

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: DSC 7
Course Code:	Course Title: Theory of Computation	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand formal languages, grammars and Chomsky hierarchy. • Design regular grammar, DFA, NFA, Mealy and Moore machine, PDA, Turing machines. • Understand the notion of decidability and computability. 			
Course Prerequisites: DSC 1			
Course Outline:			
Units	Topics		No. of Lab/Lectures
I.	Finite automaton (FA): Transition system, Acceptance by a finite automaton, Deterministic and non-deterministic automaton (DFA and NFA), Equivalence of DFA and NFA, Minimization of states in a finite automaton, Mealy and Moore machines.		12
II.	Formal languages and grammar: - Chomsky's hierarchy, Regular grammar and regular expression, REs and FA, Closure Properties, Pumping lemma for regular sets.		12
III.	Push down automaton (PDA): Acceptance by PDA using final state or empty stack, Context free language, Deterministic CFL, Deterministic PDA, Context free grammar and derivation trees, Leftmost and rightmost derivation, Ambiguity in context free grammar, Pumping lemma for context free languages.		12
IV.	Turing machine (TM): TM as computable functions and accepters, Non-deterministic TM, type-0 grammar, Halting problem of a TM, Linear bounded automaton (LBA) model, Context sensitive languages and grammars.		12
V.	Unsolvable problems: Reduction techniques, Decidability– Post's correspondence problem (PCP), Rice's theorem, Decidability of membership, Emptiness and equivalence problems of languages, P, NP, NP-Completeness; Satisfiability and Cook's theorem.		12
Textbooks:			
<ol style="list-style-type: none"> 1. KLP Mishra, N. Chandrasekaran, "Theory of Computer Science (Automata, Languages and Computation)", PHI, 3rded. 2. Peter Linz, Jones, Bartlett, "An Introduction to Formal Languages and Automata", 5thed. 3. John E. Hopcroft, J.D. Ullman, Rajiv Motwani, "Introduction to Automata Theory, Languages and Computation", Pearson Education, 3rded. 4. Michael Sipser, "Introduction to the Theory of Computation", Cengage Learning, 3rded. 			

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VII	Course Type: DSE 6
Course Code:	Course Title: Computer Networks	Credits: L T P (3 1 0)	
Course outcomes: <ul style="list-style-type: none"> Understanding Network topologies and network architecture. Demonstrate a clear understanding of the different layers of network architecture. 			
Course Prerequisites: DSC 1			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
I.	Introduction to Computer Networks and Data Communication: Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite. Analog and digital signal; data-rate limits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission; digital to analog modulation; multiplexing techniques– FDM, TDM; transmission media.	12	
II.	Networks Switching Techniques and Access mechanisms: Circuit switching; packet switching– connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.	12	
III.	Data Link Layer: Error detection and error correction techniques; data-link control-framing and flow control; Error recovery protocols– stop and wait ARQ, go-back-n ARQ; CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks– repeaters, hubs, switches, bridges, router and gateways.	12	
IV.	Network and Transport Layers: Routing; routing algorithms; Network layer protocol of Internet– IP protocol, Internet control protocols, Transport services, Transport protocols, Internet Transport Protocols– UDP, TCP.	12	
V.	Application Layer: Client–server model, Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), Telnet, Network Virtual Terminal (NVT), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), User Agent (UA), Mail Transfer Agent (MTA), Multipurpose Internet Mail Extensions (MIME), Post Office Protocol (POP), Simple Network Management Protocol (SNMP), Hypertext Transfer Protocol (HTTP), Uniform Resource Locator (URL), World Wide Web (WWW).	12	
Textbooks: <ol style="list-style-type: none"> B.A. Forouzan, “Data Communications and Networking”, THM, 4th ed., 2007. Andrew S. Tanenbaum, “Computer Networks”, PHI, 4th ed., 2003. 			
References: <ol style="list-style-type: none"> J.F. Kurose, K.W. Ross, “Computer Networking: A Top-Down Approach”, Pearson, 6th ed., 2012. Leon Garcia, Widjaja, “Communication Networks: Fundamental Concepts and Key Architectures”, Tata McGraw Hill, 2001. 			

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VII	Course Type: DSE 7
Course Code:	Course Title: Discrete Mathematics	Credits: L T P (3 1 0)	
Course outcomes: <ul style="list-style-type: none"> Understand and interpret the fundamental mathematical structures like Set theory, Relation and Functions Write recursive definitions of sequences and collections of objects Understand the concepts and applications of vector algebra Understand and interpret the basic concepts of Graph Theory Apply the use of graph theory concepts solving various Computer Science and Engineering problems. 			
Course Prerequisites: Mathematics in 10+2			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
I.	Logic and Proofs: Propositional Logic– Binary logic and propositions, Propositional Variables, Truth table, Logical connectives– Negation, Conjunction, Disjunction, Conditional, Biconditional, Universal connectives, Well-formed Formulas, Tautology, Contradiction and Contingency, Propositional Equivalences, Duality, Predicate Logic– Predicates, Quantifiers– Existential and Universal quantifier, Predicate formulas, Equivalence of formulas involving quantifiers, Normal forms– CNF/DNF, PCNF/PDNF, Normal forms for First Order Logic– Prenex Normal Form, Rules of Inference. Proof Techniques– Introduction to Proof, Definitions– Theorem, Lemma, Corollary and Conjecture, Methods of Proof– Direct Proofs, Indirect Proofs– Proof by Contraposition, Proof by Contradiction, Proof by Cases, Mathematical Induction.	12	
II.	Counting and Recurrence: Set Theory, Countable and Countably Infinite Sets, Pigeonhole Principle, Permutation and Combination, Principle of Inclusion and Exclusion, Generating functions– Definition, Generating Permutations and Combinations. Recurrence– Recurrence Relations, Linear Recurrence Relations with constant coefficients and their solution, Solving Linear Recurrence Relations using Generating Functions.	12	
III.	Binary and Ordered Relations: Binary Relation, Properties of Binary Relations– Reflexive, Symmetric and Transitive Relation, Equivalence Relation, Closure of Relations– Reflexive, Symmetric and Transitive Closure, Warshall’s algorithm, Ordered Relation– Partial Order and Posets, Hasse diagram of Poset, Maximal, Minimal, Maximum and Minimum of poset, glb and lub, Isomorphic ordered set, Well ordered	12	

	set, Lattice, Properties of lattice, Distributed and Complemented lattice, Applications of Lattice, Topological Sort.	
IV.	Graph Theory: Definition of Graph, Types– Directed and Undirected Graph, Complete Graph, Bipartite Graph, Multigraph, Weighted Graph, Graph Representation– Adjacency matrix and Adjacency list, Graph Isomorphism, Connectivity and Path, Euler and Hamiltonian Paths and Circuits, Shortest path– Dijkstra’s algorithm, Planar Graph, Euler’s theorem for Planar Graphs, Graph Coloring. Trees– Basic terminology and properties, Tree Traversal– Inorder, Preorder and Postorder, Expression Trees– Infix, Prefix and Postfix notations, Spanning Trees– Kruskal’s and Prim’s algorithms for Minimum Spanning Trees (MST).	12
V.	Algebra: Definition and elementary properties of Semigroups, Monoids, Groups, Subgroups, Generators and Cyclic group, Permutation group, Cosets, Lagrange’s Theorem, Rings, Integral Domains and Fields	12
Textbooks: <ol style="list-style-type: none"> 1. Kenneth H. Rosen, “Discrete Mathematics and Its Applications”, Tata McGraw Hill, 7th ed., 2012. 2. C. L. Liu, “Elements of Discrete Mathematics”, McGraw Hill, 2nd ed., 1986. 3. Bernard Kolman, Robert C. Busby, Sharon Cutler Ross, “Discrete Mathematical Structures”, Pearson Education, 6th ed., 2008. 4. J. P. Tremblay, R. Manohar, “Discrete Mathematical Structures with Applications to Computer Science”, Tata McGraw Hill, 1st ed., 2001. 5. Susanna S. Epp, “Discrete Mathematics with Applications”, 4th ed., 2010. 		

