THIRD SEMESTER

Course Code	:	**CSPC21**
Course Title	:	**Data Structures**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**PC**

Objectives

- To understand the various techniques of sorting and searching
- To design and implement arrays, stacks, queues, and linked lists
- To understand the complex data structures such as trees and graphs

Unit – I

Development of Algorithms - Notations and analysis - Storage structures for arrays - Sparse matrices - Stacks and Queues: Representations and applications.

Unit – II

Linked Lists - Linked stacks and queues - Operations on polynomials - Doubly linked lists - Circularly linked lists - Dynamic storage management - Garbage collection and compaction.

Unit – III

Binary Trees - Binary search trees - Tree traversal - Expression manipulation - Symbol table construction - Height balanced trees - Red-black trees.

Unit – IV

Graphs - Representation of graphs - BFS, DFS - Topological sort - Shortest path problems. String representation and manipulations - Pattern matching.

Unit – V

Sorting Techniques - Selection, Bubble, Insertion, Merge, Heap, Quick, and Radix sort - Address calculation - Linear search - Binary search - Hash table methods.

Outcomes

- Ability to develop programs to implement linear data structures such as stacks, queues, linked lists, etc.
- Ability to apply the concept of trees and graph data structures in real world scenarios
- Ability to comprehend the implementation of sorting and searching algorithms

Text Books

1. *J. P. Tremblay and P. G. Sorenson, "An Introduction to Data Structures with applications", Second Edition, Tata McGraw Hill, 1981*
2. *M. Tenenbaum and Augestien, "Data Structures using C", Third Edition, Pearson Education 2007.*

Reference Book

1. *Sartaj Sahni, "Data Structures, Algorithms and Applications in C++", Universities Press (I) Pvt. Ltd.*

Course Code	:	**CSPC22**
Course Title	:	**Digital Systems Design**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**PC**

Objectives

- To understand the essential knowledge on the fundamental of digital circuits
- To understand the overview on the design principles of digital computing systems

Unit - I

Binary codes - Weighted and non-weighted - Binary arithmetic conversion algorithms, Canonical and standard boolean expressions - Truth tables, K-map reduction - Don't care conditions - Adders / Subtractors - Carry look-ahead adder - Code conversion algorithms - Design of code converters - Equivalence functions.

Unit - II

Binary/Decimal Parallel Adder/Subtractor for signed numbers - Magnitude comparator - Decoders / Encoders - Multiplexers / Demultiplexers - Boolean function implementation using multiplexers.

Unit - III

Sequential logic - Basic latch - Flip-flops (SR, D, JK, T and Master-Slave) - Triggering of flip-flops - Counters - Design procedure - Ripple counters - BCD and Binary - Synchronous counters, Registers - Shift registers - Registers with parallel load, Reduction of state and flow tables - Race-free state assignment - Hazards.

Unit - IV

Introduction to VLSI design - Basic gate design - Digital VLSI design - Design of general boolean circuits using CMOS gates. Verilog Concepts – Basic concepts – Modules & ports & Functions – useful modeling techniques – Timing and delays – user defined primitives. Modeling Techniques

Unit - V

Advanced Verilog Concepts – Synthesis concepts – Inferring latches and flip-flops – Modeling techniques for efficient circuit design. Design of high-speed arithmetic circuits – Parallelism Pipelined Wallace tree tipliers - Systolic algorithms - Systolic matrix multiplication.

Outcomes

- Ability to design and implement complicated digital systems using Verilog
- Ability to design a VLSI circuit for an application
- Ability to comprehend the digital design logic

Text Books

1. *Morris Mano and Michael D. Ciletti, "Digital Design", 5th Ed, PHI, 2012*
2. *Samir Palnitkar, "Verilog HDL", 2nd Edition, Pearson Education, 2003*

Reference Books

1. *Michael D. Ciletti, "Advanced Digital Design with the Verilog HDL, 2nd Edition, Pearson Education, 2010*
2. *Stephen Brown, "Fundamentals of Digital Logic with Verilog", McGraw Hill, 2007*

Course Code	:	**CSPC23**
Course Title	:	**Principles of Programming Languages**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**PC**

Objectives

- To understand and describe syntax and semantics of programming languages
- To understand data, data types, and basic statements
- To understand call-return architecture and ways of implementing them
- To understand object-orientation, concurrency, and event handling in programming languages
- To develop programs in non-procedural programming paradigms

UNIT - I

Syntax and Semantics and Basic Statements: Evolution of programming languages – describing syntax & semantics –lexical analysis –parsing –recursive-decent –bottom up parsing– primitive data types –strings –array types –associative arrays –record types –union types – pointers and references –Arithmetic expressions –relational and Boolean expressions – assignment statements –mixed-mode assignments –control structures –selection –iterations – branching –guarded statements.*

UNIT - II

Subprograms and Implementations: Subprograms –design issues –local referencing – parameter passing –overloaded methods –generic methods –design issues for functions – semantics of call and return –implementing simple subprograms –stack and dynamic local variables –nested subprograms –blocks –dynamic scoping.*

UNIT - III

Object-Orientation, Concurrency, and Event Handling: Object-orientation –design issues for OOP languages –implementation of object-oriented constructs –concurrency –semaphores – monitors –message passing –threads –statement level concurrency –exception handling –even handling.*

UNIT - IV

Functional Programming: Introduction to lambda calculus –fundamentals of functional programming languages –Programming with Scheme –Introduction to LISP - Lists - Storage allocation for lists - Some useful functions - Error handling.*

UNIT - V

Logic Programming: Introduction to logic and logic programming- Computing with relations – Programming with Prolog- Data structures in Prolog - Programming techniques - Control in Prolog - Cuts.–multi-paradigm languages.*

*Programming assignments are mandatory.

Outcomes

- Describe syntax and semantics of programming languages
- Explain data, data types, and basic statements of programming languages
- Design and implement subprogram constructs
- Apply object-oriented, concurrency, and event handling programming constructs
- Develop programs in Scheme, ML, and Prolog
- Understand and adopt new programming languages

Text Books

1. *Robert W. Sebesta, "Concepts of Programming Languages", Tenth Edition, Addison Wesley, 2012.*
2. *Michael L. Scott, "Programming Language Pragmatics", Third Edition, Morgan Kaufmann,2009.*
3. *R. Kent Dybvig, "The Scheme programming language", Fourth Edition, MIT Press, 2009.*
4. *Jeffrey D. Ullman, "Elements of ML programming", Second Edition, Prentice Hall, 1998.*
5. *Richard A. O'Keefe, "The craft of Prolog", MIT Press, 2009.*
6. *W. F. Clocksin and C. S. Mellish, "Programming in Prolog: Using the ISO Standard", Fifth Edition, Springer, 2003.*

Course Code	:	**CSPC24**
Course Title	:	**Computer Organization**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**PC**

Objectives

- To understand the basic hardware and software issues of computer organization
- To understand the representation of data at machine level
- To understand how computations are performed at machine level

Unit – I

Introduction, Technologies for building Processors and Memory, Performance, The Power Wall, Operations of the Computer Hardware, Operands Signed and Unsigned numbers, Representing Instructions, Logical Operations, Instructions for Making Decisions

Unit – II

MIPS Addressing for 32-Bit Immediates and Addresses, Parallelism and Instructions: Synchronization, Translating and Starting a Program, Addition and Subtraction, Multiplication, Division, Floating Point, Parallelism and Computer Arithmetic: Subword Parallelissm, Streaming SIMD Extensions and Advanced Vector Extensions in x86.

Unit – III

Logic Design Conventions, Building a Datapath, A Simple Implementation Scheme, overview of Pipelining, Pipelined Datapath, Data Hazards: Forwarding versus Stalling, Control Hazards, Exceptions, Parallelism via Instructions, The ARM Cortex – A8 and Intel Core i7 Pipelines, Instruction –Level Parallelism and Matrix Multiply Hardware Design language

Unit – IV

Memory Technologies, Basics of Caches, Measuring and Improving Cache Performance, dependable memory hierarchy, Virtual Machines, Virtual Memory, Using FSM to Control a Simple Cache, Parallelism and Memory Hierarchy: Redundant Arrays of Inexpensive Disks, Advanced Material: Implementing Cache Controllers

Unit – V

Disk Storage and Dependability, Parallelism and Memory Hierarchy: RAID levels, performance of storage systems, Introduction to multi threading clusters, message passing multiprocessors.

Outcomes

- Ability to analyze the abstraction of various components of a computer
- Ability to analyze the hardware and software issues and the interfacing
- Ability to work out the tradeoffs involved in designing a modern computer system

Text Book

1. *David A. Patterson and John L. Hennessey, "Computer organization and design, The Hardware/Software interface", Morgan Kauffman / Elsevier, Fifth edition, 2014*
2. *Smruti Ranjan Sarangi, "Computer Organization and Architecture", McGraw Hill Education, 2015*

Reference Book

1. *V. Carl Hamacher, Zvonko G. Varanesic, and Safat G. Zaky, "Computer Organization", 6th edition, McGraw-Hill Inc, 2012*
2. *William Stallings, "Computer Organization and Architecture", 8th Edition, Pearson Education, 2010*

Course Code	:	**MAIR37**
Course Title	:	**Introduction to Probability Theory**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**GIR**

Objectives

- To introduce the fundamental concepts and theorems of probability theory
- To apply elements of stochastic processes for problems in real life
- To understand elementary queuing concepts and apply elsewhere in computer science.

Unit – I

Definitions of Probability – Notion of sample space – Events – Basics of Combinatorial Analysis – Posing Probability problems mathematically – Examples

Unit – II

Conditional Probability – Baye's Rule – Random variable – Probability mass function, Density function, Distribution function – Bernoulli Trials – Binomials Distribution – Poisson Approximation – Poisson Distribution – Normal Distribution – Moment Generating Function

Unit – III

Joint Probability Density Function – Marginal and Conditional Densities – Function of Random Variable – Covariance and Conditional Expectation – Correlation Coefficient

Unit – IV

Chebyshev Inequality – Law of Large Numbers – Central Limit Theorem – Random Process – Markov Dependence, Markov Chains, definition, examples, ergodicity

Unit – V

Finite Markov Chain – Various States – Limiting Probability – Introduction to Markov Process – M/M/l Queues with finite and infinite waiting space

Outcomes

- Ability to conceptualize the necessity of randomness concept in practical situation
- Ability to approximate the real problems using stochastic process and deduce results
- Ability to deduce useful results and interpret them based on the analysis of queuing theory

Text Books

1. *W. FELLER, An Introduction to Probability Theory and its Applications, Vol. 1, Wiley Eastern, New Delhi.*
2. *A. PAPUULIS, Probability, Random Variables and Stochastic Processes, McGraw Hill.*
3. *K. S. TRIVEDI, Probability and Statistics with Reliability and Queueing and Computer Science Applications, Prentice Hall of India, 1988.*
4. *A. O. ALLEN, Introduction to Probability, Statistics and Queueing Theory with Computer Science Applications, Academic Press, 2006 reprint.*

Course Code	:	**CSPC25**
Course Title	:	**Combinatorics and Graph Theory**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC11**
Course Type	:	**PC**

Objective

- To introduce basic combinatorics and graph theory

Unit – I

Scope of the course, Application areas in CS, A feel of some advanced problems in Combinatorial Optimization/Graph Theory, Sum/Product rules, Power set - algorithm, Bijections/Mapping/Examples Permutations and combinations, examples, Combinatorial ideas, Pascal Triangle Counting principles via examples, Insertion sort, Stirling numbers

Unit – II

Average case analysis and combinatorial ideas Double counting - Fubini's method, PHP principle, various illustrations Stirling numbers of II kind, Combinatorial identities, Binomial theorem Multinomial theorem, P(n,t1, - - - ,tp) notation, Euler PHI-function, Properties, Steps in Sieve of Eratosthenes

Unit – III

Inclusion/Exclusion Principle, Exercises, Derangements, IMO type problems, Ramsey Theory, Partition problems, Ferrar Diagrams Recurrences - Examples in CS, Substitution methods, Recurrence trees, D&C Solving Fibonacci series - GF idea, Difference equations, examples. Homogeneous case Inhomogeneous case

Unit – IV

Basics of GFs, Review problems, Examples, GF manipulations Coupled difference equations, Graph theory fundamentals, Representations, Examples in CS - MST review, Party problem Distance in graphs, Floyd-Warshall algorithm, Operations in graphs, Meanings of products

Unit – V

Regular graphs, related results, Coloring, Cliques and independent sets, Trees, definitions, related problems, properties, Network Flows, Definitions, Related discussions and Max-Flow Min-Cut Theorem, Introduction to optimization problems in CS, LP formulation, Branch-and-Bound

Outcomes

- Ability to apply combinatorial ideas in mathematical arguments in analysis of algorithms, queuing theory, etc.
- Ability to comprehend graph theory fundamentals and tackle problems in dynamic programming, network flows, etc.
- Ability to design and develop real time application using graph theory

Textbooks

1. *J. H. Van Lint and R. M. Wilson, "A course in Combinatorics", 2nd edition, Cambridge Univ. Press, 2001*
2. *G. Chartrand and P. Zhang, "Introduction to Graph Theory", McGraw-Hill, 2006*

Reference Books

1. *Kenneth H. Rosen, "Discrete Mathematics and its Applications", 7th edition, McGraw-Hill, 2012*
2. *John Harris, Jeffry L. Hirst, Michael Mossinghoff, "Combinatorics and Graph Theory", 2nd edition, Springer Science & Business Media, 2008*
3. *Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", 5th Edition, PHI/Pearson Education, 2004*
4. *Dr. D.S. Chandrasekharaiah, "Graph Theory and Combinatorics", Prism, 2005.*

Course Code	:	**CSLR21**
Course Title	:	**Data Structures Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC21**
Course Type	:	**ELR**

Objectives

- To analyze the time and space complexities and efficiency of various algorithms.
- To understand the practical application of linear and nonlinear data structures.
- To introduce and practice advanced algorithms, programming techniques necessary for developing sophisticated computer application programs.

Outcomes

- Ability to apply and implement the learned algorithm for problem solving
- Ability to identify the data structure to develop program for real time applications
- Ability to design and develop optimal algorithms

Experiments

- Problems in C/C++/ Java using data structures involving arrays, stacks, queues, strings, linked lists, trees, graphs.
- Operations on stacks, queues and linked lists
- Conversion of infix expressions to postfix and evaluation of postfix expressions
- Implementation of priority queue
- Implementation of Binary Tree and Binary Search Tree
- Implementation of Sorting Techniques

Course Code	:	**CSLR22**
Course Title	:	**Digital Systems Design Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC22**
Course Type	:	**ELR**

Objectives

- To develop programs in Hardware Description Language
- To design and implement synchronous sequential, asynchronous sequential circuits
- To be familiar with basic combinational and sequential components used in the typical data path designs

Outcomes

- Ability to design synchronous sequential circuits using basic flip-flops, counters, PLA, PAL
- Ability to design and develop basic digital systems
- Ability to debug digital circuits

Experiments

- Design of a 32-bit carry look-ahead adder with logarithmic depth using Verilog
- Design of a Wallace tree multiplier using Verilog
- Design of a 4-bit DSP processor using Verilog
- Burning the 4-bit DSP processor on a FPGA

FOURTH SEMESTER

Course Code	:	**CSPC26**
Course Title	:	**Operating Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC24**
Course Type	:	**PC**

Objectives

- To provide knowledge about the services rendered by operating systems
- To provide a detailed discussion of the various memory management techniques
- To discuss the various file-system design and implementation issues
- To discuss how the protection domains help to achieve security in a system

Unit – I

Operating Systems –Definition- Types- Functions -Abstract view of OS- System Structures – System Calls- Virtual Machines –Process Concepts –Threads –Multithreading

Unit – II

Process Scheduling- Process Co-ordination –Synchronization –Semaphores –Monitors Hardware Synchronization –Deadlocks –Methods for Handling Deadlocks

Unit – III

Memory Management Strategies –Contiguous and Non-Contiguous allocation –Virtual memory Management –Demand Paging- Page Placement and Replacement Policies

Unit – IV

File System –Basic concepts - File System design and Implementation –Case Study: Linux File Systems - Mass Storage Structure –Disk Scheduling –Disk Management –I/O Systems-System Protection and Security.

Unit – V

Distributed Systems –Distributed operating systems –Distributed file systems –Distributed Synchronization.

Outcomes

- Ability to comprehend the techniques used to implement the process manager
- Ability to comprehend virtual memory abstractions in operating systems
- Ability to design and develop file system interfaces, etc.

Text Book

1. *Silberschatz, Galvin, Gagne, "Operating System Concepts", John Wiley and Sons, 9th edition, 2013*

References Books

1. *William Stallings, "Operating Systems –Internals and Design Principles", 8/E, Pearson Publications, 2014*
2. *Andrew S. Tanenbaum, "Modern Operating Systems", 4/E, Pearson Publications, 2014*

Course Code	:	**MAIR44**
Course Title	:	**Principles of Operations Research**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**GIR**

Objectives

- To classify and formulate real-life problem for modelling, solving and applying for decision making.
- To study the formulation and various methods of solutions for linear programming, transportation, assignment, CPM and PERT problems
- To solve problems using dynamic programming method

Unit - I

Introduction to operational research-Linear programming problems (LPP)-Graphical method-Simplex method-Big M Method-Dual simplex method-Primal Dual problems

Unit – II

Dual theory and Sensitivity analysis-Transportation and assignment problems-Applications(Emphasis should be more on problems than theory)

Unit – III

CPM and PERT –Network diagram-Events and activities-Project Planning-Reducing critical events and activities-Critical path calculations-example-Sequencing problems

Unit – IV

Replacement problems-Capital equipment- Discounting costs- Group replacement - Inventory models-various costs- Deterministic inventory models-Economic lot size-Stochastic inventory models-Single period inventory models with shortage cost.

Unit – V

Dynamic programming-Formulation-Invest problem-General allocation problem-Stage coach problem-Production Scheduling.

Outcomes

- Ability to analyse problems in engineering, management, or business environment, focusing on important details
- Ability to formulate real problems in terms of input-output-parameters relationships and identify the solution procedure
- Ability to comprehend the methodologies and correlate with engineering problems

Text Books

1. *H. A. Taha, "Operations Research - An introduction", 9th edition, Prentice Hall, Macmillan, 2010*
2. *F. S. Hiller and G. J. Liebermann, "Introduction to operational research", 8th edition, McGraw-Hil, 2005*
3. *B. E. Gillet, "Introduction to operational research-A computer oriented algorithmic approach", McGraw Hill, 1989*
4. *H. M. Wagner, Principles of operational research with applications to managerial decisions, PH, Inc, 1975*

Course Code	:	**CSPC27**
Course Title	:	**Data Communications and Networks**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**PC**

Objectives

- To provide insight about fundamental concepts and reference models (OSI and TCP/IP) and its functionalists
- To gain comprehensive knowledge about the principles, protocols, and significance of Layers in OSI and TCP/IP
- To know the implementation of various protocols and cryptography techniques

UNIT- I

Introduction to computer networks: Network – Component and Categories – Topologies – Transmission Media – Reference Models: ISO/OSI Model and TCP/IP Model.*

UNIT-II

Physical Layer: Digital and analog Signals, Periodic Analog Signals, Transmission Impairments, Digital data transmission techniques, Analog data transmission techniques, Multiplexing and Spread Spectrum.*

UNIT-III

Data Link Layer: Error – Detection and Correction – Parity – LRC-CRC – Hamming Code – Flow Control and Error Control – Stop and wait – ARQ – Sliding window – HDLC –Multiple Access Protocols –IEEE 802.3 Ethernet.*

UNIT-IV

Network Layer: Packet Switching and Datagram approach – IP addressing methods – Subnetting – Routing – Distance Vector Routing – Link State Routing – Broadcast and Multicast Routing.*

UNIT-V

Transport Layer: Transport Services – UDP -TCP - Congestion Control – Quality of Services(QOS) **Application Layer:** Domain Name Space (DNS) – Electronic Mail - WWW – Cryptography Techniques.*

*Programming assignments are mandatory.

