FOURTH SEMESTER

CODE	COURSE TITLE	L	T	P	C
CS202	Computer Graphics	3	0	0	3
CS204	Digital Systems Design	3	0	0	3
CS206	Logical Foundations of Computer Science	3	1	0	4
CS208	Computer Networks	3	0	0	3
CS210	Automata and Formal Languages	3	1	0	4
MA202	Introduction to Probability Theory	3	0	0	3
CS214	Digital Systems Design Laboratory	0	0	3	2
CS218	Computer Networks Laboratory	0	0	3	2
TOTAL CREDITS **24**					

FOURTH SEMESTER

CS202 - COMPUTER GRAPHICS

Credits: 3

Objectives

- To understand basic algorithms for computer graphics
- To understand various applications of graphics

Unit-I 2D Primitives
Output primitives: Line - Circle and Ellipse drawing algorithms - Attributes - Two dimensional Geometric transformations - Two dimensional viewing - Line - Polygon - Curve and Text clipping algorithms.

Unit-II 3D Concepts
Parallel and Perspective projections: Three dimensional object representation – Polygons, Curved lines, Splines, Quadric Surfaces- Visualization of data sets - 3D transformations -Viewing -Visible surface identification - Hidden surface and line elimination.

Unit-III Graphics Programming
Color Models: RGB – YIQ – CMY -HSV – Animations – General Computer Animation, Raster and Key frame - Graphics programming using OPENGL – Basic graphics primitives – Drawing three dimensional objects and scenes.

Unit-IV Rendering
Introduction to Shading models – Flat and Smooth shading – Adding texture to faces – Adding shadows of objects - Creating shaded objects – Rendering texture – Drawing Shadows- realistic image synthesis techniques, Textures and image-based rendering; Video games and Computer animations.

Unit-V Fractals
Fractals and Self similarity – Peano curves – Creating image by iterated functions – Mandelbrot sets – Julia Sets – Random Fractals – Overview of Ray Tracing – Intersecting rays with other primitives – Adding Surface texture – Reflections and Transparency – Boolean operations on Objects.

Outcomes

- Ability to create software tools for Games and Animation
- Knowledge of Computer Graphics Techniques

Teaching and Evaluation guidelines

- 40% on Application (Higher Order Thinking), and 40% on Methods and Techniques (Medium Order Thinking), and 20% on Tool functions (Lower Order Thinking).

Text Books

1. Donald Hearn, Pauline Baker, "Computer Graphics – C Version", 2nd edition, Pearson Education, 2004.
2. F.S. Hill," Computer Graphics using OPENGL", 2nd edition, Pearson Education, 2003.

Reference Book

1. James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics-Principles and practice, Second Edition in C, Pearson Education, 2007.

CS204 - DIGITAL SYSTEMS DESIGN

Credits: 3

Objective

- To understand the architecture of basic building blocks, logic gates, adders, multipliers, shifters and other digital devices
- To understand the logic of minimization techniques including Karnaugh Maps
- To understand the structure of field programmable logic circuits FPGAs
- To analyze design of combinational logic, sequential circuits, PLA, PAL

Unit-I Introduction to VLSI design

Basic gate design - Digital VLSI design - Design of general Boolean circuits using CMOS gates - FPGA.

Unit-II Verilog Concepts

Basic concepts – Modules & ports & Functions – useful modelling techniques – Timing and delays – user defined primitives.

Unit-III Modelling Techniques

Gate level modelling – Dataflow modelling – Physical modelling – Structural / Data flow modelling – Switch level modelling.

Unit-IV Advanced Verilog Concepts

Synthesis concepts– Inferring latches and flip-flops – Modelling techniques for efficient circuit design.

Unit-V VHDL

Generics and Configurations – Subprograms and Overloading – Packages and Libraries – Advanced features – simulation semantics – modelling examples – state machine modelling using VHDL.

Outcomes

- Ability to design basic digital circuits and systems.
- Ability to understand and use high-level hardware description languages such as VHDL and Verilog to design combinational or sequential circuits

Teaching and Evaluation guidelines

- 30% Verilog coding for IC's (Higher Order Thinking), and 50% on Structural design (Medium Order Thinking), and 20% on Basic gate design (Lower Order Thinking).