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: DSE 8
Course Code:	Course Title: Research Methodology	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand the concept of research. • Understand the concept of data collection and selection for research. • Understand the applicability of research for public at large. 			
Course Prerequisites: Basic Knowledge of Computer			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
I.	Introduction to Research Methods in science – Philosophy of Science, Research methods and Creative Thinking, Evolutionary Epistemology, Scientific Methods, Hypotheses Generation and Evaluation, Code of Research Ethics, Definition and Objectives of Research, Various Steps in Scientific Research, Research presentations Types of Research – Research Purposes – Research Design – Survey Research – formulation of scientific problems and hypotheses selection of methods for solving a scientific problem Case Study Research.	20	
II.	How to perform a literature review – Sampling Methods – Data Processing and Analysis strategies - Data Analysis with Statistical Packages – Statistical Analysis – Hypothesis-testing – Generalization and Interpretation.	20	
III.	Research Reports - Structure and Components of Research Report, Types of Report, Layout of Research Report, Mechanism of writing a research report – Requirements of a good dissertation.	20	
Textbooks:			

1. Oates, B.J., (2005). Researching Information Systems and Computing. Sage Publications, UK.
2. Zobel, J. (2004). Writing for Computer Science - The art of effective communication. 2nd ed., Springer, UK.
3. Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., 2002. An introduction to Research Methodology, RBSA Publishers.
4. Kothari, C.R., 1990. Research Methodology: Methods and Techniques. New Age International. 5
5. Sinha, S.C. and Dhiman, A.K., 2002. Research Methodology.

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VII	Course Type: DSE 9/GE 8
Course Code:	Course Title: Mobile App Design and Development		Credits: L T P (3 0 1)
Course outcomes:			
<ul style="list-style-type: none"> • Learn about mobile devices, Android OS, and architecture, and how to use Android Studio. • Understand key components like Intents, Manifest files, and permissions. • Explore activity life cycles, types of intents, and data sharing using intents. • Develop skills in creating UI elements, handling events, using animations, notifications, and fragments. 			
Course Prerequisites: GE 4/DSE 2			
Course Outline:			
Units	Topics		No. of Lab/Lectures
I.	Android Systems: Introduction to Mobile devices and applications, Open Handset Alliance (OHA), Overview of Android OS and architecture, installing android studio. Introduction to Android application components, Intents, Android Manifest File and its common settings, Using Intent Filter, Permissions Activities and intents: understanding activity and its life cycle, Types of intents, intent filter, context, data sharing using intent		12
II.	Android User Interface: Basic android UI, layouts, view and view attributes, button, controls. UI events and event listeners, animations, notifications, progress dialog, Action		12

	bar, toolbar, menus and pop ups, Tab based UI, Fragment, Types of Fragment, Fragment Lifecycle, communication between fragment and activity	
III.	Android Storage and APIs: Android storage: Using Android Data and Storage APIs, Managing data using SQLite, Sharing Data between Applications with Content Providers Android APIs: Multimedia, Using Android Networking APIs, Using Android Web APIs, Using Android Telephony APIs, android location based services	12
IV.	iOS Technology Stack: Introduction to iOS technology stack: iOS architecture, StoryBoard, features of Xcode, components of iOS SDK. Introduction to swift: data types, variables, control flow and operators, Collections and functions in swift, classes and structures, inheritance, closure and enumerations	12
V.	User interactions: Controls, gesture organizers, touching views, Core Location and Mapkit, using Google Maps in iOS. Sensors in iOS. Data persistence: Core Data framework for storing persistent data, CRUD operations.	12
Lab: Mobile App Design and Development		15
Textbooks:		
<ol style="list-style-type: none"> "Android Programming: The Big Nerd Ranch Guide" by Bill Phillips and Brian Hardy - Offers a comprehensive introduction to Android development. "Android Application Development All-in-One For Dummies" by Barry Burd - Covers various aspects of Android app development, including UI design and application components. "iOS Programming: The Big Nerd Ranch Guide" by Christian Keur and Aaron Hillegass - Provides a detailed guide to iOS development, including Swift programming and user interactions. "Learning Swift: Building Apps for macOS, iOS, and Beyond" by Paris Buttfield-Addison, Jon Manning, and Tim Nugent - Focuses on Swift programming for iOS development. 		

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: Dissertation 1
Course Code:	Course Title: Dissertation on Major		Credits: 6
Course outcomes:			
<ul style="list-style-type: none"> Develop the ability to conduct thorough research, including literature review, problem formulation, and application of appropriate methodologies. Demonstrate advanced technical skills and proficiency in relevant tools, technologies, and programming languages. Enhance skills in presenting research findings and defending the dissertation through clear and coherent written and oral communication. Apply ethical principles in research and cultivate critical thinking skills for evaluating results and making informed conclusions. 			
Course Prerequisites: DSE 8			
Course Outline:			

	Writing a sample Research Project/Dissertation/Research Paper on any given topic covering all the components of Research.	
--	---	--

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VIII	Course Type: DSC 8
Course Code:	Course Title: Compiler Design	Credits: L T P (3 1 0)	
Course outcomes: <ul style="list-style-type: none"> • Understanding of Compiler Architecture • Ability to Implement Lexical and Syntax Analysis: • Knowledge of Semantic Analysis and Intermediate Code Generation • Experience with Compiler Optimization and Code Generation 			
Course Prerequisites: Basic understanding of compilers and programming languages.			

Course Outline:		
Units	Topics	No. of Lab/Lectures
I.	Compiler Structure: Analysis-synthesis model of compilation, various phases of a compiler, tool-based approach to compiler construction. Lexical analysis: Interface with input parser and symbol table, token, lexeme and patterns, difficulties in lexical analysis, error reporting and implementation. Regular grammar & language definition, Transition diagrams, Design of a typical scanner using LEX or Flex.	12
II.	Syntax Analysis: Context free grammars, ambiguity, associability, precedence, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing LL(1) grammar, Bottom up parsing, operator precedence grammars, LR parsers (SLR, LALR, LR), Design of a typical parser using YACC or Bison.	12
III.	Syntax directed definitions: Inherited and synthesized attributes, dependency graph, evaluation order, bottom up and top down evaluation of attributes, L- and S-attributed definitions. Type checking: type system, type expressions, structural and name equivalence of types, type conversion, overloaded function and operators, polymorphic function. Run time system: storage organization, activation tree, activation record, parameter passing symbol table, dynamic storage allocation. Intermediate code generation: intermediate representation, translation of declarations, assignments, Intermediate Code generation for control flow, Boolean expressions and procedure calls, implementation issues.	12
IV.	Code generation and instruction selection: Issues, basic blocks and flow graphs, register allocation, code generation, DAG representation of programs, code generation from DAGS, peep hole optimization, code generator, specification of machine.	12
V.	Code optimization: Source of optimizations, optimization of basic blocks, loops, global dataflow analysis, solution to iterative dataflow equations, code improving transformations, dealing with aliases, data flow analysis of structured flow graphs.	12
Textbooks:		
1. K. C. Loudon, "Compiler Construction, Principle and Practice", Cengage Publication, 6 th ed., 2009.		
2. Alfred V. Aho, Ravi Sethi, Jeffrey, D. Ullman, "Compilers: Principles, Techniques and Tools", Pearson, 1998.		
3. V. Raghvan, "Principles of Compiler Design", TMH, 2009.		
4. Levine, Mason, Brown, "Lex & Yacc", O' Reilly, 1998.		
5. S. S. Muchnick Harcourt Asra, "Advanced Compiler Design implementation", Morgan Kaufman, 2006.		
6. Allen, "Modern Compiler Implementation in C", Cambridge University Press, 1997.		
7. Vinu V. Das, "Compiler Design using FLEX and YACC", PHI, 2005.		
8. Cooper, "Engineering a Compiler", Elsevier, 2005.		
9. Alan I. Holub, "Compiler Design in C", PHI, 2009. 10. Fisher, "Crafting a Compiler in C", Pearson, 2005.		
10. Fisher, "Crafting a Compiler in C", Pearson, 2005.		

