



III Year - I Semester	Code: 20CS5T09	L	T	P	C
		3	0	0	3
COMPUTER NETWORKS					
Course Outcomes: After completing this course the student must demonstrate the knowledge and ability to: <ol style="list-style-type: none">1. Independently understand basic computer network technology and identify the different types of network topologies and protocols.2. Understand Network models and Physical layer3. Study the techniques used in data link layer.4. Understand the routing strategies for an IP based networking5. Understand the Importance of MAC sub layer and addressing mechanism6. Understand the Importance of Application layer and the wireless web.					
UNIT I Introduction: OSI overview, TCP/IP and other networks models, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.					
UNIT II Physical Layer – Guided Transmission Media, Digital Modulation and Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Code Division Multiplexing.					
UNIT III The Data Link Layer - Services Provided to the Network Layer – Framing – Error Control Flow Control, Error Detection and Correction, Elementary Data Link Protocols, Sliding Window Protocols. Random Access: ALOHA, MAC addresses, carrier sense multiple access (CSMA), CSMA/CD, CSMA/CA, controlled Access: Reservation, Polling, Token Passing, channelization: FDMA, TDMA, CDMA.					
UNIT IV Network Layer: Virtual circuit and Datagram subnets-Routing algorithm shortest path routing, Flooding, Hierarchical routing, Broad cast, Multi cast, distance vector routing. Transport Layer: The transport service, elements of transport protocols, congestion control, the internet transport protocols.					
UNIT V Application Layer –The Domain Name System: The DNS Name Space, Resource Records, Name Servers, Electronic Mail: Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery, The wireless web: WAP—The Wireless Application Protocol					
Text Books: <ol style="list-style-type: none">1. Computer Networks—Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI2. Data Communications and Networks—Behrouz A. Forouzan. Third Edition TMH.					
Reference Books: <ol style="list-style-type: none">1. An Engineering Approach to Computer Networks- S. Keshav, 2nd Edition, Pearson Education2. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson3. Computer Networks, 5ed, David Patterson, Elsevier					



III Year - I Semester	Code: 20CS5T10	L	T	P	C
		3	0	0	3
FORMAL LANGUAGE AND AUTOMATA THEORY					
Course Objectives: <ul style="list-style-type: none"> ➤ To learn fundamentals of Regular and Context Free Grammars and Languages ➤ To understand the relation between Regular Language and Finite Automata and machines ➤ To learn how to design Automata's and machines as Acceptors, Verifiers and Translators ➤ To understand the relation between Contexts free Languages, PDA and TM ➤ To learn how to design PDA as acceptor and TM as Calculators 					
Course Outcomes: By the end of the course students can <ul style="list-style-type: none"> ➤ Classify machines by their power to recognize languages. ➤ Summarize language classes & grammars relationship among them with the help of Chomsky hierarchy ➤ Employ finite state machines to solve problems in computing 					
UNIT I Finite Automata: Need of Automata theory, Central Concepts of Automata Theory, Automation, Finite Automata, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with ϵ -Transitions, Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applications and Limitation of Finite Automata.					
UNIT II Regular Expressions, Regular Sets, Identity Rules, Equivalence of two RE, Manipulations of REs, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence between RG and FA, Inter Conversion.					
UNIT III Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, ϵ -Productions and Unit Productions, Normal Forms-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma, Closure Properties, Applications of Context Free Grammars.					
UNIT IV Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata.					
UNIT V Turing Machine: Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Types of TMs, Church's Thesis, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.					
Text Books: <ol style="list-style-type: none"> 1) Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008 2) Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3rd Edition, PHI, 2007 					
Reference Books: <ol style="list-style-type: none"> 1) Elements of Theory of Computation, Lewis H.P. & Papadimitriou C.H., Pearson /PHI 2) Theory of Computation, V. Kulkarni, Oxford University Press, 2013 3) Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill, 2014 					



