RAMAIAH
Institute of Technology

CURRICULUM

for the Academic year 2019 – 2020

COMPUTER SCIENCE AND ENGINEERING

V & VI SEMESTER B.E

RAMAIAH INSTITUTE OF TECHNOLOGY
(Autonomous Institute, Affiliated to VTU)
Bangalore – 560054.

About the Institute:

Ramaiah Institute of Technology (RIT) (formerly known as M.S.Ramaiah Institute of Technology) is a self-financing institution established in Bangalore in the year 1962 by the industrialist and philanthropist, Late Dr. M S Ramaiah. The institute is accredited with "A" grade by NAAC in 2014 and all engineering departments offering bachelor degree programs have been accredited by NBA. RIT is one of the few institutes with prescribed faculty student ratio and achieves excellent academic results. The institute was a participant of the Technical Education Quality Improvement Program (TEQIP), an initiative of the Government of India. All the departments have competent faculty, with 100% of them being postgraduates or doctorates. Some of the distinguished features of RIT are: State of the art laboratories, individual computing facility to all faculty members. All research departments are active with sponsored projects and more than 304 scholars are pursuing PhD. The Centre for Advanced Training and Continuing Education (CATCE), and Entrepreneurship Development Cell (EDC) have been set up on campus. RIT has a strong Placement and Training department with a committed team, a good Mentoring/Proctorial system, a fully equipped Sports department, large air-conditioned library with over 1,35,427 books with subscription to more than 300 International and National Journals. The Digital Library subscribes to several online e-journals like IEEE, JET etc. RIT is a member of DELNET, and AICTE INDEST Consortium. RIT has a modern auditorium, several hi-tech conference halls and all are air-conditioned with video conferencing facilities. It has excellent hostel facilities for boys and girls. RIT Alumni have distinguished themselves by occupying high positions in India and abroad and are in touch with the institute through an active Alumni Association. RIT obtained Academic Autonomy for all its UG and PG programs in the year 2007. As per the National Institutional Ranking Framework, MHRD, Government of India, Ramaiah Institute of Technology has achieved 64th rank in 2019 among the top 100 engineering colleges across India.

About the Department:

Year of Establishment	1984
Names of the Programmes offered	1. UG: B.E. in Computer Science and Engineering 2. PG: M.Tech. in Computer Science and Engineering 3. PG: M.Tech. in Computer Networks and Engineering 4. Ph.D 5. M.Sc(Engg.) by Research

The Department of Computer Science and Engineering (CSE) has eminent emeritus professors, 15 faculty with the doctorate degree and 15 pursuing the doctoral studies. The faculty has been publishing research papers in refereed journals and in conference proceedings. The department also conducts vocational courses and proficiency courses on fundamental and new programming languages and computer science concepts. These courses are conducted beyond college hours/summer semester by the faculty of the department. Some of the faculty are involved in institutional level activities and actively involved in interdisciplinary research activities. The department has state of the art laboratories like SAP, IBM Centre of Excellence and Cuda learning center. Technical seminars, workshops and hackathons are conducted regularly for UG & PG students. The department encourages the students to conduct and participate in extra-curricular/sports activities. The alumni network is very active and regular meeting are conducted by the department. The department is accredited by Nation Board of Accreditation (NBA). The department has MoUs with leading IT Industries like NVIDIA, SAP, IBM and HP. The department conducts subjects with more of hands-on sessions and encourages students to take up MOOC based online courses in NPTEL, IIT BombayX, Coursera, Udacity and edX.

VISION OF THE INSTITUTE

To be an Institution of International Eminence, renowned for imparting quality technical education, cutting edge research and innovation to meet global socio economic needs

MISSION OF THE INSTITUTE

MSRIT shall meet the global socio-economic needs through

1. Imparting quality technical education by nurturing a conducive learning environment through continuous improvement and customization
2. Establishing research clusters in emerging areas in collaboration with globally reputed organizations
3. Establishing innovative skills development, techno-entrepreneurial activities and consultancy for socio-economic needs

QUALITY POLICY

We at M. S. Ramaiah Institute of Technology strive to deliver comprehensive, continually enhanced, global quality technical and management education through an established Quality Management System complemented by the synergistic interaction of the stake holders concerned

VISION OF THE DEPARTMENT

To build a strong learning and research environment in the field of Computer Science and Engineering that promotes innovation towards betterment of the society

MISSION OF THE DEPARTMENT

1. To produce Computer Science graduates who, trained in design and implementation of computational systems through competitive curriculum and research in collaboration with industry and research organizations.

2. To educate students in technology competencies by providing professionally committed faculty and staff.

3. To inculcate strong ethical values, leadership abilities and research capabilities in the minds of students so as to work towards the progress of the society.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs):

A B.E (Computer Science & Engineering) graduate of Ramaiah Institute of Technology should, within three to five years of graduation

PEO1 Pursue a successful career in the field of Computer Science & Engineering or a related field utilizing his/her education and contribute to the profession as an excellent employee, or as an entrepreneur

PEO2 Be aware of the developments in the field of Computer Science & Engineering, continuously enhance their knowledge informally or by pursuing doctoral studies and engage in research and inquiry leading to new innovations and products

PEO3 Be able to work effectively in multidisciplinary and multicultural environments and Be responsible members and leaders of their communities

PROGRAMME OUTCOMES (POs):

The Outcomes of the Bachelor of engineering in Computer Science & Engineering Programme are as follows:

Engineering Graduates must be able to:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments

.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAMME SPECIFIC OUTCOMES (PSOs):

PSO1: understand the principles, architecture and organization of computers, embedded systems and computer networks.

PSO2: Apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems that include both hardware and software

PSO3: Apply software design and development practices to develop software applications in emerging areas such as IoT, Data Analytics, Social Networks, Cloud and High Performance Computing.

Curriculum Course Credits Distribution

Course Components	Humanities & Social Sciences (HSS)	Basic Sciences/ Lab (BS)	Engineering Sciences / Lab (ES)	Professional Courses - Core (Hard core, soft core, Lab) (PC-C)	Professional Courses- Electives (PC-E)	Other Electives (OE)	Project Work/ Internship (PW /IN)	Extra & Co-curricular activities (EAC)	Total credits in a semester
Semester									
First	2	9	14	0	0	0	0	0	25
Second	2	9	12	0	0	0	0	0	23
Third	0	4	0	21	0	0	0	0	25
Fourth	0	4	0	21	0	0	0	0	25
Fifth	2	0	0	19	4	0	0	0	25
Sixth	0	0	0	15	4	0	6	0	25
Seventh	0	0	0	14	12	0	0	0	26
Eighth	0	0	0	0	0	4	20	2	26
Total	**6**	**23**	**26**	**90**	**20**	**4**	**26**	**2**	**200**

SCHEME OF TEACHING

V SEMESTER

SI. No	Course Code	Course Name	Category	Credits					Contact Hours
				L	T	P	S*	Total	
1.	CS51	Operating Systems	PC-C	3	0	0	1	4	42
2.	CS52	Database Systems	PC-C	3	1	0	0	4	42+28
3.	CS53	Computer Networks	PC-C	4	0	0	0	4	56
4.	CS54	Java Programming	PC-C	3	0	0	1	4	42
5.	CS55	Intellectual Property Rights	HSS	2	0	0	0	2	28
6.	CSExx	--	PC-E	*	*	*	*	4	--
7.	CSL57	Java Programming Laboratory	PC-C	0	0	1	0	1	28
8.	CSL58	Database Systems Laboratory	PC-C	0	0	1	0	1	28
9.	CSL59	Computer Networks Laboratory	PC-C	0	0	1	0	1	28
Total				**19**	**1**	**3**	**2**	**25**	**--**

Professional Elective List for V Semester

SI. No	Course code	Course Name
1.	CSE01	Data Mining (3:0:0:1)
2.	CSE02	Artificial Intelligence (3:0:0:1)
3.	CSE03	Operation Research (4:0:0:0)
4.	CSE04	Advanced Algorithms (3:0:1:0)
5.	CSE05	System Simulation (4:0:0:0)
6.	CSE06	Object Oriented Modelling and Design (3:0:1:0)

VI SEMESTER

SI. No	Course Code	Course Name	Category	Credits					Contact Hours
				L	T	P	S*	Total	
1.	CS61	Compiler Design	PC-C	3	1	0	0	4	42+28
2.	CS62	Software Engineering	PC-C	3	0	0	1	4	42
3.	CS63	Unix System Programming and Web Technologies	PC-C	3	1	0	0	4	42+28
4.	CS64	Mini Project-Themes: Software Development for Portable Device/Rich Internet Applications/Embedded Systems	PW	0	0	4	2	6	56
5.	CSExx	--	PC-E	*	*	*	*	4	--
6.	CSL66	Unix System Programming & Compiler Design Laboratory	PC-C	0	0	1	0	1	28
7.	CSL67	IoT / Embedded Systems Laboratory	PC-C	0	0	1	0	1	28
8.	CSL68	Web Technologies Laboratory	PC-C	0	0	1	0	1	28
Total				**13**	**2**	**7**	**3**	**25**	**--**

Professional Elective List for VI Semester

SI. No	Course Code	Course Name
1.	CSE07	Mobile Computing (3:0:1:0)
2.	CSE08	Computer Graphics and Visualization (3:0:1:0)
3.	CSE09	Software Defined Networks (3:0:0:1)
4.	CSE10	Soft Computing (4:0:0:0)
5.	CSE11	Machine Learning (3:0:1:0)
6.	CSE12	Natural Language Processing (3:0:1:0)
7.	CSE13	Information Retrieval (3:0:1:0)

- ***S-Self-Study Component:** The topics are included in the syllabus and are considered for evaluation during all the CIE and SEE of that particular subject
- Two credits are allocated to Extra-curricular and Co-curricular Activities(EAC) which will be evaluated during the 8th Semester considering the achievements of a student from semester 1 to 8

Operating Systems

Course Code: CS51 **Credits: 3:0:0:1**
Prerequisites: Nil **Contact Hours: 42L**
Course Coordinator/s: Vandana Sardar

Course Contents:

Unit I

Introduction to operating systems: Operating systems objectives and functions, **Virtualization**: Background, The abstraction: The process, process API, process creation, process states, data structure, programs for process creation and termination, **Mechanism**: Limited direct execution , basic technique, problem I restricted operations, problem II switching between processes, concurrency, **Scheduling:** Introduction, workload assumptions, scheduling metrics, scheduling algorithms: FIFO, SJF, STFC and RR, response time, Incorporating I/O.
Self-Study: Mechanism: problem II switching between processes.

Unit II

Multilevel feedback queue: Introduction, MLFQ rules, changing priority, priority boost, tuning MLFQ and other issues, **Advanced Multiprocessor scheduling:** background, synchronization, cache affinity, single queue scheduling, multi queue scheduling, Linux multiprocessor schedulers, **Memory virtualization:** Abstraction of address space, early systems, multiprogramming and time sharing , the address space, Fragmentation, Paging: Overview, Basic method, address translation. Storing a page table, organizing page table, slow paging, memory trace, and advanced page table: Bigger pages, paging and segments, Multi-level page tables, inverted page tables.
Self-Study: Linux multiprocessor schedulers, Multilevel page tables, inverted page tables.

Unit III

Segmentation: Generalized base/bounds, accessing a segment, stack, segment sharing, fine-grained VS coarse-grained segmentation, **TLB**: basic algorithm, example: accessing an array handling TLB miss, TLB contents, Context switches Replacement policy, Real TLB entry, Swapping mechanism: Swap space, present-bit, page fault, page replacement policy, page fault control flow, when to use replacement, Swapping policies: Cache management, optimal replacement policies, FIFO policy, Random policy, using history LRU, Thrashing.
Self-Study: Real TLB entry & Thrashing.

Unit IV

Concurrency: An introduction, threads usage, process creation, shared data, uncontrolled scheduling, atomicity, **Thread API**: process creation (in detail), process completion, locks, condition variables, compiling and running, Semaphore: A Definition, Binary Semaphores (Locks), Semaphores as Condition Variables, The Producer/Consumer (Bounded Buffer) Problem, Reader-Writer Locks, The Dining Philosophers, implementing semaphores. Deadlock: System Model. Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, recovery from deadlock.
Self-Study: Dining Philosophers, Deadlock detection & Recovery from deadlock.