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VIII	Course Type: DSE 10
Course Code:	Course Title: Design & Analysis of Algorithm	Credits: L T P (3 0 1)	
Course outcomes:			

- Understand concept of asymptotic analysis and perform complexity analysis of iterative and recursive algorithms.
- Formulate and solve time complexity recurrence relations using various techniques.
- Solve computational problems using various algorithmic paradigms like divide-and conquer, greedy, dynamic programming, backtracking, branch-and-bound.

Course Prerequisites: Basic understanding of algorithms and data structures

Course Outline:

Units	Topics	No. of Lab/Lectures
I.	Introduction: Review of Asymptotic Notations, Mathematical analysis for Recursive and Non-recursive algorithms, solving recurrence relations.	10
II.	Algorithm Design Techniques: Brute Force, Exhaustive Search, Divide and conquer, Merge sort, Quick sort, Binary search, Multiplication of Large Integers, Strassen's Matrix Multiplication. Greedy strategy– General Approach and problems like Optimal Merge Patterns, Minimum Spanning Trees algorithms, Knapsack Problem, Huffman Code, Job sequencing with deadlines, single source shortest path. Dynamic Programming– General Approach, Memoization, Multistage Graph, Matrix-Chain Multiplication, Longest Common Subsequence, Knapsack Problem, Floyd Warshall algorithm, Optimal Binary Search Trees.	20
III.	Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP, NP-Hard and NP-Complete Problems, Intractability, Cook's Theorem, Reductions. Coping with the Limitations – Backtracking concept; Branch & Bound method, Approximation Algorithms.	15
Lab: Design & Analysis of Algorithm		15

Textbooks:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Prentice Hall of India, 3rd ed., 2010.
2. R. C. T. Lee, S. S. Tseng, R. C. Chang, Y. T. Tsai, "Introduction to the Design and Analysis of Algorithms: A Strategic Approach" McGraw Hill, 2006.
3. Anany Levitin, Introduction to the Design and Analysis of Algorithms, Pearson Education, 2007.
4. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", University Press, 2nd ed., 2008.
5. Kenneth A. Berman, Jerome Paul, "Algorithms: Sequential, Parallel and Distributed", Cengage Learning, 2004.
6. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "The Design and Analysis of Computer Algorithms" Pearson Education, 2008.
7. Michael T. Goodrich, Roberto Tamassia, Algorithm Design, Wiley, 2002.
8. S. Dasgupta, C. Papadimitriou, and U. Vazirani. Algorithms. McGraw-Hill Higher Education, 2006

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VIII	Course Type: DSE 11
Course Code:	Course Title: Artificial Intelligence	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand the concept of Artificial Intelligence. • Understand to apply the knowledge and reasoning for different components. • Understand the Expert Systems and their uses. • Understand the basics of PROLOG. 			
Course Prerequisites: DSC 1			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
I.	Introduction and applications of artificial intelligence, Problem solving: State space search, Production system, Problem characteristics, Problem system characteristics, Search techniques: Generate and test, Hill climbing, Best first search, A* algorithm, Problem reduction	12	
II.	Knowledge and Reasoning: Knowledge acquisition, Knowledge engineer, Cognitive behavior, Knowledge representation: Level of representation, Knowledge representation schemes, Formal logic, Inference Engine, Semantic net, Frame, Scripts. Adversarial search, Optimal and imperfect decisions, Alpha, Beta pruning, Logical agents: Propositional logic, First order logic – Syntax and semantics – Inference in first order logic. Uncertain Knowledge and Reasoning: Uncertainty – Acting under uncertainty – Basic probability notation – Axioms of probability – Baye’s rule – Probabilistic reasoning – Making simple decisions.	12	
III.	Expert systems: Definition, Role of knowledge in expert system, Architecture, Expert System Development Life Cycle: Problem selection, Prototype construction, Formalization, Implementation, Evaluation	12	
IV.	Planning and Learning: Planning: Planning problem – Partial order planning – Planning and acting in non-deterministic domains – Learning: Learning decision trees, Knowledge in learning, Neural networks, Reinforcement learning – Passive and active.	12	
V.	PROLOG Programming: Introduction, variables, using rules, Input and Output predicates, Fail and cut predicates, Recursion, Arithmetic operation, Compound object, Dynamic database, Lists, String, File operations.	12	
Textbooks:			
<ol style="list-style-type: none"> 1. Elaine Rich, Kevin Knight, “Artificial Intelligence”, Tata McGraw Hill. 2. Dan W. Patterson, “Introduction to Artificial Intelligence and Expert Systems”, Prentice Hall of India. 3. Nils J. Nilsson, “Principles of Artificial Intelligence”, Narosa Publication house. 4. Stuart Russell, Peter Norvig, “Artificial Intelligence: A Modern Approach”, Pearson Education, 2nded. 5. Winston, Patrick, Henry, “Artificial Intelligence”, Pearson Education. 6. Gopal Krishna, Janakiraman, “Artificial Intelligence”. 			