III Year - I Semester	Code: 20CS5T11	L	T	P	C
		3	0	0	3
DATA BASE MANAGEMENT SYSTEM					
Course Objectives:					
1. To introduce about database management systems 2. To give a good formal foundation on the relational model 3. To introduce the concepts of basic SQL as a universal Database language					
Course Outcomes:					
1. Describe fundamental concepts a relational database 2. Create, maintain and manipulate a relational database using SQL 3. Apply Conceptual and Logical database design 4. Apply normalization for database design 5. Illustrate Storage management and Transaction management techniques.					
UNIT I					
Introduction: Database system, Characteristics (Database Vs File System), Database Users (Actors on Scene, Workers behind the scene), Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.					
UNIT II					
Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance. BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update), basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions (Date and Time, Numeric, String conversion).					
UNIT III					
Entity Relationship Model: Introduction, Basic features of ER model, Representation of entities, attributes, entity set, relationship, relationship set, constraints, ER diagrams Generalization/specialization and Aggregation. SQL: Creating tables with relationships, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, views(updatable and non-updatable), relational set operations.					
UNIT IV					
Schema Refinement (Normalization): Purpose of Normalization or schema refinement, concept of functional dependency, Closure of functional dependency and attribute closure, Normal forms based on functional dependency(1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form(BCNF), Lossless join and dependency preserving decomposition, Fourth normal form(4NF), Fifth Normal Form (5NF).					
UNIT V					
TRANSACTION MANAGEMENT Transaction, properties of transactions, transaction log, and transaction management with SQL using commit rollback and save point. Concurrency control for lost updates, uncommitted data, in consistent retrievals and the Scheduler. Concurrency control with locking methods: lock granularity, lock types, two phase locking for ensuring serializability, deadlocks, Concurrency control with time stamp ordering: Wait/Die and Wound/Wait Schemes.					
Text Books:					
1.Database System Concepts by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 7th Edition, McGraw-Hill Education, 2019. 2. Database Management Systems by Raghuram Ramakrishnan, Johannes Gehrke, 3rd Edition., McGraw-Hill Education (India), 2014.					
Reference Books:					
1.Database Principles: Fundamentals of Design, Implementation, and Management by Steven Morris, Keeley Crockett, Carlos Coronel, Craig Blewett, Cengage, 2020. 2. Fundamentals of Database Systems by RamezElmasri, Shamkant B. Navathe, 7th Edition, Pearson Education India, 2015. 3. Introduction to Database Systems by C J Date, 8th Edition, Pearson Education, 2009.					



III Year - I Semester	Code: 20CS5E01	L	T	P	C
		3	0	0	3
COMPUTER ORGANIZATION					
Course Objectives: The course objectives of Computer Organization are to discuss and make student familiar with the <ul style="list-style-type: none">➤ Principles and the Implementation of Computer Arithmetic➤ Operation of CPUs including RTL, ALU, Instruction Cycle and Busses➤ Fundamentals of different Instruction Set Architectures and their relationship to the CPU Design					
Course Outcomes: By the end of the course, the student will <ul style="list-style-type: none">➤ Develop a detailed understanding of computer systems➤ Cite different number systems, binary addition and subtraction, standard, floating-point, and micro operations➤ Develop a detailed understanding of architecture and functionality of central processing unit					
UNIT I Basic Structure of Computers: Basic Organization of Computers, Historical Perspective, Bus Structures, Data Representation: Data types, Complements, Fixed Point Representation. Floating, Point Representation. Other Binary Codes, Error Detection Codes. Computer Arithmetic: Addition and Subtraction, Multiplication Algorithms, Division Algorithms.					
UNIT II Register Transfer Language and Micro operations: Register Transfer language. Register Transfer Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro Operations, Shift Micro Operations, Arithmetic Logic Shift Unit. Basic Computer Organization and Design: Instruction Codes, Computer Register, Computer Instructions, Instruction Cycle, Memory – Reference Instructions. Input –Output and Interrupt, Complete Computer Description.					
UNIT III Central Processing Unit: General Register Organization, STACK Organization. Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer. Microprogrammed Control: Control Memory, Address Sequencing, Micro Program example, Design of Control Unit.					
UNIT IV Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory. Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct Memory Access.					
UNIT V Multi Processors: Introduction, Characteristics of Multiprocessors, Interconnection Structures, Inter Processor Arbitration. Pipeline: Parallel Processing, Pipelining, Instruction Pipeline, RISC Pipeline, Array Processor.					
Text Books: 1) Computer System Architecture, M. Morris Mano, Third Edition, Pearson, 2008. 2) Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5/e, McGraw Hill, 2002.					
Reference Books: 1) Computer Organization and Architecture, William Stallings, 6/e, Pearson, 2006. 2) Structured Computer Organization, Andrew S. Tanenbaum, 4/e, Pearson, 2005. 3) Fundamentals of Computer Organization and Design, Sivarama P. Dandamudi, Springer, 2006.					