Unit V

File system implementation: Overall organization, the inode, directory organization, free space management, Access paths: reading and writing, caching and buffering, **Fast file systems:** Locality and fast file systems, poor performance, Fast file system, cylinder group, allocating files and directories, measuring file locality, FSCK and journaling: Introduction, A detailed example, Hard disk drives: A simple disk drive, disk scheduling.
Self-Study t:-FSCK and journaling: Introduction, A detailed example &a simple disk drive.

Scheme for Self-Study evaluation:

- The topics are integral part of the course.
- No formal lectures will be held for the self-study topics.
- The course co-ordinator may provide reading materials for self-study topics (optional).
- The topics prescribed under self-study in curriculum are part of CIE and SEE.

Text Books:

1. Operating systems: Three easy pieces, by Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau
   URL: http://pages.cs.wisc.edu/~remzi/OSTEP/
2. Operating System Concepts 8th edition by Abraham Silberschatz Peter B. Galvin Greg Gagne

Reference Books:

1. D.M Dhamdhere: Operating systems - A concept based Approach, 3rd Edition, Tata McGraw- Hill, 2012.
2. P.C.P. Bhatt: Introduction to Operating Systems Concepts and Practice, 3rd Edition, PHI, 2010.
3. Harvey M Deital: Operating systems, 3rd Edition, Pearson Education, 2011.

Course Outcomes (COs):

At the end of the course, the student must be able to:

1. Summarize the objectives of operating system and its various functions(PO-1,2,3,5,9,10,PSO-1,2)
2. Examine different process scheduling algorithms and measure their performance(PO-1,2,3,4,5,9,10, PSO-2)
3. Illustrate various memory management schemes such as paging, segmentation and swapping mechanism(PO-1,2,3,4,5,10, PSO-2)
4. Assess different concurrency control techniques to provide co-ordination among processes(PO-1,2,3,4,5,10, PSO-2)
5. Describe the organization of files and directories highlighting the merits of using fast file systems(PO-1,2,3,4, PSO-2)

Database Systems

Course Code: CS52 **Credits: 3:1:0:0**
Prerequisites: Nil **Contact Hours: 42L+28T**
Course Coordinator/s: Ganeshayya I Shidaganti

Course Contents:

Unit I

Introduction: Characteristics of Database approach, Actors on the Scene, Workers behind the scene, Advantages of using DBMS approach, Data models, schemas and instances, Three-schema architecture and data independence, Database languages and interfaces, the database system environment, Centralized and client-server architectures, Classification of Database Management systems, Entity-Relationship Model: Conceptual Database using high level Conceptual data models for Database Design, A Sample Database Application, Entity types, Entity sets Attributes and Keys Relationship types, Relationship Sets, Roles and Structural Constraints Weak Entity Types.

Unit II

Relational Model and Relational Algebra: Relational Model Concepts, Relational Model Concepts, Relational Model Constraints and Relational Database Schema Update Operations, Transactions and Dealing with Constraint violations, Unary Relational operations, Relational Algebra Operations from Set Theory, Binary Relational Operations, JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra Relational Database Design Using ER- to-Relational Mapping.

Unit III

Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic structure of SQL Queries, Additional Basic Operations, Null values, Aggregate Functions, nested Sub queries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization. Database programming issues and techniques, Embedded SQL.

Unit IV

Database Design: Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce Codd Normal Forms, Multivalued Dependencies and IV Normal Forms, Join Dependencies and V Normal Forms, Inference Rules, Equivalence and Minimal Cover, Properties of Relational Decomposition, Algorithms for relational database schema design.

Unit V

Transaction Management: Transaction Concept, a Simple Transaction Model, Transaction Atomicity and Durability, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels. Concurrency Control: Lock-Based Protocols, Deadlock Handling. Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm.

Text Books:

1. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Addison-Wesley, 2011.
2. Silberschatz, Korth and Sudharshan: Data base System Concepts, 6th Edition, Tata McGraw Hill, 2011.

Reference Book:

1. C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson education, 2009.

Course Outcomes (COs):

At the end of the course, students should be able to:

1. Design entity-relationship diagrams to represent simple database applications and convert to Relational model (PO-2, 3, 4, 5,12, PSO-2)
2. Construct relational algebraic expressions for queries using the concepts of relational database theory (PO-1, 2, 4,5,12, PSO-2)
3. Formulate using SQL, solutions to a broad range of query and data update problems (PO-2,3,4,5,11,12, PSO-2)
4. Apply Normalization to improve database design (PO-2,3,4, PSO-2)
5. Interpret the basic issues of transaction processing, concurrency control and recovery techniques (PO-2,3,4, PSO-2)

Computer Networks

Course Code: CS53 **Credits: 4:0:0:0**
Prerequisites: Data Communication **Contact Hours: 56L**
Course Coordinator/s: Sanjeetha R, Darshana Naik

Course Contents:

Unit I

Application Layer: The Web and HTTP: Overview of HTTP, Non-Persistent and Persistent Connections, HTTP Message Format, User-Server Interaction-Cookies, Web Caching, The Conditional GET. File Transfer- FTP: FTP Commands and Replies, Electronic Mail in the Internet: SMTP, Comparison with HTTP, Mail Access Protocols. DNS—The Internet's Directory Service: Services Provided by DNS, Overview of How DNS Works, DNS Records and Messages, Peer-to Peer Applications: P2P File Distribution, Distributed Hash Tables (DHTs).

Unit II

Transport layer: Multiplexing and De-multiplexing, Connectionless Transport-UDP: UDP Segment Structure, UDP Checksum, Connection-Oriented Transport-TCP: The TCP Connection, TCP Segment Structure, Round-Trip Time Estimation and Timeout, Reliable Data Transfer, Flow Control, TCP Connection Management, TCP congestion control.

Unit III

Foundation of Networking Protocols: Internet Protocols – IP Packet, Packet fragmentation and reassembly, ICMP, IPv6.
IPv4 Addresses: Classfull addressing, classless addressing, Transition from IPv4 to IPv6
Routing and Internetworking - Network-layer routing, least-cost-path algorithms, Intra domain routing protocols, Inter domain routing protocols, Congestion Control at Network Layer.

Unit IV

Multicasting Techniques and Protocols: Basic Definitions and Techniques – IP multicast address, Basic Multicast Tree Algorithms, Classification of multicast protocols, Intra domain Multicast Protocols – DVMRP, IGMP, MOSPF, PIM, Inter domain Multicast Protocols – MBGP.
Wireless and Mobile Networks: Introduction, Wi-Fi 802.11 Wireless LANs-The 802.11 Architecture, Mobility in the Same IP Subnet, Advanced Features in 802.11. Cellular Internet Access-An Overview of Cellular Architecture, Mobility Management Principles, Addressing, Routing to a Mobile Node, Mobile IP, Managing Mobility in Cellular Networks, Routing Calls to a Mobile User, Handoffs in GSM.

Unit V

VOIP and Multimedia Networking: Overview of IP telephony, VoIP signaling protocols – SIP, H.323 protocols, Real-time media transport protocols: RTP, RTCP, Distributed multimedia networking, Stream Control Transmission Protocol (SCTP).

Quality of Service: Integrated Services QoS – Traffic shaping, Admission Control, Resource Reservation Protocol (RSVP), packet scheduling, Differentiated Services QoS – Per-Hop Behavior (PHB).

Text Books:

1. James F. Kurose and Keith W. Ross: Computer Networking: A Top-Down Approach, 6th edition, Addison-Wesley, 2013.
2. Nader F. Mir: Computer and Communication Networks, Prentice hall, 2007.
3. Forouzan: Data Communications and Networking, 5th edition, McGraw Hill Education 2013.

Reference Books:

1. Larry L. Peterson and Bruce S Davie: Computer Networks: A Systems Approach, Fifth Edition, Elsevier, 2011.
2. Tanennbaum: Computer Networks, 4th Ed, Pearson Education/PHI, 2003.
3. William Stallings: Data and Computer Communications, 8th Edition, Pearson Education, 2012.

Course Outcomes (COs):

At the end of the course, students should be able to:

1. Describe the various application layer protocols used by TCP/IP reference model (PO-1,2,3, 4, *PSO-1*)
2. Differentiate between connection oriented and connection less services of transport layer. (PO-1,2,3, 4, *PSO-1*)
3. Solve problems of routing using various routing protocols and algorithms. (PO-1,2,3, 4, *PSO-1*)
4. Identify issues related to mobility in Internet and cellular networks. (PO-1,2,3, 4, *PSO-1*)
5. Illustrate multimedia networking with respect to content delivery and Quality of Service. (PO-1,2,3, 4, *PSO-1*)

Java Programming

Course Code: CS54 **Credits: 3:0:0:1**
Prerequisites: OOPS concepts **Contact Hours: 42L**
Course Coordinator/s: Dr. J Geetha, Hanumantharaju R

Course Contents:

Unit I

Overview of Java, Understanding static, Introducing final, Introducing Nested and Inner Classes, Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When Constructors Are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, Packages, Access Protection, Importing Packages, Interfaces, Exception handling mechanisms.
Self-Study: Java Buzz Words, Arrays, Strings, String Buffer.

Unit II

Multi-Threaded Programming: What are threads? How to make the classes threadable; Extending threads; Implementing runnable; Synchronization; Changing state of the thread; Bounded buffer problems, read-write problem, producer consumer problems.
Event Handling: Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model; Adapter classes; Inner classes.
Self-Study: Java Swings and Collection Frameworks.

Unit III

Java 2 Enterprise Edition Overview & Database Access: Overview of J2EE and J2SE, The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; Result Set; Metadata, Data types; Exceptions. Transaction Processing.
Self-Study: Stored Procedures.

Unit IV

Introduction to servlet, Servlet life cycle, Developing and Deploying Servlets, Exploring Deployment Descriptor (web.xml), Handling Request and Response, Initializing a Servlet, Accessing Database, Servlet Chaining, Session Tracking & Management, Dealing with cookies, Transferring Request, Accessing Web Context, Passing INIT and CONTEXT Parameter, Sharing information using scope object, Controlling concurrent access, User Authentication.
Self-Study: Filtering Request and Response, Programming Filter, Filter Mapping, Servlet Listeners.

Unit V

Basic JSP Architecture, Life Cycle of JSP (Translation, compilation), JSP Tags and Expressions, Role of JSP in MVC-2, JSP with Database, JSP Implicit Objects, Tag Libraries, JSP Expression Language (EL), Using Custom Tag, JSP Capabilities: Exception Handling, Session Management, Directives.
Self-Study: JSP with Java Beans.

Scheme for Self-Study evaluation:

- The topics are integral part of the course.
- No formal lectures will be held for the self-study topics.
- The course co-ordinator may provide reading materials for self-study topics (optional).
- The topics prescribed under self-study in curriculum are part of CIE and SEE.

Text Books:

1. Herbert Schildt: Java the Complete Reference, 8th Edition, Tata McGraw Hill, 2014.
2. Jim Keogh: J2EE - The Complete Reference, Tata McGraw Hill, 2007.

Reference Books:

1. Y. Daniel Liang: Introduction to JAVA Programming, 7th Edition, Pearson Education, 2007.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education, 2004.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Summarize the paradigm of java programming (PO-2,3,5,12, PSO-2)
2. Develop java programs using multithreaded and event handling concepts. (PO-2,3,5,12, PSO-2)
3. Identify different methods of creating and querying the database. (PO-2,3,5,12, PSO-2)
4. Design and develop web applications using Servlets. (PO-2,3,4,5,12, PSO-2)
5. Design dynamic web pages using JSP. (PO-2,3,4,5,12, PSO-2)

Intellectual Property Rights

Course Code: CS55 **Credits: 2:0:0:0**
Prerequisites: Nil **Contact Hours: 28L**
Course Coordinator/s: Dr. T.N.R. Kumar

Course Contents:

Unit I

Basic principles of IP laws: Introduction, Concept of property, Need for a holistic approach, Constitutional aspects of IP, Evolution of the patent system in UK, US and India, Basis for protection, Invention, Criteria for patentability, Non – patentable inventions.