Text Book

1. Samir Palnitkar, "Verilog HDL Synthesis", BS Publications, Second Edition, 2008

Reference Books

1. Bhaskar, "Verilog HDL Synthesis", BS Publications, Third Edition, 2008.

CS206-LOGICAL FOUNDATIONS OF COMPUTER SCIENCE

Credits: 4

Objectives

- To study about the notions, mechanisms, and properties of weakest preconditions
- To learn how to create a strong guarded commands and its related theorems
- To learn the basics of propositional logic and its conversions
- To understand the principles and proofs of predicate calculus.

Unit-I Review of Prepositional Calculus

Validity - Satisfiability related concepts - CNF and DNF forms - Conversion of arbitrary prepositional formula to CNF or DNF.

Unit-II Compactness idea

Resolution principle and proof of the theorem - Review of predicate calculus - Interpretation of formulae in predicate calculus.

Unit-III Prenex normal form and examples

Application of logic in programming - Proof rules for structured statements (assignment, while, repeat-until, for statements).

Unit-IV Pre-conditions / Post-conditions

Weakest precondition - Notion of machine - Mechanism and Wp as a predicate transformer - Properties of Wp.

Unit-V Guarded Commands

General form of **if** command - Wp of **if** - Related theorem - General form of **do** command - Wp of **do** - Need for strong guards.

Outcomes

- Ability to define and convert the prepositional formula.
- Knowledge of predicate calculus and its application in programming.
- Ability to identify the related theorems and proofs of predicate calculus.

Text Books

1. D.Gries, "The Science of Programming", Narosa, 1981
2. S.Alagic, M.A.Arbib, "The Design of Well-Structured and Correct Programs", SpringerVerlagn, 1978
3. E.W.Djikstra, "A Discipline of Programming", Prentice Hall, Englewood Cliffs, 1976

CS208 - COMPUTER NETWORKS

Credits: 3

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 layer in OSI and TCP/IP
- To know the basic concepts of network security and its various security issues related with each layer

Unit-I Introduction

Network architecture – layers – Physical links – Channel access on links – Hybrid multiple access techniques - Issues in the data link layer - Framing – Error correction and detection – Link-level Flow Control.

Unit-II Medium Access Control (MAC) Layer

Medium access: Carrier Sense Multiple Access (CSMA) - Ethernet – Token ring (IEEE 802.5) – Fiber Distributed Data Interface (FDDI) – Wireless Local Area Network (WLAN) – Bridges and Switches.

Unit-III Network Layer

Circuit switching vs. Packet switching/ Packet switched networks. Protocols: Internet Protocol (IP) – Address Resolution Protocol (ARP) – Reverse Address Resolution Protocol (RARP) –Dynamic Host Configuration Protocol (DHCP) – Internet Control Message Protocol (ICMP). Queuing discipline – Routing algorithms: Routing Information Protocol (RIP) – Open Shortest Path First (OSPF). Subnet creation - Interdomain routing: BGP – IPv6 – Multicasting.

Unit-IV Transport Layer

User Datagram Protocol (UDP) – Transmission Control Protocol (TCP) – Adaptive Flow Control – Adaptive Retransmission - Congestion control –Congestion avoidance – Quality of Service (QoS).

Unit-V Application Layer

Protocols: Hyper Text Transfer Protocol (HTTP) – Domain Name System (DNS) – (Simple Network Management Protocol (SNMP) – File Transfer Protocol (FTP). E-mail related protocols: Simple Mail Transfer Protocol (SMTP), Multipurpose Internet Mail Extensions (MIME), Post Office Protocol (POP3). Security – Pretty Group Privacy (PGP) – Secure Socket Shell (SSH).

