Bachelor of Technology (Computer Science & Engineering)
Scheme of Studies/Examination
Semester V

S. No.	Course No.	Subject	L:T:P	Hours/ Week	Examination Schedule (Marks)				Duration of Exam (Hrs)
					Major Test	Minor Test	Practical	Total	
1	CSE 301N	Automata Theory	3:1:0	4	75	25	0	100	3
2	CSE 303 N	Computer Networks	3:1:0	4	75	25	0	100	3
3	CSE 305N	Design and Analysis of algorithms	3:1:0	4	75	25	0	100	3
4	CSE 307N	Computer organisation and Architecture	3:1:0	4	75	25	0	100	3
5	CSE 309N	Simulation & Modelling	3:1:0	4	75	25	0	100	3
6	CSE 311N	Computer Networks Lab	0:0:3	3	0	40	60	100	3
7	CSE 313N	Design and Analysis of algorithms Lab	0:0:3	3	0	40	60	100	3
8	CSE 315 N	Simulation Lab	0:0:3	3	0	40	60	100	3
9	CSE 317N	Seminar/Industrial Training*	0:0:2	2	0	40	60	100	
10	CSE 319N	Technical Communication and Soft Skills Lab	0:0:2	2	0	100	0	100	3
		Total		**33**	**375**	**385**	**240**	**1000**	

CSE-301N	Automata Theory					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3 Hrs.
Purpose	To understand the challenges for Theoretical Computer Science and its contribution to other sciences					
Course Outcomes(CO)						
CO1	Students are able to explain and manipulate the different fundamental concepts in automata theory and formal languages.					
CO2	Simplify automata and context-free grammars, Prove properties of languages, grammars and automata with rigorously formal mathematical methods, minimization.					
CO3	Differentiate and manipulate formal descriptions of push down automata, its applications and transducer machines.					
CO4	To understand basic properties of Turing machines and computing with Turing machine, the concepts of tractability and decidability.					

Unit - I

Introduction to Automata: Study and Central Concepts of Automata Theory, Applications of Finite Automata, An Introduction of Deterministic Finite Automata(DFA) and Non-Deterministic Finite Automata(NFA), Finite Automata with Epsilon (€) Transitions.
Regular Expression and Languages:-Regular Expressions (RE), Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws of Regular Expressions. Closure Properties of Regular Languages, RE to NFA, DFA Conversion and DFA to RE, Equivalence and Minimization of NFA and DFA automata.

Unit-2

Context free Grammars and Languages: Parse Trees, Context Sensitive Grammar, Applications of Context Free Grammars, Regular Grammar, Ambiguity in Grammars and Languages. Normal forms of context free grammars, Subfamilies of Context Free Languages (CFL), Closure Properties of CFL, Chomsky Theorem, Chomsky Hierarchy, Chomsky Normal Form, Greibach Normal Form.
Pumping Lemma:-Introduction to Pumping Lemma, pumping lemma for context free languages, Applications of Pumping Lemma, Minimization of Finite Automata, and Recursive Language.

Unit-3

Mealey and Moore Machines:- Definitions, Representation, Equivalence of Moore and Mealey Machines and its Designing.
Push Down Automata: Introduction of Push Down Automata (PDA), Language of PDA, Equivalence of PDA's and CFG's, Deterministic Push Down Automata, Designing of PDA, Applications of PDA. Parikh Theorem and Parikh Mapping, Kleene's Theorem.

Unit-4

Introduction to Turing Machine: The Turing Machine, Programming Techniques for Turing Machine, Extensions of Turing Machine, Restricted Turing Machines, Universal Turing Machines and Designing of Turing Machines, Time and Tape Complexity Measures of Turing machines
Decidability: Post's Correspondence Problem (PCP), Rice's Theorem, Decidability of Membership, Emptiness and Equivalence Problems of Languages.

Textbooks
1. J.E.Hopcroft, R.Motwani and J.D.Ullman , "Introduction to Automata Theory Languages and computation", Pearson Education Asia , 2001.
2. K.Krithivasan and R.Rama; Introduction to Formal Languages, Automata Theory and Computation; Pearson Education, 2009.

References
1. Peter Linz, "An Introduction to Formal Language and Automata", 4th Edition, Narosa Publishing house , 2006.
2. M.Sipser; Introduction to the Theory of Computation; Singapore: Brooks/Cole, Thomson Learning, 1997.
3. John.C.martin, "Introduction to the Languages and the Theory of Computation",Third edition, Tata McGrawHill, 2003.

CSE-303N	Computer Networks					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3 Hrs.
Purpose	To introduce the architecture and layers of computer network, protocols used at different layers.					
Course Outcomes(CO)						
CO1	To understand the basic concept of networking, types, networking topologies and layered architecture.					
CO2	To understand data link layer and MAC sub-layer`					
CO3	To understand the network Layer functioning					
CO4	To understand the transport layer and application layer operation					

Unit -1
Introduction: introduction to Computer Networks, Data Communication System and its components, Data Flow, Computer network and its goals, Types of computer networks: LAN, MAN, WAN, Wireless and Wired networks, broadcast and point-to-point networks, Network topologies, protocols, interfaces and services, ISO-OSI reference model, TCP/IP architecture.
Physical Layer: Concept of Analog & Digital Signal, Bandwidth, Transmission Impairments: Attenuation, Distortion, Noise, Multiplexing : Frequency Division, Time Division, Wavelength Division, Introduction to **Transmission Media** : Twisted pair, Coaxial cable, Fiber optics, Wireless transmission (radio, microwave, infrared), Switching: Circuit Switching, Message Switching ,Packet Switching & comparisons, narrowband ISDN, broadband ISDN and ATM.