Outcomes

- Ability to gain insight about basic network theory and layered communication architectures
- Ability to provide solutions to various problems in network theory
- Ability to conceptualize and design a network stack

Text Books

1. *Andrew S. Tanenbaum and David J. Wetherall, "Computer Networks", 5th edition, Prentice Hall, 2011*
2. *Behrouz A. Foruzan, "Data Communication and Networking", 5th edition, Science Engineering & Math Publications, 2013*

Reference Books

1. *W. Stallings, "Data and Computer Communication", 10th Edition, Pearson Education, 2014*

Course Code	:	**CSPC28**
Course Title	:	**Automata and Formal Languages**
Number of Credits	:	**3-1-0-4**
Prerequisites(Course code)	:	**CSPC11**
Course Type	:	**PC**

Objectives

- To introduce concepts in automata theory and theory of computation
- To identify different formal language classes and their relationships
- To design grammars and recognizers for different formal languages

Unit – I

Introduction: Alphabets, Strings and Languages; Automata and Grammars, Deterministic finite Automata (DFA)-Formal Definition, Simplified notation: State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence of NFA and DFA, Minimization of Finite Automata, Distinguishing one string from other, Myhill-Nerode Theorem

Unit – II

Regular Expression (RE): Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to FA, DFA to Regular expression, Arden Theorem, Non Regular Languages, Pumping Lemma for regular Languages. Application of Pumping Lemma, Closure properties of Regular Languages, Decision properties of Regular Languages, FA with output: Moore and Mealy machine, Equivalence of Moore and Mealy Machine, Applications and Limitation of FA.

Unit – III

Context Free Grammar (CFG) and Context Free Languages (CFL): Definition, Examples, Derivation, Derivation trees, Ambiguity in Grammar, Inherent ambiguity, Ambiguous to Unambiguous CFG, Useless symbols, Simplification of CFGs, Normal forms for CFGs: CNF and GNF, Closure proper ties of CFLs, Decision Properties of CFLs: Emptiness, Finiteness and Membership, Pumping lemma for CFLs

Unit – IV

Push Down Automata (PDA): Description and definition, Instantaneous Description, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, Equivalence of PDA and CFG, CFG to PDA and PDA to CFG,

Unit – V

Turing machines (TM): Basic model, definition and representation, Instantaneous Description, Language acceptance by TM, Variants of Turing Machine, TM as Computer of Integer functions, Universal TM, Chur ch's Thesis, Recursive and recursively enumerable languages, Halting problem, Introduction to Undecidability, Undecidable problems about TMs. Post correspondence problem (PCP), Modified PCP, Introduction to recursive function theory

Outcomes

- Ability to relate practical problems to languages, automata, and computability
- Ability to demonstrate an increased level of mathematical sophistication
- Ability to apply mathematical and formal techniques for solving problems

Text Book

1. *Hopcroft and Ullman, "Introduction to Automata Theory, Languages and Computation",Pearson Education, 3rd edition, 2014*

Reference books

1. *Martin J. C., "Introduction to Languages and Theory of Computations", TMH, 4th edition, 2010*
2. *Peter Linz, "An Introduction to Formal Language and Automata", Narosa Pub. House, 2011*
3. *Papadimitriou, C. and Lewis, C. L., "Elements of the Theory of Computation", PHI, 1997*

Course Code	:	**CSPC29**
Course Title	:	**Introduction to Algorithms**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC21**
Course Type	:	**PC**

Objectives

- To understand the importance of algorithm and its complexity
- To analyze the complexity of an algorithm in terms of time and space complexities
- To design and implement various programming paradigms and its complexity

Unit – I

Algorithms - Examples - Tournament method - Evaluating polynomial functions - pre-processing of coefficients - solving recurrence equations.

Unit – II

Divide and Conquer method - Strassen's matrix multiplication - Greedy method - Knapsack problem - Job sequencing with deadlines - Minimum spanning trees.

Unit – III

Dynamic Programming - Multistage graphs - All pair's shortest paths - Optimal binary search trees - Travelling salesman problem - Fast Fourier transform.

Unit – IV

Randomized Algorithms and Amortized Analysis - Las Vegas and Monte Carlo types - Randomized quick sort and its analysis - Min-Cut algorithm.

Unit – V

NP-Hard and NP-complete problems - Basic concepts - Reducibility - Cook's theorem (without proof) - Turing machines - NP-Hard graph problems.

Outcomes

- Ability to analyze the time and space complexity, given an algorithm
- Ability to apply the techniques of algorithm in solving real world problems
- Ability to develop systematically an algorithm for solving a problem

Textbook

1. *T. Cormen, C. Lieserson, R. Rivest, and C. Stein, "Introductions to Algorithms", Prentice-Hall/India, 3rd edition, 2009*

Course Code	:	**HSIR14**
Course Title	:	**Professional Ethics**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**GIR**

This Course Syllabus will provided by the Humanities Department

Course Code	:	**CSLR23**
Course Title	:	**Algorithms Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC29**
Course Type	:	**ELR**

Objectives

- To learn how to analyze the complexity of algorithms
- To compare and evaluate algorithms in terms of time and space complexity
- To program brute force, divide and conquer, decrease and conquer, transform and conquer, greedy, and dynamic techniques

Experiments

- Estimating worst-case/average-case complexity of algorithms via programs
- Determining machine constants
- Programs involving some advanced data structures
- Implementing example problems
- Illustrating the different paradigms of algorithm design
- Solving miscellaneous problems e.g. problems in string manipulation, graph theory, optimization

Outcomes

- Ability to solve and analyze general algorithms based on space and time complexity
- Ability to implement and empirically compare fundamental algorithms and data structures to real-world problems
- Ability to design, develop, and optimize algorithms in different paradigms

Course Code	:	**CSLR24**
Course Title	:	**Operating Systems Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC26**
Course Type	:	**ELR**

Objectives

- To understand the concept of Operating System
- To experience the practical side of the functioning of various blocks in OS

Experiments

1. Hands on Unix Commands
2. Shell programming for file handling
3. Shell Script programming using the commands *grep, awk,* and *sed*
4. Implementation of CPU scheduling algorithms
5. *Pthread* Programming
6. Implementation of Synchronization problems using Semaphores, Message Queues and Shared Memory
7. Implementation of Memory Management - Allocation, Placement and replacement Algorithms

Outcomes

- Ability to make use of tools for solving synchronization problems
- Ability to compare and contrast various CPU scheduling algorithms
- Ability to understand the differences between segmented and paged memories

References

1. *Silberschatz, Galvin, Gagne, "Operating System Concepts", 9/E, John Wiley and Sons 2013*
2. *William Stallings, "Operating Systems –Internals and Design Principles", 8/E, Pearson Publications, 2014*
3. *Andrew S. Tanenbaum, "Modern Operating Systems", 4/E, Pearson Publications, 2014*

FIFTH SEMESTER

Course Code	:	**CSPC31**
Course Title	:	**Computer Architecture**
Number of Credits	:	**3-1-0-4**
Prerequisites(Course code)	:	**CSPC24**
Course Type	:	**PC**

Objectives

- To understand the concept of advanced pipelining techniques
- To understand the current state of art in memory system design
- To know the working principle of I/O devices

Unit - I

Introduction, Classes of computers, Defining Computer Architecture – Trends in Technology – Trends in Power and Energy in Integrated Circuits – Trends in Cost – Dependability – Measuring, Reporting and Summarizing Performance – Quantitative Principles of Computer Design.

Unit - II

Basic and Intermediate pipelining Concepts, The Major Hurdle of Pipelining – Pipeline Hazards, Pipelining Implementation, Implementation issues that makes Pipelining hard, Extending the MIPS Pipeline to Handle Multicycle Operations, The MIPS R4000 Pipeline.

Unit - III

Instruction-Level Parallelism: Concepts and Challenges – Basic Compiler Techniques for Exposing ILP – Reducing Branch Costs with Prediction – Overcoming Data Hazards with Dynamic Scheduling – Dynamic Scheduling – Hardware-Based Speculation – Exploiting ILP Using Multiple Issue and Static Scheduling – Exploiting ILP, Advanced Techniques for Instruction Delivery and Speculation, Studies of the Limitations of ILP

Unit - IV

Vector Architecture – SIMD Instruction Set Extensions for Multimedia – Graphics Processing Units – Detecting and Enhancing Loop-Level Parallelism – Centralized Shared-Memory Architectures – Performance of Shared-Memory Multiprocessors – Distributed Shared Memory, Models of Memory Consistency, Multicore Processors and Their Performance.

Unit - V

Review of Memory Hierarchy Design, Cache Performance, Basic Cache Optimizations, Virtual Memory, Protection and Examples of Virtual Memory, Advanced Optimizations of Cache Performance, Memory Technology and Optimizations, Protection: Virtual Memory and Virtual Machines, Crosscutting Issues: The Design of Memory Hierarchies. Case Studies / Lab Exercises

Outcomes

- Ability to apply performance metrics to find the performance of systems
- Ability to identify the problems in components of computer
- Ability to comprehend and differentiate various computer architectures and hardware

Text Book

1. *David. A. Patterson and John L. Hennessy, "Computer Architecture: A Quantitative approach", Elsevier, 5th Edition, 2012*

Reference Book

1. *K. Hwang and Naresh Jotwani, "Advanced Computer Architecture, Parallelism, Scalability, Programmability", Tata McGraw Hill, 2nd Edition, 2010*

Course Code	:	**CSPC32**
Course Title	:	**Internetworking Protocols**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC27**
Course Type	:	**PC**

Objectives

- To provide insight about networks, topologies, and the key concepts
- To gain comprehensive knowledge about the layered communication architectures (OSI and TCP/IP) and its functionalities
- To understand the principles, key protocols, design issues, and significance of each layers in ISO and TCP/IP
- To know the implementation of various layers

Unit – I

Review of Reference Models, Topology and switching, IEEE Standard 802 from Ethernet, Token Bus, Token Ring and Wireless LAN, Connecting Devices

Unit - II

IPv4 headers, IP forwarding, Host Processing of IP datagrams, DHCP and Autoconfiguration, Firewalls and NAT, ICMPv4, IP Fragmentation, DNS, Broadcasting and Local Multicasting – IGMP, Routing Protocols

Unit – III

IPv6 Transition issues, Protocol basics, Addressing, Options and Extension headers, ICMPv6, Neighbor Discovery, Routing, Autoconfiguration, Multicast Listener Discovery (MLD), IPv6 and DNS

Unit – IV

Transmission Control Protocol (TCP), TCP Connection Management, TCP Data Flow and Window Management, Stream Control Transmission Protocol (SCTP), Services, SCTP Association management, SCTP flow and error control

Unit - V

Need for Mobile IP, Overview of Mobile IP, Details of Mobile IP, Tunneling, Mobility for IPv6, Applications of Mobile IP – Security primer, Campus Mobility, Internet wide mobility, A service provider perspective

Outcomes

- Ability to gain insight about basic network theory and layered communication architectures
- Ability to code and implement MAC protocols, IPv4, IPv6, and TCP
- Ability to design and develop Mobile IP
- Ability to design and develop a communication protocol

Text Books

1. *W. Richard Stevens and G. Gabrani, "TCP/IP Illustrated: The Protocols", Pearson, 2011*
2. *Peter Loshin, Morgan Kaufmann, "IPv6: Theory, Protocol, and Practice", 2nd Ed,2003*
3. *James Solomon, "Mobile IP: The Internet Unplugged", 1st Ed, Pearson Education, 2008*

Reference Books

1. *Kevin R. Fall and W. Richard Stevens, "TCP/IP Illustrated, Vol. 1- The Protocols", 2nd Edition, Addison-Wesley, 2012*
2. *Silvia Hagen, "IPv6 Essentials, 2nd Edition, O'Reilly Media, 2006*
3. *Charles E. Perkins, "Mobile IP: Design Principles and Practices", 1st Edition, Pearson Education, 2008*

Course Code	:	**CSPC33**
Course Title	:	**Database Management Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**PC**

Objectives

- To learn data models, conceptualize and depict a database system using ER diagram
- To understand the internal storage structures in a physical DB design
- To know the fundamental concepts of transaction processing techniques

Unit – I

Introduction: Purpose of Database System -– Views of data – data models, database management system, three-schema architecture of DBMS, components of DBMS. E/R Model - Conceptual data modeling - motivation, entities, entity types, attributes, relationships, relationship types, E/R diagram notation, examples.

Unit – II

Relational Model: Relational Data Model - Concept of relations, schema-instance distinction, keys, referential integrity and foreign keys, relational algebra operators, SQL - Introduction, data definition in SQL, table, key and foreign key definitions, update behaviors. Querying in SQL, notion of aggregation, aggregation functions group by and having clauses, embedded SQL

Unit – III

Database Design: Dependencies and Normal forms, dependency theory - functional dependencies, Armstrong's axioms for FD's, closure of a set of FD's, minimal covers, definitions of 1NF, 2NF, 3NF and BCNF, decompositions and desirable properties of them, algorithms for 3NF and BCNF normalization, 4NF, and 5NF

Unit – IV

Transactions: Transaction processing and Error recovery - concepts of transaction processing, ACID properties, concurrency control, locking based protocols for CC, error recovery and logging, undo, redo, undo-redo logging and recovery methods.

Unit – V

Implementation Techniques: Data Storage and Indexes - file organizations, primary, secondary index structures, various index structures - hash-based, dynamic hashing techniques, multi-level indexes, B+ trees.

Outcomes

- Ability to Install, configure, and interact with a relational database management system
- Ability to master the basics of SQL and construct queries using SQL
- Ability to design and develop a large database with optimal query processing

Text Books

1. *A. Silberschatz, Henry F. Korth, and S. Sudharshan, "Database System Concepts", 5th Ed, Tata McGraw Hill, 2006.*
2. *C. J. Date, A. Kannan and S. Swamynathan, "An Introduction to Database Systems", 8thed, Pearson Education, 2006.*

References Books

1. *RamezElmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", Fourth Edition, Pearson/Addisionwesley, 2007*
2. *Raghu Ramakrishnan, "Database Management Systems", Third Edition, McGraw Hill, 2003*
3. *S. K. Singh, "Database Systems Concepts, Design and Applications", First Edition, Pearson Education, 2006*

Course Code	:	**CSPC34**
Course Title	:	**Software Engineering**
Number of Credits	:	**3-0-1-4**
Prerequisites(Course code)	:	**-**
Course Type	:	**PC**

Objectives

- To understand the Software Engineering Practice& Process Models
- To understand Design Engineering, Web applications, and Software Project Management
- To gain knowledge of the overall project activities.

Unit-I

Introduction: Role of Software Engineer, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Quality Attributes.

Assessment: How Software Engineering Changes? Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Choosing a social relevant problem-Summary Team Report.

Unit-II

Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modelling, Data Flow Diagrams, Entity Relationship Diagrams, Designing the architecture.

Assessment: Impact of Requirement Engineering in their problem. Decision Tables, SRS Document, IEEE Standards for SRS, Architectural design, component level design, user interface design, WebApp Design. Submission of SRS Document for Team Project.

Unit-III

Quality concepts, Review techniques, Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks.

Assessment: Framing SQA Plan. ISO 9000 Models, SEI-CMM Model and their relevance to project Management-other emerging models like People CMM.

Unit –IV

Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing, Software Testing Strategies - Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Testing conventional applications, object oriented applications, and Web applications, Formal modelling and verification, Software configuration management, Product metrics.

Assessment: Team Analysis in Metrics Calculation.

Unit-V

Project Management Concepts, Process and Project Metrics, Estimation for Software projects, Project Scheduling, Risk Management, Maintenance and Reengineering.

Assessment: Preparation of Risk mitigation plan.

Outcomes

- Assessment in each module gives the overall Software engineering practice.
- Ability to enhance the software project management skills
- Ability to comprehend the systematic methodologies involved in SE
- Ability to design and develop a software product in accordance with SE principles

Text books

1. *R. S. Pressman, "Software Engineering: A Practitioners Approach", McGraw Hill, 7th edition, 2010*
2. *Rajib Mall, "Fundamentals of Software Engineering", PHI Publication, 3rd edition, 2009*
3. *PankajJalote, "Software Project Management in practice", Pearson Education, New Delhi, 2002.*

Course Code	:	**CSLR31**
Course Title	:	**Network Programming Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC32**
Course Type	:	**ELR**

Objectives

- To create client and server applications using the "Sockets" API and the implementation of Data link layer protocol and TCP layer
- To conduct computer communication network simulations
- To have a hands on experience of computer network simulation and modeling techniques using NS-3 simulation software

Experiments

1. Exercises on Socket Programming using C and Java
2. Exercises using NS-3 Network Simulator
    a. Basics of Network Simulation
        – Introduction , Platform required to run network simulator, Backend Environment of Network Simulator, Agents and applications, Tracing
    b. Simulating a Local Area Network
        – Local Area Network, LAN Topologies, MAC Protocol, Taking turns, Ethernet, Ethernet Frame Structure, Ethernet Versions, Simulating a LAN using Network Simulator 3
        – Implementation of various MAC protocols
        – Setting up of various network topologies
        – Measurement of routing protocols
    c. Measuring Network Performance
        – Network Performance Evaluation, Performance Evaluation Metrics, Parameters Affecting the Performance of Networks, Performance Evaluation Techniques, Network Performance Evaluation using NS-3
        – Setting up of network that carries various application protocols and analyzing the performances
3. Hands on experiments on Network equipments
    a. Switches, Routers
    b. Hardware firewall

Outcomes

- Ability to invoke analytical studies of Computer Networks through network simulation
- Ability to design a network using NS-3 toolkit and its importance in designing a real network
- Ability to measure and analyze the network parameters for a high throughput network

Text Books

1. *W. Richard Stevens, "UNIX Network Programming – Networking APIs: Sockets and XTI", Vol. 1, 2nd Ed, 1998, Prentice Hall*
2. *Eitan Altman and Tania Jimenez, "NS Simulator for Beginners", Morgan & Claypool Publishers, 2011*
3. *Jack L. Burbank, "An Introduction to Network Simulator 3", 1st edition, Wiley-Blackwell, 2015*

Course Code	:	**CSLR32**
Course Title	:	**Database Management Systems Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC33**
Course Type	:	**ELR**

Objectives

- To explore the features of a Database Management Systems
- To interface a database with front end tools
- To understand the internals of a database system

Experiments

- Working with DDL,DML and DCL
- Inbuilt functions in RDBMS.
- Nested Queries & Join Queries.
- Set operators & Views in SQL.
- Control structures.
- Working with Procedures and Functions.
- Triggers
- Dynamic & Embedded SQL
- Working with XML
- Forms & Reports
- Database Design and implementation (Mini Project)

Outcomes

- Ability to use databases for building client server applications
- Ability to comprehend theinternal working of a database system
- Ability to design and develop a database using SQL and the mechanism in connecting with a Web based GUI

Text Books

1. *Abraham Silberschatz, Henry F. Korth, and S. Sudharshan, "Database System Concepts", 6th edition, Tata McGraw Hill, 2011*
2. *RamezElmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 4th Edition, Pearson/Addisionwesley, 2007*

SIXTH SEMESTER

Course Code	:	**CSPC35**
Course Title	:	**Principles of Cryptography**
Number of Credits	:	**3–0-0-3**
Prerequisites(Course code)	:	**CSPC25**
Course Type	:	**PC**

Objectives

- To gain knowledge about the mathematics of the cryptographic algorithms
- To get an insight into the working of different existing cryptographic algorithms
- To learn how to use cryptographic algorithms in security

Unit – I

Number Theory: Fermat's theorem, Cauchy 's theorem, Chinese remainder theorem, Primality testing algorithm, Euclid's algorithm for integers, quadratic residues, Legendre symbol, Jacobi symbol.*

Unit – II

Cryptography and cryptanalysis, Classical Cryptography, different type of attack: CMA,CPA,CCA etc., Shannon perfect secrecy, OTP, Pseudo random bit generators, stream ciphers and RC4.*

Unit – III

Block ciphers: Modes of operation, DES and its variants, finite fields (2^n), AES, linear and differential cryptanalysis.*

Unit – IV

One-way function, trapdoor one-way function, Public key cryptography, RSA cryptosystem, Diffie-Hellman key exchange algorithm, ElGamal Cryptosystem.*

Unit – V

Cryptographic hash functions, secure hash algorithm, Message authentication, digital signature, RSA digital signature.*

*Programming assignments are mandatory.