Unit II

Patents: Introduction, Origin and meaning of the term patent, Objective of a patent law, the legislative provisions regulating patents, principles underlying the patent law in India, patentable invention. Procedure for obtaining patent: Submission of application, Filing provisional and complete specification, Examination of the application, advertisement of the acceptance, opposition, Grant and sealing of patent, Term of the patent, compulsory license. Provisional and complete specification: What is a specification? Kinds of specification, provisional specification, complete specification, Claims, Conditions for amendment.

Unit III

Rights conferred on a patentee: Patent rights, Exception and limitations, Duties of a Patentee. Transfer of patent: Forms of transfer of Patent rights, Assignment, kinds of assignment, License, kinds of license, Rights conferred on a licensee, Transmission of patent by operation of law. Infringement of patents: Construction of claims and infringement, patents held to be infringed, patents held to be not infringed. Action for Infringement: Where a suit is to be instituted, procedure followed in the suit, Onus of establishment infringement, Defense by the defendant, The Relief's, Injunction, Damages or account of profits, patent agents, patent drafting, database searching, and Case studies.

Unit IV

Copy Right: Meaning and characteristics of copy right, Indian copy right law, requirement of copy right, Illustrations copy right in literary work, Musical work, Artistic work, work of architecture, Cinematograph film, sound recording. Author and Ownership of copy right: Ownership of copy right, Contract of service, Contract for service, rights conferred by copy right, terms of copy right, license of copy right. Infringement of copy right: Acts which constitute infringement,

general principle, direct and indirect evidence of copying, Acts not constituting infringements, Infringements in literary, dramatic and musical works, Remedies against infringement of copy right, Case studies.

Unit V

Trade Marks: Introduction, Statutory authorities, procedure of registration of trademarks, rights conferred by registration of trademarks, licensing in trade mark, infringement of trade mark and action against infringement. Industrial Design: Introduction, procedure of registration of a design, Piracy of a registered design, Case studies.

Text Books:

1. Dr. T Ramakrishna: Basic principles and acquisition of Intellectual Property Rights, CIPRA, NSLIU -2005.
2. Dr. B. L. Wadehhra: Intellectual Property Law Handbook, Universal Law Publishing Co. Ltd., 2002.

References Books:

1. Dr. T Ramakrishna: Ownership and Enforcement of Intellectual Property Rights, CIPRA, NSLIU -2005.
2. Intellectual Property Law (Bare Act with short comments), Universal Law Publishing Co. Ltd., 2007.

Course Outcomes (COs):

At the end of the course the students should be able to:

1. Paraphrase the Basic Principles of IP laws like Patents, etc. (PO-6,8,9, 11, PSO-1).
2. Use the right procedure to get the Patent . (PO-4,6,8,9, 11,12, PSO-1).
3. Identify the various rights conferred to Patentee. (PO-4,6,8,9, 11,12, PSO-1).
4. Recognize the characteristics and Infringement of Copyright. (PO-6,8,9, 11,12, PSO-1).
5. Summarize the importance of Trade Marks, Industrial Design and its Infringement. (PO-6,7,8,9, 11, 12,PSO-1).

Data Mining

Course Code: CSE01 **Credits: 3:0:0:1**
Prerequisites: Nil **Contact Hours: 42L**
Course Coordinator/s: Dr. Seema S, Sowmya B J

Course Contents:

Unit I

Data Mining Introduction – Data – Types of Data – Data Mining Functionalities –Classification of Data Mining Systems – Issues –Data Preprocessing.
Association Rule Mining Mining Frequent Patterns – Apriori Algorithm Description.
Self-Study: Normalization.

Unit II

Association Rule Mining Two Illustrative Examples- Mining various Kinds of Association Rules – Correlation Analysis – Constraint Based Association Mining.
Classification and Prediction Basic Concepts - Decision Tree Induction - Bayesian Classification – Rule Based Classification.
Self-Study: Mining Sequential Patterns.

Unit III

Classification And Prediction Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction. C4.5 Algorithm Description- C4.5 Features - Two Illustrative Examples.
Self-Study: CBA, CMAR, CPAR.

Unit IV

Clustering And Trends in Data Mining Cluster Analysis - Types of Data – Partitioning Methods: K-means, K-mediods, PAM, CLARA, CLARANS – Hierarchical Methods: AGNES, DIANA, Density-Based Methods: DBSCAN – Grid Based Methods – STING, CLIQUE, Data Mining Applications.
Self-Study: Data Mining Applications.

Unit V

Mining Mining the World Wide Web - Page Rank Algorithm, Text mining, Mining Time Series Data, The CART Algorithm Briefly Stated, Ensemble methods-Increasing the Accuracy, Mining genomic data.
Self-Study: Ensemble methods-Increasing the Accuracy.

Scheme for Self-Study evaluation:

- The topics are integral part of the course.
- No formal lectures will be held for the self-study topics.
- The course co-ordinator may provide reading materials for self-study topics (optional).
- The topics prescribed under self-study in curriculum are part of CIE and SEE.

Text Books:

1. Jiawei Han and Micheline Kamber: Data Mining Concepts and Techniques, Elsevier, 2nd Edition, 2009.
2. Xindong Wu and Vipin Kumar: The top ten Algorithms in Data Mining, Chapman and Hall/CRC press.

Reference Books:

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, "Introduction to Data Mining", Person Education, 2007.
2. K.P. Soman, Shyam Diwakar and V. Aja, "Insight into Data Mining Theory and Practice", Eastern Economy Edition, Prentice Hall of India, 2006.
3. G. K. Gupta, "Introduction to Data Mining with Case Studies", Eastern Economy Edition, Prentice Hall of India, 2006.
4. Daniel T.Larose, "Data Mining Methods and Models", Wiley-Interscience, 2006.

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Describe the different data mining functionalities. (PO-2, 4,5,9,10,12, *PSO-3*)
2. Determine the kinds of patterns that can be discovered by association rule mining. (PO-1,2,3,4,5,9,10,12, *PSO-3*)
3. Indicate the working of different classification and prediction techniques. (PO-1,2,3,4,5,9,10,12, *PSO-3*)
4. Identify the clustering methods that can be used for a given data set. (PO-1,2,3,4,5,9,10,12, *PSO-3*)
5. Illustrate the use of data mining techniques in various fields like world wide web, time series data and genomic data. (PO-1,2,3,4,5,9,10,12, *PSO-3*)

Artificial Intelligence

Course Code: CSE02 **Credits: 3:0:0:1**
Contact Hours: 42L

Prerequisites: Knowledge of any advanced programming language, Algorithms and Data structures, Elementary Discrete Mathematics or similar.

Course Coordinator/s: Dr. S. Rajarajeswari and Dr. Annapurna P Patil

Course Contents:

Unit I

Introduction: What is AI? **Intelligent Agents:** Agents and environment, Rationality, the nature of environment, the structure of agents. **Problem-solving by search:** Problem-solving agents, Example problems, searching for solution, uniformed search strategies informed search strategies, Heuristic functions, On-linesearch agents and unknown environments.
Self-Study: On-line search agents and unknown environments.

Unit II

Logical Agents: Knowledge-based agents, the wumpus world, Logic, propositional logic, Reasoning patterns in propositional logic, Effective propositional model checking, Agents based on propositional logic **First-Order Logic**: Representation revisited, Syntax and semantics of first-order logic, using first-order logic, Knowledge engineering in first-order logic. **Interference in First-order Logic**: Propositional vs first-order inference, Unification and lifting, Forward chaining, Backward chaining, Resolution.
Self-Study: Forward chaining, Backward chaining.

Unit III

Planning: Definition, planning with state-space search, Planning graphs, other planning approaches analysis. **Uncertainty:** Acting under uncertainty, Basic probability Notations, Inference using full joint distributions, Independence, Bayes' rule and its use. **Learning from Examples**: Forms of Learning, Supervised Learning, Learning Decision Trees Best Hypothesis Theory Regression and Classification with Linear Models ANN, Non Parametric Models SVM, Ensemble Learning, Practical Machine Learning.
Self-Study: SVM, Ensemble Learning, Practical Machine Learning.

Unit IV

Learning Probabilistic Models: Statistical Learning, Learning with complete data Hidden Variables. **Natural Language Processing & communication:** Language Models Text Classification Information retrieval, Extraction, Phrase, structure grammars, Syntactic Analysis, Augmented Grammars & Semantic Interpretation, Machine translation, Speech recognition.
Self-Study: Machine translation, Speech recognition.

Unit V

Genetic Algorithms: GA, Significance of genetic operators, Termination parameters, Niching and Speciation, Evolving Neural Networks, Theoretical grounding, Ant Algorithms.
Perception: Image Formation Earthly Image Processing operations, Object recognition by appearance, from structural Information 3D world, using world.
Robotics: Introduction, Hardware Perception planning to Move, planning uncertain movement, moving robotic software architecture, application domains.
Self-Study: Evolving Neural Networks, Theoretical grounding, Ant Algorithms. Robotic software architecture, application domains.

Scheme for Self-Study evaluation:

- The topics are integral part of the course.
- No formal lectures will be held for the self-study topics.
- The course co-ordinator may provide reading materials for self-study topics (optional).
- The topics prescribed under self-study in curriculum are part of CIE and SEE.

Text Books:

1. Stuart Russel, Peter Norvig: Artificial Intelligence - A Modern Approach, 3rd Edition, Pearson Education, 2012. (Unit-1,2,3,4,5).
2. Elaine Rich, Kevin Knight, Shivashankar B Nair: Artificial Intelligence, 3rd Edition, Tata McGraw Hill, 2011. (Unit-5).

Reference Books:

1. Peter Jackson, “Introduction to Expert Systems”, 3rd Edition, Pearson Education, 2007.(Unit 5).
2. Deepak Khemani “Artificial Intelligence”, Tata McGraw Hill Education 2013.(unit 3).
3. http://nptel.ac.in

Course Outcomes (COs):

At the end of the course, the student should be able to:

1. Identify problems that are amenable to specific solution by appropriate AI methods.(PO-1,2,3,4,12,PSO-2,3)
2. Utilize various symbolic knowledge representation to specify domains and reasoning tasks of a situated software agent. Use different logical systems for inference over formal domain representations, and trace how a particular inference algorithm works on a given problem specification.(PO-1,2,4,5,9,12,PSO-1,2,3)
3. Formalize a given problem in the language/framework of different AI methods and solve using basic AI algorithms. (PO-1,2,3,4,5,6,7,PSO-2,3)
4. Design and carry out an empirical evaluation of different algorithms on a problem formalization, and state the conclusions that the evaluation supports.(PO-3,4,5,9,10,PSO-2,3)
5. Communicate scientific knowledge at different levels of abstraction in a variety of research settings.(PO-4,5,9,10,11,12,PSO-2,3)

Operation Research

Course Code: CSE03 **Credits: 4:0:0:0**
Prerequisites: Nil **Contact Hours: 56L**
Course Coordinator/s: Dr. Jagdish S Kallimani

Course Contents:

Unit I

Introduction, Linear Programming – 1: Introduction: The origin, nature and impact of OR; Defining the problem and gathering data; Formulating a mathematical model; Deriving solutions from the model; Testing the model; Preparing to apply the model; Implementation . Introduction to Linear Programming: Prototype example; the linear programming (LP) model.

Unit II

LP – 2, Simplex Method: Assumptions of LP; Additional examples. The essence of the simplex method; Setting up the simplex method; Algebra of the simplex method; the simplex method in tabular form; Tie breaking in the simplex method, Adapting to other model forms; Post optimality analysis; Computer implementation Foundation of the simplex method. Duality Theory: The revised simplex method, a fundamental insight. The essence of duality theory; Economic interpretation of duality, Primal dual relationship; Adapting to other primal forms.

Unit III

Duality Theory and Sensitivity Analysis, Other Algorithms for LP: The role of duality in sensitive analysis: The essence of sensitivity analysis; Applying sensitivity analysis. The dual simplex method; parametric linear programming: The upper bound technique.