Unit -2
Data link layer: Error Control, Types of errors, framing(character and bit stuffing), error detection & correction methods; Flow control; Protocols: Stop & wait ARQ, Go-Back- N ARQ, sliding window protocols, Selective repeat ARQ, HDLC
Medium access sub layer: Point to point protocol, FDDI, token bus, token ring; Reservation, polling, Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, FDMA, TDMA, CDMA, LLC, Traditional Ethernet, fast Ethernet, Network devices-repeaters, hubs, switches, Bridges, Router, Gateway

Unit-3
Network layer: Addressing : Internet address, subnetting; Routing techniques, static vs. dynamic routing , routing table, DHCP, IEEE standards 802.x, Routing algorithms: shortest path algorithm, flooding, distance vector routing, link state routing; Protocols: ARP, RARP, IP, ICMP, IGMP, IPV6; Unicast and multicast routing protocols.

Unit-4
Transport layer: Process to process delivery; UDP; TCP, RPC, Congestion control algorithm: Leaky bucket algorithm, Token bucket algorithm, choke packets; Quality of service: techniques to improve QoS.
Application layer: DNS; SMTP, SNMP, FTP, HTTP & WWW; Firewalls, Bluetooth, Email, S/MIME, IMAP, **Security**: Cryptography, user authentication, security protocols in internet, public key encryption algorithm, digital signatures.

TEXT BOOK
1. Behrouz A. Forouzan, "Data communication and Networking", Tata McGraw Hill, Fourth Edition, 2011.
2. Computer Networks, 4th Edition, Pearson Education by Andrew S. Tanenbaum
REFERENCES
1. Larry L.Peterson, Peter S. Davie, "Computer Networks", Elsevier, Fifth Edition, 2012.
2. William Stallings, "Data and Computer Communication", Eighth Edition, Pearson Education, 2007.
3. James F. Kurose, Keith W. Ross, "Computer Networking: A Top–Down Approach Featuring the Internet", Pearson Education, 2005.

CSE-305N	Design and Analysis of Algorithms					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3 Hrs.
Purpose	To introduce advanced data structures & algorithms concepts involving their implementation for solving complex applications.					
Course Outcomes (CO)						
CO1	Learn the basic concepts of data structures and their analysis.					
CO2	Study the concept of dynamic programming and various advanced data structures.					
CO3	Learn various graph algorithms and concepts of computational complexities.					
CO4	Study various Flow and Sorting Networks					

Unit 1

Introduction
Review: Elementary Data Structures, Algorithms & its complexity(Time & Space), Analysing Algorithms, Asymptotic Notations, Priority Queue, Quick Sort and merge sort.
Recurrence relation: Methods for solving recurrence(Substitution , Recursion tree, Master theorem), Strassen multiplication.
Advanced data Structures: Binomial heaps, Fibonacci heaps, Splay Trees, Red-Black Trees.

Unit 2

Advanced Design and analysis Techniques
Dynamic programming: Elements, Matrix-chain multiplication, longest common subsequence,
Greedy algorithms: Elements , Activity- Selection problem, Huffman codes, Task scheduling problem, Travelling Salesman Problem.
Backtracking algorithms: Graph coloring, N-Queen problem, Hamiltonian path and circuit.

Unit 3

Graph Algorithms
Review of graph algorithms:Traversal Methods(Depth first & Breadth first search),Topological sort, Strongly connected components, Minimum spanning trees- Kruskal's and Prim's Algorithm, Single source shortest paths, Relaxation, Dijkstra's Algorithm, Bellman- Ford algorithm, Single source shortest paths for directed acyclic graphs, Floyd-Warshall algorithm.

Unit 4

Computational Complexity: Basic Concepts, Polynomial vs Non-Polynomial Complexity, NP-hard & NP-complete classes. Flow and Sorting Networks, Flow networks, Ford- Fulkerson method, Maximum bipartite matching, Sorting Networks, Comparison network, Zero- one principle, Bitonic sorting network, merging network

Text Books:
1. Corman, Leiserson and Rivest : Introduction to Algorithms, 2/e, PHI
2. Harsh Bhaisn, Algorithms: Design And Analysis Oxford University Press,2015.