Outcomes

- Ability to understand the basic concepts of symmetric cryptosystem, public key cryptosystem and digital signature scheme
- Ability to reason about the security of cryptographic constructions
- Ability to break the cryptosystems that are not secure

Text Book

1. *Stinson. D. Cryptography: Theory and Practice, 3rd edition, Chapman & Hall/CRC, 2012*

Reference Books

1. *W. Stallings,"Cryptography and Network Security Principles and practice", 5/e, Pearson Education Asia, 2013*
2. *Behrouz A. Forouzan and DebdeepMukhopadhyay, "Cryptography and Network Security", 2nd edition, Tata McGraw Hill, 2013*
3. *Thomas Koshy, "Elementary Number Theory with Applications", Elsevier India, 2005*
4. *Online course: course on cryptography by Dan Boneh*

Course Code	:	**CSPC36**
Course Title	:	**Microprocessors and Microcontrollers**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC22**
Course Type	:	**PC**

Objectives

- To understand the concepts of Architecture of 8086 microprocessor
- To understand the design aspects of I/O and Memory Interfacing circuits
- To understand the architecture and programming of ARM processor

Unit – I

THE 8086 Microprocessor: Introduction to 8086 – Microprocessor architecture – Addressing modes - Instruction set and assembler directives – Assembly language programming – Modular Programming - Linking and Relocation - Stacks - Procedures – Macros – Interrupts and interrupt service routines – Byte and String Manipulation.

Unit – II

8086 System Bus Structure: 8086 signals – Basic configurations – System bus timing –System design using 8086 – IO programming – Introduction to Multiprogramming – System Bus Structure – Multiprocessor configurations – Coprocessor, Closely coupled and loosely Coupled configurations – Introduction to advanced processors.

Unit – III

Microcontroller: Architecture of 8051 – Special Function Registers(SFRs) - I/O Pins Ports and Circuits - Instruction set - Addressing modes -Programming 8051 Timers – Interfacing Microcontroller -Serial Port Programming - Interrupts Programming – LCD & Keyboard - External Memory Interface- Stepper Motor.

Unit – IV

Introduction to Embedded Systems: Complex systems and microprocessors– Embedded system design process – Instruction sets preliminaries - ARM Processor – CPU: programming input and output supervisor mode, exceptions and traps – Co-processors- Memory system mechanisms – CPU performance

Unit – V

Embedded Computing Platform Design and Optimization: The CPU Bus-Memory devices and systems–Designing with computing platforms – platform-level performance analysis - Components for embedded programs-Models of programs- Assembly, linking and loading – compilation techniques- Program level performance analysis – Software performance optimization – Analysis and optimization of program size- Program validation and testing.

Outcomes

- Ability to design and implement programs on 8086 microprocessor
- Ability to design I/O circuits and Memory Interfacing circuits
- Ability to design and develop components of ARM processor

Text Books

1. *Yu-Cheng Liu, Glenn A.Gibson, "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design", Second Edition, Prentice Hall of India, 2007*
2. *Mohamed Ali Mazidi, Janice GillispieMazidi, RolinMcKinlay, "The 8051 Microcontroller and Embedded Systems: Using Assembly and C", 2nd Edition, Pearson Education, 2011*
3. *Marilyn Wolf, "Computers as Components - Principles of Embedded Computing System Design", 3rd Edition "Morgan Kaufmann Publisher (An imprint from Elsevier), 2012*

References Books

1. *Doughlas V. Hall, "Microprocessors and Interfacing, Programming and Hardware", Tata McGraw-Hill, 2012*
2. Kenneth J.Ayala, "The 8051 Microcontroller Architecture, Programmingand Applications", 3 rd edition, West Publishing, 2005
3. *Jonathan W. Valvano, "Embedded Microcomputer Systems Real Time Interfacing", 3rd Edition, Cengage Learning, 2012*
4. *David. E. Simon, "An Embedded Software Primer", 1st Edition, Fifth Impression, Addison-Wesley Professional, 2007*

Course Code	:	**CSPC37**
Course Title	:	**Mobile Applications Development**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC32**
Course Type	:	**PC**

Objectives

- To learn the basics of mobile application development
- To get accustomed to Android platform
- To develop skills in developing basic Android applications

Unit – I

Introduction to Android: Native Android Application; SDK Features; Introduction to Open Handset Alliance; Development Framework; Application Fundamentals; Device Compatibility; System permissions.

Unit –II

User Interface and Application Components: Basic UI Design; Fragments; Widget Toolbox; Creating New View; Introduction to Intents; Intent Filters and broadcast Receivers; Activities; Services; Content Providers; Application Widgets; Processes and Threads.

Unit –III

Files and Database Handling: Saving Application Data; Shared Preferences; Preference Framework and Activity; Static File as Resource; File System; Introduction to SQLite Database; Querying SQLite; Storage options; Data backup

Unit – IV

User Experience Enhancement: Action Bar; Menus and Action Bar Items; Settings; Dialogs; Customizing Toast; Notifications; Search; Drag and Drop

Unit –V

Multimedia, Wireless Connectivity and Telephony: Audio and Video Handling; Manipulating Raw Audio; Sound Effects; Camera Programming; Video Recording; Managing Wireless Connectivity : WiFi, Bluetooth, Near Field Communication; Hardware Support for Telephony; Telephony Management; SMS and MMS

Outcomes

- Ability to comprehend Android platform and its usefulness in application development
- Ability to acquire skill set to execute applications in Android based devices
- Ability to design and develop deployable Android applications

Text Books

1. *Reto Meier, "Professional Android 4 Application Development", Wrox, 2012*
2. *Matt Gifford, "PhoneGap Mobile Application Development Cookbook", PACKT, 2012*
3. *Adrian Kosmaczewski, "Mobile JavaScript Application Development", O'RELLY, 2012*
4. http://developer.android.com/
5. http://www.tutorialspoint.com/mobile_development_tutorials.htm

Course Code	:	**CSLR33**
Course Title	:	**Mobile Applications Development Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC37**
Course Type	:	**ELR**

Objectives

- To learn the basics of mobile application development
- To get accustomed to Android platform
- To develop skills in developing basic Android applications

Experiments

1. Install the Android SDK and developer tools and build a test project to confirm that those tools are properly installed and configured
2. Write a program using a Table Layout for our restaurant data entry form, add a set of radio buttons to represent the type of restaurant
3. Write a program using activity class to show different events.
4. Write a program to send user from one application to another. (For example redirection to map)
5. Write a program to play audio files.
6. Write a program to play video files.
7. Write a program to capture image using built in camera.
8. Write a program to send SMS.
9. Write a program to convert text to speech.
10. Write a program to call a number.

Outcomes

- Ability to gain hands on experience in Android SDK
- Ability to design and develop applications in Android based devices
- Ability to design and develop deployable Android applications

Course Code	:	**CSELR34**
Course Title	:	**Microprocessor and Microcontroller Laboratory**
Number of Credits	:	**0-0-3-2**
Prerequisites(Course code)	:	**CSPC36**
Course Type	:	**ELR**

Objectives

- To understand and learn the assembly language programming of various microprocessor architectures.
- To obtain the practical training of interfacing the peripheral devices with the processor and microcontroller.
- To control the components of a microprocessor based system through the use of interrupts.
- To have a practical knowledge on assembling PC hardware, installation and troubleshooting the Microprocessor and Microcontrollers.

Experiments

- Solving problems using 8086 Microprocessor.
- Interfacing 8255 Programmable parallel I/O device with 8086 microprocessor.
- Interfacing A.D convertor, D/A convertor with 8086 microprocessor.
- Solving 8086 procedure and macro oriented programs in Turbo Assembler TASM
- Interfacing various devices with the microcontroller 8051 : A/D converter, D/A converter, seven segment display, LCD, stepper motor, external keyboard, interrupt controller and 8251 for serial data transfer.
- PC hardware assembly.
- Installation and trouble shooting

Outcomes

- Ability to write programs in assembly language using trainer kits
- Ability to interface development kits effectively for the real time applications of various peripheral devices with the processor
- Ability to assemble and troubleshoot hardware devices

SEVENTH SEMESTER

Course Code	:	**HSIR13**
Course Title	:	**Industrial economics and foreign trade**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**GIR**

This Course Syllabus will provided by the Humanities Department

Course Code	:	**CSPC41**
Course Title	:	**Principles of Compiler Design**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC28**
Course Type	:	**PC**

Objectives

- To introduce the major concept areas of language translation and compiler design
- To enrich the knowledge in various phases of compiler and its use
- To provide practical programming skills necessary for constructing a compiler

Unit – I

Introduction to Compiling: Compilers – Analysis of the source program – Phases of a compiler – Cousins of the Compiler – Grouping of Phases – Compiler construction tools – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens.*

Unit – II

Syntax Analysis: Role of the parser –Writing Grammars –Context-Free Grammars – Top Down parsing – Recursive Descent Parsing – Predictive Parsing – Bottom-up parsing – Shift Reduce Parsing – Operator Precedent Parsing – LR Parsers – SLR Parser – Canonical LR Parser – LALR Parser.*

Unit – III

Intermediate Code Generation: Intermediate languages – Declarations – Assignment Statements – Boolean Expressions – Case Statements – Back patching – Procedure calls.*

Unit – IV

Code Optimization and Run Time Environments: Introduction– Principal Sources of Optimization – Optimization of basic Blocks – DAG representation of Basic Blocks - Introduction to Global Data Flow Analysis – Runtime Environments – Source Language issues – Storage Organization – Storage Allocation strategies – Access to non-local names – Parameter Passing, Error detection and recovery.*

Unit – V

Code Generation: Issues in the design of code generator – The target machine – Runtime Storage management – Basic Blocks and Flow Graphs – Next-use Information – A simple Code generator – Peephole Optimization.*

*Programming Assignments are mandatory

Outcomes

- Ability to apply the knowledge of lex tool &yacc tool to develop a scanner & parser
- Ability to design and develop software system for backend of the compiler
- Ability to comprehend and adapt to new tools and technologies in compiler design

Text Books

1. *Alfred V. Aho, Jeffrey D Ullman, "Compilers: Principles, Techniques and Tools", Pearson Education Asia, 2012*
2. *Jean Paul Tremblay, Paul G Serenson, "The Theory and Practice of Compiler Writing", BS Publications, 2005*
3. *Dhamdhere, D. M., "Compiler Construction Principles and Practice", 2nd edition, Macmillan India Ltd., New Delhi, 2008*

Reference books

1. *Allen I. Holub, "Compiler Design in C", Prentice Hall of India, 2003*
2. *C. N. Fischer and R. J. LeBlanc, "Crafting a compiler with C", Benjamin Cummings, 2003*
3. *HenkAlblas and Albert Nymeyer, "Practice and Principles of Compiler Building with C", PHI, 2001*
4. *Kenneth C. Louden, "Compiler Construction: Principles and Practice", Thompson Learning, 2003*

List of Programme Elective Subjects

Course Code	:	**CSPE11**
Course Title	:	**Mobile Computing and Communication**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC27**
Course Type	:	**PE**

Objectives

- To understand the fundamentals of mobile communication.
- To understand the architecture of various Wireless Communication Networks.
- To understand the significance of different layers in mobile system

Unit – I

Introduction to Wireless Networks – Applications – History – Simplified Reference Model – Wireless transmission – Frequencies – Signals – Antennas – Signal propagation – Multiplexing – Modulation – Spread spectrum – Cellular Systems: Frequency Management and Channel Assignment- types of hand-off and their characteristics.*

Unit – II

MAC – Motivation – SDMA, FDMA, TDMA, CDMA –Telecommunication Systems – GSM: Architecture-Location tracking and call setup- Mobility management- Handover- Security- GSM SMS –International roaming for GSM- call recording functions-subscriber and service data management – DECT – TETRA – UMTS – IMT-2000.*

Unit – III

Wireless LAN – Infrared Vs Radio transmission – Infrastructure – Adhoc Network –IEEE 802.11WLAN Standards – Architecture – Services– HIPERLAN – Bluetooth Architecture & protocols.*

Unit – IV

Mobile Network Layer – Mobile IP – Dynamic Host Configuration Protocol - Mobile Transport Layer – Traditional TCP – Indirect TCP – Snooping TCP – Mobile TCP – Fast retransmit / Fast recovery – Transmission / Time-out freezing – Selective retransmission – Transaction Oriented TCP.*

Unit – V

WAP Model- Mobile Location based services -WAP Gateway –WAP protocols – WAP user agent profile- caching model-wireless bearers for WAP - WML - WML Scripts - WTA – iMode - SyncML.*

*Programming assignments are mandatory.

Outcomes

- Ability to develop a strong grounding in the fundamentals of mobile Networks
- Ability to apply knowledge in MAC, Network, and Transport Layer protocols of Wireless Network
- Ability to comprehend, design, and develop a lightweight network stack

Text Books

1. *Jochen Schiller, " Mobile Communication", 2nd Edition,Pearson Education, 2008.*
2. *Theodore and S. Rappaport, "Wireless Communications, Principles, Practice", 2nd Ed PHI, 2002*

Reference Books

1. *William Stallings, "Wireless Communications and Networks", 2nd Edition, Pearson Education, 2004*
2. *C.Siva Ram Murthy and B.S.Manoj, "Adhoc Wireless Networks: Architectures and Protocols", 2nd Edition, Pearson Education, 2008*
3. *Vijay. K. Garg, "Wireless Communication and Networking", Morgan Kaufmann Publishers, 2007.*

Course Code	:	**CSPE12**
Course Title	:	**Design and Analysis of Parallel Algorithms**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC29**
Course Type	:	**PE**

Objectives

- To understand parallel computing algorithms and models
- To analyze parallel algorithms for PRAM machines and various interconnection networks

Unit – I

Introduction to Parallel Computers - SIMD - EREW, CREW - SM-SIMD algorithms - Shared memory SIMD - Tree and mesh interconnection computers - Classifying MIMD Algorithms - Hypercube SIMD Model.*

Unit –II

Selection and Sorting – Sequential algorithm - Algorithm for parallel selection - Sorting on a linear array – broadcasting a datum- Computing all sums- Sorting on a mesh - Sorting on EREW SIMD computer - enumeration sort – parallel quick sort – hyper quicksort Sorting on other networks - Sorting on other networks.*

Unit – III

Matrix operations - Mesh transpose – Shuffle transpose - EREW transpose - Mesh multiplication - Cube multiplication - Matrix by vector multiplication - Tree multiplication.*

Unit – IV

Numerical problems - Linear equations - SIMD algorithm - Roots of nonlinear equations MIMD algorithm - Partial differential equations - Computing Eigen values. Monte Carlo methods – parallel random number generators – random number distributions.*

Unit – V

Graph problems –Definitions - Graph coloring - Computing the connectivity matrix - Finding connected components - Traversal - Minimal alpha-beta tree - Minimum Cost Spanning Tree-Addition tree-Multiplication tree.*

*Programming assignments are mandatory.

Outcomes

- Ability to analyze parallel algorithms for PRAM machines
- Ability to comprehend and apply parallel algorithms to real world applications
- Ability to design and develop optimal parallel algorithms

Text Book

1. *S. G. Akl, "The Design and Analysis of Parallel Algorithms", Prentice Hall of India, 1989*

Reference Books

1. *B. Wilkinson and M. Allen, "Parallel Programming – Techniques and applications using networked workstations and parallel computers", 2nd Edition, Pearson Education, 2005*
2. *Michael J. Quinn, "Parallel Computing : Theory & Practice", Tata McGraw Hill, 2003*
3. *S. Lakshmivarahan and S. K. Dhall, "Analysis and Design of Parallel Algorithms - Arithmetic and Matrix Problems", Tata McGraw Hill, 1990*

Course Code	:	**CSPE13**
Course Title	:	**Real Time Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC26**
Course Type	:	**PE**

Objectives

- To study issues related to the design and analysis of systems with real-time constraints.
- To learn the features of Real time OS.
- To study the various Uniprocessor and Multiprocessor scheduling mechanisms.
- To learn about various real time communication protocols.
- To study the difference between traditional and real time databases

Unit – I

Introduction to real-time computing - Structure of a real-time system - Characterization of real-time systems and tasks - Performance measures*

Unit – II

Task Assignment and Scheduling - Uniprocessor scheduling algorithms - Task assignment - Mode changes - Fault tolerant scheduling.*

Unit – III

Real-time Communication - Network topologies and architecture issues - Protocols - Contention-based, token-based, polled bus - Fault tolerant routing.*

Unit – IV

Real-time Databases - Transaction priorities and aborts - Concurrency control issues - Scheduling algorithms - Two-phase approach to improve predictability.*

Unit – V

Programming Languages and Tools - Hierarchical decomposition - Run-time error handling - Overloading - Timing specification - Recent trends and developments*

*Programming Assignments are mandatory

Outcomes

- Ability to analyze schedulability problems
- Ability to learn Real-time programming environments
- Ability to develop real time systems.