Outcomes

- Obtain insight about basic network theory and layered communication architectures
- Provide solutions to various problems in network theory

Teaching and evaluation guidelines

- 40% on Problems, 40 % on Comparisons and Statements, 20% on Techniques and Definitions

Text Books

1. Larry L. Peterson, Bruce S. Davie, "Computer Networks: A Systems Approach", Fifth Edition, Morgan Kaufmann Publishers Inc., 2012.
2. Andrew S. Tanenbaum and David J. Wetherall, "Computer Networks", Fifth Edition, 2010.
3. William Stallings, "Data and Computer Communication", Eighth Edition, Pearson Education, 2007.

Reference Books

1. James F. Kuross, Keith W. Ross, "Computer Networking, A Top-Down Approach Featuring the Internet", Third Edition, Addison Wesley, 2004.
2. Nader F. Mir, "Computer and Communication Networks", Pearson Education, 2007.
3. Comer, "Computer Networks and Internets with Internet Applications", Fourth Edition, Pearson Education, 2003.

CS210 - AUTOMATA AND FORMAL LANGUAGES

Credits: 4

Objectives

- To understand the significance of automata theory in computer science and use the conceptual outcomes and theorems for better algorithms and implementation

Unit-I Finite Automata (FA)

Deterministic, non-deterministic and equivalence - Equivalence of regular expressions and FA - Moore and Mealy machines.

Unit-II Regular Languages

Pumping lemma of regular sets - Myhill Nerode theorem - Minimization of finite automata - Chomsky hierarchy of languages.

Unit-III Context-Free Language (CFL)

Context-free grammar - Derivation trees - Ambiguity simplification - Normal forms - UVWXY theorem - Applications.

Unit-IV Pushdown Automata (PDA)

Definitions - Context free languages - Construction of PDA for simple CFLs - Linear bounded automata.

Unit-V Turing Machines

Universal Turing Machines - Types of Turing Machines - Techniques - Halting problem - Stack automata - Definitions.

Outcomes

- To be able to understand the limitations of algorithm and design optimally.

Teaching and Evaluation Guidelines

- 40% on Analysis and Application (Higher Order Thinking), and 60% on Conceptual understanding and Definitions (Lower Order Thinking).

Text Books

1. J.E.Hopcroft And J.D.Ullman, "Introduction to Automata Theory", Languages and Computation, Pearson Education, 2001
2. Peter Linz, "An Introduction to Formal Language and Automata", Narosa Pub. House, Reprint 2000

MA202 – INTRODUCTION TO PROBABILITY THEORY

Credits: 3

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 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 - Binomial 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/1 Queues with finite and infinite waiting space.

Outcomes

- Conceptualize the necessity of randomness concept in practical situation
- Approximate the real problems using stochastic process and deduce results
- 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. 1968.
2. K. S. TRIVEDI, Probability and Statistics with Reliability and Queuing and Computer Science Applications, Prentice Hall of India, 1988

Reference Book

1. O. ALLEN, Introduction to Probability, Statistics and Queueing Theory with Computer Science Applications, Academic Press, 2006 reprint.
2. PAPOULIS, Probability, Random Variables and Stochastic Processes, McGraw Hill.

CS214 - DIGITAL SYSTEMS DESIGN LABORATORY

Credits: 2

Objective

- 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

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

Outcomes

- Ability to design synchronous sequential circuits using basic flip-flops, counters, PLA, PAL
- Ability with the necessary software skills to design basic digital systems
- Ability to expertise in debugging the digital circuits

CS218 - COMPUTER NETWORKS LABORATORY

Credits: 2

Objectives

- To understand the network topologies
- To understand the socket communication and routing protocols
- To study the behavior of TCP and UDP

EXPERIMENTS

1. Study of different types of Network cables and Practically implement the cross-wired
   - cable and straight through cable using clamping tool
2. Study of basic Network commands and Network configuration commands
3. Client Server Program using TCP sockets
   - Date and Time Server
   - Chat application
4. Simulation of Sliding Window Protocol
5. Programs using raw sockets
   - Packet capturing and packet filtering
6. Client Server Program using UDP
   - DNS Implementation
   - Chat application
7. Implementation of routing protocols
   - OSPF, BGP

Outcomes

- To implement client server based communication using TCP and UDP
- To implement the routing protocols