Reference Books:
1. Aho, Hopcroft and Ullman : The Design and Analyses of Computer Algorithms. Addison Wesley.
2. R.B.Patel, Expert Data Structures with C, Khanna Publications , Delhi, India, 2ndEdition 2004, ISBN 81-87325-07-0, pp.1-909.
3. R.B.Patel & M.M.S Rauthan, Expert Data Structures with C++, Khana Publications, Delhi , India, 2ndEdition 2004,ISBN : 87522-03-8.
4. Horowitz, Ellis and Sahni, Sartaj : Fundamentals of Computer Algorithms, Galgotia Publications

CSE-307N	Computer Organization and Architecture					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3 Hrs.
Purpose	Student will be able to understand the basic concepts of computer architecture and organization, and understand the key skills of constructing cost-effective computer systems.					
Course Outcomes (CO)						
CO1	Be familiar with the functional units of the processor such as the register file and arithmetic-logical unit, and with the basics of systems topics					
CO2	Be familiar with the design trade-offs in designing and constructing a computer processor.					
CO3	Be familiar with the CPU design including the RISC/CISC architectures.					
CO4	Be familiar with the basic knowledge of I/O devices and interfacing of I/O devices with computer.					

Unit- I

Data representation and Computer arithmetic: Introduction to Computer Systems, Organization and architecture, evolution and computer generations; Fixed point representation of numbers, digital arithmetic algorithms for Addition, Subtraction, Multiplication using Booth's algorithm and Division using restoring and non restoring algorithms. Floating point representation with IEEE standards and its arithmetic operations.

Unit-II

Basic Computer organization and Design: Instruction codes, stored program organization, computer registers and common bus system, computer instructions, timing and control, instruction cycle: Fetch and Decode, Register reference instructions; Memory reference instructions. Input, output and Interrupt: configuration, instructions, Program interrupt, Interrupt cycle, Micro programmed Control organization, address sequencing, micro instruction format and microprogram sequencer.

Unit-III

Central Processing Unit: General register organization, stack organization, instruction formats, addressing modes, Data transfer and manipulation, Program control. CISC and RISC: features and comparison. Pipeline and vector Processing , Parallel Processing, Pipelining, Instruction Pipeline, Basics of vector processing and Array Processors.

Unit-IV

Input-output organization: I/O interface. I/O Bus and interface modules, I/O versus Memory Bus. Asynchronous data transfer: Strobe control, Handshaking, Asynchronous serial transfer. Modes of Transfer: Programmed I/O, Interrupt driven I/O, Priority interrupt; Daisy chaining, Parallel Priority interrupt. Direct memory Access, DMA controller and transfer. Input output Processor , CPU-IOP communication, I/O channel.

TEXT BOOK:
1. William Stallings, "Computer Organization and Architecture – Designing for Performance", Sixth Edition, Pearson Education, 2003.
2. Morris Mano, M., "Computer System Architecture," 3/e, Pearson Education, 2005.
3. John P. Hayes, "Computer Architecture and Organization," 3/e, TMH, 1998.

REFERENCES:
1. David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software interface", Third Edition, Elsevier, 2005.
3. V.P. Heuring, H.F. Jordan, "Computer Systems Design and Architecture", Second Edition, Pearson Education, 2004.
4. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, "Computer Organization", Fifth Edition, Tata McGraw Hill, 2002.

CSE 309N	Simulation and Modeling					
Lecture	**Tutorial**	**Practical**	**Major Test**	**Minor Test**	**Total**	**Time**
3	1	-	75	25	100	3 Hour
Purpose	To introduce the principles and paradigms of Computer Modeling and Simulation for solving a wide variety of problems. In addition, how to use simulator to simulate the live systems.					
Course Outcomes (CO)						
CO1	Learn the basic concepts of System, System Modeling, types of Models, simulation, and need of simulation.					
CO2	Learn the simulation of continuous and discrete systems with the help of different examples.					
CO3	Learn the concept of generation of uniformly and non-uniformly distributed random numbers.					
CO4	Learn the simulation of queuing system and PERT.					

Unit-1
Modeling: System Concepts, system boundaries and environment, continuous and discrete systems, system modeling, types of Models, Model validation, Principles & Nature of Computer modeling.
Simulation: Introduction, Basic nature of simulation, when to simulate, Advantages, disadvantages and limitations of simulation, Concepts of simulation of continuous and discrete system with the help of example.

Unit-2
Continuous System Simulation: Analog vs. digital simulation, continuous simulation vs. numerical integration, simulation of a chemical reactor, simulation of a water reservoir system.
Discrete system simulation: Fixed time-step vs. event-to-event model, Monte-Carlo computation vs. stochastic simulation, generation of random numbers, and generation of non-uniformly distributed random numbers.

Unit -3
Simulators for the Live systems: Simulation of queuing Systems: basic concepts of queuing theory, simulation of single server, two server and more general queuing system.
Simulation of PERT network: Network model of a project, analysis of an activity network, critical path computation, uncertainties in activity durations, simulation of an activity network.

Unit-4
Simulation of inventory control systems: Elements of inventory theory, inventory models, generation of Poisson and Erlang variates, simulator for complex inventory systems.
Simulation of hypothetical computers.
Design and Evaluation of Simulation Experiments: Variance reduction techniques. Experiment layout and Validation.
Case Study: SciLab, Octave.

Text Books:
1. Gordon G.: System simulation, Prentice-Hall of India Pvt. Ltd. New Delhi 1993
2. Narsingh Deo: System Simulation with Digital Computer, PHI New Delhi, 1993

Reference Books:
1. Neelankavil Frances: Computer Simulation and Modelling, John Wiley & Sons, New York, 1987.
2. Payne, James A.: Introduction to simulation: Programming Techniques and Methods of Analysis, McGraw-Hill International Editions, Computer Science services, New York (1998).
3. Reitam Julian: Computer Simulation Experiments, Wiley Interscience 1971.