Text Book

1. *C. M. Krishna and Kang G. Shin, "Real-Time Systems", International Edition, McGraw Hill Companies, Inc., New York, 1997*

Reference Book

1. *Rajib Mall, "Real-Time Systems: Theory and Practice", 1st edition, Pearson Education, 2012*

Course Code	:	**CSPE14**
Course Title	:	**Data Warehousing and Data Mining**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC33**
Course Type	:	**PE**

Objectives

- To understand the principles of Data Warehousing and Data Mining
- To know the Architecture of a Data Mining system
- To perform classification, association, and prediction of data

Unit – I

Data Warehousing and Business Analysis: - Data warehousing Components –Building a Data warehouse –Data Warehouse Architecture – Data Warehouse Schemas for Decision Support – Data Extraction, Cleanup, and Transformation Tools –Metadata – reporting – Query tools and Applications – Online Analytical Processing (OLAP) – OLAP and Multidimensional Data Analysis.*

Unit – II

Data Mining: Databases – Steps in Data mining process- Data Mining Functionalities- Architecture of a Typical Data Mining Systems- Classification of Data Mining Systems. Data Pre-processing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation.*

Unit – III

Classification and Prediction: Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section. *

Unit – IV

Association Rule Mining: - Efficient and Scalable Frequent Item set Mining Methods – Mining Various Kinds of Association Rules – Association Mining to Correlation Analysis – Constraint-Based Association Mining. Cluster Analysis: - Types of Data in Cluster Analysis – A Categorization of Major Clustering Methods – Partitioning Methods – Hierarchical methods.*

Unit – V

Applications of Data mining-Social Impacts of Data mining-Tools- Mining the World Wide Web– Spatial Data Mining – Multimedia Data Mining – Text Mining. *

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend the various architectures and its application with data mining
- Ability to design and develop data mining algorithms to analyze raw real world data
- Ability to monitor and analyze to predict online digital activities

Text Book

1. *Jiawei Han, Micheline Kamber, and Jian Pei, "Data Mining Concepts and Techniques", Third Edition, Elsevier, 2011*

Reference Books

1 *Alex Berson and Stephen J. Smith "Data Warehousing, Data Mining & OLAP", Tata McGraw – Hill Edition, Tenth Reprint 2007*

2 *K.P. Soman, ShyamDiwakar and V. Ajay "Insight into Data mining Theory and Practice", Easter Economy Edition, Prentice Hall of India, 2006*

3 *G. K. Gupta "Introduction to Data Mining with Case Studies", Easter Economy Edition, Prentice Hall of India, 2006*

4 *Pang-Ning Tan, Michael Steinbach and Vipin Kumar "Introduction to Data Mining", Pearson Education, 2007*

Course Code	:	**CSPE15**
Course Title	:	**Wireless Network Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC32**
Course Type	:	**PE**

Objectives

- To understand the fundamentals of wireless communication
- To understand the architecture of different Wireless Networks
- To understand the significance of MAC and Network layers in Wireless Network System

Unit – I

Wireless Communications & Cellular System Fundamentals: Introduction to wireless communications systems, examples, comparisons and trends, Cellular systems, Frequency Management and Channel Assignment- types of handoff and their characteristics, dropped call rates & their evaluation. MAC techniques for Wireless Communication: FDMA, TDMA, MA(FHMA/CDMA/Hybrid techniques), SDMA techniques.*

Unit – II

Wireless WAN: First Generation Analog, Second Generation TDMA – GSM, Short Messaging Service in GSM, Second Generation CDMA – IS-95, GPRS - Third Generation Systems (WCDMA/CDMA 2000).*

Unit – III

Wireless LAN: Introduction to wireless LANs - IEEE 802.11 WLAN – Architecture and Services, Physical Layer, MAC sub layer- MAC Management Sub layer, Other IEEE 802.11 standards, HIPERLAN, WiMAX standard. *

Unit – IV

Ad hoc and Sensor Networks: Characteristics of MANETs, Table-driven and Source-initiated On Demand routing protocols, Hybrid protocols, Wireless Sensor networks- Classification, MAC and Routing protocols.*

Unit – V

Wireless MAN and PAN: Wireless MANs – Physical and MAC layer details, Wireless PANs – Architecture of Bluetooth Systems, Physical and MAC layer details, Standards. *

*Programming assignments are mandatory.

Outcomes

- Ability to make critical assessment of wireless networks
- Ability to comprehend the fundamentals of Wireless Networks
- Ability to apply the knowledge gained in the development of MAC, Network Layer protocols of Wireless Network

Text Books

1. *William Stallings, "Wireless Communications and networks" Pearson / Prentice Hall of India, 2nd Ed., 2007*
2. *Dharma Prakash Agrawal & Qing-An Zeng, "Introduction to Wireless and Mobile Systems", Thomson India Edition, 2nd Ed., 2007*
3. *Theodore, S. Rappaport, "Wireless Communications, Principles, Practice", 2nd Ed., PHI, 2002*

Reference Books

1. *C. Siva Ram Murthy and B. S. Manoj, "Adhoc Wireless Networks: Architectures and Protocols", 2nd Edition, Pearson Education, 2008*
2. *Jochen Schiller, "Mobile Communications", Person Education, 2nd Ed., 2008*
3. *Vijay. K. Garg, "Wireless Communication and Networking", Morgan Kaufmann Publishers, 2007*
4. *KavethPahlavan, Prashant Krishnamurthy, "Principles of Wireless Networks", Pearson Education Asia, 2002*

Course Code	:	**CSPE16**
Course Title	:	**Principles of Processor Design**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC31**
Course Type	:	**PE**

Objectives

- To understand the basics of Verilog HDL
- To study about the design aspects of various circuits using Verilog

Unit – I

Digital Design Flow in Verilog –Design entry –Test bench in Verilog - Design validation - Compilation and synthesis –Post synthesis simulation - Timing analysis - Hardware generation- Verilog HDL –Verilog evolution- Verilog attributes -Verilog language RT level design – Control/data partitioning - Data part- Control part- Elements of Verilog –Hardware modules - Primitive instantiations- Assign statements - Condition expression - Procedural blocks- Module instantiations- Component description in Verilog – Test benches.*

Unit – II

Verilog Language Concepts – Hardware languages-Timing- Concurrency- Timing and concurrency example – Module basics – Verilog simulation model –Continuous assignments- Procedural assignments- Compiler directives – System task and function.*

Unit – III

Combinational and Sequential Circuits Description - Module wires – Gate level logic – Hierarchical logic-Describing Expressions with Assign Statements- Behavioural Combinational Descriptions- Sequential models – Basic memory components – Functional registers – State machine coding – Combinational and sequential synthesis – Latches – Flip flops – Counters.*

Unit – IV

Design Examples – Bus structure – Simple processor – Timer – SRAM – Cache – Clock synchronization, Digital filters and signal processors-Pipelined Architectures-Halftone Pixel Image Converter.*

Unit – V

Register Transfer Level Design and Test – Sequential multiplier –Shift-and-add multiplication process- Sequential multiplier design - Multiplier testing- Von Neumann computer model – Processor and memory model- Processor model specification- Designing the adding CPU- Design of datapath - Control part design- Adding CPU Verilog description- Testing adding CPU- CPU design and test.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend the intricacies in processor design
- Ability to implement a CPU to exploit its full capability
- Ability to design and develop processor circuits using Verilog

Text Books

1. *ZainalabedinNavabi, "Verilog Digital System Design", 2nd Edition, McGraw Hill, 2008*
2. *Michael D. Ciletti, "Advanced Digital Design with the Verilog HDL" 2nd edition, Pearson Edition, 2009*

Course Code	:	**CSPE17**
Course Title	:	**Advanced Database Management Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC33, CSPE14**
Course Type	:	**PE**

Objectives

- To understand the different database models and language queries to access databases
- To understand the normalization forms in building an effective database tables
- To protect the data and the database from unauthorized access and manipulation

Unit – I

Relational Model Issues: ER Model - Normalization – Query Processing – Query Optimization –Transaction Processing - Concurrency Control – Recovery - Database Tuning

Unit – II

Distributed Databases: Parallel Databases – Inter and Intra Query Parallelism – Distributed Database Features – Distributed Database Architecture – Fragmentation – Distributed Query Processing – Distributed Transactions Processing – Concurrency Control – Recovery –Commit Protocols.*

Unit – III

Object Oriented Databases: Introduction to Object Oriented Data Bases - Approaches Modelling and Design - Persistence – Query Languages - Transaction - Concurrency – Multi Version Locks –Recovery – POSTGRES – JASMINE –GEMSTONE - ODMG Model.*

Unit – IV

Emerging Systems: Enhanced Data Models - Client/Server Model - Data Warehousing and Data Mining – Web Databases – Mobile Databases- XML and Web Databases.*

Unit – V

Current Issues: Rules - Knowledge Bases - Active and Deductive Databases - Multimedia Databases Multimedia Data Structures – Multimedia Query languages - Spatial Databases.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend the complex query processing techniques
- Ability to design and implement multimedia databases and writing query structure
- Ability to develop skill set in file organization, Query Optimization, Transaction management, and database administration techniques

Text Book

1. *Thomas Connolly and CarlolynBegg, "Database Systems: A Practical Approach to Design, Implementation, and Management", 5th Edition, Addison-Wesley, 2009*

Reference Books

1. *R. Elmasri and S. B. Navathe, "Fundamentals of Database Systems", Fifth Edition, Pearson/Addison Wesley, 2006*
2. *Abraham Silberschatz, Henry F. Korth, and S. Sudharshan, "Database System Concepts",Fifth Edition, Tata McGraw Hill, 2006*
3. *C. J. Date, A. Kannan, and S. Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006*

Course Code	:	**CSPE18**
Course Title	:	**Advanced Cryptography**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC35**
Course Type	:	**PE**

Objectives

- To study the concepts of applied cryptography
- To understand the application of cryptographic techniques in real world applications
- To comprehend the notion of provable security and its implication with improved security guarantees

Unit – I

Review of number theory, group, ring and finite fields, quadratic residues, Legendre symbol, Jacobi symbol.*

Unit – II

Formal Notions of Attacks: Attacks under Message Indistinguishability: Chosen Plaintext Attack(IND-CPA), Chosen Cipher text Attacks (IND-CCA1 and IND-CCA2), Attacks under Message Non-malleability: NM-CPA and NM-CCA2, Inter-relations among the attack model.*

Unit – III

Public key cryptography, RSA cryptosystem, probabilistic encryption, homomorphic encryption, Elliptic curve cryptosystems, Blum-Goldwasser cryptosystems, identity based encryption, Cryptographic hash functions.*

Unit – IV

Digital signatures and the notion of existential unforgability under chosen message attacks, ElGamal digital signature scheme, Schnorr signature scheme, blind signature, electronic voting.*

Unit – V

Zero Knowledge Proofs and Protocols, lattice based cryptography.*

*Programming assignments are mandatory.

Outcomes

- Ability to break cryptosystems that are not provably secure
- Ability to derive simple provable security proofs for cryptographic schemes
- Ability to design and implement cryptographic protocols

Text Books

1. *W. Mao, "Modern Cryptography: Theory & Practice", Pearson Education, 2010*
2. *Thomas Koshy, "Elementary Number Theory with applications", Elsevier India, 2005*
3. *Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman, "An Introduction to Mathematical Cryptography", Springer publication*
4. *Menezes, A, et.al., "Handbook of Applied Cryptography", CRC Press, 1996*
5. *Koblitz, N., "Course on Number Theory and Cryptography", Springer Verlag, 1986*

Course Code	:	**CSPE19**
Course Title	:	**Network Processor Design**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC32, CSPE16**
Course Type	:	**PE**

Objectives

- To understand the basics of networking and network processor architecture
- To understand basic concepts of processor scheduling and other parameters used for measuring performance of network processor

Unit – I

Introduction and motivation - Network processor Ecosystem-communication system and implementation-Network element - Networking Fundamentals - Converged Networks-Access and Home Networks- Network processor Architecture.*

Unit – II

Processor scheduling- Fibre channel/ Infiniband Implementation. Performance And Analysis Packet Processing-Framing-parsing and classification- search , Lookup and Forwarding-Compression and encryption- Queueing and Traffic Management-Packet flow handling-NP Peripherals.*

Unit – III

Worst Case Execution Time Estimation for Hardware Assisted Multithreaded processor-Power consideration in NP Design. Performance and Programmability of processing Element Topologies for NP-Packet classification Termination in a Protocol-Programmable Protocol Processor-Control memory Aceess Accelerator- System performance.*

Unit – IV

Efficient and Faithful Performance Modeling for NP Based system designs - Direction in Packet Classification for Network Processors. A Network Processor: EZchip - EZchip Architecture, Capabilities, and Applications- EZchip Programming-Parsing-Searching-Resolving-Modifying.*

Unit – V

Running the Virtual Local Area Network Example - Writing Your First High-Speed Network Application. Implementing High performance, High-value Traffic management using Agere Network Processor Solutions- Nepal: A Framework for Efficiently structuring Applications for NP.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend the network processor and its communication mechanisms
- Ability to implement various programming aspects of network processors
- Ability to design and develop optimal Network Processor

Text Books

1. *Ran Giladi, "Network Processors Architecture, Programming, and Implementation", Morgan Kaufmann Publishers, 2008*
2. *Patric Crowley, Mark A. Franklin , HaldunHadimioglu, and Peter Z. Onufryk, "Network Processors Design: Issues and Practices (Volume-2)", 2004*

Course Code	:	**CSPE20**
Course Title	:	**Programming for Embedded Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC36**
Course Type	:	**PE**

Objectives

- To understand basics of embedded system programming
- To know how the intricacies of Embedded programming

Unit – I

Introduction to Embedded System Programming: Application of Embedded System, Overview of Embedded System Architecture and Instruction Set, Real Time Systems, Requirements for Embedded Systems, Embedded Software Development: Challenges and Issues, Operating Systems for Embedded Systems: Introduction and Features, Languages for Embedded System Programming.*

Unit – II

Getting Started with Embedded Programming: Assembly verses High Level language, Integrated Development Environment, Building Process for Embedded System, Types of Memory for Embedded System, Memory Management methods and Bug Handling, Interrupts and ISRs handling in Embedded Systems, Simulators and Debuggers for Embedded System.*

Unit – III

Designing Elements of Embedded System Program: Basic Input Output Device Interface Programming, Developing Programmable Interrupt Controller, Timers and Counters, LCD hardware and Programming, Analog to Digital Clock, Introduction to data EEPROM.*

Unit – IV

Real Time Programming for Embedded System: Scheduling in Real Time Environment, Real Time Clock Designing, Real Time Operating System Support for Programming, Task Management in Real Time Environment, Semaphores handling, Message Queuing: States, Content, Storage, Introduction to Kernel Objects.*

Unit –V

Case Study on Embedded System Programming: Cruise Controller in Transportation, Bioinformatics on Embedded System, Mobile Phones and Handheld Devices, Applications in Medical Field, Low Power Systems, Reconfigurable Systems, Wireless Communication in Embedded Systems, Wearable Embedded Systems.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend the importance of Embedded programming for real time systems
- Ability to analyze and design embedded systems for smart applications
- Ability to design and develop application Specific embedded System

Text Books

1. *Julio Sanchez and Maria P. Canton, "Embedded Systems Circuits and Programming", Taylor and Francis, 2012*
2. *Michael Barr and Anthony Massa, "Programming Embedded Systems: With C and GNU Development Tools, O'Reilly, 2007*
3. *Sriram V Iyer and Pankaj Gupta, "Embedded Real Time System Programming", Tata McGraw Hill, 2004*
4. *Qing Li and Caroline Yao, "Real-Time Concepts for Embedded Systems", Elsevier, 2003*
5. *Cracking the Code Programming For Embedded System, Dreamtech Software Team, Wiley, 2002*

Course Code	:	**CSPE21**
Course Title	:	**Machine Learning**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC25**
Course Type	:	**PE**

Objectives

- To understand the basic building blocks and general principles that allow one to design machine learning algorithms
- To become familiar with specific, widely used machine learning algorithms
- To learn methodology and tools to apply machine learning algorithms to real data and evaluate their performance

Unit – I

Basic Concepts, Introduction to Machine Learning, Applications of ML, Design Perspective and Issues in ML, Supervised, Unsupervised, Semi-supervised learning with applications and issues, A Formal Learning Model, The Runtime of Learning.*

Unit – II

Model (or hypothesis) representation, decision boundary, cost function, gradient descent, regularization, Diagnostic: debugging a learning algorithm, evaluating a hypothesis (Model selection), training/validating/testing procedures, diagnosing bias versus variance and vice versa, regularization and bias/variance, learning curves, Accuracy and Error measures: classifier accuracy measures, predictor error measure, evaluating the accuracy of a classifier or predictor, Confusion metric, precision, recall, tradeoff between both, accuracy.*

Unit – III

Decision Tree : representation, hypothesis, issues in Decision Tree Learning, Pruning, Rule extraction from Tree, Learning rules from Data, Probabilistic classifier: Bayes rule, Maximum Likelihood Estimation, case study, Support Vector Machine, Nearest Neighbor.*

Unit – IV

Clustering: Unsupervised learning technique, Similarity and Distance Measures, k-means and k-medoids algorithm, optimization objective, random initialization, choosing value of k, EM algorithm Bayesian networks, bag of words classifiers, N-gram models; Markov and Hidden Markov models, Graphical Models, Combining Multiple Learners.*

Unit –V

Reinforcement Learning: Elements of Reinforcement Learning, Model-Based Learning, Temporal Difference Learning, Generalization, Design and Analysis of Machine Learning Experiments.*

*Programming assignments are mandatory.

Outcomes

- Ability to implement and apply machine learning algorithms to real-world applications.
- Ability to identify and apply the appropriate machine learning technique to classification, pattern recognition, optimization and decision problems.
- Ability to understand how to perform evaluation of learning algorithms and model selection.

Text Book

1. *Shai Shalev-Shwartz, Shai Ben-David, Understanding Machine Learning From Theory to Algorithms, Cambridge University Press, 2014*

Reference Books

1. *Ethem Alpaydin, Introduction to Machine Learning, PHI, 2005*
2. *H. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and TechniquesMorgan Kaufmann, 2000*
3. *Tom Mitchell, Machine Learning, McGraw-Hill, 1997*

Course Code	:	**CSPE22**
Course Title	:	**Randomized Algorithms**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC29**
Course Type	:	**PE**

Objectives

- To introduce the concept of randomized algorithms
- To apply the concepts of probabilistic analysis of algorithms

Unit – I

Elements of probability theory, Verification of strings, poly identities, matrix multiplication Las Vegas and MonteCarlo algorithms, Expectations, Jensen's Inequality, Coupon collector's problem, geometric distribution.*

Unit – II

Randomized Quick Sort and its expected run-time, Variance and moments, Chebyshev's inequality, Coupon collector's problem, randomized median finding, analysis, moment generating functions.*

Unit – III

Derivation and application of Chernoff's bounds, Sum of Poisson Trials, Coin flips, Set balancing, Packet routing in sparse networks, permutation routing on the hypercube, butterfly.*

Unit – IV

Birthday paradox, balls and bins model, application to bucket sort, Poisson distribution, Application to hashing, random graph models, Hamiltonian cycles in random graphs.*

Unit – V

Markov chains, representations, randomized algorithm for 2-satisfiability and3-satisfiability, classification of states, gambler's ruin, random walks on undirected graphs, s-t connectivity algorithm.*

*Programming assignments are mandatory.