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VIII	Course Type: DSE 12
Course Code:	Course Title: Computer Graphics	Credits: L T P (3 1 0)	
Course outcomes: <ul style="list-style-type: none"> • Understand the concept of Graphics. • Understand the concept of Transformation of Images. • Understand the concept of rendering related to surface. • Understand the Graphics Programming 			
Course Prerequisites: DSC 1 & 10+2 with Mathematics			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
I.	Introduction: Computer Graphics– Overview, Basic elements, Animation and Multimedia Applications, Pictures– Representation, Storage and Display, Visualization and Image Processing, RGB color model, Output/Display Devices– Cathode Ray Tube (CRT), Refreshing Display Devices– Raster scan display device– Pixel, Frame Buffer, Color Display, Random scan display device, Plotters, Printers, Digitizers, Tablets, Light Pen, 3D viewing devices, Active and Passive Graphic Devices, Software for Computer Graphics. Lines– Point Plotting Techniques, Points and Lines, Line drawing algorithms– Digital Differential Analyzer (DDA) algorithm, Bresenham’s algorithm, Circle and Ellipse drawing algorithms, Region filling algorithms– Boundary Seed Fill algorithm, Flood Fill algorithm.	12	
II.	Transformations: 2D and 3D Transformations– Translation, Rotation, Scaling and other transformations, Matrix Representation of Points, Homogeneous Coordinate System, 2D and 3D Viewing Transformations, Parallel and Perspective Projections, Clipping and Windowing, Line Clipping algorithms– Cohen-Sutherland Line Clipping algorithm, Cyrus-Beck Line Clipping algorithm.	12	
III.	Curves and Surfaces: Curve representation, Polygon representation methods, Bezier curves, Bezier surfaces, Spline representations, B-spline methods, Hidden Surface Removal– Z-buffer algorithm, Back face detection, Binary Space Partitioning (BSP) tree method, Scan Line Coherence algorithm, Hidden Line Elimination	12	
IV.	Surface Rendering: Illumination/Lighting and Shading Models, Surface Lighting Effects, Basic Lighting Models– Ambient Lighting, Diffuse Lighting, Specular Reflection Lighting Model (Phong Specular Reflection Model), combined effect of Ambient, Diffuse and/or Specular Reflection. Shading– Gouraud Shading, Phong Shading Model, Creating Shadowed Objects, Drawing Shadows, Rendering Texture.	12	
V.	Graphics Programming: Graphics Programming using OpenGL, Programming 2D Applications, The OpenGL API, Primitives and Attributes, Color, Viewing, Control Functions, Polygons and Recursion, The Three-Dimensional Gasket, Plotting Implicit Functions, Interaction, Input Devices, Clients and Servers, Display Lists.	12	
Textbooks: 1. F. S. Hill, “Computer Graphics Using OpenGL”, Pearson Education, 2nded., 2007.			

2. Donald D. Hearn, M. Pauline Baker, "Computer Graphics with OpenGL", Pearson Education, 3rd ed., 2004.
3. David Rogers, "Procedural Elements of Computer Graphics", McGraw Hill, 2nd ed., 2001.

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VIII	Course Type: DSE 13
Course Code:	Course Title: Web App Design and Development		Credits: L T P (3 0 1)
Course outcomes:			
<ul style="list-style-type: none"> • Develop interfaces for single page applications • Develop a complete client-side solutions using angular js • Develop a RESTful web service. • Apply form validations 			
Course Prerequisites: DSC 1 & DSC 5			
Course Outline:			
Units	Topics		No. of Lab/Lectures
I.	Introduction to React: Definition of React, React library, React Developer tools, Introduction to ES6, Declaring variables, Arrow Functions, Objects and Arrays, modules, Introduction to AJAX, Functions in AJAX Pure React: Page setup, virtual DOM, React Element, React DOM, Constructing Elements with Data, React Components, DOM Rendering, First React Application using Create React App, React with JSX, React Element as JSX Props, State and Component Tree: Property Validation, Validating Props with createClass, Default Props, ES6 Classes and stateless functional components, React state management, State within the component tree, state vs props, Forms in React		20
II.	Rest APIs: JSON: Introduction, Syntax, Data Types, Objects, Schema. REST API: Introduction, WRML, REST API Design, Identifier Design with URIs, Interaction Design with HTTP, Representation Design, Caching, Security.		10
III.	Angular.js.: Introduction to Angular: Angular architecture; introduction to components, component interaction and styles; templates, interpolation and directives; forms, user input, form validations; data binding and pipes; retrieving data using HTTP; Angular modules		15
Lab: Web App Design and Development			15
Textbooks:			
<ol style="list-style-type: none"> 1. D. Brad, B. Dayley and C. Dayley, Node. js, MongoDB and Angularjs Web Development: The definitive guide to using the MEAN stack to build web applications, 2nd edition, Addison-Wesley, 2018. 2. D. Herron, Node.js Web Development, 5th edition, Packt Publishing, 2020. 3. A. Banks and E. Porcello, Learning React: Functional Web Development with React and Redux, 1st edition, O'Reilly, 2017. 4. M. Masse, REST API – Design Rulebook, 1st edition, O'Reilly, 2011. 			

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VIII	Course Type: DSE 14/GE 9
Course Code:	Course Title: Cloud Computing		Credits: L T P (3 1 0)
Course outcomes: <ul style="list-style-type: none"> Understand the evolution of Cloud Computing and compare with traditional Computing Remember the key terminologies used in Cloud Computing and understand key concepts Describe virtualization architecture and implement the virtualization using open-source tools Identify the advantages and disadvantages of various cloud computing platforms and service models. Classify security and privacy issues in cloud computing. Apply various cloud services to understand elasticity, scalability and availability properties of Cloud services and also their usage towards web service deployments. 			
Course Prerequisites: DSC 6			
Course Outline:			
Units	Topics		No. of Lab/Lectures
I.	Overview of Computing Paradigm: History with overview of Computing Paradigm, Cluster Computing, Grid Computing, Distributed Computing, Utility Computing, Cloud Computing versus Traditional Computing		8
II.	Introduction to Cloud Computing: Introduction to Cloud Computing, Different Perspectives on Cloud Computing, Characteristics, Different Stakeholders in Cloud Computing, Cloud NIST Reference Architecture		8
III.	Service Level Agreements (SLAs), Total cost of ownership (TCO), Benefits and limitations of Cloud Computing, Open Challenges		8
IV.	Virtualization: Introduction & need of Virtualization, Definition & types of Hypervisors, Characteristics of Virtualized Environments, Virtualization and Cloud Computing, System calls & Ring Privileges, Machine Reference Architecture, Xen Hypervisor Architecture, Pros and Cons of Virtualization		8
V.	Cloud Computing Architecture: Traditional Computing Architecture-Client-Server Architecture, Peer to Peer Architecture, OpenStack-based Cloud Computing Architecture, Cloud Reference Architecture: Service Models Perspective-Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Deployment Models- Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud		10

VI.	Cloud Security: Introduction, Cloud Security Issues such as Application-level Security, Network Level Security, Data-level Security, Virtualization Security, Identity Management & Access Control	9
VII.	Case Studies: Implementation of Cloud Services: AWS Cloud Services, Google Cloud Services, Apply Cloud Services for Hosting the Website	9

Textbooks:

1. Raj Kumar Buyya, Mastering the Cloud Computing, MacGraw Hill Education (India), 2013
2. Tim Mather, SubraKumaraswamy, ShahedLatif: Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance
3. J.R. ("Vic") Winkler: Securing the Cloud
4. Haley Beard, Cloud Computing Best Practices for Managing and Measuring Processes for On-demand Computing, Applications and Data Centers in the Cloud with SLAs, Emereo Pty Limited, July 2008.

Reference Books:

1. Michael Miller, Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online, Que Publishing, August 2008.
2. David Chisnall, The Definitive Guide to Xen Hypervisor, Prentice Hall; Reprint edition (9 November 2007)

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VIII	Course Type: Dissertation 2
Course Code:	Course Title: Dissertation on Major		Credits: 6
Course outcomes:			
<ul style="list-style-type: none"> • Develop the ability to conduct thorough research, including literature review, problem formulation, and application of appropriate methodologies. • Demonstrate advanced technical skills and proficiency in relevant tools, technologies, and programming languages. • Enhance skills in presenting research findings and defending the dissertation through clear and coherent written and oral communication. • Apply ethical principles in research and cultivate critical thinking skills for evaluating results and making informed conclusions. 			
Course Prerequisites: DSE 8			
Course Outline:			
	Writing a sample Research Project/Dissertation/Research Paper on any given topic covering all the components of Research.		