CSE-311N	Computer Networks Lab					
Lecture	**Tutorial**	**Practical**	**Minor Test**	**Practical**	**Total**	**Time**
--	--	3	40	60	100	3 Hour
Purpose	To explore networking concepts using Java programming & networking tools.					
Course Outcomes (CO)						
CO1	Do Problem Solving using algorithms.					
CO2	Design and test simple programs to implement networking concepts using Java.					
CO3	Document artifacts using applied addressing & quality standards.					
CO4	Design simple data transmission using networking concepts and implement.					

COMPUTER NETWORKS (Lab)
1. Create a socket for HTTP for web page upload and download.
2. Write a code simulating ARP /RARP protocols.
3. Study of TCP/UDP performance.
4. Performance comparison of MAC protocols
5. Performance comparison of routing protocols.
6. Write a program:
    a. To implement echo server and client in java using TCP sockets.
    b. To implement date server and client in java using TCP sockets.
    c. To implement a chat server and client in java using TCP sockets.
7. Write a program:
    a. To implement echo server and client in java using UDP sockets
    b. To implement a chat server and client in java using UDP sockets.
    c. To implement a DNS server and client in java using UDP sockets.
8. To flood the server from a spoofed source address leading to a DoS attack.
9. To sniff and parse packets that pass through using raw sockets.
10. To implement simple calculator and invoke arithmetic operations from a remote client.
11. To implement bubble sort and sort data using a remote client.
12. To simulate a sliding window protocol that uses Go Back N ARQ.

CSE-313N	Design and Analysis of algorithms Lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
--	--	3	40	60	100	3 Hour
Purpose	The student will learn the algorithm analysis techniques, become familiar with the different algorithm design techniques and Understand the limitations of Algorithm power.					
Course Outcomes (CO)						
CO1	The student should be able to Design algorithms for various computing problems					
CO2	The student should be able to Analyse the time and space complexity of algorithms.					
CO3	The student should be able to critically analyse the different algorithm design techniques for a given problem.					
CO4	The student should be able to modify existing algorithms to improve efficiency.					

List of Practical

1. Sort a given set of elements using the Quick sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
2. Using Open, implement a parallelized Merge Sort algorithm to sort a given set of elements and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
3. a. Obtain the Topological ordering of vertices in a given digraph.
    b. Compute the transitive closure of a given directed graph using Warshall's algorithm.
4. Implement 0/1 Knapsack problem using Dynamic Programming.
5. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
6. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.
7. a. Print all the nodes reachable from a given starting node in a digraph using BFS method.
    b. Check whether a given graph is connected or not using DFS method.
8. Find a subset of a given set S = {sl,s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions{1,2,6}and{1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.
9. Implement any scheme to find the optimal solution for the Traveling Salesperson problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.
10. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
11. Implement All-Pairs Shortest Paths Problem using Floyd's algorithm. Parallelize this algorithm, implement it using Open and determine the speed-up achieved.
12. Implement N Queen's problem using Back Tracking.
13. Implement Graph Coloring.
14. Find Hamiltonian Path using Back Tracking.
15. Implement longest common subsequence.
16. Implement Huffman code using Greedy approach.

CSE 315N	Simulation lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
-	-	3	40	60	100	3 Hour
Purpose	To introduce the principles and paradigms of Computer Simulation for solving a wide variety of problems. In addition, how to use simulator to simulate the live systems.					
Course Outcomes (CO)						
CO1	Learn the simulation of continuous and discrete systems with the help of different examples.					
CO2	Learn the concept of generation of uniformly and non-uniformly distributed random numbers.					
CO3	Learn the simulation of queuing system.					
CO4	Learn the concept of simulation CPM and PERT.					
CO5	Learn the concept of simulation of inventory control system.					

LIST OF EXPERIMENTS

1: Write a program to print the detailed marks certificate (D.M.C) of a student by using different binary operators.
2: Write a program to Draw graph of sine wave with respect to the time.
3: Write a program to solve following differential equation
$dy/dt = -\exp(-t) \times y^2$ by using any simulation technique.
4: Write a program to solve following differential equation by using 4th order Runge-Kutta method
$dy/dx = -2x-y$, with initial condition $y = -2$ when $x = 0$.
5: Write a program to simulate Pure-Pursuit problem of continuous system simulation.
6: Write a program to select a policy among different given policies with minimum total cost of an inventory system.
7: Write a program to generate and print a sequence of 30 pseudo random numbers between 150 to 250 by using any simulation technique.
8: Write a program to determine the approximate value of $\sqrt{2}$ using 1000 random numbers.
9: Write a program to generate a sample of pseudo random values by using rejection method from a given non-uniform distribution, when the probability function of the distribution is non-zero over finite interval (a, b).
10: Write a program to simulate single server queuing system with Poisson arrival pattern and FCFS queue discipline.
11: Write a program to find minimum time of completing the project by PERT.
12: Write a program to simulate an inventory system with the objective to determine the re-order combination (P,Q) which yields the highest service level for a given value of average stock.