III Year - I Semester	Code : 20CS5E02	L	T	P	C
		3	0	0	3
OBJECT ORIENTED PROGRAMMING					
Course Objectives: 1. Understanding the OOP's concepts, classes and objects, threads, files, applets, swings and act. 2. This course introduces computer programming using the JAVA programming language with object-oriented programming principles.					
Course Outcomes: 1. Understand Java programming concepts and utilize Java Graphical User Interface in Program writing. 2. Write, compile, execute and troubleshoot Java programming for networking concepts. 3. Build Java Application for distributed environment.					
UNIT I Introduction to OOP, procedural programming language and object oriented language, principles of OOP, applications of OOP, history of java, java features, JDK, JVM, program structure. Variables, primitive data types, identifiers, literals, operators, expressions, precedence rules and associativity, primitive type conversion and casting, flow of control.					
UNIT II Arrays, command line arguments, Classes and objects, class declaration, creating objects, methods, constructors and constructor overloading, garbage collector, importance of static keyword and examples, this keyword, nested classes.					
UNIT III Inheritance, types of inheritance, super keyword, final keyword, overriding and abstract class. Interfaces, creating the packages, using packages, importance of CLASSPATH and java.lang package. Exception handling, importance of try, catch, throw, throws and finally block, user defined exceptions, Assertions.					
UNIT IV Multithreading: introduction, thread life cycle, creation of threads, thread priorities, thread synchronization, communication between threads. Reading data from files and writing data to files, random access file					
UNIT V Introduction to Java FX, AWT: introduction, components and containers, Button, Label, Checkbox, Radio Buttons, List Boxes, Choice Boxes, Container class, Layouts, Menu and Scrollbar.					
Text Books: 1. The complete Reference Java, 8th edition, Herbert Schildt, TMH. 2. Programming in JAVA, Sachin Malhotra, SaurabhChoudary, Oxford. 3. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson.					
Reference Books: 1. Swing: Introduction, JFrame, JApplet, JPanel, Componets in Swings, Layout Managers in 2. Swings, JList and JScrollPane, Split Pane, JTabbedPane, JTree, JTable, Dialog Box.					