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: DSC 7
Course Code:	Course Title: Theory of Computation	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand formal languages, grammars and Chomsky hierarchy. • Design regular grammar, DFA, NFA, Mealy and Moore machine, PDA, Turing machines. • Understand the notion of decidability and computability. 			
Course Prerequisites: DSC 1			
Course Outline:			

Units	Topics	No. of Lab/Lectures
VI.	Finite automaton (FA): Transition system, Acceptance by a finite automaton, Deterministic and non-deterministic automaton (DFA and NFA), Equivalence of DFA and NFA, Minimization of states in a finite automaton, Mealy and Moore machines.	12
VII.	Formal languages and grammar: - Chomsky's hierarchy, Regular grammar and regular expression, REs and FA, Closure Properties, Pumping lemma for regular sets.	12
VIII.	Push down automaton (PDA): Acceptance by PDA using final state or empty stack, Context free language, Deterministic CFL, Deterministic PDA, Context free grammar and derivation trees, Leftmost and rightmost derivation, Ambiguity in context free grammar, Pumping lemma for context free languages.	12
IX.	Turing machine (TM): TM as computable functions and accepters, Non-deterministic TM, type-0 grammar, Halting problem of a TM, Linear bounded automaton (LBA) model, Context sensitive languages and grammars.	12
X.	Unsolvable problems: Reduction techniques, Decidability– Post's correspondence problem (PCP), Rice's theorem, Decidability of membership, Emptiness and equivalence problems of languages, P, NP, NP-Completeness; Satisfiability and Cook's theorem.	12

Textbooks:

5. KLP Mishra, N. Chandrasekaran, "Theory of Computer Science (Automata, Languages and Computation)", PHI, 3rded.
6. Peter Linz, Jones, Bartlett, "An Introduction to Formal Languages and Automata", 5thed.
7. John E. Hopcroft, J.D. Ullman, Rajiv Motwani, "Introduction to Automata Theory, Languages and Computation", Pearson Education, 3rded.
8. Michael Sipser, "Introduction to the Theory of Computation", Cengage Learning, 3rded.

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: DSE 6
Course Code:	Course Title: Computer Networks	Credits: L T P (3 1 0)	
Course outcomes:			

- Understanding Network topologies and network architecture.
- Demonstrate a clear understanding of the different layers of network architecture.

Course Prerequisites: DSC 1

Course Outline:

Units	Topics	No. of Lab/Lectures
VI.	Introduction to Computer Networks and Data Communication: Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite. Analog and digital signal; data-rate limits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission; digital to analog modulation; multiplexing techniques– FDM, TDM; transmission media.	12
VII.	Networks Switching Techniques and Access mechanisms: Circuit switching; packet switching– connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.	12
VIII.	Data Link Layer: Error detection and error correction techniques; data-link control-framing and flow control; Error recovery protocols– stop and wait ARQ, go-back-n ARQ; CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks– repeaters, hubs, switches, bridges, router and gateways.	12
IX.	Network and Transport Layers: Routing; routing algorithms; Network layer protocol of Internet– IP protocol, Internet control protocols, Transport services, Transport protocols, Internet Transport Protocols– UDP, TCP.	12
X.	Application Layer: Client–server model, Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), Telnet, Network Virtual Terminal (NVT), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), User Agent (UA), Mail Transfer Agent (MTA), Multipurpose Internet Mail Extensions (MIME), Post Office Protocol (POP), Simple Network Management Protocol (SNMP), Hypertext Transfer Protocol (HTTP), Uniform Resource Locator (URL), World Wide Web (WWW).	12

Textbooks:

3. B.A. Forouzan, “Data Communications and Networking”, THM, 4th ed., 2007.
4. Andrew S. Tanenbaum, “Computer Networks”, PHI, 4th ed., 2003.

References:

2. J.F. Kurose, K.W. Ross, “Computer Networking: A Top-Down Approach”, Pearson, 6th ed., 2012.
3. Leon Garcia, Widjaja, “Communication Networks: Fundamental Concepts and Key Architectures”, Tata McGraw Hill, 2001.

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: DSE 7
Course Code:	Course Title: Discrete Mathematics	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand and interpret the fundamental mathematical structures like Set theory, Relation and Functions • Write recursive definitions of sequences and collections of objects • Understand the concepts and applications of vector algebra • Understand and interpret the basic concepts of Graph Theory • Apply the use of graph theory concepts solving various Computer Science and Engineering problems. 			
Course Prerequisites: Mathematics in 10+2			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
VI.	Logic and Proofs: Propositional Logic– Binary logic and propositions, Propositional Variables, Truth table, Logical connectives– Negation, Conjunction, Disjunction, Conditional, Biconditional, Universal connectives, Well-formed Formulas, Tautology, Contradiction and Contingency, Propositional Equivalences, Duality, Predicate Logic– Predicates, Quantifiers– Existential and Universal quantifier, Predicate formulas, Equivalence of formulas involving quantifiers, Normal forms– CNF/DNF, PCNF/PDNF, Normal forms for First Order Logic– Prenex Normal Form, Rules of Inference. Proof Techniques– Introduction to Proof, Definitions– Theorem, Lemma, Corollary and Conjecture, Methods of Proof– Direct Proofs, Indirect Proofs– Proof by Contraposition, Proof by Contradiction, Proof by Cases, Mathematical Induction.	12	
VII.	Counting and Recurrence: Set Theory, Countable and Countably Infinite Sets, Pigeonhole Principle, Permutation and Combination, Principle of Inclusion and Exclusion, Generating functions– Definition, Generating Permutations and Combinations. Recurrence– Recurrence Relations, Linear Recurrence Relations with constant coefficients and their solution, Solving Linear Recurrence Relations using Generating Functions.	12	
VIII.	Binary and Ordered Relations: Binary Relation, Properties of Binary Relations– Reflexive, Symmetric and Transitive Relation, Equivalence Relation, Closure of Relations– Reflexive, Symmetric and Transitive Closure, Warshall’s algorithm, Ordered Relation– Partial Order and Posets, Hasse diagram of Poset, Maximal, Minimal, Maximum and Minimum of poset, glb and lub, Isomorphic ordered set, Well ordered set, Lattice, Properties of lattice, Distributed and Complemented lattice, Applications of Lattice, Topological Sort.	12	
IX.	Graph Theory: Definition of Graph, Types– Directed and Undirected Graph, Complete Graph, Bipartite Graph, Multigraph, Weighted Graph, Graph Representation– Adjacency matrix and Adjacency list, Graph Isomorphism, Connectivity and Path, Euler and Hamiltonian Paths and Circuits, Shortest path– Dijkstra’s algorithm, Planar Graph, Euler’s theorem for Planar Graphs, Graph Coloring. Trees– Basic terminology and properties, Tree Traversal– Inorder, Preorder and Postorder, Expression Trees– Infix, Prefix and Postfix notations, Spanning Trees– Kruskal’s and Prim’s algorithms for Minimum Spanning Trees (MST).	12	

X.	Algebra: Definition and elementary properties of Semigroups, Monoids, Groups, Subgroups, Generators and Cyclic group, Permutation group, Cosets, Lagrange's Theorem, Rings, Integral Domains and Fields	12
<p>Textbooks:</p> <ol style="list-style-type: none"> 6. Kenneth H. Rosen, "Discrete Mathematics and Its Applications", Tata McGraw Hill, 7th ed., 2012. 7. C. L. Liu, "Elements of Discrete Mathematics", McGraw Hill, 2nd ed., 1986. 8. Bernard Kolman, Robert C. Busby, Sharon Cutler Ross, "Discrete Mathematical Structures", Pearson Education, 6th ed., 2008. 9. J. P. Tremblay, R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill, 1st ed., 2001. 10. Susanna S. Epp, "Discrete Mathematics with Applications", 4th ed., 2010. 		