CSE-319N	Technical Communication and Soft Skills Lab					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
-	-	2	0	100	100	3 Hours
Purpose	To enhance the students' oral communication skills in English					
Course Outcomes(CO)						
CO1	Develop oral communicative competence in English					
CO2	Improve fluency in English and thereby respond confidently due to reduced communication apprehension					
CO3	Identify and explain the biological and physiological characteristic of proper voice and diction production					
CO4	Develop correct and better pronunciation through stress on word accent, intonation, and weak forms					
CO5	Participate in Group Discussions effectively					
CO6	Make effective oral presentations in English					

LIST OF TOPICS FOR LAB ACTIVITIES

The following topics are prescribed to conduct the activities in the lab:

1. Articulation of Consonant sounds
2. Articulation of Vowel sounds
3. Pronunciation
4. Word Accent
5. Weak Forms
6. Intonation
7. Conversation in different formal situations
8. Group Discussion
9. Oral presentation

Bachelor of Technology (Computer Science & Engineering)
Scheme of Studies/Examination
Semester VI

S. No.	Course No.	Subject	L:T:P	Hours / Week	Examination Schedule (Marks)				Duration of Exam (Hrs)
					Major Test	Minor Test	Practical	Total	
1	CSE 302N	Compiler Design	3:1:0	4	75	25	0	100	3
2	CSE 304N	Essential of Information Technology	3:1:0	4	75	25	0	100	3
3	CSE 306N	Mobile Computing	3:1:0	4	75	25	0	100	3
4	CSE 308N	Web Technology	3:1:0	4	75	25	0	100	3
5	CSE 310N	Software Engineering	3:1:0	4	75	25	0	100	3
6	HS 303N	Business Intelligence and Entrepreneurship	4:0:0	4	75	25	0	100	3
7	CSE 312N	Web Technology Lab	0:0:3	3	0	40	60	100	3
8	CSE 314N	Essential of Information Technology Lab	0:0:3	3	0	40	60	100	3
9	CSE 316N	Software Engineering Lab	0:0:3	3	0	40	60	100	3
		Total		**33**	**450**	**270**	**180**	**900**	

CSE-302N	Compiler Design					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3 Hrs.
Purpose	At the end of the course, the student will be able to design and implement a compiler.					
Course Outcomes (CO)						
CO1	To understand, design and implement a lexical analyzer.					
CO2	To understand, design and implement a parser.					
CO3	To understand, design code generation schemes.					
CO4	To understand optimization of codes and runtime environment					

UNIT I

Introduction to Compiling

Analysis of the source program, Phases of a compiler, Cousins of the Compiler, Grouping of Phases, Compiler construction tools.

Lexical Analysis –Regular Expression, Introduction to Finite Automata and Regular Expression, Conversion of Regular Expression to NFA, Role of Lexical Analyzer, Input Buffering, Specification of Tokens.

UNIT II

Syntax   Analysis

Role of the Parser, Writing Grammars, Symbol Table, Context-Free Grammars, Top Down Parsing with or without Backtracking, Recursive Descent Parsing, Non-Recursive Descent Parsing, SLR Parser, Canonical LR Parser, LALR Parser.

UNIT III

Intermediate Code Generation and Code

Intermediate languages, Declarations, Assignment Statements, Boolean Expressions, Case Statements, DAG representation of Basic Blocks, A simple Code generator from DAG, Issues in the design of code generator , The target machine , Runtime Storage management, Error Handling- Type checking,

UNIT 1V

Code Optimization and Run Time Environments

Principal Sources of Optimization, Optimization of Basic Blocks, Peephole Optimization, Introduction to Global Data Flow Analysis, Source Language issues, Storage Organization, Static Storage Management, Heap Storage management, Access to non-Local Names, Parameter Passing.

TEXT BOOK
1.   Alfred Aho, Ravi Sethi, Jeffrey D Ullman, "Compilers Principles, Techniques and Tools", Pearson Education Asia, 2003.
REFERENCES
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.   J.P. Bennet, "Introduction to Compiler Techniques", Second Edition, Tata McGraw-Hill, 2003.
4.   Henk Alblas and Albert Nymeyer, "Practice and Principles of Compiler Building with C", PHI, 2001.
5.   Kenneth C. Louden, "Compiler Construction: Principles and Practice", Thompson Learning, 2003

CSE-304N	Essentials of Information Technology					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	0	75	25	100	3 Hrs.
Purpose	To introduce the concepts of Object Oriented Programming using Java and RDBMS					
Course Outcomes (CO)						
CO1	Do Problem Solving using algorithms					
CO2	Design and test simple programs to implement Object Oriented concepts using Java					
CO3	Document artifacts using common quality standards					
CO4	Design simple data store using RDBMS concepts and implement					

Focus Area 1: Object Oriented Programming using Java
Unit I:
Problem Solving Techniques: Introduction to problem solving, Computational problem and its classification - Logic and its types, Introduction to algorithms and flowchart, Searching algorithms: linear search, binary search and sorting algorithms: insertion, quick, merge and selection sort, Introduction and classification to Data Structures, Basic Data Structures: array, stack, and queue.