III Year - I Semester	Code: 20CS5D01	L	T	P	C
		3	0	0	3
DATA COMMUNICATION					
Course Objectives: <ul style="list-style-type: none"> To have a detailed study of various analog and digital modulation and demodulation techniques To have a thorough knowledge of various multiplexing schemes and Data communication protocols To know about the standards and mechanisms of television systems. 					
Course Outcomes: <ul style="list-style-type: none"> Knowledge of working of basic communication systems Ability to evaluate alternative models of communication system design 					
UNIT I INTRODUCTION TO DATA COMMUNICATIONS AND NETWORKING: Standards Organizations for Data Communications, Layered Network Architecture, Open Systems Interconnection, Data Communications Circuits, Serial and parallel Data Transmission, Data communications Networks, Alternate Protocol Suites, Signal Analysis, Electrical Noise and Signal-to-Noise Ratio, Analog Modulation Systems, Information Capacity, Bits, Bit Rate, Baud, and M- ary Encoding, Digital Modulation					
UNIT II METALLIC CABLE TRANSMISSION MEDIA: Metallic Transmission Lines, Transverse Electromagnetic Waves, Characteristics of Electromagnetic Waves.					
UNIT III DIGITAL TRANSMISSION: Pulse Modulation, Pulse code Modulation, Dynamic Range, Signal Voltage to-Quantization Noise Voltage Ratio, Linear Versus Nonlinear PCM Codes, Companding, PCM Line Speed, Delta Modulation PCM and Differential PCM. MULTIPLEXING AND T CARRIERS: Time- Division Multiplexing, T1 Digital Carrier System, Digital Line Encoding, T Carrier systems, Frequency- Division Multiplexing, Wavelength- Division Multiplexing, Synchronous Optical Network.					
UNIT IV WIRELESS COMMUNICATIONS SYSTEMS: Electromagnetic Polarization, Electromagnetic Radiation, Optical Properties of Radio Waves, Terrestrial Propagation of Electromagnetic Waves, Skip Distance, Free-Space Path Loss.					
UNIT V Data Communications Character Codes, Bar Codes, Error Control, Error Detection and Correction, Character Synchronization. Digital Service Unit and Channel Service Unit, Voice- Band Data Communication Modems, Bell Systems-Compatible Voice- Band Modems.					
Text Books: 1. Introduction to Data Communications and Networking, Wayne Tomasi, Pearson Education					
Reference Books: 1.Data Communications and Networking, Behrouz A Forouzan, Fourth Edition.TMH. 2.Data and Computer communications, 8/e, William Stallings, PHI. 3.Computer Communications and Networking Technologies, Gallow, Second Edition.					



III Year - I Semester	Code: 20CS5D02	L	T	P	C
		3	0	0	3
PRINCIPLES OF PROGRAMMING LANGUAGES					
Course Objectives: <ul style="list-style-type: none">➤ To understand and describe syntax and semantics of programming languages➤ To understand data, data types, and basic statements➤ To understand call-return architecture and ways of implementing them					
Course Outcomes: <ul style="list-style-type: none">➤ Describe the syntax and semantics of programming languages and gain practical knowledge in lexical analysis and parsing phases of a compiler➤ Make use of different constructs in programming languages with merits and demerits➤ Design and implement sub programs in various programming languages					
UNIT I Syntax and semantics: Evolution of programming languages, describing syntax, context, free grammars, attribute grammars, describing semantics, lexical analysis, parsing, recursive - decent bottom - up parsing.					
UNIT II Data, data types, and basic statements: Names, variables, binding, type checking, scope, scope rules, lifetime and garbage collection, primitive data types, strings, array types, associative arrays, record types, union types, pointers and references, Arithmetic expressions, overloaded operators, type conversions, relational and Boolean expressions, assignment statements, mixed mode assignments, control structures – selection, iterations, branching, guarded Statements.					
UNIT III Subprograms and implementations: Subprograms, design issues, local referencing, parameter passing, overloaded methods, generic methods, design issues for functions, semantics of call and return, implementing simple subprograms, stack and dynamic local variables, nested subprograms, blocks, dynamic scoping.					
UNIT IV Object- orientation, concurrency, and event handling: Object – orientation, design issues for OOP languages, implementation of object, oriented constructs, concurrency, semaphores, Monitors, message passing, threads, statement level concurrency, exception handling, event handling.					
UNIT V Functional programming languages: Introduction to lambda calculus, fundamentals of functional programming languages, Programming with Scheme, Programming with ML Logic programming languages: Introduction to logic and logic programming, Programming with Prolog, multi - paradigm languages.					
Text Books: <ol style="list-style-type: none">1) Robert W. Sebesta, “Concepts of Programming Languages”, Tenth Edition, Addison Wesley, 2012.2) Programming Languages, Principles & Paradigms, 2ed, Allen B Tucker, Robert E Noonan, TMH.					
Reference Books: <ol style="list-style-type: none">1) R. Kent Dybvig, “The Scheme programming language”, Fourth Edition, MIT Press, 2009.2) Jeffrey D. Ullman, “Elements of ML programming”, Second Edition, Prentice Hall, 1998.3) Richard A. O’Keefe, “The craft of Prolog”, MIT Press, 2009.4) W. F. Clocksin and C. S. Mellish, “Programming in Prolog: Using the ISO Standard”, Fifth Edition, Springer, 2003.					