Unit IV

Transportation and Assignment Problems: The transportation problem: A streamlined simplex method for the transportation problem; The assignment problem; A special algorithm for the assignment problem.

Unit V

Game Theory, Decision Analysis: Game Theory: The formulation of two persons, zero sum games; Solving simple games- a prototype example; Games with mixed strategies; Graphical solution procedure; Solving by linear programming, Extensions. Decision Analysis: A prototype example; Decision making without experimentation; Decision making with experimentation; Decision trees. Metaheuristics: The nature of Metaheuristics, Tabu Search, Simulated Annealing, Genetic Algorithms.

Text Book:

1. Frederick S. Hillier and Gerald J. Lieberman: Introduction to Operations Research: Concepts and Cases, 8th Edition, Tata McGraw Hill, 2005. (Chapters: 1, 2, 3.1 to 3.4, 4.1 to 4.8, 5, 6.1 to 6.7, 7.1 to 7.3, 8, 13, 14, 15.1 to 15.4)

Reference Books:

1. Wayne L. Winston: Operations Research Applications and Algorithms, 4th Edition, Cengage Learning, 2003.
2. Hamdy A Taha: Operations Research: An Introduction, 8th Edition, Pearson Education, 2007.

Course Outcomes (COs):

At the end of the course the student will be able to:

1. Explain the concepts of LPP and formulations. (PO-1,2,3, PSO-2)
2. Construct problems under simplex methods and its types. (PO-1,2,3, PSO-2)
3. Identify problems under duality and justify them. (PO-3,5,7, PSO-2)
4. Solve assignment and transportation problems. (PO-4, PSO-2)
5. Illustrate game theory and decision making problems. (PO-2,4,11, PSO-2)

Advanced Algorithms

Course Code: CSE04 **Credits: 3:0:1:0**
Prerequisites: Algorithms **Contact Hours: 42L+28P**
Course Coordinator/s: Dr. Jagdish S Kallimani

Course Contents:

Unit I

Analysis Techniques: Growth of Functions, Asymptotic notations, Standard notations and common functions, Recurrences and Solution of Recurrence equations – The Substitution method, The recurrence – tree method, The master method, Amortized Analysis: Aggregate, Accounting and Potential Methods.

Unit II

Graph Algorithms: Bellman-Ford Algorithm, Single source shortest paths in a DAG, Johnson's Algorithm for sparse graphs, Maximum bipartite matching. Trees: B-trees, Red-Black trees.
Hashing: General Idea, Hash Function, Separate Chaining, Open addressing, Rehashing, Extendible hashing.

Unit III

Number – Theoretic Algorithms: Elementary notations, GCD, Modular Arithmetic, Solving modular linear equations, The Chinese remainder theorem, Powers of an element, RSA cryptosystem. Heaps: Heaps, Priority Queues, Binomial Heaps, Fibonacci Heaps.

Unit IV

String Matching Algorithms: Naïve string matching, Rabin – Karp algorithm, String matching with finite automata, Knuth-Morris-Pratt algorithm, Boyer-Moore Algorithms.

Unit V

Algorithmic Puzzles: Magic Square, n-queens problem, Glove Selection, Ferrying Soldiers, Jigsaw Puzzle Assembly, A Stack of Fake Coins, Maximum Sum Descent, Hats of Two Colors, Pluses and Minuses, Searching for a Pattern, Locker Doors, Palindrome Counting, Inverting a Coin Triangle, Sorting 5 in 7.

Text Books:

1. T H Cormen, C E Leiserson, R L Rivest and C Stein: Introduction to Algorithms, 3/e, PHI, 2011.
2. Mark Allen Weiss: Data Structures and Algorithm Analysis in C++, 3rd Edition, Pearson Education, 2011.
3. Anany Levitin and Maria Levitin: Algorithmic Puzzle, Oxford University Press, 2011

Reference Books:

1. Ellis Horowitz, Sartaj Sahni, S Rajasekharan: Fundamentals of Computer Algorithms, University Press, 2007.
2. Alfred V Aho, John E Hopcroft, J D Ullman: The Design and Analysis of Computer Algorithms, Pearson Education, 2011.

Course Outcomes (COs):

At the end of the course the student will be able to:

1. Devise recurrence relations and amortized cost of various operations. (PO-1,2,4,PSO-1,2)
2. Illustrate graph algorithms such as Bellman-Ford, Shortest path, and Bipartite matching, B-trees, Red-Black trees and hashing techniques. (PO-1,2,3,PSO-1,2)
3. Identify the methods for solving modular linear equations, Chinese remainder theorem and RSA cryptosystem, Describe types of heaps such as Binomial and Fibonacci heaps.(PO-1,2,4,6,9,10-PSO-1,2)
4. Assess the string matching algorithms such as Boyer-Moore and Knuth-Morris-Pratt algorithm. (PO-1,2,3,6,9,10,PSO-1,2)
5. Compose mathematical models, objective functions and constraints to solve algorithmic puzzles. (PO-1,2,3,4,9,10,12, ,PSO-1,2)

List of Lab Exercises:

The students are expected to get the proficiency in solving problems in the laboratory on growth of functions

1. Graph Algorithms.
2. B-Trees.
3. RB Trees.
4. Hashing.
5. Heaps.
6. Number-Theoretic Algorithms.
7. String Matching Algorithms.
8. Algorithmic Puzzles and Others.

System Simulation

Course Code: CSE05 **Credits: 4:0:0:0**
Prerequisites: Nil **Contact Hours: 56L**
Course Coordinator/s: Dr. Diwakar Harekal

Course Contents:

Unit I

Introduction: When simulation is the appropriate tool and when it is not appropriate, Advantages and disadvantages of Simulation, Areas of application, Systems and system environment components of a system, Discrete and continuous systems, Model of a system, Types of Models, Discrete Event System Simulation, Steps in a Simulation Study. Simulation examples: Simulation of queuing systems, Simulation of inventory systems, Other examples of simulation.

Unit II

General Principles, Simulation Software Concepts in Discrete Event Simulation, The Event Scheduling / Time Advance Algorithm, World Views, Manual simulation Using Event Scheduling, List processing. Simulation in Java, Simulation in GPSS. Queuing Models: Characteristics of Queuing Systems.

Unit III

Random Number Generation, Random Variate Generation-Properties of random numbers, Generation of pseudo random numbers, Techniques for generating random numbers, Tests for Random Numbers. Random Variate Generation: Inverse transform technique, Acceptance Rejection technique, Special properties.

Unit IV

Input Modeling: Data Collection, Identifying the distribution with data, Parameter estimation, Goodness of Fit Tests, Fitting a non-stationary Poisson process, Selecting input models without data, Multivariate and Time Series input models.

Unit V

Output Analysis for a Single Mode: Types of simulations with respect to output analysis,
Stochastic nature of output data, Measures of performance and their estimation, Output analysis for terminating simulations, Output analysis for steady state simulations. Verification and Validation of Simulation Models: Model building, verification and validation, Verification of simulation models, Calibration and validation of models. Optimization via Simulation.

Text Book:

1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete Event System Simulation, 4th Edition, Pearson Education, 2012.

Reference Books:

1. Lawrence M. Leemis, Stephen K. Park: Discrete –Event Simulation: A First Course, First edition, Pearson / Prentice-Hall, 2006.
2. Averill M. Law: Simulation Modeling and Analysis, 4th Edition, Tata McGraw-Hill, 2011.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Identify modeling system and types of simulation tools. (PO-1,2,5, PSO-2)
2. Understand the concepts of scheduling /queuing system using simulation software. (PO-1,2,5, PSO-2)
3. Test and Analyze random function generation through various transform techniques. (PO-1,2,3,5, PSO-2)
4. Analyze the data collection process. (PO-1,2,4,5 PSO-2)
5. Interpret the stochastic nature of output data. (PO-1, 2,4,5, PSO-2)

Object Oriented Modeling and Design

Course Code: CSE06 **Credits: 3:0:1:0**
Prerequisites: Nil **Contact Hours: 42L+28P**
Course Coordinator/s: Meeradevi K

Course Contents:

Unit I

UML Diagrams: What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history. UML diagrams – Use Case – Class Diagrams– Interaction Diagrams – State Diagrams – Activity Diagrams – Package, component and Deployment Diagrams.

Unit II

Design Patterns: Design Patterns: Creational Pattern Abstract Factory, Builder, Factory Method, Object Pool, Prototype, Structural Pattern: Adapter, Bridge, Composite, Decorator, Facade, Flyweight,, Behavioral Pattern: Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Null Object, Observer.

Unit III

Case Study: Case study – the Next Gen POS system, ATM system - Inception - Use case Modeling - Relating Use cases – include, extend and generalization - Elaboration - Domain Models - Finding conceptual classes and description classes – Associations – Attributes – Domain model refinement – Finding conceptual class Hierarchies - Aggregation and Composition. - Abstract classes: Multiple inheritance; Metadata; Reification; Constraints.

Unit IV

Advanced Modeling And Design: System sequence diagrams - Relationship between sequence diagrams and use cases Logical architecture and UML package diagram – Logical architecture refinement - UML class diagrams derived data; Packages; Practical tips. State Modeling, Advanced: Events, States, Transitions and Conditions; State diagrams; State diagram behavior nested states signal generalization concurrency; Relation of class and state models.

Unit V

Implementation Modeling, Coding And Testing: UML interaction diagrams - Applying GoF design patterns. Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models, Mapping design to code – Testing: Issues in OO Testing – Class Testing – OO Integration Testing – GUI Testing – OO System Testing.

Text Books:

1. Craig Larman, "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development", Third Edition, Pearson Education, 2005.(unit 1,3,4,5)
2. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2nd Edition, Pearson Education, 2005. (Unit 1,3,4,5)
3. http://sourcemaking.com/design_patterns (Unit : 2)

Reference Books:

1. Simon Bennett, Steve McRobb and Ray Farmer, "Object Oriented Systems Analysis andDesign Using UML", Fourth Edition, Mc-Graw Hill Education, 2010.
2. Erich Gamma, a n d Richard Helm, Ralph Johnson, John Vlissides, "Design patterns:Elements of Reusable Object-Oriented Software", Addison-Wesley, 1995.
3. Martin Fowler, "UML Distilled: A Brief Guide to the Standard Object Modeling Language", Third edition, Addison Wesley, 2003.
4. Paul C. Jorgensen, "Software Testing: A Craftsman's Approach", Third Edition, Auerbach Publications, Taylor and Francis Group, 2008.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Design projects using OO concepts and UML diagrams (PO-1,2,3,4,5, PSO-2,3)
2. Apply design pattern concepts on real time applications.(PO-2,3,4,5,11, PSO-2,3)
3. Create use case& conceptual charts diagram for the chosen real time application.(PO-3,4,5,6,7,8, PSO-2,3)
4. Apply the advanced modeling & design to the chosen real time application(PO-2,3,5,9, PSO-2,3)
5. Deploy the designed model & various test cases for the app chosen (PO-4,5,9,10,11, PSO-2,3)

List of Lab Exercises:

1. Draw the Class diagram containing Classes with methods, Relationship names with attributes of relationship types for E-book Management System, ATM System and Result analysis management System using UML.
2. Activity diagram containing activities and their association with conditions for Department Office Management System and Online Examination System.
3. Draw use case diagrams containing include and extend relationship using UML for Student Information System and Online Railway Ticket Reservation System.
4. Draw state chart diagrams for using UML for Course Registration System and Student Attendance Management System and Health Center Record Management System.
5. Draw sequence diagrams for Airline Reservation System (Ticket Booking, Cancel), Sports Club Management System and Department Timetable Management System.
6. Draw the component diagrams for Book bank management System, Graduation rank electing System (UG/PG), Result analysis management System.

Java Laboratory

Course Code: CSL57 **Credits: 0:0:1:0**
Prerequisites: Nil **Contact Hours: 28P**
Course Coordinator/s: Dr. J Geetha

Course Contents:

1. Programs on Basics of Java.
2. Implementing OOP concepts.
3. Programs on inheritance.
4. Programs on interface and packages.
5. Programs on Exception Handling.
6. Programs on multi-threading.
7. Programs on Synchronization of threads.
8. Programs on Event Handling.
9. Programs on JDBC.
10. Programs on Servlets.
11. Programs on Sessions.
12. Programs on JSP.
13. Programs on JSP and Servlets using JDBC.
14. Programs on Java Bean.