Unit II:
Programming Basics: Identifiers, variables, data types, operators, control structures, type conversion, casting, arrays, strings
Object Oriented Concepts fundamentals: class & object, instance variables & methods, access specifiers, reference variables, parameter passing techniques, constructors, this reference, static, and command line arguments
Introduction to UML: Use case diagrams – Class diagrams

Unit III:
Relationships: aggregation, association, Inheritance, types of inheritance, Static Polymorphism: method overloading, constructor overloading, Dynamic polymorphism: method overriding, abstract, interface, introduction to packages Industry Coding Standards and Best Practices, code tuning & optimization, clean code & refactoring

Focus Area 2: Relational Database Management System
Unit IV:
RDBMS- data processing, the database technology, data models, ER modelling concept, notations, converting ER diagram into relational schema, Logical database design, normalization (1NF, 2NF and 3NF)
SQL: DDL statements, DML statements, DCL statements, Joins, Sub queries, Views, Database design Issues, SQL fine-tuning

Books on Java
1. **Java**™: The **Complete Reference**,. Seventh Edition. Herbert Schildt
2. Programming with **Java 3e A Primer** by E **Balagurusamy**
3. Introduction to Java Programming by K. Somasundaram , Jaico Publishing House; 1 edition
Books on RDBMS, Oracle, MYSQL
1. Fundamentals of Database Systems, with E-book (3rd Edition) by Shamkant B. Navathe, Ramez Elmasri, Published January 15th 2002 by Addison Wesley Longman
2. MySQL by Paul DuBoisNew Riders Publishing
3. Murach's MySQL Paperback – 2012, by Joel Murach , Publisher: Shroff/Murach (2012)
4. SQL: The Complete Reference by James R. Groff, Paul N. Weinberg, Published March 1999 by McGraw-Hill Companies
5. Schaum's Outline of Fundamentals of Relational Databases by Ramon Mata-Toledo, Published November 15th 2000 by McGraw-Hill

CSE-306N			Mobile Computing			
Lecture	**Tutorial**	**Practical**	**Major Test**	**Minor Test**	**Total**	**Time**
3	1	0	75	25	100	3 Hrs.
Purpose	To impart knowledge of mobile and wireless computing systems and techniques.					
Course Outcomes(CO)						
CO1	Describe the concepts of mobile computing and cellular networks.					
CO2	Learn the basic concepts of wireless networks.					
CO3	Study of various issues of mobile computing and basics of cloud computing.					
CO4	Description and applications of Ad hoc networks.					

UNIT – I

Introduction, issues in mobile computing, overview of wireless telephony: cellular concept, Mobile computing Architecture, Design considerations for mobile computing, Mobile Computing through Internet, Making existing applications mobile enabled. GSM: air-interface, channel structure, location management: HLR-VLR, hierarchical, handoffs, channel allocation in Cellular systems, WCDMA, GPRS 3G, 4G.

UNIT – II

Wireless Networking, Wireless LAN Overview: MAC issues, IEEE 802.11, Blue Tooth, Wireless multiple access protocols, TCP over wireless, Wireless applications, data broadcasting, Mobile IP, WAP : Architecture, Traditional TCP, Classical TCP, improvements in WAP, WAP applications.

UNIT – III

Data management issues, data replication for mobile computers, adaptive clustering for mobile wireless networks, File system, Disconnected operations Mobile Agents computing, security and fault tolerance, transaction processing in mobile computing environment.
Cloud Architecture model, Types of Clouds: Public Private & Hybrid Clouds, Resource management and scheduling, Clustering, Data Processing in Cloud: Introduction to Map Reduce for Simplified data processing on Large clusters.

UNIT – IV

Ad hoc networks, localization, MAC issues, Routing protocols, global state routing (GSR), Destination sequenced distance vector routing (DSDV), Dynamic source routing (DSR), Ad Hoc on demand distance vector routing (AODV), Temporary ordered routing algorithm (TORA), QoS in Ad Hoc Networks, applications.

Text Books:
1. Rajkamal, Mobile Computing, 2/E Oxford University Press,2011.
2. J. Schiller, Mobile Communications, Addison Wesley
3. Yi Bing Lin, Wireless and Mobile Networks Architecture , John Wiley.

Reference Books

1. A. Mehrotra , GSM System Engineering.
2. M. V. D. Heijden, M. Taylor, Understanding WAP, Artech House.
3. Charles Perkins, Mobile IP, Addison Wesley.
4. Charles Perkins, Ad hoc Networks, Addison Wesley.
5. Judith Hurwitz, Robin Bllor, Marcia Kaufmann, Fern Halper, Cloud Computing for Dummies, 2009.

CSE-308N	Web Engineering					
Lecture	**Tutorial**	**Practical**	**Major Test**	**Minor Test**	**Total**	**Time**
3	1	-	75	25	100	3
Purpose	To gain a broad understanding of the discipline of Web engineering and its application to the development and management of Web Applications.					
Course Outcomes						
CO1	Learn the basic concepts of information and web architecture.					
CO2	Learn about the skills that will enable to design and build high level web enabled applications.					
CO3	Understand the applicability of Java Script as per current software industry standards.					
CO4	Acquaint the latest programming language for the implementation of object based and procedure based applications using Python.					