III Year - I Semester	Code: 20CS5D03	L	T	P	C
		3	0	0	3
SOFTWARE ENGINEERING					
<p>Course Objectives: This course is designed to:</p> <ul style="list-style-type: none"> ✓ Give exposure to phases of Software Development, common process models including Waterfall, and the Unified Process, and hands-on experience with elements of the agile process ✓ Give exposure to a variety of Software Engineering practices such as requirements analysis and specification, code analysis, code debugging, testing, traceability, and version control ✓ Give exposure to Software Design techniques 					
<p>Course Outcomes: Students taking this subject will gain software engineering skills in the following areas:</p> <ul style="list-style-type: none"> ✓ Ability to transform an Object-Oriented Design into high quality, executable code ✓ Skills to design, implement, and execute test cases at the Unit and Integration level ✓ Compare conventional and agile software methods 					
<p>UNIT I The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths. A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology.</p>					
<p>UNIT II Agility, Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, A Tool Set for the Agile Process, Software Engineering Knowledge, Core Principles, Principles That Guide Each Framework Activity, Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.</p>					
<p>UNIT III Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling, Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model, Patterns for Requirements Modelling, Requirements Modeling for WebApps.</p>					
<p>UNIT IV Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model, Software Architecture, Architectural Genres, Architectural Styles, Assessing Alternative Architectural Designs, Architectural Mapping Using Data Flow, Components, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps, Designing Traditional Components, Component-Based Development.</p>					
<p>UNIT V The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evaluation, Elements of Software Quality Assurance, SQA Tasks, Goals & Metrics, Statistical SQA, Software Reliability, A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Test Strategies for WebApps, Validation Testing, System Testing, The Art of Debugging, Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing.</p>					
<p>Text Books: 1) Software Engineering a practitioner's approach, Roger S. Pressman, Seventh Edition, McGraw Hill Higher Education. 2) Software Engineering, Ian Sommerville, Ninth Edition, Pearson.</p>					
<p>Reference Books: 1) Software Engineering, A Precise Approach, PankajJalote, Wiley India, 2010. 2) Software Engineering, Ugrasen Suman, Cengage.</p>					