Text Books:

1. Herbert Schildt: Java the Complete Reference, 8th Edition, Tata McGraw Hill, 2014.
2. Jim Keogh: J2EE - The Complete Reference, Tata McGraw Hill, 2007.

Reference Books:

1. Y. Daniel Liang: Introduction to JAVA Programming, 7th Edition, Pearson Education, 2007.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education, 2004.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Develop Java programs using multithreaded and event handling concepts.(PO-1,2,3 PSO-2)
2. Identify the different methods of creating and querying the database. (PO-1,2,3 PSO-2)
3. Design and develop web applications using Servlets and JSP. (PO-1,2,3 PSO-2)

Database Systems Laboratory

Course Code: CSL58 **Credits: 0:0:1:0**
Prerequisites: Nil **Contact Hours: 28P**
Course Coordinator/s: Ganeshayya I Shidaganti

Course Contents:

1. Introduction to MongoDB and CRUD Operations.
2. MongoDB Usage in Enterprise Applications.
3. Develop an Entity-Relationship(ER) Model and Mapping to Relational Model.
4. Implement SQL Queries using DDL,DML Statements.
5. Build an Application model in Oracle DB using Nested queries, Triggers and Views.
6. Design a Database application for a particular case study using Visual Basic/Java Script in visual studio /Eclipse Tool.

Reference Books:

1. "Database Management Systems" by Raghu Ramakrishnan, Johanners Gehrke, Second Edition. McGraw-Hill Education.
2. "Fundamentals of Database Systems" by Ramez Elmasri, Shamkant B. Navathe ,Fifth Edition, Pearson Publications.
3. "Database System Concepts" by Abraham Silberschatz , Henry F. Korth, sixth Edition ,McGraw Hill Education.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Manage the Core MongoDB Operations(PO-2,5,PSO-1,2)
2. Design an Oracle DB Application Using SQL DDL statements, DML statements and Queries(PO-1,3,4,5,PSO-1,2)
3. Develop a Real time database Application Using IDE of student's choice (PO-1,3,4,5,PSO-1,2)

Computer Networks Laboratory

Course Code: CSL59 **Credits: 0:0:1:0**
Prerequisites: Data Communication **Contact Hours: 28P**
Course Coordinator/s: Sanjeetha R, DarshanaNaik

Course Contents:

Note: Student is required to solve one problem from **PART-A** and one problem from **PART-B.** The questions are allotted based on lots. Both questions carry equal marks.

PART – A

Implement the following in C/C++ or Wireshark as suitable.

1. Write a program for error detection using CRC-CCITT (16-bits).
2. Write a program to generate Hamming Code for error detection and correction.
3. Trace Hypertext Transfer Protocol.
4. Trace File Transfer protocol, Trace Transmission control protocol
5. Trace Domain Name Server.
6. Write a client-server program using TCP/IP sockets in which client requests for a file by sending the file name to the server, and the server sends back the contents of the requested file if present.
7. Trace Internet Protocol and Internet Control Message Protocol .
8. Trace Dynamic Host Configuration Protocol.
9. Write a program to implement traffic policing using Leaky bucket algorithm.
10. Write a program to implement traffic policing using Token bucket algorithm.

PART-B

The following experiments shall be conducted using either NS-2/NS3/OMNET++ or any other suitable simulator.

1. Simulate a three nodes point-to-point network with duplex links between them. Set the queue size vary the bandwidth and find the number of packets dropped.
2. Simulate a four node point-to-point network, and connect the links as follows: n0-n2, n1-n2 and n2-n3. Apply TCP agent between n0-n3 and UDP agent between n1-n3. Apply relevant applications over TCP and UDP agents by changing the parameters and determine the number of packets sent by TCP/UDP.

3. Simulate simple Extended Service Set with transmitting nodes in wireless LAN and determine the performance with respect to transmission of packets.
4. Simulate a wireless network, generate traffic and analyze its performance.
5. Simulate a transmission of ping message over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.

Reference Books:

1. James F. Kurose and Keith W. Ross: Computer Networking: A Top-Down Approach, 5th edition, Addison-Wesley, 2009.
2. Larry L. Peterson and Bruce S Davie: Computer Networks: A Systems Approach, Fifth Edition, Elsevier, 2011.
3. Behrouz A. Forouzan: Data communication and Networking, 4th edition, Tata McGraw-Hill, 2012.
4. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff: Unix Network programming, The sockets networking API, Addison-Wesley Professional, 2004.

Course Outcomes (COs):

At the end of the course the student will be able to:

1. Illustrate networking concepts using programming languages like C/C++/Java/Python. (PO-1, 2, 3,4, 5, *PSO-1*)
2. Use packet sniffing tools like Wireshark to intercept & analyze the packets at different network layers. (PO-1,2, 3, 4,5, *PSO-1*)
3. Use simulators like NS2/NS3. (PO-3, 5, *PSO-1*)

Compiler Design

Course Code: CS61 **Credits: 3:1:0:0**
Prerequisites: Theory of Computation **Contact Hours: 42L+28T**
Course Coordinator/s: Dr. A Parkavi

Course Contents:

Unit I

Introduction, Lexical Analysis: Language processors, The Structure of Compilers, Lexical analysis: The Role of Lexical Analyzer, Input Buffering, Specifications of Tokens, recognition of Tokens. Syntax Analysis: Introduction, Writing a Grammar.

Unit II

Parsing: Top-down Parsing, Bottom-up Parsing, Introduction to LR Parsing: Simple LR parser. More Powerful LR Parsers: Canonical parser, LALR parser.

Unit III

Syntax-Directed Definitions: Evaluation order for SDDs, Applications of Syntax-directed translation, Syntax-directed translation schemes. Run-Time Environments: Storage Organization, Stack allocation of space.

Unit IV

Intermediate Code Generation: Variants of syntax trees, Three-address code, Types and declarations, Translation of expressions, Type checking, Control flow, Back patching, and Switch statements, Intermediate code for procedures.

Unit V

Code Generation: Issues in the design of Code Generator, The Target language, Addresses in the target code, Basic blocks and Flow graphs, Optimization of basic blocks, A Simple Code Generator.

Text Book:

1. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman: Compilers-Principles, Techniques and Tools, 2nd Edition, Pearson education, 2012.

Reference Books:

1. Kenneth C Louden: Compiler Construction - Principles & Practice, First Edition, Brooks/Cole, CENGAGE learning, 1997.
2. Andrew W Appel: Modern Compiler Implementation in C, First Edition, Cambridge University Press, 2010.

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Identify the similarities and differences among various parsing techniques and grammar transformation techniques. (PO-1,2,3,4,5,12,PSO-1,2)

2. Design the structures and support required for compiling advanced language features. (PO-1,2,3,4,5,12,PSO-1,2)
3. Illustrate lexical syntactic and semantic analysis into meaningful flases for a compiler to undertake language translation. (PO-1,2,3,4,5,12,PSO-1,2)
4. Design and implement an intermediate code generator based on given code pattern. (PO-1,2,3,4,5,12,PSO-1,2)
5. Describe the techniques for intermediate code and machine code optimization. (PO-1,2,3,4,5,12,PSO-1,2)

Software Engineering

Course Code: CS62 **Credits: 3:0:0:1**
Prerequisites: Nil **Contact Hours: 42L**
Course Coordinator/s: Dr. Annapurna P Patil

Course Contents:

Unit I

The Software Problem & Processes: Cost, Schedule & Quality, Scale & Change, Software Processes: Process & Project, Component Software Processes, Software Development Process Models, Agile Process – Scrum, Project Management Process.
Self-Study- Activity based learning of Process models.

Unit II

Requirements Analysis & Project Planning: Requirements Analysis & Specification: Value of a Good SRS, Requirements Process, Requirements Specification, Functional Specification with Use Cases, Other Approaches for Analysis.
Planning a Software Project: Effort Estimation, Project Schedule & Staffing, Quality Planning, Risk Management Planning, Project Monitoring Plan.
Self-Study- Case study for planning.

Unit III

Design, Coding: Design: Design Concepts, Function-oriented Design, Object-oriented Design, Detailed Design, Metrics.
Coding: Programming Principles & Guidelines, Incrementally Developing Code, Managing Evolving Code.
Self-Study- Design activities.

Unit IV

Unit testing and Testing: Unit Testing, Code Inspection, Metrics Testing Concepts, Testing Process, Black-box Testing, White-box Testing, Metrics.
Self-Study- study of testing tools

Unit V

Web Engineering: Web Engineering: The Need for Web Engineering: Introduction, Web Applications vs Conventional Software, The Need for an Engineering Approach, Empirical Assessment.
Self-Study- role play for agile

Scheme for Self-Study evaluation:

- The topics are integral part of the course.
- No formal lectures will be held for the self-study topics.
- The course co-ordinator may provide reading materials for self-study topics (optional).
- The topics prescribed under self-study in curriculum are part of CIE and SEE.

Text Books:

1. Pankaj Jalote: A Concise Introduction to Software Engineering , Springer, 2008 (Chapters: 1-4, 6-8).
2. Emilia Mendes, Nile Mosley: Web Engineering, Springer, 2006 (Chapter: 1).

Reference Books:

1. Roger S. Pressman: Software Engineering A Practitioner's Approach, 7th Edition, McGraw Hill, 2010.
2. David Gustafson: Software Engineering, Schaum's Outline Series, McGraw Hill, 2002 (Chapters: 6).

Course Outcomes (COs):

At the end of the course, the students should be able to:

1. Recall the principles and techniques of Software Engineering (PO-3,5,6,10,11,12,PSO-2,3)
2. Demonstrate on understanding of activities in project management, requirement engineering process and the different types of system models. (PO-2,3,5,6,10,11,12,PSO-2,3)
3. Demonstrate an ability to use techniques & tools necessary for software design & development. (PO-1,2,3,4,5,6,9,10,11,12,PSO-2,3)
4. Work as an individual & as a part of multi-disciplinary team to develop & deliver quality software with necessary techniques. (PO-2,3,5,9,10,11,12)
5. An ability to use techniques & tools necessary for one or more significant application domains. (PO-2,3,4,5,6,8,9,10,11,12)

UNIX System Programming and Web Technologies

Course Code: CS63 **Credits: 3:1:0:0**
Prerequisites: Nil **Contact Hours: 42L+28T**
Course Coordinator/s: Aparna R

Course Contents:

Unit I

Unix Basics & Fundamentals of JavaScript:
Unix Basics: File Types, The UNIX and POSIX File System, The UNIX File Attributes, Inodes in UNIX, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs.
HTML5 and Java script Basics: Introduction, JavaScript syntax, Types of Data and Variables, Operations and calculations, Arrays, Decisions, Loops, Using Functions, Using Objects, The String Objects, The Math class, The Array objects, The Document Object, Using Events.

Unit II

Process and JavaScript Advanced:
Process: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, UNIX Kernel Support for Processes.
JavaScript Advanced: Scopes and Closures, Understand "this" and prototypes, OO concepts as applied to JS and prototypal inheritance, Understanding the meaning of asynchronous.

Unit III

Process Control and Node.js
Introduction: Process Identifiers, fork, vfork, exit, wait, waitpid Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection. Process Relationships: Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, Job Control, Shell Execution of Programs, Orphaned Process Groups.
Introduction to Server-side JS Framework. Node.js : Introduction - What is Node.js, Architecture, Feature of Node JS, Installation and setup - Creating web servers with HTTP (Request & Response), Understand npm and package.json structure File system APIs.

Unit IV

Signals and Node.js with MONGODB

Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, Kill, Alarm, Interval Timers. Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model

IPC: IPC Overview of IPC Methods, Pipes, popen, pclose Functions, Coprocesses, FIFOs.

CRUD Operations using Node.js: Event Handling - GET & POST implementation, Use Express.js to create a REST API. Use GET, POST. Connect to NoSQL Database using Node.js, Implementation of CRUD operations.