Outcomes

- Ability to apply basics of probability theory in the analysis of algorithms
- Ability to comprehend randomized algorithms and its advantages to traditional algorithm
- Ability to design and implement randomized techniques in solving real world problems

Textbook

1. *M. Mitzenmacher and E. Upfal, "Probability and computing: Randomized algorithms and Probabilistic analysis", Cambridge, 2005*

Course Code	:	**CSPE23**
Course Title	:	**Natural Language Processing**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC28**
Course Type	:	**PE**

Objectives

- To understand the application of computational methods in linguists
- To apply statistical and probabilistic methods for parameter estimation and inference
- To know how the computational methods give insight into observed human language phenomena

Unit – I

Sound: Biology of Speech Processing; Place and Manner of Articulation; Word Boundary Detection; Argmax based computations; HMM and Speech Recognition.*

Unit – II

Words and Word Forms: Morphology fundamentals; Morphological Diversity of Indian Languages; Morphology Paradigms; Finite State Machine Based Morphology; Automatic Morphology Learning; Shallow Parsing; Named Entities; Maximum Entropy Models; Random Fields.*

Unit – III

Structures: Theories of Parsing, Parsing Algorithms; Robust and Scalable Parsing on Noisy Text as in Web documents; Hybrid of Rule Based and Probabilistic Parsing; Scope Ambiguity and Attachment Ambiguity resolution.*

Unit – IV

Meaning: Lexical Knowledge Networks, Wordnet Theory; Indian Language Wordnets and Multilingual Dictionaries; Semantic Roles; Word Sense Disambiguation; WSD and Multilinguality; Metaphors; Coreferences.*

Unit – V

Web 2.0 Applications: Sentiment Analysis; Text Entailment; Robust and Scalable Machine Translation; Question Answering in Multilingual Setting; Cross Lingual Information Retrieval (CLIR).*

*Programming Assignments are mandatory

Outcomes

- Ability to compare and contrast approaches to natural language processing
- Ability to comprehend and analyze the various elements of speech processing
- Ability to design and develop machine learning techniques in the area of NLP

Text books

1. *Jurafsky, Dan and Martin, James, "Speech and Language Processing", 2nd Edition, Prentice Hall, 2008*
2. *Manning, Christopher and Heinrich, Schutze, "Foundations of Statistical Natural Language Processing", MIT Press, 1999*
3. *Allen James, "Natural Language Understanding", 2nd edition, Benjamin Cumming, 1995*
4. *Charniack, Eugene, "Statistical Language Learning", MIT Press, 1993*

Course Code	:	**CSPE24**
Course Title	:	**Artificial Intelligence and Expert Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPE21**
Course Type	:	**PE**

Objectives

- To learn the concepts of Artificial Intelligence
- To learn the methods of solving problems using Artificial Intelligence
- To introduce the concepts of Expert Systems and machine learning

Unit – I

Introduction to AI, Control strategies, Search strategies, Production system characteristics - Specialized production system- Problem solving methods - Problem graphs, Matching, Indexing and Heuristic functions -Hill Climbing-Depth first and Breath first, Constraints satisfaction - Related algorithms, Measure of performance and analysis of search algorithms.*

Unit – II

Game playing - Knowledge representation, Knowledge representation using Predicate logic, Introduction to predicate calculus, Resolution, Use of predicate calculus, Knowledge representation using other logic-Structured representation of knowledge.*

Unit – III

Knowledge representation -Production based system, Frame based system. Inference – Backward chaining, Forward chaining*, Rule value approach, Fuzzy reasoning - Certainty factors, Bayesian Theory-Bayesian Network-Dempster - Shafer theory.

Unit – IV

Basic plan generation systems - Strips -Advanced plan generation systems – K strips –Strategic explanations -Why, Why not and how explanations. Learning- Machine learning, adaptive Learning

Unit – V

Expert systems - Architecture of expert systems, Roles of expert systems - Knowledge Acquisition –Meta knowledge, Heuristics. Typical expert systems - MYCIN, DART, XOON, Expert systems shells.

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend AI & ES to analyze and map real world activities to digital world
- Ability to identify problems that are amenably solved by AI methods
- Ability to design and carry out an empirical evaluation of different AI algorithms

Text Books

1. *Kevin Night and Elaine Rich, Nair B., "Artificial Intelligence (SIE)", McGraw Hill, 2008*
2. *Dan W. Patterson, "Introduction to AI and ES", Pearson Education, 2007*

Reference Books

1. *Peter Jackson, "Introduction to Expert Systems", 3rd Edition, Pearson Education, 2007*
2. *Stuart Russel and Peter Norvig, "AI – A Modern Approach, Pearson Education, 2nd Edition, 2007*
3. *G.Luger, W.A.Sttubblefield, "Artificial Intelligence", 3rd edition, Addison-Wesley Longman, 1998.*
4. *N.J.Nilson, "Principles of Artificial Intelligence", Narosa Publishing House, 1980.*

Course Code	:	**CSPE25**
Course Title	:	**Software Quality Assurance**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC34**
Course Type	:	**PE**

Objectives

- To understand software quality management process and quality management models
- To learn software quality metrics, assurance and various software standards

Unit – I

Defining Software Quality -Software Quality factors - Components of software quality assurance - pre project software quality components- Contract Review - Development and Quality Plans

Unit-II

Integrating Quality Activities in the Project Life Cycle – Reviews - Software Testing – Strategies - Software Testing –Implementation - Assuring the Quality of Software Maintenance - Assuring The Quality of External Participants' Parts - Case Tools and their Affect on Software Quality.

Unit-III

Software Quality Infrastructure Components- Procedures and Work Instructions - Supporting Quality Devices - Staff Training, Instructing and Certification - Preventive and Corrective Actions - Configuration Management - Documentation and Quality Records Controls

Unit-IV

Management Components Software Quality - Project Progress Control- Components, Internal & External Participants, Progress control regimes, Computerized tools, Software Quality Metrics – Objective, Classification, Process & Product Metrics, Implementation & Limitation of Software Metrics - Software Quality Costs – Objective, Classification Model of cost, Extended Model and Applications

Unit-V

Standards, Certification And Assessment - Need for standards, SQA Standards – ISO9001 Certification - Software Process Assessment, Organizing for Quality Assurance -Management and its Role in Quality Assurance - The Software Quality Assurance Unit - SQA Trustees and Committees, Six Sigma concepts.

Outcomes

- Ability to comprehend industrial standards in maintaining SQA
- Ability to apply basic software quality assurance practices to ensure software quality and standards
- Ability to design and model software projects that conform to international quality standards and practices

Text Books

1. *Daniel Galin, "Software Quality Assurance: From Theory to Implementation", Pearson Addison-Wesley, 2nd edition, 2012*
2. *Jeff Tian, "Software Quality Engineering: Testing, Quality Assurance, and Quantifiable", Wiley, 2005*

Course Code	:	**CSPE26**
Course Title	:	**Parallel Architectures and Programming**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC26, CSPC31**
Course Type	:	**PE**

Objectives

- To understand the fundamental principles and engineering trade-offs involved in designing modern parallel computers
- To develop programming skills to effectively implement parallel architecture

Unit – I

Introduction: The need for parallelism, Forms of parallelism (SISD, SIMD, MISD, MIMD), Moore's Law and Multi-cores, Fundamentals of Parallel Computers, Communication architecture, Message passing architecture, Data parallel architecture, Dataflow architecture, Systolic architecture, Performance Issues.*

Unit – II

Large Cache Design: Shared vs. Private Caches, Centralized vs. Distributed Shared Caches, Snooping-based cache coherence protocol, directory-based cache coherence protocol, Uniform Cache Access, Non-Uniform Cache Access, D-NUCA, S-NUCA, Inclusion, Exclusion, Difference between transaction and transactional memory, STM, HTM.*

Unit – III

Graphics Processing Unit: GPUs as Parallel Computers, Architecture of a modern GPU, Evolution of Graphics Pipelines, GPGPUs, Scalable GPUs, Architectural characteristics of Future Systems, Implication of Technology and Architecture for users, Vector addition, Applications of GPU.*

Unit – IV

Introduction to Parallel Programming: Strategies, Mechanism, Performance theory, Parallel Programming Patterns: Nesting pattern, Parallel Control Pattern, Parallel Data Management, Map: Scaled Vector, Mandelbrot, Collative: Reduce, Fusing Map and Reduce, Scan, Fusing Map and Scan, Data Recognition: Gather, Scatter, Pack , Stencil and Recurrence, Fork-Join, Pipeline.*

Unit – V

Parallel Programming Languages: Distributed Memory Programming with MPI: trapezoidal rule in MPI, I/O handling, MPI derived datatype, Collective Communication, Shared Memory Programming with Pthreads: Conditional Variables, read-write locks, Cache handling, Shared memory programming with Open MP: Parallel for directives, scheduling loops, Thread Safety, CUDA: Parallel programming in CUDA C, Thread management, Constant memory and Event, Graphics Interoperability, Atomics, Streams.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend parallel architecture and its importance in solving engineering problems
- Ability to design parallel programs to enhance machine performance in parallel hardware environment
- Ability to design and implement parallel programs in modern environments such as CUDA, OpenMP, etc.

Text Books

1. *D. E. Culler, J. P. Singh, and A. Gupta, "Parallel Computer Architecture", Morgan-Kaufmann, 2004*
2. *Rajeev Balasubramonian, Norman P. Jouppi, and Naveen Muralimanohar, "Multi-Core Cache Hierarchies", Morgan & Claypool Publishers, 2011*
3. *Peter and Pach Eco, "An Introduction to Parallel Programming", Elsevier, 2011*
4. *James R. Larus and Ravi Rajwar, "Transactional Memory", Morgan & Claypool Publishers, 2007*
5. *David B. Kirk, Wen-mei W. Hwu, "Programming Massively Parallel Processors: A Hands-on Approach", 2010*
6. *Barbara Chapman, F. Desprez, Gerhard R. Joubert, Alain Lichnewsky, Frans Peters "Parallel Computing: From Multicores and GPU's to Petascale", 2010*
7. *Michael McCool, James Reinders, Arch Robison, "Structured Parallel Programming: Patterns for Efficient Computation", 2012*
8. *Jason Sanders, Edward Kandrot, "CUDA by Example: An Introduction to General-Purpose GPU Programming", 2011*

Course Code	:	**CSPE27**
Course Title	:	**Service Oriented Architecture**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**PE**

Objectives

- To provide an overview of XML Technology and modeling databases in XML
- To provide an overview of Service Oriented Architecture and Web services and their importance
- To introduce Security solutions in XML and Web Services and to introduce Security standards for Web Services

UNIT-I

XML Technology: XML – XML and Web - Name Spaces – XML Document Structure - Structuring with Schemas and DTD - Modeling Databases in XML – XQuery. *

UNIT - II

SOA Basics: Service Oriented Architecture (SOA) – Comparing SOA with Client-Server and Distributed architectures - Characteristics of SOA – Benefits of SOA -- Principles of Service orientation – Service layers - Business Process management

UNIT - III

Web Services (WS): SOA and Web Services – Web Services Protocol Stack – Service descriptions – WSDL – Messaging with SOAP – Service discovery – UDDI. Service-Level Interaction patterns – XML and Web Services - Enterprise Service Bus - .NET and J2EE Interoperability. *

UNIT - IV

WS Technologies and Standards: Web Services Technologies - JAX-RPC, JAX-WS.* Web Service Standards – WS-RM, WS-Addressing, WS-Policy.Service Orchestration and Choreography – Composition Standards - BPEL. Service Oriented Analysis and Design.

UNIT- V

XML and WS Security: XML Security Overview – Canonicalization – XML Security Framework – XML Encryption – XML Signature – XKMS Structure. Web Services Security - XACML - WS-Security.

*Programming assignments are mandatory.

Outcomes

- Ability to design and develop real work applications using the concepts of SOA and Web services
- Ability to comprehend approaches for providing security for XML documents as well as messages exchanged among Web Services
- Ability to develop an application using .NET and J2EE enterprise technology

Text Books

1. *Ron Schmelzer et al. "XML and Web Services", Pearson Education, 2008. (Unit 1 and 3)*
2. *Thomas Erl, " Service Oriented Architecture: Concepts, Technology, and Design", Pearson Education, 2005 (Unit 2, 3, 4, and 5)*
3. *Frank P. Coyle, "XML, Web Services and the Data Revolution", Pearson Education, 2002 (Unit 5)*

Reference Books

1. *Eric Newcomer, Greg Lomow, "Understanding SOA with Web Services", Addison Wesley, 2005.*
2. *James McGovern, Sameer Tyagi, Michael E Stevens, Sunil Mathew, "Java Web Services Architecture", Elsevier, 2011.*
3. *Mark O' Neill, et al., "Web Services Security", Tata McGraw-Hill Edition, 2003*
4. *Sandeep Chatterjee and James Webber, "Developing Enterprise Web Services: An Architect's Guide", Prentice Hall, 2004*

Course Code	:	**CSPE28**
Course Title	:	**Data Sciences**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC33, CSPE14**
Course Type	:	**PE**

Objectives

- To understand the concepts of machine learning
- To appreciate supervised and unsupervised learning and their applications
- To learn aspects of computational learning theory

Unit – I

Introduction to Data Science - Overview of the Data Science process -Introduction to Data science technologies -Introduction to Machine Learning – Regressions –Classification-Clustering-Recommendation.*

Unit – II

Working with Data in Azure ML- Data Acquisition - Data Ingestion and Ingress-Data Sampling and Quantization-Data Cleaning and Transformation.*

Unit – III

Building and Evaluation of Models- Data Exploration and Visualization of Models- Business Metrics and Cost-Based Metrics-Model Evaluation- Comparison and Selection.*

Unit – IV

Supervised learning-Logistic regression- Perceptron- Exponential family.-Generative learning algorithms - Gaussian discriminant analysis- Naive Bayes -Support vector machines-Model selection and feature selection-Ensemble methods: Bagging, boosting-Evaluating and debugging learning algorithms.*

Unit –V

Unsupervised learning -Clustering- K-means - EM - Mixtures of Gaussians - The EM Algorithm in General -Model selection for latent variable models - high-dimensional spaces -- The Curse of Dimensionality -Dimensionality Reduction - Factor analysis - Principal Component Analysis - Probabilistic PCA- Independent components analysis.*

*Programming assignments are mandatory.

Outcomes

- To implement a neural network for an application of your choice using an available tool
- To implement probabilistic discriminative and generative algorithms for an application of your choice and analyze the results
- To use a tool to implement typical clustering algorithms for different types of applications

Text Books

1. Christopher Bishop, “Pattern Recognition and Machine Learning” Springer, 2006
2. Kevin P. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012
3. Hastie, Tibshirani, Friedman, “The Elements of Statistical Learning” (2nd ed)., Springer, 2008

Minors Offered

Course Code	:	**CSMI11**
Course Title	:	**Data Structures and Algorithms**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**MI**

Objectives

- To understand the various techniques of sorting and searching
- To design and implement arrays, stacks, queues, and linked lists
- To understand the complex data structures such as trees and graphs
- To design and implement various programming paradigms and its complexity

Unit – I

Development of Algorithms - Notations and analysis - Storage structures for arrays - Sparse matrices - Stacks and Queues: Representations and applications. linked lists –Doubly linked lists- Circular linked lists.

Unit – II

Trees: Preliminaries – Binary Trees – Search Tree ADT – Binary Search Trees – Hashing: ADT – Hash Function – Separate Chaining – Open Addressing – Rehashing – Extendible Hashing

Unit – III

Graphs - Representation of graphs - BFS, DFS - Topological sort- Shortest path problems - Dijkstra's algorithm, Floyd-Warshall, Minimum spanning trees- prims algorithm, Kruskal algorithm.

Unit – IV

Algorithmic paradigms - Divide and Conquer method - Strassen's matrix multiplication - Greedy method - Knapsack problem - Job sequencing with deadlines – Dynamic Programming- Travelling salesman problem.

Unit – V

Searching and Sorting Techniques - Selection, Bubble, Insertion, Merge, Quick, and Radix sort - Address calculation - Linear search - Binary search .

Outcomes

- Ability to develop programs to implement linear data structures such as stacks, queues, linked lists, etc.
- Ability to apply the concept of trees and graph data structures in real world scenarios
- Ability to comprehend the implementation of sorting and searching algorithms

Text Books

1. *J. P. Tremblay and P. G. Sorenson, "An Introduction to Data Structures with applications", Second Edition, Tata McGraw Hill, 1981*
2. *T. Cormen, C. Lieserson, R. Rivest, and C. Stein, "Introductions to Algorithms", Prentice-Hall/India, 3rd edition, 2009*
3. *M. Tenenbaum and Augestien, "Data Structures using C", Third Edition, Pearson Education 2007.*

Reference Book

1. *SartajSahni, "Data Structures, Algorithms and Applications in C++", Universities Press (I) Pvt. Ltd.*

Course Code	:	**CSMI12**
Course Title	:	**Computer Organization**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**MI**

Objectives

- To understand the basic hardware and software issues of computer organization
- To understand the representation of data at machine level
- To understand how computations are performed at machine level

Unit – I

Introduction, Technologies for building Processors and Memory, Performance, The Power Wall, Operations of the Computer Hardware, Operands Signed and Unsigned numbers, Representing Instructions, Logical Operations, Instructions for Making Decisions

Unit – II

MIPS Addressing for 32-Bit Immediate and Addresses, Parallelism and Instructions: Synchronization, Translating and Starting a Program, Addition and Subtraction, Multiplication, Division, Floating Point, Parallelism and Computer Arithmetic: Subword Parallelism, Streaming SIMD Extensions

Unit – III

Logic Design Conventions, Building a Datapath, A Simple Implementation Scheme, overview of Pipelining, Pipelined Datapath, Data Hazards: Forwarding versus Stalling, Control Hazards, Exceptions, Parallelism via Instructions

Unit – IV

Memory Technologies, Basics of Caches, Measuring and Improving Cache Performance, dependable memory hierarchy, Virtual Machines, Virtual Memory, Using FSM to Control a Simple Cache, Parallelism and Memory Hierarchy: Redundant Arrays of Inexpensive Disks

Unit – V

Disk Storage and Dependability, Parallelism and Memory Hierarchy: RAID levels, Performance of storage systems, Introduction to multi threading clusters, message passing multiprocessors.

Outcomes

- Ability to analyze the abstraction of various components of a computer
- Ability to analyze the hardware and software issues and the interfacing
- Ability to work out the tradeoffs involved in designing a modern computer system

Text Book

1. *David A. Patterson and John L. Hennessey, "Computer organization and design, The Hardware/Software interface", Morgan Kauffman / Elsevier, Fifth edition, 2014*
2. *Smruti Ranjan Sarangi, "Computer Organization and Architecture", McGraw Hill Education, 2015*

Reference Book

1. *V. Carl Hamacher, Zvonko G. Varanesic, and Safat G. Zaky, "Computer Organization", 6th edition, McGraw-Hill Inc, 2012*
2. *William Stallings, "Computer Organization and Architecture", 8th Edition, Pearson Education, 2010*

Course Code	:	**CSMI13**
Course Title	:	**Operating Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSMI13**
Course Type	:	**MI**

Objectives

- To provide knowledge about the services rendered by operating systems
- To provide a detailed discussion of the various memory management techniques
- To discuss the various file-system design and implementation issues
- To discuss how the protection domains help to achieve security in a system

Unit – I

Basic OS Concepts- User's view of the OS - Architectural support – System calls- Thread and process scheduling - Pre-emptive and non-pre-emptive - FCFS, SJF, Round Robin, Multilevel Queue.

Unit – II

Inter process synchronization, Mutual exclusion algorithms, Hardware support, Semaphores, Concurrent programming using semaphores.

Unit – III

Inter process communication, Deadlocks: Characterization, Prevention, Avoidance, detection and recovery, combined approach to deadlock handling.

Unit – IV

Contiguous allocation, Static and dynamic partitioned memory allocation, Segmentation, Non-contiguous allocation, Paging, Hardware support, Virtual Memory, Demand Paging.

Unit – V

Need for files, File abstraction, File naming, File system organization, File system optimization, Reliability, Security and protection, I/O management and disk scheduling, Recent trends and developments.

Outcomes

- Ability to comprehend the techniques used to implement the process manager
- Ability to comprehend virtual memory abstractions in operating systems
- Ability to design and develop file system interfaces, etc.

Text Book

1. *Silberschatz, Galvin, Gagne, "Operating System Concepts", John Wiley and Sons, 9th edition, 2013*

References Books

1. *William Stallings, "Operating Systems –Internals and Design Principles", 8/E, Pearson Publications, 2014*
2. *Andrew S. Tanenbaum, "Modern Operating Systems", 4/E, Pearson Publications, 2014*

Course Code	:	**CSMI14**
Course Title	:	**Database Management Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**MI**

Objectives

- To learn data models, conceptualize and depict a database system using ER diagram
- To understand the internal storage structures in a physical DB design
- To know the fundamental concepts of transaction processing techniques

Unit – I

Introduction: Purpose of Database System -– Views of data – Data Models – Database Languages - Database System Architecture – components of DBMS –Entity–Relationship model (E-R model) – E-R Diagram notation, Examples.