III Year - I Semester	Code: 20CS5D04	L	T	P	C
		3	0	0	3
ADVANCED COMPUTER ARCHITECTURE					
Course Outcomes: Gain knowledge of 1. Computational models and Computer Architectures. 2. Concepts of parallel computer models. 3. Scalable Architectures, Pipelining, Superscalar processors, multiprocessors					
UNIT I Theory of Parallelism, Parallel computer models, The State of Computing, Multiprocessors and Multicomputers, Multivector and SIMD Computers, PRAM and VLSI models, Architectural development tracks, Program and network properties, Conditions of parallelism.					
UNIT II Principals of Scalable performance, Performance metrics and measures, Parallel Processing applications, Speed up performance laws, Scalability Analysis and Approaches, Hardware Technologies, Processes and Memory Hierarchy, Advanced Processor Technology.					
UNIT III Bus Cache and Shared memory, Backplane bus systems, Cache Memory organizations, Shared- Memory Organizations, Sequential and weak consistency models, Pipelining and superscalar techniques, Linear Pipeline Processors, Non-Linear Pipeline Processors, Instruction Pipeline design, Arithmetic pipeline design, superscalar pipeline design.					
UNIT IV Parallel and Scalable Architectures, Multiprocessors and Multicomputers, Multiprocessor system interconnects, cache coherence and synchronization mechanism, Three Generations of Multicomputers, Message-passing Mechanisms, Multivector and SIMD computers, Vector Processing Principals, Multivector Multiprocessors, Compound Vector processing, SIMD computer Organizations.					
UNIT V Scalable, Multithreaded and Dataflow Architectures, Latency-hiding techniques, Principals of Multithreading, Fine-Grain Multicomputers, Scalable and multithreaded Architectures, Dataflow and hybrid Architectures.					
Text Books: 1. Advanced Computer Architecture Second Edition, Kai Hwang, Tata McGraw Hill.					
Reference Books: 1. Computer Architecture, Fourth edition, J. L. Hennessy and D.A. Patterson. ELSEVIER. 2. Advanced Computer Architecture, D. Sima, T. Fountain, P. Kacsuk, Pearson education 3. Computer Architecture, B. Parhami, Oxford Univ. Press					



III Year - I Semester	Code: 20CS5L11	L	T	P	C
		3	0	0	3
COMPUTER NETWORKS LAB					
Course Objectives: <ul style="list-style-type: none">✓ Understand and apply different network commands✓ Analyze different networking functions and features for implementing optimal solutions Apply different networking concepts for implementing network solution✓ Implement different network protocols					
Course Outcomes: <ul style="list-style-type: none">✓ Apply the basics of Physical layer in real time applications✓ Apply data link layer concepts, design issues, and protocols✓ Apply Network layer routing protocols and IP addressing✓ Implement the functions of Application layer and Presentation layer paradigms and Protocols					
Experiments: <ol style="list-style-type: none">1) Implement the data link layer framing methods such as character stuffing and bit stuffing.2) Write a C program to develop a DNS client server to resolve the given hostname.3) Implement on a data set of characters the three CRC polynomials – CRC-12, CRC-16 and CRC-CCIP.4) Implement Dijkstra’s algorithm to compute the Shortest path in a graph.5) Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm6) Take an example subnet of hosts. Obtain broadcast tree for it.7) Write a client-server application for chat using UDP8) Implement programs using raw sockets (like packet capturing and filtering)9) Write a C program to perform sliding window protocol.10) Get the MAC or Physical address of the system using Address Resolution Protocol.11) Simulate the OPEN SHORTEST PATH FIRST routing protocol based on the cost assigned to the path.					



III Year - I Semester	Code : 20CS5L12	L	T	P	C
		0	0	3	1.5
DBMS LAB					
Course Objectives:					
<ol style="list-style-type: none"> 1. Populate and query a database using SQL DDL/DML Commands 2. Declare and enforce integrity constraints on a database 3. Writing Queries using advanced concepts of SQL 4. Programming PL/SQL including procedures, functions, cursors and triggers 					
Course Outcomes:					
<ol style="list-style-type: none"> 1. Utilize SQL to execute queries for creating database and performing data manipulation operations 2. Apply Queries using SQL 3. Build PL/SQL programs including stored procedures, functions, cursors and triggers 					
Experiments					
<ol style="list-style-type: none"> 1 Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command. 2 Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class. 3 Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views. 4 Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date) 5 i. Create a simple PL/SQL program which includes declaration section, executable section and exception – Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found) ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block. 6 Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions. 7 Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, User defined Exceptions, RAISE APPLICATION ERROR. 8 Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES. 9 Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions. 10 Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables. 11 Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers 					