Unit V

Introduction to Client-side JS Framework – Basics of Angular 4.0:

Introduction to Angular 4.0, Needs & Evolution, Features-Setup and Configuration, Components and Modules–Templates, Change Detection, Directives, Data Binding, Pipes, Nested Components, Model Driven Forms or Reactive Forms.

Text Books:

1. Nate Murray, Felipe Coury, Ari Lerner and Carlos Taborda, “ng-book, The Complete Book on Angular 4” September 2016
2. Krasimir Tsonev, “Node.js by Example Paperback”, May 2015
3. W. Richard Stevens: Advanced Programming in the UNIX Environment, Second Edition, Pearson education, 2011.

Reference Books:

1. Terrence Chan: UNIX System Programming Using C++, First edition, Prentice Hall India, 2011.
2. Kay A Robbins and Steve Robbins: Unix Systems Programming, First Edition, Pearson Education, 2009.
3. Marc J. Rochkind: Advanced UNIX Programming, 2nd Edition, Pearson
4. Web link for Angular4.0: https://angular.io/
5. Web link for Node.js : https://nodejs.org/en/
6. Web link for MongoDB: https://www.mongodb.com/

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Identify & use Unix utilities to create & manage file processing operations with Unix communication & networking commands. (PO-2,3,4,5,12,PSO-1,2)
2. Describe the different components of process & its stages. (PO-2,3,5,12,PSO-1,2)
3. Demonstrate an ability to use techniques in Unix operating systems in solving process communication issues. (PO-1,2,3,4,5,12,PSO-1,2)
4. Design & implement dynamic web pages with good aesthetic sense of designing & technicality. (PO-2,3,4,5,10, 12,PSO-1,2,3)
5. Combine multiple technologies to create advanced web components.(PO-2,3,4,5,12,PSO-1,2,3)

Mini Project

Software Development for Portable Device/Rich Internet Applications/Embedded Systems

Course Code: CS64 **Credits: 0:0:4:2**
Prerequisites: HTML, JAVA, Microprocessors **Contact Hours: --**
Course Coordinator/s: Dr. Anita Kanavalli , Veena G S, Pramod C Sunagar

Course Contents:

Software Development for Portable Devices:

1. Introduction to Android
2. Introduction to Android Studio, AVD, DDMS & SDK
3. Exploring Android Project and related files
4. Android Programs on Activity class and Widgets
5. Android Programs on Explicit Intents
6. Android Programs on Implicit Intents
7. Android Programs on Picker Views
8. Android Programs on multimedia files
9. Android Programs on SQLite
10. Android Programs on Messaging Services
11. Android Programs on Bluetooth, Accelerometer
12. Android Programs on Location Based Services
13. Mini Project
14. Mini Project

Embedded Systems:

1. Work with different embedded hardware launch pads and software IDEs .
2. Work with different communication interfaces like ADC, UART, I2C, SPI, between processing nodes and sensors and end devices
3. Work on calibration, sensitivity of the sensors and actuators reading their datasheets

Reference Books:

1. Beginning Android Programming with Android Studio by J. F. DiMarzio, 4th Edition, Wiley Publishing, 2016.
2. Android 6 for Programmers: An App-Driven Approach by Dr. Harvey Deitel Paul Deitel, 2016.
3. NuMicro™ NUC100 Series NUC130/NUC140 Technical Reference Manual
4. Cortex-M4 Technical Reference Manual.
5. Nodemcu Documentation./ Embedded Systems: Introduction to Arm® Cortex™-M Microcontrollers , Fifth Edition (Volume 1) 5th Edition.

Scheme for Self-Study Evaluation:

- The topics are integral part of the course.
- No formal lectures will be held for the self-study topics.
- The course co-ordinator may provide reading materials for self-study topics (optional).
- The topics prescribed under self-study in curriculum are part of CIE and SEE.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Acquire practical knowledge within the chosen area of technology for project development.(PO-1,2,3,4,5,12,PSO-1,2,3)
2. Identify, analyze, formulate & handle project involving multiple hardware & software technologies. .(PO-1,2,3,4,5,12,PSO-1,2,3)
3. Implement quality based solutions using different platforms, tools & programming languages for the chosen area of project development.(PO-1,2,3,4,5,6,7,8,10,11,12,PSO-1,2,3)
4. Contribute as an individual or in a team in development of technical projects. (PO-2,3,5,6,7,8,9,10,11,12,PSO-1,2,3)
5. Develop effective communication skills for presentation of project related activities. (PO-2,4,5,6,7,8,9,10,11,12,PSO-1,2,3)

Mobile Computing

Course Code: CSE07 **Credits: 3:0:1:0**
Prerequisites: Computer Networks **Contact Hours: 42L+28P**
Course Coordinator/s: Dr. Monica R Mundada

Course Contents:

Unit I

Mobile Computing Architecture: Types of Networks, Architecture for Mobile Computing, 3-tier Architecture, Design Considerations for Mobile Computing.
Wireless Networks – 1: GSM and SMS
Global Systems for Mobile Communication, GSM and Short Service Messages (SMS):GSM Architecture, Entities, Call routing in GSM, PLMN Interface, GSM Addresses and Identities, Network Aspects in GSM, Mobility Management, GSM Frequency allocation Introduction to SMS, SMS Architecture, SM MT, SM MO.

Unit II

Wireless Networks – 2: GPRS
GPRS and Packet Data Network, GPRS Network Architecture, GPRS Network Operations, data Services in GPRS, Applications for GPRS, Billing and Charging in GPRS.
Wireless Networks – 3: CDMA, 3G and WiMAX
Spread Spectrum technology, IS-95, CDMA versus GSM, Wireless Data, Third Generation Networks, Applications on 3G, Introduction to WiMAX.

Unit III

Mobile Telecommunications Systems
Introduction to 1G, 2G, 3G systems, TETRA, DECT, UMTS. **Mobile Client:** Moving beyond desktop, Mobile handset overview, Mobile phones and their features, PDA, Design Constraints in applications for handheld devices.

Unit IV

Mobile OS and Computing Environment
Smart Client Architecture, the Client: User Interface, Data Storage, Performance, Data Synchronization, Messaging. The Server: Data Synchronization, Enterprise Data Source Messaging. Mobile Operating Systems: WinCE, Palm OS, Simian OS, Linux, and Proprietary OS Client Development: The development process, Need analysis Phase, Design phase, Implementation and Testing phase, Deployment phase, Development Tools, Device Emulators.

Unit V

Building, Mobile Internet Applications

Thin client: Architecture, the client, Middleware, messaging Servers, Processing a Wireless request, Wireless Applications Protocol (WAP) Overview, Wireless Languages: Markup Languages, HDML, WML, HTML, cHTML, XHTML, VoiceXML.

J2ME

Introduction, CDC, CLDC, MIDP; Programming for CLDC, MIDlet model, Provisioning, MIDlet life-cycle, Creating new application, MIDlet event handling, GUI in MIDP, Low level GUI Components, Multimedia APIs; Communication in MIDP, Security.

Text Books:

1. Dr. Ashok Talukder, Ms Roopa Yavagal ,Mr . Hasan Ahmed: Mobile Computing, Technology, Applications and Service Creation, 2d Edition, Tata McGraw Hill, 2010.
2. Martyn Mallik: Mobile and Wireless Design Essentials, Wiley, 2003.
3. William Stallings, "Wireless Communications and Networks", 2nd ed., Prentice Hall of India / Pearson Education, 2007.

Reference Books:

1. Raj Kamal: Mobile Computing, Oxford University Press, 2007.
2. Iti Saha Misra: Wireless Communications and Networks, 3G and Beyond, Tata McGraw Hill, 2009.

Course Outcomes (COs):

At the end of the course, students should be able to:

1. Explain GSM, GPRS and Bluetooth software models for mobile computing. (PO-2,3,4,5,12, PSO-2,3)
2. Describe various wireless standards and its applications. (PO-2,3,4,5,12, PSO-2,3)
3. Describe the basic concepts and principles in mobile computing on different platforms. (PO-2,3,4,5,12, PSO-2,3)
4. Interpret client/ server architecture and various mobile operating systems. (PO-2,3,4,5,12, PSO-2,3)
5. Design mobile application using various Wireless and J2ME languages. (PO-2,3,4,5,12, PSO-2,3)

List of Lab Exercises:

1. To implement Code Division Multiple Access (CDMA)/
2. To study frequency reuse concept.
3. To study basic concept of J2ME.
4. To study various classes (such as Text Box, Choice Group , Drop Down menus etc.) and their implementation in J2ME.
5. To design a simple WML page using various WML tags.
6. To implement mobile network using NS2.
7. Study Assignment 1: Detailed study of Bluetooth.
8. Study Assignment 2: Detailed study of Wireless Application Protocol.
9. Set up and configuration of access point/
10. Study Assignment 3: To study network security software.

Computer Graphics and Visualization

Course Code: CSE08 **Credits: 3:0:1:0**
Prerequisites: Data structures **Contact Hours: 42L+28P**
Course Coordinator/s: Dr. D.S. Jayalakshmi

Course Contents:

Unit I

Introduction: Applications of computer graphics, A graphics system, Images: Physical and synthetic, Imaging Systems, The synthetic camera model, The programmer's interface, Graphics architectures, Programmable Pipelines Graphics Programming: Programming two-dimensional applications, OpenGL application programming interface, Primitives and attributes, color, viewing, control functions, the gasket program, polygons and recursions, the three dimensional gasket, adding interactions, menus.

Unit II

Geometric Objects and Transformations: Scalars, Points, and Vectors, Three-dimensional Primitives, Coordinate Systems and Frames, Modeling a Colored Cube, Affine Transformations, Rotation, Translation and Scaling, Transformation in Homogeneous Coordinates, Concatenation of Transformations, OpenGL Transformation Matrices, Spinning of cube, Interfaces to three-dimensional applications.

Unit III

Implementation: Basic Implementation Strategies, Four major tasks, Clipping, Line-segment clipping, Polygon clipping, Clipping of other primitives. Clipping in three dimensions, Rasterization: Bresenham's algorithm, Polygon Rasterization, Hidden-surface removal .

Unit IV

Viewing: Classical and computer viewing, Viewing with a Computer, Positioning of the camera, Parallel Projections, Perspective projections, Projections in OpenGL, Hidden-surface removal, Parallel-projection matrices, Perspective-projection matrices, Interactive Mesh Displays, Projections and Shadows.

Unit V

Lighting and Shading: Light and Matter, Light sources, The Phong reflection model, Polygon shading, Approximation of sphere by recursive subdivision, Specifying lighting parameters, Implementing a lighting model/

Text Book:

1. Edward Angel and Dave Shreiner: Interactive Computer Graphics - A Top-Down Approach with Shader-based OpenGL, 6th Edition, Pearson Education, 2011.

Reference Books:

1. Donald Hearn and Pauline Baker: Computer Graphics with OpenGL, 3rd Edition, Pearson Education, 2011.
2. F.S. Hill Jr.: Computer Graphics Using OpenGL, 3rd Edition, Pearson Education, 2009.
3. James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes: Computer Graphics, 2nd Edition, Pearson Education, 2011.

Course Outcomes (COs):

At the end of the course, students should be able to:

1. Understand the components of a graphics system with the building blocks & the overall architecture. (PO-1,2,3,4,5,12,PSO-2,3)
2. Derive the geometrical transformations used in interactive computer graphics in different coordinate systems and for viewing and projections. (PO-1,2,3,4,5,12,PSO-2,3)
3. Discuss the different algorithms for clipping and rasterization of lines and polygons, and for hidden surface removal. (PO-1,2,3,4,5,12,PSO-2,3)
4. Illustrate different lighting and shading models. (PO-1,2,3,4,5,12,PSO-2,3)
5. Implement 3D computer graphics applications in OpenGL using knowledge of display systems, image synthesis, and interactive control. (PO-1,2,3,4,5,12,PSO-2,3)

List of Lab Exercises:

1. Handling graphic primitives.
2. Sierpinski gasket.
3. Input interactions.
4. Modeling and transformation of a colored cube.
5. Isometric view of a cube.
6. Approximation of a sphere.
7. Implementation of Clipping algorithms.
8. Implementation of rasterization algorithms.
9. Program to fill any given polygon using scan-line area filling algorithm. (Use appropriate data structures.).
10. Lighting and shading.