Unit – II

Relational Model: Relational Data Model - Concept of relations, schema-instance distinction, keys, integrity rules, relational algebra operators, SQL: data definition, data manipulation, aggregate function, Null Values, nested sub queries, Joined relations. Work with MySQL Workbench.

Unit – III

Database Design: Dependencies and Normal forms, dependency theory - functional dependencies, Armstrong's axioms for FD's, closure of a set of FD's, minimal covers, definitions of 1NF, 2NF, 3NF and BCNF 4NF, and 5NF, decompositions and desirable properties of them.

Unit – IV

Transaction management: ACID properties, serializability and concurrency control, Lock based concurrency control (2PL, Deadlocks), database recovery management.

Unit – V

Implementation Techniques: Overview of Physical Storage Media – Magnetic Disks – RAID – Tertiary storage – Organization of Records in Files – Indexing and Hashing–Ordered Indices - primary, secondary index structures –Static Hashing – Dynamic Hashing.

Outcomes

- Ability to Install, configure, and interact with a relational database management system
- Ability to master the basics of SQL and construct queries using SQL
- Ability to design and develop a large database with optimal query processing

Text Books

1. *A. Silberschatz, Henry F. Korth, and S. Sudharshan, "Database System Concepts", 5th Ed, Tata McGraw Hill, 2006.*
2. *C. J. Date, A. Kannan and S. Swamynathan, "An Introduction to Database Systems", 8thed, Pearson Education, 2006.*

References Books

1. *RamezElmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", Fourth Edition, Pearson/Addisionwesley, 2007*
2. *Raghu Ramakrishnan, "Database Management Systems", Third Edition, McGraw Hill, 2003*
3. *S. K. Singh, "Database Systems Concepts, Design and Applications", First Edition, Pearson Education, 2006*

Course Code	:	**CSMI15**
Course Title	:	**Software Engineering**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	
Course Type	:	**MI**

Objectives

- To understand the Software Engineering Practice& Process Models
- To understand Design Engineering, Web applications, and Software Project Management
- To gain knowledge of the overall project activities.

Unit-I

The Evolving role of Software, The changing Nature of Software, Legacy software, A generic view of process, A layered Technology, A Process Framework, Software Development Life cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Incremental Process Model, Concurrent Development Model, The Unified Process.

Unit-II

Software Requirements, Functional & non-functional, Software requirements document, Requirement engineering process: Feasibility studies, Elicitation, Validation & Management, Software prototyping, Analysis and modeling.

Unit-III

Design Concepts and Principles – Modular design – Design heuristic – S/W architecture – Data design – Architectural design – Transform & Transaction mapping –Introduction to SCM process – Software Configuration Items.

Unit –IV

Software Testing, Taxonomy of S/W testing, Black box testing, Testing boundary conditions, Structural testing, Regression testing, S/W testing strategies : Unit testing, Integration testing Validation testing, System testing and debugging.

Unit-V

Software Project Management, S/W cost estimation, Function point models, COCOMO model, Project Scheduling, S/W maintenance.

Outcomes

- Ability to enhance the software project management skills
- Ability to comprehend the systematic methodologies involved in SE
- Ability to design and develop a software product in accordance with SE principles

TEXT BOOKS:

1. Roger S.Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill International edition, Seventh edition, 2009.
2. Ian Sommerville, Software Engineering, 9th Edition, Pearson Education, 2011.

REFERENCES:

1. Pankaj Jalote, "An Integrated Approach to software Engineering", Springer Verlag, 1997.

2. Pfleeger and Lawrence Software Engineering: Theory and Practice, Pearson Education, second edition, 2001

Course Code	:	**CSMI16**
Course Title	:	**Digital Systems Design**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**--**
Course Type	:	**MI**

Objectives

- To understand the essential knowledge on the fundamental of digital circuits
- To understand the overview on the design principles of digital computing systems

Unit - I

Binary codes - Weighted and non-weighted - Binary arithmetic conversion algorithms, Canonical and standard boolean expressions - Truth tables, K-map reduction - Don't care conditions - Adders / Subtractors - Carry look-ahead adder - Code conversion algorithms - Design of code converters - Equivalence functions.

Unit - II

Binary/Decimal Parallel Adder/Subtractor for signed numbers - Magnitude comparator - Decoders / Encoders - Multiplexers / Demultiplexers - Boolean function implementation using multiplexers.

Unit - III

Sequential logic - Basic latch - Flip-flops (SR, D, JK, T and Master-Slave) - Triggering of flip-flops - Counters - Design procedure - Ripple counters - BCD and Binary - Synchronous counters, Registers - Shift registers.

Unit - IV

Introduction to VLSI design - Basic gate design - Digital VLSI design - Design of general boolean circuits using CMOS gates. Verilog Concepts – Basic concepts – Modules & ports, Gate level modeling, Data flow modelling, Behavioral modeling, Tasks and functions.

Unit - V

Timing and delays – Switch level modelling, User defined primitives, Modeling Techniques

Outcomes

- Ability to design and implement complicated digital systems using Verilog
- Ability to design a VLSI circuit for an application
- Ability to comprehend the digital design logic

Text Books

1. *Morris Mano and Michael D. Ciletti, "Digital Design", 5th Ed, PHI, 2012*
2. *Samir Palnitkar, "Verilog HDL", 2nd Edition, Pearson Education, 2003*

Reference Books

1. *Michael D. Ciletti, "Advanced Digital Design with the Verilog HDL, 2nd Edition, Pearson Education, 2010*
2. *Stephen Brown, "Fundamentals of Digital Logic with Verilog", McGraw Hill, 2007*

Course Code	:	**CSMI17**
Course Title	:	**Data Communications and Networks**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**MI**

Objectives

- To provide insight about fundamental concepts and reference models (OSI and TCP/IP) and its functionalists
- To gain comprehensive knowledge about the principles, protocols, and significance of Layers in OSI and TCP/IP
- To know the implementation of various protocols and cryptography techniques

UNIT- I

Data Communications –Data Transmission- Multiplexing- Data Encoding Techniques-Introduction to computer networks -Network –Topologies- Reference Models: ISO/OSI Model and TCP/IP Model.

UNIT-II

Physical Layer-Transmission Media-Analog signals-Digital Signals-**Data Link Layer**-Error Detection and Correction – Parity – LRC-CRC – Hamming Code – Flow Control and Error Control – Stop and wait – ARQ – Sliding window –IEEE 802.3 Ethernet.

UNIT-III

Network Layer-Packet Switching and Circuit Switching – IP addressing methods – Subnetting – Supernetting-Routing Protocols: IP-ARP-RARP-DHCP-Routing Algorithms: Distance Vector Routing – Link State Routing.

UNIT-IV

Transport Layer-Transport Services – UDP -TCP - Congestion Control – Quality of Services(QOS).

UNIT-V

Application Layer-Domain Name Space (DNS) – Electronic Mail – HTTP-WWW.

Outcomes

- Ability to gain insight about basic network theory and layered communication architectures
- Ability to provide solutions to various problems in network theory
- Ability to conceptualize and design a network stack

Text Books

1. Andrew S. Tanenbaum and David J. Wetherall, “Computer Networks”, 5th edition, Prentice Hall, 2011
2. Behrouz A. Foruzan, “Data Communication and Networking”, 5th edition, Science Engineering& Math Publications, 2013

Reference Books

1. W. Stallings, "Data and Computer Communication", 10th Edition, Pearson Education, 2014

List of Open Elective Subjects

Course Code	:	**CSOE11**
Course Title	:	**Computer Graphics**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**OE**

Objectives:

- To understand the basics of various inputs and output computer graphics hardware devices.
- Exploration of fundamental concepts in 2D and 3D computer graphics.
- To know 2D raster graphics techniques, 3D modelling, geometric transformations, 3D viewing and rendering.

Unit - I

Basic of Computer Graphics: Applications of computer graphics, Display devices, Random and Raster scan systems, Graphics input devices, Graphics software and standards.*

Unit - II

Graphics Primitives: Points, lines, circles and ellipses as primitives, scan conversion algorithms for primitives, Fill area primitives including scan-line polygon filling, inside-outside test, boundary and flood-fill, character generation, line attributes, area-fill attributes, character attributers. *

OpenGL primitives: Functions, pipeline, sample programs for drawing 2-D, 3-D objects; event handling and view manipulation.*

Unit - III

2D transformation and viewing: Transformations, matrix representation, homogeneous coordinates, composite transformations, reflection and shearing, viewing pipeline and coordinates system, window-to-viewport transformation, clipping including point clipping, line clipping, polygon clipping.*

Unit - IV

3D concepts and object representation: 3D display methods, polygon surfaces, tables, equations, meshes, curved lies and surfaces, quadric surfaces, spline representation, cubic spline interpolation methods, Bazier curves and surfaces, B-spline curves and surfaces.*

3D transformation and viewing: 3D scaling, rotation and translation, composite transformation, viewing pipeline and coordinates, parallel and perspective transformation, view volume and general (parallel and perspective) projection transformations.*

Unit - V

Advance topics: visible surface detection concepts, back-face detection, depth buffer method, illumination, light sources, illumination methods (ambient, diffuse reflection, specular eflection), Color models: properties of light, XYZ, RGB, YIQ and CMY color models.*

*Programming assignments are mandatory.

Outcomes

- Ability to understand the various computer graphics hardware and display technologies.
- Ability to implement various 2D and 3D objects transformation techniques.
- Ability to apply 2D and 3D viewing technologies into the real world applications

Text Books:

1. *Computer Graphics; Principles and practice; Second Edition in C; J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes; Addison Wesley, 1997.*
2. *Computer Graphics - C version; D. Hearn and M. P. Baker; Pearson Education, 2004.*
3. *Computer Graphics - OpenGL version; D. Hearn and M. P. Baker; Pearson Education.*

Reference Books

1. *Mathematical elements for Computer Graphics; 2nd edn.; D. F. Rogers and J. A. Adams; McGraw-Hill International. Edn., 1990.*
2. *Computer Graphics using OpenGL; 2nd edn.; F. S. Hill Jr.; Pearson Education, 2003.*
3. *The OpenGL Reference Manual - The Bluebook, Version 1.4, Fourth Edition, Addison-Wesley.*
4. *The OpenGL Programming Guide - The Redbook, Version 2, Fifth Edition, Addison-Wesley.*
5. *OpenGL Programming Guide, 6th. Edition, Pearson Education, Shreiner, Woo, Neider, Davis, 2008.*

Course Code	:	**CSOE12**
Course Title	:	**Human Computer Interaction**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**OE**

Objectives

- To gain knowledge on the interplay between humans, tasks, technology, and contexts
- To gain knowledge on important human factors that affect HCI
- To be able to apply HCI principles, guidelines, methods, and techniques

Unit – I

Introduction to Human-computer Interaction - Methodology for Designing User-computer Interfaces -Task analysis -Conceptual, semantic, syntactic, and lexical models.*

Unit – II

Design of an interactive system - Interaction Styles -Question and answer -Form-based -Command language -Menus -Natural language -Direct manipulation -Virtual Reality -Augmented Reality -Other emerging interaction styles.*

Unit – III

Design and Evaluation Process -Prototyping -Testing and evaluating interface designs -Guidelines and criteria for designing UI, UI Software and Specifications -Languages and tools for specifying and building interfaces -Dialogue independence –UIMSLanguages and software abstractions -Programming support tools -. Basic Interaction Tasks, Techniques, and Devices.*

Unit – IV

Human Performance -Scientific foundations for designing user interfaces -Visual presentation of information -Graphical design -Designing experiments - Introduction to Research in Human-Computer Interaction -Why do HCI research? -Research prototypes -Interdisciplinary nature of HCI research -Examples of HCI research.*

Unit – V

New Interaction Techniques -New modes of human-computer communication -Voice Gesture -Eye movement -Tangible user interfaces -Brain-computer interfaces - Case Study.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend the basics of human and computational abilities and limitations
- Ability to evaluate the quality of a user interface
- Ability to apply appropriate HCI techniques to design systems that are usable by people

Text Books

1. *Wilbert O Galitz, "The Essential Guide To User Interface Design", Wiley Dreamatech, 3rd edition, 2007*
2. *Ben Shneidermann, "Designing The User Interface - Strategies for Effective Human-Computer Interaction", 4th Edition, Pearson Education Asia, 2004*
3. *Alan Dix, Janet Fincay, GreGoryd, Abowd, and Russell Bealg, "Human – Computer Interaction", 3rd edition, Pearson, 2003*
4. *Yvonne Rogers , Helen Sharp, and Jenny Preece, "Interaction Design: Beyond Human - Computer Interaction", 3rd edition, Wiley, 2011*

Course Code	:	**CSOE13**
Course Title	:	**Web Technology**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**OE**

Objectives

- To understand the basics of Web Designing using HTML, DHTML, and CSS
- To learn the basics about Client side scripts and Server side scripts

Unit - I

HTML- List, Tables, Images, Forms, Frames, Cascading Style sheets.* XML- Document type definition, XML Schemas,* Document Object model

Unit – II

Java Script -Control statements, Functions, Arrays, Objects, Events, Dynamic HTML with Java Script, Ajax. *

Unit – III

Web servers –IIS (XAMPP, LAMPP)and Tomcat Servers. Java Web Technologies- Servlets, JavaServer Pages, Java Server Faces, Web Technologies in Netbeans, Building a Web Application in Netbeans, JSF Components, Session Tracking, Cookies. *

Unit – IV

PHP- Basics, String Processing and Regular Expressions, Form Processing and Business Logic, Using Cookies *, Dynamic Content, Operator Precedence Chart

Unit – V

Database Connectivity with MySQL - Servlets, JSP, PHP.* Case Studies- Student information system, Health Management System.

*Programming assignments are mandatory.

Outcomes

- Ability to design and develop client side scripting techniques
- Ability to build real world applications using client side and server side scripting languages
- Ability to design and develop an e-Governance application using web technology

Text books

1. *Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, "Internet & World Wide Web How to Program", Deitel series, 5th edition, 2012*
2. *Jason Gilmore, "Beginning PHP and MySQL From Novice to Professional", 4th Edition, Apress Publications, 2010*

Reference Books

1. *Robert W. Sebesta, "Programming with World Wide Web", Pearson, 4th edition, 2008*
2. *David William Barron, "The World of Scripting Languages", Wiley Publications, 2000*

Course Code	:	**CSOE14**
Course Title	:	**Multimedia Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSOE11**
Course Type	:	**OE**

Objectives

- To understand the different media and design issues in multimedia systems
- To understand Multimedia security and data hiding for image/video

Unit – I

Multimedia Elements: Introduction – Definitions – Applications – Elements - Text – Image/Graphics Audio – Video – Animation.*

Unit – II

Data and File Formats : J Compression Techniques – Lossless, Lossy – JPEG, MPEG, TIFF, RIFF- H.261, H.J262, H.263 -File formats-Display Technologies.*

Unit – III

Multimedia Networks : Protocol - QOS Issues - RTP, RTCP, RTSP, SIP - Media on demand –ITV - STB Broadcast Schemes for VoD Buffer Management- Multimedia over wireless networks.*

Unit – IV

Multimedia Security and Forensics: Multimedia encryption - Digital Watermarking Security Attacks- Digital Forensics taxonomy, goals/requirements - Forensic Data Acquisition -Forensics Analysis and Validation.*

Unit – V

Multimedia Data Hiding: Overview– Data hiding framework-Key elements -Basic embedding mechanisms- Techniques for Embedding multiple bits-Quantitative model for Uneven embedding Capacity- CER-VER -Data Hiding in Binary image-Multilevel embedding-Multilevel image and video data hiding.*

*Programming assignments are mandatory.

Outcomes

- Ability to design multimedia components efficiently
- Ability to develop integrated, collaborative multimedia systems
- Ability to develop data hiding algorithms for the specialized applications

Text Books

1. *K. Andleigh, Kiran Thakrar , Multimedia Systems Design, PHI, 2007*
2. *ZeNian Li, S. Drew, "Fundamentals of Multimedia", PHI, 2006.*

Reference Books

1. *Ralf Steinmetz and Klara, "Multimedia Computing, Communications and Applications", Pearson Education, 2009*
2. *Min Wu, Bede Liu, "Multimedia Data Hiding", Springer-Verlag, 2002*
3. *I. Cox, M. Miller, and J. Bloom, "Digital Watermarking", Morgan Kaufman Publishers, 2001*
4. *Chun-Shien Lu, "Multimedia Security : Steganography and Digital Watermarking techniques for Protection of Intellectual Property", Springer Inc 2007*
5. *Wenjun Zeng, Heather Yu and Ching, Yung Lin, "Multimedia Security technologies for Digital rights Management", Elsevier Inc 2006*

Course Code	:	**CSOE15**
Course Title	:	**Cloud Computing**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSMI17**
Course Type	:	**OE**

Objectives

- To provide comprehensive knowledge of fundamental concepts and of cloud computing
- To demonstrate an understanding of Service models, deployment models, Virtualization
- To describe the programming and software environments of Cloud
- To shed light on the security issues in Cloud

Unit – I

Overview of Distributed Computing, Cluster Computing and Grid Computing – Technologies for Network based systems – Software environments for Distributed Systems and Clouds –Overview of Services and Service oriented Architecture.*

Unit – II

Virtual Machines and Virtualization – Implementation levels of Virtualization – Virtualization structures/tools and Mechanisms – Virtualization of CPU, Memory and I/O Devices – Storage Virtualization.*

Unit – III

Cloud Computing – Properties – challenges – Service models – IaaS, PaaS and SaaS Deployment models – Service Composition and orchestration – Architecture design of Compute and Storage cloud – Public Cloud Platforms – Inter Cloud Resource Management.*

Unit – IV

Cloud Programming and Software Environments – Parallel and Distributed Programming paradigms – Programming on AWS, Azure and GAE – Cloud software environments Eucalyptus – Open Stack – Open Nebula.*

Unit – V

Cloud Security – Infrastructure security – Data security – Identity and access management Privacy- Audit and Compliance.*

*Programming assignments are mandatory.

Outcomes

- Ability to articulate the Virtualization concepts
- Ability to identify the architecture, service models and deployment models of Cloud
- Ability to master the programming aspects of Cloud

Text Book

1. *Kai Hwang, Geoffrey C. Fox and Jack J. Dongarra, "Distributed and Cloud Computing from Parallel Processing to the Internet of Things", Morgan Kaufmann, Elsevier, 2012*

Reference Books

1. Barrie Sosinsky, "Cloud Computing Bible" John Wiley & Sons, 2010
2. Tim Mather, Subra Kumaraswamy, and Shahed Latif, "Cloud Security and Privacy An Enterprise Perspective on Risks and Compliance", O'Reilly 2009

Course Code	:	**CSOE16**
Course Title	:	**Network Security**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSMI17**
Course Type	:	**OE**

Objectives

- To understand the network security, services, attacks, mechanisms, types of attacks
- To comprehend and apply authentication services, authentication algorithms
- To comprehend and apply network layer security protocols, Transport layer security protocols, Web security protocols.

