & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Windows Phone – MCommerce – Structure – Pros & Cons – Mobile Payment System – Security Issues.

Text Book:

1. J. Schiller – Mobile Communication

References:

1. V.K.Garg & J.E.Wilks:Wireless and Personal Communication Systems: Fundamentals and Applications, IEEE Press and Prentice Hall,1996.
2. T.S.Rappaport, B.D.Woerner and J.H. Reed:Wireless Personal Communications: The Evolution of PCS, Dkyener Academic, 1996.
3. G.I. Stuber: Principles of Mobile Communication,Kluener Academic,1996.
4. U.Black:Mobile and Wireless Networks, Prentice Hall PTR,1996.
5. Charles Parkins – Mobile Adhoc Ntworks
6. W. Stallings Wireless- Communication

3.2 Fuzzy Logic

Module I : Biological foundations to intelligent systems I: Artificial neural networks, Back-propagation networks, Radial basis function networks, and recurrent networks.
Module II : Biological foundations to intelligent systems II: Fuzzy logic, knowledge representation and inference mechanism, genetic algorithm, and fuzzy neural networks.

Module III : Fuzzy and expert control (standard, Takagi-Sugeno, mathematical characterizations, design example), Parametric optimization of fuzzy logic controller using genetic algorithm.
Module IV : System identification using neural and fuzzy neural networks.
Module V : Stability analysis: Lyapunov stability theory and Passivity Theory.
Module VI : Adaptive control using neural and fuzzy neural networks, Direct and Indirect adaptive control, and Self-tuning Pill Controllers.
Module VII : Applications to pH reactor control, flight control, robot manipulator dynamic control, underactuated systems such as inverted pendulum and inertia wheel pendulum control and visual motor coordination.

Text Book:

1. Stanislaw H. Zak, Systems and Control, Oxford University Press, 2003

Reference Books

1. A.S. Poznyak, E. N. Sanchez and Wen Yu, Differential Neural Networks for Robust Nonlinear Control, World Scientific, 2001.
2. Kevin M. Passino and Stephen Yurkovich, Fuzzy Control, Addison Wesley Longman, Menlo Park, CA, 1998.

3.3 Data Warehousing and Data Mining

Need for strategic information, Decision support system, Knowledge discovery & decision making, need for data warehouse, definitions of Data warehousing and data mining, common characteristics of Data warehouse, Data Marts, Metadata, Operational versus analytical databases, trends and planning of Data warehousing. Defining business requirements, Data modeling strategy, Fact tables, dimensions, Star schema and other schemas, Multi dimensional data models, Data Cube presentation of fact tables, using the Data warehouse, Designing tools for Data warehouse, OLAP models and operations. Architectural components, Infrastructure: Operational & Physical, Extraction, Transformation and Loading, Components of an Oracle Data warehouse, Data Transformation Functions, DBA responsibilities, Capacity Planning. Implementation of Data warehouse, Physical design: steps, considerations, physical storage, indexing, Performance Optimization, Data warehouse deployment activities, Data security, backup and recovery concepts, Data warehouse Maintenance. Basics of data mining, related concepts, Data mining techniques, Data Mining Algorithms -- Classification, Clustering, and Association rules, Knowledge Discovery in databases(KDD) Process, Introduction to Web Mining:
Text Book:

1. Data Warehousing Fundamentals , by Paulraj Ponnian, John Wiley.
2. Data warehousing with oracle by sima yazdani – shirley s. Wong

Reference:

1. Data Mining Concepts and Techniques, Han Kamber, Morgan Kaufmann
2. C. S. Jensen,Torben Bach Pedersen,Christian Thomsen Introduction to Business Intelligence and Data Warehousing, PHI
3. Ralph Kimball, John Wiley The Data Warehouse Life-cycle toolkit,
4. Berson, Data warehousing, Data mining & OLAP, TMH
5. Tan, Introduction to Data Mining, Pearson

3.4. Quantum Computing

Quantum computing has emerged about a decade ago as a branch of theoretical computer science, with more and more connections to classical computer science. In this course we aim to give a basic introduction to this exciting field, giving students the basis to undertake research in this area, to integrate it into related areas, or simply to gain a deeper understanding of what quantum computing is. We will study: axioms of quantum mechanics, quantum circuits, quantum algorithms (up to Shor's algorithm for factoring), quantum error correction, quantum complexity (classes, complete problems etc.), some quantum cryptography, quantum communication complexity, some classical results obtained "the quantum way".
Text Book:

1. Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambr. Univ. Press
2. A. Yu. Kitaev, A. H. Shen, M. N. Vyalyi, Classical and Quantum Computation, Amer. Mathematical Society

Reference:

3. Julia Kempe,Quantum Algorithms,
4. E. Kushilevitz and N. Nisan, Communication Complexity, Cambr. Univ. Press (1997)
5. Ronald de Wolf, Quantum Communication and Complexity,
6. Julia Kempe, Approaches to Quantum Error Correction, survey (2006)
7. Sahni, Quantum Computing, TMH

3.5 Parallel Algorithms

Overview, need for parallel computing, basic concepts and terminology -Flynn's classical taxonomy, general parallel terminologies, issues in high performance computing Architecture and interconnection of parallel computers: Memory architectures -shared memory, distributed memory, hybrid distributed-shared memory. Interconnection networks Parallel Programming Models: Overview, shared memory model, threads model, message passing model, data parallel model, advanced Models Designing Parallel Algorithms: Automatic vs. manual parallelization. partitioning, communications, synchronization, data dependencies, load balancing, granularity, limits and costs of parallel programming, performance analysis and tuning Parallel computing examples: array processing, PI calculation, simple heat equation, matrix vector multiplication, matrix-matrix multiplication, combinational search
Text Book:

1. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar "Introduction to Parallel Computing", Second Edition, Addison Wesley, 2003. ISBN: 0-201-64865.
2. S.G.Akl, "The Design and Analysis of Parallel Algorithms", PHI, 1989.

REFERENCES

1. F.T.Leighton, "Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes", MK Publishers, San Mateo California, 1992.
2. Wilkinson, M.Allen,"Parallel Programming Techniques and Applications using networked workstations and parallel computers", Prentice Hall, 1999.
3. Michael J. Quinn, "Parallel computer theory and practice", McGraw Hill, Second Edition, 1994.
4. S. Rajasekaran and J. Reif, Handbook of Parallel Computing: Models, Algorithms and Applications, Chapman and Hall/CRC, 2008.
5. J'aJ'a, J., An Introduction to Parallel Algorithms, Addison-Wesley Pub Co, Reading, MA, 1992.
6. F. T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, Morgan Kaufmann, CA 1992.
7. J. H. Rief, Synthesis of Parallel Algorithms, Morgan Kaufman, San Mateo, CA, 1993.