Software Defined Networks

Course Code: CSE09 **Credits: 3:0:0:1**
Contact Hours: 42L
Prerequisites: Data communications and Computer networks
Course Coordinator/s: Sanjeetha R

Course Contents:

Unit I

Introduction - Traditional Switch Architecture.
Why SDN-Evolution of Switches and Control Planes, Cost, SDN Implications for Research and Innovation, Data Center Innovation, Data Center Needs.
The Genesis of SDN: Forerunners of SDN: Early efforts, Network Access Control, Orchestration, Virtualization manager network plugins, FORCES, 4D Centralized Network Control, Ethane, Software Defined Networking is Born, Network Virtualization.
How SDN Works - Fundamental Characteristics of SDN, SDN Operation, SDN Devices, SDN Controller, SDN Applications, Alternate SDN Methods.
Self-Study: Data Center Needs, Network Virtualization.

Unit II

How SDN Works Contd. -SDN Controller- SDN controller core modules, SDN controller interfaces, Existing controller implementations, potential issues with the SDN Controller, SDN Applications, Alternate SDN Methods – SDN via APIs, Benefits and Limitations of SDN via APIs, SDN via hypervisor based overlay networks.
The OpenFlow Specification–OpenFlow Overview – The OpenFlow switch, The OpenFlow Controller, The OpenFlow protocol, The Controller-switch secure channel, OpenFlow 1.0 and OpenFlow Basics, OpenFlow 1.1 Additions, OpenFlow 1.2 Additions, OpenFlow 1.3 Additions, OpenFlow Limitations.
Self-Study: Benefits and Limitations of SDN via APIs, OpenFlow Limitations.

Unit III

The OpenFlow Specification Contd. Openflow 1.4 additions – Bundles, Eviction and vacancy events, enhanced support for multiple controller, optical port support, and flow table synchronization.
Alternative Definitions of SDN - Potential Drawbacks of Open SDN, SDN via APIs – Legacy APIs in Network Devices, NETCONF/YANG, BGP-LS/PCE-P, REST, Examples of SDN via APIs, Ranking SDN via APIs, SDN via Hypervisor-Based Overlays – Overlay Controller, Overlay Operation, Examples of SDN via Hypervisor-Based Overlays, Ranking SDN via Hypervisor-Based Overlays, SDN via Opening Up the Device, Network Functions Virtualization, Alternatives Overlap and Ranking.
Self-Study: flow table synchronization, Alternatives Overlap and Ranking.

Unit IV

Emerging Protocol, Controller - Additional SDN Protocol Models, Using Existing Protocols to Create SDN Solutions, Using the Netconf Protocol, BGP Protocol, BGP-LS Protocol, PCE-P Protocol MPLS Protocol For SDN. Additional SDN Controller Models - Controllers with Multiple Southbound Plugins, Controllers Targeting Service Provider Solutions, Controllers Built For Scalability, Intents-Based Applications.

SDN in the Data Center-Data Center Demands – Overcoming Current Network Limitations, MAC address explosion, Number of VLANs, Spanning tree, adding, moving and deleting resources, Failure recovery, multitenancy, Tunneling Technologies for the Data Center.

Self-Study: Intents-Based Applications, Tunneling Technologies for the Data Center.

Unit V

SDN in the Data Center- Path Technologies in the Data Center, Ethernet Fabrics in the Data Center, SDN Use Cases in the Data Center, Comparison Of Open SDN, Overlays, and APIs.

SDN in Other Environments - Wide Area Networks, Service Provider and Carrier Networks, Campus Networks, Mobile Networks, Optical Networks.

Network Functions Virtualization - Definition Of NFV, What Can We Virtualize? SDN Vs NFV, When Should NFV Be Used With SDN?, In-Line Network Functions, SDN Applied To Server Load-Balancing, Firewalls and Intrusion Detection.

SDN Applications- Application Types, a Simple Reactive Java Application - Blacklisting Hostnames and IP Addresses, Offloading Flows in the Data Center.

Self-Study: Optical Networks, Firewalls and Intrusion Detection, Offloading Flows in the Data Center.

Self-Study Evaluation:

- The topics are integral part of the course.
- No formal lectures will be held for the self-study topics.
- The course co-ordinator may provide reading materials for self-study topics (optional).
- The topics prescribed under self-study in curriculum are part of CIE and SEE.

Text Book:

1. Paul Goransson, Chuck Black, and Timothy Culver: Software Defined Networks A Comprehensive Approach, Second Edition, Elsevier, 2014.

Reference Book:

1. Thomas D.Nadeau & Ken Gray: SDN Software Defined Networks O'Reilly publishers, Second edition, 2017.

Course Outcomes (COs):

At the end of the course, students should be able to:

1. Describe the fundamental characteristics of SDN. (PO-2,3,4,5,12,PSO-1,2,3)
2. Differentiate between various OpenFlow specifications. (PO-2,3,4,5,12,PSO-1,2,3)
3. Understand the design & implementation techniques for SDN. (PO-2,3,4,5,12,PSO-1,2,3)
4. Compare and contrast different types of controller models in SDN. (PO-2,3,4,5,12,PSO-1,2,3)
5. Illustrate use of SDN in Data Centers and other environments. (PO-2,3,4,5,12,PSO-1,2,3)

Soft Computing

Course Code: CSE10 **Credits: 4:0:0:0**
Prerequisites: Nil **Contact Hours:56L**
Course Coordinator/s: Nagbhushan A M

Course Contents:

Unit I

Introduction: Neural networks, Fuzzy logic, Genetic algorithms, Hybrid systems, Artificial Neural Networks: Fundamental concept, Evolution, Basic model of ANN, Important terminologies of ANN, MP neuron, Hebb Network.

Unit II

Supervised Learning Network: Perceptron Networks, Adaptive linear neuron, multiple adaptive linear neurons, Back propagation Network.

Unit III

Introduction to Fuzzy logic, classical sets and fuzzy sets: Classical sets, Fuzzy sets. Classical relations and fuzzy relations: Cartesian product of relation, Classical relation, Fuzzy relations, Tolerance and equivalence relations. Membership functions: Features, Fuzzification, methods of membership value assignments.

Unit IV

Defuzzification: Lambda-cuts for fuzzy sets, Lambda-cuts for fuzzy relations, Defuzzification methods. Fuzzy decision making: Individual, multi person, multi objective, multi attribute, and fuzzy Bayesian decision making.

Unit V

Genetic algorithms: Introduction, Basic operations, Traditional algorithms, Simple GA, General genetic algorithms, the schema theorem, Genetic programming, applications.

Text Book:

1. Principles of Soft computing, S N Sivanandam, Deepa S. N, Wiley, India, (Chapters 1, 2, 3(Up to 3.5), 7, 8, 9, 10, 13, 15 (up to 15.6 & 15.9,15,10).

Reference Book:

1. Neuro-fuzzy and soft computing, J.S.R. Jang, C T Sun, E Mizutani, PHI (EEE edition) ISBN: 978-81-203-2243-1.

Course Outcomes (COs):

At the end of the course, the student should be able to:

1. Describe various problems on artificial neural networks and familiarize all concepts on various networks and applications in them. (PO-2,3,4,5,12, PSO-2,3)
2. Identify the compositions of neural networks, perceptrons and other networks. (PO-1,2,3,4,5,12,PSO-2,3)
3. Examine various fuzzification techniques and practice them. (PO-1,2,3,4,5,12,PSO-2,3)
4. Design problems and obtain crisp values from fuzzy data using defuzzification. (PO-1,2,3,4,5,12,PSO-2,3)
5. Summarise various compositions and complexities of genetic algorithms. (PO-1,2,3,4,5,12,PSO-2,3)

Machine Learning

Course Code: CSE11 **Credits: 3:0:1:0**
Prerequisites: AI **Contact Hours:42L+28P**
Course Coordinator/s: Dr. S Rajarajeswari

Course Contents:

Unit I

Introduction

Learning – Types of Machine Learning – Supervised Learning – The Brain and the Neuron –
Design a Learning System – Perspectives and Issues in Machine Learning – Concept Learning Task – Concept Learning as Search – Finding a Maximally Specific Hypothesis – Version Spaces – Find S, Candidate Elimination Algorithm – Linear Discriminants – Perceptron – Linear Separability – Linear Regression.

Unit II

Linear Models

Multi-layer Perceptron – Going Forwards – Going Backwards: Back Propagation Error – Multilayer Perceptron in Practice – Examples of using the MLP – Overview – Deriving Back Propagation– Radial Basis Functions and Splines – Concepts – RBF Network – Curse of Dimensionality – Interpolations and Basis Functions – Support Vector Machines.

Unit III

Tree and Probabilistic Models

Learning with Trees – Decision Trees – Constructing Decision Trees – Classification and
Regression Trees – Ensemble Learning – Boosting – Bagging – Different ways to Combine
Classifiers – Probability and Learning – Data into Probabilities – Basic Statistics – Gaussian
Mixture Models – Nearest Neighbor Methods – Unsupervised Learning – K means Algorithms –Vector Quantization – Self Organizing Feature Map.

Unit IV

Dimensionality Reduction and Evolutionary Models

Dimensionality Reduction – Linear Discriminant Analysis – Principal Component Analysis – Factor Analysis – Independent Component Analysis – Locally Linear Embedding – Isomap – Least Squares Optimization – Evolutionary Learning – Genetic algorithms – Genetic Offspring: - Genetic Operators – Using Genetic Algorithms – Reinforcement Learning – Overview – Getting Lost Example – Markov Decision Process.

Unit V

Graphical Models

Markov Chain Monte Carlo Methods – Sampling – Proposal Distribution – Markov Chain Monte
Carlo – Graphical Models – Bayesian Networks – Markov Random Fields – Hidden Markov
Models – Tracking Methods.

Text Books:

1. Ethem Alpaydin "Introduction To Machine Learning" 2nd Edition PHI Learning Pvt. Ltd-New Delhi.
2. Tom M Mitchell, ―Machine Learning‖, First Edition, McGraw Hill Education, 2013.

Reference Books:

1. Peter Flach, ―Machine Learning: The Art and Science of Algorithms that Make Sense of Data‖, First Edition, Cambridge University Press, 2012.
2. Stephen Marsland, ―Machine Learning – An Algorithmic Perspective‖, Second Edition,
3. Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
4. Trevor Hastie, Robert Tibshirani, Jerome Friedman, h The Elements of Statistical Learning, 2nd edition, springer series in statistics.
5. Jason Bell, ―Machine learning – Hands on for Developers and Technical Professionals‖, FirstEdition, Wiley, 2014

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Characterize machine learning algorithms as supervised, unsupervised & semi supervised. (PO-1,2,3,5,12,PSO-1,2,3)
2. Understand machine learning techniques & computing environment that are suitable for the applications. (PO-1,2,3,4,5,12,PSO-1,2,3)
3. Analyze & evaluate algorithms for pattern classification. (PO-1,2,3,4,5,12,PSO-1,2,3)
4. Describe the multiple criteria for analyzing & improving efficiency of different ML algorithms. (PO-1,2,3,4,5,12,PSO-1,2,3)
5. Describe the computational analysis problems & implement with suitable ML algorithms. (PO-1,2,3,4,5,12,PSO-1,2,3)

List of Lab Exercises:

1. Introduction to python Numpy, scipyscikit and exercises.
2. Data sampling Visualization, Learning and Classification-Maximum margin classification, Classification errors, regularization, and logistic regression.
3. Simple Linear regression, estimator bias and variance, active learning.
4. Dynamic Version spaces in ML.
5. Bayesian Decision Exercises.
6. Logistic, Multivariate logistic Regression.
7. Support vector machine (SVM) and kernels, kernel optimization.
8. Pattern Recognition and Applications.
9. K-Means Clustering & PCA.
10. Time series analysis.
11. Missing value updation, Error correction.
12. Elements of Linear discriminant functions.
13. Anomaly detection and recommendation.
14. Clustering, spectral clustering, Markov models.