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: DSE 8
Course Code:	Course Title: Research Methodology	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand the concept of research. • Understand the concept of data collection and selection for research. • Understand the applicability of research for public at large. 			
Course Prerequisites: Basic Knowledge of Computer			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
IV.	Introduction to Research Methods in science – Philosophy of Science, Research methods and Creative Thinking, Evolutionary Epistemology, Scientific Methods, Hypotheses Generation and Evaluation, Code of Research Ethics, Definition and Objectives of Research, Various Steps in Scientific Research, Research presentations Types of Research – Research Purposes – Research Design – Survey Research – formulation of scientific problems and hypotheses selection of methods for solving a scientific problem Case Study Research.	20	
V.	How to perform a literature review – Sampling Methods – Data Processing and Analysis strategies - Data Analysis with Statistical Packages – Statistical Analysis – Hypothesis-testing – Generalization and Interpretation.	20	
VI.	Research Reports - Structure and Components of Research Report, Types of Report, Layout of Research Report, Mechanism of writing a research report – Requirements of a good dissertation.	20	
Textbooks:			
<ol style="list-style-type: none"> 6. Oates, B.J., (2005). Researching Information Systems and Computing. Sage Publications, UK. 7. Zobel, J. (2004). Writing for Computer Science - The art of effective communication. 2nd ed., Springer, UK. 8. Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., 2002. An introduction to Research Methodology, RBSA Publishers. 9. Kothari, C.R., 1990. Research Methodology: Methods and Techniques. New Age International. 5 10. Sinha, S.C. and Dhiman, A.K., 2002. Research Methodology. 			

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VII	Course Type: DSE 9/GE 8
Course Code:	Course Title: Mobile App Design and Development	Credits: L T P (3 0 1)	
Course outcomes:			
<ul style="list-style-type: none"> Learn about mobile devices, Android OS, and architecture, and how to use Android Studio. Understand key components like Intents, Manifest files, and permissions. Explore activity life cycles, types of intents, and data sharing using intents. Develop skills in creating UI elements, handling events, using animations, notifications, and fragments. 			
Course Prerequisites: GE 4/DSE 2			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
VI.	Android Systems: Introduction to Mobile devices and applications, Open Handset Alliance (OHA), Overview of Android OS and architecture, installing android studio. Introduction to Android application components, Intents, Android Manifest File and its common settings, Using Intent Filter, Permissions Activities and intents: understanding activity and its life cycle, Types of intents, intent filter, context, data sharing using intent	12	
VII.	Android User Interface: Basic android UI, layouts, view and view attributes, button, controls. UI events and event listeners, animations, notifications, progress dialog, Action bar, toolbar, menus and pop ups, Tab based UI, Fragment, Types of Fragment, Fragment Lifecycle, communication between fragment and activity	12	
VIII.	Android Storage and APIs: Android storage: Using Android Data and Storage APIs, Managing data using SQLite, Sharing Data between Applications with Content Providers Android APIs: Multimedia, Using Android Networking APIs, Using Android Web APIs, Using Android Telephony APIs, android location based services	12	
IX.	iOS Technology Stack: Introduction to iOS technology stack: iOS architecture, StoryBoard, features of Xcode, components of iOS SDK. Introduction to swift: data types, variables, control flow and operators, Collections and functions in swift, classes and structures, inheritance, closure and enumerations	12	
X.	User interactions: Controls, gesture organizers, touching views, Core Location and Mapkit, using Google Maps in iOS. Sensors in iOS. Data persistence: Core Data framework for storing persistent data, CRUD operations.	12	

Lab: Mobile App Design and Development	15
Textbooks:	
<ol style="list-style-type: none"> 5. "Android Programming: The Big Nerd Ranch Guide" by Bill Phillips and Brian Hardy - Offers a comprehensive introduction to Android development. 6. "Android Application Development All-in-One For Dummies" by Barry Burd - Covers various aspects of Android app development, including UI design and application components. 7. "iOS Programming: The Big Nerd Ranch Guide" by Christian Keur and Aaron Hillegass - Provides a detailed guide to iOS development, including Swift programming and user interactions. 8. "Learning Swift: Building Apps for macOS, iOS, and Beyond" by Paris Buttfield-Addison, Jon Manning, and Tim Nugent - Focuses on Swift programming for iOS development. 	

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VII	Course Type: Dissertation 1
Course Code:	Course Title: Dissertation on Major		Credits: 6
Course outcomes:			
<ul style="list-style-type: none"> • Develop the ability to conduct thorough research, including literature review, problem formulation, and application of appropriate methodologies. • Demonstrate advanced technical skills and proficiency in relevant tools, technologies, and programming languages. • Enhance skills in presenting research findings and defending the dissertation through clear and coherent written and oral communication. • Apply ethical principles in research and cultivate critical thinking skills for evaluating results and making informed conclusions. 			
Course Prerequisites: DSE 8			
Course Outline:			
	Writing a sample Research Project/Dissertation/Research Paper on any given topic covering all the components of Research.		

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VIII	Course Type: DSC 8
Course Code:	Course Title: Compiler Design	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understanding of Compiler Architecture • Ability to Implement Lexical and Syntax Analysis: • Knowledge of Semantic Analysis and Intermediate Code Generation • Experience with Compiler Optimization and Code Generation 			
Course Prerequisites: Basic understanding of compilers and programming languages.			
Course Outline:			
Units	Topics		No. of Lab/Lectures
VI.	Compiler Structure: Analysis-synthesis model of compilation, various phases of a compiler, tool-based approach to compiler construction. Lexical analysis: Interface with input parser and symbol table, token, lexeme and patterns, difficulties in lexical analysis, error reporting and implementation. Regular grammar & language definition, Transition diagrams, Design of a typical scanner using LEX or Flex.		12
VII.	Syntax Analysis: Context free grammars, ambiguity, associability, precedence, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing LL(1) grammar, Bottom up parsing, operator precedence grammars, LR parsers (SLR, LALR, LR), Design of a typical parser using YACC or Bison.		12
VIII.	Syntax directed definitions: Inherited and synthesized attributes, dependency graph, evaluation order, bottom up and top down evaluation of attributes, L- and S-attributed		12

	definitions. Type checking: type system, type expressions, structural and name equivalence of types, type conversion, overloaded function and operators, polymorphic function. Run time system: storage organization, activation tree, activation record, parameter passing symbol table, dynamic storage allocation. Intermediate code generation: intermediate representation, translation of declarations, assignments, Intermediate Code generation for control flow, Boolean expressions and procedure calls, implementation issues.	
IX.	Code generation and instruction selection: Issues, basic blocks and flow graphs, register allocation, code generation, DAG representation of programs, code generation from DAGS, peep hole optimization, code generator, specification of machine.	12
X.	Code optimization: Source of optimizations, optimization of basic blocks, loops, global dataflow analysis, solution to iterative dataflow equations, code improving transformations, dealing with aliases, data flow analysis of structured flow graphs.	12

Textbooks:

1. K. C. Loudon, "Compiler Construction, Principle and Practice", Cengage Publication, 6th ed., 2009.
2. Alfred V. Aho, Ravi Sethi, Jeffrey, D. Ullman, "Compilers: Principles, Techniques and Tools", Pearson, 1998.
3. V. Raghvan, "Principles of Compiler Design", TMH, 2009.
4. Levine, Mason, Brown, "Lex & Yacc", O' Reilly, 1998.
5. S. S. Muchnick Harcourt Asra, "Advanced Compiler Design implementation", Morgan Kaufman, 2006.
6. Allen, "Modern Compiler Implementation in C", Cambridge University Press, 1997.
7. Vinu V. Das, "Compiler Design using FLEX and YACC", PHI, 2005.
8. Cooper, "Engineering a Compiler", Elsevier, 2005.
9. Alan I. Holub, "Compiler Design in C", PHI, 2009.
10. Fisher, "Crafting a Compiler in C", Pearson, 2005.
10. Fisher, "Crafting a Compiler in C", Pearson, 2005.