III Year - I Semester	Code: 20CS5S03	L	T	P	C
		1	0	2	2
SKILL ADVANCED COURSE :APP DEVELOPMENT					
<p>Course Outcomes: A student will be able</p> <ul style="list-style-type: none"> ➤ Understand android application development platform ➤ Explore Android architecture and ADT. ➤ Illustrate user interface components. 					
<p>UNIT I</p> <p>Introduction to Android App, Tools/environment setup & troubleshooting, Create 1st Android App – HelloWorld. Features & Architecture of Android, Android Devices in the Market, The Android Developer Community, Obtaining the Required Tools , Android SDK, Installing the Android SDK Tools, Configuring the Android SDK Manager. Eclipse, Android Development Tools (ADT), Creating Android Virtual Devices (AVDs) , Creating Your First Android Application , Anatomy of an Android Application</p>					
<p>UNIT II</p> <p>Understanding Activities , Applying Styles and Themes to an Activity , Hiding the Activity Title, Displaying a Dialog Window , Displaying a Progress Dialog , Displaying a More Sophisticated Progress Dialog , Linking Activities Using Intents , Resolving Intent Filter Collision , Returning Results from an Intent , Passing Data Using an Intent Object , Fragments</p>					
<p>UNIT III</p> <p>Working with the User Interface, Development tools, Applications App manifest, Resources, Application types, Activities, Activity Life Cycle Introduction to UI, Layouts, Fragments, Adapters, Action bar, Dialogs, Notifications UI best practices</p>					
<p>UNIT IV</p> <p>Intents and Broadcast Receivers, Intents, Pending intents, Intent resolution, Native broadcast intents, Preferences and saving state, Content Providers, Background processing Services, Intent Service, AsyncTask, Alarms</p> <p>Creating Your Own Services, Performing Long-Running Tasks in a Service, Performing Repeated Tasks in a Service Executing Asynchronous Tasks on Separate Threads Using Intent Service , Establishing Communication between a Service and an Activity Preparing for Publishing , Versioning Your Application , Digitally Signing Your Android Applications, Deploying APK Files , Using the adb.exe Tool, Using a Web Server , Publishing on the Android Market</p>					
<p>UNIT V</p> <p>Displaying Maps , Creating the Project, Obtaining the Maps API Key, Displaying the Map, Displaying the Zoom Control, Changing Views , Navigating to a Specific Location, Adding Markers , Getting the Location That Was Touched , Geo-coding and Reverse Geo-coding , Getting Location Data , Monitoring a Location.</p>					
<p>Text Books:</p> <ol style="list-style-type: none"> 1.Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 201 2.Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox) , 2012 					
<p>Reference Books:</p> <ol style="list-style-type: none"> 1.Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013 					



BONAM VENKATA CHALAMAYYA ENGINEERING COLLEGE ODALAREVU – 533

210, Andhra Pradesh, India

FOREIGN LANGUAGE-(GERMAN LANGUAGE)

(Course Code: 20HS5M04)

Unit-1

Introducing oneself and others; Grammar: W questions, personal pronouns, simple sentence, verb conjugation, hobbies, the week, numbers, the alphabet, months, seasons.

Unit-II

Grammar: articles, plural, the verbs to have and to be in the city/naming places and buildings, means of transport, basic directions/ Grammar: definite and indefinite articles; negation – kein and nicht; imperative.

Unit-III

Food, drink, family/groceries and meals/Grammar: the accusative everyday life, telling time, making appointments/Grammar: prepositions am, um, von bis, modal verbs, possessive articles.

Unit-IV

Leisure activity, celebrations/Grammar: separable verbs, the accusative, past tense of to have and to be
Contacts, writing letters/Grammar: dative my apartment, rooms, furniture, colours/Grammar: changing propositions Professions.

Unit-V

Clothes/Grammar: perfect tense and dative Health and the body / Grammar, the imperative and modal verbs
Holiday and weather.

Text book:

1. NETZWERK Deutsch als Fremdsprache A1(Goyal, New Delhi, 2015)