Unit-1

Information Architecture: The role of Information Architect, Collaboration and communication, Organizing information, organizational challenges, Organizing web sites and Intranets, Creating cohesive organization systems, designing navigation systems, types of navigation systems, Integrated navigation elements, designing elegant navigation systems, Searching systems, Searching your web site, designing the search interface, Indexing the right stuff, To search or not to search grouping content, conceptual design, High level Architecture Blueprint. Architectural Page Mockups, Design Sketches.

Unit-2

Introduction to XHTML and HTML5: Origins and Evolution of HTML and XHTML, Basic Syntax, Standard XHTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, HTML5, Syntactic Differences between HTML and XHTML.
Cascading Style Sheets: Introduction, Levels of Style Sheets, Style Specification Formats, Selector Forms, Property Value Forms, Font Properties, List Properties, Color, Alignment of Text, Box Model, Background Images, Conflict Resolution.

Unit -3

Java Script: Overview of JavaScript, Object Orientation and JavaScript, General Syntactic Characteristics, Primitives, Operations, and Expressions, Screen Output and Keyboard Input, Control Statements, Object Creation and Modification, Arrays, Functions, Constructors, Pattern Matching Using Regular Expressions, Errors in Scripts

Unit -4

Python: Introduction to Python, Data Types and Expressions, Control Statements, Strings and Text Files, Lists and Dictionaries, Design with Functions, Design with Classes

Text Books
1. By Peter Morville, Louis Rosenfeld, "Information Architecture on the World Wide Web", O'Reilly Media, 2006.
2. Robert W. Sebesta, "Programming The World Wide Web", Eight Edition, Pearson India, 2015.
3. Kenneth A. Lambert, "The Fundamentals of Python: First Programs", 2011, Cengage Learning.
Reference Book
1. Thomas A Powell, "HTML The Complete Reference", Tata McGraw Hill Publications.

CSE-310N	Software Engineering					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3
Purpose	To gain a broad understanding of the discipline of software engineering and its application to the development and management of software process.					
Course Outcomes(CO)						
CO1	To understand the basic concepts of Software Engineering.					
CO2	To learn about the skills that will enable to construct high quality software.					
CO3	To understand the software process models.					
CO4	To understand the fundamental concept of requirements engineering and Analysis Modelling.					
CO5	To understand the different design techniques and their implementation.					
CO6	To learn about software testing and maintenance measures.					

Unit-I

Introduction: Introduction to Software Engineering, Software Characteristics, Software Crisis, The Evolving role of Software, Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, RAD, V Model.

Unit-II

Software Requirement Specification: Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Data Flow Diagrams, Decision Tables, SRS Document, IEEE Standard for SRS.
Software Quality: Software Quality, Concept of Software Quality Assurance (SQA), SEI-CMM Model.
Introduction to Software Risk Management and Software Configuration Management

Unit-III

Software Design: Basic Concept of Software Design, Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion.
Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design.
Software Measurement and Metrics: Various Size Oriented Measures: Halstead's Software Science, Function Point (FP) Based Measures, COCOMO, Cyclomatic Complexity Measures: Control Flow Graphs.

Unit-IV

Software Construction: Software construction fundamentals, minimizing complexity, Top-Down and Bottom –Up programming, structured programming, Compliance with Design and Coding Standards.
Testing: Testing Objectives, Unit Testing, Integration Testing, system testing, Acceptance Testing, Regression Testing, Structural Testing, Functional Testing, debugging.
Maintenance: key issues, Types of software Maintenance, Cost of Maintenance, Software Re-Engineering.

Text Books:
1.   R. S. Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
2.   K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.
Reference Books:
1.   Pankaj Jalote, Software Engineering, Wiley India.
2.   Rajib Mall, Fundamentals of Software Engineering, PHI Publication.
3.   Ian Sommerville, Software Engineering, Addison Wesley.

HS-303N	Business Intelligence & Entrepreneurship					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
4	-	-	75	25	100	3
Course Outcomes						
CO1	Students will be able understand who the entrepreneurs are and what competences needed to become an Entrepreneur					
CO2	Students will be able understand insights into the management, opportunity search, identification of a Product; market feasibility studies; project finalization etc. required for small business enterprises.					
CO3	Students can be able to write a report and do oral presentation on the topics such as product identification, business idea, export marketing etc.					
CO4	Students be able to know the different financial and other assistance available for the establishing small industrial units.					

Unit -I

Entrepreneurship: Concept and Definitions; Entrepreneurship and Economic Development; Classification and Types of Entrepreneurs; Entrepreneurial Competencies; Factor Affecting Entrepreneurial Growth – Economic, Non-Economic Factors; EDP Programmes; Entrepreneurial Training; Traits/Qualities of an Entrepreneurs; Entrepreneur; Manager Vs. Entrepreneur.

Unit -II

Opportunity / Identification and Product Selection: Entrepreneurial Opportunity Search and Identification; Criteria to Select a Product; Conducting Feasibility Studies; Project Finalization; Sources of Information.

Unit -III

Small Enterprises and Enterprise Launching Formalities : Definition of Small Scale; Rationale; Objective; Scope; Role of SSI in Economic Development of India; SSI; Registration; NOC from Pollution Board; Machinery and Equipment Selection; Project Report Preparation; Specimen of Project Report; Project Planning and Scheduling using Networking Techniques of PERT / CPM; Methods of Project Appraisal.