Unit -I

Overview of Network Security, Security services, attacks, Security Issues in TCP/IP suite-Sniffing, spoofing, buffer overflow, ARP poisoning, ICMP Exploits, IP address spoofing, IP fragment attack, routing exploits, UDP exploits, TCP exploits.*

Unit-II

Authentication requirements, Authentication functions - Message Authentication Codes - Hash Functions - Security of Hash Functions and MACs - MD5 message Digest algorithm - Secure Hash Algorithm - RIPEMD - HMAC Digital Signatures, Authentication protocols-Kerberos, X.509.*

Unit-III

IP Security-AH and ESP, SSL/TLS, SSH, Web Security-HTTPS, DNS Security, Electronic Mail Security (PGP, S/MIME).*

Unit-IV

Intruders, Viruses, Worms, Trojan horses, Distributed Denial-Of-Service (DDoS), Firewalls, IDS, Honey nets, Honey pots.*

Unit-V

Introduction to wireless network security, Risks and Threats of Wireless networks, Wireless LAN Security (WEP, WPA).*

*Programming assignments are mandatory.

Outcomes

- Ability to determine appropriate mechanisms for protecting the network.
- Ability to design and develop security solutions for a given application or system
- Ability to develop a secure network stack

Text Books

1. *W. Stallings, "Cryptography and Network Security: Principles and Practice", 5/E, Prentice Hall, 2013*
2. *Yang Xiao and Yi Pan, "Security in Distributed and Networking Systems", World Scientific, 2007, Chapter 1.*
3. *Aaron E. Earle, "Wireless Security Handbook", Auerbach publications, Taylor & Francis Group, 2006.*

Reference Books

1. *AtulKahate, "Cryptography and Network Security", Tata McGraw-Hill, 2003*

Course Code	:	**CSOE17**
Course Title	:	**Big Data Analytics**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSMI16**
Course Type	:	**OE**

Objectives

- To understand the financial value of big data analytics
- To explore tools and practices for working with big data
- To understand how big data analytics can leverage into a key component

UNIT – I

Introduction to Big Data: Analytics – Nuances of big data – Value – Issues – Case for Big data – Big data options Team challenge – Big data sources – Acquisition – Features of Big Data - Security, Compliance, auditing and protection - Evolution of Big data – Best Practices for Big data Analytics - Big data characteristics - Volume, Veracity, Velocity, Variety – Big Data Architecture – Big Data and Cloud.*

UNIT – II

Data Analysis: Evolution of analytic scalability – Convergence – parallel processing systems – Cloud computing – grid computing – Map reduce Basics – Map Reduce Algorithm Design - enterprise analytic sand box – analytic data sets – Analytic methods – analytic tools – Cognos – Microstrategy – Pentaho - Regression Modeling - Multivariate Analysis - Bayesian Modeling - Inference and Bayesian Networks - Support Vector and Kernel Methods.*

UNIT – III

Stream Computing : Introduction to Streams Concepts – Stream data model and architecture - Stream Computing, Sampling data in a stream – Filtering streams – Counting distinct elements in a stream – Estimating moments – Counting oneness in a window – Decaying window - Realtime Analytics Platform(RTAP) applications IBM Infosphere – Big data at rest – Infosphere streams – Data stage – Statistical analysis – Intelligent scheduler – Infosphere Streams.*

UNIT – IV

Predictive Analytics and Visualization: Predictive Analytics – Supervised – Unsupervised learning – Neural networks – Kohonen models – Normal – Deviations from normal patterns – Normal behaviours – Expert options – Variable entry - Mining Frequent itemsets - Market based model – Apriori Algorithm – Handling large data sets in Main memory – Limited Pass algorithm – Counting frequent itemsets in a stream – Clustering Techniques – Hierarchical – K- Means – Clustering high dimensional data Visualizations - Visual data analysis techniques, interaction techniques; Systems and applications.*

UNIT – V

Frameworks and Applications: IBM for Big Data – Map Reduce Framework - Hadoop – Hive – Sharding - MongoDB – NoSQL Databases - S3 - Hadoop Distributed file systems – Hbase – Impala – Analyzing big data with Twitter and Facebook – Big data for E-commerce – Big data for blogs.*

*Programming assignments are mandatory.

Outcomes

- Ability to apply the concepts of big data analytics for a domain
- Ability to design and develop Hadoop and Map Reduce Framework
- Ability to contextually integrate and correlate large amounts of information

Text Books

1. *Frank J Ohlhorst, "Big Data Analytics: Turning Big Data into Big Money", Wiley and SAS Business Series, 2012.*
2. *Colleen Mccue, "Data Mining and Predictive Analysis: Intelligence Gathering and Crime Analysis", Elsevier, 2007*
3. *Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.*
4. *AnandRajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2012.*
5. *Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", Wiley and SAS Business Series, 2012.*
6. *Paul Zikopoulos, Chris Eaton, Paul Zikopoulos, "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data", McGraw Hill, 2011.*
7. *Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch , James Giles, David Corrigan, "Harness the Power of Big data – The big data platform", McGraw Hill, 2012.*
8. *Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, 2007*
9. *Pete Warden, Big Data Glossary, O'Reilly, 2011.*
10. *Jiawei Han, Micheline Kamber "Data Mining Concepts and Techniques", Second Edition, Elsevier, Reprinted 2008.*
11. *Data-Intensive Text Processing with MapReduce, Jimmy Lin and Chris Dyer University of Maryland, College Park. Morgan & Claypool Publishers.*

Course Code	:	**CSOE18**
Course Title	:	**Image Processing**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSOE11**
Course Type	:	**OE**

Objectives

- To understand the fundamentals of Digital imaging and Image Processing techniques
- To be familiar with image compression and segmentation

Unit – I

Introduction: Fundamentals of Image Processing, Applications of Image Processing, Human Visual Perception, Introduction to Image Formation, Sampling and Quantization, Binary Image, Three-Dimensional Imaging, Image file formats. Color and Color Imagery: Perception of Colors.*

Unit – II

Image Transformation: Fourier Transforms, Discrete Cosine Transform, Walsh-adamard Transform, Karhaunen-Loeve Transform or PCA. Discrete Wavelet Transform: Wavelet Transform, Extension to 2D Signals, Lifting Implementation of the Discrete Wave Transforms.*

Unit – III

Image Enhancement and Restoration : Introduction, Distinction between image enhancement and restoration, Histrogram-based Contrast Enhancement, Frequency Domain Methods of Image Enhancement, Noise Modeling, Image Restoration, Image Reconstruction, Image Segmentation.*

Unit – IV

Recognition of Image Patterns : Introduction, Decision Theoretic Pattern Classification, Baesian Decision Theory, Nonparametric Classification, Linear Discriminant Analysis, Unsupervised Classification Strategies-clustering, K-means clustering algorithm, Syntactic Pattern Classification, Syntactic Inference, Symbolic Projection method. Texture and Shape Analysis .*

Unit – V

Fuzzy Set Theory in Image Processing : Introduction, Use of Fuzzy Image, Preliminaries and Background, Image as a Fuzzy Set, Fuzzy Methods of Contrast Enhancement, Image Segmentation using Fuzzy Methods, Fuzzy Approaches to Pixel Classification, Fuzzy c-Means Algorithm, Fusion of Fuzzy logic with neural network. Image mining and Content-Based Retrieval.*

*Programming assignments are mandatory.

Outcomes

- Ability to design and apply image enhancement and restoration techniques
- Ability to apply image compression and segmentation Techniques
- Ability to design and develop image processing techniques for assisting digital forensics

Text Book

1. *Maria Petrou and Costas Petrou , "Image Processing the Fundamentals", John-Wiley and Sons Publishers, 2nd edition, 2010*
2. *Rafael C. Gonzalez , Richard E. Woods, Steven L. Eddins, "Digital Image Processing Using MATLAB", 2nd edition, Gatesmark Publishing, 2009*
3. *Tinku Acharya and Ajoy K. Ray, "Image Processing Principles and Applications", John Wiley & Sons publishers, 2005*

Course Code	:	**CSOE19**
Course Title	:	**Internet of Things**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSMI17**
Course Type	:	**OE**

Objectives

- To learn the basic issues, policy and challenges in the Internet
- To get an idea of some of the application areas where Internet of Things can be applied.
- To understand the cloud and internet environment.
- To understand the various modes of communications with Internet.

Unit – I

Introduction: Definition – Foundations – Challenges and Issues - Identification - Security. Components in internet of things**:** Control Units – Sensors – Communication modules –Power Sources – Communication Technologies – RFID – Bluetooth – Zigbee – Wifi – Rflinks –Mobile Internet – Wired Communication-IoT Platform Overview-Raspberry pi-Arduino boards.*

Unit – II

IoT Protocols: Protocol Standardization for IoT-M2M and WSN Protocols-SCADA and RFID Protocols-Issues with Iot Standardization-Protocols-IEEE 802.15.4-BACNet Protocol-Zigbee Architecture - Network layer – APS Layer – Security.*

Unit – III

Resource Management in the Internet of Things: Clustering - Software Agents - Data Synchronization - Clustering Principles in an Internet of Things Architecture - The Role of Context - Design Guidelines -Software Agents for Object – Data Synchronization- Types of Network Architectures - Fundamental Concepts of Agility and Autonomy-Enabling Autonomy and Agility by the Internet of Things - The Evolution from the RFID-based EPC Network to an Agent based Internet of Things- Agents for the Behaviour of Objects.*

Unit – IV

Case Study and IoT Application Development: IoT applications in home- infrastructures-security-Industries- IoT electronic equipments. Use of Big Data and Visualization in IoT-Industry 4.0 concepts - Sensors and sensor Node –Interfacing using Raspberry Pi/Arduino- Web Enabled Constrained Devices.*

Unit – V

Web of Things: Web of Things versus Internet of Things-Architecture Standardization for WoT-Platform Middleware for WoT- WoT Portals and Business Intelligence-Cloud of Things:Grid/SOA and Cloud Computing-Cloud Standards –Cloud of Things Architecture-Open Source e-Health sensor platform.*

*Programming assignments are mandatory.

Outcomes

- Identify the components of IoT
- Analyze various protocols of IoT
- Design portable IoT using appropriate boards
- Develop schemes for the applications of IOT in real time scenarios
- Design business Intelligence and Information Security for WoT

Text Books

1. *Honbo Zhou, "The Internet of Things in the Cloud:A Middleware Perspective" — CRC Press-2012.*
2. *Dieter Uckelmann, Mark Harrison, "Architecting the Internet of Things", Springer-2011.*
3. *Arshdeep Bahga, Vijay Madisetti, "Internet of Things (A Hands-On-Approach)", VPT, 2014.*
4. *Olivier Hersent, David Boswarthick, Omar Elloumi, "The Internet of Things – Key applications and Protocols", Wiley, 2012.*

References

1. *Luigi Atzori, Antonio Lera, Giacomo Morabito, "The Internet of Things: A Survey", Journal on Networks, Elsevier Publications, October, 2010.*
2. *http://www.theinternetofthings.eu/what-is-the-internet-of-things.*

Course Code	:	**CSOE20**
Course Title	:	**Bitcoin and Crypto Currencies**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**OE**

Objectives

- To understand the basic concept of Cryptographic Hash Functions, Hash Pointers and Elliptic Curve Digital Signature Algorithm.
- To get an insight into the working of the Bitcoin network, wallet, Bitcoin mining and distributed consensus for reliability.

Unit – I

Introduction to Cryptography, Cryptographic Hash Functions, SHA-256 , Hash Pointers and Data Structures, Merkle tree.*

Unit – II

Digital Signatures, Elliptic curve group, Elliptic Curve Digital Signature Algorithm (ECDSA). Public Keys as Identities, A Simple Crypto currency. *

Unit – III

Centralization vs. Decentralization, Distributed consensus, Consensus without identity using a block chain, Incentives and proof of work. Bitcoin transactions, Bitcoin Scripts, Applications of Bitcoin scripts, Bitcoin blocks, The Bitcoin network.*

Unit – IV

Simple Local Storage, Hot and Cold Storage, Splitting and Sharing Keys, Online Wallets and Exchanges, Payment Services, Transaction Fees, Currency Exchange Markets.*

Unit – V

Bitcoin Mining, Mining pools, Mining incentives and strategies. Bitcoin and Anonymity: Anonymity Basics, Mixing, Zerocoin and Zerocash.*

*Programming assignments are mandatory.

Outcomes

- Able to understand the how Bitcoin and other crypto currencies work.
- Able to do mining job in Bitcoin transaction.

Text Books

1. *Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies, 2016.*

Course Code	:	**CSOE21**
Course Title	:	**Probability, Queuing Theory, and Statistics for CS**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**OE**

Objectives

- To refresh fundamentals in probability theory
- To focus on Queuing theory with relevant CS examples
- To introduce elements of statistical analysis

Unit – I

Basic notions of probability, Bernoulli trials, Random variables and associated parameters, distributions – binomial, geometric, Poisson, uniform, exponential, Gaussian, conditional probability, probability distributions, central limit theorem, transform methods.

Unit – II

Elements of a queuing system, Standard notations and definitions, Little's Law, birth- and – death process models, Poisson process and its properties.

Unit – III

M/M/1, M/M/m, M/G/1 queuing systems and their characteristics, Embedded Markov chains, other standard results from the literature, basic ideas of priority queuing systems.

Unit – IV

Modeling of computer systems, finite population models, Jackson networks and Baskett-Chandy-Muntz-Palacios generalizations, other examples from data networks.

Unit – V

Basics of statistical inference, estimators, confidence intervals, exploratory data analysis, hypothesis testing, test of means, variances, ANOVA, ideas in regression and correlation analyses.

Outcomes

- Ability to appreciate performance modeling of data networks
- Ability to solve Queuing theory problems relevant to CS

Text Books

1. A.O.Allen, Probability, Statistics and Queuing Theory with Computer Science Applications, 2/e, Academic Press, Indian Reprint 2006
2. S.K.Bose, An Introduction to Queuing Systems, Springer/New York, 2002.

Course Code	:	**CSOE22**
Course Title	:	**Software Project Management**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**-**
Course Type	:	**OE**

Objectives

- To understand the basic concepts and issues of software project management
- To understand successful software projects that support organization's strategic goals

Unit – I

SPM concepts: Definition – components of SPM – challenges and opportunities – tools and techniques – managing human resource and technical resource – costing and pricing of projects – training and development – project management techniques.*

Unit – II

Software Measurements: Monitoring & measurement of SW development – cost, size and time metrics – methods and tools for metrics – issues of metrics in multiple projects.*

Unit – III

Software Quality: Quality in SW development – quality assurance – quality standards and certifications – the process and issues in obtaining certifications – the benefits and implications for the organization and its customers – change management.*

Unit – IV

Risk Issues : The risk issues in SW development and implementation – identification of risks – resolving and avoiding risks – tools and methods for identifying risk management.*

Unit – V

SPM Tools: Software project management using Primavera & Redmine and case study on SPM tools.*

*Programming assignments are mandatory.

Outcomes

- Ability to maintain software projects and monitor software project process
- Ability to design and develop project modules and assign resources
- Ability to comprehend, assess, and calculate the cost of risk involved in a project management

Text Books

1. *Richard H. Thayer, "Software Engineering Project Management", John Wiley & Sons, 2nd edition, 2001*
2. *Royce, Walker, "Software Project Management", Pearson Education, 2002*
3. *Kelker, S. A., "Software Project Management", Prentice Hall, 2003*

HONORS ELECTIVE

Course Code	:	**CSHO11**
Course Title	:	**Distributed Algorithms**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC29**
Course Type	:	**HONORS**

Objectives

- To understand the fundamental algorithms and protocols that are commonly used in distributed computing
- To learn the basics about synchronous and asynchronous models

Unit - I

Introduction, Synchronous Network Model, Leader election in a synchronous ring, Algorithms in general synchronous networks, Distributed consensus with link failures, Distributed consensus with process failures.*

Unit - II

Asynchronous system model, Asynchronous shared memory model, mutual exclusion, resource allocation, consensus and atomic objects.*

Unit - III

Asynchronous network model, basic asynchronous network algorithms and synchronizers.*

Unit - IV

Shared memory versus networks, logical time, global snapshots and stable properties, network resource allocation, partially synchronous system models.*

Unit - V

Fault Tolerance in distributed systems, Fault Tolerance in asynchronous systems, Fault Tolerance in asynchronous systems, failure detection – stabilization.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend distributed protocols and algorithms
- Ability to comprehend, develop, and analyze distributed algorithms for mission critical applications
- Ability to design and develop distributed algorithms for real world problems

Text Books

1. *Nancy A Lynch, "Distributed Algorithms", Morgan Kaufman Publishers 1996*
2. *Gerard Tel, "Introduction to Distributed Algorithms", Cambridge University Press, 2nd edition, 2000*

Course Code	:	**CSHO12**
Course Title	:	**High Speed Networks**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC27**
Course Type	:	**HONORS**

Objectives

- To understand up-to-date survey of developments in High SpeedNetworks
- To know how techniques involved to support real-time traffic andcongestion control
- To understand different levels of quality of service (QoS) to differentapplications

Unit – I

Frame Relay Networks – Asynchronous transfer mode – ATM Protocol Architecture,ATM logical Connection, ATM Cell – ATM Service Categories – AAL, High Speed LANs:Fast Ethernet, Gigabit Ethernet, Fiber Channel – Wireless LANs: applications,requirements – Architecture of 802.11*

Unit – II

Queuing Analysis- Queuing Models – Single Server Queues – Effects of Congestion Control – Traffic Management – Congestion Control in Packet SwitchingNetworks – Frame Relay Congestion Control*

Unit – III

TCP Flow control – TCP Congestion Control – Retransmission – Timer Management - Exponential RTO backoff – KARN's Algorithm – Window management – Performance ofTCP over ATM. Traffic and Congestion control in ATM – Requirements – Attributes –Traffic Management Frame work, Traffic Control – ABR traffic Management*

Unit – IV

Integrated Services Architecture – Approach, Components, Services- QueuingDiscipline, FQ, PS, BRFQ, GPS, WFQ – Random Early Detection, Differentiated Services*

Unit – V

RSVP – Goals & Characteristics, Data Flow, RSVP operations, Multiprotocol Label Switching – Operations, Label Stacking, Protocol details – RTP –Protocol Architecture*

*Programming Assignments are mandatory

Outcomes

- Ability to comprehend protocols for high speed networks
- Ability to analyze and compare the parameters of high speed networks and architectures
- Ability to design, develop, and analyze High speed network scenarios

Text Books

1. *William Stallings, "High Speed Networks and Internet", PearsonEducation, 2nd Edition, 2002*
2. *Warland, Pravin Varaiya, "High performance communication networks", 2nd Edition, Jean Harcourt Asia Pvt. Ltd., 2001*
3. *IrvanPepelnjk, Jim Guichard, Jeff Apcar, "MPLS and VPN architecture", CiscoPress, Volume 1 and 2, 2003*
4. *Abhijit S. Pandya and Ercan Sea, "ATM Technology for Broad Band TelecommunicationNetworks", CRC Press, New York, 2004*

Course Code	:	**CSHO13**
Course Title	:	**Software Defined Networking**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC27**
Course Type	:	**HONORS**

Objectives

- To know the reduced Complexity of Network Operation
- To understand the concepts of minimize Layer and maximize Network Resources
- To understand the Faster Time to Revenue for New Applications

Unit – I

Introduction, Control Plane, Data Plane, Distributed Control Planes, IP and MPLS, Creating the IP Underlay, Convergence Time, Load Balancing High Availability, Creating the MPLS Overlay, Replication, Centralized Control Planes – Logical Versus Litera, ATM/LANE, Route Servers, Wire Protocol, FAWG, Config and Extensibility, Architecture, Hybrid Approaches – Ships in the Night, Dual Function Switches.*