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VIII	Course Type: DSE 10
Course Code:	Course Title: Design & Analysis of Algorithm	Credits: L T P (3 0 1)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand concept of asymptotic analysis and perform complexity analysis of iterative and recursive algorithms. • Formulate and solve time complexity recurrence relations using various techniques. • Solve computational problems using various algorithmic paradigms like divide-and conquer, greedy, dynamic programming, backtracking, branch-and-bound. 			
Course Prerequisites: Basic understanding of algorithms and data structures			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
IV.	Introduction: Review of Asymptotic Notations, Mathematical analysis for Recursive and Non-recursive algorithms, solving recurrence relations.	10	
V.	Algorithm Design Techniques: Brute Force, Exhaustive Search, Divide and conquer, Merge sort, Quick sort, Binary search, Multiplication of Large Integers, Strassen's Matrix Multiplication. Greedy strategy– General Approach and problems like Optimal	20	

	Merge Patterns, Minimum Spanning Trees algorithms, Knapsack Problem, Huffman Code, Job sequencing with deadlines, single source shortest path. Dynamic Programming– General Approach, Memoization, Multistage Graph, Matrix-Chain Multiplication, Longest Common Subsequence, Knapsack Problem, Floyd Warshall algorithm, Optimal Binary Search Trees.	
VI.	Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP, NP-Hard and NP-Complete Problems, Intractability, Cook's Theorem, Reductions. Coping with the Limitations – Backtracking concept; Branch & Bound method, Approximation Algorithms.	15
Lab: Design & Analysis of Algorithm		15
Textbooks:		
1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Prentice Hall of India, 3 rd ed., 2010.		
2. R. C. T. Lee, S. S. Tseng, R. C. Chang, Y. T. Tsai, "Introduction to the Design and Analysis of Algorithms: A Strategic Approach" McGraw Hill, 2006.		
3. Anany Levitin, Introduction to the Design and Analysis of Algorithms, Pearson Education, 2007.		
4. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", University Press, 2 nd ed., 2008.		
5. Kenneth A. Berman, Jerome Paul, "Algorithms: Sequential, Parallel and Distributed", Cengage Learning, 2004.		
6. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "The Design and Analysis of Computer Algorithms" Pearson Education, 2008.		
7. Michael T. Goodrich, Roberto Tamassia, Algorithm Design, Wiley, 2002.		
8. S. Dasgupta, C. Papadimitriou, and U. Vazirani. Algorithms. McGraw-Hill Higher Education, 2006		

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VIII	Course Type: DSE 11
Course Code:	Course Title: Artificial Intelligence	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand the concept of Artificial Intelligence. • Understand to apply the knowledge and reasoning for different components. • Understand the Expert Systems and their uses. • Understand the basics of PROLOG. 			
Course Prerequisites: DSC 1			
Course Outline:			
Units	Topics		No. of Lab/Lectures

VI.	Introduction and applications of artificial intelligence, Problem solving: State space search, Production system, Problem characteristics, Problem system characteristics, Search techniques: Generate and test, Hill climbing, Best first search, A* algorithm, Problem reduction	12
VII.	Knowledge and Reasoning: Knowledge acquisition, Knowledge engineer, Cognitive behavior, Knowledge representation: Level of representation, Knowledge representation schemes, Formal logic, Inference Engine, Semantic net, Frame, Scripts. Adversarial search, Optimal and imperfect decisions, Alpha, Beta pruning, Logical agents: Propositional logic, First order logic – Syntax and semantics – Inference in first order logic. Uncertain Knowledge and Reasoning: Uncertainty – Acting under uncertainty – Basic probability notation – Axioms of probability – Baye’s rule – Probabilistic reasoning – Making simple decisions.	12
VIII.	Expert systems: Definition, Role of knowledge in expert system, Architecture, Expert System Development Life Cycle: Problem selection, Prototype construction, Formalization, Implementation, Evaluation	12
IX.	Planning and Learning: Planning: Planning problem – Partial order planning – Planning and acting in non-deterministic domains – Learning: Learning decision trees, Knowledge in learning, Neural networks, Reinforcement learning – Passive and active.	12
X.	PROLOG Programming: Introduction, variables, using rules, Input and Output predicates, Fail and cut predicates, Recursion, Arithmetic operation, Compound object, Dynamic database, Lists, String, File operations.	12

Textbooks:

1. Elaine Rich, Kevin Knight, “Artificial Intelligence”, Tata McGraw Hill.
2. Dan W. Patterson, “Introduction to Artificial Intelligence and Expert Systems”, Prentice Hall of India.
3. Nils J. Nilsson, “Principles of Artificial Intelligence”, Narosa Publication house.
4. Stuart Russell, Peter Norvig, “Artificial Intelligence: A Modern Approach”, Pearson Education, 2nded.
5. Winston, Patrick, Henry, “Artificial Intelligence”, Pearson Education.
6. Gopal Krishna, Janakiraman, “Artificial Intelligence”.

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4 th	Semester: VIII	Course Type: DSE 12
Course Code:	Course Title: Computer Graphics	Credits: L T P (3 1 0)	
Course outcomes:			
<ul style="list-style-type: none"> • Understand the concept of Graphics. • Understand the concept of Transformation of Images. • Understand the concept of rendering related to surface. 			

- Understand the Graphics Programming

Course Prerequisites: DSC 1 & 10+2 with Mathematics

Course Outline:

Units	Topics	No. of Lab/Lectures
VI.	Introduction: Computer Graphics– Overview, Basic elements, Animation and Multimedia Applications, Pictures– Representation, Storage and Display, Visualization and Image Processing, RGB color model, Output/Display Devices– Cathode Ray Tube (CRT), Refreshing Display Devices– Raster scan display device– Pixel, Frame Buffer, Color Display, Random scan display device, Plotters, Printers, Digitizers, Tablets, Light Pen, 3D viewing devices, Active and Passive Graphic Devices, Software for Computer Graphics. Lines– Point Plotting Techniques, Points and Lines, Line drawing algorithms– Digital Differential Analyzer (DDA) algorithm, Bresenham’s algorithm, Circle and Ellipse drawing algorithms, Region filling algorithms– Boundary Seed Fill algorithm, Flood Fill algorithm.	12
VII.	Transformations: 2D and 3D Transformations– Translation, Rotation, Scaling and other transformations, Matrix Representation of Points, Homogeneous Coordinate System, 2D and 3D Viewing Transformations, Parallel and Perspective Projections, Clipping and Windowing, Line Clipping algorithms– Cohen-Sutherland Line Clipping algorithm, Cyrus-Beck Line Clipping algorithm.	12
VIII.	Curves and Surfaces: Curve representation, Polygon representation methods, Bezier curves, Bezier surfaces, Spline representations, B-spline methods, Hidden Surface Removal– Z-buffer algorithm, Back face detection, Binary Space Partitioning (BSP) tree method, Scan Line Coherence algorithm, Hidden Line Elimination	12
IX.	Surface Rendering: Illumination/Lighting and Shading Models, Surface Lighting Effects, Basic Lighting Models– Ambient Lighting, Diffuse Lighting, Specular Reflection Lighting Model (Phong Specular Reflection Model), combined effect of Ambient, Diffuse and/or Specular Reflection. Shading– Gouraud Shading, Phong Shading Model, Creating Shadowed Objects, Drawing Shadows, Rendering Texture.	12
X.	Graphics Programming: Graphics Programming using OpenGL, Programming 2D Applications, The OpenGL API, Primitives and Attributes, Color, Viewing, Control Functions, Polygons and Recursion, The Three-Dimensional Gasket, Plotting Implicit Functions, Interaction, Input Devices, Clients and Servers, Display Lists.	12

Textbooks:

1. F. S. Hill, “Computer Graphics Using OpenGL”, Pearson Education, 2nded., 2007.
2. Donald D. Hearn, M. Pauline Baker, “Computer Graphics with OpenGL”, Pearson Education, 3rded., 2004.
3. David Rogers, “Procedural Elements of Computer Graphics”, McGraw Hill, 2nded., 2001.

Subject: Computer Science

Programme/Class: B.Sc. (H)

Year: 4th

Semester: VIII

Course Type: DSE 13

Course Code:		Course Title: Web App Design and Development	Credits: L T P (3 0 1)
Course outcomes:			
<ul style="list-style-type: none"> • Develop interfaces for single page applications • Develop a complete client-side solutions using angular js • Develop a RESTful web service. • Apply form validations 			
Course Prerequisites: DSC 1 & DSC 5			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
IV.	Introduction to React: Definition of React, React library, React Developer tools, Introduction to ES6, Declaring variables, Arrow Functions, Objects and Arrays, modules, Introduction to AJAX, Functions in AJAX Pure React: Page setup, virtual DOM, React Element, React DOM, Constructing Elements with Data, React Components, DOM Rendering, First React Application using Create React App, React with JSX, React Element as JSX Props, State and Component Tree: Property Validation, Validating Props with createClass, Default Props, ES6 Classes and stateless functional components, React state management, State within the component tree, state vs props, Forms in React	20	
V.	Rest APIs: JSON: Introduction, Syntax, Data Types, Objects, Schema. REST API: Introduction, WRML, REST API Design, Identifier Design with URIs, Interaction Design with HTTP, Representation Design, Caching, Security.	10	
VI.	Angular.js.: Introduction to Angular: Angular architecture; introduction to components, component interaction and styles; templates, interpolation and directives; forms, user input, form validations; data binding and pipes; retrieving data using HTTP; Angular modules	15	
Lab: Web App Design and Development			15
Textbooks:			
<ol style="list-style-type: none"> 1. D. Brad, B. Dayley and C. Dayley, Node. js, MongoDB and Angularjs Web Development: The definitive guide to using the MEAN stack to build web applications, 2nd edition, Addison-Wesley, 2018. 2. D. Herron, Node.js Web Development, 5th edition, Packt Publishing, 2020. 3. A. Banks and E. Porcello, Learning React: Functional Web Development with React and Redux, 1st edition, O'Reilly, 2017. 4. M. Masse, REST API – Design Rulebook, 1st edition, O'Reilly, 2011. 			

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VIII	Course Type: DSE 14/GE 9
Course Code:	Course Title: Cloud Computing		Credits: L T P (3 1 0)
Course outcomes:			
<ul style="list-style-type: none"> • Understand the evolution of Cloud Computing and compare with traditional Computing • Remember the key terminologies used in Cloud Computing and understand key concepts • Describe virtualization architecture and implement the virtualization using open-source tools • Identify the advantages and disadvantages of various cloud computing platforms and service models. • Classify security and privacy issues in cloud computing. • Apply various cloud services to understand elasticity, scalability and availability properties of Cloud services and also their usage towards web service deployments. 			
Course Prerequisites: DSC 6			
Course Outline:			
Units	Topics	No. of Lab/Lectures	
VIII.	Overview of Computing Paradigm: History with overview of Computing Paradigm, Cluster Computing, Grid Computing, Distributed Computing, Utility Computing, Cloud Computing versus Traditional Computing	8	
IX.	Introduction to Cloud Computing: Introduction to Cloud Computing, Different Perspectives on Cloud Computing, Characteristics, Different Stakeholders in Cloud Computing, Cloud NIST Reference Architecture	8	
X.	Service Level Agreements (SLAs), Total cost of ownership (TCO), Benefits and limitations of Cloud Computing, Open Challenges	8	
XI.	Virtualization: Introduction & need of Virtualization, Definition & types of Hypervisors, Characteristics of Virtualized Environments, Virtualization and Cloud Computing, System calls & Ring Privileges, Machine Reference Architecture, Xen Hypervisor Architecture, Pros and Cons of Virtualization	8	
XII.	Cloud Computing Architecture: Traditional Computing Architecture-Client-Server Architecture, Peer to Peer Architecture, OpenStack-based Cloud Computing Architecture, Cloud Reference Architecture: Service Models Perspective-Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Deployment Models- Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud	10	
XIII.	Cloud Security: Introduction, Cloud Security Issues such as Application-level Security, Network Level Security, Data-level Security, Virtualization Security, Identity Management & Access Control	9	
XIV.	Case Studies: Implementation of Cloud Services: AWS Cloud Services, Google Cloud Services, Apply Cloud Services for Hosting the Website	9	
Textbooks:			
<ol style="list-style-type: none"> 1. Raj Kumar Buyya, Mastering the Cloud Computing, MacGraw Hill Education (India), 2013 2. Tim Mather, SubraKumaraswamy, ShahedLatif: Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance 3. J.R. ("Vic") Winkler: Securing the Cloud 4. Haley Beard, Cloud Computing Best Practices for Managing and Measuring Processes for On-demand Computing, Applications and Data Centers in the Cloud with SLAs, Emereo Pty Limited, July 2008. Reference Books:			

Reference Books:

1. Michael Miller, Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online, Que Publishing, August 2008.
2. David Chisnall, The Definitive Guide to Xen Hypervisor, Prentice Hall; Reprint edition (9 November 2007)

Subject: Computer Science			
Programme/Class: B.Sc. (H)	Year: 4th	Semester: VIII	Course Type: Dissertation 2
Course Code:	Course Title: Dissertation on Major		Credits: 6
Course outcomes: <ul style="list-style-type: none">• Develop the ability to conduct thorough research, including literature review, problem formulation, and application of appropriate methodologies.• Demonstrate advanced technical skills and proficiency in relevant tools, technologies, and programming languages.• Enhance skills in presenting research findings and defending the dissertation through clear and coherent written and oral communication.• Apply ethical principles in research and cultivate critical thinking skills for evaluating results and making informed conclusions.			
Course Prerequisites: DSE 8			
Course Outline:			
	Writing a sample Research Project/Dissertation/Research Paper on any given topic covering all the components of Research.		