Natural Language Processing

Course Code: CSE12 **Credits: 3:0:1:0**
Prerequisites: AI **Contact Hours: 42L+28P**
Course Coordinator/s: Dr. Jagdish S Kallimani

Course Contents:

Unit I

Introduction: Knowledge in Speech and Language Processing, Ambiguity, Models and Algorithms; Language, Thought, and Understanding; The State of the Art and The Near-Term Future; Regular Expressions and Automata; Morphology and Finite-State Transducers: Lexicon-free FSTs: The Porter Stemmer, Human Morphological Processing.

Unit II

N-grams: Counting Words in Corpora, Smoothing, N-grams for Spelling and Pronunciation, Entropy; Word Classes and Part-of-Speech Tagging: Part-of-Speech Tagging , Rule-based Part-of-speech Tagging, Stochastic Part-of-speech Tagging, Transformation-Based Tagging; Context-Free Grammars for English: Constituency, Context-Free Rules and Trees, Sentence-Level Constructions, The Noun Phrase.

Unit III

Parsing with Context-Free Grammars: The Earley Algorithm; Features and Unification: Feature Structures, Unification of Feature Structures, Features Structures in the Grammar, Implementing Unification, Parsing with Unification Constraints; Lexicalized and Probabilistic Parsing: Probabilistic Context-Free Grammars, Problems with PCFGs.

Unit IV

Representing Meaning: First Order Predicate Calculus, Some Linguistically Relevant Concepts, Related Representational Approaches, Alternative Approaches to Meaning; Semantic Analysis : Syntax-Driven Semantic Analysis, Attachments for a Fragment of English; Lexical Semantics : Relations Among Lexemes and Their Senses, WordNet: A Database of Lexical Relations, The Internal Structure of Words.

Unit V

Discourse: Reference Resolution, Text Coherence, Discourse Structure; Generation: Introduction to Language Generation, An Architecture for Generation; Machine Translation: Language Similarities and Differences, The Transfer Metaphor.

Text Book:

1. Daniel Jurafsky and James H Martin, "Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", Prentice Hall, 2nd Edition, 2008.

Reference Book:

1. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval", Oxford University Press, 2008.

Course Outcomes (COs):

At the end of the course, the students should be able to:

1. Interpret how speech and language technology relies on formal models to capture knowledge, and language processing deals with subparts of words (morphology). (PO-1,2,3,4,5,12, PSO-2,3)
2. Illustrate the way *N*-gram tool is used for spelling and pronunciation processing, and part-of-speech tagging mechanism using various categories. (PO-2,3,4,5,12, PSO-2,3)
3. Describe feature structures and unification operation which is used to combine them, and probabilistic parsing to capture more syntactic information. (PO-2,3,4,5,12, PSO-2,3)
4. Outline representations used to bridge the gap from language to commonsense Knowledge (semantic processing), and meanings associated with lexical items. (PO-2,3,4,5,12, PSO-2,3)
5. Emphasize problems that NLP systems face, natural language outputs construction from non-linguistic inputs and machine translation framework approaches. (PO-2,3,4,5,12, PSO-2,3)

List of Lab Exercises:

The students are expected to get the proficiency in solving problems in the laboratory on:

1. Writing Regular Expressions.
2. Constructing Context-Free Grammars.
3. Compute Probabilities for Individual Words of a Sentence.
4. Finite-State Automaton fora Dialogue Manager.
5. Identification and Disambiguation of Polysemy Words and Others.

Information Retrieval

Course Code: CSE13 **Credits: 3:0:1:0**
Prerequisites: Nil **Contact Hours: 42L+28P**
Course Coordinator/s: Vandana Sardar

Course Contents:

Unit I

Introduction: Overview, History of IR, Text Operations: Document pre-processing, Document Clustering, Text Compression, Indexing: Inverted files, Mathematics for IR: Set Theory, Mathematical Logic, Probability and Linear algebra, Classic IR Models: Boolean Model, Vector space model: tf-idf weighing, Probabilistic Model. Language models for IR: The language model and the query likelihood model.

Unit II

Evaluation Measures: Precision, Recall, Alternative Measures, Reference Collections: TREC, **Relevance Feedback and Query Expansion, Text Classification**: The text classification problem, **Flat clustering:** Clustering in information retrieval, Problem Statement, **Hierarchical clustering:** Hierarchical agglomerative clustering,, Single-link and Complete-link clustering.

Unit III

String Matching algorithms: Knuth Morris Pratt and Rabin Karp, **Stemming algorithm:** Porter, **Map reduce algorithms:** tf- idf calculation and indexing**, Classification:** Naive Bayes algorithm, **Clustering:** k-means algorithm. **Machine learning Algorithms:** Learning to rank: Point wise, Pair wise and List wise approaches.

Unit IV

Web search basics: web characteristics**, Architecture of web search engine, Web Crawling and Indexing:** Overview, Crawling, Distributing Indexes, Connectivity, Servers. **Link analysis:** Web as a graph**,** Page Rank, Hubs and Authorities.

Unit V

XML Retrieval: Basic XML Concepts, Challenges in XML retrieval, a vector space model for XML retrieval, **Introduction to Semantic Web:** Purpose, Semantic Web Stack, RDF, RDFS, Ontology, Web ontology language (OWL) and ontology tools**.**

Text Books/ Reference Books:

1. Ricardo Baeza-Yates, Berthier Ribeiro-Neto: Modern Information Retrieval, Pearson Education, 1999.
2. William B Frakes, Ricardo Baeza Yates: Information Retrieval Data Structures and Algorithms, PH PTR, 1992.
3. David A Grossman, OphirFrieder: Information Retrieval Algorithms and Heuristics, 2e, Springer, 2004.

Online Books:

1. Introduction to Information Retrieval. C.D. Manning, P. Raghavan, H. Schütze. Cambridge UP, 2008.
2. Mathematics for Classical Information Retrieval: Roots and Applications: Dariush Alimohammadi, http://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1002&context=zeabook.
3. http://www.dcc.fc.up.pt/~zp/aulas/1213/pde/geral/bibliografia/MIT.Press.A.Semantic.Web.Primer.eBook-TLFeBOOK.pdf.

Course Outcomes (COs):

At the end of the course, student should be able to

1. Understand different components of an information retrieval system. (PO-2,3,4,5,12,PSO-2,3)
2. Understand the common algorithms & techniques for information retrieval (Document indexing, Query processing) (PO-2,3,4,5,12,PSO-2,3)
3. Describe the use of quantitative evaluation methods for IR systems. (PO-2,3,4,5,12,PSO-2,3)
4. Implement techniques employed by web search engines for IR systems. (PO-2,3,4,5,12,PSO-2,3)
5. Illustrate the techniques of semantic web & XML retrieval for IR systems. (PO-2,3,4,5,12,PSO-2,3)

List of Lab Exercises:

1. Implementation of string matching algorithms such as KMP and Rabin Karp.
2. Implementation of Construction of Inverted index.
3. Implementation of tf-idf calculations for sample text documents.
4. Implementation of Naïve Bayes Algorithm.
5. Implementation of K- means clustering algorithm.
6. Implementation of Map –Reduce for word count problem.
7. Implementation of a Web Crawler.
8. Exploring various ontology tools.

UNIX System Programming and Compiler Design Laboratory

Course Code: CSL66 **Credits: 0:0:1:0**
Prerequisites: Nil **Contact Hours: 28P**
Course Coordinator/s: Mallegowda M

Course Contents:

Part A

1. Basic file I/O functions & properties of a file.
2. File Types, File access permission and File links.
3. Creating the process and process accounting.
4. Feature provided by different signal implementation.
5. Coding rules and Characteristics of Daemon Process

Part B

1. Tokenization of input
2. Validating the syntax of the input
3. Performing syntax directed translation
4. Verification of semantic
5. Generation of intermediate code
6. Optimization of code
7. Generation of assembly language code

Text Books:

1. W. Richard Stevens: Advanced Programming in the UNIX Environment, Second Edition, Pearson education, 2011.
2. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman:Compilers-Principles, Techniques and Tools, 2nd Edition, Addison-Wesley, 2007.

Reference Books:

1. Terrence Chan: UNIX System Programming Using C++, First edition, Prentice Hall India, 2011.
2. Kay A Robbins and Steve Robbins: Unix Systems Programming, First Edition, Pearson Education, 2009.
3. Marc J. Rochkind: Advanced UNIX Programming, 2nd Edition, Pearson
4. Kenneth C Louden: Compiler Construction - Principles & Practice, Brooks/Cole, CENGAGE learning, 1997.
5. Andrew W Appel: Modern Compiler Implementation in C, Cambridge University Press, 1999.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Demonstrate the advanced Unix operating system concepts & analyze the different concepts of tokens in a compiler. (PO-2,3,4,5,12,PSO-1,2,3)
2. Create programs for code generation & code optimization utilizing the special commands in Unix. (PO-2,3,4,5,12,PSO-1,2,3)
3. Apply the advanced Unix programming to different compiler sorting tools in different phases. (PO-2,3,4,5,12,PSO-1,2,3)

IOT/Embedded Systems Laboratory

Course Code: CSL67 **Credits: 0:0:1:0**
Prerequisites: Microprocessors/Microcontrollers **Contact Hours: 28P**
Course Coordinator/s: Dr. Diwakar Harekal

Course Contents:

1. Study of simple GPIO programs to use the ports.
2. Study use of interrupts and peripherals like LCD 16x2.
3. Study of ADC programs Keyboard, seven segments.
4. Study of ADC programs and graphical LCD 128x64.
5. Study of UART programming.
6. Study of enabling the I2C with LCD 16x2.
7. Study of enabling the I2C.
8. Study use of PWM with DC motor/ servo motor.
9. Study of Node MCU wireless features.
10. Use of SPI with LCD graphical 128X64.

Text Books:

1. Wayne Wolf "Computers as Components Principles of Embedded Computer System Design", Second Edition, Elsevier, 2008.
2. Joseph Yiu, "The Definitive Guide to the ARM Cortex-M0 ", 1st edition, Newness – an imprint of Elsevier, 2011.
3. Lyla B. Das, "Embedded Systems an integrated approach "1st edition, Pearson, 2013.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Investigate a variety of devices & technologies in IoT systems. (PO-1 to 7, 12, PSO1-3)
2. Design & implement programs using various ICS & micro controllers for IoT& embedded systems. (PO-1 to 7, 12, PSO1-3)
3. Understand the characteristics of real time by implementing programs to interface memory, I/O with processor. (PO-1 to 7, 12, PSO1-3)

Web Technologies Laboratory

Course Code: CSL68 **Credits: 0:0:1:0**
Prerequisites: Nil **Contact Hours: 28P**
Course Coordinator/s: Dr. J Geetha

Course Contents:

There shall be a minimum of 2 exercises conducted on each of the following topics:

1. HTML5
2. Java Script
3. Advanced java Script
4. Inheritance In java Script
5. Scopes and Closures
6. Develop a Server side programming using java Script : Node.Js
7. Implement CRUD Operation using MongoDB and Node.Js
8. Develop an Client side Application using Angular Js
9. Implement MEAN Stack.

Reference Books:

1. Web Application Design and Implementation: Apache 2, PHP5, MySQL, JavaScript, and Linux/UNIX Steven A. Gabarro, December 2006, ©2007, Wiley-IEEE Computer Society Press.
2. Nate Murray, Felipe Coury, Ari Lerner and Carlos Taborda, "ng-book, The Complete Book on Angular 4" September 2016 3. Krasimir Tsonev, "Node.js by Example Paperback", May 2015.
3. Web link for Angular4.0: https://angular.io/
4. Web link for Node.js : https://nodejs.org/en/
5. Web link for MongoDB: https://www.mongodb.com

Course Outcomes (COs):

At the end of the course the student will be able to:

1. Develop web pages with various media contents using HTML5. (PO-2,3,4,5,10, 12,PSO-1,2,3)
2. Create a robust Client side validation with java script. (PO-2,3,4,5,10, 12,PSO-1,2,3)
3. Design dynamic data-driven Web sites using MongoDB and Node.js (PO-2,3,4,5,10, 12,PSO-1,2,3)