Unit – II

VMware, Nicira, Mininet, NOX/POX, Trema, Ryu, Big Switch Networks/Floodlight, Layer 3 Centric – L3VPN, Path Computation Element Server, Plexxi Affinity, Cisco OnePK, Management Interface, Network Divide, Modern Programmatic Interfaces, Modern Orchestration.*

Unit – III

Multitenant Data Center, Virtualized Multitenant Data Center, SDN Solutions for Data Center Network, VLANs, EVPN, VxLan, NVGRE, Virtualization and Data Plane I/O, Services Engineered Path, Service Locations and Chaining, NEV at ETSI, Non-ETSI NEV Work.*

Unit – IV

Network Topology, Traditional Methods, LLDP, BGP-TE/LS, ALTO, I2RS, Build Code First, The Juniper SDN Framework(s), Open Daylight Controller/Framework, Policy.*

Unit – V

Bandwidth Scheduling, Manipulation, Calendaring - Bandwidth Calendaring, Big Data and Application Hyper – Virtualization for Instant CSPF, Expanding Technology, Use Cases for Data Center Overlays, Big Data, Network Function Virtualization - Data Center Orchestration, Puppet, Network Function Virtualization, Optimized Big Data, - Firewall as Service, Network Access Control Replacement, Virtual Firewall, Feed Back and Optimization, Intrusion Detection/Threat Mitigation.*

*Programming Assignments are mandatory

Outcomes

- Ability to comprehend Software Defined Networks
- Ability to compare and analyze the advantages of SDN over traditional network
- Ability to design and implement software defined network

Textbooks

1. *Thomas D. Nandeau and Ken Gray, "Software Defined Networks", O' Reilly Media, Inc., First Edition, 2013*
2. *FEI HU , "Network Innovation through OpenFlow and SDN: Principles and Design", CRC Press, Taylor & Francis Group, 2014*

Course Code	:	**CSHO14**
Course Title	:	**Transaction Processing Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC26**
Course Type	:	**HONORS**

Objectives

- To knowhow in processing data generated by and about transactions that maintain high degree of accuracy and integrity
- To understand and recognize fraudulent transactions and produce timely user responses and reports

Unit – I

Consistency, Atomicity, Durability, Isolation, Flat Transactions, Providing Structure within a Transaction, Structuring an Application as Multiple Transactions.*

Unit – II

Schedules and Schedule Equivalence, Recoverability, Cascaded Aborts and Strictness, Models for Concurrency Control, A Strategy for Immediate-Update Pessimistic Concurrency Controls, Design of an Immediate-Update Pessimistic Concurrency Control, Objects and Semantic Commutativity, Atomicity, Recoverability and Compensating Operations, Locking and SQL Isolation Levels, Granular Locking: Intention Locks and Index Locks, Tuning Transactions, Multiversion Concurrency Controls.*

Unit – III

Crash, Abort and Media Failure, Immediate-Update Systems and Write-Ahead Logs, Recovery in Deferred-Update Systems, Recovery from Media Failure.*

Unit – IV

Transaction Processing in a Centralized System, Transaction Processing in a Distributed System, Global Atomicity and the Transaction Manager, Remote Procedure Call, Pear-to-Pear Communication, Event Communication, Storage Architectures, Transaction Processing on the Internet, Implementing the ACID Properties, Distributed Deadlock, Global Serialization.*

Unit – V

Authentication, Authorization and Encryption, Digital Signatures, Key Distribution and Authentication, Authorization, Authenticated RPC, Electronic Commerce, Certificates, Passport: SSO, SET Protocol: Dual Signatures, Goods Atomicity, Certified Delivery, and Escrow.*

*Programming assignments are mandatory.

Outcomes

- Ability to develop solutions that addresses all of the information processes
- Ability to design and develop techniques where information systems shall meet emerging needs
- Ability to analyze situations, identify needs, propose and develop solutions

Textbooks

1. *Michael Kifer, Arthur Bernstein and Philip M. Lewis, "Database Systems: An Application-Oriented Approach", Addison Wesley, 2006*
2. *Philip A. Bernstein and Eric Newcomer, "Principles of Transaction Processing", 2nd Edition, Morgan Kaufmann Publishers, Elsevier, 2009*

Course Code	:	**CSHO15**
Course Title	:	**Pervasive Computing**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC27**
Course Type	:	**HONORS**

Objectives

- To understand the characteristics and principles of Pervasive computing and solutions
- To design and implement pervasive application that are embedded into cars, airplanes, ships, bikes, posters, signboards, walls and even clothes

Unit – I

Introduction: Pervasive Computing- Principles, Characteristics- interaction transparency, context aware, automated experience capture, Vision and challenges of pervasive computing, Pervasive computing infrastructure- Architecture for pervasive computing- Pervasive devices- embedded controls.- smart sensors and actuators -Context communication and access services.*

UNIT – II

Technologies: Device Technology for Pervasive Computing: Hardware, Human-machine interfaces, Biometrics, Operating Systems, Java for pervasive devices- Voice Technology: Basics of Speech Recognition, Voice standards, Speech Applications, Speech and Pervasive Computing, Security- Personal Digital Assistants.*

UNIT – III

Sensor Networks and RFID: Introduction to Sensor networks – Sensor Node Architecture – Sensor Network Architecture - Types of sensor networks – Platforms for Wireless sensor networks – Applications of Wireless Sensor networks - Introduction to RFID – transponder and reader architecture - Types of tags and readers - Frequencies of operation – Application of RFID Technologies.*

UNIT – IV

Web based Applications: Web application concepts for pervasivecomputing: History, WWW architecture, Protocols, Trans-coding, Client Authentication via the Internet for pervasive computing, XML and its rôle in Pervsive computing, WAP and beyond: Introduction, Components of the WAP architecture, WAP infrastructure, WAP security issues, Wireless MarkupLanguage, WAP push, Products, i-Mode.*

UNIT – V

Programming And Applications: Server-side programming (Java) for pervasive computing: Java 2 Enterprise Edition (Overview), Servlets, Enterprise Java Beans, Java Server Pages, Extensible Markup Language, Web Services, Model-View-Controller pattern, Application Examples of Pervasive Computing: Retail, Airline Check-in and booking, Sales force automation, Healthcare, Tracking, Car Information Systems, Email Access via WAP and voice.*

*Programming assignments are mandatory.

Outcomes

- Ability to analyze and compare the performance of different data dissemination techniques
- Ability to develop solutions with comparisons for problems related to pervasive computing system through investigation
- Ability to design, analyze, and develop smart computing techniques

Text Books

1. *Jochen Burkhardt, Horst Henn, Stefan Hepper, Thomas Schaec& Klaus Rindtorff: "Pervasive Computing: Technology and Architecture of Mobile Internet Applications", Pearson Education, New Delhi, Sixth Edition, 2009*
2. *Seng Loke, "Context-Aware Computing Pervasive Systems", Auerbach Pub., Taylor and Francis Group, New York, 2007*
3. *Rahul Banerjee, "Lecture Notes in Pervasive Computing", Outline Notes, BITS-Pilani, 2012.*
4. *Genco, S. Sorce, "Pervasive Systems and Ubiquitous Computing", WIT Press, 2012*
5. *Guruduth S. Banavar, Norman H. Cohen, and Chandra Narayanaswami, "Pervasive Computing: An Application-Based Approach", Wiley Interscience, 2012*
6. *Frank Adelstein, S K S Gupta, G G Richard, and L Schwiebert, "Fundamentals of Mobile and Pervasive Computing", Tata McGraw-Hill, New Delhi, 2005*
7. *StefenPoslad, "Ubiquitous Computing: Smart Devices, Environments and Interactions", Wiley, Second Edition, 2010*

Course Code	:	**CSHO16**
Course Title	:	**Programming for Multi-Core Systems**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC36**
Course Type	:	**HONORS**

Objectives

- To understand the fundamentals of multi-core architecture
- To be able to know the basic concepts of multi core programming using threads
- To be able to understand various programming constructs in multi-core architecture

Unit – I

Fundamentals of Multi core : Fundamentals of Quantitative Design and Analysis-Dependability-Measuring, Reporting and Summarizing Performance-Quantitative principles of computer Design, Instruction Level Parallelism-Data level and Thread level Parallelism. Multi core Architecture-Motivation for Concurrency -Parallel Computing in Micro processors-Gustafson's law.*

Unit – II

Introduction to Threads: Defining threads-System View of threads-Threading above the OS-Inside the OS-Threads inside the Hardware-What happened When a thread is created-Application Programming models and threading-VMs and Platforms-Run time Virtualization, System Virtualization. *

Unit –III

Thread Programming Types and APIs: Synchronization-Critical Section-Deadlock-Synchronization Primitives-Semaphores-Locks-Condition Variables-Flow Control based Concepts-Implementation based Threading Features-Threading APIs for Microsoft Windows-Threading API for .NET framework,POSIX Threads-Programming WithPthreads,OpenMP-Challenges in threading a loop-Minimizing threading overhead-Performance oriented programming-JavaThreads.*

Unit – IV

Thread Handling and Debugging: Too many threads-Data Races, Deadlock and Live locks-Heavily Contended Locks-Non-blocking algorithms-Thread safe functions and libraries-Memory Issues –Cache Related Issues-Avoiding Pipeline Stalls in IA-32-Data Organization for High Performance, Multithreaded Debugging Techniques: General Debugging Techniques.*

Unit – V

Implementation of the Programming Constructs: Foundations of Shared Memory, Spin Locks and Contention-Monitors and Blocking Synchronization- Concurrent Queues and the ABA Problem- Concurrent Stacks and Elimination-Counting, Sorting, and Distributed Coordination Concurrent Hashing and Natural Parallelism- Skip lists and Balanced Search-Futures, Scheduling, and Work Distribution- Barriers-Transactional Memory - Software Transactional Memory-hardware Transactional Memory –Threading on Intel Multicore Processors.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend the programming constructs of multi-core systems
- Ability to exploit the benefit of parallel programming
- Ability to design and develop APIs for Multithreaded Applications

Text Books

1. *Shameem Akhter and Jason Roberts, "Multi-Core Programming: Increasing Performance through Software Multi Threading", Intel Press, 2006*
2. *Maurice Herlihy and NirShavit, "The Art of Multiprocessor Programming",Revised First Edition, Elsevier Publication, 2012*

Reference Books

1. *John L. Hennesy, and David E. Patterson, "Computer Architecture: A Quantitative Approach", 5th Edition, Elsevier Publication, 2012*
2. *Thomas Rauber and GudulaRünger, "Parallel Programming: for Multi-core and Cluster Systems" , 2nd Edition, Springer Publication, 2010*

Course Code	:	**CSHO17**
Course Title	:	**Soft Computing**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC25**
Course Type	:	**HONORS**

Objectives

- To understand the concepts of feed forward &feedback neural networks
- To understand the concept of fuzziness involved in various systems
- To expose the ideas about genetic algorithm
- To provide adequate knowledge about of FLC and NN toolbox

Unit – I

Introduction of soft computing - soft computing vs. hard computing- various types of soft computing techniques- applications of soft computing-Neuron- Nerve structure and synapse-Artificial Neuron and its model- activation functions- Neural network architecture- single layer and multilayer feed forward networks- McCullochPitts neuron model- perceptron model- MLP-back propagation learning methods- effect of learning rule coefficient.*

Unit – II

Counter propagation network- architecture- functioning & characteristics of counter-Propagation network-Hopfield/ Recurrent network- configuration- stability constraints-associative memory- and characteristics- limitations and applications- Hopfield v/s Boltzman machine- Adaptive Resonance Theory- Architecture- classifications-Implementation and training-Associative Memory.*

Unit – III

Different faces of imprecision - inexactness, Ambiguity, Undecidability, Fuzziness and certainty, Fuzzy sets and crisp sets. Intersections of Fuzzy sets, Union of Fuzzy sets, the complement of Fuzzy sets - Fuzzy reasoning. Linguistic variables, Fuzzy propositions, Fuzzy compositional rules of inference - Methods of decompositions and defuzzification.*

Unit – IV

Basic concept of Genetic algorithm and detail algorithmic steps-adjustment of free Parameters-Solution of typical control problems using genetic algorithm- Concept on some other search techniques like tabu search and ant colony search techniques for solving optimization problems.*

Unit – V

GA application to optimization problems- Case studies: Identification and control of linear and nonlinear dynamic systems using MATLAB-Neural Network toolbox. Stability analysis of Neural Network interconnection systems- Implementation of fuzzy logic controller using MATLAB fuzzy logic toolbox-Stability analysis of fuzzy control systems.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend machine learning and soft computing techniques in solving real world applications
- Ability to design and develop ML techniques with assistance of MATLAB
- Ability to visualize and analyze behavioural pattern to develop evolutionary algorithm

Text Books

1. *Timothy J. Ross, "Fuzzy Logic with Engineering Applications", Wiley India, 3rd edition, 2012*
2. *Zimmermann H. J. "Fuzzy set theory and its Applications" Springer international edition, 2011*
3. *David E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Pearson Education, 2009*
4. *Laurene V. Fausett, "Fundamentals of Neural Networks: Architectures, Algorithms, And Applications", Pearson Education, 1st edition, 1993*
5. *W. T. Miller, R. S. Sutton and P. J. Webros, "Neural Networks for Control", MIT Press, 1996*

Course Code	:	**CSHO18**
Course Title	:	**Digital System Testing and Verification**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC22**
Course Type	:	**HONORS**

Objectives

- To design the Models at various levels and detects the faults in modeling
- To learn the testability techniques and to learn the Verilog for building the systems
- To test and verify the validity of the Model

Unit I

Modeling : Basic Concepts, Functional Modeling at the Logic Level, Functional Modeling at the Register Level, Structural Models, Level of Modeling, Logic Simulation: Problems in simulation-Based Design Verification, Types of Simulation, The Unknown Logic Value, Compiled Simulation, Event Driven Simulation, Delay Models, Element Evaluation, Hazard Detection, Gate Level Event-Driven simulation, Other Logic Values, Other Delay Models.*

Unit II

Fault Modeling : Logical Fault Models, Fault Detection and redundancy, Fault Equivalent and Fault Location, Fault Dominance, The single Stuck-Fault Model, The Multiple Stuck Fault-Model, Stuck RTL Variable, Fault Variables. Fault Simulation : General Fault Simulation Techniques, Fault Simulation for Combinational Circuits, Fault Sampling, Statistical Fault analysis.*

Unit III

Testing for Bridging Fault: The Bridging Fault Model, Detection of Non-feedback Bridging Faults, Detection of Feedback Bridging Faults, Bridging Fault Simulation, Test Generation for Bridging Faults. Functional Testing : Functional Testing without Fault Models, Exhaustive and Pseudo exhaustive Testing, Functional Testing with Specific Fault Models.*

Unit IV

Design for Testability : Testability, Ad Hoc Design for Testability Techniques, Controllability and Observability by means of Scan Registers, Generic Scan-Based Designs, Storage cells for Scan designs, Classical scan designs, Scan Design Costs, Board level and system level DFT Approaches, Advanced scan concepts, Boundary Scan Standards.*

Unit V

Basics of Test and Role of HDLs : Design and Test, Test Concerns, HDLs in Digital System Test. Verilog HL for Design and Test : HDL for developing test methods, Using verilog in design, Using verilog in test, Basic structures of verilog, Combinational Circuits, Sequential circuits. Fault and detection modeling using verilog.*

*Programming assignments are mandatory.

Outcomes

- Ability to design the modeling of systems
- Ability to write the test bench to test the validity of the model
- Ability to write Verilog code to built the systems

Text Books

1. *ZainalabedinNavabi, "Digital System Test an Testable Desin using HDL Models and Architectures", Springer publications, 2010.*
2. *MironAbramovici, Melvin A. Breuer, and Arthur D. Friedman, "Digital Systems Testing and Testable Design", Wiley publications1990.*

Reference Book

1. *Jha, Niraj K., and Sandeep Gupta, "Testing of digital systems", Cambridge University Press, 2003.*

Course Code	:	**CSHO19**
Course Title	:	**CAD for VLSI**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC22**
Course Type	:	**HONORS**

Objectives

- To provide experience designing integrated circuits using Computer Aided Design (CAD) Tools
- To introduce the concepts and techniques of modern integrated circuit design and testing (CMOS VLSI)
- To understand the programming paradigms of Hardware Description language (HDL)

Unit I

Introduction to CAD tools, Evolution of Design Automation, Basic Transistor Fundamentals, CMOS realizations of basic gates.*

Unit II

Modelling techniques, Types of CAD tools and Introduction to logic simulation*

Unit III

Verilog: Syntax, Hierarchical modelling and Delay modelling, Verilog constructs, Memory modelling.*

Unit IV

Logic Synthesis: Introduction synthesis of different Verilog constructs.*

Unit V

Introduction to Reconfigurable computing, FPGAs, the AltraQuartus II flow.*

*Programming assignments are mandatory.

Outcomes

- Ability to acquire hands-on skills of using CAD tools in VLSI design
- Ability to develop coding skill set using HDL
- Ability to design and develop VLSI project having a set of objective criteria and design constraints

Text Books

1. *Samir Palnitkar, "Verilog HDL", 2nd Edition, Pearson Education, 2004.*
2. *J. Bhaskar, "Verilog HDL Synthesis", BS publications, 2001*

Course Code	:	**CSHO20**
Course Title	:	**Middleware Technologies**
Number of Credits	:	**3-0-0-3**
Prerequisites(Course code)	:	**CSPC32**
Course Type	:	**HONORS**

Objectives

- To understand the essence of client-server and middleware architectures
- To learn the basics of CORBA and C#.NET technologies

Unit – I

Introduction to client server computing-client server models, Benefits of client server computing, pitfalls of client server programming, Middleware − Client / server building blocks, RPC, RMI.*

Unit – II

Middleware – Objects, Elements, Architecture, Middleware distributed applications, middleware types, transaction oriented middleware.*

Unit – III

CORBA with Java - Client/Server CORBA-style, CORBA with Java, Static CORBA, ORBlets with Applets, Dynamic, CORBA Beans, CORBA initialization protocol, CORBA activation services, CORBA java- to- IDL mapping.*

Unit – IV

EJBs and CORBA - Object transaction monitors CORBA OTM's, EJB and CORBA OTM's, EJB container frame work, Session and Entity Beans, EJB client/server development Process The EJB container protocol, support for transaction EJB packaging EJB design Guidelines.*

Unit – V

C# and .NET Platform- .NET Assemblies, Object Oriented Programming with C#, Callback Interfaces, Delegates, and Events, Type Reflection, Late Binding, and Attribute-Based Programming, Object Serialization and the .NET Remoting Layer.*

*Programming assignments are mandatory.

Outcomes

- Ability to comprehend of Middleware tools
- Ability to build real time applications based on .Net and C#
- Ability to design, develop, and analyze middleware architecture in developing enterprise technologies

Text Books

1. *Andrew Troelsen, "C# and the .NET Platform", Apress Wiley-Dreamtech, India Pvt Ltd, 2nd Edition, 2007*
2. *Chris Britton, "IT Architectures and Middleware: Strategies for Building Large, Integrated Systems", Pearson Education, 2nd Edition , 2004*
3. *Robert Orfali, Dan Harkey, and Jeri Edwards, "The Essential Client/Server Survival Guide", John Wiley & Sons , 3rd Edition, 1999*
4. *Robert Orfali and Dan Harkey, "Client/Server programming with Java and CORBA", John Wiley & Sons , SPD 2nd Edition, 1998*
5. *Jesse Liberty, "Programming C#", 2nd Edition, O'Reilly Press, 2002*