Unit -IV

Role of Support Institutions and Management of Small Business : Director of Industries; DIC; SIDO; SIDBI; Small Industries Development Corporation (SIDC); SISI; NSIC; NISBUD; State Financial Corporation SIC; Marketing Management; Production Management; Finance Management; Human Resource Management; Export Marketing; Case Studies-At least one in whole course.

Text Books:

1. Small-Scale Industries and Entrepreneurship. Himalaya Publishing House, Delhi -Desai, Vasant, 2003.
2. Entrepreneurship Management -Cynthia, Kaulgud, Aruna, Vikas Publishing House, Delhi, 2003.
3. Entrepreneurship Ideas in Action- L. Greene, Thomson Asia Pvt. Ltd., Singapore, 2004.

CSE-312N	Web Engineering Lab					
Lecture	**Tutorial**	**Practical**	**Minor Test**	**Practical**	**Total**	**Time**
0	0	3	40	60	100	3 Hrs.
Purpose	To introduce the concepts of HTML5, JavaScript and Python.					
Course Outcomes (CO)						
CO1	Design webpages using HTML, JavaScript and CSS.					
CO2	Design and test simple function/program to implement Searching and sorting techniques using Python.					
CO3	Develop program in Java Script for pattern matching using regular expressions and errors in scripts.					
CO4	Design client-server based web applications.					

[1]   Create your own page with your favorite hobbies using HTML, JavaScript and CSS.
[2]   Create a frameset in HTML that is divided into three sections. The frameset should have three zones.
   a.   The Topmost section of the frameset should take up about just 15% of the browser window. Name this frame title.
   b.   The middle section should be 75% of the browser window. Name this frame title.
   c.   The lower section should be 10% of the browser window. Name this frame menu.
[3]   Create pages for each section. For the lowermost section, create page that loads the content into the middle section. The topmost section should contain a page describing the web page itself.
[4]   Create a web page, which displays the map of your country Link, each city /state on the image map, such that the respective HTML page of the city/state is displayed when the user selects an area.
[5]   Add the tickertape applet to your page by customizing it for the following settings:
   a.   Increase the count by one.
   b.   Accordingly update the message count.
   c.   Change the text color to (237,192,171)
   d.   Experiment with changing the scrolling speed.
   e.   Customize the message text as per your page requirement.
[6]   Incorporate a quest book into the Diary Food Webpage and use Java Script to build validations into the form.
[7]   Use Cascading Style sheets (CSS) to modify the following:
   a.   Change background.
   b.   Change font type, face and color.
   c.   Align Text.
   d.   Remove underlines from hyperlinks.
[8]   Write the program for using JavaScript by using for – loops (through a block of code a number of times), for/in - loops (through the properties of an object), while - loops (through a block of code while a specified condition is true), do/while - loops (through a block of code while a specified condition is true).
[9]   Write a program in Java Script for the following:
   a.   Copying, passing, and comparing by value
   b.   Copying, passing, and comparing by reference
   c.   References themselves are passed by value
[10]   Write program in Java Script for pattern matching using regular expressions and errors in scripts.
[11]   Write a Python function/program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is an equilateral triangle.
[12]   Write the Python functions for linear search, binary search, selection sort, Bubble Sort, Insertion Sort and converting Fibonacci to a linear algorithm.
[13]   Write program in Python using Lists and dictionaries, Control statements and Strings and text files.

CSE-314N	Essentials of Information Technology Lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
0	0	3	40	60	100	3 Hrs.
Purpose	To introduce the concepts of Object Oriented Programming using Java and RDBMS					
Course Outcomes (CO)						
CO1	Do Problem Solving using algorithms					
CO2	Design and test simple programs to implement Object Oriented concepts using Java					
CO3	Document artifacts using common quality standards					
CO4	Design simple data store using RDBMS concepts and implement					

Students should implement at least 4-5 problems from the real world related to concern engineering branch for following both focus area during Practical hours:

1. Programs using Java Language
2. RDBMS Queries using MySQL

Tools:

- Understanding basic programming constructs using Scratch Tool - Flowcharts implementation through RAPTOR tool
- Eclipse IDE for Java programming

CSE-316N	Software Engineering Lab						
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time	
-	-	3	40	60	100	3	
Purpose	To gain a broad understanding of the discipline of software engineering implementation.						
Course Outcomes							
CO1	To understand the basic concepts of Software Engineering.						
CO2	To learn about the reasons for the software crisis.						
CO3	To understand the software testing techniques.						
CO4	To understand the software metrics.						
CO5	To understand the different design techniques and their implementation.						
CO6	To learn about software testing and maintenance measures.						

List of Practical's

1. To identify the role of the software in today's world across a few significant domains related to day to day life.
2. To identify the problem related to software crisis for a given scenario.
3. To classify the requirement into functional and non-functional requirements.
4. To implement at least four software metrics.
5. Preparation of requirement document for standard application problems in standard format.(e.g Library Management System, Railway Reservation system, Hospital management System, University Admission system)
6. To prepare Project Schedule for standard application problems in standard format.
7. To implement the functional testing techniques.
8. To implement the structural testing techniques